

THE MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION



ISSN 2687-0517

---

---

# **Computing, Telecommunications and Control**

---

---

**Vol. 18, No. 1  
2025**

Peter the Great St. Petersburg  
Polytechnic University  
2025

# COMPUTING, TELECOMMUNICATIONS AND CONTROL

## EDITORIAL COUNCIL

Prof. Dr. *Dmitry G. Arseniev* corresponding member of RAS, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Vladimir V. Voevodin* corresponding member of RAS, Lomonosov Moscow State University, Russia;  
Prof. Dr. *Vladimir S. Zaborovsky*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Dmitry P. Zegzhda*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Vladimir N. Kozlov*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Assoc. Prof. Dr. *Ivan S. Mukhin*, Alferov University, St. Petersburg, Russia;  
Prof. Dr. *Igor G. Chernorutsky*, Peter the Great St. Petersburg Polytechnic University, Russia.

## EDITORIAL BOARD

### Editor-in-chief

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

### Members:

Assoc. Prof. Dr. *Pavel D. Drobintsev*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Assoc. Prof. Dr. *Vladimir M. Itsyson*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Philippe Ferrari*, Grenoble Alpes University, France;  
Prof. Dr. *Yevgeni Koucheryavy*, Tampere University of Technology, Finland;  
Prof. Dr. *Wolfgang Krauschneider*, Hamburg University of Technology, Germany;  
Prof. Dr. *Fa-Long Luo*, University of Washington, USA;  
Prof. Dr. *Sergey B. Makarov*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Emil Novakov*, Grenoble Alpes University, France;  
Prof. Dr. *Nikolay N. Prokopenko*, Don State Technical University, Russia;  
Prof. Dr. *Mikhail G. Putrya*, National Research University of Electronic Technology, Russia;  
Sen. Assoc. Prof. Dr. *Evgeny Pyshkin*, University of Aizu, Japan;  
Prof. Dr. *Viacheslav P. Shkodyrev*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Vladimir A. Sorotsky*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Igor A. Tsikin*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Sergey M. Ustinov*, Peter the Great St. Petersburg Polytechnic University, Russia;  
Prof. Dr. *Lev V. Utkin*, Peter the Great St. Petersburg Polytechnic University, Russia.

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

Open access journal is to publish articles of a high scientific level covering advanced experience, research results, theoretical and practical problems of informatics, electronics, telecommunications, and control.

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

The journal is registered with the Federal Service for Supervision in the Sphere of Telecom, Information Technologies and Mass Communications (ROSKOMNADZOR). Certificate ЭЛ No. ФС77-77378 issued 25.12.2019.

Editorial office

Dr. Sc., Professor A.S. Korotkov – Editor-in-Chief;

Ph.Ch.S. Bastian – literary editor, proofreader; G.A. Pyshkina – editorial manager; A.A. Kononova – computer layout; I.E. Lebedeva – English translation.

Address: 195251 Polytekhnikeskaya Str. 29, St. Petersburg, Russia.

+7 (812) 552-6216, e-mail: infocom@spbstu.ru

Release date: 31.03.2025

© Peter the Great St. Petersburg Polytechnic University, 2025

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ



ISSN 2687-0517

---

---

# **Информатика, телекоммуникации и управление**

---

---

**Том 18, № 1  
2025**

Санкт-Петербургский политехнический  
университет Петра Великого  
2025

# ИНФОРМАТИКА, ТЕЛЕКОММУНИКАЦИИ И УПРАВЛЕНИЕ

## РЕДАКЦИОННЫЙ СОВЕТ ЖУРНАЛА

*Арсеньев Д.Г.*, чл.-кор. РАН, д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Россия; *Воеводин В.В.*, чл.-кор. РАН, Московский государственный университет им. М.В. Ломоносова, Москва, Россия; *Заборовский В.С.*, д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Россия; *Зегжда Д.П.*, чл.-кор. РАН, д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Россия; *Козлов В.Н.*, д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Россия; *Мухин И.С.*, д-р физ.-мат. наук, доцент, Санкт-Петербургский национальный исследовательский Академический университет им. Ж.И. Алферова Российской академии наук, Санкт-Петербург, Россия; *Черноруцкий И.Г.*, д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Россия.

## РЕДАКЦИОННАЯ КОЛЛЕГИЯ ЖУРНАЛА

### Главный редактор

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

### Редакционная коллегия:

*Дробинцев П.Д.*, канд. техн. наук, доцент, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Россия;

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

*Феррари Ф.*, профессор, Университет Гренобль-Альпы, Гренобль, Франция;

*Краутишнайдер В.*, профессор, Гамбургский технический университет, Гамбург, Германия;

*Кучерявый Е.А.*, канд. техн. наук, профессор, Университет Тампере, Финляндия.

*Люо Ф.-Л.*, University of Washington, Washington, USA;

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

*Новаков Э.*, профессор, Университет Гренобль-Альпы, Гренобль, Франция;

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

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

*Пышкин Е.В.*, профессор, Университет Айзу, Айзу-Вакаматсу, Япония;

*Сороцкий В.А.*, д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Россия;

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

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

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

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

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

Сетевое издание открытого доступа публикует статьи высокого научного уровня, освещающие передовой опыт, результаты НИР, теоретические и практические проблемы информатики, электроники, телекоммуникаций, управления.

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

Журнал зарегистрирован Федеральной службой по надзору в сфере информационных технологий и массовых коммуникаций (Роскомнадзор). Свидетельство о регистрации Эл № ФС77-77378 от 25.12.2019.

Учредитель и издатель: Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Российская Федерация.

Редакция журнала

д-р техн. наук, профессор А.С. Коротков – главный редактор;

Ф.К.С. Бастиан – литературный редактор, корректор; Г.А. Пышкина – ответственный секретарь, выпускающий редактор;

А.А. Кононова – компьютерная вёрстка; И.Е. Лебедева – перевод на английский язык.

Адрес редакции: Россия, 195251, Санкт-Петербург, ул. Политехническая, д. 29.

Тел. редакции +7(812) 552-62-16, e-mail редакции: infocom@spbstu.ru

Дата выхода: 31.03.2025

© Санкт-Петербургский политехнический университет Петра Великого, 2025

# Contents

## Intelligent Systems and Technologies, Artificial Intelligence

|  |    |
|--|----|
| <b>Ageev A.Yu., Utkin L.V., Konstantinov A.V.</b> Improved anomaly detection by using the attention-based isolation forest with trainable scoring function .....               | 7  |
| <b>Sabutkevich A.M., Nikiforov I.V., Samochadin A.V.</b> A method for modeling of individual agent behavior in the process-network paradigm of discrete-event simulation ..... | 23 |
| <b>Alekseev E.A., Lomanov A.N., Ivanov D.S.</b> A technique for automated analysis of the blade surface for defects under UV light .....                                       | 36 |
| <b>Shulgin S.A., Benderskaya E.N.</b> Leveraging natural language processing techniques for enhanced recommender systems .....   | 48 |

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

|  |    |
|--|----|
| <b>Aliyev A.A., Molodyakov S.A.</b> ResNet-SV: Fast and accurate speaker verification with a multi-layer cascade attention mechanism ..... | 60 |
| <b>Pham H.D., Sorotsky V.A.</b> Characteristics of Class E power amplifier with complex impedance load .....                               | 72 |
| <b>Ivanov N.V., Romyancev I.A.</b> Monolithic microwave bandpass filters design for S- and C- bands ....                                   | 85 |

## Software and Hardware of Computer, Network, Telecommunication, Control, and Measurement Systems

|  |    |
|--|----|
| <b>Tulaev A.T., Kozlov A.S., Kostygov D.V., Kuznecov K.P., Belyaev Ya.V., Loboda V.V.</b> A micromechanical pressure sensor with reconfigurable ASIC ..... | 98 |
|--|----|

## Computer Simulations of Telecommunication, and Control Systems

|  |     |
|--|-----|
| <b>Khrustaleva I.N., Shkodyrev V.P., Khokhlovskiy V.N., Chernyh L.G., Stepanov S.N.</b> Optimization model of the processing parameters for structural elements of a product ..... | 111 |
| <b>Novak D.A., Nushtaev N.A., Kozhubaev Yu.N.</b> Technological process control of oil-based gas absorption .....  | 130 |



# Содержание

## **Интеллектуальные системы и технологии, искусственный интеллект**

- Агеев А.Ю., Уткин Л.В., Константинов А.В.** Улучшенное обнаружение аномалий с помощью леса изоляции на основе внимания с обучаемой функцией оценки ..... 7
- Сабуткевич А.М., Никифоров И.В., Самочадин А.В.** Метод симуляции индивидуального поведения агентов в процессно-сетевой парадигме дискретно-событийного моделирования .... 23
- Алексеев Е.А., Ломанов А.Н., Иванов Д.С.** Методика автоматизированного анализа поверхности лопатки на наличие дефектов под УФ-светом ..... 36
- Шульгин С.А., Бендерская Е.Н.** Использование методов обработки естественного языка в продвинутых рекомендательных системах ..... 48

## **Устройства и системы передачи, приема и обработки сигналов**

- Алиев А.А., Молодяков С.А.** ResNet-SV: Быстрая и точная верификация спикера с использованием многоуровневого каскадного механизма внимания ..... 60
- Фам Х.Д., Сороцкий В.А.** Характеристики усилителя мощности класса E при работе на комплексную нагрузку ..... 72
- Иванов Н.В., Румянцев И.А.** Разработка монолитных интегральных схем полосовых СВЧ фильтров S- и C-диапазонов ..... 85

## **Компьютерные сети, вычислительные, телекоммуникационные, управляющие и измерительные системы**

- Тулаев А.Т., Козлов А.С., Костыгов Д.В., Кузнецов К.П., Беяев Я.В., Лобода В.В.** МЭМС датчик давления с реконфигурируемой интегральной схемой ..... 98

## **Моделирование вычислительных, телекоммуникационных и управляющих систем**

- Хрусталева И.Н., Шкодырев В.П., Хохловский В.Н., Черных Л.Г., Степанов С.Н.** Модель оптимизации параметров обработки структурных элементов изделия ..... 111
- Новак Д.А., Нуштаев Н.А., Кожубаев Ю.Н.** Управление технологическим процессом масляной абсорбции газа ..... 130

# Intelligent Systems and Technologies, Artificial Intelligence

## Интеллектуальные системы и технологии, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

Research article

DOI: <https://doi.org/10.18721/JCSTCS.18101>

UDC 004.85



### IMPROVED ANOMALY DETECTION BY USING THE ATTENTION-BASED ISOLATION FOREST WITH TRAINABLE SCORING FUNCTION

*A.Yu. Ageev* ✉, *L.V. Utkin*, *A.V. Konstantinov*

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

✉ [andreyageev1@mail.ru](mailto:andreyageev1@mail.ru)

**Abstract.** This paper proposes a novel anomaly detection model, called Attention-Based Isolation Forest with trainable Scoring Function (ABIF-SF). ABIF-SF enhances the original isolation forest algorithm by incorporating attention weights determined by scoring functions whose parameters are trained using gradient descent. The attention weights indicate the relevance of each data instance to the anomaly assessment task for each tree in the isolation forest. Two scoring functions are explored – scaled dot product and additive attention. Numerical experiments on real-world datasets demonstrate that ABIF-SF achieves better anomaly detection performance compared to isolation forest and attention-based isolation forest with the contamination model. The proposed method simplifies the computation of attention weights by using scoring functions and hinge loss optimization. The code implementation of ABIF-SF has been made publicly available for further research and benchmarking. Overall, the incorporation of trainable scoring functions to compute context-aware attention weights improves isolation forests for anomaly detection tasks.

**Keywords:** anomaly detection, attention mechanism, isolation forest, Nadaraya–Watson regression, quadratic programming, contamination model, additive attention

**Acknowledgements:** The research is partially financially supported by the Ministry of Science and Higher Education of the Russian Federation within the framework of the state assignment “Development and research of machine learning models for solving fundamental problems of artificial intelligence for the fuel and energy complex” (FSEG-2024-0027).

**Citation:** Ageev A.Yu., Utkin L.V., Konstantinov A.V. Improved anomaly detection by using the attention-based isolation forest with trainable scoring function. *Computing, Telecommunications and Control*, 2025, Vol. 18, No. 1, Pp. 7–22. DOI: 10.18721/JCSTCS.18101

Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18101>

УДК 004.85



## УЛУЧШЕННОЕ ОБНАРУЖЕНИЕ АНОМАЛИЙ С ПОМОЩЬЮ ЛЕСА ИЗОЛЯЦИИ НА ОСНОВЕ ВНИМАНИЯ С ОБУЧАЕМОЙ ФУНКЦИЕЙ ОЦЕНКИ

*А.Ю. Агеев* ✉, *Л.В. Уткин*, *А.В. Константинов*

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

✉ [andreyageev1@mail.ru](mailto:andreyageev1@mail.ru)

**Аннотация.** В данной статье предлагается новая модель обнаружения аномалий, называемая лесом изоляции на основе внимания с обучаемой функцией оценки (Attention-Based Isolation Forest with trainable Scoring Function, ABIF-SF). ABIF-SF улучшает исходный алгоритм леса изоляции, включая веса внимания, определяемые функциями оценки, параметры которых обучаются с помощью градиентного спуска. Веса внимания указывают на релевантность каждого экземпляра данных для задачи оценки аномалии для каждого дерева в лесу изоляции. Исследуются две функции оценки – масштабированное скалярное произведение и аддитивное внимание. Численные эксперименты на реальных наборах данных показывают, что ABIF-SF достигает лучшей производительности обнаружения аномалий по сравнению с лесом изоляции и лесом изоляции на основе внимания с моделью загрязнения. Предложенный метод упрощает вычисление весов внимания за счет использования функций оценки и оптимизации потерь шарнира. Реализация кода ABIF-SF была сделана общедоступной для дальнейших исследований и сравнительного анализа. В целом, включение обучаемых функций оценки для вычисления весов внимания с учетом контекста улучшает леса изоляции для задач обнаружения аномалий.

**Ключевые слова:** обнаружение аномалий, механизм внимания, лес изоляции, регрессия Надарая–Уотсона, квадратичное программирование, модель загрязнения, аддитивное внимание

**Финансирование:** Исследование выполнено при частичной финансовой поддержке Министерства науки и высшего образования Российской Федерации в рамках государственного задания «Разработка и исследование моделей машинного обучения для решения фундаментальных задач искусственного интеллекта в топливно-энергетическом комплексе» (FSEG-2024-0027).

**Для цитирования:** Ageev A.Yu., Utkin L.V., Konstantinov A.V. Improved anomaly detection by using the attention-based isolation forest with trainable scoring function // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 7–22. DOI: 10.18721/JCSTCS.18101

### Introduction

Anomalies are objects or events that significantly differ from normal or expected objects or events [1]. Anomalies can occur for various reasons, such as measurement errors, malicious attacks, equipment malfunctions or rare natural phenomena [2].

There are several classifications of anomalies in data. One of them is based on five dimensions: data type, relationship cardinality, anomaly level, data structure, and data distribution [3]. These dimensions lead to three broad groups of anomalies: point, collective, and contextual [3].

Anomaly detection is the process of identifying and detecting such anomalous data [4–7]. Anomaly detection is a challenging task due to the high dimensionality of data, noise, and heterogeneous distributions [5].



Depending on the type of anomalies, there are different detection methods that take into account the data characteristics. Point anomalies are individual data points that deviate significantly from the rest of the data in the set [6].

Point anomaly detection is the task of detecting such points and labeling them as anomalous or normal. Point anomaly detection can be useful in many cases, such as detecting malfunctions in industrial systems [7], identifying fraud in financial transactions [8], determining unusual user behavior in cybersecurity [9], etc.

Collective anomalies are those where a single data object in isolation appears normal. However, if it is considered in relation to other data objects or a subset of data objects, the object appears anomalous [6, 10].

Contextual anomalies are those that depend on the situation or context. Anomaly is determined based on certain conditions or rules. For example, high air temperature may be normal in summer but anomalous in winter [11].

There are many methods for point anomaly detection, which can be divided into three main groups: statistical methods based on probability distribution of data [12–14]; methods based on measuring proximity or distance between data points [15–17]; and density-based methods based on estimating the density of data in local neighborhoods [18–20]. Classical methods for detecting such anomalies include the Z-score, Tukey's test, and Grubb's test for statistical approach; k-nearest neighbors and local outlier factor for density-based approach; k-means and DBSCAN for clustering approach; isolation forest for isolation approach, which constructs random decision trees for separating normal and anomalous objects [21–25]. These methods work well for small and simple data but have their limitations. For example, they are sensitive to parameter selection, do not consider dependencies between features or temporal structure of data, and are not capable of generalizing to new types of anomalies.

Various machine learning (ML) methods can also be used for anomaly detection, which play an important role in this field. Depending on the presence or absence of class labels for normal and anomalous objects, ML methods can be divided into three types: supervised, semi-supervised, and unsupervised [26]. Supervised methods require enough examples for each class and are suitable for classification or regression tasks. Semi-supervised methods use only examples from one class (usually normal) and are suitable for one-class learning or generating new examples.

More modern methods use deep learning for detecting anomalies in complex and large data. They are based on constructing a model of normal data behavior using different neural network architectures: autoencoders, generative adversarial networks, recurrent networks, convolutional networks and others [27–31]. These methods have their own features and improvements compared to classical methods. For example, they can detect complex dependencies in data, working with weakly annotated or unlabeled data altogether.

Recently, attention-based methods have started to gain popularity, which allows models to focus on the most important parts of the data [32–36].

Attention weights are numerical coefficients that determine the degree of relevance of each data element to the task at hand. The use of attention weights can improve the quality of anomaly detection by providing a more accurate representation of the data and taking context into account [37]. Attention weights can be applied to various types of data, such as images, text, or time series [5, 37]. In the context of anomaly detection, the attention mechanism can be used to highlight the features or subsequences of data that are most relevant to determining the normality or abnormality of an object or event. Thus, the attention mechanism can help the model better understand the structure and dynamics of the data and increase the accuracy and efficiency of anomaly detection.

This article introduces a new method, ABIF-SF, based on isolation forest algorithm, which improves anomaly detection by incorporating a scoring function with trained weights in the attention mechanism. The attention weight computation process is simplified using gradient descent and hinge loss function.

The effectiveness of ABIF-SF is demonstrated through numerical experiments on real datasets and shows promising results.

Our contributions are:

1. We propose ABIF-SF, a novel anomaly detection method that enhances isolation forests through an attention mechanism implemented as a trainable scoring function. This allows the model to learn contextual weights indicating the relevance of different regions of the isolation forest for assessing anomalies.
2. A simplified optimization approach for computing attention weights based on gradient descent and the hinge loss function is introduced. Avoiding more complex contamination models streamlines training.
3. Demonstration of the effectiveness of ABIF-SF through numerical experiments on real datasets, which showed promising results.

## Related works

### *Approaches to anomaly detection*

Anomaly detection, a critical and well-explored problem across various domains, has seen significant advancements through deep learning techniques. Key methods include self-supervised learning [38], One-Class Classification (OCC) [39], time series anomaly detection [40], and domain-specific deep learning-based techniques [41]. Additionally, the use of deep learning for log file anomaly detection [42], GAN-based methods [43], video anomaly detection [44], and medical imaging [45] highlight the diversity of applications in this field.

### *Anomaly detection using the attention mechanism*

The attention mechanism, vital for emphasizing significant data features in anomaly detection, originated in text translation models and has since expanded to other data types and tasks [5, 36, 37, 46, 47]. Significant works include anomaly detection in semiconductor production using GANs with attention [46], attention-based deep learning for vector magnetic field anomalies [47], and graph-based anomaly detection leveraging attention mechanisms [36].

### *iForest and its variations*

The Isolation Forest (iForest) algorithm [23], known for its efficiency in large datasets, identifies anomalies based on the ease of isolation in binary trees. Despite its popularity, iForest faces limitations like feature correlation ignorance and potential normal sample misclassification [48]. To address these, enhancements such as local anomaly detection through k-means [49], the k-means-based iForest [50], and the minimum spanning tree-based approach [51] have been proposed.

### *Attention-Based iForest (ABIForest)*

Building on the concept of iForest, ABIForest (ABIF) [32] incorporates an attention mechanism through Nadaraya–Watson regression to refine anomaly detection. This method, inspired by the ABRF model [52], requires careful parameter tuning for both the attention mechanism and the iForest component.

## Preliminaries

### *Attention Mechanism as Nadaraya–Watson Regression*

The attention mechanism prioritizes relevant elements in input data for specific tasks [53, 54], using the softmax function for weight calculation. Given a vector  $\mathbf{z} = (z_1, \dots, z_n)$  the softmax function is:

$$\text{softmax}(z_j) = \frac{e^{z_j}}{\sum_j e^{z_j}}, \quad (1)$$

where each  $z_i$  is an element of the vector, and the denominator normalizes the sum of weights to 1.

Nadaraya–Watson regression [55, 56] uses weighted averages for prediction. Let  $\mathbf{x}_i \in \mathbb{R}^d$  be the  $i$ -th data point,  $y_i \in \mathbb{R}$  – its value, and  $w_i(\mathbf{x})$  – the weight based on its proximity to target  $\mathbf{x}$ . The regression formula is:

$$\hat{y}(\mathbf{x}) = \frac{\sum_{i=1}^n w_i(\mathbf{x}) y_i}{\sum_{i=1}^n w_i(\mathbf{x})}, \quad (2)$$

where  $\hat{y}(\mathbf{x})$  is the predicted value. The weights represent an attention mechanism, where the kernel function determines the similarity between query  $\mathbf{x}$  and keys  $\mathbf{x}_i$ .

The attention weight  $\alpha(\mathbf{x}, \mathbf{x}_i)$  is given by:

$$\alpha(\mathbf{x}, \mathbf{x}_i) = \frac{w(\mathbf{x}, \mathbf{x}_i)}{\sum_{i=1}^n w(\mathbf{x}, \mathbf{x}_i)}, \quad (3)$$

for a Gaussian kernel with parameter  $\omega$ :

$$\alpha(\mathbf{x}, \mathbf{x}_i) = \sigma\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{\omega}\right), \quad (4)$$

where  $\sigma$  is the softmax function, and the expression within is the scoring function  $\alpha(\mathbf{x}, \mathbf{x}_i)$ .

#### **Attention Scoring Function**

Scoring functions calculate relevance weights in the attention mechanism [54, 57]. The additive scoring function, for vectors  $\mathbf{q} \in \mathbb{R}^k$  and  $\mathbf{k} \in \mathbb{R}^k$ , is:

$$\alpha(\mathbf{q}, \mathbf{k}) = \mathbf{w}^T \tanh(\mathbf{W}_q \mathbf{q} + \mathbf{W}_k \mathbf{k}), \quad (5)$$

where  $\mathbf{w} \in \mathbb{R}^m$ ,  $\mathbf{W}_q \in \mathbb{R}^{m \times k}$ , and  $\mathbf{W}_k \in \mathbb{R}^{m \times k}$  are weight matrices and vectors.

The dot product with scaling, for vectors of dimensionality  $k$ , is:

$$\alpha(\mathbf{q}, \mathbf{k}) = \frac{\mathbf{q}^T \cdot \mathbf{k}}{\sqrt{k}}. \quad (6)$$

These functions use softmax to assign weights:

$$\text{softmax}(\alpha(\mathbf{q}, \mathbf{k})) = \frac{e^{\alpha(\mathbf{q}, \mathbf{k})}}{\sum_j e^{\alpha(\mathbf{q}, \mathbf{k}_j)}}. \quad (7)$$

#### **iForest**

The iForest algorithm [23] identifies anomalies, especially effective in large datasets. It isolates anomalies using binary trees from random data subsets.

Each tree randomly selects a feature and a value, splitting the data into two groups. This continues until maximum depth or isolation is achieved.

The anomaly score is the average path length from the root to the leaf across trees. The formal definition involves  $\mathbf{x}_i \in \mathbb{R}^d$  in a forest  $F$  of  $T$  trees. The isolation degree  $h(\mathbf{x})$  is:

$$h(\mathbf{x}, F) = 2^{\frac{E[h(\mathbf{x})]}{c(T)}}, \quad (8)$$

with  $c(T) = 2H(T-1) - \frac{2(T-1)}{n}$ , where  $H(i)$  is the harmonic series, and  $n$  is the sample size.

An object's classification as an anomaly uses threshold  $\tau$ :

$$y(\mathbf{x}) = \begin{cases} \text{anomaly, if } h(\mathbf{x}, F) < \tau \\ \text{non - anomaly, otherwise} \end{cases}. \quad (9)$$

iForest's performance is sensitive to hyperparameters, requiring careful tuning for large datasets.

#### Attention-based iForest with scoring function

We propose a new method for ABIF-SF anomaly detection that incorporates an attention mechanism into iForest using scoring functions with trainable parameters.

##### *Attention mechanism: query, keys, values*

In the iForest method, the average path length  $h_k(\mathbf{x})$  for a point  $\mathbf{x}$  over all  $T$  trees can be expressed as follows:

$$E[h(\mathbf{x})] = \frac{1}{T} \sum_{k=1}^T h_k(\mathbf{x}), \quad (10)$$

where  $h_k(\mathbf{x})$  is the path length of instance  $\mathbf{x}$  in tree  $k$  and serves as the value. Using the attention mechanism allows us to rewrite the computation of the expected path length  $E[h(\mathbf{x})]$  in iForest using attention weights  $\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w})$  [32, 52]:

$$E[h(\mathbf{x})] = \sum_{k=1}^T \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}) \cdot h_k(\mathbf{x}), \quad (11)$$

where  $\mathbf{x} \in \mathbb{R}^d$ ,  $\mathbf{A}_k(\mathbf{x})$  is the average vector of all vectors  $\mathbf{x}_j$  with indices  $j \in J_i(k)$  in the  $i$ -th leaf of the  $k$ -th tree that contains the feature vector  $\mathbf{x}$ , and  $J_i(k)$  is the set of indices  $n_i(k)$  of training instances that also fall into the same leaf, and  $\mathbf{w}$  is a set of trainable parameters.

$$\mathbf{A}_k(\mathbf{x}) = \frac{1}{n_i(k)} \sum_{j \in J_i(k)} \mathbf{x}_j. \quad (12)$$

This vector characterizes the group of instances in the corresponding leaf and serves as the key, while  $\mathbf{x}$  serves as the query.

$\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w})$  represents the importance of the average instance  $\mathbf{A}_k(\mathbf{x})$  for the vector  $\mathbf{x}$  and satisfies the following conditions:

$$\sum_{k=1}^T \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}) = 1, \quad \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}) \geq 0, \quad k = 1, \dots, T. \quad (13)$$

In [32], the authors used Huber's contamination model with weights of the following form:

$$\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}) = (1 - \varepsilon) \cdot \sigma \left( -\frac{\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2}{\omega} \right) + \varepsilon \cdot w_k, \quad (14)$$

where  $\varepsilon \in \mathbb{R}$ ,  $\omega \in \mathbb{R}$  and  $\sigma$  is a sigmoid function. This equation shows that the attention weight depends linearly on the trainable parameters  $\mathbf{w} = (w_1, \dots, w_T)$  where  $T$  is the number of components. The softmax operation depends only on the hyperparameter  $\omega$ . The trainable parameters  $w$  are restricted to the unit simplex  $\mathcal{S}(1, T)$ , which means that the constraints on  $w$  are linear ( $w_i \geq 0$  and  $w_1 + \dots + w_T = 1$ ).

One drawback of this formulation is that there are no trainable parameters in the sigmoid function. Essentially, the expression inside the sigmoid function represents scoring functions:

$$W_{opt} = \arg \min_{w \in M} \sum_{S=1}^n \max \left( 0, y_S \left( \sum_{k=1}^T \alpha(\mathbf{x}, A_k(\mathbf{x}), W_Q, W_X) \cdot h_k(\mathbf{x}_S) - \gamma \right) \right). \quad (15)$$

By using these scoring functions inside the sigmoid without the Huber model, the attention mechanism can learn to assign higher weights to more relevant components in the input, leading to improved accuracy. The Huber model is designed to be more robust to outliers, but it may also smooth out the gradients and make the learning process slower. By using only the sigmoid function with the scoring functions the model can directly optimize the attention weights based on the relevance of the components in the input, without the additional smoothing effect of the Huber model. This can lead to faster convergence and better performance in some cases.

#### Scoring functions as attention weights

Scoring functions can be used as  $\alpha(\mathbf{x}, A_k(\mathbf{x}), \mathbf{w})$  using, for example, dot-scale and additive attention.

For dot-scale attention,  $\alpha(\mathbf{x}, A_k(\mathbf{x}), \mathbf{w})$  can be defined as follows:

$$\alpha(\mathbf{x}, A_k(\mathbf{x}), W_Q, W_X) = \sigma \left( \frac{W_Q^T \cdot \mathbf{x} \cdot A_k(\mathbf{x}) \cdot W_X^{(k)T}}{\sqrt{d}} \right), \quad (16)$$

where  $W_Q \in \mathbb{R}^d$ ,  $W_X^{(k)} \in \mathbb{R}^d$  are trainable parameter vectors ( $W_X \in \mathbb{R}^{d \times T}$ ),  $d$  is the dimensionality of vectors  $\mathbf{x}$  and  $A_k(\mathbf{x})$ , and  $\sigma$  is the softmax function.

For additive attention,  $\alpha(\mathbf{x}, A_k(\mathbf{x}), \mathbf{w})$  can be defined as follows:

$$\alpha(\mathbf{x}, A_k(\mathbf{x}), W_Q, W_X) = \sigma \left( \tanh \left( W_Q^T \cdot \mathbf{x} + W_X^{(k)T} \cdot A_k(\mathbf{x}) \right) \right). \quad (17)$$

The final form of computing  $E[h(\mathbf{x})]$  for additive attention can be written as:

$$E[h(\mathbf{x})] = \sum_{k=1}^T \sigma \left( \tanh \left( W_Q^T \cdot \mathbf{x} + W_X^{(k)T} \cdot A_k(\mathbf{x}) \right) \right) \cdot h_k(\mathbf{x}), \quad (18)$$

and for dot-scale:

$$E[h(\mathbf{x})] = \sum_{k=1}^T \sigma \left( \frac{W_Q^T \cdot \mathbf{x} \cdot A_k(\mathbf{x}) \cdot W_X^{(k)T}}{\sqrt{d}} \right) \cdot h_k(\mathbf{x}). \quad (19)$$

In both cases, trainable parameters are included in the expression through  $W_Q$ ,  $W_X^{(k)}$ .

To determine whether an object is an anomaly, a reformulation of the decision making from the classic isolation forest ( $h(\mathbf{x}, F) < \tau$ ) should be used to make a decision based on  $E[h(\mathbf{x})]$  [34].

$$y(\mathbf{x}) = \{\text{anomaly}, E[h(\mathbf{x})] \leq \gamma, \text{otherwise}\}. \quad (20)$$

Training attention weights allows the iForest models to better consider relationships between instances and each tree, which can overall improve the quality of anomaly detection.

#### Loss function

Standard optimization methods such as gradient descent or its variants can be used to train the parameters  $W_Q$  and  $W_X^{(k)}$ .

To use optimization methods, it is necessary to define a loss function between the model prediction expressed through the expression  $E[h(\mathbf{x}_S)] - \gamma$  and the label of the data  $y_S$ , where the index  $S$  indicates the objects from the training dataset.

The loss function  $L$  has the form:

$$L(E[h(\mathbf{x}_S)] - \gamma, y_S) = \max(0, y_S(E[h(\mathbf{x}_S)] - \gamma)), \quad (21)$$

where  $y_S$  is the label of the instances (the label of instance  $y_S$  is 1 if  $\mathbf{x}_S$  is anomalous and  $-1$  if it is normal).

In [32],  $\gamma$  was calculated as follows:

$$\gamma = -c(n) \cdot \log_2(\tau). \quad (22)$$

We propose to include  $\gamma$  as a trainable parameter along with  $\mathbf{W}_Q$  and  $\mathbf{W}_X^{(k)}$ .

The general form of the minimization problem can be written as follows:

$$\mathbf{W}_{opt} = \arg \min_{\mathbf{w} \in M} \sum_{S=1}^n \max\left(0, y_S \left( \sum_{k=1}^T \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{W}_Q, \mathbf{W}_X) \cdot h_k(\mathbf{x}_S) - \gamma \right)\right), \quad (23)$$

where  $M$  is the space of trainable parameters for  $\mathbf{W}_Q$ ,  $\mathbf{W}_X$  and  $\gamma$ ,  $s$  is the index of training instances, of which there are  $n$  instances.

When using gradient descent, gradients with respect to the trainable parameters are used. The general parameter optimization step is classical for gradient descent algorithms and its modifications.

### Numerical experiments

The aim of this chapter is to provide a comprehensive evaluation of the proposed method using numerical experiments. The experiments are designed to demonstrate the effectiveness of the method in comparison to the other described in this article approaches, and to show the impact of various parameters on the performance of the method. In the experiments, we will compare the performance of the three models on a variety of datasets and use standard evaluation metrics such as F1-score to assess the performance of each model. The results will be presented in the form of tables and graphs to allow for a clear and comprehensive comparison of the models.

Gradient descent is used for optimization with the following parameters: learning rate is 0.001, optimizer – ADAM.

The experiments utilized both real-world and synthetic datasets spanning anomaly detection challenges across different domains:

- Arrhythmia<sup>1</sup> – electrocardiogram (ECG) slices from the Kaggle repository;
- Credit<sup>2</sup> – credit card transaction dataset from the Kaggle repository;
- Pima<sup>3</sup> – Pima Indians Diabetes Database from the NIDDK;
- EEG Eye<sup>4</sup> – electroencephalogram (EEG) eye state samples from the Kaggle repository;
- Haberman<sup>5</sup> – Haberman's survival dataset from the Kaggle repository;

<sup>1</sup> Tavares M. Binary classification on arrhythmia dataset. Kaggle, 2023. Available: <https://www.kaggle.com/code/mtavares51/binary-classification-on-arrhythmia-dataset> (Accessed 29.08.2024)

<sup>2</sup> Sekra S. Credit card fraud detection – EDA & Isolation Forest. Kaggle, 2023. Available: <https://www.kaggle.com/code/shivamsekra/credit-card-fraud-detection-eda-isolation-forest> (Accessed 29.08.2024)

<sup>3</sup> Ramadan H. Data science project III. Kaggle, 2023. Available: <https://www.kaggle.com/code/hafizramadan/data-science-project-iii> (Accessed 29.08.2024)

<sup>4</sup> Scube R. Eye state classification EEG dataset. Kaggle, 2023. Available: <https://www.kaggle.com/datasets/robikscube/eye-state-classification-eeg-dataset> (Accessed 29.08.2024)

<sup>5</sup> Sousa G. Haberman's survival data set. Kaggle, 2023. Available: <https://www.kaggle.com/datasets/gilsousa/habermans-survival-data-set> (Accessed 29.08.2024)

- HTTP<sup>6</sup> – HTTP network intrusion dataset from the OpenML repository;
- Ionosphere<sup>7</sup> – radar returns from the ionosphere dataset in the Kaggle repository;
- Mulcross<sup>8</sup> – synthetically generated multivariate normal distribution with anomaly clusters from the OpenML repository.

Table 1

### A brief introduction of the datasets

| Dataset    | Normal | Abnormal | Number of features |
|------------|--------|----------|--------------------|
| Arrhythmia | 386    | 66       | 17                 |
| Credit     | 1500   | 400      | 30                 |
| Pima       | 500    | 268      | 8                  |
| EEG Eye    | 847    | 653      | 11                 |
| Haberman   | 225    | 81       | 3                  |
| HTTP       | 500    | 50       | 3                  |
| Ionosphere | 225    | 126      | 33                 |
| Mulcross   | 1800   | 400      | 4                  |

To facilitate computational efficiency, smaller excerpted samples rather than full dataset volumes were utilized for larger real-world sources (Table 1). Certain distributions also underwent preprocessing including normalization and feature selection to conform inputs to model assumptions, with code available in the ABIF-SF repository (<https://github.com/AndreyAgeev/abif-sf>).

In the experiments, we use the following evaluation metrics to assess the performance of the method:

- F1-score: The harmonic mean of precision and recall.

The proposed method was implemented using the programming language Python and the library PyTorch.

The method was compared with the following approaches:

- IForest;
- ABIF.

## Experimental Results

### *Comparison between iForest, ABIF and ABIF-SF*

To measure the performance, we use the F1-score, which is a commonly used metric in anomaly detection. We compare the F1-score dependence on the number of epochs on several datasets. To evaluate the F1-score, 66.7% of the data were randomly selected for training and 33.3% were randomly selected for testing.

The performance of the proposed method was compared with iForest and ABIF.

The results are shown in Table 2.

For these experiments, 5000 training epochs were carried out, the best weights was taken from the minimum error value on the training set. This approach was used to obtain a result from the point of view of a practitioner who could use a similar approach to quickly obtain a result without setting parameters, validation dataset and other parameters.

For the models, the number of trees 150 was chosen.

<sup>6</sup> HTTP. OpenML. 2023. Available: <https://www.openml.org/search?type=data&sort=runs&id=40897&status=active> (Accessed 29.08.2024)

<sup>7</sup> Zymzym. Classification of the Ionosphere dataset by KNN. Kaggle, 2023. Available: <https://www.kaggle.com/code/zymzym/classification-of-the-ionosphere-dataset-by-knn> (Accessed 29.08.2024)

<sup>8</sup> Mulcross. OpenML. 2023. Available: <https://www.openml.org/search?type=data&sort=runs&id=40897&status=active> (Accessed 29.08.2024)

Table 2

Comparison of algorithms on different data sets

| Dataset    | ABIF             |        |                  | iForest      |        | Additive     | Dot-scale    |              |
|------------|------------------|--------|------------------|--------------|--------|--------------|--------------|--------------|
|            | $\epsilon_{opt}$ | $\tau$ | $\tau_{softmax}$ | F1           | $\tau$ | F1           | F1           |              |
| Arrhythmia | 1.0              | 0.45   | —                | 0.472        | 0.45   | 0.484        | <b>0.853</b> | 0.849        |
| Credit     | 0.5              | 0.55   | 0.1              | 0.862        | 0.45   | 0.798        | 0.930        | <b>0.932</b> |
| Pima       | 0.75             | 0.45   | 10               | 0.555        | 0.4    | 0.532        | <b>0.667</b> | 0.648        |
| EEG Eye    | 1.0              | 0.45   | —                | <b>0.724</b> | 0.35   | <b>0.724</b> | 0.5          | 0.543        |
| Haberman   | 1.0              | 0.45   | —                | 0.486        | 0.45   | 0.473        | <b>0.732</b> | 0.728        |
| HTTP       | 0.0              | 0.55   | 0.1              | 0.739        | 0.5    | 0.628        | <b>0.901</b> | 0.880        |
| Ionosphere | 1.0              | 0.45   | —                | 0.649        | 0.45   | 0.652        | 0.679        | <b>0.686</b> |
| Mullcross  | 0.0              | 0.6    | 0.1              | 0.525        | 0.5    | 0.538        | 0.852        | <b>0.897</b> |

Hyperparameters for the isolation forest and attention-based models were selected through a grid search over reasonable values, following a procedure like that used by the authors of the original ABIF paper. Specifically, we predefined grids of potential hyperparameters, including contamination model epsilon values and anomaly thresholds. Models were trained and evaluated on a test set across the grid space. The best performing hyperparameter configuration on the test data was then selected and used to produce the primary results and comparisons between ABIF, ABIF-SF, and isolation forest reported in this work. We use 10 different seeds when building trees, and 10 times shuffle train/test dataset, and then average the results of the metrics.

In the experiments, we used a smaller number of dataset partitions and different seeds due to the addition of new algorithms when comparing, and therefore, on average, the best results could be obtained with  $\epsilon$  equal to 0 or 1, which does not coincide with the results of the author of the article on ABIF. When using more seeds and experiments on average on datasets, it is preferable to choose  $\epsilon$  not equal to 0.

**Analysis of learning dynamics**

Monitoring model performance across training epochs provides insight into learning dynamics – identifying overfitting, suitable regularization, optimal timing to stop training, etc. Here we track the F1 score after each epoch on the test set to assess ABIF-SF’s resilience to overfitting as additional iterations may better fit the training distribution without improving generalization. Ideally, test set metrics should steadily improve before plateauing once the intrinsic complexity is reached. Declining scores indicates overfitting – losing generalization due to redundant adaptation on noise or spurious patterns. The scoring functions contain little explicit regularization, hence the trends characterize inherent resistance to overlearning.

We trained dot-product and additive models for 5000 epochs on the Arrhythmia, Credit, EEG Eye, Haberman, HTTP, Ionosphere, and Mullcross datasets. The number of trees was fixed at five to better stress test potential overfitting. At each epoch, the parameter set minimizing training error was evaluated on the test data.

Fig. 1 shows the F1-score learning curves on the test datasets over training progression. The dot-product scoring consistently demonstrates stable or gradually improving F1 while not overfitting even after thousands of iterations. The additive attention exhibits more volatility, with drops in some datasets. For example, in the Arrhythmia set, additive scoring peaks at epoch 1000 before declining by nearly 3% in F1-score. However, dot-product matches best performance around epoch 4000 and smoothly converges thereafter. The EEG Eye dataset proves challenging for both formulations, plateauing below 60% F1-score. Still dot-product dominates additively, backed by superior Arrhythmia and Haberman results. The trends indicate inherent regularization properties differentiate the scoring mechanisms. Dot-product generalizes



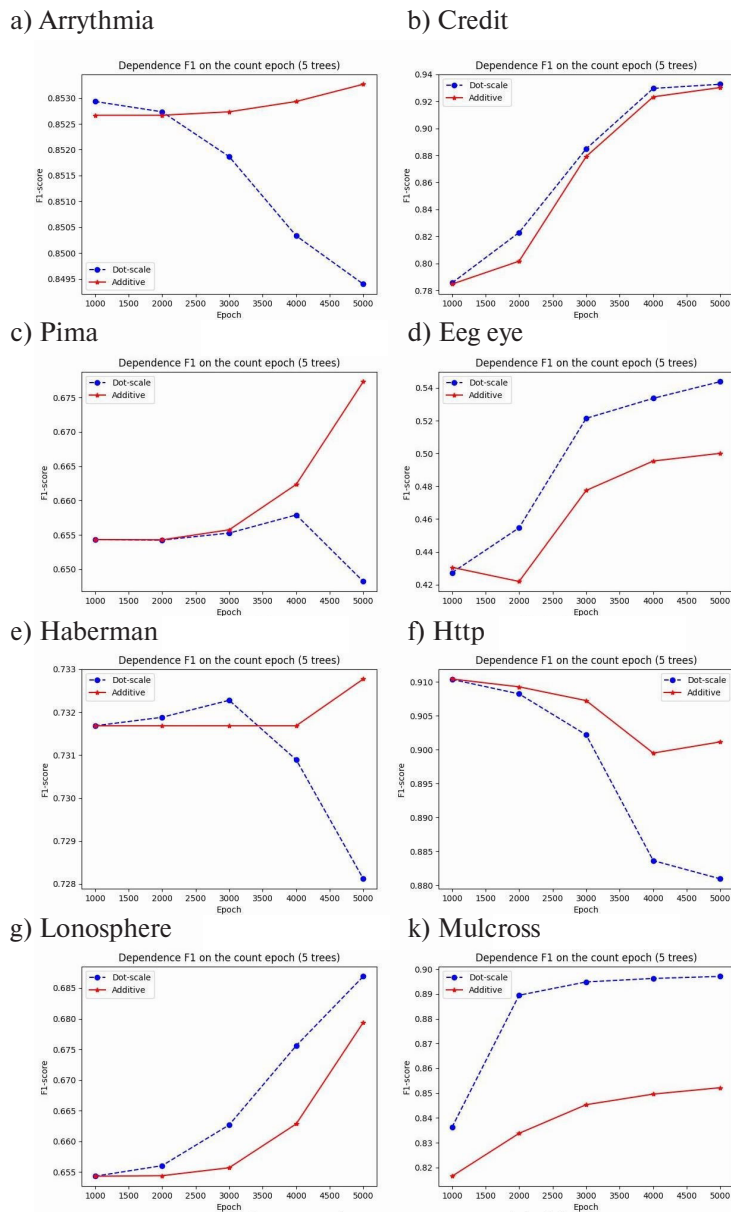


Fig. 1. Comparison ABIF-SF scoring function

reliably with extensive epochs while additive learning can become unstable. Sensitivity to initial conditions, co-adaptation of weights, and disparate gradient behaviors likely explain discrepancies. The results also highlight the harder EEG Eye distribution where better feature extraction is essential. In summary, tracking F1-score across training epochs reveals additive attention more vulnerable to overfitting than dot-product formulations. This highlights the greater regularization of dot mechanisms, also backed by consistently good performance into thousands of iterations. The analysis also identifies limitations modeling certain distributions and suggests enhancements like constrained optimization, dropout, or batch normalization to further boost robustness.

**Impact of training set size**

In real-world scenarios, the volume of quality training data available can vary significantly across anomaly detection tasks. To characterize the data efficiency and generalization capability of the proposed ABIF-SF model, we investigated performance with enlarged and reduced dataset sizes. Intuitively,

additional high-quality examples should enable better learning of normal vs anomalous patterns. However, insufficient or redundant data could respectively lead to underfitting or overfitting. We evaluated the F1 score on the test set for the additive and dot-product scoring functions using 80%, 100%, and 120% of the original training set sizes. The Credit, HTTP, and Mulcross datasets were employed for these experiments with 2000 training epochs. At each epoch, the best-performing model parameterization on the training set was selected for final evaluation on the test data. Results are shown in Table 3.

Table 3

**F1-Score vs training set size**

| Dataset  | Additive |       |       | Dot-Product |       |       |
|----------|----------|-------|-------|-------------|-------|-------|
|          | 80%      | 100%  | 120%  | 80%         | 100%  | 120%  |
| Credit   | 0.794    | 0.801 | 0.834 | 0.814       | 0.822 | 0.853 |
| HTTP     | 0.907    | 0.909 | 0.903 | 0.907       | 0.908 | 0.908 |
| Mulcross | 0.824    | 0.833 | 0.850 | 0.869       | 0.889 | 0.894 |

The Credit and Mulcross datasets exhibit consistent improvement in anomaly detection accuracy (F1-score) as more training examples are provided, plateauing at the maximum 120% volume. This demonstrates both scoring functions can effectively leverage additional representative data to better learn normal vs anomalous patterns in these distributions. However, the story differs markedly on the HTTP dataset. Surprisingly, the additive scoring function shows a decline in accuracy from 0.909 to 0.903 when switching from 100% to 120% training data volumes. At the same time, the dot-product scoring function remains stable at 0.908 F1-score despite variations in data amount. Decreasing the dataset size also only causes minor performance changes for both approaches. This reversal in trends for the HTTP dataset suggests differing generalization capabilities between the scoring formulations. The additive model appears to overfit on the augmented 120% training set – overly adapting to patterns that do not transfer to the test data. Meanwhile, the performance consistency of dot-product scoring implies it has saturated learning from this distribution once 100% examples are available. Additional data volume provides redundancy rather than meaningful new information. Furthermore, both functions achieve their maximal accuracy with only 80% subset, confirming enough representative information was intrinsically available in the original dataset. In conclusion, while ABIF-SF can leverage increased training data for some distributions, performance plateaus or drops past distribution-dependent optimal training set sizes. Choosing appropriate volumes with sufficient but concise representative examples is vital for efficiently learning anomalies, avoiding under- or over-fitting tendencies. Our experiments also highlight distinctions between the scoring formulations – additive functions may better model some distributions but are more prone to overfitting compared to more robust dot-product attention.

**Conclusion**

This article presents a novel anomaly detection model, the attention-based isolation forest with scoring function (ABIF-SF), which is an improvement of the original iForest. The proposed model utilizes attention weights, which are determined by scoring functions, to enhance its performance. The experiments conducted using real datasets demonstrate the superiority of the proposed model compared to the original isolation forest and the attention-based isolation forest eps-contamination model. The source code for this algorithm has been made publicly available for further research and development.

## REFERENCES

1. **Chandola V., Banerjee A., Kumar V.** Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 2009, Vol. 41, No. 3, Pp. 1–58. DOI: 10.1145/1541880.1541882
2. **Barnett V., Lewis T.** *Outliers in Statistical Data*, 3<sup>rd</sup> ed. Chichester: Wiley. 1994. 584 p.
3. **Foorthuis R.** On the nature and types of anomalies: a review of deviations in data. *International Journal of Data Science and Analytics*, 2021, Vol. 12, Pp. 297–331. DOI: 10.1007/s41060-021-00265-1
4. **Liao Y., Bartler A., Yang B.** Anomaly detection based on selection and weighting in latent space. *arXiv:2103.04662*, 2021. DOI: 10.48550/arXiv.2103.04662
5. **Xu J., Wu H., Wang J., Long M.** Anomaly transformer: Time series anomaly detection with association discrepancy. *arXiv:2110.02642*, 2022. DOI: 10.48550/arXiv.2110.02642
6. **Li Z., Xiang Z., Gong W., Wang H.** Unified model for collective and point anomaly detection using stacked temporal convolution networks. *Applied Intelligence*, 2022, Vol. 52, Pp. 3118–3131. DOI: 10.1007/s10489-021-02559-0
7. **Chatterjee A., Ahmed B.S.** IoT anomaly detection methods and applications: A survey. *Internet of Things*, 2022, Vol. 19, Art. no. 100568. DOI: 10.1016/j.iot.2022.100568
8. **Wu T., Wang Y.** Locally interpretable one-class anomaly detection for credit card fraud detection. *arXiv:2108.02501*, 2021. DOI: 10.48550/arXiv.2108.02501
9. **Bierbrauer D.A., Chang A., Kritzer W., Bastian N.D.** Cybersecurity anomaly detection in adversarial environments. *arXiv:2105.06742*, 2021. DOI: 10.48550/arXiv.2105.06742
10. **Fisch A.T.M., Eckley I.A., Fearnhead P.** A linear time method for the detection of point and collective anomalies. *arXiv:1806.01947*, 2018. DOI: 10.48550/arXiv.1806.01947
11. **Li Z., van Leeuwen M.** Robust and explainable contextual anomaly detection using quantile regression forests. *arXiv:2302.11239v1*, 2023.
12. **Madhuri G.S., Rani M.U.** Anomaly detection techniques. *2018 IADS International Conference on Computing, Communications & Data Engineering (CCODE)*, 2018. DOI: 10.2139/ssrn.3167172
13. **Gafni T., Wolff B., Revach G., Shlezinger N., Cohen K.** Anomaly search over discrete composite hypotheses in hierarchical statistical models. *IEEE Transactions on Signal Processing*, 2023, Vol. 71, Pp. 202–217. DOI: 10.1109/TSP.2023.3242074
14. **Kandanaarachchi S., Hyndman R.J.** Anomaly detection in dynamic networks. *arXiv:2210.07407*, 2022. DOI: 10.48550/arXiv.2210.07407
15. **Hou Y., Chen Z., Wu M., Foo C.-S., Li X., Shubair R.M.** Mahalanobis distance based adversarial network for anomaly detection. *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, Pp. 3192–3196. DOI: 10.1109/ICASSP40776.2020.9053206
16. **Sarmadi H., Karamodin A.** A novel anomaly detection method based on adaptive Mahalanobis-squared distance and one-class kNN rule for structural health monitoring under environmental effects. *Mechanical Systems and Signal Processing*, 2020, Vol. 140, Art. no. 106495. DOI: 10.1016/j.ymsp.2019.106495
17. **Souto Arias L.A., Oosterlee C.W., Cirillo P.** AIDA: Analytic isolation and distance-based anomaly detection algorithm. *arXiv:2212.02645*, 2022. DOI: 10.48550/arXiv.2212.02645
18. **Wang W., Zhang B., Wang D., Jiang Y., Qin S., Xue L.** Anomaly detection based on probability density function with Kullback–Leibler divergence. *Signal Processing*, 2016, Vol. 126, Pp. 12–17. DOI: 10.1016/j.sigpro.2016.01.008
19. **Liu B., Tan P.-N., Zhou J.** Unsupervised anomaly detection by robust density estimation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, Vol. 36, No. 4, Pp. 4101–4108. DOI: 10.1609/aaai.v36i4.20328
20. **Le Lan C., Dinh L.** Perfect density models cannot guarantee anomaly detection. *arXiv:2012.03808*, 2020. DOI: 10.48550/arXiv.2012.03808

21. **Zimek A., Filzmoser P.** There and back again: Outlier detection between statistical reasoning and data mining algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2018, Vol. 8, No. 6, Art. no. e1280. DOI: 10.1002/widm.1280
22. **Knorr E.M., Ng R.T., Tucakov V.** Distance-based outliers: algorithms and applications. *The VLDB Journal*, 2000, Vol. 8, Pp. 237–253. DOI: 10.1007/s007780050006
23. **Liu F.T., Ting K.M., Zhou Z.-H.** Isolation forest. *2008 8<sup>th</sup> IEEE International Conference on Data Mining*, 2008, Pp. 413–422. DOI: 10.1109/ICDM.2008.17
24. **Fanaee-T H., Gama J.** Tensor-based anomaly detection: An interdisciplinary survey. *Knowledge-Based Systems*, 2016, Vol. 98, Pp. 130–147. DOI: 10.1016/j.knosys.2016.01.027
25. **Zimek A., Schubert E., Krieger H.-P.** A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining*, 2012, Vol. 5, Pp. 363–387. DOI: 10.1002/sam.11161
26. **Nassif A.B., Talib M.A., Nasir Q., Dakalbab F.M.** Machine learning for anomaly detection: A systematic review. *IEEE Access*, 2021, Vol. 9, Pp. 78658–78700. DOI: 10.1109/ACCESS.2021.3083060
27. **Alloqmani A., Abushark Y.B., Khan A.I., Alsolami F.** Deep learning based anomaly detection in images: Insights, challenges and recommendations. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 2021, Vol. 12, No. 4, Pp. 205–215. DOI: 10.14569/IJACSA.2021.0120428
28. **Matsuo H., Nishio M., Kanda T., Kojita Y., Kono A.K., Hori M., Teshima M., Otsuki N., Nibu K.-i., Murakami T.** Diagnostic accuracy of deep-learning with anomaly detection for a small amount of imbalanced data: discriminating malignant parotid tumors in MRI. *Scientific Reports*, 2020, Vol. 10, Art. no. 19388. DOI: 10.1038/s41598-020-76389-4
29. **Kim M., Moon K.-R., Lee B.-D.** Unsupervised anomaly detection for posteroanterior chest -rays using multiresolution patch-based self-supervised learning. *Scientific Reports*, 2023, Vol. 13, Art. no. 3415. DOI: 10.1038/s41598-023-30589-w
30. **Maya S., Ueno K., Nishikawa T.** dLSTM: a new approach for anomaly detection using deep learning with delayed prediction. *International Journal of Data Science and Analytics*, 2019, Vol. 8, Pp. 137–164. DOI: 10.1007/s41060-019-00186-0
31. **Kurniabudi, Purnama B., Sharipuddin, Darmawijoyo, Stiawan D., Samsuryadi, Heryanto A., Budiarto R.** Network anomaly detection research: A survey. *Indonesian Journal of Electrical Engineering and Informatics (IJEI)*, 2019, Vol. 7, No. 1, Pp. 37–50. DOI: 10.11591/ijeie.v7i1.773
32. **Utkin L.V., Ageev A.Y., Konstantinov A.V., Muliukha V.A.** Improved anomaly detection by using the attention-based isolation forest. *Algorithms*, 2023, Vol. 16, No. 1, Art. no. 19. DOI: 10.3390/a16010019
33. **Takimoto H., Seki J., Situju S.F., Kanagawa A.** Anomaly detection using Siamese network with attention mechanism for few-shot learning. *Applied Artificial Intelligence*, 2022, Vol. 36, No. 1, Art. no. 2094885. DOI: 10.1080/08839514.2022.2094885
34. **Zhou H., Xia H., Zhan Y., Mao Q.** Salient attention model and classes imbalance remission for video anomaly analysis with weak label. *Human Centered Computing (HCC 2020)*, 2021, Pp. 126–135. DOI: 10.1007/978-3-030-70626-5\_13
35. **Dong F., Chen S., Demachi K., Yoshikawa M., Seki A., Takaya S.** Attention-based time series analysis for data-driven anomaly detection in nuclear power plants. *Nuclear Engineering and Design*, 2023, Vol. 404, Art. no. 112161. DOI: 10.1016/j.nucengdes.2023.112161
36. **Yu Y., Zha Z., Jin B., Wu G., Dong C.** Graph-based anomaly detection via attention mechanism. *Intelligent Computing Theories and Application (ICIC 2022)*, 2022, Pp. 401–411. DOI: 10.1007/978-3-031-13870-6\_33
37. **Zhu Y., Newsam S.** Motion-aware feature for improved video anomaly detection. *arXiv:1907.10211*, 2019. DOI: 10.48550/arXiv.1907.10211
38. **Hojjati H., Ho T.K.K., Armanfard N.** Self-supervised anomaly detection: A survey and outlook. *arXiv:2205.05173v5*, 2024.
39. **Perera P., Oza P., Patel V.M.** One-class classification: A survey. *arXiv:2101.03064*, 2021. DOI: 10.48550/arXiv.2101.03064

40. Darban Z.Z., Webb G.I., Pan S., Aggarwal C.C., Salehi M. Deep learning for time series anomaly detection: A survey. *arXiv:2211.05244*, 2022. DOI: 10.48550/arXiv.2211.05244
41. Chalapathy R., Chawla S. Deep learning for anomaly detection: A survey. *arXiv:1901.03407*, 2019. DOI: 10.48550/arXiv.1901.03407
42. Landauer M., Onder S., Skopik F., Wurzenberger M. Deep learning for anomaly detection in log data: A survey. *arXiv:2207.03820*, 2022. DOI: 10.48550/arXiv.2207.03820
43. Di Mattia F., Galeone P., De Simoni M., Ghelfi E. A survey on GANs for anomaly detection. *arXiv:1906.11632*, 2019. DOI: 10.48550/arXiv.1906.11632
44. Suarez J.J.P., Naval Jr. P.C. A survey on deep learning techniques for video anomaly detection. *arXiv:2009.14146*, 2020. DOI: 10.48550/arXiv.2009.14146
45. Tschuchnig M.E., Gadermayr M. Anomaly detection in medical imaging – A mini review. *Data Science – Analytics and Applications (iDSC 2021)*, 2022, Pp. 33–38. DOI: 10.1007/978-3-658-36295-9\_5
46. Hashimoto M., Ide Y., Aritsugi M. Anomaly detection for sensor data of semiconductor manufacturing equipment using a GAN. *Procedia Computer Science*, 2021, Vol. 192, Pp. 873–882. DOI: 10.1016/j.procs.2021.08.090
47. Wu X., Huang S., Li M., Deng Y. Vector magnetic anomaly detection via an attention mechanism deep-learning model. *Applied Sciences*, 2021, Vol. 11, No. 23. Art. no. 11533. DOI: 10.3390/app112311533
48. Cortes D. Isolation forests: looking beyond tree depth. *arXiv:2111.11639*, 2021. DOI: 10.48550/arXiv.2111.11639
49. Gao R., Zhang T., Sun S., Liu Z. Research and improvement of isolation forest in detection of local anomaly points. *Journal of Physics: Conference Series*, 2019, Vol. 1237, Art. no. 052023. DOI: 10.1088/1742-6596/1237/5/052023
50. Karczmarek P., Kiersztyn A., Pedrycz W., Al E. K-Means-based isolation forest. *Knowledge-Based Systems*, 2020, Vol. 195, Art. no. 105659. DOI: 10.1016/j.knosys.2020.105659
51. Galka L., Karczmarek P., Tokovarov M. Isolation forest based on minimal spanning tree. *IEEE Access*, 2022, Vol. 10, Pp. 74175–74186. DOI: 10.1109/ACCESS.2022.3190505
52. Utkin L.V., Konstantinov A.V. Attention-based random forest and contamination model. *arXiv:2201.02880*, 2022. DOI: 10.48550/arXiv.2201.02880
53. Niu Z., Zhong G., Yu H. A review on the attention mechanism of deep learning. *Neurocomputing*, 2021, Vol. 452, Pp. 48–62. DOI: 10.1016/j.neucom.2021.03.091
54. Bahdanau D., Cho K., Bengio Y. Neural machine translation by jointly learning to align and translate. *arXiv:1409.0473*, 2014. DOI: 10.48550/arXiv.1409.0473
55. Cai Z. Weighted Nadaraya–Watson regression estimation. *Statistics & Probability Letters*, 2001, Vol. 51, No. 3, Pp. 307–318. DOI: 10.1016/S0167-7152(00)00172-3
56. Chen X., Li D., Li Q., Li Z. Nonparametric estimation of conditional quantile functions in the presence of irrelevant covariates. *Journal of Econometrics*, 2019, Vol. 212, No. 2, Pp. 433–450. DOI: 10.1016/j.jeconom.2019.04.037
57. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I. Attention is all you need. *arXiv:1706.03762*, 2017. DOI: 10.48550/arXiv.1706.03762

## INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Ageev Andrey Yu.  
 Ageev Андрей Юрьевич  
 E-mail: andreyageev1@mail.ru

**Utkin Lev V.**

**Уткин Лев Владимирович**

E-mail: lev.utkin@gmail.com

**Konstantinov Andrei V.**

**Константинов Андрей Владимирович**

E-mail: andrue.konst@gmail.com

*Submitted: 21.10.2024; Approved: 17.01.2025; Accepted: 12.03.2025.*

*Поступила: 21.10.2024; Одобрена: 17.01.2025; Принята: 12.03.2025.*

Research article

DOI: <https://doi.org/10.18721/JCSTCS.18102>

UDC 004.94



## A METHOD FOR MODELING OF INDIVIDUAL AGENT BEHAVIOR IN THE PROCESS-NETWORK PARADIGM OF DISCRETE-EVENT SIMULATION

*A.M. Sabutkevich* ✉, *I.V. Nikiforov*, *A.V. Samochadin*

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

✉ [artem.sabut@gmail.com](mailto:artem.sabut@gmail.com)

**Abstract.** Due to the active growth of demand for cloud resources, the task of increasing the efficiency of their use becomes relevant. One of the approaches to solving this problem is the application of discrete-event simulation in the process-network paradigm, which allows describing the modeled process in the form of processing nodes united in a single network. However, this paradigm does not consider the individual behavior of agents, which reduces the adequacy of the resulting models. The paper presents an approach to assessing the adequacy and significance of simulation models, and proposes a method that allows specifying and considering of the individual behavior of agents in the simulation process, implemented in accordance with the process-network paradigm. The application of this method allows increasing the applied significance of the models of the processes under study. The paper describes the integration of the proposed method into systems implementing the process-network paradigm and presents its software implementation. The latter allows to investigate the influence of considering individual agent behavior on the adequacy and significance of the model of virtual machines placement relative to physical servers. Due to the application of the method, it was possible to achieve an increase in the adequacy of the model under study by an average of 12.5% and, as a consequence, to increase the number of significant models by 65% for the selected adequacy threshold value.

**Keywords:** discrete-event simulation, agent-based simulation, process-network paradigm, model adequacy, simulation of cloud infrastructures

**Citation:** Sabutkevich A.M., Nikiforov I.V., Samochadin A.V. A method for modeling of individual agent behavior in the process-network paradigm of discrete-event simulation. *Computing, Telecommunications and Control*, 2025, Vol. 18, No. 1, Pp. 23–35. DOI: 10.18721/JCSTCS.18102

Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18102>

УДК 004.94



## МЕТОД СИМУЛЯЦИИ ИНДИВИДУАЛЬНОГО ПОВЕДЕНИЯ АГЕНТОВ В ПРОЦЕССНО-СЕТЕВОЙ ПАРАДИГМЕ ДИСКРЕТНО-СОБЫТИЙНОГО МОДЕЛИРОВАНИЯ

*А.М. Сабуткевич* ✉, *И.В. Никифоров*, *А.В. Самочадин*

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

✉ [artem.sabut@gmail.com](mailto:artem.sabut@gmail.com)

**Аннотация.** В связи с активным ростом спроса на облачные ресурсы актуальной становится задача повышения эффективности их использования. Одним из подходов для решения данной задачи является применение дискретно-событийного моделирования в процессно-сетевой парадигме, позволяющей описать моделируемый процесс в виде узлов-обработчиков, объединенных в единую сеть. Однако данная парадигма не учитывает индивидуальное поведение агентов, что снижает адекватность получаемых моделей. В статье представлен подход к оценке адекватности и значимости имитационных моделей, а также предложен метод, позволяющий задавать и учитывать индивидуальное поведение агентов в процессе симуляции, реализуемой в соответствии с процессно-сетевой парадигмой. Применение данного метода позволяет повысить прикладную значимость моделей исследуемых процессов. В работе описан процесс интеграции предложенного метода в системы, реализующие процессно-сетевую парадигму, а также представлена программная реализация предложенного метода, с помощью которой проведено исследование влияния учета индивидуального поведения агента на адекватность и значимость модели размещения виртуальных машин относительно физических серверов. Благодаря применению метода удалось достичь повышения адекватности исследуемой модели в среднем на 12,5% и, как следствие, повысить количество значимых моделей на 65% для выбранного порогового значения адекватности.

**Ключевые слова:** дискретно-событийное моделирование, агентное моделирование, процессно-сетевая парадигма, адекватность модели, моделирование облачных инфраструктур

**Для цитирования:** Sabutkevich A.M., Nikiforov I.V., Samochadin A.V. A method for modeling of individual agent behavior in the process-network paradigm of discrete-event simulation // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 23–35. DOI: 10.18721/JCSTCS.18102

### Introduction

The use of cloud infrastructures is one of the most promising areas of technology development [1], but the effective utilization of cloud distributed resources is often associated with the solution of NP problems such as determination of optimal resource allocation and defragmentation [2], organization of distributed training of machine learning models [3] and others. Simulation modeling [4], in particular process-network paradigm of the discrete-event simulation [5], is used to solve problems of this class.

The process-network paradigm allows to describe the simulated process in the form of processing nodes connected in a single network, through which agents move during a simulation [6]. Generally, the individual behavior of agents is not considered in discrete-event simulation. However, in order to create adequate and significant models of complex processes in the field of cloud technologies, the ability to consider and further analyze the behavior of agents is necessary.

Consequently, the task of improving the adequacy and significance of models by simulating individual behavior of agents within the process-network paradigm of discrete-event simulation is relevant.



The purpose of this work is to improve the adequacy and significance of complex process models in the cloud technology domain through a method of simulating individual agent behavior in a process-network paradigm implemented in a discrete-event simulation system.

### Metrics of model adequacy and significance

The model adequacy metric determines the degree of closeness of the analyzed model to the corresponding representation of the real-world process, which has a direct impact on the informativeness and applied significance of the obtained simulation results [7].

The comparison of simulation results obtained by the model with the actual results of the behavior of the real process at equal values of input parameters is used to assess adequacy [8]. Actual results are the results obtained experimentally or theoretically, for example, by using heuristics. There are two main types of approaches to assessing model adequacy:

- 1) according to variance of deviations of simulation results from the mean value of actual results;
- 2) according to the average values of simulation and actual results.

The adequacy metric can be calculated based on variance of deviations of simulation results from the mean value of actual results, since the obtained results and initial values of parameters are statistical data, which allows using methods of statistical theory of estimation and hypothesis testing for this comparison [9–11]. The F-test, the Pearson's chi-squared test or the Kolmogorov–Smirnov test can be used for comparison of variances.

However, this approach imposes limitations on the analyzed results considered as statistical data. For example, to use the F-test, as the universal method, which is independent of the sample size, it is necessary to guarantee that the data have a normal distribution.

In this regard, the use of the approach of assessing the adequacy of the model by the variance of deviations of simulation results from the mean value of actual results is not possible within this work, since the resulting data can be distributed in any way.

Assessment of model adequacy by mean values of the results involves checking the proximity of the mean values of each component of the simulation results to the known mean values of the corresponding component of the actual results [12, 13].

This approach will be used in this work, since it does not impose additional restrictions on the analyzed data.

#### *Proposed metrics of adequacy and significance*

As a quantitative indicator of the model adequacy metric, we will use the ratio of actual estimates of the model target metrics determined by the model developer and their values obtained in the simulation process as follows:

$$MA = \frac{\sum_{i=1}^n w_i |v_{t_i} - v_{s_i}|}{\sum_{i=1}^n w_i |v_{t_i}|},$$

where  $n$  is the number of target model metrics selected by the model developer;  $w$  is the weight of each target metric;  $v_t$  is the actual evaluation of the value of the target metrics;  $v_s$  is the value of the target metric obtained during the simulation.

The model significance metric is determined based on the adequacy metric value as follows:

$$MS = \begin{cases} 0, & MA < T \\ 1, & MA \geq T \end{cases},$$

where  $T$  is the model adequacy threshold defined by the model developer.

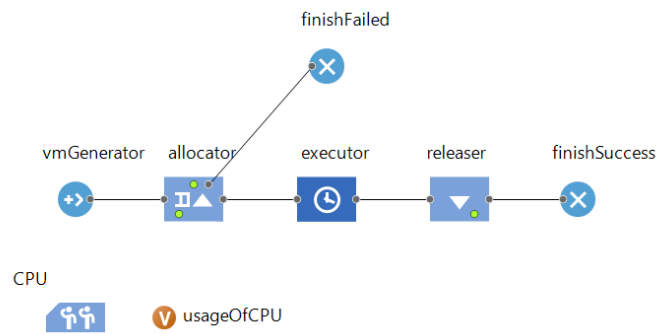


Fig. 1. A simplified model of virtual machines placements in accordance with the process-network paradigm

If the value of the model significance metric takes the value that is equal to one, then the model is considered significant for the selected adequacy threshold.

### Components of the process-network paradigm notation

The simulation systems (AnyLogic [14], Arena [15], SIMAN [16]), which implement discrete-event simulation in the process-network paradigm, are based on similar proprietary notations and languages, which in general can be represented through the components described below.

The process under study, for example, the selection of an optimal route for sending a packet within a physical network, is formally specified by means of a model. A model is an entity that is characterized by general information and aggregates components used to describe the properties and behavior of a process [17]. The same process can be described through different models depending on the requirements to the modeling results.

The main types of model components in considered modeling notations, in accordance with the process-network paradigm, are blocks and links between them, defining in aggregate a directed network [5]. Each block has a behavior specific to its type and the number of ports used to specify the links between blocks. An example of a simplified model of virtual machines placement relative to physical servers, implemented using AnyLogic in accordance with the process-network paradigm, is presented in Fig. 1.

During simulation, blocks can generate events, which are applied to the model at a given point in model time [17]. Each of the events contains a pointer to the handler that will be called at the moment of event application. The handler in the model can be either the block that generated the event or a third-party service that implements the handler interface.

It is assumed that the event changes the state of the model, where the state of the model means the set of values of its properties at a certain point in time.

Another type of components is an agent which is a digital twin of a real-world object [18], characterized by information of general identification and classification nature. The movement of agents within the network is carried out through events that contain pointers to agents. By sequentially creating and processing events by blocks, agents move from generating blocks to absorbing blocks.

Blocks maintain a common interface for interacting with each other, which includes the operations listed below:

- initialization – implements the logic of initializing the initial state of a block when the simulation starts. During initialization events can be created, if no events are created at this stage – the simulation is terminated;
- check of the possibility of moving to a block – determines the possibility of an agent moving to a block based on the position of the currently processed block and the specified network structure. For example, if the number of agents in a block has reached its capacity, it is impossible to move to it;

- transition to a block – implements specific behavior of a computational block, including the creation of events. In the base case – it checks the possibility of transition to the next block, and if the transition is possible, it processes it;

- agent request from a block – notifies the block that one of the next blocks in the network is ready to accept an agent, in other words – a transition can be realized to one of the next blocks in the network.

The simulation process is implemented by auxiliary components: global model time and an array of scheduled events [6]. The event with the shortest application time is selected from the array of scheduled events. The global model time [19] is raised to the application time of the selected event, and then the handler of this event is called. The above steps are repeated until the array of scheduled events is empty or a critical modeling situation occurs.

The result of the simulation is a list of events applied to the model, which can be used to reconstruct the state of the model at each discrete point in time.

Another important result is a set of target metrics values, which are defined by the model developer. These values can be represented in various visual formats, which allows their further analysis to assess the adequacy and significance of the models.

### **Approaches to simulate agent behavior within discrete-event simulation**

#### ***Agent-based simulation***

In the field of simulation modeling, the agent-based modeling method is used to simulate the behavior of agents. Agent-based modeling is the most modern approach of simulation modeling [18]. When using this method, the modeled process is represented by means of a set of agents. The Fig. 2 shows an example of a simplified virtual machine life cycle model. This model is built in accordance with the method of agent-based modeling in AnyLogic environment.

Agents within this approach are autonomous real-world objects selected in some system of relations defined by the modeling goals [20]. It is worth noting that the same model can be represented through a different set of agents. Various notations, including our own, can be used to describe models.

Behavior rules are specified for each agent separately, and the behavior of the whole system is determined based on the result of their interaction.

In a more general case, the task considered in this paper can be reduced to the integration of the agent-based modeling method into the discrete-event modeling system.

#### ***Approaches of integration of agent-based simulation into discrete-event systems***

In [21], an approach based on an initial classification of agents into types is considered. For each type a default behavior is defined, which cannot be overridden by the developer of the model, which is the main disadvantage of this solution.

The implementation of this approach is related to the creation of an additional component called the agent behavior simulator, which provides an interface to interact with the used simulation platform AutoMod [21].

In [22], a mechanism for translating a model built in accordance with the agent-based approach into a discrete-event model for its further simulation is described. The main purpose of applying this mechanism to the original model is to improve performance when simulating high-dimensional models.

The author of the article provides a formal proof that the behavior of agents in a discrete-event model can be implemented by means of a set of events for each of the possible actions of the agent. In addition, the implementation of the action itself, its influence on the state of the model, – by means of external functions. The need to specify an event for each possible action of an agent is a disadvantage of this approach, as it complicates the scalability of the model.

The FAMOS solution described in [23] is an agent-based modeling module for the DESMO-J platform that implements the discrete-event approach. Within this solution, when the state of the model changes, an event is generated and processed by the agent. While processing the event, the agent delegates its state change to another object that encapsulates its behavior.

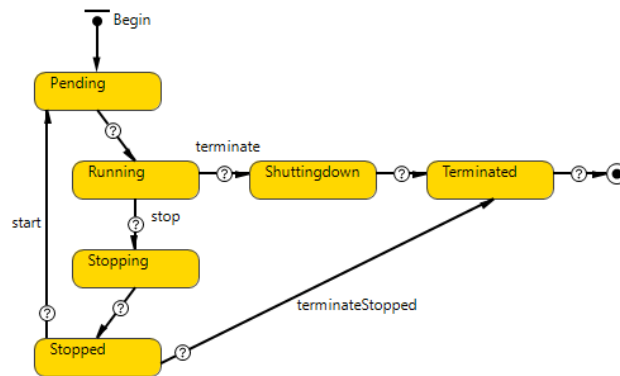


Fig. 2. A simplified model of the virtual machine lifecycle

A common drawback of the considered approaches and solutions is their focus on integration with specific modeling systems. Due to this, as well as the disadvantages mentioned earlier – the impossibility of overriding the behavior of agents and the need to manually set an event for each agent action, the task of developing our own solution is relevant. The solution proposed in this paper allows us to integrate the method of accounting for individual agent behavior into any modeling system supporting the process-network paradigm of the discrete-event approach. The adequacy and significance metrics proposed in this paper, the evaluation of which was not performed in the reviewed works, will allow us to quantitatively evaluate the obtained result of accounting for individual agent behavior in models.

### Proposed method

The method proposed within this work based on:

- extending the modeling notation through properties, classes of states and trajectories of agent behavior to enable individual agent behavior to be considered;
- adding a mechanism for tracking changes in the agent's state using events;
- implementing an algorithm for simulating the agent's behavior in accordance with its life cycle described by means of a state machine.

The application of the proposed method will make it possible to specify and consider the individual behavior of agents in the simulation process, implemented in accordance with the process-network paradigm of discrete-event simulation, in order to improve the adequacy and significance of models.

The method has the following characteristics and properties:

- possibility of application for simulation modeling systems implementing the discrete-event approach;
- independence of implementation from the behavior of a particular type of agents;
- support of any level of detailing of model characteristics.

### Extension of notation

To ensure that the individual behavior of the agent, considered as a digital twin of the real-world object, is considered, the modeling notation has been extended. A description of the changes is presented below.

One of the characteristics of an agent is a finite set of properties. Each agent is characterized by general identification and classification information, as well as a set of related special entities: properties, classes of states, and actions.

Properties describe key attributes of an agent. An agent does not possess all the attributes of a real-world object, as their number may be infinite. The attributes should be selected in such a way that they describe the distinctive features of the real-world object in the most accurate and complete way. The selected features are also determined by the modeling objectives.

Each of the given properties of the agent should take an initial value. In the process of modeling, the value can be changed both under the influence of external information entering the model and aimed at its actualization [24], and through various factors within the model.

A set of properties and their corresponding values at some point in time define a particular state of the agent. Some of the states having insignificant differences in property values can be combined into a single class of states.

A class of states is a set of admissible values specified for the agent's properties, which together specify some expected status of the agent. An agent can be in some class of states only if the values of its properties satisfy all the constraints specified for the related properties of the agent in the considered class of state [4].

Along with the allowed values for the class of states, the identification characteristics are specified, as well as the properties of the class of states that form the local environment for the agents in this class of states.

The change of class of states during simulation takes place by changing the values of agent's properties.

The set of classes of agent states and transitions that determine the possibility of their change can be formally represented in the form of a behavior graph. A behavior graph is an oriented finite graph [25]  $GB = (CS, T)$ , the nodes of which are classes of agent states (set  $CS$ ), and the arcs (set  $T$ ) are uniquely matched to transitions.

An individual behavior trajectory of an agent is a sequence of states classes connected by transitions, formed based on the initial behavior graph and observation of the agent in the process of modeling.

Formally, the behavioral trajectory is a mathematical oriented graph, where each vertex, except for the source and the sink, has a half-degree of output and a half-degree of input [26] equal to one.

#### ***Proposed simulation mechanism***

##### *Tracking agent state changes*

In the course of simulation, the state of the agent changes due to the sequential execution of the logic of behavior of blocks. To track these changes, a new type of events was defined – tracking events. Events of this type are created when the values of the agent property change and contain a pointer to the agent, as well as information about the name of the property, its previous and new values.

##### *Algorithm for simulating agent behavior*

At the top level, the algorithm for simulating the agent's behavior is represented as a state machine [27] describing the agent's life cycle (see Fig. 3). Each of the agent's states within the life cycle is defined by its own algorithm of system behavior.

The state of identification of the initial class of states is the initial state for all agents of the model after starting the modeling process. Based on the initial values of the agents' properties set at the model creation stage, the only possible initial class of states is defined. In case an initial class cannot be defined for an agent or an agent can be placed in several initial classes of states at once, an agent life cycle modeling error occurs. If an initial class of states is defined, the agent enters the state of waiting for transition to another class of states.

The agent remains in state of waiting for transition to other classes of states until the next tracking event containing a pointer to it is processed. Based on the given graph of the agent's behavior and its current class of states, a set of possible classes of states to which this agent can transition is determined. This state can also be considered as an absorbing state, since the agent may not change its class of states during the whole simulation.

From the previously defined set of possible target state classes, the one whose constraints are satisfied by the new values of the agent's properties is selected. This is processed within state of checking attribute values are consistent with classes of states.

If the agent's property values do not satisfy the constraints of any of the selected classes of states, an agent life cycle error occurs. If the agent's property values satisfy the constraints of several classes of states at once, a single class is randomly selected.

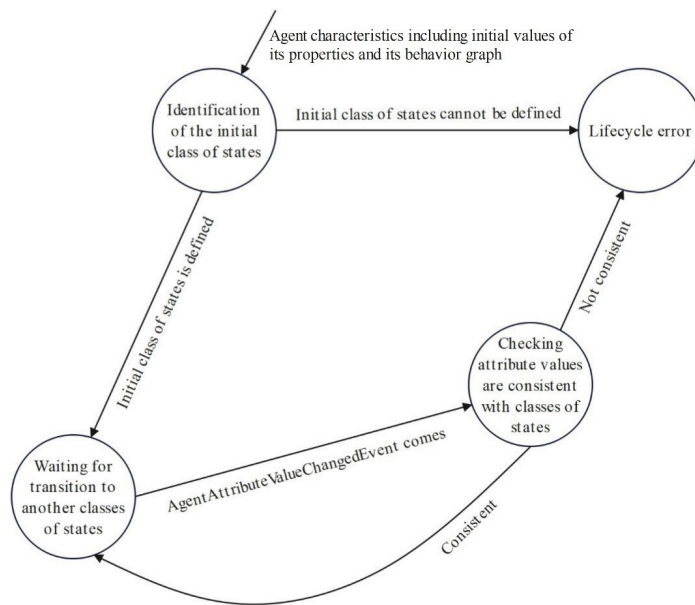


Fig. 3. Agent lifecycle state machine

If the agent is successfully placed in a class of states, it enters the state of waiting for transition to other classes.

### Implementation of the proposed method

In order to integrate the proposed method, a prototype of the discrete-event simulation system was developed. It satisfies the generalized characteristics and requirements for the implemented notation described previously. This decision is explained by the need to test the method in a universal software environment, independent of the peculiarities of the implementation of private solutions and their additional functions.

The prototype of the system is implemented as a console application using the Java 17 program language. The diagram of classes implementing the proposed method as one of the system components is presented in Fig. 4.

#### *Extension of notation within the data model*

The necessary extension of the notation described in previous section involves changing the program data model, in particular, the agent entity.

Agent properties are implemented as generated fields of the corresponding “Agent” class and accessed by the “AgentAttributeValueChangedSupport” wrapper class. Additionally, to access agent properties without using reflection, a collection of pointers to them, stored as a separate attributes field, is implemented. It is not possible to get a value or change an agent property directly.

The class of states is implemented by means of the “ClassOfStates” class, which contains as collections pointers to agent properties and their corresponding ranges of possible values defined by means of the “AbstractRange” class. This class provides an abstract value checking method that must be overridden by the model developer.

Agent behavior is specified at the “Agent” class level as a collection of transitions between classes of states, implemented using the “Transition” class. It contains pointers to the source and target classes of states.

#### *Implementation of the method at the level of the simulation execution*

The event that tracks the change in the value of agent properties is represented by the “AgentAttributeValueChangedEvent” class, which implements a common event interface “Event”. The attributes unique to other events for this event are:

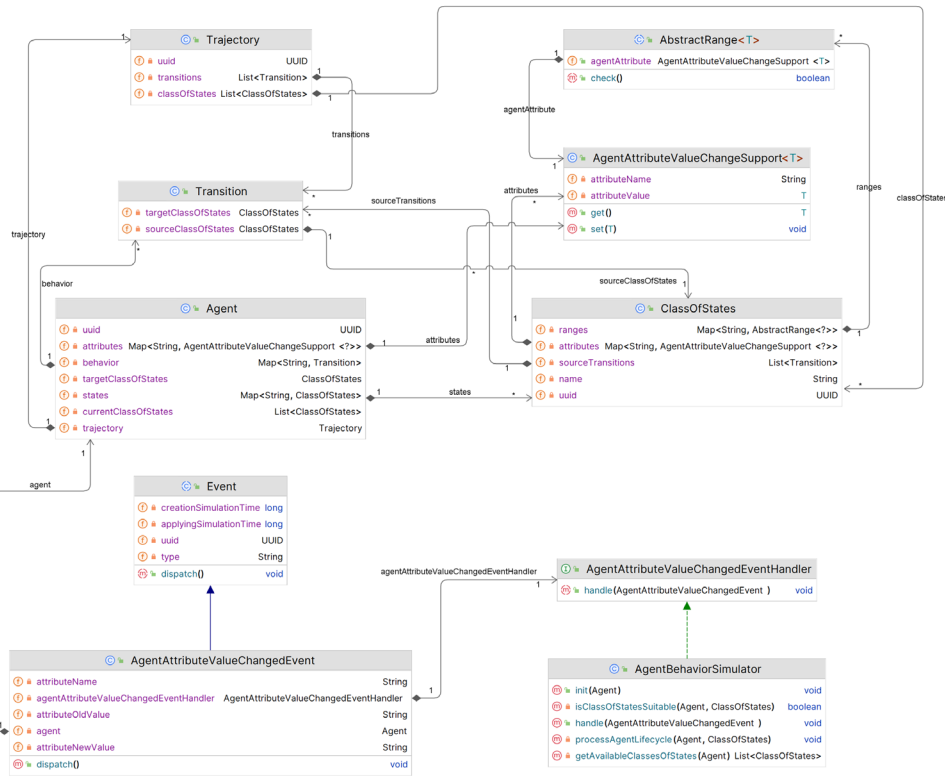


Fig. 4. Diagram of classes implementing the proposed method

- pointer to the handler of this event, whose interface is implemented by the “AgentBehaviorSimulator” class. It is a service that encapsulates the logic of simulating agent behavior according to the method described earlier;
- pointer to the agent whose state has been changed, and hence whose behavior change should be processed;
- name of the agent property whose value has been changed, as well as the previous and new values.

Event creation is implemented through the previously mentioned “AgentAttributeValueChangedSupport”, through which interaction with agent properties is performed.

When the value of an agent property changes, a tracking event is created with an application time that coincides with the model time of its creation. This event is placed in an array of scheduled events. At the same model time, the event is retrieved from the array and processed in the same way as other events in the system by supporting a common interface, namely by calling the “dispatch” method.

Implementation of the “dispatch” method for the tracking event is reduced to calling the handle method of the “AgentBehaviorSimulator” handler, which takes this event as an argument, further extracting the agent from it for subsequent operations according to the method described earlier.

### Application of the proposed method to create an experimental model

In the process-network paradigm of discrete-event simulation, agents do not have individual behavior, and thus are indistinguishable in their state.

One example of a model in which, without considering the individual behavior of agents, the results obtained have little adequacy and significant for the analysis of a real process is the model of virtual machine placement relative to physical servers [28]. This is due to the fact that the server has individual behavior – they may fail, which will affect the configuration of the whole cluster.

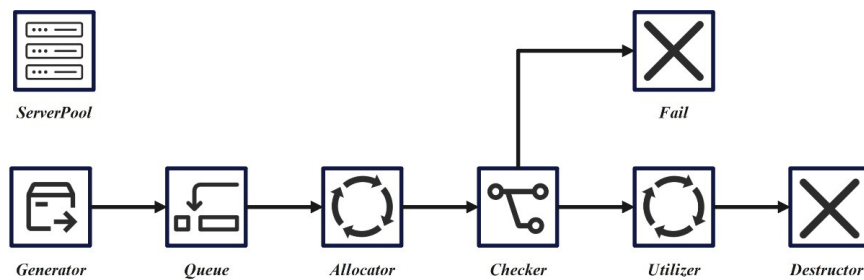


Fig. 5. The scheme of the process-network model of virtual machines placement

The use of simulation to analyze the process of virtual machine placement is significant due to the need to improve resource utilization efficiency, to predict the workload of the physical cluster and to scale it in a timely manner.

The target metric for resource utilization efficiency in this model is the resource allocation ratio, which ranges from 0 to 100%. In case the allocation ratio reaches its maximum value, it indicates a lack of resources and the need to increase the number of servers in the cluster. A low value of the allocation ratio corresponds to the case of inefficient resource utilization and the need to release servers from the cluster.

The model of this process is described by means of a process-network (see Fig. 5), which includes the following components:

- Generator – the initial unit that provides generation of requests from virtual machines to capture resources;
- Queue – realizes the accumulation of requests;
- Allocator – realizes the allocation of computational resources of servers for requests;
- Checker – checks resource allocation. If the resources are allocated - the resources can be used, otherwise an error of request processing occurs;
- Utilizer – delay block that simulates the time of using the captured resources;
- Destructor – releases the captured resources;
- Fail – terminates processing of the request with a resource allocation error;
- Servers pool – collection of available servers.

The presented model does not consider the behavior of agents; therefore, the simulation does not consider the possibility of breakdown or temporary decommissioning of a server defined as an agent within the model.

Considering this feature of server behavior in the process of modeling and further analysis of the results is important for the adequacy of the model, as it has a direct impact on the efficiency of computing resources.

Consequently, a behavior graph can be defined for the server, which describes a set of its possible classes of states. For simplification, the presented graph (see Fig. 6) contains three classes of states: exploited, failed and decommissioned.

Additionally, when attempting to place virtual machines, a check of the current server class of states has been added – it must be exploited. Separately handled cases of changing the server class of states at the moment of its use by a virtual machine in order to reallocate it on another server in case of failure of the current one.

As one of the simulation results, individual server behavior trajectories were generated that describe the sequence of state class changes. Based on these trajectories, an analytical model of cluster operation can be built.

An experiment was conducted for the described model. It consists of single simulation runs in order to calculate the target metric of the model – the resource allocation ratio for different cluster configurations



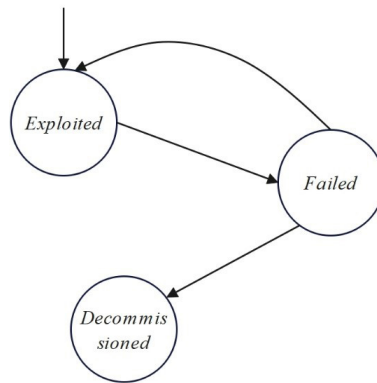


Fig. 6. Server behavior graph

and behaviors, as well as a set of virtual machine allocation requests. In the experiment, we used the adequacy threshold value  $T = 0.9$ , stochastic values are absent, which is explained by the need to fix the initial values of all parameters for accurate analysis of the obtained results. The obtained results are presented in Table 1.

Table 1

**Results of assessment of adequacy and significance for the model,  
which describes the virtual machine placement relative to physical servers**

| No. | Cluster dimension | Number of requests | Number of server failures | Adequacy       |             | Significance   |             |
|-----|-------------------|--------------------|---------------------------|----------------|-------------|----------------|-------------|
|     |                   |                    |                           | Without method | With method | Without method | With method |
| 1   | 12                | 2500               | 0                         | 97%            | 97%         | 1              | 1           |
| 2   | 12                | 2500               | 3                         | 89%            | 95%         | 0              | 1           |
| 3   | 40                | 10000              | 14                        | 76%            | 95%         | 0              | 1           |

Based on the table we can conclude that for configuration No. 1, where there were no server failures, the adequacy and significance estimates coincide regardless of the application of the method. However, for configurations No. 2 and 3, where there is a need to consider the individual behavior of servers, it was possible to achieve an increase in the adequacy of the model on average by 12.5% and consequently increase the number of significant models by 65% for the selected threshold value of adequacy.

### Conclusion

This paper proposes a method for simulating individual agent behavior in the process-network paradigm of discrete-event simulation. This method consists in extending the modeling notation by introducing entities that collectively describe the individual behavior of agents, as well as modifying the simulation algorithm.

The paper presents model adequacy and significance metrics to assess the degree of closeness of the model under study to the corresponding representation of the real-world process.

The application of the proposed method to the experimental model of virtual machines placement relative to physical servers demonstrated an increase in the model adequacy by 12.5%, and increased the number of significant models by 65%.

## REFERENCES

1. **Hedge R.R., Narayan D.R., Moolya S., Chethan, Pushparani M.K.** A review on the future of technology: How cloud computing is changing the game. *International Research Journal on Advanced Engineering Hub (IRJAEH)*, 2024, Vol. 2, No. 6, Pp. 1784–1793. DOI: 10.47392/IRJAEH.2024.0245
2. **Bohez S., Verbelen T., Simoens P., Dhoedt B.** Discrete-event simulation for efficient and stable resource allocation in collaborative mobile cloudlets. *Simulation Modelling Practice and Theory*, 2015, Vol. 50, Pp. 109–129. DOI: 10.1016/j.simpat.2014.05.006
3. **Hasan S.S., Zeebaree S.R.M.** Distributed systems for machine learning in cloud computing: A review of scalable and efficient training and inference. *Indonesian Journal of Computer Science*, 2024, Vol. 13, No. 2, Pp. 1685–1707. DOI: 10.33022/ijcs.v13i2.3814
4. **Sabutkevich A.M., Vikhlyayev D.A., Nikiforov I.V., Samochadin A.V.** Imitatsionnoye modelirovaniye povedeniya slozhnykh mnogoagentnykh sistem s ispolzovaniyem veroyatnostnoy modeli. [Simulation of behavior of complex multi-agent systems using probabilistic model]. *Sovremennye tekhnologii v teorii i praktike programirovaniya [Modern technologies in theory and practice of programming]*, 2022, Pp. 98–100.
5. **Wagner G.** Introduction to information and process modeling for simulation. *2017 Winter Simulation Conference (WSC)*, 2017, Pp. 520–534. DOI: 10.1109/WSC.2017.8247812
6. **Wagner G.** An abstract state machine semantics for discrete event simulation. *2017 Winter Simulation Conference (WSC)*, 2017, Pp. 762–773. DOI: 10.1109/WSC.2017.8247830
7. **Robinson S.** Exploring the relationship between simulation model accuracy and complexity. *Journal of the Operational Research Society*, 2023, Vol. 74, No. 9, Pp. 1992–2011. DOI: 10.1080/01605682.2022.2122740
8. **Ilin V.A., Kiryushov N.P.** Method of testing the training model for the adequacy. *Software & Systems*, 2021, Vol. 34, No. 1, Pp. 61–66. DOI: 10.15827/0236-235X.133.061-066
9. **Potapov A.N., Ovcharov B.B.** Assessment of the adequacy of simulation in simulators, èrgotekhnicheskikh operators with hierarchical structure of building. *Bulletin of Voronezh State Technical University*, 2013, Vol. 9, No. 3–1, Pp. 45–48.
10. **Belyakov V.V., Tumasov A.V., Butin D.A., Vashurin A.S.** Adequacy simulation model of a light commercial car. *Trudy NGTU im. R.E. Alekseeva [Proceedings of NSTU n. a. R.E. Alekseev]*. 2021, Vol. 132, No. 1, Pp. 62–69. DOI: 10.46960/1816-210X\_2021\_1\_62
11. **Spear R.C.** Large simulation models: calibration, uniqueness and goodness of fit. *Environmental Modelling & Software*, 1997, Vol. 12, No. 2–3, Pp. 219–228. DOI: 10.1016/S1364-8152(97)00014-5
12. **Minaev V.A., Stepanov R.O., Faddeev A.O.** Modeling of energy transition in a stress-strain geological environment for seismic risk assessment (Part 1). *Modeling, Optimization and Information Technology*, 2022, Vol. 10, No. 1, Pp. 1–19. DOI: 10.26102/2310-6018/2022.36.1.007
13. **Godunov A.I., Brostilov A.N.** Statisticheskiye kriterii otsenki adekvatnosti imitatsionnogo modelirovaniya v trenazherostroyenii. [Statistical criteria for assessing the adequacy of simulation modeling in simulator building]. *International Symposium “Reliability and Quality”*, 2005, Pp. 161–163.
14. **Borshchev A.V., Karpov Y.G., Kharitonov V.V.** Distributed simulation of hybrid systems with AnyLogic and HLA. *Future Generation Computer Systems*, 2002, Vol. 18, No. 6, Pp. 829–839. DOI: 10.1016/S0167-739X(02)00055-9
15. **Dias A.S.M.E, Antunes R.M.G, Abreu A., Anes V., Navas H.V.G., Morgado T., Calado J.M.F.** Utilization of the Arena simulation software and Lean improvements in the management of metal surface treatment processes. *Procedia Computer Science*, 2022, Vol. 204, Pp. 140–147. DOI: 10.1016/j.procs.2022.08.017
16. **Wagner G., Seck M., McKenzie F.** Process modeling for simulation: Observations and open issues. *2016 Winter Simulation Conference (WSC)*, 2016, pp. 1072–1083. DOI: 10.1109/WSC.2016.7822166
17. **Borshchev A.V.** Prakticheskoye agentnoye modelirovaniye i yego mesto v arsenale analitika. [Practical agent modeling and its place in the analyst's arsenal]. *Exponenta Pro*, 2004, Vol. 3–4, Pp. 38–47.

18. **Jabri A., Zayed T.** Agent-based modeling and simulation of earthmoving operations. *Automation in Construction*, 2017, Vol. 81, Pp. 210–223. DOI: 10.1016/j.autcon.2017.06.017
19. **Okol'nishnikov V.V.** Time representation in imitational modelling. *Computational Technologies*, 2005, Vol. 10, No. 5, Pp. 57–80.
20. **Gorodetskiy V.I., Karsayev O.V., Samoylov V.V., Konyushiy V.G.** Yazyk opisaniya mnogoagentnykh sistem. [Multi-agent system description language]. *Journal of Instrument Engineering*, 2008, Vol. 51, No. 11, Pp. 7–12.
21. **Dubiel B., Tsimhoni O.** Integrating agent based modeling into a discrete event simulation. *2005 Winter Simulation Conference (WSC)*, 2005, pp. 1029–1037. DOI: 10.1109/WSC.2005.1574355
22. **Onggo B.S.** Running agent-based models on a discrete-event simulator. *Proceedings of the 24<sup>th</sup> European Simulation and Modelling Conference*, 2010, pp. 51–55.
23. **Page B., Knaak N., Kruse A.** A discrete event simulation framework for agent-based modelling of logistic systems. *INFORMATIK 2007*, 2007, pp. 397–404.
24. **Gorodetskiy V.I.** Printsipy avtonomnogo gruppovogo upravleniya [Principles of Autonomous Group Management]. *Integrirovannye modeli i myagkie vychisleniya v iskusstvennom intellekte [Integrated Models and Soft Computing in Artificial Intelligence]* (IMMV-2021), 2021, Vol. 1, Pp. 16–48.
25. **Riensch A., Severson J., Yavari R., Piercy N.L., Cole K.D., Rao P.** Thermal modeling of directed energy deposition additive manufacturing using graph theory. *Rapid Prototyping Journal*, 2023, Vol. 29, No. 2, Pp. 324–343. DOI: 10.1108/RPJ-07-2021-0184
26. **Nikiforov I.V., Petrov A.V., Yusupov Yu.V., Kotlyarov V.P.** Usage of formalization approaches for creation of system models from UCM-specification. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems*, 2011, Vol. 126, No. 3, Pp. 180–184.
27. **Sakellariou I.** Agent based modelling and simulation using state machines. *Proceedings of the 2<sup>nd</sup> International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2012)*, 2012, Vol. 1., Pp. 270–279. DOI: 10.5220/0004164802700279
28. **Dong D.** Agent-based cloud simulation model for resource management. *Journal of Cloud Computing*, 2023, Vol. 12, Art. no. 156. DOI: 10.1186/s13677-023-00540-5

#### INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

**Sabutkevich Artem M.**  
**Сабуткевич Артем Михайлович**  
 E-mail: artem.sabut@gmail.com

**Nikiforov Igor V.**  
**Никифоров Игорь Валерьевич**  
 E-mail: igor.nikiforovv@gmail.com

**Samochadin Alexander V.**  
**Самочадин Александр Викторович**  
 E-mail: samochadin@gmail.com

*Submitted: 08.11.2024; Approved: 10.02.2025; Accepted: 13.03.2025.*

*Поступила: 08.11.2024; Одобрена: 10.02.2025; Принята: 13.03.2025.*

Research article

DOI: <https://doi.org/10.18721/JCSTCS.18103>

UDC 681.5.08



## A TECHNIQUE FOR AUTOMATED ANALYSIS OF THE BLADE SURFACE FOR DEFECTS UNDER UV LIGHT

*E.A. Alekseev*<sup>1</sup> , *A.N. Lomanov*<sup>2</sup> , *D.S. Ivanov*<sup>1</sup>

<sup>1</sup> PJSC “UEC-Saturn”, Rybinsk, Russian Federation;

<sup>2</sup> P.A. Solovyov Rybinsk State Aviation Technical University,  
Rybinsk, Russian Federation

 [evgeny.alekseev@uec-saturn.ru](mailto:evgeny.alekseev@uec-saturn.ru)

**Abstract.** A technique for automated analysis of the blade surface for defects under UV light is presented. The basis of control operations when inspecting blade surfaces is the use of machine vision. The technique solves several key problems: obtaining a package of inspection images of a complex profile object of inspection (an aircraft blade), determining the actual parameters (sizes) of glows for single and group defects, generating expert recommendations (digital trace) for determining the presence of defects on the inspected surfaces for the operator or automated systems. An algorithm for processing images obtained from a video camera is presented, and approaches to compensating for the shift of blades in a frame during inspection rotation are described. The technique describes the following sequentially performed stages: shooting of the blade surface; searching for glows in a two-dimensional image; converting two-dimensional coordinates of the glows into three-dimensional ones; determining the actual parameters of the glows; determining the position of the glows relative to each other; determining the degree of suitability of the blade based on the obtained information about the glows.

**Keywords:** non-destructive testing, machine vision, technological process automation, measurement systems, recommendation system

**Citation:** Alekseev E.A., Lomanov A.N., Ivanov D.S. A technique for automated analysis of the blade surface for defects under UV light. *Computing, Telecommunications and Control*, 2025, Vol. 18, No. 1, Pp. 36–47. DOI: 10.18721/JCSTCS.18103



Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18103>

УДК 681.5.08



## МЕТОДИКА АВТОМАТИЗИРОВАННОГО АНАЛИЗА ПОВЕРХНОСТИ ЛОПАТКИ НА НАЛИЧИЕ ДЕФЕКТОВ ПОД УФ-СВЕТОМ

Е.А. Алексеев<sup>1</sup> , А.Н. Ломанов<sup>2</sup> , Д.С. Иванов<sup>1</sup>

<sup>1</sup> ПАО «ОДК-Сатурн», Рыбинск, Российская Федерация;

<sup>2</sup> Рыбинский государственный авиационный технический университет  
имени П.А. Соловьева, Рыбинск, Российская Федерация

 [evgeny.alekseev@uec-saturn.ru](mailto:evgeny.alekseev@uec-saturn.ru)

**Аннотация.** Представлена методика автоматизированного анализа поверхности лопатки на наличие дефектов под УФ-светом. В основе контрольных операций при осмотре поверхностей лопатки лежит использование машинного зрения. Методика решает несколько ключевых задач: получение пакета инспекционных изображений сложного профильного объекта контроля (авиационной лопатки), определение реальных параметров (размеров) свечений для единичных и групповых дефектов, формирование экспертных рекомендаций (цифрового следа) по определению наличия дефектов на инспектируемых поверхностях для оператора или автоматизированных систем. Приведен алгоритм обработки изображения, получаемого с видеокамеры, описаны подходы по компенсации сдвига лопаток в кадре при инспекционном вращении. Методика описывает следующие последовательно выполняемые этапы: съемка поверхности лопатки, поиск свечений на двумерном изображении, преобразование двумерных координат свечений в трехмерные, определение реальных параметров свечений, определение положения свечений друг относительно друга, определение степени годности лопатки по полученной информации о свечениях.

**Ключевые слова:** неразрушающий контроль, машинное зрение, автоматизация технологических процессов, системы измерения, рекомендательная система

**Для цитирования:** Alekseev E.A., Lomanov A.N., Ivanov D.S. A technique for automated analysis of the blade surface for defects under UV light // Computing, Telecommunications and Control. 2025. T. 18, № 1. С. 36–47. DOI: 10.18721/JCSTCS.18103

### Introduction

In the aviation industry, methods that help detect surface material discontinuities are widely used during inspection [1, 2–6]. One of them is the capillary method, a non-destructive testing technique [7]. The fluorescent penetrant inspection, being a subtype of this method, exhibits a high sensitivity to the size of defects [8]. The main feature of this type of inspection is filling the surface discontinuities in the material of objects under inspection with fluorescent liquid that has high penetration capability, followed by recording the obtained readings under UV light at the place of the defect visually or using optical devices<sup>1</sup> [9, 10].

The manual implementation of this method, i.e. without the use of automated systems, has a number of disadvantages. The main ones are the following:

- low operating speed (approximately 1 blade per 3 minutes);
- impossibility of performing inspections in the evening and at night due to the reduced attention of flaw detector operators, which significantly reduces the amount of manufactured products.

In order to address the problem of blade surface analysis in this article, a multi-step approach is proposed, which includes the following:

- shooting of the blade surface;

<sup>1</sup> Literature and journals on capillary testing, Available: <https://ndt-testing.ru/literature.html> (Accessed: 18.12.2024)

- searching for indications in a two-dimensional (2D) image;
- converting 2D coordinates of indications into three-dimensional (3D) ones;
- determining the actual parameters of indications;
- determining the position of indications relative to each other;
- determining the degree of suitability of the blade based on the obtained information about the indications.

The defects detected by fluorescent penetrant inspection are very small (from 300  $\mu\text{m}$ ) and barely visible, but under UV light they become clearly visible and contrast strongly with the blade surface<sup>2</sup>. Since the defects produce fluorescence in the visible range, for the needs of this article a camera with a CMOS matrix for high-resolution color images was used to detect the defects. The blade should be photographed from different angles to cover its entire surface, for this reason, a device for rotating the blade was used.

### Problem formulation and method of solution

When determining the actual shape of indications on the blade surface, the method of converting 2D coordinates of indications into 3D ones, is used. In order to implement this method, the obtained image was compared with the reference image, where each pixel is characterized by its own 3D coordinates on the blade surface [11, 12, 13]. Once all the characteristics of indications were identified, they were classified and the degree of suitability of the blade was determined according to the developed algorithms.

There are many methods for determining the size of objects in an image [14–20]. The main task of all these methods is to identify coordinates of the points of measured objects in 3D space.

The most common methods are based on stereoscopic vision [21–25]. It works like the human eye. The control point, the coordinates of which must be determined, will be in different positions in each camera thereby enabling the system to calculate the position of that point in space. The main advantage of this method are its relative cheapness, a large number of implementations, and the fact that systems using this method can immediately capture high-resolution images (e.g., for data storage). The main disadvantage of this method is that it is necessary to search for pixels representing the same point in two images in order to determine the coordinates of the desired point. If the object being photographed has a monotonous surface, then this search is performed inaccurately, resulting in frequent errors. In this case, an analysis of the blade surface, which is monotonous – gray, would be required. Therefore, this method should not be used in this situation.

In addition to stereoscopic vision, time-of-flight (TOF) cameras can be used [26–29]. The camera illuminates the scene with a modulated light source and observes the reflected light. The phase shift between emission and reflection is then measured and converted into distance. As a result of shooting, the camera provides a distance image for each object. The distances between the desired points can be calculated from the image data. The disadvantage of such cameras is that they do not provide a color image where the defects themselves need to be identified. Therefore, this method requires an additional camera that can capture a colored image of the blade, which is necessary for comparison with the distance image. Furthermore, this method has a relatively low accuracy. The most expensive camera options allow achieving an accuracy of up to 5 mm.

In order to solve the problem, a method is used that, using one camera and a 3D model of an object, will allow determining the size of defects located on its surface [25]. As part of this research of the method for searching the blade surface indications, several problems were solved.

The first problem that needs to be solved to determine the size and position of the indications is finding them in the image. For this purpose, the next obtained image of the blade undergoes multistage processing, namely:

- compensation for the blade shift in the frame;

<sup>2</sup> Literature and journals on capillary testing, Available: <https://ndt-testing.ru/literature.html> (Accessed: 18.12.2024)

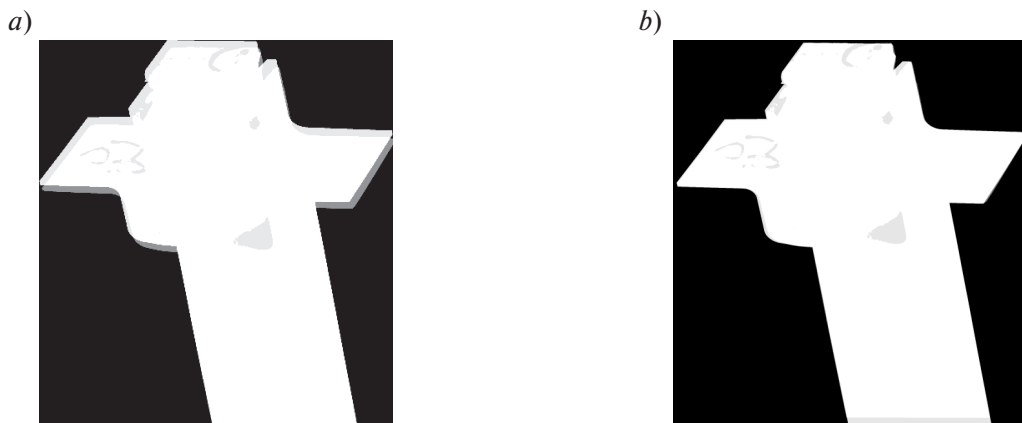


Fig. 1. Result of the blade shift algorithm in the frame:  
 a) silhouette of the blade before shift relative to the standard;  
 b) silhouette of the blade after shift relative to the standard

- application of filters that highlight the indication.

First, we determine whether the blade in the frame is shifted relative to the specified position. To do this, its boundaries are calculated using the Canny operator, and then their shift relative to the reference position obtained during the preparation stage is calculated. If the blade in the image is shifted, then it is aligned using affine transformations. If a very large shift is required (more than 2 mm in any direction), then the transformation is not performed, because it may reduce the image detail. The result of the blade shift algorithm in the frame is shown in Fig. 1.

Once the required position has been determined, filters are sequentially applied to the image to highlight the indications, as follows:

- Gaussian blur is applied to suppress noise that may occur due to the use of a color image;
- HSV transformation plus filtering is applied to transform the color palette of a BGR image into HSV; a color threshold filter leaves only those areas that are green;
- threshold filter that enables only bright indications, cutting off the dim ones according to the threshold value;
- morphological transformations (erode and dilate) are used to suppress very small indications (noise), that do not require consideration during analysis;
- clustering plus filtering for clustering of indications using DBSCAN algorithm, determination of median brightness of clusters, filtering by threshold value of median brightness.

After applying the described filters to the image, only the areas with indications remain on it. The indication areas themselves represent an array of pixel coordinates painted white. These areas should be saved and then processed using an algorithm for determining the indication size and position. The result of indication filtering on the image is shown in Fig. 2.

The main problem in determining the size of the indications is that they are located on the surfaces with intricate shape. Each indication in the image is represented by an array of pixels. The size of each array can be calculated, but the values will be obtained in pixels, which do not directly correlate with actual values, for example, millimeters. In addition, all indications are on a curved surface, and the resulting image is 2D, which means that the indications are flat, which does not allow to capture their actual shape. This means that it is practically impossible to calculate the real size and shape of each indication with high accuracy.

In order to solve the problem, image correlation maps are used in the process. For each obtained image, a prepared data set is generated in advance, enabling to correlate each pixel of the image to a point on the blade in 3D space.

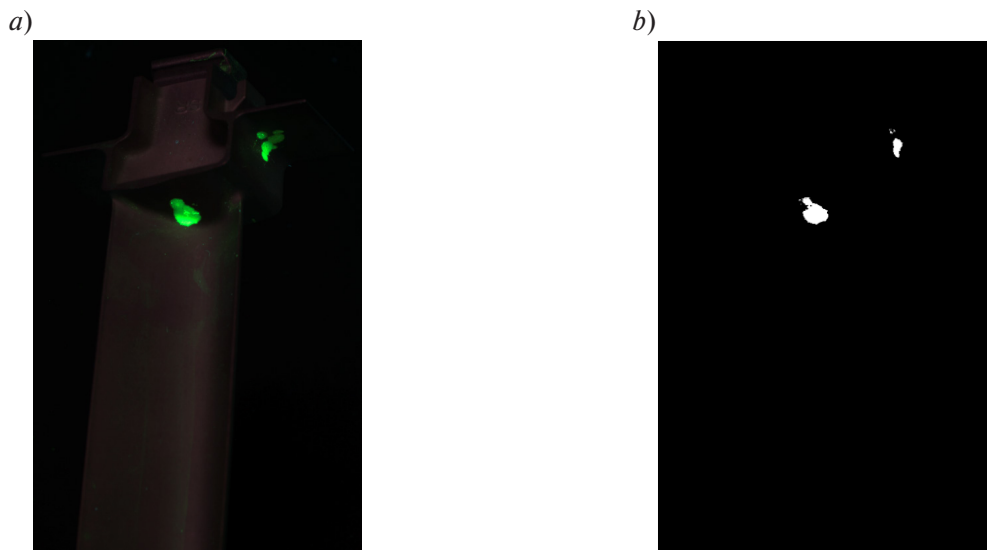


Fig. 2. Result of indication filtering in the image:  
 a) image before filtering; b) image after filtering

For the process of generating correlation maps, the developed program was used, which is based on the *OpenGL* specification for simplifying the execution of operations in 3D space. The *PyOpenGL* library is used to implement this specification.

In order to generate an image correlation map for an image, the following steps must be performed:

- 3D pre-zoned model of the blade is loaded into the developed program (the zones are necessary for further analysis);
- 3D scene with the loaded blade model is displayed;
- the camera is positioned in such a manner that the view from the actual image matches that in the scene;
- compilation of a map is launched, which relates the coordinates of each pixel and their coordinates in 3D space.

Thus, image correlation maps are prepared for each position of the blade shooting. This completes the preparatory stage of the work, after which it is possible to launch the algorithm for determining the size of the indications.

The result of this algorithm is an array of indication parameters, namely:

- indication length, mm;
- indication width, mm;
- mean normal vector of indications;
- center of mass of indications;
- blade's number of the zone (or zones) containing an indication;
- distance from the center of mass to its farthest point (radius of the inscribed circle);
- angle between the camera normal and the surface normal (for further calculation of the distance between the indications and duplicates removal);
- coordinates of the square inscribed in the indication in the original image (for further displaying of data to NDT operator).

Specially prepared image correlation maps are used to transform 2D coordinates of indications into 3D ones.

The block diagram of the algorithm of indication parameters identification method is presented in Fig. 3.



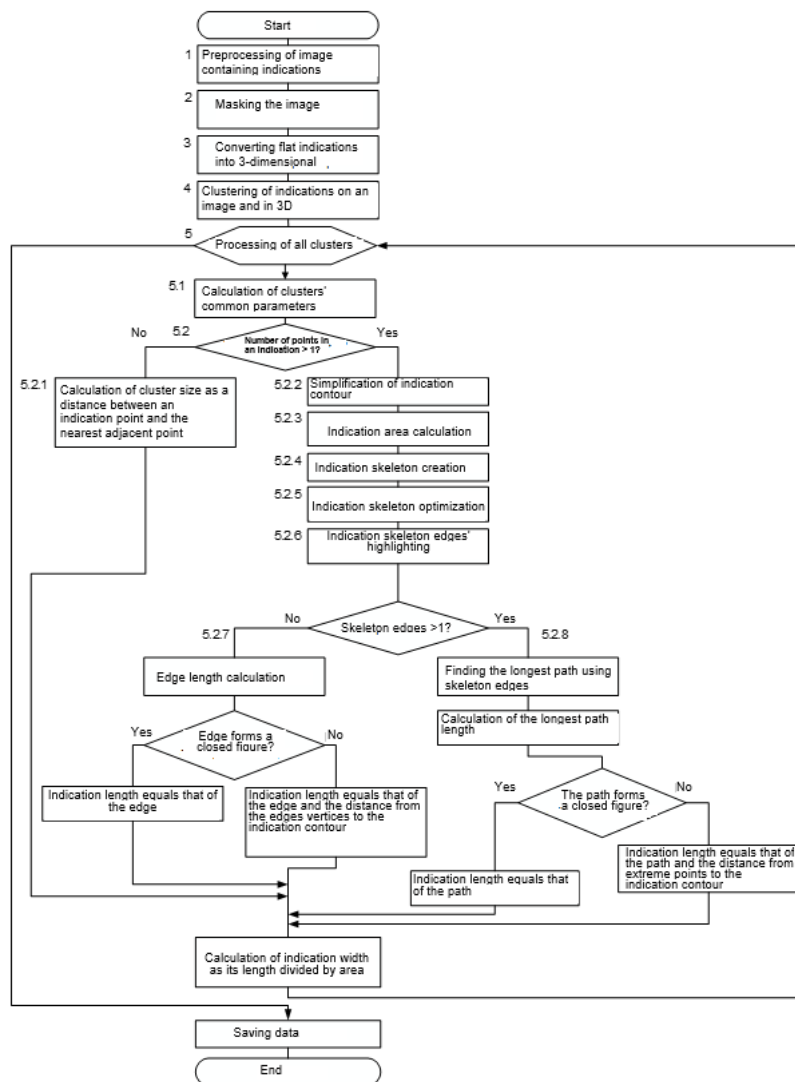


Fig. 3. Algorithm of indication parameters identification method

During digital processing of indication parameters, five main points of the algorithm are consistently implemented:

- 1) image preprocessing;
- 2) masking the image;
- 3) transformation of 2D indications into 3D ones;
- 4) clustering of indications;
- 5) processing of all clusters.

Explanation of each point is given below.

1. The input to the algorithm is a monochromatic image comprising of white areas that serve as indications. In the image, the blade does not take up the entire area, so the area with the blade is cut out. The square area with the blade is cut out to optimize speed and memory space.

2. If indications on the background of the blade were detected (a penetrant drop on the background), they will be ignored. For this purpose, the area obtained in the first step is multiplied by the mask, which is the silhouette of the blade. Thus, only those indications remain that are located on the blade.

3. 2D coordinates of indications are converted into 3D ones according to the prepared image correlation maps.

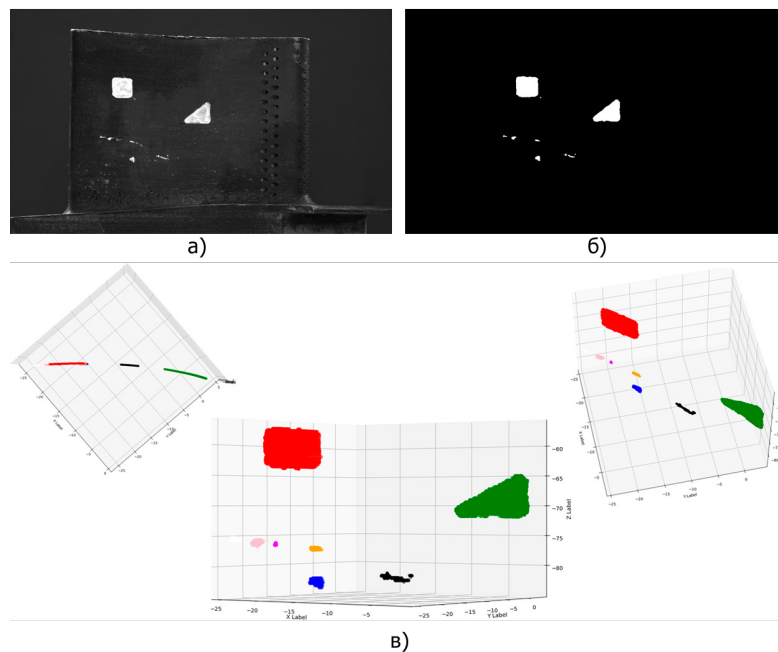


Fig. 4. Algorithm for determining the size and position of indications in space in graphical form:  
 a) original image; b) result of the algorithm for searching for image indications;  
 c) result of clustering of defects in 3D space

4. Clustering of indications is performed using DBSCAN algorithm in 2D and 3D space. In 2D space, it is performed in order to merge nearby pixels into one cluster (assumed indication), and then in 3D space – to determine the actual indications and to isolate them.

For 2D clustering, the following parameters are used:

- *eps* (maximum distance between points to form a cluster). The used distance value  $\sqrt{2} * 1,05$  is chosen empirically (with the minimum distance between pixels and a slightly greater distance to accommodate diagonal ones, but not to include pixels through which the diagonal passes).
- *min\_sample* (minimum number of points to form a cluster). The smallest cluster can consist of 1 point. This number was chosen empirically, based on the fact that very small defects need to be detected.

For 3D clustering, the following parameters are used:

- *eps* – value 1 is used as a distance, which corresponds to 1 mm;
- *min\_sample* – the smallest cluster can consist of 1 point.

The result of the first four points of the algorithm is shown in Fig. 4.

Next, the largest section of a 3D cluster (if any) is selected from each 2D cluster. That is, if several indications have merged into one, they will be divided into several clusters in 3D representation, from which the one with the largest number of points will be selected.

5. For all clusters that are periodically processed in the analysis phase, a series of computational procedures is performed.

5.1. Computation of cluster parameters:

- calculation of actual 3D coordinates (clustering taken into account);
- calculation of the center of indications in 3D space (center of mass);
- calculation of coordinates of the extreme point of indications from its center of mass;
- identification of the zone (or zones) on the blade where the indication is found;
- calculation of the normal vector of each point of indications, and calculation of the mean normal vector of indications;

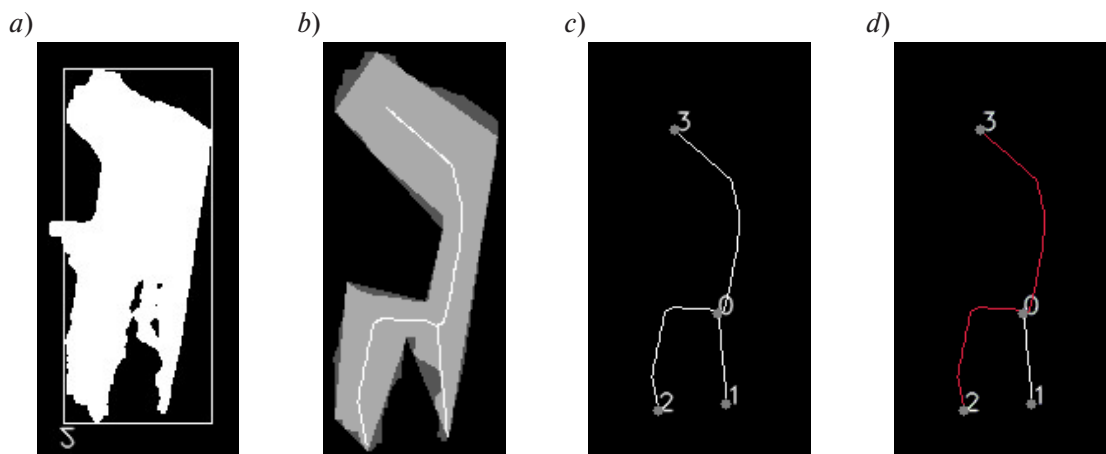


Fig. 5. Visualization of the algorithm for determining the indication length:  
 a) points forming one cluster (indication); b) result of simplification of the indication contour;  
 c) result of skeletonization of indications; d) maximum path that forms the length of indications

- calculation of the angle between the mean normal vector of indications and the normal vector of the camera.

5.2. Next, the geometric dimensions of the indications are calculated:

- indication is represented by a single pixel, its size is considered as the minimum distance to the adjacent pixel. For this purpose, the closest pixel in 3D space is selected from four adjacent pixels;
- if an indication is represented by more than one pixel, its set of points (a list of pixel coordinates) is converted into a binary matrix (image), “cleaned” from various image defects (noise, etc.) using morphological operations. After “cleaning”, the indications contour is determined and simplified by using an approximating function that reduces the number of isolated indication contours (segments) by 25 times. The coefficient is calculated empirically;
- area and perimeter of the contour (original, not simplified) are calculated;
- simplified contour is re-filled, and then skeletonized, which causes the edges of the binary matrix to erode until only the center lines remain. In some figures, such as a circle, the skeletonizing may leave no points, in which case the point is set at the center;
- the skeleton is processed – the indication skeleton is optimized by excluding all unnecessary points from it;
- skeleton is broken into segments (edges) at the junctions of several branches. The obtained edges are processed, their endpoints and lengths are calculated, deleted points are restored in case of indications skeleton ruptures;
- if, as a result, there is one segment left, that segment is checked to see if it is a closed one. If so, the segment length is used as the length. If the segment is not a closed one, then the distance to the contour is added to its length. In both cases, the width is calculated as the area divided by the resulting length;
- if segments are more than one, the matrix of distances between the vertices of segments is calculated, the longest path from which the length of indications is calculated as the sum of lengths of all segments of the path. Then proceed similarly to previous action.

Fig. 5 provides a visualization of the algorithm for determining the indication length.

All the calculated parameters for each of the indications are stored in one array.

#### Analysis of the results

When testing the algorithm for determining the size of the defects, it was found that it works best when the surface is inspected perpendicular to the surface. The closer the inspection angle (the smallest angle

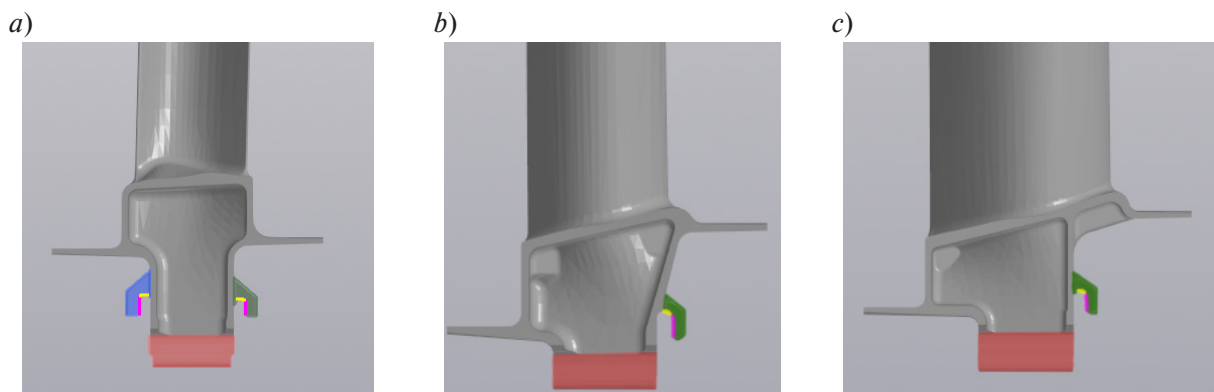


Fig. 6. Blind spots on the blade: a) 2<sup>nd</sup> stage; b) 3<sup>rd</sup> stage; c) 4<sup>th</sup> stage

between the surface plane and the beam from the camera to a point on the surface under inspection) is to  $90^\circ$ , the more accurate are the results. Accordingly, the farther away you go, the less accurate it becomes. This is because the shape of the indications and the surface to be captured are severely deformed when captured at an acute angle. This causes the area of both the surface and the indications in the image to change. This means that the indication starts to occupy a larger area, and therefore its size is calculated incorrectly. In terms of the algorithm, the problem is that when a surface is taken at a right angle, one pixel in the image falls into only one or two polygons on the 3D model, which is used to calculate the size. The smaller the angle of the inspection, the more polygons one pixel starts to occupy, so its size also increases. In this article, areas that cannot be captured at an angle or close to  $90^\circ$ , are called “blind”.

In order to avoid blind spots on the surface of the item under inspection, the following solutions are possible:

- adjust the capture positions so that all surfaces of interest are captured at right angles;
- increase the camera resolution and the number of polygons of the model;
- if possible, inspect hard-to-reach areas of the part separately from the part (e.g. before they are welded together).

An example of difficult to capture zones, as well as “blind” zones, should be given with reference to the blades analyzed in this article. Fig. 6 shows the most difficult areas to inspect, with 2.4.5a showing the 2<sup>nd</sup> stage, 2.4.5b showing the 3<sup>rd</sup> stage, and 2.4.5c showing the 4<sup>th</sup> stage.

The lowest part of the lock (highlighted in red) has a rounded shape and, therefore, its base cannot be captured at a right angle with a single shot. Based on visual analysis of the images, it was concluded that if the number of capture positions is increased, the entire area with the correct viewing angle will be captured.

The hooks on both sides of the lock are the most difficult to capture. The entire outer part of the hooks is captured by the system without any problems (green and dark-blue areas). Inspection of the inner surface of the hooks is particularly difficult.

The lower part (yellow area) is examined in the photo in its entirety, but its viewing angle goes further than  $90^\circ$ , so its measurements are less accurate.

At the inner wall (pink area), the entire area cannot be inspected at right angle, since the lock parts obstruct the view. This surface can only be viewed at an angle of approximately  $28^\circ$  (calculated from 3D model of the blade). This angle is very small to ensure accurate measurements.

### Conclusion

When analyzing the blade surface it was found that defects can be detected, classified and sized on 98% of the surface area using machine vision. In the yellow and pink areas, defects are visible, but it is difficult to classify and size defects using automated methods, because the  $90^\circ$  viewing angle cannot be

maintained according to the procedure. The probability of defects in these zones has not been assessed, however, according to the consultation with the NDT operators, defects in the yellow and pink zones are detected not more than once per 100–200 defective blades. The size of defects is determined visually by the NDT operator, referring to the standard.

Based on the analysis we can conclude that it is possible to use machine vision based automated control means to identify defects under UV light. Sizing of defects from 100 microns is possible. The problem of impossibility of sizing the defects in “blind” spots can be solved by rejection of parts containing any detected indications.

## REFERENCES

1. Byrkov I.A., Vyzhletsov V.V., Kozhanov N.Yu., Mishin S.A., Okov I.N. *Method for processing images by convolutional neural networks*. Patent Russia No. RU 2771442 C1, 2020.
2. Nesterov A.V. *Programmno-apparatnyj kompleks mashinnogo zreniya dlya opredeleniya i kontrolya shiriny mezhvitkovogo zazora [Machine vision hardware and software system for determining and controlling the width of the interturn gap]*. Patent Russia No. RU 126490 U1, 2013.
3. Kilian K., Mazur V., Fan K. *System for machine vision, which enables to determine depth non-uniformity of image objects*. Patent Russia No. RU 2604168 C2, 2016.
4. Ono K., Tate M. *Method for detecting surface defects, device for detecting surface defects, method for producing steel materials, method for steel material quality control, steel materials production plant, method for generating models for determining surface defects and a model for determining surface defects*. Patent Russia No. RU 2764644 C1, 2022.
5. Nigmatullin F.T., Sorvin I.I. *Method of non-destructive optical and visual testing of items by means of computer vision*. Patent Russia No. RU 2777718 C1, 2022.
6. Li C., Li L., Jiang H., Weng K., Geng Y., Li L., Ke Z., Li Q., Cheng M., Nie W., Li Y., Zhang B., Liang Y., Zhou L., Xu X., Chu X., Wei X., Wei X. YOLOv6: A single-stage object detection framework for industrial applications, 2022. DOI: 10.48550/arXiv.2209.02976
7. Poletaev V.A. *Tekhnologiya avtomatizirovannogo proizvodstva lopatok gazoturbinnih dvigatelej [Technology of automated production of gas turbine engine blades]*. Moscow: Mashinostroenie, 2006. 256 p.
8. Glazkov Yu.A. *Kapillyarnyj kontrol' [Capillary testing]*. Moscow: ID Spektr, 2011. 144 p.
9. Kalinichenko N.P., Kuleshov V.K., Kalinichenko A.N. *Kontrol' kachestva pronikayushchimi veshchestvami. Kapillyarnyj kontrol' [Quality control by penetrants. Capillary testing]*, 2<sup>nd</sup> ed. Tomsk: Izd-vo Tomskogo politekhnicheskogo universiteta, 2007. 203 p.
10. Popko E.A., Vorob'ev A.P., Vainshtein I.A. Experience of machine vision application in optical NDT systems. *Svarka i diagnostika: sbornik dokladov mezhdunarodnogo foruma [Welding and diagnostics: collection of reports of the international forum]*, 2015, Pp. 401–406.
11. Egunov O.V., Hartasenko A.V., Chepchurov M.S. *Ustrojstvo beskontaktnogo izmereniya sherohovatosti poverhnoy detalej slozhnoy formy [Device for non-contact measurement of surface roughness of complex-shaped parts]*. Patent Russia No. RU 104697 U1, 2011.
12. GitHub – ultralytics/ultralytics: Ultralytics YOLO11, Available: <https://github.com/ultralytics/ultralytics> (Accessed: 19.12.2024)
13. Sychev I.E. Avtomaticheskaya sistema raspoznavaniya defektov na baze tekhnicheskogo zreniya [Automatic defect recognition system based on machine vision]. *Al'ternativnaya i intellektual'naya energetika: Materialy II Mezhdunarodnoj nauchno-prakticheskoy konferencii [Alternative and Intelligent Energy: Proceedings of the II International Scientific and Practical Conference]*, 2020, Pp. 18–19.
14. Ermakov A.A. *Metody i algoritmy obrabotki i analiza snimkov v kapillyarnoj defektoskopii. Avtoref. kand. tekhn. nauk. [Methods and algorithms for processing and analyzing images in capillary flaw detection. Cand. tech. sci. diss.]*. Vladimir, 2009. 19 p.

15. Shipway N.J., Barden T.J., Huthwaite P., Lowe M.J.S. Automated defect detection for Fluorescent Penetrant Inspection using Random Forest. *NDT & E International*, 2019, Vol. 101, Pp. 113–123. DOI: 10.1016/j.ndteint.2018.10.008
16. Shipway N.J., Huthwaite P., Lowe M.J.S., Barden T.J. Using ResNets to perform automated defect detection for Fluorescent Penetrant Inspection. *NDT & E International*, 2021, Vol. 119, Article no. 102400. DOI: 10.1016/j.ndteint.2020.102400
17. Tout K. *Automatic Vision System for Surface Inspection and Monitoring: Application to Wheel Inspection. PhD thesis*. Troyes, 2018. 188 p.
18. Bobkov A.V. Intercepting Image in Problem of Orientation by Visual Information. *Herald of the Bauman Moscow State Technical University. Series Instrument Engineering*, 2002, Vol. 48, No. 3. Pp. 45–54.
19. Aust J., Shankland S., Pons D., Mukundan R., Mitrovic A. Automated defect detection and decision-support in gas turbine blade inspection. *Aerospace*, 2021, Vol. 8, Article no. 30. DOI: 10.3390/aerospace8020030
20. Wang C.-Y., Bochkovskiy A., Liao H.-Y. M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, 2022. DOI: 10.48550/arXiv.2207.02696
21. Gonzalez R.C., Woods R.E. *Digital image processing*, 3<sup>rd</sup> ed. London: Pearson, 2007. 976 p.
22. Soifer V.A. *Computer Image Processing, Part II: Methods and algorithms*. Riga: VDM Verlag Dr. Müller, 2010. 584 p.
23. Priorov A.L., Khriashchev V.V., Topnikov A.I. *Obrabotka i peredacha mul'timedijnoj informacii [Processing and transmission of multimedia information]*. Yaroslavl: Yaroslavskij gosudarstvennyj universitet im. P.G. Demidova, 2022. 248 p.
24. Korotaev V.V., Krasnyashchih A.V. *Televizionnye izmeritel'nye sistemy [Television measuring systems]*. St. Petersburg: SPbGU ITMO, 2008. 108 p.
25. Korneichuk V.S., Kotlyar D.I., Lomanov A.N., Medvedev E.Yu. *Application of machine vision for calculating defect sizes during fluorescent penetrating testing of gas turbine engine blades*. Bulletin of the Cherepovets State University. Cherepovets: Cherepovets State University, 2022, No. 1 (106), Pp. 31–41. DOI: 10.23859/1994-0637-2022-1-106-3
26. Kazakov O.D., Romashov N.E. Real-time object detection and recognition using a machine learning model. *Vyzovy cifrovoj ekonomiki: razvitie komfortnoj gorodskoj sredy [Challenges of the digital economy: development of a comfortable urban environment]*, 2020, Pp. 353–356.
27. Yamshchikov S.A. Komp'yuternoe zrenie v nerazrushayushchem kontrole [Computer vision in non-destructive testing]. *Novye materialy, oborudovanie i tekhnologii v promyshlennosti: Materialy Mezhdunarodnoj nauchno-tekhnicheskoy konferencii molodyh uchenyh [New Materials, Equipment and Technologies in Industry: Proceedings of the International Scientific and Technical Conference of Young Scientists]*, 2021, P. 163.
28. Chikmarev D.D., Habarov A.R., Karelskaya K.A. Computer vision system for objects defectoscopy. *Informacionnye resursy i sistemy v ekonomike, nauke i obrazovanii: sbornik statej XI Mezhdunarodnoj nauchno-prakticheskoy konferencii [Information Resources and Systems in Economics, Science and Education: proceedings of the XI International Scientific and Practical Conference]*, 2021, Pp. 171–174.
29. Forsyth D., Pons J. *Computer vision: A modern approach*, 2<sup>nd</sup> ed. London: Pearson, 2011. 800 p.

#### INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

**Alekseev Evgeny A.**  
**Алексеев Евгений Александрович**  
 E-mail: evgeny.alekseev@uec-saturn.ru

**Lomanov Alexey N.**  
**Ломанов Алексей Николаевич**  
 E-mail: frei@rsatu.ru  
 ORCID: <https://orcid.org/0000-0001-9271-1552>

**Ivanov Dmitry S.**  
**Иванов Дмитрий Станиславович**  
E-mail: dmitry.ivanov@uesc-saturn.ru

*Submitted: 12.06.2024; Approved: 02.09.2024; Accepted: 20.12.2024.*

*Поступила: 12.06.2024; Одобрена: 02.09.2024; Принята: 20.12.2024.*

Research article

DOI: <https://doi.org/10.18721/JCSTCS.18104>

UDC 004.85



## LEVERAGING NATURAL LANGUAGE PROCESSING TECHNIQUES FOR ENHANCED RECOMMENDER SYSTEMS

*S.A. Shulgin*  , *E.N. Benderskaya*

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

 [shulginsergey0@gmail.com](mailto:shulginsergey0@gmail.com)

**Abstract.** Recommender systems often use NLP methods primarily for content processing. In this study, we propose a new approach to building recommender systems, in which user interaction data with content is considered within the framework of a natural language model. Thus, the user preference vectorization model Pref2Vec is proposed as the basis for a hybrid recommender system. Moreover, a concept of a user embedding space (UES) is introduced, which represents a set of extended embeddings that capture end-user preferences. A new method of applying clustering analysis to the recommendation process is also proposed. The Pref2Vec model and the UES class were implemented in the Python programming language as an extension of the functionality of the Gensim library. The model was evaluated using Recall@k and NDCG@k metrics. The comparative analysis showed that the results obtained are comparable with the performance of the BPRMF, GRU4Rec and NextItRec models, which indicates the potential of the proposed model.

**Keywords:** recommender system, natural language processing, NLP methods, cluster analysis, 2Vec models, vectorization of user preferences, embedding

**Citation:** Shulgin S.A., Benderskaya E.N. Leveraging natural language processing techniques for enhanced recommender systems. Computing, Telecommunications and Control, 2025, Vol. 18, No. 1, Pp. 48–59. DOI: 10.18721/JCSTCS.18104



Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18104>

УДК 004.85



## ИСПОЛЬЗОВАНИЕ МЕТОДОВ ОБРАБОТКИ ЕСТЕСТВЕННОГО ЯЗЫКА В ПРОДВИНУТЫХ РЕКОМЕНДАТЕЛЬНЫХ СИСТЕМАХ

С.А. Шульгин , Е.Н. БендерскаяСанкт-Петербургский политехнический университет Петра Великого,  
Санкт-Петербург, Российская Федерация [shulginsergey0@gmail.com](mailto:shulginsergey0@gmail.com)

**Аннотация.** Многие рекомендательные системы используют методы NLP преимущественно в обработке контента. В работе предлагается новый подход к построению рекомендательных систем, в котором данные о пользовательских взаимодействиях с контентом рассматриваются в рамках модели естественного языка. Таким образом, предлагается модель векторизации пользовательских предпочтений Pref2Vec в качестве основы гибридной рекомендательной системы. Кроме того, предложена концепция пользовательского пространства эмбедингов (UES), которая представляет собой набор расширенных эмбедингов, отображающих предпочтения конечного пользователя. Также предложен новый способ применения задачи кластеризации в решении задачи построения рекомендаций. Модель Pref2Vec и класс UES были реализованы на языке программирования Python в качестве расширения функциональности библиотеки Gensim. Была произведена оценка модели при помощи метрик Recall@k и NDCG@k. Сравнительный анализ показал, что полученные результаты сравнимы с показателями моделей BPRMF, GRU4Rec и NextItRec, что говорит о перспективности предложенной модели.

**Ключевые слова:** рекомендательная система, обработка естественного языка, методы NLP, задача кластеризации, модели 2Vec, векторизация пользовательских предпочтений, эмбединг

**Для цитирования:** Shulgin S.A., Benderskaya E.N. Leveraging natural language processing techniques for enhanced recommender systems // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 48–59. DOI: 10.18721/JCSTCS.18104

### Introduction

The development of recommender systems (RS) is an actively developing area of IT. Natural language processing (NLP) methods allow to change the established paradigm of RS development – data about content, users and their sessions are considered within the framework of a natural language model. This approach opens new opportunities for interpreting a large amount of data.

The task of any RS is to extract and interpret information about user, content, and user's interactions with content to recommend them another content [1]. For this purpose, data mining methods have been used [2]. The different architectures of RS stem from specific information processing tasks: content recommendation, collaborative filtering, knowledge-based recommendation, and hybrid recommendation, the former being the most comprehensive one [1, 3].

The interest in research and development of RS is still wide and the issues are still challenging despite many advances in this area [4]. These issues arise at various stages of RS design and are typically formulated as five core tasks:

- *Evaluation* of RS performance and availability of datasets, including the evaluative criteria and appropriate indicators choice and calculation;
- *Sparsity overcoming*, which arises both from a large volume of input data and users' reluctance to evaluate content;

- *Scalability* of RS, which assumes that all the algorithms used should undergo performance tests on large amount of data;
- *Contextual dependence* that permits an RS to obtain information about users' environment. It improves the accuracy of recommendations;
- *The «gray sheep» problem*, which is the presence of users whose opinions do not coincide with the opinions of the group they belong to.

Machine learning methods are the one way to overcome the above problems. It enables RS to recognize regularities and patterns in amounts of data with the purpose of accuracy improvement of data representation and interpretation. While considering the main methods used in the development of RS [2], it was decided to focus on 2Vec models (Word2Vec, Paragraph2Vec, and Doc2Vec). These models are NLP models that use a neural network to represent entities (words, paragraphs, documents, etc.) as embeddings based on the context of their usage in a corpus [4, 5]. The software implementation of 2Vec models is presented in the Gensim library [6].

Thus, the user preference vectorization model (Pref2Vec) is proposed in this paper as a basis for a hybrid recommender system. The model uses the Doc2Vec model. It extends the functionality of the Gensim library. In addition, the class of user embedding space (UES) is proposed. It is used in constructing the Pref2Vec model. Further, it allows to vectorize end-user preferences by applying a new method of vectorizing text data, which considers the type of user's interaction with the content.

### Vectorization of user preferences

Vectorization (2Vec models) converts text into embeddings that reflect text semantics. Handling embeddings enables to use simple, effective and intuitive algorithms in development. For example, the k-nearest neighbors (KNN) algorithm can be used within the search for the nearest embeddings in the embedding space [7]. Almost all known 2Vec models are put into practice in isolation from RS development, i.e. the models process the text only [5]. In this way, it seems appropriate to develop a model for vectorizing text information that considers the type of user's interaction with the processed content (text). Pref2Vec is a new approach to RS building, which solves the problem of implementation context dependence into the core of a RS by moving to a higher level of abstraction. It means that recommendations will be generated depending on the preferences of a specific user. This approach also permits to follow the principles of constructing a hybrid recommendation, since both static (content) and dynamic (user preferences) data are subjects to the analysis.

Let us impose constraints on the data processed by the RS core. The content will be presented as text. User's interaction with the content will be formalized as differentiated content assessment. It helps to understand the degree of user's interest in a specific content. Based on these constraints, it was decided to utilize the dataset from the GoodReads website [8]. A more detailed exploration of the input data is presented in Sec. «Evaluation of the proposed model».

The Pref2Vec model is based on Doc2Vec, which is used for the initial processing of the content, i.e. for the initial embeddings formation. The model converts documents (text fragments) to fixed-length feature vectors using the PV-DM model (Distributed Memory model of Paragraph Vector) [9]. The algorithm is built on the Word2Vec model: the task of word vectorization consists of its statistical analysis (CBoW or Skip-gram models) relative to other words by researching the context of their use. Vectorization is reached by using a neural network with stochastic gradient descent (SGD) and backpropagation [10]. Further vectorization of documents consists of considering the vectors of all words used in the document using the PV-DM model and the neural network. The optimal embedding size for Doc2Vec depends on the specific application and the size of the data. The general range of embedding sizes is from 100 to 300. In this regard, it was decided to form 200-dimensional embeddings.

It is necessary to overcome the contextual dependence problem. One of the solutions to this problem is to include a type of user's interaction semantics into an initial embedding (a vector representation

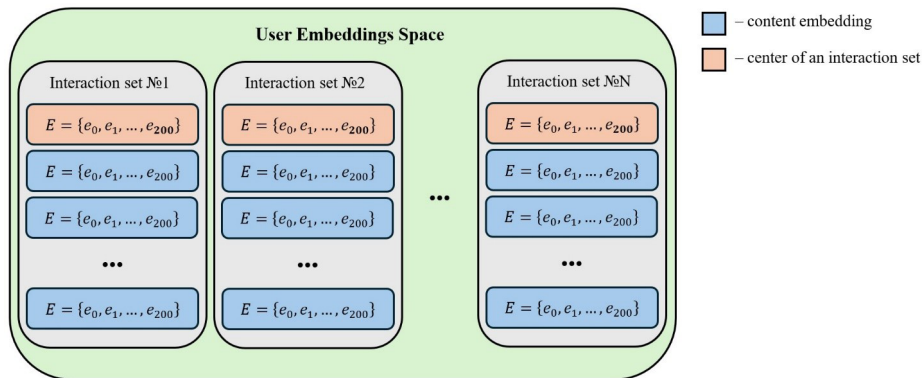


Fig. 1. Scheme of the UES content

of a text formed by Doc2Vec). Initial embeddings will be expanded by adding another one dimension that characterizes the type of user's interaction with content as the subsequent phase of the processing. Expansion of initial embeddings plays the same role as tone in tonal languages. For instance, different tones in the Chinese interpret the same orthography in different ways. In this way, a type of user's interaction with content determines the context of use.

Sets of embeddings divided by interaction types can be interpreted as clusters. The proposed application of clustering algorithms is to build recommendations analyzing information about the interaction clusters of different users. Considering the large dimensionality of the expanded embeddings only key information about the clusters should be used while building recommendations. We can say that the centers of the clusters reflect an average user preference within a specific type of interaction. Hence, let us take the centers as the key information for each user. This application of the cluster analysis permits to reduce the amount of data involved in computations. In addition, it reduces the impact of outliers in interactions clusters given the presence of "gray sheep" in the system. Therefore, there are solutions to the problem of input data sparsity and the "gray sheep" problem. This non-classical application of the clustering analysis (by computing the centers of interaction clusters) is a helpful method in designing the Pref2Vec model. It was decided to use the  $k$ -medoids [11] algorithm as the implementation of the cluster analysis. The algorithm is generally similar to  $k$ -means [12], but the centers of the clusters are medoids (elements of the clusters). This feature of the algorithm also helps to reduce the impact of the outliers on the process of building recommendations, because the effect of cluster elements distant from the majority decreases while computing the center of its medoids.

Result of the data processing for one user (space of extended embeddings) is referred to as the UES. It represents sets of extended embeddings divided by type of user's interactions with the content. The UES should likewise contain the computed centers for each set. Fig. 1 shows the scheme of the UES content.

Based on the above, the process of converting content and users' data is as follows:

- 1) Texts are transformed into 200-dimensional embeddings (initial embeddings) using the Doc2Vec model;
- 2) Embeddings are modified by adding an extra dimension, which will allow reflecting content's belonging to a specific type of user's interaction with the content. Subsequently, a UES is constructed;
- 3) Interaction clusters (sets) are formed as a result of constructing the UES. Afterwards, the clusters need to be analyzed by computing for their centers using the  $k$ -medoids algorithm.

The scheme of the content processing of one category of user's interactions is shown in Fig. 2.

The UES was implemented in Python using the Doc2Vec model from the Gensim library and  $k$ -medoids algorithm from the Scikit-Learn library. The Doc2Vec model was trained on annotations corpus

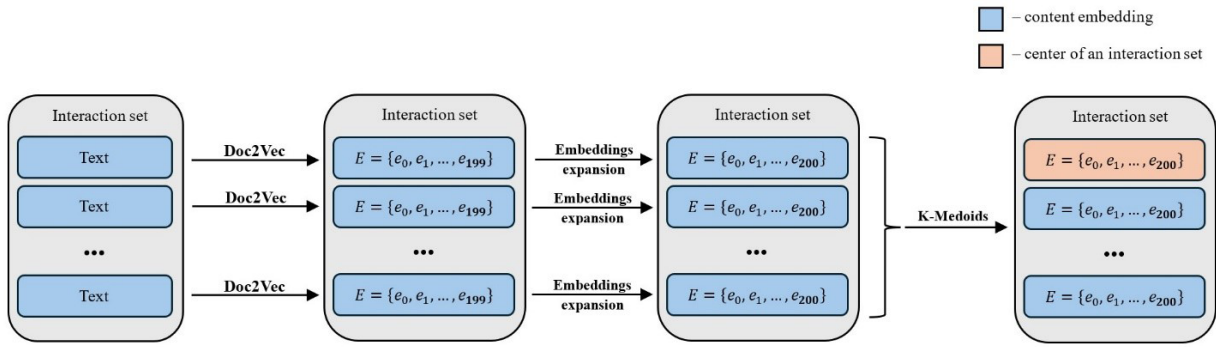


Fig. 2. Scheme of the content processing of one interaction set

from the GoodReads dataset [8]. The t-SNE method was also used to reduce the dimensionality of the embeddings for the purpose of visualization. Fig. 3 illustrates an example of the UES for one user described in the dataset. We can see that cluster partitioning of the content into interaction sets is achieved by expansion of the embeddings. This application of the cluster analysis will further search for users with similar preferences.

### Mechanism of recommendation building

It is proposed to build the core of an RS at a higher level of abstraction – recommendations will be built not only depending on a content similarity, but also depending on the preferences of a specific user, i.e. on their UES. There is an opportunity to vectorize all users' interactions with content as users' preferences by the UES concept. The prospect of searching for similar UES and subsequently building recommendations based on the data obtained opens up. Therefore, it is necessary to determine a method for similar UES searching.

It has been said before that each UES has a collection of interaction centers. We can say that each collection reflects the main direction of user interests at every interaction type. Consequently, all other embeddings except centers can be removed from further calculations. We will reduce the number of embeddings considered in interaction sets. Thus, similar UES searching will be done through a comparison of interaction centers. This approach will permit to avoid a large number of calculations. It also will permit to improve the level of accuracy in determining similar UES. In this case, one should resort to calculating the cosine similarity between centers of considering UES. It is a measure of similarity between two vectors of a Hausdorff pre-Hilbert space, which is used to measure the cosine of the angle between them [13]. The measure is also actively used in 2Vec models [9]. For two vectors  $A$  and  $B$ , the cosine similarity  $\cos(\theta)$  can be represented as follows:

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}}.$$

The Pref2Vec model is assumed to have a users' data corpus, i.e. a corpus of calculated UES. Next, we obtain a list of the UES, closest to a target UES in cosine similarity, when forming a recommendation for a target user. The contents of each UES from the list are potential recommendations, since these are contents that users with similar preferences to a target user interacted with. Due to this, it is necessary to rank the contents of similar UES to build a recommendation. The cosine similarity is also used to solve this problem – embeddings of the nearest UES content will be compared with content embeddings that a

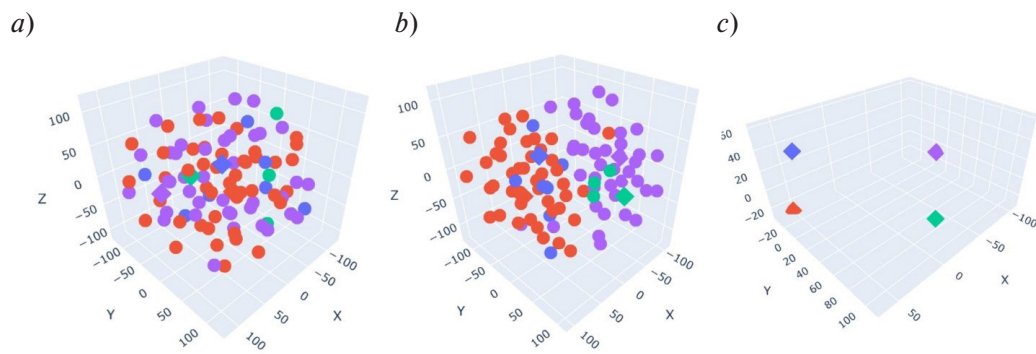


Fig. 3. Visualization of the transforming content and user data into an UES: (a) embeddings formation, (b) embeddings expansion, and (c) computing centers of interaction sets

target user interacted with. Based on the calculations, we will interpret a result list of the closest content as a list of Top-K recommendations.

As a consequence, the algorithm of recommendation building is as follows:

- 1) Centers of interaction sets are calculated for a target user (their UES);
- 2) Further, a search for the nearest UES to the target is performed on the model corpus based on the calculated centers. In other words, it is a search for users with similar preferences in the corpus;
- 3) Elements of similar UES are compared with elements of the target UES and ranked according to a degree of similarity;
- 4) Ranked list of the content is a result list of Top-K recommendations.

The above-described algorithm for recommendation building by the Pref2Vec model is presented as a scheme in Fig. 4.

The Pref2Vec model was also implemented in Python. The model computes a corpus of UES by the UES class previously developed in Python (see Sec. “Vectorization of user preferences”). It furthermore allows to investigate the corpus in order to build recommendations for a target user. The developed model has a *depth* parameter, which characterizes the depth of the search for similar UES, i.e. the number of UES considered in the search. There is also a *target\_interactions* parameter, which reflects the types of interactions used in the recommendations building. An example of a prediction for one of the users from the GoodReads dataset is shown in Fig. 5. We can see that the recommended books (Fig. 5, a) can be semantically correlated with the word clouds of the target UES (Fig. 5, c) based on the titles and annotations of the books (Fig. 5, b). The word clouds represent statistics of words occurrence for content a user interacted with [14] (the content is divided on types of interaction). A more detailed model evaluation will be made below.

### Evaluation of the Pref2Vec model

For evaluation of the model, it was decided to use the GoodReads website dataset [8], since the dataset meets the requirements for the input data (see Sec. “Vectorization of user preferences”). The website provides free access to an extensive database of books and their metadata. Hence, the dataset is a collection of information obtained from the website: book data, user bookshelves data (anonymized information about user interactions with books) and various user reviews [8, 15]. Content of user bookshelves can be used to determine whether a user has read a book on their shelf and how quickly he/she did it, whether a user gave a book a differential assessing (rating), and whether he/she wrote a book review. The main interest is text data (book annotations) and user interactions data within the evaluation of the core of a RS. The interactions data will allow to categorize the text data by interaction types. The collections are also divided into different genre categories. We will use the data from the categories

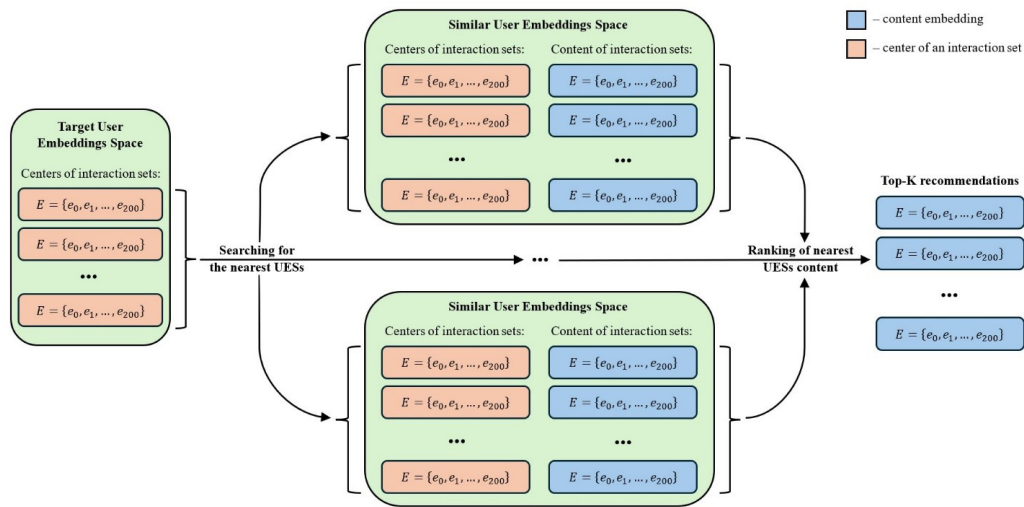


Fig. 4. Scheme of recommendation building by Pref2Vec model

“Children” and “Comics & Graphic”. In total, the collections contain information on 213493 books and 17406979 user interactions.

Book annotations were preprocessed and vectorized using the Gensim library [6]. In addition, a dataset of user preferences was formed by storing book identifiers into different interaction sets (*read*, *shelved*, *rating\_0*, *rating\_1*, *rating\_2*, *rating\_3*, *rating\_4*, *rating\_5*) for each user. The resulting dataset was noise filtered – users whose set *rating\_5* contained less than 40 elements were removed. The Pref2Vec model corpus was composed of 80% of user interactions from the dataset (sample *X*). The remaining 20% of interactions (sample *y*) were used to assess the quality of recommendation building. Elements of sample *y* were ranked by the maximum value of cosine similarity with sample *X* elements. This approach opens an opportunity to estimate not only a percentage of an inclusion of recommendation elements in the reference sample, but also to evaluate the quality of the ranking ability of the model.

Significantly, it is quite difficult to evaluate the performance of a RS due to the non-trivial nature of the recommendation task. For this reason, the range of metrics used is individual for each developed system [16]. The documentation of the RecPack metrics module [17] was discussed in that context. The module contains a large number of baseline and metrics commonly used to evaluate state-of-the-art recommendation algorithms. The Pref2Vec model generates a list of Top-K recommendations, so in this way it is necessary to refer to the metrics for evaluating the ranking quality. The following evaluation methodologies were chosen to achieve comparability with the performance of relevant models [18].

Table 1

Listwise metrics utilized in the model evaluation

| Metric     | Formula  | Description  |
|------------|--|--|
| $Recall@k$ | $Recall(u) = \frac{\sum_{i \in Top-N(u)} y_{u,i}^{true}}{\sum_{j \in 1} y_{u,j}^{true}}$                   | Computes the fraction of true interactions that made it into the Top-K recommendations   |
| $NDCG@k$   | $NDCG(u) = \frac{DCG(u)}{IDCG(u)}$<br>$IDCG(u) = \frac{1}{\sum_{j=1}^{\min(K,  y_u^{true} )} \log_2(j+1)}$ | Computes the normalized sum of gains of all items in a recommendation list (NCDG@k – Normalized Discounted Cumulative Gain, ICDG@k – Ideal Discounted Cumulative Gain) |

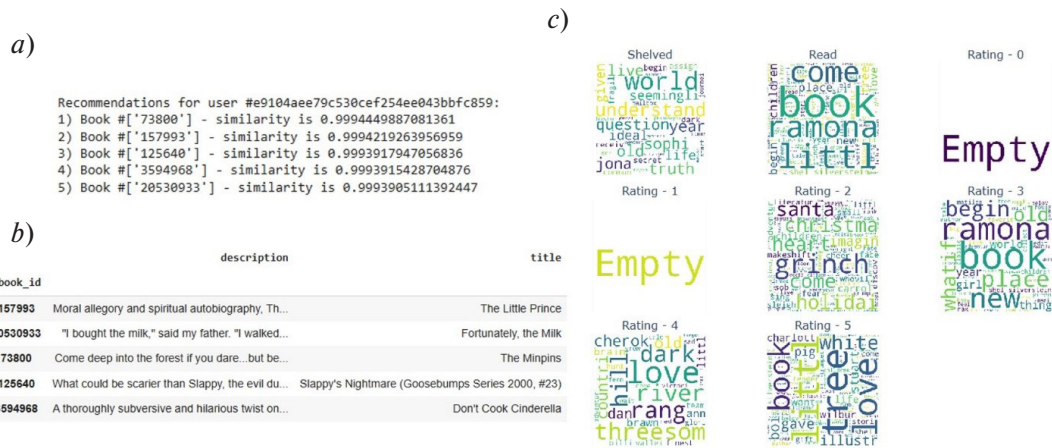


Fig. 5. Result of Top-K recommendation building for a target user: (a) recommendations for the user, (b) recommended content, (c) word clouds of the target UES

Building a Top-K recommendation for one user by the Pref2Vec model is a rather long process. Due to this limitation, the evaluation of the RS core was performed for 200 users. The model corpus consisted of interaction data about 100000 users (UES objects). As a result of the evaluation, the values presented in Table 2 were obtained.

Table 2

**NDCG@k and Recall@k metrics performance of the Pref2Vec model**

| Dataset            | Recall@10 | Recall@15 | Recall@20 | NDCG@10 | NDCG@15 | NDCG@20 |
|--------------------|-----------|-----------|-----------|---------|---------|---------|
| GoodReads-Children | 0.0904    | 0.1056    | 0.1585    | 0.0996  | 0.0769  | 0.0628  |
| GoodReads-Comics   | 0.0542    | 0.1033    | 0.1627    | 0.1140  | 0.1099  | 0.0933  |

The obtained values were compared with the performance of the BPRMF, GRU4Rec, GRU4Rec+, NextItRec, Caser, SASRec and HGN models [18]. The results of the comparison are presented in Fig. 6 and 7. In addition, there is a more detailed comparison of the values for the NDCG@10 and Recall@10 metrics in Tables 3 and 4.

The Recall@k metric values of the Pref2Vec evaluation were not as high as those of some considering recommendation algorithms: HGN, SASRec, and Caser. However, at the same time, the metric values are approximately at the same level as other models: BPRMF, GRU4Rec, and NextItRec. It is also evident that the values of the Recall@k have a higher growth rate with the growth of the parameter k than most of the existing models. On the contrary, the values of the NDCG@k metrics were relatively high. This indicates a good ability of the Pref2Vec to rank recommendations. We can say that the results of the model evaluation seem to be adequate to meet expectations considering the introduction of a new approach to processing user data and the time limitations of the Pref2Vec model.

**Discussion**

We can say that the values obtained are approximately at the same level as some of the relevant models in the field of modern RS design. Let us consider how this model can be improved:

- The performance of the model is definitely affected by its corpus size. Because of time and resource constraints, a corpus of 100000 users data was used during the evaluation, although the GoodReads dataset has data on 876145 users in total [8]. It is likely that a larger corpus size would increase the performance.

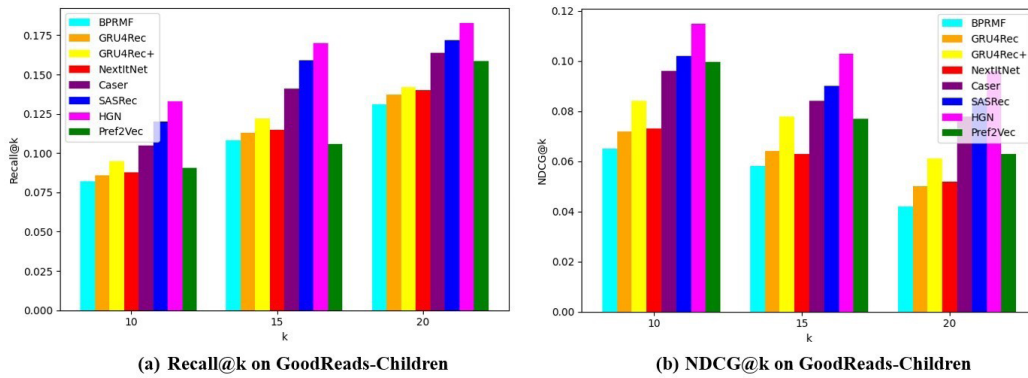


Fig. 6. Comparison of the Recall@k (a) and NDCG@k (b) metrics on the «GoodReads-Children» dataset

Table 3

**Comparison of the NDCG@10 and Recall@10 metrics values of the Pref2Vec model with the performance of relevant RSs (part 1)**

| Dataset                   | BPRMF  | GRU4Rec | GRU4Rec+ | NextItRec | Pref2Vec      |
|---------------------------|--------|---------|----------|-----------|---------------|
| <i>Recall@10</i>          |        |         |          |           |               |
| <i>GoodReads-Children</i> | 0.0814 | 0.0857  | 0.0978   | 0.0879    | <b>0.0904</b> |
| <i>GoodReads-Comics</i>   | 0.0788 | 0.0958  | 0.1288   | 0.1078    | <b>0.0542</b> |
| <i>NDCG@10</i>            |        |         |          |           |               |
| <i>GoodReads-Children</i> | 0.0664 | 0.0715  | 0.0821   | 0.0720    | <b>0.0996</b> |
| <i>GoodReads-Comics</i>   | 0.0713 | 0.0912  | 0.1328   | 0.1171    | <b>0.1140</b> |

– The *depth* hyperparameter impact on the efficiency of the model was not investigated in the paper. Presumably, increasing the value of this hyperparameter will have a positive effect on the metrics, since more content will be examined during the recommendation building.

– The quality of content vectorization directly affects the quality of the recommendation building. It is possible to increase performance by varying the Doc2Vec model hyperparameters, which are used to generate the initial embeddings.

– In addition, the model performance is affected by the method of searching for interaction cluster centers. The use of methods such as Affinity Propagation [19] or Hierarchical Agglomerative Clustering [20] can have a positive impact on the values of the metrics.

– In addition, it is possible to increase the model's performance by implementing spatial computing methods from the SciPy library [21] other than cosine similarity: for example, computing the Euclidean distance [22] or the Mahalanobis distance [23].

**Conclusion**

In this paper, the user preferences vectorization model Pref2Vec was proposed as the core of a hybrid RS. The model is a new approach to RS design, which solves the problem of introducing context dependence into the RS core. We managed to achieve it by moving to a higher level of abstraction, where recommendations are built depending on the preferences of a specific user. The model also permits adhering to the principles of hybrid recommendation, which is the most relevant approach in RS development. Furthermore, the concept of the UES was proposed as the result of end-user data processing. It is a set of extended embeddings that display the semantics of vectorized content and the semantics



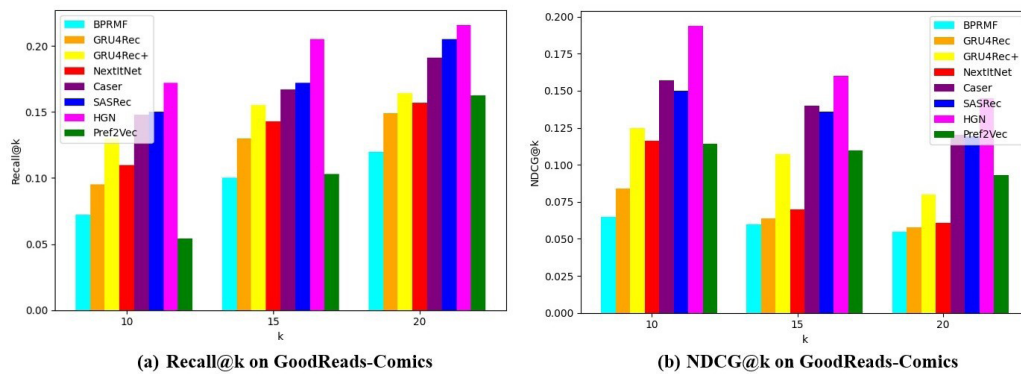


Fig. 7. Comparison of the Recall@k (a) and NDCG@k (b) metrics on the «GoodReads-Comics» dataset

Table 4

**Comparison of the NDCG@10 and Recall@10 metrics values  
of the Pref2Vec model with the performance of relevant RSs (part 2)**

| Dataset                   | Caser  | SASRec | HGN    | Pref2Vec      |
|---------------------------|--------|--------|--------|---------------|
| <i>Recall@10</i>          |        |        |        |               |
| <i>GoodReads-Children</i> | 0.1060 | 0.1165 | 0.1263 | <b>0.0904</b> |
| <i>GoodReads-Comics</i>   | 0.1473 | 0.1494 | 0.1743 | <b>0.0542</b> |
| <i>NDCG@10</i>            |        |        |        |               |
| <i>GoodReads-Children</i> | 0.0943 | 0.1007 | 0.1130 | <b>0.0996</b> |
| <i>GoodReads-Comics</i>   | 0.1629 | 0.1592 | 0.1927 | <b>0.1140</b> |

of user's interaction with a content. Clustering helps to tackle data sparsity, improves RS scalability, and manages the problem of “gray sheep”.

The Pref2Vec model and the UES class were implemented in Python as an extension of the Gensim library functionality. Afterwards, the model was evaluated using the Recall@k and NDCG@k metrics. Comparative analysis showed that the results obtained are comparable with the performance of the BPRMF, GRU4Rec and NextItRec models. Values of the Recall@k metric of the Pref2Vec model have a higher growth rate with the growth of the k parameter than most of the existing models. In addition, values of the NDCG@k metric showed that the Pref2Vec model has a high quality of ranking recommendations.

## REFERENCES

1. Jannach D., Zanker M., Felfernig A., Friedrich G. *Recommender Systems: An Introduction*. Cambridge: Cambridge University Press, 2010, 352 p. DOI: 10.1017/CBO9780511763113
2. Karatzoglou A., Hidasi B. Deep learning for recommender systems. *11<sup>th</sup> ACM Conference on Recommender Systems (RecSys '17)*, 2017, Pp. 396–397. DOI: 10.1145/3109859.3109933
3. Jannach D., Zanker M., Friedrich G. Tutorial: Recommender Systems. *International Joint Conference on Artificial Intelligence (IJCAI-13)*, 2013, pp. 221–236.
4. Khuro S., Ali Z., Ullah I. Recommender systems: Issues, challenges, and research opportunities. *Information Science and Applications (ICISA)*, 2016, Vol. 376, Pp. 1179–1189. DOI: 10.1007/978-981-10-0557-2\_112

5. **Barkan O., Koenigstein N.** ITEM2VEC: Neural item embedding for collaborative filtering. *2016 IEEE 26<sup>th</sup> International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016, Pp. 1–6. DOI: 10.1109/MLSP.2016.7738886
6. **Řehůřek R., Sojka P.** Software framework for topic modelling with large corpora. *LREC 2010 Workshop on New Challenges for NLP Frameworks*, 2010, pp. 45–50. DOI: 10.13140/2.1.2393.1847
7. **Gong C., Shen G., Guo L., Tallent N., Zhao D.** OPDR: Order-Preserving Dimension Reduction for semantic embedding of multimodal scientific data. *arXiv:2408.10264*, 2024. DOI: 10.48550/arXiv.2408.10264
8. **Wan M., McAuley J.** Item recommendation on monotonic behavior chains. *12<sup>th</sup> ACM Conference on Recommender Systems (RecSys '18)*, 2018, Pp. 86–94. DOI: 10.1145/3240323.3240369
9. **Le Q.V., Mikolov T.** Distributed representations of sentences and documents. *arXiv:1405.4053*, 2014. DOI: 10.48550/arXiv.1405.4053
10. **Mikolov T., Chen K., Corrado G., Dean J.** Efficient estimation of word representations in vector space. *arXiv:1301.3781*, 2013. DOI: 10.48550/arXiv.1301.3781
11. **Schubert E., Rousseeuw P.J.** Fast and eager k-medoids clustering:  $O(k)$  runtime improvement of the PAM, CLARA, and CLARANS algorithms. *Information Systems*, 2021, Vol. 101, Art. no. 101804. DOI: 10.1016/j.is.2021.101804
12. **Camilli G., Suter L.** NLP cluster analysis of common core state standards and NAEP item specifications. *arXiv:2412.04482*, 2024. DOI: 10.48550/arXiv.2412.04482
13. **Manning C.D., Raghavan P., Schütze H.** *An Introduction to Information Retrieval*. Cambridge: Cambridge University Press, 2008, 544 p. DOI: 10.1017/CBO9780511809071
14. **Mueller A.C.** Wordcloud (1.9.4). *Zenodo*, 2024. DOI: 10.5281/zenodo.14062883
15. **Wan M., Misra R., Nakashole N., McAuley J.** Fine-grained spoiler detection from large-scale review corpora. *57<sup>th</sup> Conference of the Association for Computational Linguistics (ACL)*, 2019, Pp. 2605–2610. DOI: 10.18653/v1/P19-1248
16. **Schröder G., Thiele M., Lehner W.** Setting goals and choosing metrics for recommender system evaluations. *2<sup>nd</sup> Workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces (UCERSTI2)*, 2011, Pp. 23–53.
17. **Michiels L., Verachtert R., Goethals B.** RecPack: An(other) experimentation toolkit for Top-N recommendation using implicit feedback data. *16<sup>th</sup> ACM Conference on Recommender Systems (RecSys '16)*, 2022, Pp. 648–651. DOI: 10.1145/3523227.3551472
18. **Ma C., Kang P., Liu X.** Hierarchical gating networks for sequential recommendation. *25<sup>th</sup> ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD '19)*, 2019, Pp. 825–833. DOI: 10.1145/3292500.3330984
19. **Hämäläinen M., Rueter J., Alnajjar K.** Analyzing Pokémon and Mario streamers' twitch chat with LLM-based user embeddings. *4<sup>th</sup> International Conference on Natural Language Processing for Digital Humanities (NLP4DH)*, 2024, Pp. 499–503. DOI: 10.18653/v1/2024.nlp4dh-1.48
20. **Aufschläger R., Wilhelm S., Heigl M., Schramm M.** ClustEm4Ano: Clustering text embeddings of nominal textual attributes for microdata anonymization. *arXiv:2412.12649*, 2024. DOI: 10.48550/arXiv.2412.12649
21. **Virtanen P., Gommers R., Oliphant T.E. et al.** SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 2020, Vol. 17, Pp. 261–272. DOI: 10.1038/s41592-019-0686-2
22. **Lo K.I., Sadrzadeh M., Mansfield S.** Quantum-like contextuality in large language models. *arXiv:2412.16806*, 2024. DOI: 10.48550/arXiv.2412.16806
23. **Hart S.N., Tavorara T.E.** Measuring what matters: Intrinsic distance preservation as a robust metric for embedding quality. *arXiv:2407.21590*, 2024. DOI: 10.48550/arXiv.2407.21590

**INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ**

**Shulgin Sergey A.**

**Шульгин Сергей Александрович**

E-mail: shulginsergey0@gmail.com

ORCID: <https://orcid.org/0009-0009-1102-7333>

**Benderskaya Elena N.**

**Бендерская Елена Николаевна**

E-mail: helen.bend@gmail.com

*Submitted: 29.12.2024; Approved: 18.02.2025; Accepted: 04.03.2025.*

*Поступила: 29.12.2024; Одобрена: 18.02.2025; Принята: 04.03.2025.*

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

## Устройства и системы передачи, приема и обработки сигналов

Research article

DOI: <https://doi.org/10.18721/JCSTCS.18105>

UDC 004.89



### RESNET-SV: FAST AND ACCURATE SPEAKER VERIFICATION WITH A MULTI-LAYER CASCADE ATTENTION MECHANISM

A.A. Aliyev  , S.A. Molodyakov 

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

 [aliyev.aa@edu.spbstu.ru](mailto:aliyev.aa@edu.spbstu.ru)

**Abstract.** One of the most challenging issues of voice biometrics rapid development is the need to develop methods that can combine speed and accuracy. Traditional solutions tend to choose a compromise between these two aspects, which either complicates the speaker verification process or reduces accuracy, especially under real-world conditions in which background noise and fluctuation in speech are substantial obstacles. This paper examines modern approaches and their architectural features. The architecture is based on ResNet, originally designed for computer vision tasks, which was modified and adapted for optimal performance in speech processing. The proposed modification method based on a multi-layer cascade attention mechanism for feature extraction from convolutional blocks is described in detail. This modification allows using fewer layers for feature extraction, thereby increasing the speed of the model, and allows to deal more effectively with the noise in the audio signal. The paper concludes with the model parameters used in the training process, as well as key metrics such as EER and minDCF computed on the VoxCeleb1 dataset. The results are compared with solutions built on other architectures. Through experimentation, the authors were able to achieve a high level of accuracy, with a smaller number of the neural network model parameters. This work brings us closer to a wider application of voice biometric systems in various scenarios.

**Keywords:** speaker verification, speaker identification, voice biometrics, convolutional neural networks, attention mechanism, speech processing

**Citation:** Aliyev A.A., Molodyakov S.A. ResNet-SV: Fast and accurate speaker verification with a multi-layer cascade attention mechanism. Computing, Telecommunications and Control, 2025, Vol. 18, No. 1, Pp. 60–71. DOI: 10.18721/JCSTCS.18105

Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18105>

УДК 004.89



## RESNET-SV: БЫСТРАЯ И ТОЧНАЯ ВЕРИФИКАЦИЯ СПИКЕРА С ИСПОЛЬЗОВАНИЕМ МНОГОУРОВНЕВОГО КАСКАДНОГО МЕХАНИЗМА ВНИМАНИЯ

А.А. Алиев  , С.А. Молодяков Санкт-Петербургский политехнический университет Петра Великого,  
Санкт-Петербург, Российская Федерация [aliev.aa@edu.spbstu.ru](mailto:aliev.aa@edu.spbstu.ru)

**Аннотация.** Одной из самых сложных проблем быстрого развития голосовой биометрии является необходимость разработки методов, способных сочетать скорость и точность. Традиционные решения, как правило, выбирают компромисс между этими двумя аспектами, что приводит либо к усложнению процесса верификации спикеров, либо к снижению точности, особенно в реальных условиях, когда фоновый шум и колебания речи являются существенными препятствиями. В данной статье рассматриваются современные подходы и их архитектурные особенности. Основой для разработки архитектуры послужила ResNet, изначально предназначенная для задач компьютерного зрения, которая была модифицирована и адаптирована для оптимальной работы в области обработки речи. Подробно описывается предложенный метод модификации на основе многослойного каскадного механизма внимания для извлечения признаков из сверточных блоков. Такая модификация позволяет использовать меньшее количество слоев для извлечения признаков, тем самым увеличивая скорость работы модели, а также позволяет более эффективно бороться с возникшими шумами в аудиосигнале. В заключении статьи представлены параметры модели, использованные в процессе обучения, а также ключевые метрики, такие как EER и minDCF, рассчитанные на выборке данных VoxCeleb1. Результаты сравниваются с решениями, построенными на других архитектурах. В ходе экспериментов авторам удалось достичь высокого уровня точности при меньшем количестве параметров модели нейронной сети. Эта работа приближает нас к более широкому применению систем голосовой биометрии в различных сценариях.

**Ключевые слова:** верификация спикеров, идентификация спикеров, голосовая биометрия, сверточные нейронные сети, механизм внимания, обработка речи

**Для цитирования:** Aliyev A.A., Molodyakov S.A. ResNet-SV: Fast and accurate speaker verification with a multi-layer cascade attention mechanism // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 60–71. DOI: 10.18721/JCSTCS.18105

### Introduction

Speaker verification is the core of the authentication process for many applications: security systems, access control, financial transactions, virtual assistants, etc. It is a way to identify a person using his or her voice, which is unobtrusive and natural. Its importance has increased with the digital revolution, with the increased need for highly secure, reliable and user-friendly mechanisms of authentication. These speaker verification systems use the personal characteristics of an individual's voice. They happen to be much more effective in authenticating access than the traditional passwords or PIN codes, which can easily be stolen or forged.

Although, despite the numerous advances in speaker verification technology, there still are quite many difficulties [1] that affect their effectiveness and wider application. The main issue is the variability of the human voice, which is affected by various conditions – from illness and emotional state to aging. The human voice is also affected by environmental conditions – background noise and acoustic environment of the room. Such variabilities can greatly affect the ability of the system to accurately

recognize the speaker's voice; this again leads to an increase on error rate. It also depends on trade-off between accuracy and speed, which is an extremely important issue: faster systems decrease accuracy, while ultra-precise systems may not work in real-time conditions for many applications.

The purpose of this work, therefore, is to address the above-stated challenges by exploring the potential of different convolutional neural network (CNN) architectures [2] to improve the speed and accuracy of the speaker verification systems. CNNs, therefore, provide state-of-the-art performance over a wide range of deep learning architectures, since they allow gradients to propagate efficiently through a deep network. With all this in view, we tend to develop a speaker verification system based on these architectures that can achieve high accuracy in the face of voice variability and environmental noise, and operate at a speed sufficient for the system to be used in real-time applications.

Therefore, this paper makes two main contributions to the field of speaker verification. The first one is introducing a new architecture based on the best practices from different CNN architectures, designed to fit speaker verification problems. The second one is bringing state-of-the-art advances in deep learning techniques, such as feature aggregation techniques with attention mechanisms, to better extract and process voice characteristics. We also conduct rigorous testing of the system to demonstrate that it outperforms the existing benchmarks in both accuracy and speed. Finally, the study provides valuable insights into the application of CNNs in speech processing tasks, hence opening a way for further research and development in this area.

## Related Works

### *Survey of existing methods*

Speaker verification technologies have come a long way from traditional to very advanced deep learning-based approaches. Firstly, the traditional methodologies, Gaussian Mixture Models (GMM) [3] and Hidden Markov Models (HMM) [4] approaches led to statistical modeling of voice characteristics. These methods mostly used hand-crafted features, such as Mel-Frequency Cepstral Coefficients (MFCCs) [5], designed to characterize the speaker's voice. With the enormous success of deep learning, neural network-driven methods have managed to revolutionize the speaker verification landscape. Most of them use three types of models: Deep Neural Networks (DNN), CNNs, and Recurrent Neural Networks (RNN), which have proven their ability to automatically learn hierarchical representations of raw audio data. Later Long Short-Term Memory (LSTM) [6] networks emerged, and more recently – attention mechanisms to better capture the temporal dynamics and dependencies in speech data.

### *Reference architectures in speaker verification*

To benchmark the performance of our ResNet-SV architecture, we compared it with several established models in speaker verification:

- **Wav-LM** [7]: it is the state-of-the-art speech model developed by Microsoft, generally used for various types of speech recognition and understanding. It leverages the transformer architecture, proven quite effective for a wide range of applications demanding natural language processing. Wav-LM operates directly on raw audio waveforms with self-supervised training on massive amounts of unlabeled audio data. This way, the model gets to learn powerful speech representations useful for multiple downstream tasks, such as speech transcription, speaker identification, and emotion detection. The model demonstrates robustness, showing good performance over different accents and in varied speech scenarios.

- **ECAPA-TDNN** (Emphasized Channel Attention, Propagation and Aggregation in Time Delay Neural Network-Based Speaker Verification) [8]: it advances the TDNN architecture, including channel-wise attention mechanisms and methods for improving feature aggregation. The model delivers an exceptionally high performance in the extraction of the fine-grained speaker characteristics from complex audio input and thus provides an extremely useful application for such biometric authentication and forensics areas. ECAPA-TDNN is particularly noteworthy in low-resource and challenging acoustic settings.

- **Deep Speaker Recognition (ResNet-50)** [9]: a ResNet-based model that extracts speaker embedding directly from spectrograms for verification. This model, like ours, is based on the original ResNet model, but without the improvements, we made in this study, and is a direct competitor to our model.

Thus, these are baseline models, both traditional and deep learning-based, and as such they represent a relatively comprehensive overview of current methods and best practices in speaker verification.

#### *Limitations*

Despite these remarkable achievements, current speaker verification techniques have some limitations. Classical techniques perform quite well, but fail to cope with high dimensionality of the speech data and usually provide much lower accuracy than deep learning models. Another challenge with deep learning-based methods is that they are computationally inefficient. This is especially true for deeper models, which require enormous resources for training and computation. In addition, the accuracy of many other models is reduced by issues, such as background noise, variations in emotional state, and speech irregularities associated with illness. Some of them suffer from overtraining due to the peculiarities of training pipelines and architectures. Therefore, there is an obvious interest in building more robust and efficient models that can maintain high accuracy even in the most challenging situations.

#### *Rationale for CNNs*

CNNs have shown the potential to address such limitations observed in current speaker verification approaches. First developed for the task of image recognition, very deep CNNs like ResNet have shown success in varied domains of deep learning due to an innovative architectural feature that allowed training networks hundreds of layers deep. The main idea of ResNet is the learning through shortcut connections, allowing the gradients to flow across the network without any obstructions, hence avoiding the vanishing gradient problem, that most deeper architectures face. This increases not only the learning capacity of the model, but also its computational efficiency, since this allows learning more representations without significant growth in the usage of computational resources. ResNet shows great promise for speaker verification, since it is able to capture the temporal dynamics and features of normal to complex voice signals, even under adverse conditions, with greater accuracy than other architectures. Besides, their efficiency and scalability would make them to be used in the cryptographic scheme for real-time verification applications, an area where there is a pressing need. The proven success in various domains and unique strengths of the explored application of CNNs in the field of speaker verification make it a quite logical and very promising way to overcome the existing problems.

## Materials and methods

### *Architecture overview*

The strategic architectural decisions in the proposed convolutional network architecture for speaker verification should be aimed at achieving the best trade-off between speed and accuracy. Thus, the architecture developed in this work is supposed to be a ResNet structure with modified ResNet blocks, each containing two main paths: the main learning path and the shortcut connection. At the convolutional layers, batch normalization is used, and then ReLU is implemented as an activation function to enable the learning of non-linearity. Both convolutional layers are set to learn and extract features of the input signal at various abstraction levels, and the shortcut connections help the gradients flow directly without facing the problem of vanishing gradients.

As you can see from Fig. 1, the architecture consists of the following parts:

1. **Input:** Starts with a raw audio waveform, which is transformed into a Mel spectrogram using F-Banks (Filter Banks) [10] features.
2. **ResNet-SV Blocks with Convolutional Layers:** A series of 3x3 and 1x1 convolutional layers with increasing feature maps are applied, capturing hierarchical audio features. We have reduced the number of channels in convolutional layers, unlike the original ResNet or Deep Speaker Recognition ResNet.

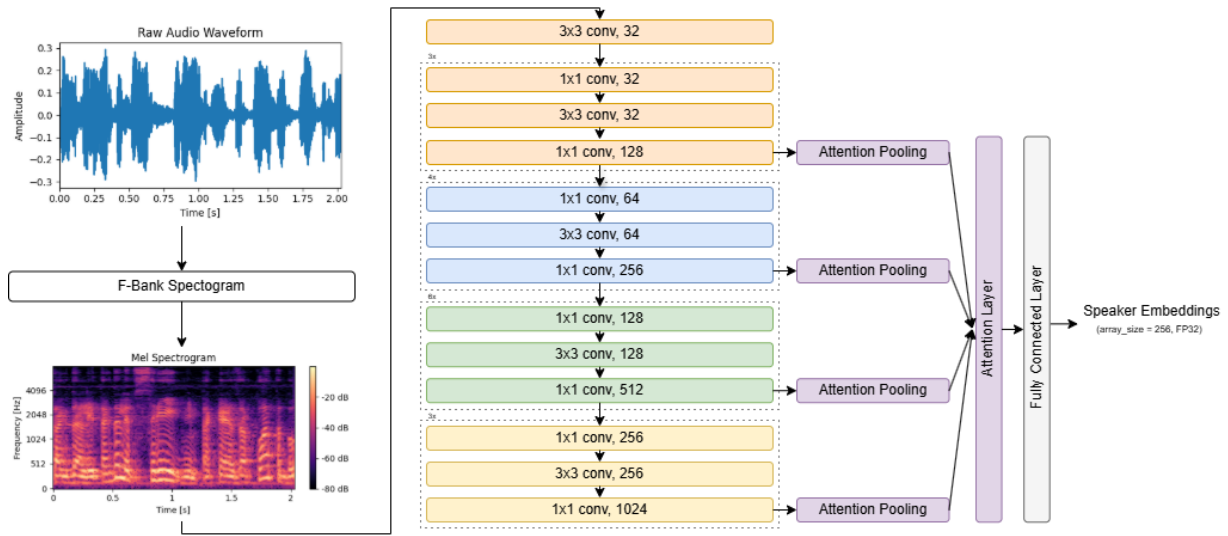


Fig. 1. ResNet-SV architecture

3. **Attention Pooling:** Each convolutional block is followed by an attention [11] pooling layer to focus on important parts of the feature maps.

4. **Attention Layer:** Outputs from all attention-pooled layers are merged.

5. **Fully Connected Layer:** Finally, a fully connected layer generates 256-dimensional speaker embedding as the output.

We use features from different blocks of our convolutional architecture to be able to capture different levels of abstraction of audio features, the upper layers capture more complex audio data, while the lower layers often display more basic information. A more detailed architecture of ResNet-SV blocks is shown in Fig. 2.

As can be seen from Table 1, unlike the original ResNet paper [9], we have reduced the number of channels in the convolutional layers by a factor of two, which naturally had a strong impact on the learning and inference speed, while not significantly affecting the quality of the neural network due to the way the architecture is structured.

As in other similar ResNet architectures, we can increase the size of our network by increasing the number of ResNet-SV blocks, some of them are shown in Table 1.

One of the significant improvements of ResNet-SV is the inclusion of an attention mechanism [12] between the convolutional blocks, designed specifically to make the model better focus on those salient features that are critical in between-speaker discrimination. The context-aware attention module is used here to dynamically weight between diverse region importance of the input feature map. The mechanism of our attention pooling layers is as follows:

1. First, we calculate  $\alpha$  (attention) for the original input feature map ( $x_i$ ). We use two Conv1D layers with tanh activation function. Here we take tanh function activation instead of ReLU, because tanh function converges better and faster. After this, we calculate SoftMax function values for output.

$$\alpha = \sigma\left(\text{Conv1D}\left(\tanh\left(\text{Conv1D}\left(x_i\right)\right)\right)\right);$$

$$\sigma(x)_i = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}.$$



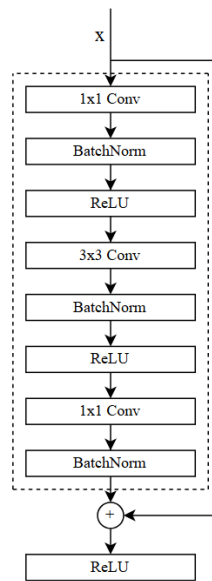


Fig. 2. ResNet-SV block structure

Table 1

**ResnetSV architecture**

| Layer name                             | Output size                 | ResNet-SV-50   | ResNet-SV-101  | ResNet-SV-152  |
|--|-----------------------------|--|--|--|
| conv1                                  | $T \times 80 \times 32$     | $3 \times 3.32$  |  |  |
| ResNet-SV-Block<br>(resnet-sv_conv2_x) | $T \times 80 \times 128$    | $\begin{bmatrix} 1 \times 1.32 \\ 3 \times 3.32 \\ 1 \times 1.128 \end{bmatrix} \times 3$    | $\begin{bmatrix} 1 \times 1.32 \\ 3 \times 3.32 \\ 1 \times 1.128 \end{bmatrix} \times 3$    | $\begin{bmatrix} 1 \times 1.32 \\ 3 \times 3.32 \\ 1 \times 1.128 \end{bmatrix} \times 3$    |
| ResNet-SV-Block<br>(resnet-sv_conv3_x) | $T/2 \times 40 \times 256$  | $\begin{bmatrix} 1 \times 1.64 \\ 3 \times 3.64 \\ 1 \times 1.256 \end{bmatrix} \times 4$    | $\begin{bmatrix} 1 \times 1.64 \\ 3 \times 3.64 \\ 1 \times 1.256 \end{bmatrix} \times 4$    | $\begin{bmatrix} 1 \times 1.64 \\ 3 \times 3.64 \\ 1 \times 1.256 \end{bmatrix} \times 4$    |
| ResNet-SV-Block<br>(resnet-sv_conv4_x) | $T/4 \times 20 \times 512$  | $\begin{bmatrix} 1 \times 1.128 \\ 3 \times 3.128 \\ 1 \times 1.512 \end{bmatrix} \times 6$  | $\begin{bmatrix} 1 \times 1.128 \\ 3 \times 3.128 \\ 1 \times 1.512 \end{bmatrix} \times 6$  | $\begin{bmatrix} 1 \times 1.128 \\ 3 \times 3.128 \\ 1 \times 1.512 \end{bmatrix} \times 6$  |
| ResNet-SV-Block<br>(resnet-sv_conv5_x) | $T/8 \times 10 \times 1024$ | $\begin{bmatrix} 1 \times 1.256 \\ 3 \times 3.256 \\ 1 \times 1.1024 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1.256 \\ 3 \times 3.256 \\ 1 \times 1.1024 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1.256 \\ 3 \times 3.256 \\ 1 \times 1.1024 \end{bmatrix} \times 3$ |
| Attention Pooling<br>(atn_conv2_x)     | 20480                       | -  |  |  |
| Attention Pooling<br>(atn_conv3_x)     | 20480                       | -  |  |  |
| Attention Pooling<br>(atn_conv4_x)     | 20480                       | -  |  |  |
| Attention Pooling<br>(atn_conv5_x)     | 20480                       | -  |  |  |
| Attention                              | 81920                       | -  |  |  |
| Fully Connected                        | 256                         | -  |  |  |

2. Then we calculate the mean and variance [13], taking into account the weights from the attention layer.

$$\text{mean} = \frac{\sum_{i=1}^T (\alpha_i \cdot x_i)}{T},$$

$$\text{var} = \frac{\sum_{i=1}^T (\alpha_i \cdot x_i - \text{mean})^2}{T}.$$

3. Then we calculate the standard deviation based on the variance.

$$\text{std} = \sqrt{\text{var}}.$$

We perform these calculations for each feature, in our case  $F_{num} = 10240$ . Since we count mean and std for each feature, we get  $F_{num} = 2 \times 10240 = 20480$  features. We use the attention layers in this way after each of our ResNet blocks. At the end we use the obtained arrays of mean and std values, combine them and feed them further to the next layer of attention.

In order to improve robustness, especially in noisy conditions, a noise-aware training strategy is implemented into the model. The general approach for improved invariance is to augment training data with different types of background noise. This, in turn, increases invariance and helps the network learn more discriminative features in noisy environments.

#### ***Data preprocessing***

Below are the pre-processing stages that the pipeline undergoes in preparing the raw audio data for effective learning.

In the first stage, we augment the data with a speed perturbation and a volume change. In addition, we use various noise and other signals from datasets such as RIRS [14] and MUSAN [15]. Each of these types of augmentation is applied to each example with a probability of 0.6. This approach allows us to secure our neural network from overfitting and improves performance under challenging conditions.

Since raw audio contains too much data that we often do not need, we need to convert raw audio into a more meaningful data type. In speech processing tasks, spectrograms are often used. There are several ways to create a spectrogram. We decided to use filter banks (F-banks). In the case of audio signal processing, F-banks serve to segment the signals into shorter, overlapping frames to capture the dynamic nature of speech more effectively. F-banks apply a set of band-pass filters corresponding to the critical bands of human hearing to each frame to obtain a representation with bands representative of specific frequency ranges. This follows close to the auditory scale of human perception in every frame. These F-bank energies have a normal zero-mean and unit variance across all frames. Normalization acts as a standardization method that reduces influence due to varying amplitudes of signals on the performance of the model, thus assuring that the focus is on the spectral characteristics of the audio rather than on loudness.

We used two second segments, the frame size of which was 25 ms with a 10 ms step. Taking into account the input segment size and step size, we get the final input tensor as (B, 80, 200), where B is the batch size.

#### ***Training procedure***

ResNet-SV is trained on curated data that collects several publicly available corpora, providing variations in accent, dialect, and recording conditions. The training, validation, and test splits of the dataset are kept independent of each other, preserving speaker independence from one split to another, except that data values do not creep through splits.

The AAM-Softmax [16] loss is applied to the model, which helps keep high classification accuracy value. We use the Adam [17] optimizer with adaptive learning rate features and fast convergence. The

parameters are trained using a learning rate initialized at 0.001 and have a schedule decay. The batch size is 64 and the epochs are set to a maximum of 100, with early stopping based on the validation loss to curb overfitting.

### *Evaluation metrics*

We evaluate ResNet-SV using two core metrics that have seen wide use in the speaker verification community – **Equal Error Rate (EER)** [18] and the **minimum Detection Cost Function (minDCF)** [19]. The EER point is the point, at which the false acceptance rate is exactly equal to the false rejection rate and points out equal errors for both, which can be taken as the overall error rate of the system. The minDCF is one more factor that has been implemented to correct the decision threshold in the verification cost evaluation. The minDCF is found by searching for the particular system configuration's cost that minimizes the detection cost function. These metrics provide holistic views on the effectiveness and practical utility of the model used for speaker verification tasks.

The EER itself cannot be represented as a mathematical formula because it refers to the point of intersection of two errors:

$$FAR(t) = FRR(t),$$

where  $FAR(t)$  is the rate, at which impostor attempts are incorrectly accepted above the threshold  $t$ ;  $FRR(t)$  is the rate, at which genuine attempts are incorrectly rejected below the threshold  $t$ .

The minDCF formula is as follows:

$$C_{det}(t) = C_{miss} \times P_{miss}(t) \times P_{target} + C_{fa} \times P_{fa}(t) \times (1 - P_{target}),$$

where  $C_{miss}$  is the cost of a miss (false rejection);  $C_{fa}$  is the cost of a false alarm (false acceptance);  $P_{miss}(t)$  is the probability of a miss (false rejection rate) at threshold  $t$ ;  $P_{fa}(t)$  is the probability of a false alarm (false acceptance rate) at threshold  $t$ ;  $P_{target}$  is the a priori probability of the speaker being the target (i.e., a genuine user).

The minDCF is then calculated by finding the value of  $t$  that minimizes  $C_{det}(t)$ .

## Experiments

### *Datasets*

In this study, the VoxCeleb [20] datasets have been used for training and testing the proposed ResNet-SV model. This dataset is one of the best in benchmarking for speaker verification.

**Training dataset** VoxCeleb2 contains over a million utterances from 6112 celebrities collated from YouTube video sources. It represents a full range of accents, languages, and acoustic conditions, making it perfect for training very vibrant models in speaker verification. The quantity and diversity of VoxCeleb2 makes it possible to train deep neural networks, which means that models trained on it generalize well to populations of different speakers and conditions.

**Testing Dataset** VoxCeleb1 includes more than 100000 utterances of 1251 celebrities. This dataset shares the same characteristics with VoxCeleb2, except that the speakers are different, so it provides a rigorous test to the model's generalization ability on unseen data. This is a standard dataset type in speaker-verification research tasks, given that VoxCeleb1 is intended for testing and hence for direct comparison with earlier works.

These VoxCeleb datasets, containing real-world complexity, diversity, natural noise, and other variability, are a very suitable domain for evaluating the efficacy of the proposed noise-robust convolutional network architecture. We also chose the VoxCeleb dataset, as it is a global benchmark standard for speaker verification tasks and allows us to more easily compare our work with other works and simplifies the replication of our work.

### *Experimental setup*

Experiments are carried out on an NVIDIA SuperPOD A100 high-performance computing cluster node, which was equipped with 8x NVIDIA A100-SXM4-40GB graphics cards and with 2x AMD EPYC 7742 processors with 64 cores, 128 threads with 1TB of RAM.

### *Implementation details*

For replicability, the following implementation details are provided:

*Preprocessing:* The audio was resampled at 16 kHz, and the feature extracted was the Mel spectrogram using F-bank features. The short-time Fourier transform (STFT) of the audio signal was used to get the window size at 25 ms with a 10 ms step.

*Model Training Parameters:* The model was trained using an Adam optimizer. For training and evaluation, the batch size used was 64 samples.

*Evaluation Protocol:* The speaker verification performance was evaluated on a standard split of the VoxCeleb1 test set, which ensures a fair comparison of the performance presented in this experiment with the baseline and other architectures.

This will give an even more robust, and at the same time, reproducible setup for other researchers.

## Results and discussion

### *Performance evaluation*

In summarizing the evaluation results of the proposed ResNet-SV architecture and comparing it with baseline models, we focus mainly on EER and minDCF.

Table 2

Speaker verification models comparisons

| Model                | Params        | EER         |             |             | minDCF       |              |        |
|----------------------|---------------|-------------|-------------|-------------|--------------|--------------|--------|
|                      |               | Vox1-O      | Vox1-E      | Vox1-H      | Vox1-O       | Vox1-E       | Vox1-H |
| WavLM Large          | 316.62M       | 0.62        | 0.66        | 1.32        | –            | –            | –      |
| WavLM Base+          | 94.70M        | 0.84        | 0.93        | 1.76        | –            | –            | –      |
| ECAPA-TDNN (C=512)   | 6.2M          | 1.01        | 1.24        | 2.32        | 0.1274       | 0.1418       | 0.2181 |
| ECAPA-TDNN (C=1024)  | 14.7M         | 0.87        | 1.12        | 2.12        | 0.1066       | 0.1318       | 0.2101 |
| ResNet-50            | 25.6M         | 3.95        | 4.42        | 7.33        | 0.4290       | 0.5240       | 0.6730 |
| ResNet-SV-50 (Ours)  | <b>13.76M</b> | 0.68        | 0.82        | 1.48        | 0.060        | 0.089        | 0.135  |
| ResNet-SV-101 (Ours) | 18.52M        | 0.62        | 0.73        | 1.36        | 0.055        | 0.078        | 0.127  |
| ResNet-SV-152 (Ours) | 22.45M        | <b>0.54</b> | <b>0.68</b> | <b>1.29</b> | <b>0.049</b> | <b>0.071</b> | 0.119  |

Table 2 presents performance characteristics for various speaker verification models with three test conditions of the VoxCeleb dataset: Vox1-O (original), Vox1-E (extended), and Vox1-H (hard). Each model is evaluated using EER and, where available, minDCF. The models vary in complexity, from the large-scale WavLM Large with over 300 million parameters to smaller models like the ECAPA-TDNN. Notably, the WavLM models do not report minDCF values. The data demonstrates a range of performances with some models, particularly the ResNet-SV series developed by us, showing notably lower EER and minDCF, suggesting better verification accuracy.

Moreover, in terms of performance, ResNet-SV improves remarkably with respect to the baseline models, reporting the lowest EER and minDCF over all the considered methods. This is an indication of the clear existence of a substantially higher level of accuracy and lower cost of errors in the task of speaker verification, providing further evidence of the effectiveness of our devised attention mechanisms and noise-aware training of the convolutional network architecture.

### *Analysis*

The results indicate several key insights.

*Robust to Noise:* ResNet-SV performs better, hence has stronger feature extraction capabilities, where the model could focus well on salient speaker characteristics while effectively reducing the effect of background noise by a built-in attention mechanism and training strategy aware of the noise. This is especially evident in the Vox1-H dataset, where there are complex examples.

*Efficiency and Scalability:* Convolutional connections in the deep architecture of ResNet-SV assure efficient training and inference with real-time applications, where modern hardware requires high throughput.

Strong generalization to different speakers and environments is a key attribute in practical speaker verification systems. The improved performance on the VoxCeleb1 test set – unseen during training – provides empirical evidence of the improved generalization of ResNet-SV.

### *Interpretation of the results*

We validate the effectiveness and efficiency of the proposed ResNet-SV architecture for speaker verification through our experimental results. With a significantly lower EER and minDCF compared to baseline models, it proves that it can indeed verify speakers more accurately, even under adverse conditions. In particular, attention mechanisms and noise-aware training strategies have improved the attention to relevant speaker characteristics and the robustness to background noise in the model.

The convolutional connections ensure the efficiency of the model. Thus, ResNet-SV quickly processes the inputs and makes verification decisions even with the much-required deep architecture in real-time applications. ResNet-SV balances high accuracy and performance efficiency, which is an important achievement for speaker verification.

### *Comparison with previous work*

Compared to traditional models such as ResNet and newer deep learning models like Wav-LM and ECAPA-TDNN, ResNet-SV significantly improves the verification performance. Similar attempts have largely failed to achieve this goal in their approach towards restoring the degraded accuracy of noisy conditions, while achieving computational efficiency. The improved performance of ResNet-SV, in this case, is measured by lower EER and minDCF, indicating a significant improvement over these models. Additionally, the attention mechanisms embedded in the architecture of the convolutional network for the nuances of speaker verification clearly separate our work from previous works, which mainly tackled either architectural advances or feature-level improvements, without properly covering both aspects.

### *Limitations and challenges*

Despite the promising results, a number of limitations and challenges must be acknowledged:

*Computational Resources:* Training and fine-tuning of ResNet-SV requires enormous computations, which might not be available in most of the research or production systems.

*Model Complexity:* Despite very good performance, model complexity prevents the developed architecture from being used in resource-limited environments or in applications that are sensitive to the extremely low-latency processing of information.

*Generalization in Various Conditions:* How well the model generalizes to such a wide range of settings – more diverse than those in VoxCeleb datasets and potentially including conditions that are more extreme – remains to be fully tested.

The design choices will inevitably lead to increased complexity in both the system and the training, which will help to expand the frontiers of finding more accurate, efficient, and robust speaker verification systems.

Future studies should aim to address these limitations, as well as discuss the strategies that might reduce the computational demand, make models more portable, and improve the generalization capability to cover a wider range of speakers and conditions.

### Conclusion and further research

In this work, we proposed a new convolutional network architecture for speaker verification, named ResNet-SV, having an attention mechanism and trained with noise-awareness to ensure the accuracy and robustness of the model to environmental noise. The rigorous experimental comparison also proved that the tested ResNet-SV model achieved significantly improved performance compared to the already existing baseline models, with significantly lower EER and minDCF. The integrated attention mechanisms in the convolutional blocks played an important role in focusing on only relevant speaker characteristics, while the noise-aware training strategy further improved the resilience of the model to background noise. Those results indicate the potential applicability of deep learning, more precisely the ResNet family of CNNs, to enhance the state-of-the-art in speaker verification systems towards more accurate, efficient, and robust setups.

This is very important in that it allows us to chart the path forward for speaker verification systems that can effectively work in noisy conditions with variables found in the real world, conditions, which is known to be particularly challenging. ResNet-SV not only addresses these issues, but also sets a new standard in terms of performance and makes speaker verification technology generally more applicable to many, if not most, scenarios and environments.

### REFERENCES

1. **Singh N., Agrawal A., Khan R.A.** Automatic speaker recognition: Current approaches and progress in last six decades. *Global Journal of Enterprise Information System*, 2017, Vol. 9, No. 3, Pp. 38–45. DOI: 10.18311/gjeis/2017/15973
2. **He K., Zhang X., Ren S., Sun J.** Deep residual learning for image recognition. *arXiv:1512.03385*, 2015. DOI: 10.48550/arXiv.1512.03385
3. **Chakroun R., Frikha M.** Robust text-independent speaker recognition with short utterances using Gaussian mixture models. *2020 International Wireless Communications and Mobile Computing (IWCMC)*, 2020, Pp. 2204–2209. DOI: 10.1109/IWCMC48107.2020.9148102
4. **Wei Y.** Adaptive speaker recognition based on Hidden Markov Model parameter optimization. *IEEE Access*, 2020, Vol. 8, Pp. 34942–34948. DOI: 10.1109/ACCESS.2020.2972511
5. **Ayvaz U., Gürüler H., Khan F., Ahmed N., Whangbo T., Abdusalomov A.B.** Automatic speaker recognition using Mel-Frequency Cepstral Coefficients through machine learning. *Computers, Materials & Continua*, 2022, Vol. 71, No. 3, Pp. 5511–5521. DOI: 10.32604/cmc.2022.023278
6. **Yu Y., Si X., Hu C., Zhang J.** A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, 2019, Vol. 31, No. 7, Pp. 1235–1270. DOI: 10.1162/neco\_a\_01199
7. **Chen S., Wang C., Chen Z., Wu Y., Liu S., Chen Z. et al.** WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 2022, Vol. 16, No. 6, Pp. 1505–1518. DOI: 10.1109/JSTSP.2022.3188113
8. **Desplanques B., Thienpondt J., Demuyne K.** ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN based speaker verification. *arXiv:2005.07143*, 2020. DOI: 10.48550/arXiv.2005.07143
9. **Chung J.S., Nagrani A., Zisserman A.** VoxCeleb2: Deep speaker recognition. *arXiv:1806.05622*, 2018. DOI: 10.48550/arXiv.1806.05622
10. **Wu Y.-P., Mao J.-M., Li W.-F.** Robust speech recognition by selecting mel-filter banks. *Advances in Engineering Research (AER)*, 2016, Vol. 117, Pp. 407–416.
11. **Wijayasingha L., Stankovic J.A.** Robustness to noise for speech emotion classification using CNNs and attention mechanisms. *Smart Health*, 2021, Vol. 19, Art. no. 100165. DOI: 10.1016/j.smhl.2020.100165

12. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I. Attention is all you need. *arXiv:1706.03762*, 2017. DOI: 10.48550/arXiv.1706.03762
13. Snyder D., Garcia-Romero D., Povey D., Khudanpur S. Deep neural network embeddings for text-independent speaker verification. *Proceedings of Interspeech*, 2017, Pp. 999–1003. DOI: 10.21437/Interspeech.2017-620
14. Snyder D., Chen G., Povey D. MUSAN: A music, speech, and noise corpus. *arXiv:1510.08484*, 2015. DOI: 10.48550/arXiv.1510.08484
15. Ko T., Peddinti V., Povey D., Seltzer M.L., Khudanpur S. A study on data augmentation of reverberant speech for robust speech recognition. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, Pp. 5220–5224. DOI: 10.1109/ICASSP.2017.7953152
16. Deng J., Guo J., Xue N., Zafeiriou S. ArcFace: Additive Angular Margin Loss for deep face recognition. *arXiv:1801.07698*, 2018. DOI: 10.48550/arXiv.1801.07698
17. Kingma D.P., Ba J. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980*, 2014. DOI: 10.48550/arXiv.1412.6980
18. Thian N.P.H., Bengio S. *Evidences of equal error rate reduction in biometric authentication fusion*. Switzerland: IDIAP, 2004. 27 p.
19. Scheffer N., Ferrer L., Graciarena M., Kajarekar S., Shriberg E., Stolcke A. The SRI NIST 2010 speaker recognition evaluation system. *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, Pp. 5292–5295. DOI: 10.1109/ICASSP.2011.5947552
20. Nagrani A., Chung J.S., Zisserman A. VoxCeleb: a large-scale speaker identification dataset. *arXiv:1706.08612*, 2017. DOI: 10.48550/arXiv.1706.08612

#### INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

**Aliyev Ali A.**

**Алиев Али Ахмед оглы**

E-mail: aliev.aa@edu.spbstu.ru

ORCID: <https://orcid.org/0000-0002-2813-2676>

**Molodyakov Sergey A.**

**Моляков Сергей Александрович**

E-mail: samolodyakov@mail.ru

ORCID: <https://orcid.org/0000-0003-2191-9449>

*Submitted: 24.11.2024; Approved: 22.01.2025; Accepted: 27.01.2025.*

*Поступила: 24.11.2024; Одобрена: 22.01.2025; Принята: 27.01.2025.*

Research article

DOI: <https://doi.org/10.18721/JCSTCS.18106>


UDC 621.37



## CHARACTERISTICS OF CLASS E POWER AMPLIFIER WITH COMPLEX IMPEDANCE LOAD

*H.D. Pham*  , *V.A. Sorotsky*

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

 [phamduc2511997@gmail.com](mailto:phamduc2511997@gmail.com)

**Abstract.** Unlike the well-known publications focused on the analysis of the characteristics of a Class E power amplifier (PA), in which the authors limit themselves to considering a particular case of a real load, this paper presents the results of calculating the characteristics of a Class E PA with a complex impedance load. It is especially relevant when operating in a frequency band or amplifying broadband signals. The relations given in this paper can be used to solve two types of problems. In the first case, related to “soft-switching” mode Class E PAs characteristics can be determined, when the voltage on the transistor and its derivative at the moment of turn-on are equal to zero, which eliminates switching losses. In the second case, the problem of synthesizing a matching circuit that ensures the operation of the PA in the extended frequency band can be solved. The matching circuit synthesis can be carried out under the limitations on the acceptable change of output power and voltage drop on transistor just before switching.

**Keywords:** power amplifier, efficiency, Class E, complex impedance load, analytical model, simulation modeling, harmonic balance

**Citation:** Pham H.D., Sorotsky V.A. Characteristics of Class E power amplifier with complex impedance load. *Computing, Telecommunications and Control*, 2025, Vol. 18, No. 1, Pp. 72–84. DOI: 10.18721/JCSTCS.18106



Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18106>

УДК 621.37



## ХАРАКТЕРИСТИКИ УСИЛИТЕЛЯ МОЩНОСТИ КЛАССА E ПРИ РАБОТЕ НА КОМПЛЕКСНУЮ НАГРУЗКУ

*Х.Д. Фам*  , *В.А. Сороцкий*Санкт-Петербургский политехнический университет Петра Великого,  
Санкт-Петербург, Российская Федерация [phamduc2511997@gmail.com](mailto:phamduc2511997@gmail.com)

**Аннотация.** В отличие от известных публикаций, посвященных анализу характеристик усилителя мощности (УМ) класса E, в которых авторы ограничиваются рассмотрением частного случая вещественной нагрузки, в настоящей работе представлены результаты расчета характеристик УМ класса E при работе на комплексную нагрузку, что особенно актуально при работе в полосе частот или усилении широкополосных сигналов. Приведенные в работе соотношения могут быть использованы для решения задач двух типов. В первом случае может быть осуществлен расчет характеристик УМ в «гладком» режиме, когда напряжение на транзисторе и его производная в момент коммутации электронного прибора равны нулю, что позволяет устранить коммутационные потери. Во втором случае может быть решена задача синтеза согласующей цепи, обеспечивающей работу УМ в расширенной полосе частот. Решение задачи синтеза согласующей цепи предусматривает учет ограничений на допустимое изменение выходной мощности и напряжения на транзисторе в момент коммутации.

**Ключевые слова:** усилитель мощности, КПД, класс E, комплексная нагрузка, аналитическая модель, имитационное моделирование, гармонический баланс

**Для цитирования:** Pham H.D., Sorotsky V.A. Characteristics of Class E power amplifier with complex impedance load // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 72–84. DOI: 10.18721/JCSTCS.18106

### Introduction

Along with the increase of information transmission rate, the most significant tendencies of radio communication and telecommunications equipment characteristics improvement include efficiency increase. This allows not only better use of batteries, but also opens up the possibility to reduce mass-size characteristics of devices at the expense of reduction or even full exclusion of cooling elements.

This problem can be solved by modifying power amplifiers (PAs) into the switched-mode operation, where, as is known, the efficiency can reach values of 90% and higher [1–3]. However, taking into account that switching losses due to overcharging of the output capacitance of transistors increase with increasing frequency, the most attractive mode in terms of PAs efficiency improvement is the use of the class E mode. In it, due to the use of the forming LC-circuit, switching at zero voltage (ZVS) and zero current (ZVDS) can be realized [1–3, 5, 11].

Even though the Class E mode of operation has been known for quite a long time, due to its obvious advantages, it still arouses the interest of specialists in the field of radio- and telecommunications. This has been reflected not only in a number of monographs by well-known specialists in switched-mode PAs [1–3, 6–8], but also in numerous publications that have appeared in recent years [9, 10, 12, 14 etc.].

Unfortunately, the authors of these publications limited themselves to considering only one of the possible cases, when the amplifier operates on a resistive load at a fixed frequency. At the same time, it should

be taken into account that antennas used in wireless communication systems usually have  $VSWR \leq 2$ . It follows that the PA load in general is a complex impedance load and the use of the relations obtained just for resistive load can lead to an error.

The relations given in this paper can be used for solving two types of problems. In the first case, using known parameters of the transistor (maximum drain-to-source voltage and drain current), it is possible to determine the values of the real and imaginary parts of the complex impedance load, which are necessary for the transistor commutation in the “soft-switching” mode, when drain-to-source voltage and its derivative are simultaneously equal to zero, when the transistor is turned on [1–3, 5–9, 11]. Implementation of these conditions leads to an increase of PAs efficiency due to the elimination of switching power losses in the transistor, but it should be noted that a rigorous solution of this problem can be realized only at a fixed frequency.

No less actual for practice is the task of the second type, when it is necessary to ensure the operation of Class E PAs in the frequency band provided that the reduction of efficiency does not exceed the permissible values. This approach can be useful when using Class E PAs for amplification of signals with high peak-to-average power ratio (PAPR). In other words, the solution of the second type of problem does not guarantee the complete elimination of switching power losses at each of the operating frequencies or when changing the output power level. At the same time, this solution can lead to a significant reduction of switching losses in a certain frequency band within permissible change in the PA output power.

Taking into account the above, the paper goals are as follows:

- 1) determination of Class E PA characteristics when operating on a complex load;
- 2) estimation of the load impedance real and imaginary parts variation limits proceeding from allowable deviations from the nominal values of PA output power and transistor turn-on voltage, which is relevant for PA operation in the frequency band.

#### Analysis of class E power amplifier with a complex impedance load

The schematic of Class E PA is shown in Fig. 1, *a*. The analytical model was developed using several assumptions, including:

- output capacitance of the transistor does not depend upon the drain-to-source voltage and has a constant value;
- transistor turn-on and turn-off times are negligibly small compared to the duration of the output waveform period;
- on-state resistance of the transistor can be neglected;
- blocking capacitance  $C_b$  is large enough, so that the voltage across it can be considered constant and equal to the supply voltage  $V_{dd}$ . This gives a reason to replace it with a constant voltage source during the analysis.

Since in transmitters at the PA output frequency-selective circuits are used in order to attenuate higher harmonics, we will assume that the voltage across the load  $Z(j\omega) = R + jX$  (Fig. 1, *a*) varies according to the harmonic law. This opens the possibility to apply the harmonic balance method when creating the analytical PA model, replacing the complex load by a voltage source  $V_m \sin(\omega t + \varphi)$  [4, 14], with unknown amplitude  $V_m$  and initial phase  $\varphi$  to be determined as a result of the analysis (Fig. 1, *b*).

Considering the adopted assumptions, transistor can be represented in the equivalent circuit (Fig. 1, *b*) as an ideal switch in parallel with a capacitance. This capacitance is equal to the combined output capacitance of the transistor and stray capacitances.

Let us examine the steady-state operation of the circuit, assuming that during the time interval  $0 < \omega t \leq \pi$ , the transistor is in the off-state (switch  $S$  open). In the most interesting “soft-switching” mode of operation of Class E PA, two conditions must be satisfied [1–3, 11]:

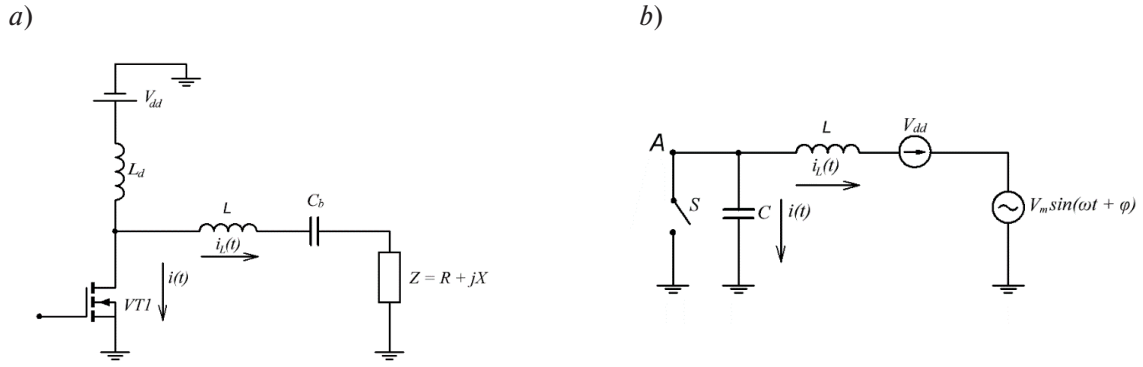


Fig. 1. a) functional diagram of a Class E PA; b) Class E PA equivalent substitution diagram

$$\begin{cases} v_C(t) \Big|_{t=\frac{\pi}{\omega}} = 0, & (1) \\ \frac{dv_C(t)}{dt} \Big|_{t=\frac{\pi}{\omega}} = 0. & (2) \end{cases}$$

According to Kirchhoff's laws for currents and voltages during the time interval  $0 < \omega t \leq \pi$  (stage 1), the following system of equations can be created for the circuit in Fig. 1, b):

$$\begin{cases} L \frac{di_L(t)}{dt} + v_C(t) - V_m \sin(\omega t + \varphi) - V_{dd} = 0, & (3) \\ C \frac{dv_C(t)}{dt} - i_L(t) = 0, & (4) \end{cases}$$

where  $v_C(t)$  is the voltage stress across the switch,  $i_L(t)$  is the current in inductance of forming circuit (FC).

The voltage across the FC capacitance can be found solving the system of equations (3) and (4):

$$v_C(t) = k_1 \sin(\omega_0 t) + k_2 \cos(\omega_0 t) + 1 + \frac{\omega_0^2}{\omega_0^2 - \omega^2} V_m \sin(\omega t + \varphi), \quad (5)$$

where  $\omega_0 = \frac{1}{\sqrt{LC}}$ ,  $k_{1,2}$  is the constants, which are determined using the initial conditions.

To facilitate further calculations, it is advisable to express the voltage across the capacitance (5) in normalized form:

$$v(\theta) = k_1 \sin(v\theta) + k_2 \cos(v\theta) + 1 + \frac{v^2}{v^2 - 1} q \sin(\theta + \varphi), \quad (6)$$

where

$$v(\theta) = v_C(t) / V_{dd}, \quad q = V_m / V_{dd}, \quad v = \omega_0 / \omega, \quad \theta = \omega t. \quad (7)$$

At the time interval  $\pi < \omega t \leq 2\pi$  (stage 2), with the switch  $S$  in the closed state, the current flowing through the inductance  $L$  is given by:

$$i_L(\theta) = \frac{1}{\omega L} \int_{\pi}^{\theta} [V_{dd} + V_m \sin(\theta + \varphi)] d\theta + I_{02}, \quad (8)$$

where  $I_{02}$  is the current through the inductance at the time  $\omega t = \pi$ .

After normalization, expression (8) will be modified as follows:

$$i(\theta) = v(\theta - \pi) - vq [\cos(\theta + \varphi) + \cos \varphi] + \frac{I_{02}}{V_{dd}/\rho}, \quad (9)$$

where  $i(\theta) = \frac{i_L(\theta)}{V_{dd}/\rho}$ ,  $\rho = \sqrt{L/C}$ .

Relations (6) and (9), which describe the behavior of current and voltage across the FC capacitance, depend on six unknown parameters:  $v, q, \varphi, k_1, k_2, I_{02}$ .

To find the value of  $I_{02}$  using equation (9), let us apply the following condition:

$$I_{02} = i_L(\theta) \Big|_{\theta=\pi} = i_C(\theta) \Big|_{\theta=\pi} = \omega C \frac{dv(\theta)}{d\theta} \Big|_{\theta=\pi} = 0. \quad (10)$$

At the time  $\theta = 0$ , the transistor was in the on-state and the capacitance  $C$  was discharged. Using equation (6), we can derive the equation to determine  $k_2$ :

$$k_2 + 1 + \frac{v^2}{v^2 - 1} q \sin \varphi = 0. \quad (11)$$

The equation used to determine  $k_1$  is derived from equation (6), while also considering condition (1):

$$k_1 \sin(\pi v) + k_2 \cos(\pi v) + 1 - \frac{v^2}{v^2 - 1} q \sin \varphi = 0. \quad (12)$$

By solving equations (11) and (12), we can express the constants  $k_1, k_2$  in terms of the unknowns  $q$  and  $\varphi$ :

$$k_1 = -\text{ctg}(\pi v) k_2 - \left( 1 - \frac{v^2}{v^2 - 1} q \sin \varphi \right) / \sin(\pi v); \quad (13)$$

$$k_2 = -1 - \frac{v^2}{v^2 - 1} q \sin \varphi. \quad (14)$$

Taking into account that the  $DC$  resistance of the inductor  $L_d$  is zero, the average voltage at point  $A$  (Fig. 1,  $b$ ) is equal to the supply voltage  $V_{dd}$ . Considering this, we can express it using equation (6):

$$\frac{v_{cp}}{V_{dd}} = \frac{1}{2\pi} \int_0^{2\pi} v(\theta) d\theta = 1. \quad (15)$$

After performing the necessary calculations, we get the equation that includes the unknown variables  $q$  and  $\varphi$ :

$$k_1 [\cos(\pi\nu) - 1] - k_2 \sin(\pi\nu) - \pi\nu - \frac{2\nu^2}{\nu^2 - 1} q \sin \varphi = 0. \quad (16)$$

The second equation for finding these unknowns is obtained by differentiating (6) and equating the resulting expression, in accordance with condition (2), to zero:

$$k_1 \cos(\pi\nu) - k_2 \sin(\pi\nu) - \frac{\nu^2}{\nu^2 - 1} q \cos \varphi = 0. \quad (17)$$

By substituting relations (13) and (14) into equations (16) and (17), we finally get a system of two equations to determine the unknowns  $q$  and  $\varphi$ . The normalized frequency  $\nu$  present in equations (16) and (17) can be treated as an independent variable.

Solving (16) and (17) bring us to:

$$\varphi = \frac{1}{2} \arcsin(x); \quad (18)$$

$$q = \pm \sqrt{\frac{1 + \sqrt{1 - x^2}}{2}} \frac{\nu^2}{\nu^2 - 1} (A - B), \quad (19)$$

where,

$$A = k_1 \cos(\pi\nu); \quad (20)$$

$$B = k_2 \sin(\pi\nu); \quad (21)$$

$$x = 1 - \frac{\pi\nu}{A - B}. \quad (22)$$

The time diagrams of the normalized voltage on FC capacitance  $v(\theta)$  and the normalized current  $i(\theta)$  referring to various values of the parameter  $\nu$  are shown in Fig. 2.

As can be seen in Fig. 2, the behavior of the voltage across the capacitance  $C$  at  $\theta = \pi$  satisfies conditions (1) and (2). In this case, the change of the relative frequency in the range from  $\nu = 1.3$  to  $\nu = 2.1$  has a comparatively weak effect on the value of the maximum voltage across the FC capacitance, the deviation of which does not exceed 5%. As for the maximum current through the transistor, here the influence of the parameter  $\nu$  is much stronger and is accompanied by its decrease by approximately four times.

#### The load impedance to provide a “soft-switching” operation

The harmonic balance method enables to determine the characteristics of Class E PAs in a general form, eliminating the need to consider the structure and parameters of the load circuit. For this purpose, the complex load is replaced by an equivalent harmonic voltage source. When addressing the issues related to the operation of the PAs in a frequency band, it is advisable to shift from representing the load as a voltage source to using complex impedance  $Z$ . This will allow us to determine the behavior of the real and imaginary parts of  $Z$ , which are necessary to provide a “soft-switching” mode while the relative frequency

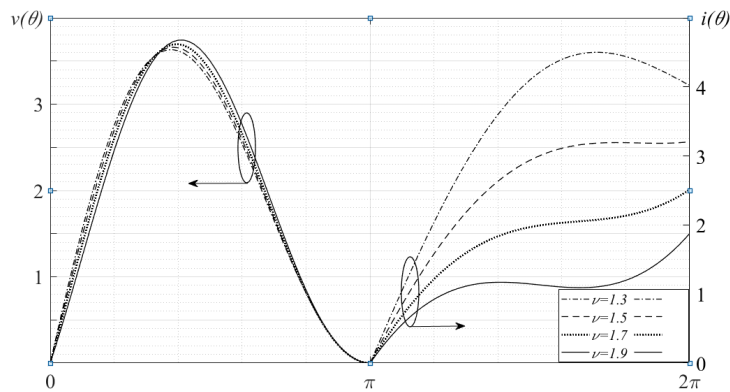


Fig. 2. Time-normalized diagrams of the voltage across the FC capacitance and current through the transistor at various values of the parameter  $\nu$

$\nu$  changes. It will open the possibility of solving the problem of synthesizing a matching circuit, which enables the operation of the PA in the frequency band.

To determine the real and imaginary parts of  $Z$ , we use equation (9) to find the amplitude and phase of the first harmonic in the load current:

$$|I_1| = \sqrt{a_1^2 + b_1^2}; \quad (23)$$

$$\varphi_I = \arctg \frac{b_1}{a_1}, \quad (24)$$

where  $a_1$  and  $b_1$  are the coefficients of the Fourier series:

$$a_1 = \frac{1}{\pi} \int_0^{2\pi} i_L(\theta) \sin \theta d\theta = \frac{V_{dd}\nu}{\pi\rho} \left( \pi + \frac{1}{2} q\pi \sin \varphi + 2q \cos \varphi \right); \quad (25)$$

$$b_1 = \frac{1}{\pi} \int_0^{2\pi} i_L(\theta) \cos \theta d\theta = \frac{V_{dd}\nu}{\pi\rho} \left( 2 + \frac{1}{2} q\pi \cos \varphi \right). \quad (26)$$

The modulus and phase of the complex impedance load are equal to:

$$|Z| = \frac{|U_1|}{|I_1|} = \frac{V_m}{\sqrt{a_1^2 + b_1^2}}; \quad (27)$$

$$\varphi_Z = \varphi - \varphi_I. \quad (28)$$

Using relations (27) and (28), it is straightforward to determine both the real and imaginary parts of the load impedance:

$$R = |Z| \cos \varphi_Z; \quad (29)$$

$$X = |Z| \sin \varphi_Z. \quad (30)$$

The current consumed from the power supply is equal to:

$$I_0 = \frac{1}{2\pi} \int_0^{2\pi} i(\theta) d\theta = \frac{V_{dd}}{2\pi\rho} \left[ \frac{\pi^2 v}{2} - vq(2 \sin \varphi + \pi \cos \varphi) \right]. \quad (31)$$

Fig. 3 shows the behavior of the normalized characteristics of the PA as the parameter  $v$  is varied. It includes the normalized power of the first harmonic  $P_{1n}$ , the normalized real  $R$  and imaginary  $X$  parts of the load impedance, the current stress on transistor  $I_{max}$ , the maximum voltage  $V_{max}$  across the FC capacitance, and the current consumed from the power supply  $I_0$ .

Based on the analysis of the curves shown in Fig. 3, it can be seen that the real part of the load impedance decreases rapidly to zero when the parameter  $v$  exceeds 2. Concurrently, there is a noticeable reduction in both the load power and the current consumed from the power supply. As the parameter  $v$  decreases, the current stress  $I_{max}$  increases, particularly when  $v$  is less than 1.2. This trend can lead to some complications in selecting the appropriate transistor.

The features outlined above lead to the following conclusion. When the PA operates within the specified frequency band, it is reasonable to limit the variation of the relative frequency dispersion to the range of  $1.3 \leq v \leq 2.0$ . The values of the normalized parameters that correspond to this condition, obtained from analytical calculations and ensuring the realization of the “soft-switching” mode in the PA, are given in Table 1.

Table 1

**Results of the analytical calculations**

| $v$               | 1.3   | 1.4   | 1.5   | 1.6   | 1.7   | 1.8   | 1.9   | 2.0   |
|-------------------|-------|-------|-------|-------|-------|-------|-------|-------|
| $k_1$             | 3.97  | 3.17  | 2.60  | 2.15  | 1.77  | 1.42  | 1.10  | 0.79  |
| $k_2$             | 2.94  | 1.47  | 0.60  | 0.03  | -0.36 | -0.64 | -0.85 | -1.00 |
| $q$               | 1.61  | 1.24  | 1.02  | 0.92  | 0.93  | 0.99  | 1.08  | 1.18  |
| $\varphi$ , deg   | -89.1 | -76.6 | -60.5 | -42.8 | -26.7 | -14.4 | -5.7  | 0     |
| $\varphi_r$ , deg | -70.8 | -70.9 | -71.1 | -71.4 | -72.0 | -73.0 | -75.0 | -79.2 |
| $\varphi_z$ , deg | -18.3 | -5.7  | 10.6  | 28.6  | 45.7  | 58.6  | 69.3  | 79.2  |

To evaluate the reliability of the analytical model, one can compare the data presented in Table 2. This table lists the main characteristics of the PA at various relative frequency values, obtained through both analytical calculations (A) and simulation (S). The discrepancy between these results can be assessed using the relative error ( $\Delta\delta$ ).

Table 2

**Comparison of analytical and simulation results**

| $v$  | $V_s(\max)/V_{dd}$ |      |                    | $q$  |      |                    | $\varphi$ , deg |       |                |
|------|--------------------|------|--------------------|------|------|--------------------|-----------------|-------|----------------|
|      | A                  | S    | $\Delta\delta$ , % | A    | S    | $\Delta\delta$ , % | A               | S     | $\Delta\delta$ |
| 1.40 | 3.63               | 3.60 | 0.8                | 1.24 | 1.29 | 3.9                | -76.6           | -70.1 | 5.5            |
| 1.60 | 3.68               | 3.65 | 0.8                | 0.94 | 0.99 | 5.1                | -42.8           | -40.4 | 2.4            |
| 1.80 | 3.70               | 3.66 | 1.1                | 0.99 | 1.13 | 12.4               | -14.4           | -18.5 | -4.1           |
| 2.00 | 3.78               | 3.70 | 2.2                | 1.17 | 1.30 | 10.0               | 0.10            | -5.0  | -5.1           |

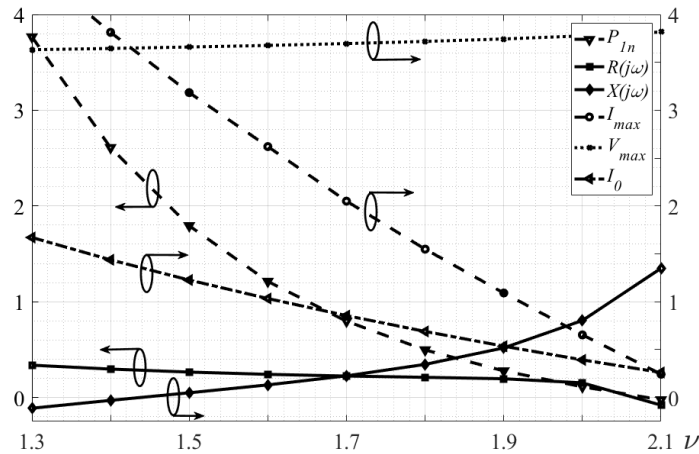


Fig. 3. Normalized characteristics of the PA as the parameter  $\nu$  is varied

The table shows that the error in determining the voltage amplitude across the load does not exceed 10.0...12.4%. The error in determining transistor voltage stress is no more than 2.2%, and the initial phase  $\varphi$  is 5.5 degrees, respectively.

#### Equal values of output power and switching losses lines diagram

The dependencies of the real and imaginary parts of the load impedance, identified earlier, enable to formulate the problem of designing a matching circuit that ensures the operation of PA in the frequency band under specified permissible deviation of output power and efficiency reduction on account of transistor switching losses.

When amplifying variable-envelope signals, as well as when operating within the frequency band, the conditions for “soft-switching” (1) and (2) cannot be always met [15]. We assume that the voltage across the transistor at turn-on is equal to  $\delta v$  ( $0 \leq |\delta v| < 1$ ):

$$v_{Cs}(\theta) \Big|_{\theta=\pi} = \delta v. \quad (32)$$

Using (6), we write (27) in the following form:

$$k_1^* \sin(\pi\nu) + k_2^* \cos(\pi\nu) + 1 - \frac{\nu^2}{\nu^2 - 1} q^* \sin \varphi^* = \delta v. \quad (33)$$

In a similar manner to equations (13) and (14), we can express:

$$k_1^* = -\cot(\pi\nu) k_2^* - \csc(\pi\nu) \left( 1 - \frac{\nu^2}{\nu^2 - 1} q^* \sin \varphi^* \right) + \csc(\pi\nu) \delta v; \quad (34)$$

$$k_2^* = -1 - \frac{\nu^2}{\nu^2 - 1} q^* \sin \varphi^*. \quad (35)$$

To find two unknown parameters  $q^*$  and  $\varphi^*$  we use relations (2) and (16), respectively:

$$k_1^* [\cos(\pi\nu) - 1] - k_2^* \sin(\pi\nu) - \pi\nu - \frac{2\nu^2}{\nu^2 - 1} q^* \sin \varphi^* = 0; \quad (36)$$



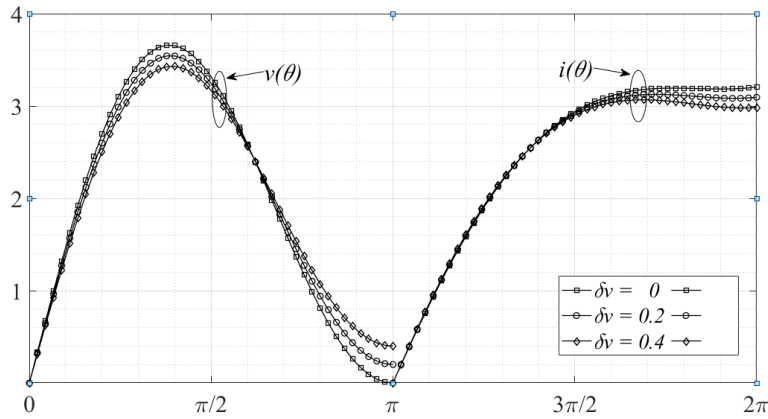


Fig. 4. Normalized voltage and current curves at different  $\delta v$

$$k_1^* \cos(\pi v) - k_2^* \sin(\pi v) - \frac{v^2}{v^2 - 1} q^* \cos \varphi^* = 0. \quad (37)$$

By solving equations (32), (33), (36), and (37), one can determine the parameters  $q^*$ ,  $\varphi^*$ ,  $k_1^*$  and  $k_2^*$ , which correspond to the PA mode with non-zero voltage across the transistor at turn-on. Time diagrams illustrating the behavior of the voltage across the FC capacitance and transistor current for different values of  $\delta v$  are shown in Fig. 4.

The sensitivity of the parameter  $\delta v$  to the deviation of the real and imaginary parts of the load impedance from their nominal values determined by relations (24) and (25) can be evaluated using Fig. 5, *a, b*.

The analysis of these graphs shows that the voltage across the transistor at turn-on is rather sensitive to  $\Delta R(j\omega)$  and  $\Delta X(j\omega)$  variations. Specifically, if we assume that turn-on voltage does not exceed a value of  $\delta v = 0.2$ , then for  $v = 1.5$ , the imaginary part of the load impedance can deviate from the nominal value by 23%, while the real part can only deviate by 4.3%. Conversely, when  $\delta v = 0.2$  and  $v = 1.7$ , the relative deviation of the imaginary part must not exceed 3%, while the deviation of the real part can be no more than 5.3%.

Taking into account such an ambiguous character of the sensitivity of  $\delta v$  to deviations in  $\Delta R(j\omega)$  and  $\Delta X(j\omega)$ , it is prudent to utilize a series of curves representing equal power values (continuous lines) alongside a series of curves for equal values of the parameter  $\delta v$  (dashed lines). Both series of curves are presented in the plane  $[ReZ(j\omega), ImZ(j\omega)]$  (Fig. 6).

The use of these series of curves opens up the approach for solving the problem of synthesizing a matching circuit based on the condition of achieving the maximum frequency bandwidth:

$$\Delta v = (v_{\max} - v_{\min}) \rightarrow \max, \quad (38)$$

at the specified allowable limits for output power variation and transistor turn-on voltage:

$$\left[ P(\vec{x}) - P_n \right] / P_n \leq \Lambda; \quad (39)$$

$$v_c(\vec{x}) \Big|_{\omega t = \pi} \leq (v_c)_{al}, \quad (40)$$

where  $\vec{x}$  is the vector of variable parameters and  $\Lambda$ ,  $(v_c)_{al}$  are the allowable limits for output power variation and transistor turn-on voltage.

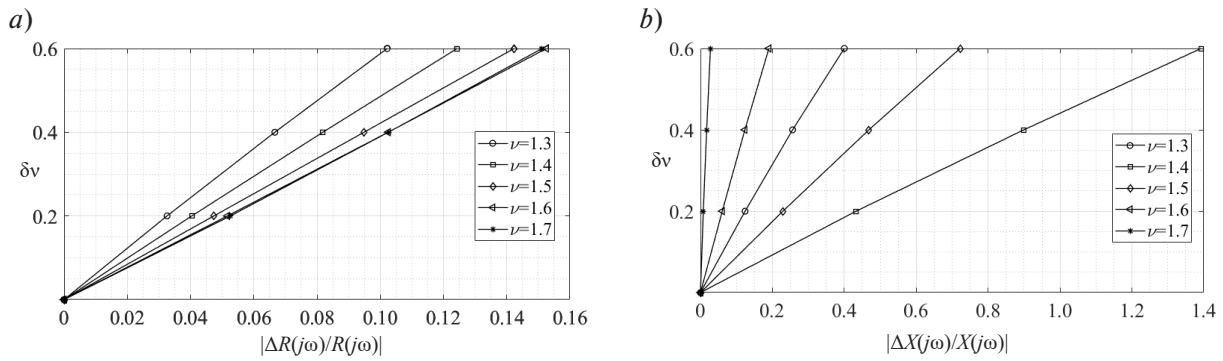


Fig. 5. The sensitivity of the parameter  $\delta\nu$  to the deviation of the real (a) and imaginary (b) parts of the load resistance from their nominal values

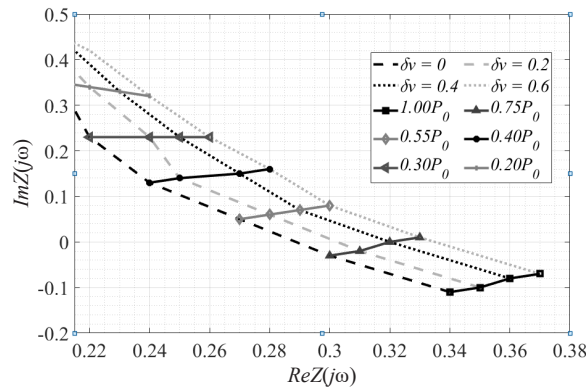


Fig. 6. Series of curves of equal output power values and equal  $\delta\nu$  values in  $[ReZ(j\omega), ImZ(j\omega)]$  plane

It is important to note that based on the behavior of both curves series presented in Fig. 6, it is impossible to increase the operating frequency bandwidth of a Class E PA without negatively impacting at least one of its characteristics – either the relative output power or the switching losses in the transistor. To illustrate this, one can fix either the output power or the voltage across the transistor during turn-on. The relative frequency of the signal  $\nu$  being varied, the trajectory of the corresponding point of load impedance  $Z$  will intersect the lines of equal levels associated with the other parameter.

### Conclusion

Summarizing the results presented in this paper, we highlight the following:

1. Based on the harmonic balance method, the analysis of Class E PA when operating under complex load was carried out. The relationships have been obtained that allow one to determine the real and imaginary parts of the load impedance for different values of the relative frequency  $\nu = \omega_0/\omega$ , necessary for the implementation of the “soft-switching” mode, that means elimination of switching power losses in transistors.

2. The assessment of the adequacy of the proposed analytical model has been carried out which confirmed that the error in determining the voltage amplitude across the load does not exceed 10.0...12.4%. The error in determining the transistor voltage stress is no more than 2.2%, while the initial phase  $\varphi$  has an error of approximately 5.5 degrees.

3. An approach is proposed to solve the problem of matching a Class E PA with antenna when operating in a frequency band proceeding from the allowable output power deviation and efficiency reduction under the frequency change.

## REFERENCES

1. **Grebennikov A.** *RF and Microwave Power Amplifier Design*, 2<sup>nd</sup> ed. New York: McGraw-Hill Professional Engineering, 2015. 672 p.
2. **Kazimierczuk M.K.** *RF Power Amplifiers*, 2<sup>nd</sup> ed. New Jersey: Wiley, 2014. 688 p.
3. **Sokal N.O., Grebennikov A.** *Switchmode RF Power Amplifiers*, 1<sup>st</sup> ed. USA, UK: Elsevier Inc., 2007. 425 p. DOI: 10.1016/B978-0-7506-7962-6.X5028-X
4. **Sorotsky V., Pham H.D.** A novel approach to studying class E power amplifier with a complex impedance load. *2023 International Conference on Electrical Engineering and Photonics (EExPolytech)*, 2023, Pp. 38–41. DOI: 10.1109/EExPolytech58658.2023.10318779
5. **Acar M., Annema A.J., Nauta B.** Analytical design equations for class-E power amplifiers. *IEEE Transactions on Circuits and Systems – I: Regular Papers*, 2007, Vol. 54, No. 12, Pp. 2706–2717. DOI: 10.1109/TCSI.2007.910544
6. **Al Tanaly A., Sayed A., Boeck G.** Broadband GaN switch mode class E power amplifier for UHF applications. *2009 IEEE MTT-S International Microwave Symposium Digest*, 2009, Pp. 761–764. DOI: 10.1109/MWSYM.2009.5165808
7. **Chen K., Peroulis D.** Design of highly efficiency broadband class-E power amplifier using synthesized low-pass matching networks. *IEEE Transactions on Microwave Theory and Techniques*, 2011, Vol. 59, No. 12, Pp. 3162–3173. DOI: 10.1109/TMTT.2011.2169080
8. **Lee Y.-S., Jeong Y.-H.** A high-efficiency class-E GaN HEMT power amplifier for WCDMA applications. *IEEE Microwave and Wireless Components Letters*, 2007, Vol. 17, No. 8, Pp. 622–624. DOI: 10.1109/LMWC.2007.901803
9. **Raab F.H.** HF class-E power amplifier with improved efficiency for mismatched loads, *2023 53<sup>rd</sup> European Microwave Conference (EuMC)*, 2023, Pp. 384–387. DOI: 10.23919/EuMC58039.2023.10290244
10. **Tong Z., Rivas-Davila J.M.** Wideband push-pull class E amplifier for RF power delivery. *2023 IEEE 24<sup>th</sup> Workshop on Control and Modeling for Power Electronics (COMPEL)*, 2023, Pp. 1–7. DOI: 10.1109/COMPEL52896.2023.10220982
11. **Eroglu A.** *Introduction to RF Power Amplifier Design and Simulation*, 1<sup>st</sup> ed. Boca Raton: CRC Press, 2016. 449 p. DOI: 10.1201/9781315215297
12. **Aditya K., Pradhan S., Raj A.** Class-E power amplifier design for wireless power transfer. *2024 IEEE 3<sup>rd</sup> International Conference on Electrical Power and Energy Systems (ICEPES)*, 2024, Pp. 1–5. DOI: 10.1109/ICEPES60647.2024.10653495
13. **Sorotsky V.A., Pham H.D., Zudov R.** Design of a class E power amplifier with complex impedance load. *2024 International Conference on Electrical Engineering and Photonics (EExPolytech)*, 2024, Pp. 76–78. DOI: 10.1109/EExPolytech62224.2024.10755740
14. **Racha G., Kishore K.L., Kamatham Y., Perumalla S.R.** Design of 13.56 MHz class-E power amplifier using inductive load with shunt capacitance for short-range communications. *2024 3<sup>rd</sup> International Conference for Advancement in Technology (ICONAT)*, 2024, Pp. 1–5. DOI: 10.1109/ICONAT61936.2024.10774703
15. **Acar M., Annema A.J., Nauta B.** Generalized analytical design equations for variable slope class-E power amplifiers. *2006 13<sup>th</sup> IEEE International Conference on Electronics, Circuits and Systems*, 2006, Pp. 431–434. DOI: 10.1109/ICECS.2006.379817

## INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Pham Huu Duc

Фам Хью Дык

E-mail: phamduc2511997@gmail.com

ORCID: <https://orcid.org/0009-0004-1628-1772>

**Sorotsky Vladimir A.**  
**Сороцкий Владимир Александрович**  
E-mail: sorotsky@mail.spbstu.ru

*Submitted: 31.12.2024; Approved: 13.02.2025; Accepted: 18.02.2025.*

*Поступила: 31.12.2024; Одобрена: 13.02.2025; Принята: 18.02.2025.*

Research article

DOI: <https://doi.org/10.18721/JCSTCS.18107>

UDC 621.372.543



## MONOLITHIC MICROWAVE BANDPASS FILTERS DESIGN FOR S- AND C-BANDS

*N.V. Ivanov*  , *I.A. Rumyancev* 

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

 [ivanovnick@mail.ru](mailto:ivanovnick@mail.ru)

**Abstract.** This paper discusses the development of monolithic integrated circuits of S- and C-band bandpass filters from modeling to measurement of experimental samples within the pHEMT05D technology. Bandpass filters with central frequencies of 2.5, 3 and 5 GHz with a relative bandwidth of 40% and a rejection in the stop band of at least –30 dB have been developed. When developing the topology, optimization methods were used according to the requirements for frequency characteristics. Two variants of carrying out measurements of experimental samples are proposed: with welding on PCB using the Kulicke & Soffa 4256 station and from the plate using the Cascade Microtech EP6RF probe station. The measurements were carried out using the Rohde Schwarz ZVA40 vector network analyzer. The obtained samples showed deviations in frequency characteristics of approximately 10% and an increase in losses in the passband within 2 dB. High repeatability of the result within the manufactured plate is demonstrated, which confirms the stability of the obtained result.

**Keywords:** MMIC design, bandpass filters, S-band, C-band, GaAs pHEMT

**Citation:** Ivanov N.V., Rumyancev I.A. Monolithic microwave bandpass filters design for S- and C- bands. Computing, Telecommunications and Control, 2025, Vol. 18, No. 1, Pp. 85–97. DOI: 10.18721/JCSTCS.18107

Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18107>

УДК 621.372.543



## РАЗРАБОТКА МОНОЛИТНЫХ ИНТЕГРАЛЬНЫХ СХЕМ ПОЛОСОВЫХ СВЧ ФИЛЬТРОВ S- И C-ДИАПАЗОНОВ

Н.В. Иванов  , И.А. Румянцев 

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

 [ivanovnick@mail.ru](mailto:ivanovnick@mail.ru)

**Аннотация.** В данной статье рассматривается разработка монолитных интегральных схем полосовых фильтров S- и C-диапазона от моделирования до измерения экспериментальных образцов в рамках технологии pHEMT05D. Разработаны полосовые фильтры с центральными частотами 2,5, 3 и 5 ГГц с относительной шириной полосы пропускания 40% и подавлением в полосе заграждения не менее –30 дБ. При разработке топологии применялись методы оптимизации по требованиям к частотным характеристикам. Предложены два варианта проведения измерений экспериментальных образцов: с разваркой на платы посредством станции Kulicke & Soffa 4256 и с пластины посредством зондовой станции Cascade Microtech EP6RF. Измерения проводились при помощи векторного анализатора цепей Rohde Schwarz ZVA40. Полученные образцы показали отклонения по частотным характеристикам около 10% и увеличение потерь в полосе пропускания в пределах 2 дБ. Продемонстрирована высокая повторяемость результата в пределах изготовленной пластины, что подтверждает стабильность полученного результата.

**Ключевые слова:** проектирование МММС, полосовые фильтры, S-диапазон, C-диапазон, GaAs pHEMT

**Для цитирования:** Ivanov N.V., Rumyancev I.A. Monolithic microwave bandpass filters design for S- and C- bands // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 85–97. DOI: 10.18721/JCSTCS.18107

### Introduction

In most modern telecommunication systems, microwave bandpass filters are used as a preselectors in front-end of the design. As system-on-chip concept is widely used, MMIC realization is of interest. In the last decade, a large number of works [1–9] have been devoted to the problems of designing MMIC filters.

In recent years, many restrictions have been introduced in Russia on the import of semiconductor components. In addition, cooperation with foreign semiconductor manufacturers has become difficult. These circumstances make the task of development of semiconductor integrated circuits using domestic technologies relevant.

In this work, S- and C- bands MMIC filter designs are presented from realization choice up to samples measurement. The GaAs pHEMT technological process of Svetlana-Rost JSC (pHEMT05D) was used for design and sample production [10–13].

### Design parameters

For this work, a series of bandpass filters with a fractional bandwidth (FBW) of 40% were designed. They cover range from 2 to 6 GHz with center frequencies of 2.5, 3 and 5 GHz. Stopband attenuation is set to 30 dB as one of the typical parameters for general applications. Insertion loss (IL) is set to be up to 2 dB as desired parameter, expecting reasonable exceedances for real samples. Since the approximate dimensions of a microstrip realization, even a compact hairpin structure, are too large to consider (for a center frequency of 2.5 GHz, the topology area is 1560×10675 μm excluding contact pads and requirements for

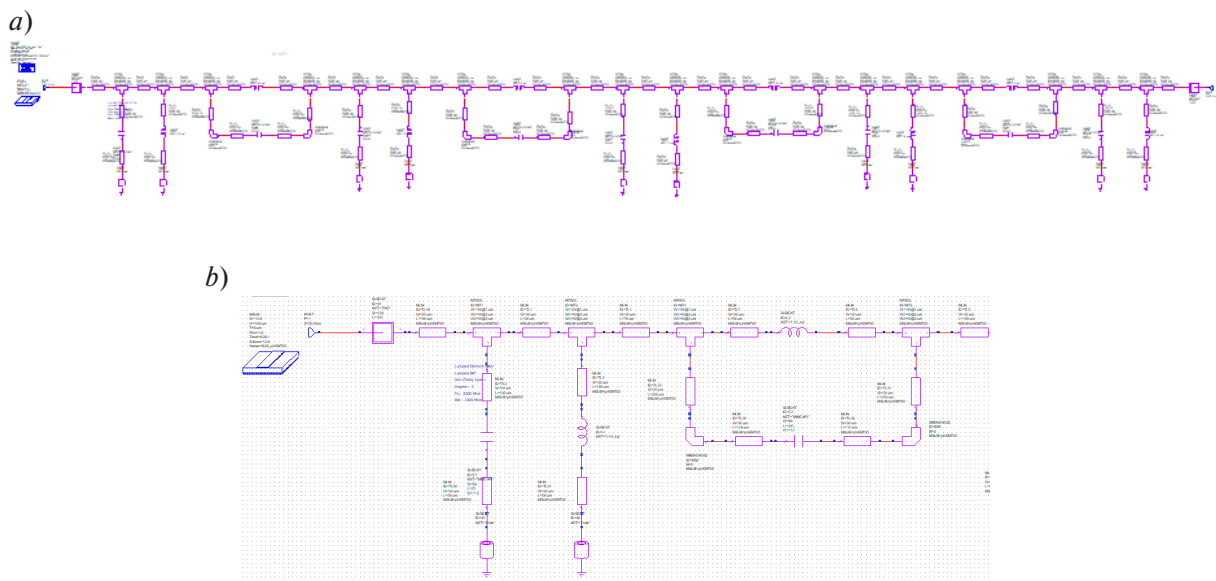


Fig. 1. Structure of a lumped element filter (a) with first two resonators in an enlarged scale (b)

the crystal boundary), lumped element structure is used in this work. The parameters for filter design are summarized in Table 1.

Table 1

**Filter design parameters**

| Center frequency, GHz | Passband, GHz | Stopband, GHz | Insertion Loss, dB | Stopband attenuation, dB |
|-----------------------|---------------|---------------|--------------------|--------------------------|
| 2.5                   | 1             | 2.5           | 2                  | 30                       |
| 3                     | 1.2           | 3             |                    |                          |
| 5                     | 2             | 5             |                    |                          |

For filters based on lumped elements, a structure based on the Type 1 generalized Chebyshev approximation with a passband ripple of 0.011 dB was selected. The structure of the prototype for the case of a center frequency of 2.5 GHz is shown in Fig. 1. The remaining filters have a similar structure; only the capacitance and inductance values differ proportionally to the frequency change.

**Simulation and optimization**

The S-parameters simulation results of the filter based on lumped elements with a center frequency of 2.5 GHz before optimization, after optimization and after rounding the values for production are shown in Fig. 2.

Similarly, Fig. 3 and 4 show the S-parameters simulation results for filters with center frequencies of 3 and 5 GHz, respectively. It should be noted that with increasing frequency, the influence of the inaccuracy of the implemented nominal values and microstrip lines on the frequency characteristics becomes greater, while the passband shifts below the target.

**Filter topologies**

Fig. 5–8 show the developed topologies of bandpass filters on lumped elements, in order of increasing central frequency. The sizes of the topologies (excluding the requirements for the crystal boundary) are given in the figure captions.

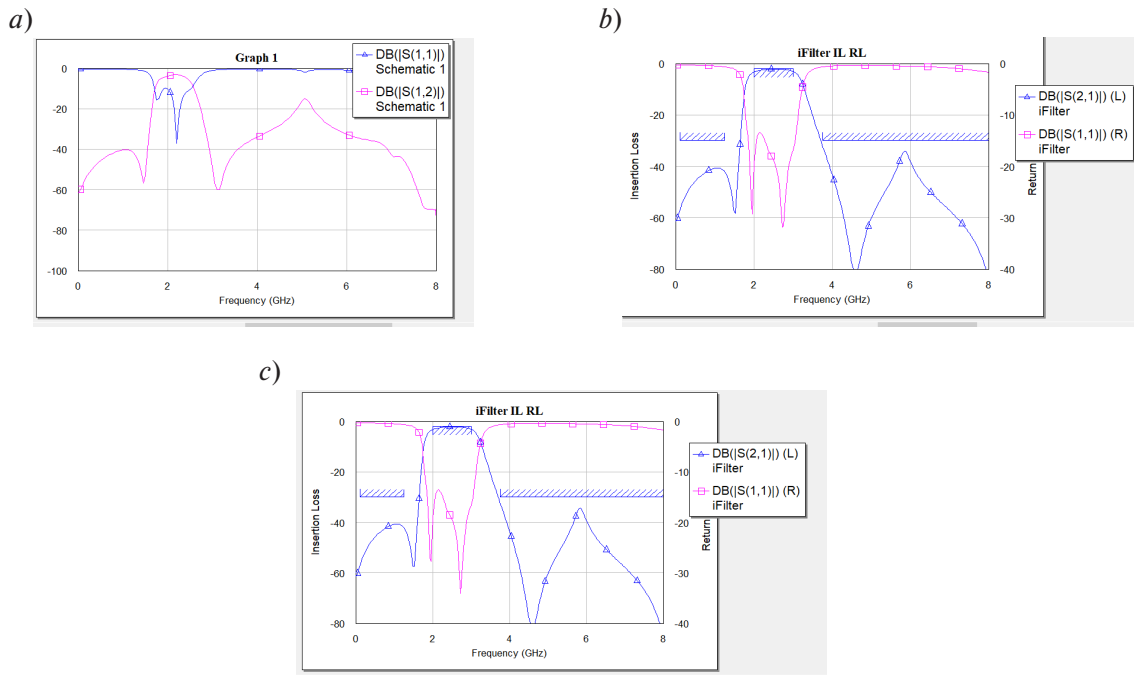


Fig. 2. S-parameters simulation results of the bandpass filter based on lumped elements with a center frequency of 2.5 GHz: before optimization (a), after optimization (b), after rounding the values (c)

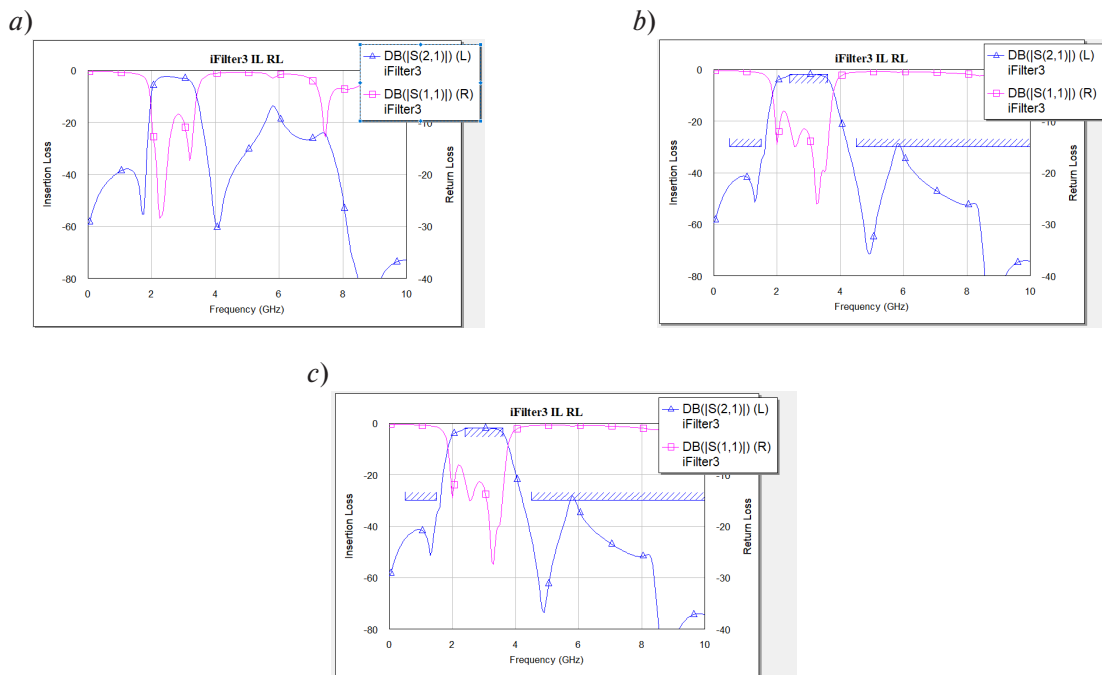


Fig. 3. S-parameters simulation results of the bandpass filter based on lumped elements with a center frequency of 3 GHz: before optimization (a), after optimization (b), after rounding the values (c)

The description of the contact pads is given in Table 2. The contact pads by inputs and outputs are grouped in a GSG formation, which means that there is one signal pad (S) and two ground pads (G), symmetrically located relative to the signal pad.



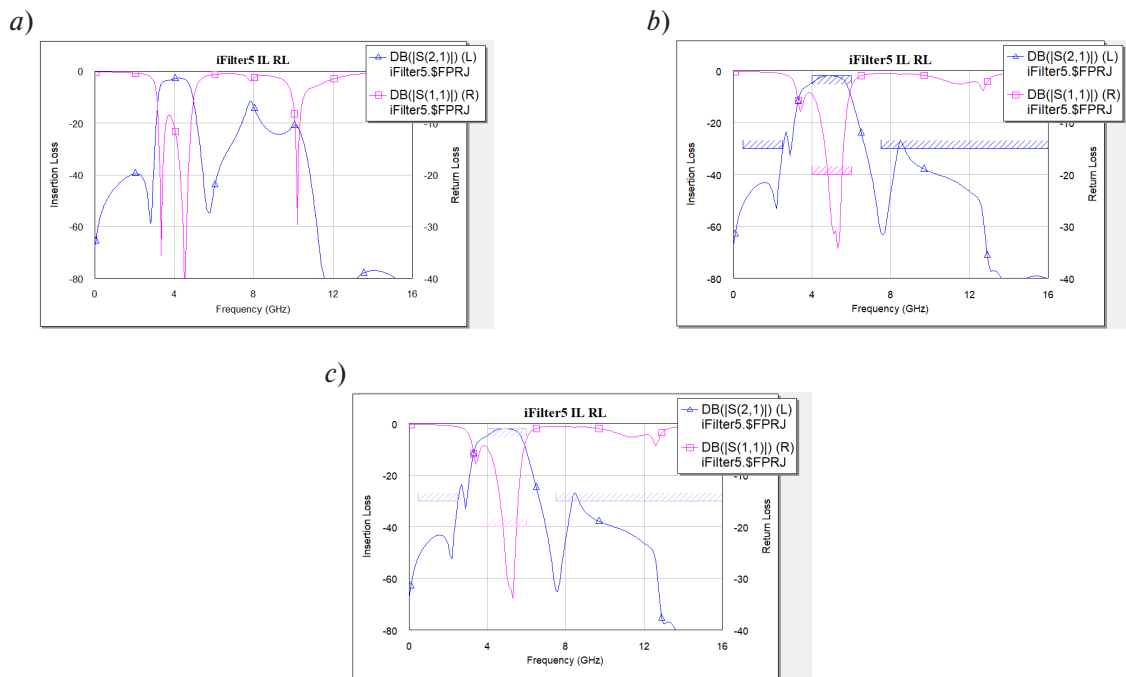


Fig. 4. S-parameters simulation results of the bandpass filter based on lumped elements with a center frequency of 5 GHz: before optimization (a), after optimization (b), after rounding the values (c)

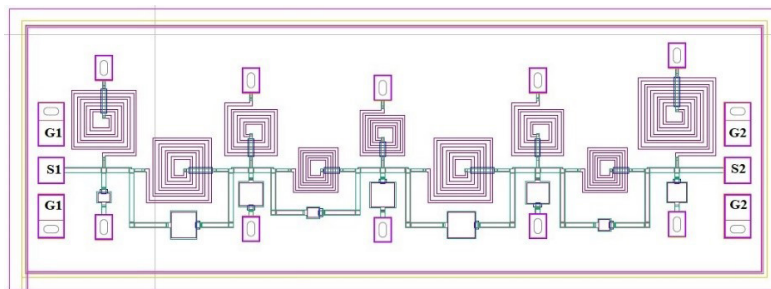


Fig. 5. Topology of the bandpass filter crystal with a center frequency of 2.5 GHz. Cell size 3100×1130  $\mu\text{m}$

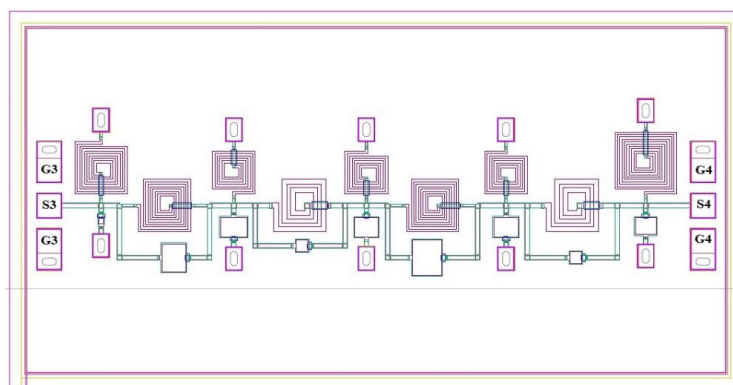


Fig. 6. Topology of the bandpass filter crystal with a center frequency of 3 GHz. Cell size 3100×1600  $\mu\text{m}$

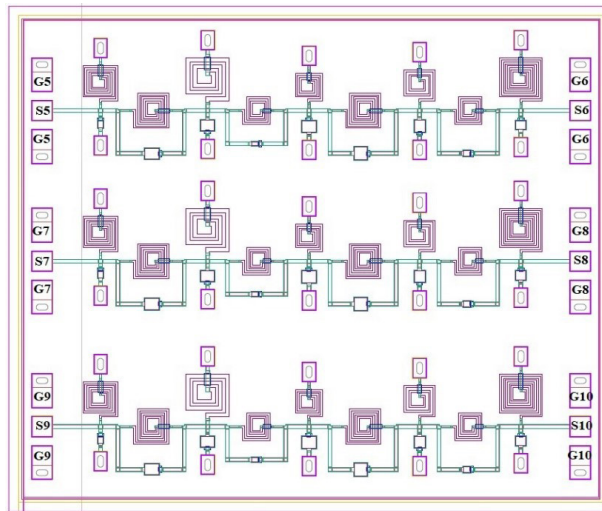


Fig. 7. Topology of the bandpass filter crystal with a center frequency of 5 GHz, repeated three times. Cell size 3100×2600 μm

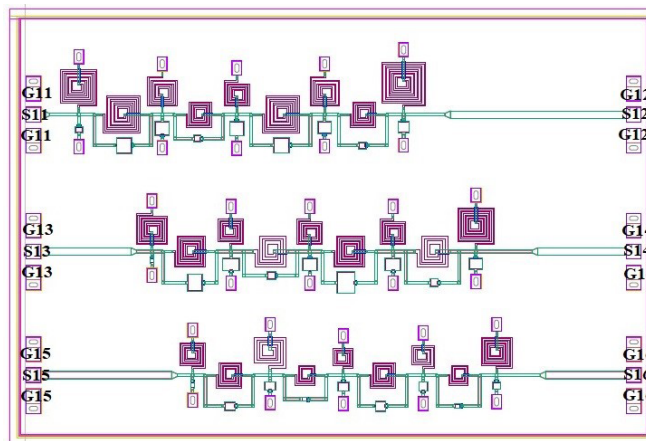


Fig. 8. Topologies of three filters (2.5, 3 and 5 GHz) in one cell. Cell size 4600×3100 μm

Table 2

**Description of contact pads of filter crystals**

| Output name | Purpose   |
|-------------|---|
| G1...G16    | Ground pads are repeated to the left and right of each signal pad.  |
| S1...S16    | Signal pads correspond to the inputs and outputs of each topology. Each filter topology corresponds to one pair of odd $S_i$ and even $S_{(i+1)}$ inputs and outputs. |

**Measurement results**

For connection to contact pads, test samples can be welded onto test PCB boards using a Kulicke & Soffa 4256 wedge micro welding machine. In this case, connection to the vector network analyzer is made via SMA connectors, which are soldered to a conductive line matched to 50 Ohm. For the samples in Fig. 7 and 8, three pairs of connectors are required, or three pairs of samples with welding of the corresponding pairs of inputs and outputs 5–6, 7–8, 9–10, 11–12, 13–14, 15–16.

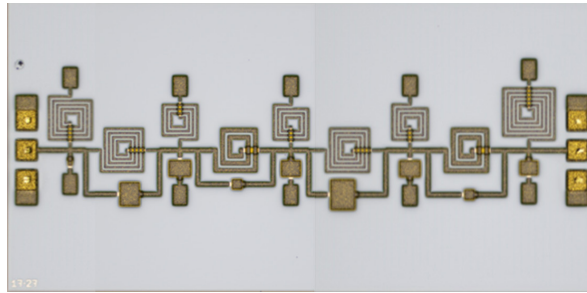


Fig. 9. Test sample microphotography (2.5 GHz center frequency)

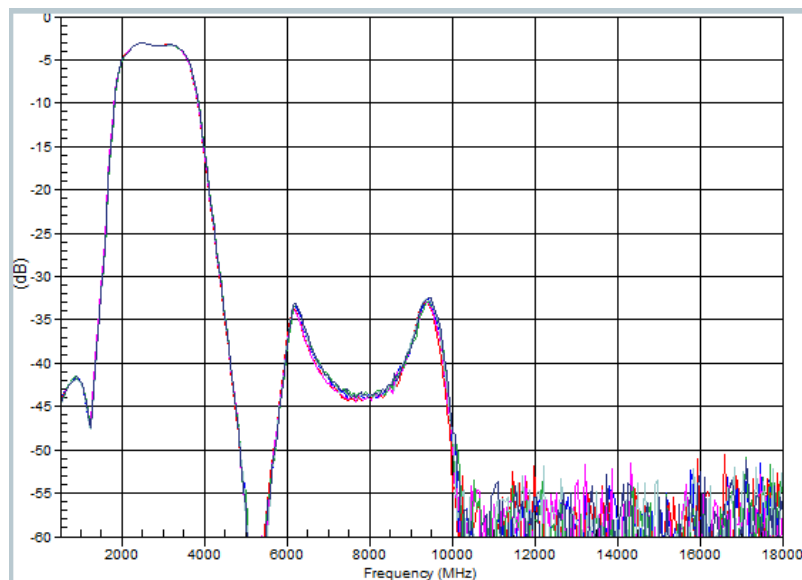


Fig. 10. S21 measurement results

Alternatively, without welding the experimental samples, a Cascade Microtech EP6RF probe station can be used. GSG-type probes (two outer contacts are “ground”, the central one is signal) are connected via coaxial wires to a Rohde Schwarz ZVA40 vector network analyzer. In the case of the topologies shown in Fig. 5 and 6, the probes are connected once. For the topologies in Fig. 7 and 8, the samples are connected one by one in the order from the top to the bottom. The connection is checked visually. The S-parameters (transmission and reflection coefficients, phase characteristics) are measured in the two-port mode of the vector analyzer. The results are recorded both on the device display and saved in the s2p file format for further processing in CAD. The measurements are performed in two modes: in a wide frequency range to evaluate the frequency characteristics as a whole and in an enlarged scale in the filter passband. Since the filters are passive and non-tunable, no power supply or control signals are required. After receiving the s2p files, all secondary characteristics (e.g., VSWR) can be obtained in CAD by converting the measured S-parameters.

A microphotograph for a test sample for 2.5 GHz frequency is shown in Fig. 9. All samples show good correlation between CAD topology simulation results and production.

The results of measurement of six microwave filter samples with a central frequency of 2.5 GHz according to the described measurement procedure are shown in Fig. 10–12. Fig. 10 shows the modulus of the parameter S21, Fig. 11 – the reflection coefficient at the input S11, Fig. 12 – the reflection coefficient at the output S22.

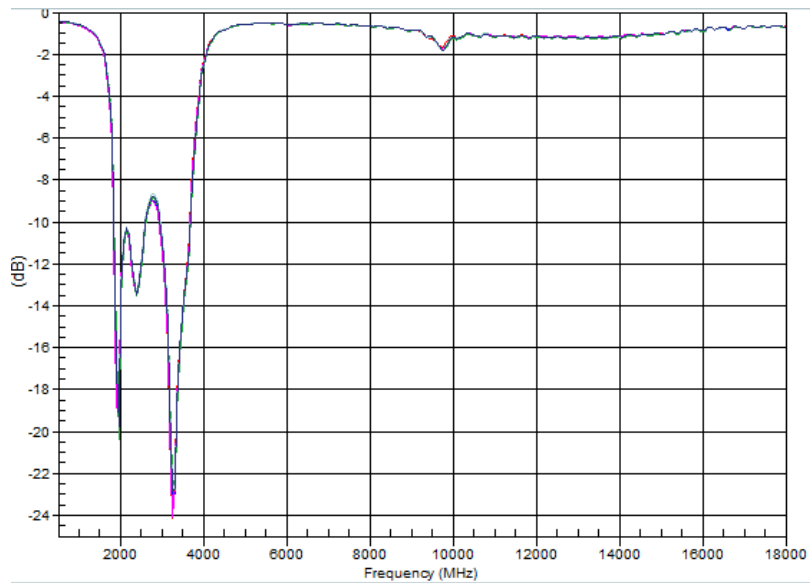


Fig. 11. S11 measurement results

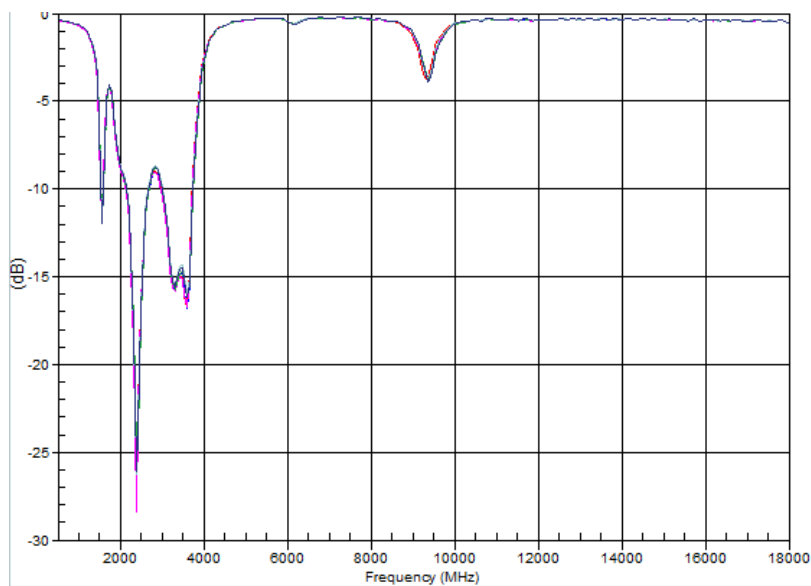


Fig. 12. S22 measurement results

The microwave filter with a central frequency of 2.5 GHz has a working frequency band of 2.2–3.4 GHz instead of the calculated one from 2 to 3 GHz, which indicates a slight shift of the central frequency by 12% due to technological deviations in the parameters of the integrated circuit elements. But at the same time, the nature of the transfer characteristic of the filter is preserved. Suppression of more than 30 dB is ensured for frequencies of 4.4 GHz and higher. Insertion losses increased by less than 2 dB compared to circuit simulation. It should be noted, that for all six samples (as for future measurements for other samples) difference in measurement results is smaller than vector analyzer sensitivity, which proves repeatability and stability of achieved result.

In Fig. 13–15 comparison of measurement versus simulation for filters with center frequencies 2.5, 3 and 5 GHz, respectively, is presented. Difference between simulation and measurement results is about

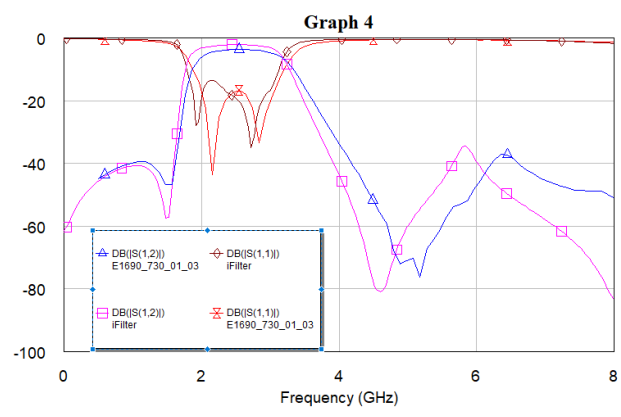


Fig. 13. Comparison of simulation (square, rhomb) and measurement (rectangle, “clock”) for 2.5 GHz center frequency

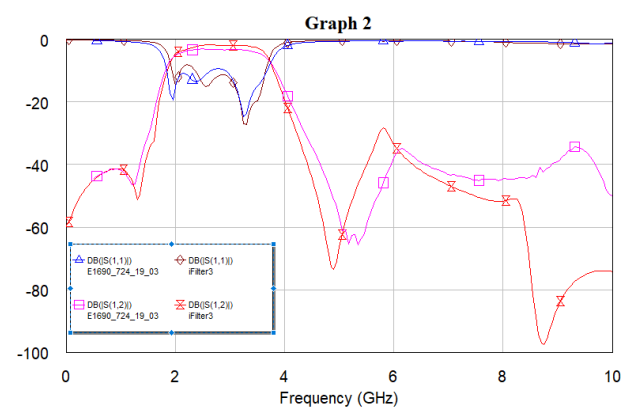


Fig. 14. Comparison of simulation (rhomb, “clock”) and measurement (rectangle, square) for 3 GHz center frequency

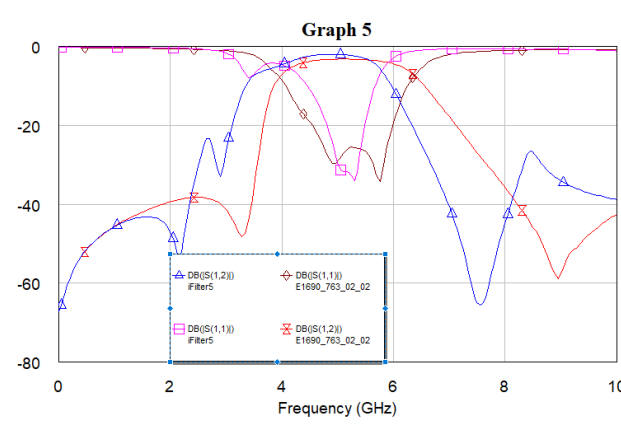


Fig. 15. Comparison of simulation (rectangle, square) and measurement (rhomb, “clock”) for 5 GHz center frequency

10% for frequencies, and in fact is closer to initial desired parameters of table 1. For 5 GHz center frequency upper stopband is 7.65 GHz from measurement with 7.5 GHz desired value. Insertion loss increases no more than by 2 dB.

Fig. 16 provides overall measurement results from different samples. It should be noted that results from stand-alone designs as in Fig. 5 and 6 correspond with the results from combined topology as in Fig. 8, which additionally proves repeatability and stability of achieved result.

In Tables 3–5, comparison of results versus target values is provided. For 5 GHz filter, best optimization result has inaccuracy in frequency response which was compensated in real sample. In Table 6, a comparison with known results is held. [14, 15] are products of Svetlana-Rost JSC (pHEMT05D), and results of this work are quite close, taking into account different fractional bandwidth (40% in this work, 23% in [14] and 24% in [15]). In [16], a bandpass filter from well-known Marki Microwave has a better insertion loss, but for 95% fractional bandwidth.

Table 3

Parameters comparison for 2.5 GHz filter

|                          | Target | Simulation | Measurements |
|--------------------------|--------|------------|--------------|
| Center frequency, GHz    | 2.5    | 2.41       | 2.55         |
| Passband, GHz            | 1      | 0.99       | 0.9          |
| Stopband, GHz            | 2.5    | 2.1        | 2.2          |
| Insertion Loss, dB       | 2      | 2.1        | 3.7          |
| Stopband attenuation, dB | 30     | 30         | 30           |

Table 4

Parameters comparison for 3 GHz filter

|                          | Target | Simulation | Measurements |
|--------------------------|--------|------------|--------------|
| Center frequency, GHz    | 3      | 2.73       | 2.81         |
| Passband, GHz            | 1.2    | 1.3        | 1.3          |
| Stopband, GHz            | 3      | 2.87       | 2.92         |
| Insertion Loss, dB       | 2      | 1.9        | 3.4          |
| Stopband attenuation, dB | 30     | 30         | 30           |

Table 5

Parameters comparison for 5 GHz filter

|                          | Target | Simulation | Measurements |
|--------------------------|--------|------------|--------------|
| Center frequency, GHz    | 5      | 4.8        | 5.1          |
| Passband, GHz            | 2      | 1.77*      | 2.1          |
| Stopband, GHz            | 5      | 6.2*       | 5.1          |
| Insertion Loss, dB       | 2      | 2          | 3.2          |
| Stopband attenuation, dB | 30     | 30         | 30           |

\* Inaccuracy of the frequency response could not be compensated with optimization

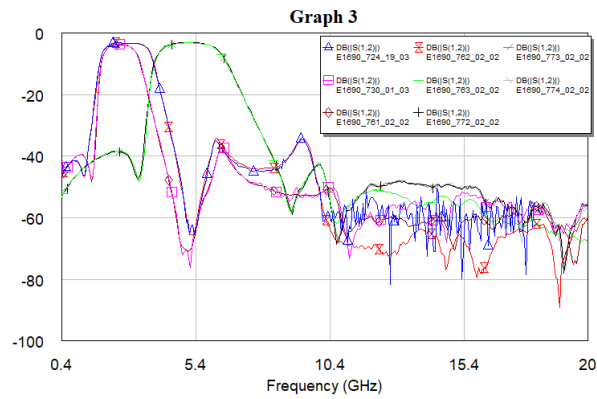


Fig. 16. Measurement results of S21 for all types of samples

Table 6

**Comparison with known solutions**

|              | Center frequency, GHz | Passband, GHz | Stopband, GHz | Insertion Loss, dB | Stopband attenuation, dB |
|--------------|-----------------------|---------------|---------------|--------------------|--------------------------|
| This article | 2.55                  | 0.9           | 2.2           | 3.7                | 30                       |
| This article | 2.81                  | 1.3           | 2.92          | 3.4                |                          |
| This article | 5.1                   | 2.1           | 5.1           | 3.2                |                          |
| [14]         | 6.6                   | 1.5           | 5.5           | 4.3                | 40                       |
| [15]         | 2.05                  | 0.5           | 2.5           | 4.3                | 60                       |
| [16]         | 3.5                   | 3.3           | 5.23          | 1.63               | 30                       |

**Conclusion**

In this paper, design of monolithic microwave integrated circuit bandpass filters for S- and C-bands was presented for pHEMT05D technology. Three filters for 2.5, 3 and 5 GHz center frequency with 40% fractional bandwidth and 30 dB stopband attenuation were considered. Lumped elements realization for Type 1 generalized Chebyshev approximation was used.

Measurement setup was described using Rohde Schwarz ZVA40 vector network analyzer and Cascade Microtech EP6RF probe station. For first six samples, frequency shift of 12% for center frequency was found, as well as increase of insertion loss by 2 dB. It should be noted that difference between samples is smaller than vector analyzer sensitivity, which proves repeatability and stability of achieved result.

For other center frequencies, difference between model and simulation is about 10% for frequencies, and in fact is closer to initial desired parameters of Fig. 1. For 5 GHz center frequency upper stopband is 7.65 GHz from measurement with 7.5 GHz desired value. Stopband attenuation of -30 dB is achieved for all samples. Insertion loss increases no more than by 2 dB in comparison with simulation. Difference between samples is also under vector analyzer sensitivity, which proves quality of the achieved result.

Achieved results are comparable to that of Svetlana-Rost JSC products and are close to results of Marki Microwave.

## REFERENCES

1. Klimenko D.V., Nikitin A.B., Stroganov A.A., Tsikin I.A. Design of monolithic microwave integrated circuits bandpass filters on GaAs pHEMT. *2023 International Conference on Electrical Engineering and Photonics (EExPolytech)*, 2023, Pp. 10–13. DOI: 10.1109/EExPolytech58658.2023.10318671
2. Taslimi A., Mouthaan K. Design of optimized minimum inductor bandpass filters. *IEEE Transactions on Microwave Theory and Techniques*, 2017, Vol. 65, No. 2, Pp. 484–495. DOI: 10.1109/TMTT.2016.2618390
3. Yang Y., Zhu X., Dutkiewicz E., Xue Q. Design of a miniaturized on-chip bandpass filter using edge-coupled resonators for millimeter-wave applications. *IEEE Transactions on Electron Devices*, 2017, Vol. 64, No. 9, Pp. 3822–3828. DOI: 10.1109/TED.2017.2720185
4. RincyJuliet M., Vallikannu R., Kavitha B.C. A novel design of hybrid microwave bandpass filter using lumped element and microstrip line technology. *2017 International Conference on Intelligent Computing and Control (I2C2)*, 2017, Pp. 1–4. DOI: 10.1109/I2C2.2017.8321876
5. Miljanović D.M., Potřebić M.M., Tošić D.V. Microwave bandpass filter with quasi-lumped elements. *2015 23<sup>rd</sup> Telecommunications Forum Telfor (TELFOR)*, 2015, Pp. 551–558. DOI: 10.1109/TELFOR.2015.7377528
6. Hepburn L., Hong J. On the development of compact lumped-element LCP filters. *2014 44<sup>th</sup> European Microwave Conference*, 2014, Pp. 544–547. DOI: 10.1109/EuMC.2014.6986491
7. Liu S., Cheong P., Choi W.-W. Synthesis of wideband bandpass filter with optimized matrix scaling for direct circuit implementation using lumped elements. *IEEE Microwave and Wireless Technology Letters*, 2025, Vol. 35, No. 1, Pp. 23–26. DOI: 10.1109/LMWT.2024.3486222
8. Shen G., Che W., Feng W., Shi Y., Shen Y. Low insertion-loss MMIC bandpass filter using lumped-distributed parameters for 5G millimeter-wave application. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2021, Vol. 11, No. 1, Pp. 98–108. DOI: 10.1109/TCPMT.2020.3039987
9. Jorgesen D., Marki C. MMIC filters’ time has come. *Microwave Journal*, 2022, Vol. 65 No. 10. Pp. 48–60.
10. Krasovitsky D.M., Dudin A.L., Katsavets N.I., Kokin S.V., Filaretov A.G., Chaly V.P. Challenges and prospects of A3B5 microwave foundry. *2012 22<sup>nd</sup> International Crimean Conference “Microwave & Telecommunication Technology” (CriMiCo’2012)*, 2012, Pp. 615–616.
11. Krasovitsky D.M., Filaretov A.G., Chaly V.P. Physics and technological aspects of microwave foundry development: “Svetlana-Rost” Experience. *Mokerovskie chteniia [Moker Readings]. 10<sup>th</sup> Anniversary International Scientific and Practical Conference on Physics and Technology of Nanoheterostructural Microwave Electronics*, 2019, Pp. 29–32.
12. Filaretov A.G., Chaly V.P., Shukov I.V., Dudin A.L., Fazyhanov O.R., Krasovitsky D.M. Setting up a highly reliable solid state microwave components production – Routines and first experience. *Nanoindustry*, 2021, Vol. 14, No. 7s (107), Pp. 408–410. DOI: 10.22184/1993-8578.2021.14.7s.408.410
13. Berezniak A.F., Korotkov A.S. Design method and manufacturing monolithic microwave integrated circuit switches on GaAs pHEMT. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems*, 2019, Vol. 12, No. 4, Pp. 84–96. DOI: 10.18721/JCSTCS.12407
14. Technical documentation for the SVA1001 MMIC. Available: [http://www.svetlana-rost.ru/content/site1/pages/filters/SVA1001\\_PPF\\_6-7GHz.pdf](http://www.svetlana-rost.ru/content/site1/pages/filters/SVA1001_PPF_6-7GHz.pdf) (Accessed: 28.02.2025).
15. Technical documentation for the SVA1002 MMIC. Available: [http://www.svetlana-rost.ru/content/site1/pages/filters/SVA1002\\_PPF\\_1,75-2,35GHz.pdf](http://www.svetlana-rost.ru/content/site1/pages/filters/SVA1002_PPF_1,75-2,35GHz.pdf) (Accessed: 28.02.2025).
16. Technical documentation for the MFBP-00112CH microcircuit. Available: <https://markimicrowave.com/assets/3895c8ca-a953-4e52-96e0-5a94f030e39b/MFBP-00112CH-Passive%20GaAs%20MMIC%202.2%20-%205.5%20GHz%20Bandpass%20Filter%20.pdf> (Accessed: 28.02.2025).



**INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ**

**Ivanov Nikita V.**

**Иванов Никита Валерьевич**

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

ORCID: <https://orcid.org/0000-0001-8222-9869>

**Rumyantsev Ivan A.**

**Румянцев Иван Александрович**

E-mail: [i.a.rumiantsev@spbstu.ru](mailto:i.a.rumiantsev@spbstu.ru)

ORCID: <https://orcid.org/0000-0002-9477-2022>

*Submitted: 26.01.2025; Approved: 04.03.2025; Accepted: 13.03.2025.*

*Поступила: 26.01.2025; Одобрена: 04.03.2025; Принята: 13.03.2025.*

# Computer Simulations of Telecommunication, and Control Systems

## Компьютерные сети, вычислительные, телекоммуникационные, управляющие и измерительные системы





Research article

DOI: <https://doi.org/10.18721/JCSTCS.18108>

UDC 681.586.2, 621.3.049.7



### A MICROMECHANICAL PRESSURE SENSOR WITH RECONFIGURABLE ASIC

*A.T. Tulaev*<sup>1</sup> , *A.S. Kozlov*<sup>1</sup>, *D.V. Kostygov*<sup>1</sup> ,  
*K.P. Kuznecov*<sup>1</sup> , *Ya.V. Belyaev*<sup>1</sup>, *V.V. Loboda*<sup>2</sup> 

<sup>1</sup> Concern CSRI Elektropribor, JSC,  
St. Petersburg, Russian Federation;

<sup>2</sup> Peter the Great St. Petersburg Polytechnic University,  
St. Petersburg, Russian Federation

✉ [artulaev@gmail.com](mailto:artulaev@gmail.com)

**Abstract.** In this paper, a MEMS pressure sensor with reconfigurable ASIC and measurement range of 10 kPa to 100 MPa was designed, based on the system model design approach. The competitive advantages of the method include rapid prototyping and controlling parameters of micromechanical pressure sensor at all design stages; thus, reducing the device development and manufacturing costs. The set of unified piezoresistive sensing elements with different full scale pressure and sensitivity was implemented on a pre-doped silicon on insulator (SOI) wafer. The optimal parameters of sensing elements and integrated circuit were obtained using complex optimization criterion by system level simulation and refined by finite element method. The reconfiguration requirements were obtained by simulation of technological process variations. The reconfigurable ASIC is implemented using 0.18  $\mu\text{m}$  SOI technology. The ASIC provides integrated solution with on-chip programmable offset trimming, temperature sensing, clock generation and digital signal processing. The system level and schematic simulations were performed during ASIC development. The digital signal processing verification was performed by FPGA prototyping. The experimental studies were carried out for sensor prototypes with 100 kPa and 1 MPa full scale range. The developed pressure sensor based on micro-assembly achieves the 0.06% main full-scale error.

**Keywords:** MEMS, ASIC, SOI, pressure sensor, system model, reconfiguration

**Citation:** Tulaev A.T., Kozlov A.S., Kostygov D.V., et al. A micromechanical pressure sensor with reconfigurable ASIC. Computing, Telecommunications and Control, 2025, Vol. 18, No. 1, Pp. 98–110. DOI: 10.18721/JCSTCS.18108





Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18108>

УДК 681.586.2, 621.3.049.7



## МЭМС ДАТЧИК ДАВЛЕНИЯ С РЕКОНФИГУРИРУЕМОЙ ИНТЕГРАЛЬНОЙ СХЕМОЙ

А.Т. Тулаев<sup>1</sup> , А.С. Козлов<sup>1</sup>, Д.В. Костыгов<sup>1</sup> ,  
К.П. Кузнецов<sup>1</sup> , Я.В. Беляев<sup>1</sup>, В.В. Лобода<sup>2</sup> 

<sup>1</sup> АО «Концерн «ЦНИИ «Электроприбор»,  
Санкт-Петербург, Российская Федерация;

<sup>2</sup> Санкт-Петербургский политехнический университет Петра Великого,  
Санкт-Петербург, Российская Федерация

✉ [artulaev@gmail.com](mailto:artulaev@gmail.com)

**Аннотация.** В данной работе с помощью методики синтеза, основанной на методе системного моделирования, был разработан МЭМС-датчик давления с реконфигурируемой интегральной схемой и верхними пределами измерений от 10 кПа до 100 МПа. Преимущества данной методики заключаются в возможностях быстрого прототипирования и контроля параметров на всех этапах разработки, что снижает себестоимость разработки и изготовления устройства. На предварительно легированной примесью р-типа КНИ-пластине был изготовлен набор тензорезистивных чувствительных элементов с разной чувствительностью к давлению. Оптимальные параметры чувствительных элементов были получены по комплексному критерию оптимизации с помощью моделирования на системном уровне и уточнены с помощью метода конечных элементов. Требования к реконфигурации интегральной схемы были получены путем моделирования технологических отклонений. Реконфигурируемая интегральная схема была выполнена по технологии КНИ с проектной нормой 180 нм. Интегральная схема представляет собой единое схемотехническое решение, позволяющее проводить подстройку рабочей точки датчика, измерение температуры, генерирование тактового сигнала для цифровой части схемы и цифровую обработку полезного сигнала. Синтез интегральной схемы проводился с помощью системного и схемотехнического моделирования. Верификация алгоритмов цифровой обработки сигналов проводилась с помощью прототипирования на ПЛИС. Для образцов с верхними пределами измерений 100 кПа и 1 МПа были проведены экспериментальные исследования, показавшие, что разработанный датчик давления на основе микросборки позволяет достичь основной ошибки не более 0,06%.

**Ключевые слова:** МЭМС, ИС, КНИ, датчик давления, системная модель, реконфигурируемость

**Для цитирования:** Tulaev A.T., Kozlov A.S., Kostygov D.V., et al. A micromechanical pressure sensor with reconfigurable ASIC // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 98–110. DOI: 10.18721/JCSTCS.18108

### Introduction

According to analytical company Yole Development, MEMS pressure sensors occupy more than a third of the current MEMS market and will amount to 2.2 billion US dollars by 2026 [1]. The micromechanical pressure sensors (MMPS) are mainly used in the automotive, aircraft, shipbuilding, medicine, energy, and mining industries. To increase the overall technological capacity of local manufacturing, it is critical to replace the existing pressure sensors in industrial control and management systems in order to improve their efficiency. The competitive advantages of MMPS are low production costs due to serial production technology, small size, low energy consumption, and high measurement accuracy. A standard MEMS pressure sensor consists of a sensing element (SE) and an electronic processing circuit, i.e. an application-specific integrated circuit (ASIC) [2]. The main characteristics of MEMS

pressure sensors are full scale (FS) measurement range, measurement accuracy of the FS range, operating temperatures range, bandwidth and power consumption. In general, the SE is based on a pressure difference-sensitive diaphragm. Due to a simple manufacturing, the majority of research of the MMPS consider the pressure sensors, based on the piezoresistive principle, using one or several Wheatstone bridges made of pressure-sensitive resistors located on the SE. As a basis for the fabrication of MEMS structures, a P-doped silicon wafer is used.

#### *Synthesis technique based on the system model*

The MMPS design starts with analyzing its main characteristics and components. The major design difficulties of the modern integrated circuits are the extreme complexity of the projects, infeasibility of getting intermediate results of prototyping, and the high manufacturing and designing costs. Integrated circuits for micromechanical sensors include additional criteria related to the provision of accurate operation with SE that are based on different physical principles.

Unlike the traditional technological processes in microelectronics, where technological and statistical process variations ranges are defined, standardized and described in a process design kit (PDK), in micromechanics the existing variety of designs and technologies determines the challenges of taking into account the different technological and statistical process variations when synthesizing the SE.

In microelectronics, the design methodology referred to as “top-down” design is applied to solve the issues described above.

However, when designing the micromechanical sensors, the following problems arise due to the heterogeneous character of the main components:

- lack of the design phase of upper-level model;
- lack of early co-simulating of components;
- lack of early prototyping.

To define the accurate interaction between heterogeneous components, it is feasible to carry out design using system level simulation. The MEMS system model considers functional, structural, and behavioral aspects of a microelectromechanical system, which includes heterogeneous physical subsystems and allows their fast and efficient simulation, preferably in a single simulation environment [3]. To solve the task of synthesizing the MMPS with reconfigurable [4, 5] ASIC, this research considers the sensor system model synthesis method with stepwise optimization of the sensor components based on the complex criterion. Such criterion was developed by applying the SWaP-C [6, 7] criterion to a MMPS and extending it to consider heterogeneous design and technological constraints and process variation of the component parameters.

#### *Architecture*

This research focuses on the architecture based on the direct conversion circuit. To design the MMPS, optimal SE structures with different sensitivity of pressure conversion and a reconfigurable ASIC can be used. Fig. 1 shows the MMPS system architecture.

The system consists of SE with the optimal characteristics for the required upper pressure measurement limit (UPML) and the reconfigurable ASIC. The SE is chosen during the sensor microassembly, depending on sensor requirements. The reconfigurable ASIC consists of analog front-end and digital parts. The analog front-end is controlled by digital part and can be reprogrammed. The optimal configuration can be stored and initialized from external non-volatile memory (FLASH).

The SE consists of a piezoresistive Wheatstone bridge formed in P-doped device layer of a SE silicon diaphragm. The ASIC consists of a low-noise programmed analog readout interface based on the instrumentation amplifier with chopping technique, a digital-to-analog converter (DAC) for adjusting the offset after the first gain stage, a switch capacitor low pass filter, a multi-stage noise shaping (MASH)- $\Sigma\Delta$  analog-to-digital converter (ADC), a temperature measurement channel with the option of working with external or internal sensor, a digital filtration and processing, and SPI-interface. The supply voltage of ASIC and the SE piezoresistive bridge is 1.8 V.

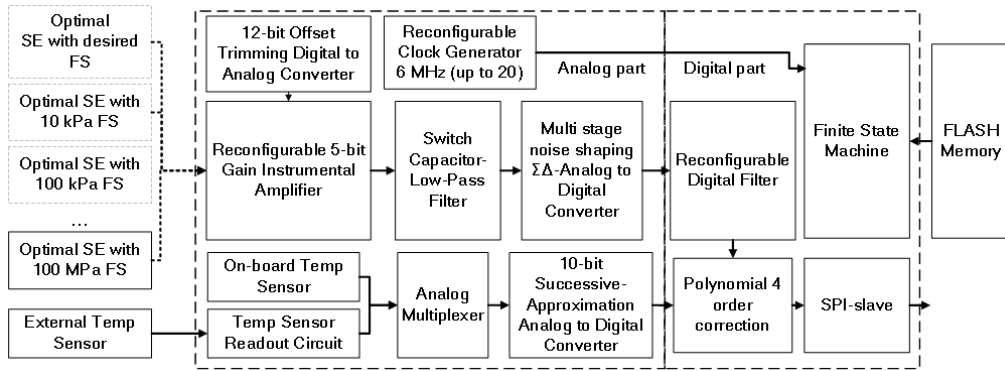


Fig. 1. MEMS pressure sensor system architecture

### System model

It is possible to solve the task of the MMPS optimal synthesis with minimal design and manufacturing costs by using system level approach. To synthesize the system model, the analysis of the detection physical principles was carried out. The main formulae describing the physical behavior of a sensing element are as follows [8]:

$$V_{out} = V_A - V_B = \left( \frac{R_4}{R_3 + R_4} - \frac{R}{R_1 + R_2} \right) V_S; \quad (1)$$

$$\frac{\Delta R}{R} \approx \frac{\Delta \rho}{\rho}; \quad (2)$$

$$(\Delta \rho / \rho)_l = \pi_l \sigma_l + \pi_t \sigma_t; \quad (3)$$

$$(\Delta \rho / \rho)_t = \pi_l \sigma_l + \pi_t \sigma_t; \quad (4)$$

$$\pi_l \approx -\pi_t \approx \frac{\pi_{44}}{2}; \quad (5)$$

$$\sigma_x = f(P, a, b, th, x, y), \quad (6)$$

where  $V_{out}$  is the output voltage;  $V_A$  and  $V_B$  are Wheatstone bridge nodes voltages;  $R_1 - R_4$  are the bridge piezoresistances;  $V_S$  is the bridge supply voltage;  $\rho$  is the resistivity;  $\Delta \rho$  is the resistivity change;  $\pi_l$  and  $\pi_t$  are the longitudinal and transverse gauge coefficients/factors;  $\pi_{44}$  is the piezoresistivity coefficient;  $\sigma$  is the directional mechanical stress as a function of load pressure ( $P$ ), membrane width ( $a$ ,  $b$ ), thickness ( $th$ ) and resistors coordinates ( $x$ ,  $y$ ).

Based on the provided description, the system model of a SE was defined. To define the optimal parameters for the SE variety in the range from 10 kPa to 100 MPa, a simulation with various geometrical parameters was carried out. As particular optimization criteria, the nonlinearity of the pressure conversion into the output voltage (no more than 0.05–0.5%), the FS normalized pressure sensitivity (no less than 10 mV/V for the full measurement range), the maximum membrane deflection (no more than 20% from the membrane thickness) and the structure integrity (the maximal mechanical stress of no more than 10% from the ultimate tensile strength) were chosen. As a result of the simulation, the

optimal set of SE structures with the unified geometrical parameters was identified, and the technical requirements for the ASIC model were defined. As the initial constraints were the maximum and minimum thickness and width of the membrane, which ranged from 30 to 450  $\mu\text{m}$  and from 200 to 2000  $\mu\text{m}$ .

The MMPS system model was modified by adding an ASIC model. The system model of the integrated circuit consists of the analog and digital parts.

The analog front-end of the integrated circuit includes the following main processing units:

- input readout interface circuit;
- analog low pass filter;
- ADC.

The input readout interface circuit and the analog low pass filter can be represented as a set of blocks with the corresponding transfer functions, the blocks simulating their nonlinear and noise parameters, and the initial SE offset correction blocks. The dynamic parameters – SFDR and SNR – were selected as the main parameters of the processing scheme, while SNR parameter was simulated, taking into account both white and flicker noise ( $1/f$ ). The earlier reported [9] MASH- $\Sigma\Delta$ -ADC was used as ADC model. The MASH 2-2 architecture provides characteristics of higher order modulators and preserving stability. The determination of the optimal requirements to the analog electronic blocks was carried out by solving the optimization task. The simulating results show that to obtain 0.05% main FS error, it is necessary to have an analog front-end with SFDR parameters of more than 61 dB and SNR of more than 70.4 dB [10]. As a result of the simulation, the technical requirements for the digital signal processing (DSP) were defined.

The standard minimal DSP requirements for micromechanical sensors include the calibration coefficients to compensate the sensor offset and the gain error that are generally realized by means of the temperature-dependent polynomial [11]. Besides temperature compensation, the main requirement for the MMPS is the option to control sampling frequency and the bandwidth that is defined by the output digital filters. In practice, to filter  $\Sigma\Delta$ -sequences, the recursive and non-recursive cascaded integrator–comb (CIC) filters are applied. The main advantage of the given filter type is the lack of multiplications in its structure. Non-recursive form used [12] in case of low decimation coefficient (up to 32 times), low bit wordlength of the input sequence (1–2 bits) and high operating frequencies ( $>1$  MHz), that is not suitable for this research. For this reason, the structure with applied cascade recursive CIC with a reconfigurable halfband filter and a reconfigurable FIR compensator was selected. As a result of the system simulation, the optimal parameters, such as the filters orders, the number of coefficients and their bit wordlength, were defined. As a result of verifying at the system level, the required value of the main FS error was confirmed.

## Application

### *A sensing element*

The Concern CSRI Elektropribor, JSC, was chosen as the semiconductor fabrication for MEMS manufacturing. This semiconductor fabrication plant has limitations on silicon ion-doping operations. The manufacture of the SE without this operation is described in [13]. The SE is manufactured on a pre-doped p-type silicon-on-insulator (SOI) wafer with (100) plane orientation. In this research a standard SOI wafer with conductivity of 0.015  $\text{Ohm}\cdot\text{cm}$ , that equals the carrier's concentration of  $\sim 5 \cdot 10^{18}$   $1/\text{cm}^3$ , is used. Structurally, the SE consists of three layers: a structural layer, a dielectric layer and a bulk. The structural layer thickness of 500 nm was selected as an initial having the minimal surface roughness because of the Smart Cut technology [14]. The structural layer consists of four piezoresistors connected in a Wheatstone bridge configuration with a crystallographic direction [1 1 0] to provide the maximal sensitivity to the mechanical stresses. The piezoresistors and their interconnection are made in a structural layer by means of photolithography and further deep reactive-ion etching (DRIE) from the upper side to the dielectric layer. The dielectric layer consists of 1  $\mu\text{m}$  of  $\text{SiO}_2$  that electrically insulated the structural

layer from the bulk. The bulk thickness is 450  $\mu\text{m}$ . The piezoresistive bridge is placed on the membrane formation.

The structure's synthesis of MMPS SE was carried out by application of the finite element methods. This research employed the COMSOL Multiphysics package to solve the simulation tasks as this software's competitive advantage is micromechanical system simulation option [15, 16]. Integrated COMSOL library materials were used, in particular, the model of silicon with p-type doping and anisotropic elasticity [17], and the model of silicon oxide. The orientation of silicon crystallographic direction was set as [1 1 0] in accordance with the system model. The search of the structure geometric parameters was carried out by taking into the account the limitations identified at the system simulation phase. The model calculation was carried out in a combined multiphysics by applying the *Piezoresistivity*, *Domain Currents* module that includes *Solid Mechanics* module and *Electric Currents* module.

According to the optimization results of the main and additional geometric parameters, the following set of the final structures with their output parameters was obtained (Table 1). The optimal synthesis procedure is not the main topic of this article and will be discussed in future articles.

Table 1

Main parameters of SE

| UPML  | 10 kPa | 100 kPa | 1 MPa | 10 MPa | 100 MPa                             |
|---|--------|---------|-------|--------|-------------------------------------|
| Half membrane width, $\mu\text{m}$                      | 1800   | 800     | 1800  | 800    | 300                                 |
| Membrane thickness, $\mu\text{m}$                       | 30     | 30      | 250   | 250    | 250                                 |
| Sensitivity, normalized to the full range ( $S$ ), mV/V | -13.2  | -22.8   | -16.9 | -28.3  | 27.1<br>(Pressure act on the clamp) |
| Nonlinearity to the full range $NL_{FS}$ , %            | 0.35   | 0.08    | 0.1   | 0.06   | 0.5                                 |

The sensitivity test to the technological process variation ranges and additional geometric parameters for MMPS SE in the UPML range from 10 kPa to 10 MPa was carried out. According to the calculation results, the maximal sensitivity to process variation happens in the values of membrane thickness and piezoresistors thickness. Thus, the greatest impact occurs on the structures with a small membrane thickness as the variation range of the membrane etching width can reach in this case up to 20%.

#### ***Integrated circuit***

Integrated circuit design for MMPS was performed using the PDK of 180 nm SOI technology carried out by the local silicon chips manufacturer JSC "Mikron" with a unipolar voltage of 1.8 V.

The folded-cascode operational transconductance amplifiers (OTA) were used as core amplifiers for switched capacitor integrators in  $\Sigma\Delta$ -modulators. This OTA was selected as it can provide high direct current (DC) gain ( $\sim 80$  dB), wide input signal range and does not require the frequency compensation. The integrator gain is set by capacitors ratio in accordance with the system model of  $\Sigma\Delta$ -modulator. The nominal value of 1 pF was used as the optimum one in terms of the ratio of kTC noise and the crystal area.

The schematic design of low pass filter was chosen based on the first order non-inverting switching capacitance low pass filter [18] using the folded-cascode amplifier (120 dB DC gain) with a class AB output stage as rail-to-rail output provides the minimization of nonlinear distortions.

Fig. 3 illustrates the schematic of the instrumental amplifier [19, 20]. The architecture of the instrumental amplifier assumes a zero offset correction of the SE by 12-bit fully matched resistive array DAC, combined with chopping technique of useful signal from SE and offset correction signal. For the minimization of nonlinear distortions, the single-ended version of the low pass filter core operational amplifier was used. To ensure operation in the optimal amplitude range, taking into account different

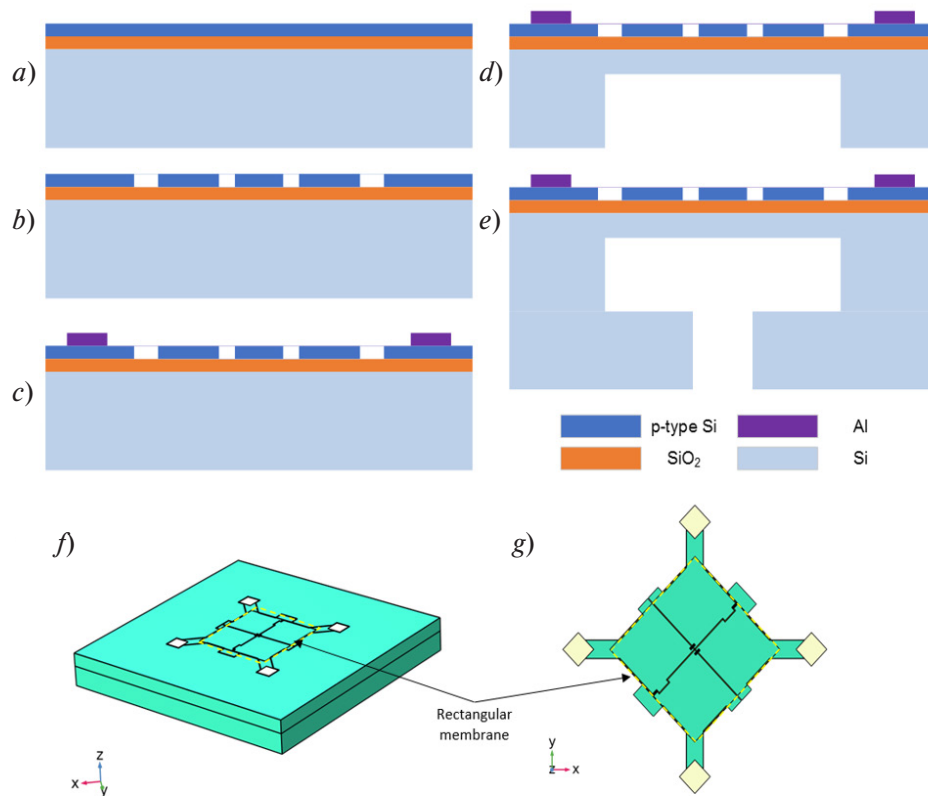


Fig. 2. SE manufacturing: *a)* SOI wafer, *b)* top DRPE for resistors and interconnections, *c)* Al deposition, *d)* bot DRPE for membrane cavity, *e)* connection to SE substrate, *f)* 3D-model of SE, *g)* Wheatstone bridge structure

SE sensitivities and technological process variations, a 5-bit reconfigurable gain from 1 to 32 times was implemented via resistive matched array.

The chopping frequency was selected as  $\frac{1}{4}$  sampling frequency in accordance with the system model. The residual signal filtration at the modulation frequency is carried out by means of non-inverting switching capacitance low pass filter and a digital CIC filter.

Simulation results show that the developed schematic model of the analog front-end with MMPS SE achieves following characteristics: SFDR ranging from 71 dB to 84 dB for operating voltage range from 300 mV to 450 mV that satisfies the developed system model requirements.

The analog part also includes the reconfigurable ring-oscillator for the clock signal generation and the temperature sensor readout circuit, consisting of 10-bit SAR-ADC, with the option of working with external or internal sensor.

The digital part was synthesized based on the developed system description. This digital part includes the cascade connection of CIC filter of order 3-3-6 with the decimation coefficient for each stage equal to 4 and application of CIC register pruning technique. The digital part also includes polynomial temperature compensation of 4<sup>th</sup> order. The digital part was prototyped in FPGA and tested by applying a  $\Sigma\Delta$ -bitstream input data derived from system model simulation and SPICE simulation. The prototyping stage allows the verification of the digital part algorithms and client software backend.

## Results and discussion

### *Sensing element*

MMPS were assembled and manufactured at the production facility of The Concern CSRI Elektropribor, JSC. Each MMPS was optimally configured for maximizing its performance.



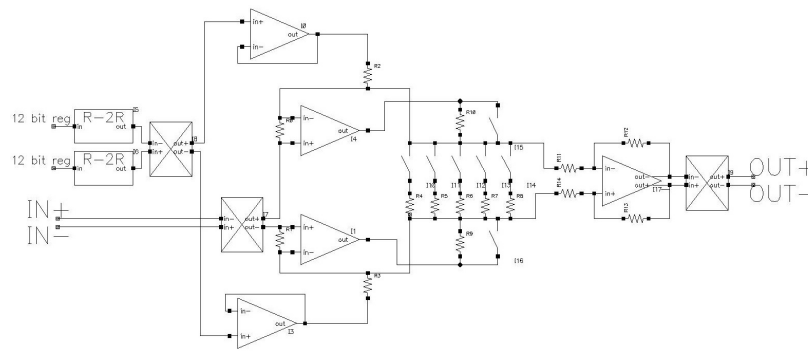


Fig. 3. Schematic of instrumental amplifier

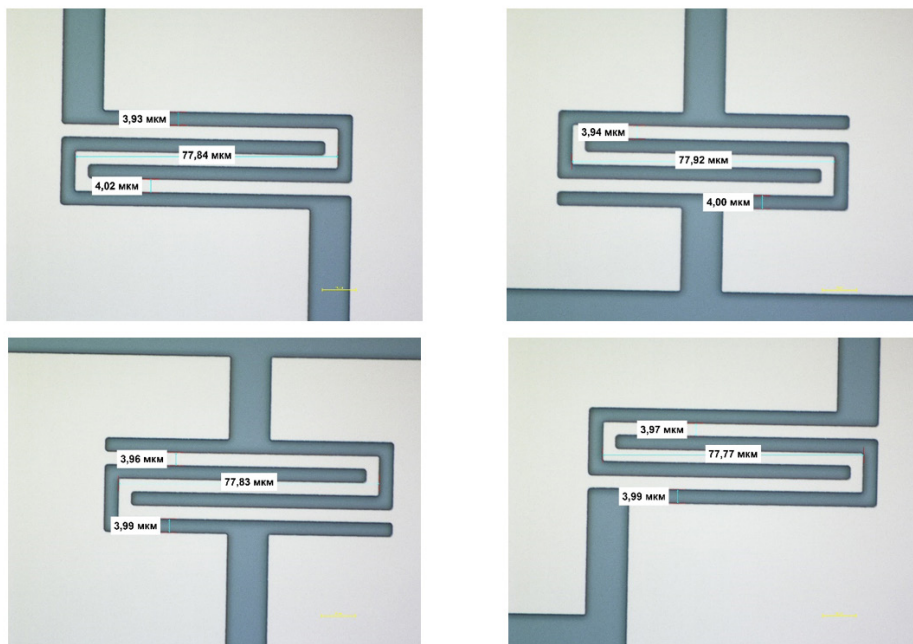


Fig. 4. Optical measurements of geometrical parameters of SE piezoresistors

The measurement of MMPS SE characteristics was carried out during several stages. Testing was done by the group method on the silicon wafer with SE. Five sampled MMPS SE on the wafer were subjected to the non-destructive optical control (Fig. 4) over the main geometrical parameters of piezoresistors. As a result of the optical control, the parameters variations of the piezoresistors did not exceed 2%.

The next stage consisted of measuring the resistances of the parts of piezobridges by applying the group method using SUMMIT 1200B-AP probe station, Agilent 34401A digital multimeter and B2201A switch matrix. The measurement results showed that the piezoresistance sections correspond with the finite element model with nominal values ranges from 14 to 17 kOhm, depending on design.

**Integrated circuit**

The measurements of different integrated circuit (Fig. 5) characteristics were carried out under normal climate conditions. To test the dynamic parameters the Stanford Research Systems DS360 signal generator was used in the mode of differential harmonic signal with the constant value 900 mV.

To acquire data from ASIC by means of SPI interface the National Instruments USB-8452 circuit board was used. Three samples were selected for testing. Fig. 6 shows the dynamic parameters dependence

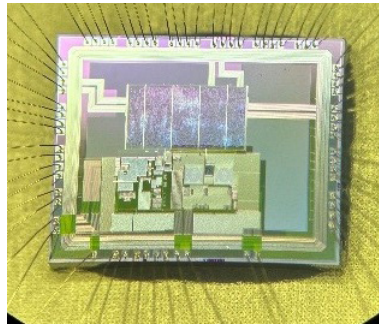


Fig. 5. Developed ASIC packaging microphotograph [21]

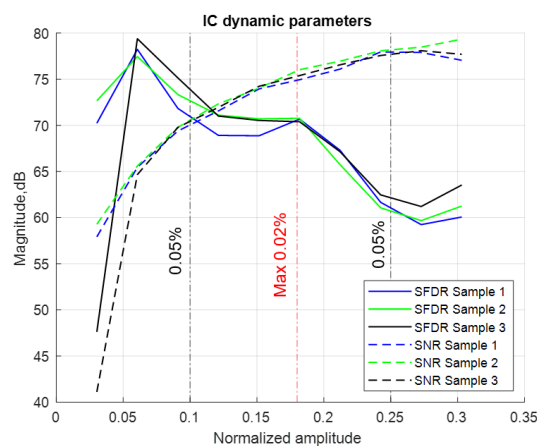


Fig. 6. Dynamic parameters of reconfigurable ASIC

on the normalized amplitude of the input signal in the range from 0.033 to 0.33. The figure illustrates that after the fabrication the SFDR parameter with the optimal amplitude of 360 mV or 1/5 the source voltage is lower than the expected by ~16 dB, when the SFDR parameter reached its peak with the low normalized amplitude of 0.07 with the maximal value of 79 dB. The area of the optimal parameters with nonlinearity is achieved with the normalized amplitudes from 0.05 to 0.1, while the optimal noise parameters are achieved within the normalized amplitudes from 0.1 to 0.3. Based on test results and additional schematic analysis in this research, it is supposed that the reason behind the conversion linearity decrease is the charge leakage effect in switched-capacitor circuits [22]. The integrated circuit provides precise measurement of almost 0.05% accuracy in the optimal range of the output amplitudes [10].

#### ***Micromechanical pressure sensor***

To carry out testing of three samples, the MMPS were assembled by wire bonding MMPS SE and MMPS integrated circuit from each other onto a special circuit board. Taking into account MMPS SE and MMPS integrated circuit parameters, the optimal settings of MMPS integrated circuit amplification coefficient for each type of MMPS SE were selected based on the criterion of minimizing the possible error. To carry out MMPS testing, the measurement system presented in Fig. 7 was applied. The measurement unit consists of the following:

- 1) ELMETRO-Paskal-03/04 pressure calibrator with modifications made by 160K (160 kPa) convertor and 1M (1MPa) convertor was used for controlling and setting the pressure.
- 2) B5-71/1-pro power source was used to power the hardware kit and the control circuit board.

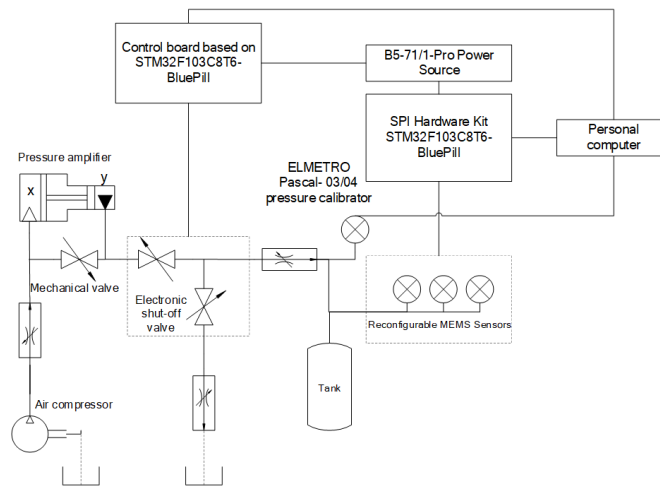


Fig. 7. The structural diagram of the measuring unit

To automate the samples testing, the hardware kit based on STM32F103C8T6-BluePill microcontroller was used for multichannel digital reading from digital SPI interface of the MMPS as well as automated setting the pressure by the relay control in accordance with the reference pressure gauge.

To define the pressure measurement precision under the normal climatic conditions the pressure cycle with 22 load points from 0 to 105% with 5% step from UPML was used. For the experimental studies samples with UPML from the most common and practically applied ranges (100 kPa, 1 MPa) were chosen. The measurement precision of MMPS was calculated using the following equation:

$$\Delta Error = \Delta NL + \Delta Noise, \tag{7}$$

where  $\Delta Error$  is the pressure measurement inaccuracy;  $\Delta NL$  is the measurement inaccuracy caused by the pressure conversion nonlinearity calculated by the least square method;  $\Delta Noise$  is the pressure measurement inaccuracy caused by sensor noise.

The measurement results of MMPS characteristics are presented in Table 2.

The overall pressure measurement error for two samples of MMPS with the UPML of 1 MPa makes up no more than 0.09%, which correlates with the theoretical results obtained earlier. Whilst the testing results for sample No. 3 demonstrated a higher nonlinearity caused by the significant hysteresis of the output value. This hysteresis value can be explained by the presence of residual mechanical stress in the structure as well as the influence of adhesive connection of the SE with a stainless-steel frame. The overall pressure measurement error for three samples of MMPS with the UPML of 100 kPa makes up less than 0.075% that corresponds to the theoretical results that were obtained earlier.

Table 2

**Measured characteristics**

| Sample No. | $\Delta Error, \% \text{ from FS}$ |       |
|------------|------------------------------------|-------|
|            | 100 kPa                            | 1 MPa |
| FS         |                                    |       |
| 1          | 0.075                              | 0.09  |
| 2          | 0.064                              | 0.09  |
| 3          | 0.06                               | 0.34  |

## Conclusion

This article deals with the development piezoresistive MMPS with reconfigurable integrated circuit. The MMPS was synthesized using system level design method. The competitive advantages of the method consist in rapid prototyping and controlling parameters of MMPS at all stages; thus, reducing the device development and manufacturing costs.

The set of unified piezoresistive SE with different FS pressure and sensitivity was implemented on a pre-doped SOI wafer. The optimal parameters of sensing elements and integrated circuit were obtained using complex optimization criterion by system level simulation and refined by finite element and schematic simulations. The reconfiguration requirements were obtained by simulation of technological variations.

The reconfigurable ASIC is implemented in 0.18  $\mu\text{m}$  SOI technology. The ASIC provides integrated solution with on-chip programmable offset trimming, temperature sensing, clock generation and DSP. The ASIC was developed using system level design method with schematic simulations. The DSP verification was performed by FPGA prototyping. For sensors prototypes with 100 kPa and 1 MPa FS range experimental studies were carried out.

The developed pressure sensor based on micro-assembly achieves the 0.06% main FS error.

## REFERENCES

1. Tulaev A.T., Kozlov A.S., Belyaev J.V., Loboda V.V., Bellavin M.A., Korotkov A.S. MEMS pressure sensors design, simulation, manufacturing, interface circuits: A review. *IEEE Sensors Journal*, 2024, Vol. 24, No. 6, Pp. 7395–7405. DOI: 10.1109/JSEN.2024.3358951
2. Nam K., Kim H., Kwon Y., Choi G., Kim T., Kim C., Cho D., Lee J., Ko H. A four-channel low-noise readout IC for air flow measurement using hot wire anemometer in 0.18  $\mu\text{m}$  CMOS technology. *Sensors*, 2021, Vol. 21, No. 14, Art. no. 4694. DOI: 10.3390/s21144694
3. Bechtold T., Schrag G., Feng L.-H. *System-level Modeling of MEMS*. Weinheim: Wiley-VCH, 2013.
4. You D., Kim H., Kim J., Han K., Heo H., Kwon Y., Kim G., Sul W.S., Lee J.W., Lee B.J., Ko H. Low-noise multimodal reconfigurable sensor readout circuit for voltage/current/resistive/capacitive microsensors. *Applied Sciences*, 2020, Vol. 10, No. 1, Art. no. 348. DOI: 10.3390/app10010348
5. Park K., Kim S.M., Eom W.-J., Kim J.-J. A reconfigurable readout integrated circuit for heterogeneous display-based multi-sensor systems. *Sensors*, 2017, Vol. 17, No. 4, Art. no. 759. DOI: 10.3390/s17040759
6. Peshekhonov V.G. The outlook for gyroscopy. *Gyroscopy and Navigation*, 2020, Vol. 11, Pp. 193–197. DOI: 10.1134/S2075108720030062
7. Alshahry S.M., Alshehry A.H., Alhazmi A.K., Chodavarapu V.P. A size, weight, power, and cost-efficient 32-channel time to digital converter using a novel wave union method. *Sensors*, 2023, Vol. 23, No. 14, Art. no. 6621. DOI: 10.3390/s23146621
8. Zhang J., Chen J., Li M., Ge Y., Wang T., Shan P., Mao X. Design, fabrication, and implementation of an array-type MEMS piezoresistive intelligent pressure sensor system. *Micromachines*, 2018, Vol. 9, No. 3, Art. no. 104. DOI: 10.3390/mi9030104
9. Kozlov A.S., Tulaev A.T., Kostygov D.V., Belyaev Y.V. Design of cascaded sigma-delta analog-to-digital converter. *Navigatsiia i upravlenie dvizheniem [Navigation and traffic control]*, 2022, Pp. 68–70.
10. Tulaev A., Loboda V., Belyaev Y. System level IC analog processing design for piezoresistive MEMS pressure sensor. *2024 International Conference on Electrical Engineering and Photonics (EExPolytech)*, 2024, Pp. 414–417. DOI: 10.1109/EExPolytech62224.2024.10755734
11. Tran A.V., Zhang X., Zhu B. Effects of temperature and residual stresses on the output characteristics of a piezoresistive pressure sensor. *IEEE Access*, 2019, Vol. 7, Pp. 27668–27676. DOI: 10.1109/ACCESS.2019.2901846

12. **Abbas M., Gustafsson O., Wanhammar L.** Power estimation of recursive and non-recursive CIC filters implemented in deep-submicron technology. *The 2010 International Conference on Green Circuits and Systems*, 2010, Pp. 221–225. DOI: 10.1109/ICGCS.2010.5543063
13. **Meng Q., Lu Y., Wang J., Chen D., Chen J.** A piezoresistive pressure sensor with optimized positions and thickness of piezoresistors. *Micromachines*, 2021, Vol. 12, No. 9, Art. no. 1095. DOI: 10.3390/mi12091095
14. **Bruel M.** Process for the production of thin semiconductor material films. United States patent, 1992, No. US 5374564 A.
15. **Tran A.V., Zhang X., Zhu B.** The development of a new piezoresistive pressure sensor for low pressures. *IEEE Transactions on Industrial Electronics*, 2017, Vol. 65, No. 8, Pp. 6487–6496. DOI: 10.1109/TIE.2017.2784341
16. **Li C., Cordovilla F., Jagdheesh R., Ocaña J.L.** Design optimization and fabrication of a novel structural SOI piezoresistive pressure sensor with high accuracy. *Sensors*, 2018, Vol. 18, No. 2, Art. no. 439. DOI: 10.3390/s18020439
17. Comsol MEMS Material Database, Available: [https://doc.comsol.com/5.5/doc/com.comsol.help.comsol/comsol\\_ref\\_materials.16.57.html](https://doc.comsol.com/5.5/doc/com.comsol.help.comsol/comsol_ref_materials.16.57.html) (Accessed: 31.01.2025).
18. **Korotkov A.S.** *Switched-capacitor filter designs: tutorial*. St. Petersburg: Polytechnic University Publishing House, 2014. 191 p. DOI: 10.18720/SPBPU/2/si21-236
19. **Kwon Y., Kim H., Kim J., Han K., You D., Heo H., “Dan” Cho D., Ko H.** Fully differential chopper-stabilized multipath current-feedback instrumentation amplifier with R-2R DAC offset adjustment for resistive bridge sensors. *Applied Sciences*, 2019, Vol. 10, No. 1, Art. no. 63. DOI: 10.3390/app10010063
20. **Pham X.T., Nguyen N.T., Nguyen V.T., Lee J.-W.** A 0.6- $\mu$ W chopper amplifier using a noise-efficient DC servo loop and squeezed-inverter stage for power-efficient biopotential sensing. *Sensors*, 2020, Vol. 20, No. 7, Art. no. 2059. DOI: 10.3390/s20072059
21. **Tulaev A.T., Kozlov A.S., Kostygov D.V., Kuznetsov K.P., Beliaev Ia.V.** *Topologiia integral'noi mikroskhemiy mikromekhanicheskogo datchika davleniia [Micromechanical pressure sensor integrated circuit topology]*, 2024, RU 2024630263.
22. **Zhong L., Xu D., Lai X., Wang Y., Liao X., Fang Z.** Precision improvement of power-efficient capacitive sensor readout circuit using multi-nested clocks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2020, Vol. 67, No. 8, Pp. 2578–2587. DOI: 10.1109/TCSI.2020.2979316

#### INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

**Tulaev Artyom T.**  
Тулаев Артём Толибович  
E-mail: artulaev@gmail.com

**Kozlov Alexey S.**  
Козлов Алексей Сергеевич  
E-mail: kas573@yandex.ru

**Kostygov Dmitrii V.**  
Костыгов Дмитрий Вадимович  
E-mail: dkost92@mail.ru  
ORCID: <https://orcid.org/0000-0003-4379-5803>

**Kuznetsov Kirill P.**  
Кузнецов Кирилл Петрович  
E-mail: pirropod1@gmail.com  
ORCID: <https://orcid.org/0009-0003-3140-9896>

**Belyaev Yakov V.**

**Беляев Яков Валерьевич**

E-mail: jak0b@mail.ru

**Loboda Vera V.**

**Лобода Вера Владимировна**

E-mail: vera\_loboda@mail.ru

ORCID: <https://orcid.org/0000-0003-3103-7060>

*Submitted: 13.01.2025; Approved: 26.03.2025; Accepted: 28.03.2025.*

*Поступила: 13.01.2025; Одобрена: 26.03.2025; Принята: 28.03.2025.*

# Simulations of Computer, Telecommunications and Control Systems

## Моделирование вычислительных, телекоммуникационных и управляющих систем



Research article

DOI: <https://doi.org/10.18721/JCSTCS.18109>

UDC 62-519



### OPTIMIZATION MODEL OF THE PROCESSING PARAMETERS FOR STRUCTURAL ELEMENTS OF A PRODUCT

*I.N. Khrustaleva*  , *V.P. Shkodyrev*,  
*V.N. Khokhlovskiy*, *L.G. Chernyh*, *S.N. Stepanov*

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

 [irina.khrustaleva@mail.ru](mailto:irina.khrustaleva@mail.ru)

**Abstract.** Optimization of the target indicators of the technological process is a key factor in increasing the efficiency of the product manufacturing process. The efficiency of the optimization process directly depends on the degree of detail of the control object. The purpose of the study is to increase the efficiency of the process of forming individual geometric elements of a part through multi-criteria optimization of the technological process parameters. The paper presents a structural hierarchical model of optimizing the parameters of the process of forming a geometric element. This model is a structural decomposition of the goals to be achieved within the identified control level. Based on the structural decomposition, four levels of process control are identified. This hierarchy of goals allows increasing the efficiency of the geometric element formation process through detailed analysis and optimization of target indicators at each stage of the process. The paper considers an example of optimization of the process parameters for machining a group of threaded holes M27x2-6H in a product made of dispersion-hardened composite alloy SAS-50. Optimum values of the process parameters for each forming stage are determined for the investigated group of holes according to the structural model of the process. As a result of optimizing the process parameters, the accuracy of manufacturing a group of threaded holes increased by 22.2%, while the labor intensity increased by 13.69%.

**Keywords:** multi-criteria optimization, geometric element, processing route, structural hierarchical model, control level

**Acknowledgements:** The research was supported by the Russian Science Foundation grant No. 23-29-00551 “Methods and algorithms for constructing intelligent cyber-physical systems to ensure semantic interoperability”. Available online: <https://rscf.ru/project/23-29-00551/>.

**Citation:** Khrustaleva I.N., Shkodyrev V.P., Khokhlovskiy V.N., et al. Optimization model of the processing parameters for structural elements of a product. *Computing, Telecommunications and Control*, 2025, Vol. 18, No. 1, Pp. 111–129. DOI: 10.18721/JCSTCS.18109


Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18109>

УДК 62-519



## МОДЕЛЬ ОПТИМИЗАЦИИ ПАРАМЕТРОВ ОБРАБОТКИ СТРУКТУРНЫХ ЭЛЕМЕНТОВ ИЗДЕЛИЯ

*И.Н. Хрусталева* , *В.П. Шкодырев,*  
*В.Н. Хохловский, Л.Г. Черных, С.Н. Степанов*

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

✉ [irina.khrustaleva@mail.ru](mailto:irina.khrustaleva@mail.ru)

**Аннотация.** Оптимизация целевых показателей технологического процесса является ключевым фактором повышения эффективности процесса изготовления изделия. Эффективность процесса оптимизации напрямую зависит от степени детализации объекта управления. Целью исследования является повышение эффективности процесса формообразования отдельных геометрических элементов детали за счет многокритериальной оптимизации параметров технологического процесса. В работе представлена структурная иерархическая модель оптимизации параметров процесса формообразования геометрического элемента. Данная модель представляет собой структурную декомпозицию целей, которые должны быть достигнуты в рамках выделенных уровней управления. На основе структурной декомпозиции выделено четыре уровня управления процессом. Данная иерархия целей позволяет повысить эффективность процесса формообразования геометрического элемента за счет детального анализа и оптимизации целевых показателей на каждом этапе процесса. В работе рассмотрен пример оптимизации параметров процесса обработки группы резьбовых отверстий М27х2-6Н в изделии, изготавливаемом из дисперсно-упрочненного композиционного сплава САС-50. Для исследуемой группы отверстий определены оптимальные значения технологических параметров для каждого этапа формообразования согласно структурной модели процесса. В результате оптимизации параметров процесса точность изготовления группы резьбовых отверстий повысилась на 22,2%, при этом трудоемкость увеличилась на 13,69%.

**Ключевые слова:** многокритериальная оптимизация, геометрический элемент, маршрут обработки, структурная иерархическая модель, уровень управления

**Финансирование:** Исследование выполнено за счет гранта Российского научного фонда в рамках реализации проекта «Методы и алгоритмы построения интеллектуальных кибер-физических систем для обеспечения семантической интероперабельности» (Соглашение № 23-29-00551, <https://rscf.ru/project/23-29-00551/>).

**Для цитирования:** Khrustaleva I.N., Shkodyrev V.P., Khokhlovskiy V.N., et al. Optimization model of the processing parameters for structural elements of a product // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 111–129. DOI: 10.18721/JCSTCS.18109

### Introduction

Each product in mechanical engineering, supplied in conditions of fierce competition to the domestic and foreign markets, must have a new level of properties and meet the requirements imposed on potential consumers for the functional properties of the product. Therefore, one of the main goals for machine-building enterprises is the constant improvement of the parameters of both the product itself and the manufacturing process [1].

Modern growth rates of the global economy require machine-building production to produce competitive products with minimal time costs and high performance characteristics. In modern economic conditions, increasing the efficiency of the production process by optimizing technological parameters is a priority for industrial enterprises. Increasing the efficiency of processing is possible by improving existing



and creating new methods for assigning processing modes, which will improve the quality of products, ensure high performance and reduce the complexity of processing. The efficiency of the technological operation is determined by the quality of the surface, accuracy and productivity. A large number of works related to the development of methods and models for optimizing production processes are devoted to solving this scientific problem [2–9].

One of the main management tasks in the process preparation of production is the optimization of process parameters. The paper discusses the issues of multi-criteria optimization of the process of forming geometric elements based on a structural hierarchical process model. In this case, the task is to form the best – optimal – management strategy within a multi-level hierarchical system. The general target state of the control object is characterized by a balanced system of targets that determine the overall efficiency of the system.

A large number of scientific papers [10–15] are devoted to the problems of optimizing the parameters of the technological process and the introduction of digital technologies. Today's digital technologies, such as IoT, cloud computing, big data analytics and AI, can dramatically improve the efficiency of the manufacturing process [16].

The optimization of production processes is based on big data analytics, which includes data collection, processing and analysis based on developed methods and algorithms [17–19].

Currently, knowledge graphs are widely used to solve optimization problems [20]. A knowledge graph is a semantic network that contains information about the structural elements of a research object and the relationships between them. The use of knowledge graphs for solving practical production problems is presented in [20–22].

The purpose of the study is to increase the efficiency of the product manufacturing process by optimizing the parameters of the process of manufacturing its individual elements.

The objectives of the study are to analyze the factors affecting the efficiency of the formation process of individual geometric elements; to develop a structural hierarchical model of geometric elements shaping process.

The object of the study is the process of forming geometric elements that form the structure of mechanical engineering products.

### Structural model of the control object

Investigation of geometric element shaping process and determination of optimal values of processing parameters is based on structural decomposition of investigated process. The structural model of the process contains the following set of structural elements forming the corresponding control levels (Fig. 1):

- The first process control level: the technological processing route.
- The second process control level: the processing stage.
- The third process control level: the technological transition.
- The fourth process control level: the working stroke, the auxiliary transition.

The control object, which graph is shown in Fig. 1, reflects the sequence of intermediate states of the geometric element (vertex of the graph) and the conditions for the transition of the control object from the  $i$ -th state to the  $(i + 1)$ -th (arcs of the graph). Table 1 presents a list of tasks to be solved within each process control level.

At the first level, the control object has two states  $S_{wp}$  and  $S_{ge}$  (Fig. 2). The condition for changing the properties of the object within the first process control level can be described by the following expression:

$$S_{ge} = f(S_{wp}, U_1), U_1 \in D, \tag{1}$$

where  $S_{wp}$  is the state of the control object corresponding to the stock properties;  $S_{ge}$  is the condition of the control object corresponding to properties of a finished product which parameters are regulated by

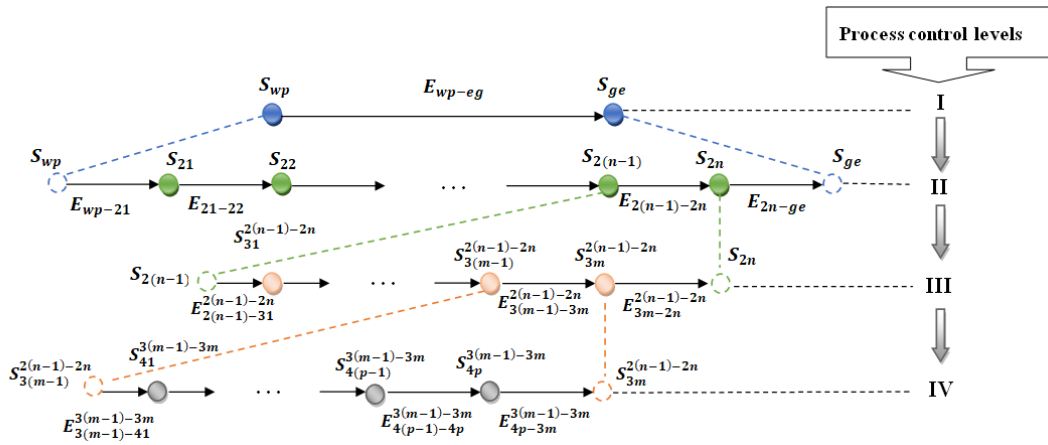


Fig. 1. The structural model of the control object

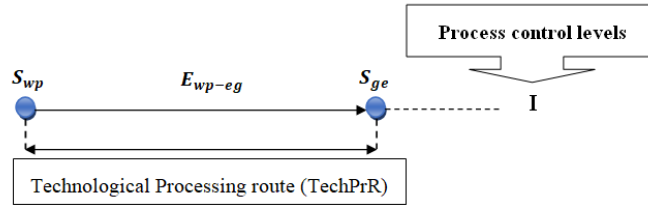


Fig. 2. Object state graph at the first process control level

design documentation;  $U_1$  is the set of the control parameters corresponding to the first process control level;  $D$  is the set of the optimization model control parameters.

The  $U_1$  set has the following structure (2):

$$U_1 = \left\{ N^{PrSt}, \left( Type_{ShM}^{PrSt} \right)_n, \beta_{bas}^{TechPrR}, \beta_{aux}^{TechPrR}, \alpha_{ct}^{TechPrR}, \alpha_w^{TechPrR}, \alpha_{qwq}^{TechPrR}, \alpha_{am}^{TechPrR}, \alpha_{rep}^{TechPrR}, \alpha_{el.en.}^{TechPrR}, \alpha_{dif}^{TechPrR} \right\} \quad (2)$$

where  $N^{PrSt}$  is the number of processing stages within the geometric element technological processing route, pcs;  $\left( Type_{ShM}^{PrSt} \right)_n$  are the types of shaping methods used to implement the respective processing steps;  $\beta_{bas}^{TechPrR}$  is the share of time aimed at implementation of a set of basic actions in the total labor intensity of the technological processing route;  $\beta_{aux}^{TechPrR}$  is the share of time aimed at implementation of a set of auxiliary actions in the total labor intensity of the technological processing route;  $\alpha_{ct}^{TechPrR}$  is the share of cutting tool costs in the total amount of operating costs for the implementation of the technological processing route;  $\alpha_w^{TechPrR}$  is the share of costs intended for payment of wages to production workers in the total amount of operating costs for the implementation of the technological processing route;  $\alpha_{qwq}^{TechPrR}$  is the share of quick-wear equipment costs in the total amount of operating costs for the implementation of the technological processing route;  $\alpha_{am}^{TechPrR}$  is the share of depreciation expenses in the total amount of operating costs for the implementation of the technological processing route;  $\alpha_{rep}^{TechPrR}$  is the share of costs intended for maintenance and repair of technological equipment in the total amount of operating costs for the implementation of the technological processing route;  $\alpha_{el.en.}^{TechPrR}$  is the share of electricity costs in the total amount of operating costs for the implementation of the

Table 1

**The list of tasks to be solved within process control level**

| Process control level       | Tasks solved within the process control level  |
|-----------------------------|--|
| Process control level No. 1 | 1. Optimization of technological route processing structure of a geometric element, which consists of determining the optimal number of processing stages and shaping methods used to change the properties of the control object.<br>2. Optimization of values of the components of the process cost parameter within the implementation of the technological processing route.<br>3. Optimization of labor intensity of the complex of basic and auxiliary actions within the technological processing route.<br>4. Optimization of accuracy values of geometrical parameters of the control object generated during implementation of the corresponding processing stage. |
| Process control level No. 2 | 1. Optimization of processing stages structure, consisting in determining the optimal number of technological transitions necessary to change the properties of the control object.<br>2. Optimization of values of process cost parameter components within implementation of n-th processing stage.<br>3. Optimization of labor intensity of main and auxiliary actions within n-th processing stage.<br>4. Optimization of accuracy values of geometric parameters of the control object formed as a result of implementation of the m-th technological transition performed within the the n-th processing stage.  |
| Process control level No. 3 | 1. Optimization of the structure of technological transitions, which consists in determining the optimal number of working strokes used to change the properties of the control object within the m-th technological transition.<br>2. Optimization of values of the components of the process cost parameter as part of the implementation of the m-th technological transition.<br>3. Optimization of labor intensity of basic and auxiliary actions as part of the implementation of the m-th technological transition.<br>4. Optimization of tolerance fields for each geometric parameter, within the p-th working stroke of the m-th technological transition.         |
| Process control level No. 4 | 1. Optimization of values of the components of the process cost parameter within the implementation of the p-th working stroke.<br>2. Optimization of labor intensity of p-th working stroke implemented within m-th technological transition.<br>3. Optimization of the accuracy of a feature's geometric parameters within a specified tolerance field.  |

technological processing route;  $\alpha_{dif}^{TechPrR}$  is the share of other costs in the total amount of operating costs for the implementation of the technological processing route.

Three sets of targets are defined for the first process control level:

$$Tr^{11} = \left\{ \left( Er_1^{PrSt} \right)_1, \dots, \left( Er_1^{PrSt} \right)_n, \left( Er_i^{PrSt} \right)_1, \dots, \left( Er_i^{PrSt} \right)_n \right\}, Tr^{11} \subset Tr^1, \quad (3)$$

where  $Tr^1$  is the set of the targets for the first process control level;  $Tr^{11}$  is the subset of target indicators characterizing the accuracy of the parameters of the geometric element after the implementation of the corresponding processing stage;  $\left( Er_1^{PrSt} \right)_1, \dots, \left( Er_1^{PrSt} \right)_n, \left( Er_i^{PrSt} \right)_1, \dots, \left( Er_i^{PrSt} \right)_n$  is the accuracy of the 1...i-th parameter of the geometric element formed during the implementation of the 1...n-th processing stage,  $\mu m$ .

$$Tr^{12} = \left\{ C_{ct}^{TechPrR}, C_w^{TechPrR}, C_{qwq}^{TechPrR}, C_{am}^{TechPrR}, C_{rep}^{TechPrR}, C_{el.en.}^{TechPrR}, C_{dif.}^{TechPrR} \right\}, Tr^{12} \subset Tr^1, \quad (4)$$

where  $Tr^{12}$  is a subset of target indicators characterizing the cost values for the corresponding calculation items that arise during the implementation of the technological processing route, rubles;  $C_{ct}^{TechPrR}$  is the

amount of cutting tool costs used in the process of the implementation of the technological processing route, rubles;  $C_w^{TechPrR}$  is the amount of expenses required for payment of wages to production workers involved in the implementation of the technological processing route, rubles;  $C_{qwq}^{TechPrR}$  is the amount of the quick-wear equipment costs used in the process of the implementation of the technological processing route, rubles;  $C_{am}^{TechPrR}$  is the amount of depreciation expenses as part of the implementation of the amount of the quick-wear equipment costs used in the process of the implementation of the technological processing route, rubles;  $C_{rep}^{TechPrR}$  is the amount of expenses intended for maintenance and repair of equipment used in the implementation of the technological processing route, rubles;  $C_{el.en.}^{TechPrR}$  is the amount of energy costs required to the implementation of the technological processing route, rubles;  $C_{dif.}^{TechPrR}$  is the amount of other costs required to the implementation of the technological processing route, rubles.

$$Tr^{13} = \{T_{bas}^{TechPrR}, T_{aux}^{TechPrR}\}, Tr^{13} \subset Tr^1, \quad (5)$$

where  $Tr^{13}$  is the subset of target indicators characterizing the labor intensity of performing complexes of basic and auxiliary actions in the process of the implementation of the technological processing route;  $T_{bas}^{TechPrR}$  is the amount of time spent on performing a set of basic actions in the process of the implementation of the technological processing route, min.;  $T_{aux}^{TechPrR}$  is the amount of time spent on performing a set of auxiliary actions in the process of the implementation of the technological processing route, min.

At the second process control level, parameters are analyzed and optimized within the processing stage (Fig. 3).

The change of object properties within the second process control level is described by the following sequence of intermediate states corresponding to the properties of the control object after the implementation of the corresponding processing step:

$$S_{wp} \rightarrow S_{21} \rightarrow \dots \rightarrow S_{2(n-1)} \rightarrow S_{2n} \rightarrow S_{ge}. \quad (6)$$

The condition for changing the properties of the control object within the  $n$ -th processing stage can be described by the following expression:

$$S_{2n} = f(S_{2(n-1)}, U_2), U_2 \in D, \quad (7)$$

where  $S_{2n}$  is the state of the control object corresponding to the properties of the geometric element after the implementation of the  $n$ -th processing stage;  $S_{2(n-1)}$  is the state of the control object corresponding to the properties of the geometric element after the implementation of the  $(n-1)$ -th processing stage;  $U_2$  is the set of control parameters corresponding to the second process control level.

The  $U_2$  set has the following structure:

$$U_2 = \left\{ \left( N^{TechTr} \right)_n, \left( \beta_{bas}^{PrSt} \right)_n, \left( \beta_{aux}^{PrSt} \right)_n, \left( \alpha_{ct}^{PrSt} \right)_n, \right. \\ \left. \left( \alpha_w^{PrSt} \right)_n, \left( \alpha_{qwq}^{PrSt} \right)_n, \left( \alpha_{am}^{PrSt} \right)_n, \left( \alpha_{rep}^{PrSt} \right)_n, \left( \alpha_{el.en.}^{PrSt} \right)_n, \left( \alpha_{dif}^{PrSt} \right)_n \right\}, \quad (8)$$

where  $\left( N^{TechTr} \right)_n$  is the number of technological transitions within the  $n$ -th processing stage of geometric element, pcs;  $\left( \beta_{bas}^{PrSt} \right)_n$  is the share of the main time for the implementation of the  $n$ -th processing stage in the total labor intensity of performing a set of main actions within the technological processing route;  $\left( \beta_{aux}^{PrSt} \right)_n$  is the share of auxiliary time for the implementation of the  $n$ -th processing stage in the total labor intensity of performing a set of auxiliary actions within the technological processing route;

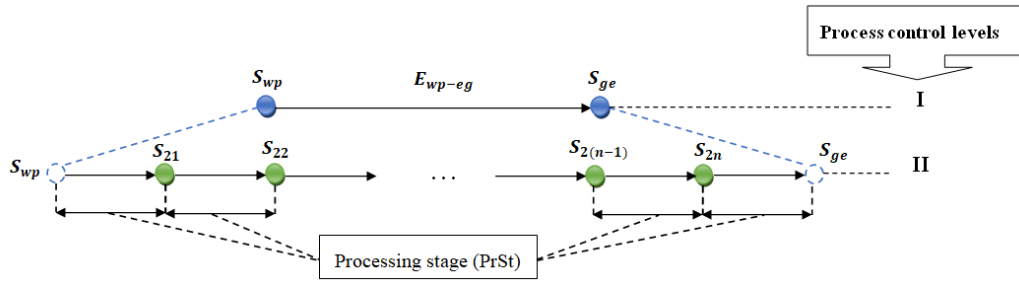


Fig. 3. Object state graph at the second process control level

$(\alpha_{ct}^{PrSt})_n$  is the share of cutting tool costs for  $n$ -th processing stage in the total amount of operating costs for the cutting tools for the implementation of the technological processing route;  $(\alpha_w^{PrSt})_n$  is the share of costs intended for payment of wages to production workers involved in the implementation of the  $n$ -th processing stage in the total amount of operating costs for wages for the implementation of the technological processing route;  $(\alpha_{qwq}^{PrSt})_n$  is the share of quick-wear equipment costs for  $n$ -th processing stage in the total amount of operating costs for the quick-wear equipment for the implementation of the technological processing route;  $(\alpha_{am}^{PrSt})_n$  is the share of costs intended for depreciation within the  $n$ -th processing stage in the total amount of operating costs for depreciation expenses within the implementation of the technological processing route;  $(\alpha_{rep}^{PrSt})_n$  is the share of costs for maintenance and repair of process equipment within the  $n$ -th processing stage in the total cost of maintenance and repair of process equipment within the implementation of the technological processing route;  $(\alpha_{el.en.}^{PrSt})_n$  is the share of the cost of electricity spent on the implementation of the  $n$ -th processing stage in the total cost of electricity spent on the implementation of the technological processing route;  $(\alpha_{dif}^{PrSt})_n$  is the share of other costs for the implementation of the  $n$ -th processing stage in the total amount of other costs required for the implementation of the technological processing route.

Three sets of targets are defined for the second control level:

$$Tr^{21} = \left\{ (Er_1^{TechTr})_1, \dots, (Er_1^{TechTr})_n, (Er_i^{TechTr})_1, \dots, (Er_i^{TechTr})_n \right\}, Tr^{21} \subset Tr^2, \quad (9)$$

where  $Tr^2$  is the set of the targets for the second process control level;  $Tr^{21}$  is the subset of target indicators characterizing the accuracy parameters of the control object after the implementation of the corresponding processing stage;  $(Er_1^{PrSt})_1, \dots, (Er_1^{PrSt})_n, (Er_i^{PrSt})_1, \dots, (Er_i^{PrSt})_n$  is the accuracy of the  $i$ -th geometric parameter of the control object generated during the implementation of the  $n$ -th technological transition,  $\mu\text{m}$ .

$$Tr^{22} = \left\{ \begin{array}{l} (C_{ct}^{PrSt})_1, \dots, (C_{ct}^{PrSt})_n, (C_w^{PrSt})_1, \dots, (C_w^{PrSt})_n, (C_{qwq}^{PrSt})_1, \dots, (C_{qwq}^{PrSt})_n, \\ (C_{am}^{PrSt})_1, \dots, (C_{am}^{PrSt})_n, (C_{rep}^{PrSt})_1, \dots, (C_{rep}^{PrSt})_n, (C_{el.en.}^{PrSt})_1, \dots, (C_{el.en.}^{PrSt})_n, \\ (C_{dif}^{PrSt})_1, \dots, (C_{dif}^{PrSt})_n \end{array} \right\}, Tr^{22} \subset Tr^2, \quad (10)$$

where  $Tr^{22}$  is the subset of the target indicators characterizing the sizes of the expenses under the relevant articles of accounting arising during the implementation of the  $n$ -th processing stages;  $(C_{ct}^{PrSt})_1, \dots,$

$(C_{ct}^{PrSt})_n$  is the amount of cutting tool costs used in the process of the implementation of the  $n$ -th processing stages, rubles;  $(C_w^{PrSt})_1, \dots, (C_w^{PrSt})_n$  is the amount of expenses required for the payment of wages to production workers involved in the implementation of the  $n$ -th processing stages, rubles;  $(C_{qwq}^{PrSt})_1, \dots, (C_{qwq}^{PrSt})_n$  is the amount of quick-wearing equipment costs used in the process of the implementation of the  $n$ -th processing stages, rubles;  $(C_{am}^{PrSt})_1, \dots, (C_{am}^{PrSt})_n$  is the amount of expenses allocated for depreciation expenses as part of the implementation of the  $n$ -th processing stages, rubles;  $(C_{rep}^{PrSt})_1, \dots, (C_{rep}^{PrSt})_n$  is the amount of expenses intended for maintenance and repair of equipment used in the implementation of  $n$ -th processing stage, rubles;  $(C_{el.en.}^{PrSt})_1, \dots, (C_{el.en.}^{PrSt})_n$  is the amount of energy costs required to realize  $n$ -th processing stage, rubles;  $(C_{dif}^{PrSt})_1, \dots, (C_{dif}^{PrSt})_n$  is the amount of other costs required for the implementation of the  $n$ -th processing stage, rubles.

$$Tr^{23} = \left\{ (T_{bas}^{PrSt})_1, \dots, (T_{bas}^{PrSt})_n, (T_{aux}^{PrSt})_1, \dots, (T_{aux}^{PrSt})_n \right\}, Tr^{23} \subset Tr^2, \quad (11)$$

where  $Tr^{23}$  is the subset of target indicators characterizing the labor intensity of performing complexes of basic and auxiliary actions in the process of the implementation of the  $n$ -th processing stage;  $(T_{bas}^{PrSt})_1, \dots, (T_{bas}^{PrSt})_n$  is the amount of time spent on performing a set of basic actions during the implementation of the  $n$ -th processing stage, min.;  $(T_{aux}^{PrSt})_1, \dots, (T_{aux}^{PrSt})_n$  is the amount of time spent on performing a set of auxiliary actions during the implementation of the  $n$ -th processing stage, min.

At the third process control level, process parameters are analyzed and optimized as part of the technological transition (Fig. 4).

The change of object properties within the third process control level is described by the following sequence of intermediate states corresponding to the properties of the control object after the implementation of the  $m$ -th technological transition:

$$S_{2(n-1)} \rightarrow S_{31}^{2(n-1)-2n} \rightarrow S_{32}^{2(n-1)-2n} \rightarrow \dots \rightarrow S_{3(m-1)}^{2(n-1)-2n} \rightarrow S_{3m}^{2(n-1)-2n} \rightarrow S_{2n}. \quad (12)$$

The condition for changing the properties of a control object within a technological transition can be described by the following expression:

$$S_{3m} = f(S_{3(m-1)}, U_3), U_3 \in D, \quad (13)$$

where  $S_{3m}$  is the state of the control object corresponding to the geometry properties after the  $m$ -th technological transition;  $S_{3(m-1)}$  is the state of the control object corresponding to the properties of the geometric element after the  $(m-1)$ -th technological transition;  $U_3$  is the set of control parameters corresponding to the third process control level.

The  $U_3$  set has the following structure:

$$U_3 = \left\{ \left( N^{WSt} \right)_{mn}, \left( \beta_{bas}^{TechTr} \right)_{mn}, \left( \beta_{aux}^{TechTr} \right)_{mn}, \left( \alpha_{ct}^{TechTr} \right)_{mn}, \left( \alpha_w^{TechTr} \right)_{mn}, \left( \alpha_{qwq}^{TechTr} \right)_{mn}, \left( \alpha_{am}^{TechTr} \right)_{mn}, \left( \alpha_{rep}^{TechTr} \right)_{mn}, \left( \alpha_{el.en.}^{TechTr} \right)_{mn}, \left( \alpha_{dif}^{TechTr} \right)_{mn} \right\}, \quad (14)$$

where  $(N^{WSt})_{mn}$  is the number of working strokes within the  $m$ -th technological transition of the  $n$ -th processing stage;  $(\beta_{bas}^{TechTr})_{mn}$  is the labor intensity of the complex of main actions within the  $m$ -th technological transition in the total labor intensity of the complex of main actions of the  $n$ -th processing stage;

$(\beta_{aux}^{TechTr})_{mn}$  is the labor intensity of the set of auxiliary actions within the  $m$ -th technological transition in the total labor intensity of the set of auxiliary actions of the  $n$ -th processing stage;  $(\alpha_{ct}^{TechTr})_{mn}$  is the share of cutting tool costs for the  $m$ -th technological transition in the total amount of operating costs for the cutting tools for the implementation of the  $n$ -th processing stage;  $(\alpha_w^{TechTr})_{mn}$  is the share of costs intended for payment of wages to production workers involved in the implementation of the  $m$ -th technological transition in the total amount of operating costs for wages to production workers involved in the implementation of the  $n$ -th processing stage;  $(\alpha_{qwq}^{TechTr})_{mn}$  is the share of quick-wear equipment costs for the  $m$ -th technological transition in the total amount of operating costs for the quick-wear equipment for the implementation of the  $n$ -th processing stage;  $(\alpha_{am}^{TechTr})_{mn}$  is the share of depreciation costs for the  $m$ -th technological transition in the total amount of operating expenses for depreciation for the  $n$ -th processing stage;  $(\alpha_{rep}^{TechTr})_{mn}$  is the share of costs intended for maintenance and repair of technological equipment involved in the implementation of the  $m$ -th technological transition in the total amount of operating costs for maintenance and repair of technological equipment involved in the implementation of the  $n$ -th stage of processing;  $(\alpha_{el.en.}^{TechTr})_{mn}$  is the share of the cost of electricity spent on the implementation of the  $m$ -th technological transition in the total operating costs of electricity spent on the implementation of the  $n$ -th processing stage;  $(\alpha_{dif}^{TechTr})_{mn}$  is the share of other costs for the implementation of the  $m$ -th technological transition in the total amount of other costs for the implementation of the  $n$ -th processing stage.

Three sets of targets are defined for the third process control level:

$$Tr^{31} = \left\{ (Er_1^{WSt})_{1p}, \dots, (Er_1^{WSt})_{pm}, (Er_i^{WSt})_{1p}, \dots, (Er_i^{WSt})_{pm} \right\}, Tr^{31} \subset Tr^3, \quad (15)$$

where  $Tr^3$  is the set of the targets for the third process control level;  $Tr^{31}$  is a subset of target indicators characterizing the accuracy parameters of the  $i$ -th geometric parameter of the control object after the implementation of the  $p$ -th working stroke within the  $m$ -th technological transition;  $(Er_1^{WSt})_{1p}, \dots, (Er_1^{WSt})_{pm}, (Er_i^{WSt})_{1p}, \dots, (Er_i^{WSt})_{pm}$  is the accuracy of the  $i$ -th geometric parameter of the control object formed during the implementation of the  $p$ -th working stroke within the  $m$ -th technological transition.

$$Tr^{32} = \left\{ \begin{array}{l} (C_{ct}^{TechTr})_{1n}, \dots, (C_{ct}^{TechTr})_{mn}, (C_w^{TechTr})_{1n}, \dots, (C_w^{TechTr})_{mn}, (C_{qwq}^{TechTr})_{1n}, \dots, (C_{qwq}^{TechTr})_{mn}, \\ (C_{am}^{TechTr})_{1n}, \dots, (C_{am}^{TechTr})_{mn}, (C_{rep}^{TechTr})_{1n}, \dots, (C_{rep}^{TechTr})_{mn}, (C_{el.en.}^{TechTr})_{1n}, \dots, (C_{el.en.}^{TechTr})_{mn}, \\ (C_{dif}^{TechTr})_{1n}, \dots, (C_{dif}^{TechTr})_{mn} \end{array} \right\}, \quad (16)$$

$$Tr^{32} \subset Tr^3,$$

where  $Tr^{32}$  is a subset of target indicators characterizing the cost values for the corresponding calculation items arising during the implementation of the  $m$ -th technological transition within the  $n$ -th processing stage;  $(C_{ct}^{TechTr})_{1n}, \dots, (C_{ct}^{TechTr})_{mn}$  is the amount of cutting tool costs used in the process of the implementation of the  $m$ -th technological transition within the  $n$ -th processing stage;  $(C_w^{TechTr})_{1n}, \dots, (C_w^{TechTr})_{mn}$  is the amount of costs required to pay wages to production workers involved in the imple-

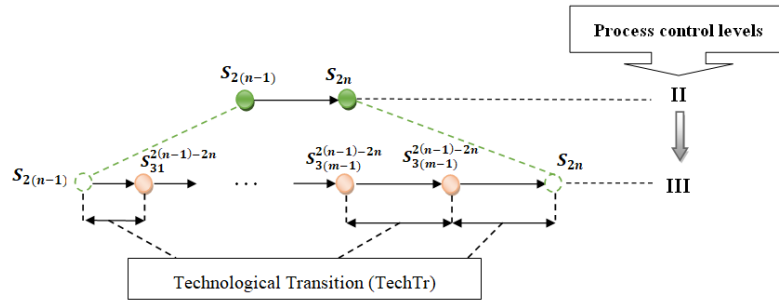


Fig. 4. Object state graph at the third process control level

mentation of the  $m$ -th technological transitions of the  $n$ -th processing stage;  $(C_{qwq}^{TechTr})_{1n}, \dots, (C_{qwq}^{TechTr})_{mn}$  is the amount of quick-wear equipment costs used in the process of the implementation of the  $m$ -th technological transitions of the  $n$ -th processing stage;  $(C_{am}^{TechTr})_{1n}, \dots, (C_{am}^{TechTr})_{mn}$  is the amount of expenses allocated for depreciation expenses as part of the implementation of the  $m$ -th technological transitions of the  $n$ -th processing stage;  $(C_{rep}^{TechTr})_{1n}, \dots, (C_{rep}^{TechTr})_{mn}$  is the amount of costs intended for maintenance and repair of equipment used as part of the implementation of the  $m$ -th technological transition of the  $n$ -th processing stage;  $(C_{el.en.}^{TechTr})_{1n}, \dots, (C_{el.en.}^{TechTr})_{mn}$  is the amount of energy costs required for the implementation of the  $m$ -th technological transition of the  $n$ -th processing stage;  $(C_{dif}^{TechTr})_{1n}, \dots, (C_{dif}^{TechTr})_{mn}$  is the amount of other costs required for the implementation of the  $m$ -th technological transition of the  $n$ -th processing stage.

$$Tr^{33} = \left\{ (T_{bas}^{TechTr})_{1n}, \dots, (T_{bas}^{TechTr})_{mn}, (T_{aux}^{TechTr})_{1n}, \dots, (T_{aux}^{TechTr})_{mn} \right\}, Tr^{33} \subset Tr^3, \quad (17)$$

where  $Tr^{33}$  is a subset of targets characterizing the complexity of performing complexes of basic and auxiliary actions during the implementation of the  $m$ -th technological transition of the  $n$ -th processing stage;  $(T_{bas}^{TechTr})_{1n}, \dots, (T_{bas}^{TechTr})_{mn}$  is the amount of time spent on performing a set of main actions during the implementation of the  $m$ -th technological transition of the  $n$ -th processing stage;  $(T_{aux}^{TechTr})_{1n}, \dots, (T_{aux}^{TechTr})_{mn}$  is the amount of time spent on performing a set of auxiliary actions during the implementation of the  $m$ -th technological transition of the  $n$ -th processing stage.

At the fourth control level, the process parameters are analyzed and optimized as part of the working stroke (Fig. 5).

The change of object properties within the fourth process control level is described by the following sequence of intermediate states corresponding to the properties of the control object after the implementation of the  $p$ -th working stroke:

$$S_{wp} \rightarrow S_{41}^{3(m-1)-3m} \rightarrow \dots \rightarrow S_{4(p-1)}^{3(m-1)-3m} \rightarrow S_{4p}^{3(m-1)-3m} \rightarrow S_{ge}. \quad (18)$$

The condition for changing the properties of the control object within the work stroke can be described by the following expression:

$$S_{4p} = f(S_{4(p-1)}, U_4), U_4 \in D, \quad (19)$$



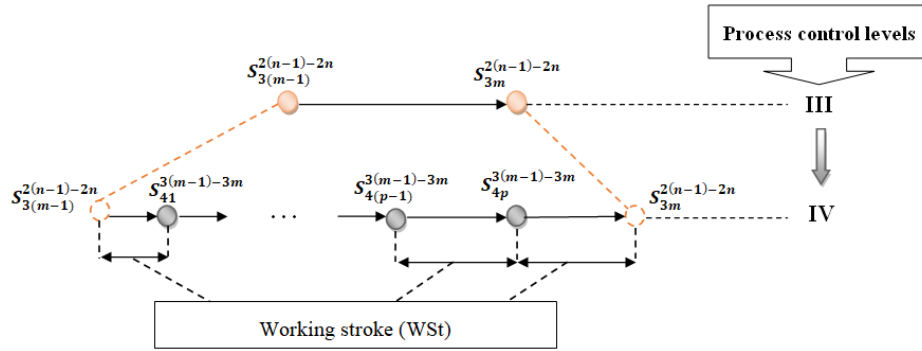


Fig. 5. Object state graph at the fourth process control level

where  $S_{4p}$  is the state of the control object corresponding to the product properties after the  $p$ -th working stroke;  $S_{4(p-1)}$  is the state of the control object corresponding to the product properties after the  $(p-1)$ -th working stroke;  $U_4$  is the set of control parameters corresponding to the fourth process control level.

The  $U_4$  set has the following structure:

$$\begin{aligned}
 U_4 = & \left\{ \beta_{pm}^{WSt}, \beta_{rm}^{AuxTr}, (\alpha_{ct}^{WSt})_{pm}, (\alpha_w^{WSt})_{pm}, (\alpha_w^{AuxTr})_{rm}, (\alpha_{qwq}^{WSt})_{pm}, \right. \\
 & (\alpha_{am}^{WSt})_{pm}, (\alpha_{am}^{AuxTr})_{rm}, (\alpha_{rep}^{WSt})_{pm}, (\alpha_{rep}^{AuxTr})_{rm}, (\alpha_{el.en.}^{WSt})_{pm}, (\alpha_{el.en.}^{AuxTr})_{rm}, \\
 & \left. (\alpha_{dif}^{WSt})_{pm}, (\alpha_{dif}^{AuxTr})_{rm}, (\gamma_i^{EIDif.})_{pm}^{WSt}, (\gamma_i^{Dim.W})_{pm}^{WSt}, (\gamma_i^{ThSt.})_{pm}^{WSt} \right\},
 \end{aligned} \quad (20)$$

where  $\beta_{pm}^{WSt}$  is the share of the time spent on the implementation of the  $p$ -th working stroke in the total labor intensity of the complex of basic actions within the  $m$ -th technological transition;  $\beta_{rm}^{AuxTr}$  is the share of the time spent on the implementation of the  $r$ -th auxiliary transition in the total complexity of performing a set of auxiliary actions within the  $m$ -th technological transition;  $(\alpha_{ct}^{WSt})_{pm}$  is the share of cutting tool costs for  $p$ -th working stroke in the total amount of operating costs for the cutting tools for the implementation of the  $m$ -th technological transition;  $(\alpha_w^{WSt})_{pm}$  is the share of costs intended for payment of wages to production workers involved in the implementation of  $p$ -th working stroke in the total amount of operating costs for wages to production workers involved in the implementation of the  $m$ -th technological transition;  $(\alpha_w^{AuxTr})_{rm}$  is the share of costs intended for payment of wages to production workers involved in the implementation of the  $r$ -th auxiliary transition in the total amount of operating costs for wages to production workers involved in the implementation of the  $m$ -th technological transition;  $(\alpha_{qwq}^{WSt})_{pm}$  is the share of quick-wear equipment costs for  $p$ -th working stroke in the total amount of operating costs for the quick-wear equipment for the implementation of the  $m$ -th technological transition;  $(\alpha_{am}^{WSt})_{pm}$  is the share of depreciation expenses incurred during the implementation of the  $p$ -th working stroke in the total amount of operating expenses for depreciation expenses incurred during the implementation of the  $m$ -th technological transition;  $(\alpha_{am}^{AuxTr})_{rm}$  is the share of depreciation expenses incurred during the implementation of the  $r$ -th auxiliary transition in the total amount of operating expenses for depreciation expenses incurred during the implementation of the  $m$ -th technological transition;  $(\alpha_{rep}^{WSt})_{pm}$  is the share of costs intended for maintenance and repair of technological

equipment arising during the implementation of the  $p$ -th working stroke in the total amount of operating costs for maintenance and repair of technological equipment arising within the  $m$ -th technological transition;  $\left(\alpha_{rep}^{AuxTr}\right)_{rm}$  is the share of costs intended for maintenance and repair of technological equipment arising during the implementation of the  $r$ -th auxiliary transition in the total amount of operating costs for maintenance and repair of technological equipment arising within the  $m$ -th technological transition;  $\left(\alpha_{el.en.}^{WSt}\right)_{pm}$  is the share of costs intended to pay for electricity required for the implementation of the  $p$ -th working stroke in the total amount of operating costs for electricity for the  $m$ -th technological transition;  $\left(\alpha_{el.en.}^{AuxTr}\right)_{rm}$  is the share of costs intended to pay for electricity required to implement the  $r$ -th auxiliary transition in the total operating costs of electricity for the  $m$ -th technological transition;  $\left(\alpha_{dif}^{WSt}\right)_{pm}$  is the share of other costs required to implement the  $p$ -th working stroke in the total amount of other costs arising from the implementation of the  $m$ -th technological transition;  $\left(\alpha_{dif}^{AuxTr}\right)_{rm}$  is the share of other costs required for the implementation of the  $r$ -th auxiliary transition in the total amount of other costs arising from the implementation of the  $m$ -th technological transition;  $\left(\gamma_i^{EIDif.}\right)_{pm}^{WSt}$  is the share of error caused by elastic deformations of the process system in the total processing error of the  $i$ -th geometric parameter within the implementation of the  $p$ -th working stroke;  $\left(\gamma_i^{Dim.W}\right)_{pm}^{WSt}$  is the share of the error caused by dimensional wear of the cutting tool in the total error of processing the  $i$ -th geometric parameter within the implementation of the  $p$ -th working stroke;  $\left(\gamma_i^{ThSt.}\right)_{pm}^{WSt}$  is the share of the error caused by thermal deformations of the process systems in the total processing error of the  $i$ -th geometric parameter within the implementation of the working stroke.

Four sets of targets are defined for the fourth process control level:

$$Tr^{A1} = \left\{ \left( Er_i^{EIDif.} \right)_{1m}^{WSt}, \dots, \left( Er_i^{EIDif.} \right)_{pm}^{WSt}, \left( Er_i^{Dim.W} \right)_{1m}^{WSt}, \dots, \left( Er_i^{Dim.W} \right)_{pm}^{WSt}, \left( Er_i^{ThSt} \right)_{1m}^{WSt}, \dots, \left( Er_i^{ThSt} \right)_{pm}^{WSt} \right\}, \quad (21)$$

$$Tr^{A1} \subset Tr^4,$$

where  $Tr^A$  is the set of the targets for the fourth process control level;  $Tr^{A1}$  is the subset of target indicators characterizing the accuracy parameters of the  $i$ -th geometric parameter of the control object after the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $\left( Er_i^{EIDif.} \right)_{1m}^{WSt}, \dots, \left( Er_i^{EIDif.} \right)_{pm}^{WSt}$  is the value of error of the  $i$ -th geometric parameter caused by elastic deformations of the technological system, formed during the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $\left( Er_i^{Dim.W} \right)_{1m}^{WSt}, \dots, \left( Er_i^{Dim.W} \right)_{pm}^{WSt}$  is the value of the  $i$ -th geometric parameter error caused by dimensional wear of the cutting tool, formed during the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $\left( Er_i^{ThSt} \right)_{1m}^{WSt}, \dots, \left( Er_i^{ThSt} \right)_{pm}^{WSt}$  is the value of the  $i$ -th geometric parameter error caused by thermal deformations of the technological system, formed during the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition.

$$Tr^{A2} = \left\{ \left( C_{ct}^{WSt} \right)_{1m}, \dots, \left( C_{ct}^{WSt} \right)_{pm}, \left( C_w^{WSt} \right)_{1m}, \dots, \left( C_w^{WSt} \right)_{pm}, \left( C_{qwq}^{WSt} \right)_{1m}, \dots, \left( C_{qwq}^{WSt} \right)_{pm}, \right. \\ \left. \left( C_{am}^{WSt} \right)_{1m}, \dots, \left( C_{am}^{WSt} \right)_{pm}, \left( C_{rep}^{WSt} \right)_{1m}, \dots, \left( C_{rep}^{WSt} \right)_{pm}, \left( C_{el.en.}^{WSt} \right)_{1m}, \dots, \left( C_{el.en.}^{WSt} \right)_{pm}, \left( C_{dif}^{WSt} \right)_{1m}, \dots, \left( C_{dif}^{WSt} \right)_{pm} \right\},$$

$$Tr^{A2} \subset Tr^4, \quad (22)$$

where  $Tr^{42}$  is the subset of target indicators characterizing the amount of costs for the corresponding calculation items arising during the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $(C_{ct}^{WSt})_{1m}, \dots, (C_{ct}^{WSt})_{pm}$  is the amount of cutting tool costs used in the process of the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $(C_w^{WSt})_{1m}, \dots, (C_w^{WSt})_{pm}$  is the amount of costs required to pay wages to production workers involved in the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $(C_{qwq}^{WSt})_{1m}, \dots, (C_{qwq}^{WSt})_{pm}$  is the amount of costs required for the purchase of quick-wear equipment used in the process of the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $(C_{am}^{WSt})_{1m}, \dots, (C_{am}^{WSt})_{pm}$  is the amount of expenses allocated for depreciation expenses as part of the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $(C_{rep}^{WSt})_{1m}, \dots, (C_{rep}^{WSt})_{pm}$  is the amount of costs intended for maintenance and repair of equipment used in the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition;  $(C_{el.en.}^{WSt})_{1m}, \dots, (C_{el.en.}^{WSt})_{pm}$  is the amount of energy costs required to implement the  $p$ -th working stroke of the  $m$ -th technological transition;  $(C_{dif}^{WSt})_{1m}, \dots, (C_{dif}^{WSt})_{pm}$  is the amount of other costs required for the implementation of the  $p$ -th working stroke of the  $m$ -th technological transition.

$$Tr^{43} = \left\{ \begin{array}{l} (C_w^{AuxTr})_{1m}, \dots, (C_w^{AuxTr})_{rm}, (C_{am}^{AuxTr})_{1m}, \dots, (C_{am}^{AuxTr})_{rm}, \\ (C_{rep}^{AuxTr})_{1m}, \dots, (C_{rep}^{AuxTr})_{rm}, (C_{el.en.}^{AuxTr})_{1m}, \dots, (C_{el.en.}^{AuxTr})_{rm}, (C_{dif}^{AuxTr})_{1m}, \dots, (C_{dif}^{AuxTr})_{rm} \end{array} \right\}, \quad (23)$$

$$Tr^{43} \subset Tr^4,$$

where  $Tr^{43}$  is the subset of target indicators characterizing the cost of the corresponding calculation items arising during the implementation of the  $r$ -th auxiliary transition of the  $m$ -th technological transition;  $(C_w^{AuxTr})_{1m}, \dots, (C_w^{AuxTr})_{rm}$  is the amount of costs required to pay wages to production workers involved in the implementation of the  $r$ -th auxiliary transition of the  $m$ -th technological transition;  $(C_{am}^{AuxTr})_{1m}, \dots, (C_{am}^{AuxTr})_{rm}$  is the amount of expenses allocated for depreciation expenses as part of the implementation of the  $r$ -th auxiliary transition of the  $m$ -th technological transition;  $(C_{rep}^{AuxTr})_{1m}, \dots, (C_{rep}^{AuxTr})_{rm}$  is the amount of costs intended for maintenance and repair of equipment used in the implementation of the  $r$ -th auxiliary transition of the  $m$ -th technological transition;  $(C_{el.en.}^{AuxTr})_{1m}, \dots, (C_{el.en.}^{AuxTr})_{rm}$  is the amount of energy costs required for the implementation of the  $r$ -th auxiliary transition of the  $m$ -th technological transition;  $(C_{dif}^{AuxTr})_{1m}, \dots, (C_{dif}^{AuxTr})_{rm}$  is the amount of other costs required for the implementation of the  $r$ -th auxiliary transition of the  $m$ -th technological transition.

$$Tr^{44} = \{T_{1m}^{WSt}, \dots, T_{pm}^{WSt}, T_{1m}^{AuxTr}, \dots, T_{rm}^{AuxTr}\}, \quad Tr^{44} \subset Tr^4, \quad (24)$$

where  $Tr^{44}$  is the subset of targets characterizing the labor intensity of working strokes and auxiliary transitions within the  $m$ -th technological transition;  $T_{1m}^{WSt}, \dots, T_{pm}^{WSt}$  is the amount of time spent on the  $p$ -th working stroke of the  $m$ -th technological transition, min.;  $T_{1m}^{AuxTr}, \dots, T_{rm}^{AuxTr}$  is the amount of time spent on performing the  $r$ -th auxiliary transition of the  $m$ -th technological transition, min.

### Optimization of parameters of manufacturing process of a group of threaded holes

Based on the model described above, the problems of optimizing the technological parameters of the process of manufacturing a group of threaded holes M27-2-6H (20 pcs) were solved (Fig. 6). The detail “Plate” is made from dispersed-hardened composite alloy SAS-50.

The following optimization task was determined: increasing the accuracy of manufacturing threaded holes M27x2-6N by at least 20%, while increasing the total labor intensity of the processing route should not exceed 30%.

The structural model of the process of machining a group of holes is shown in Fig. 7.

As a result of optimization, the structure of the investigated process was determined, containing three processing stages. The processing stages have the following structure:

- The first processing stage: two technological transitions, each technological transition contains one working stroke. The processing method is drilling.
- The second processing stage: one technological transition containing 11 working strokes. The machining method is milling.
- The third processing stage: one technological transition containing four working strokes. The processing method is thread milling.

Thus, the technological processing route of the investigated group of threaded holes contains four technological transitions:

- Technological transition No. 1 – “Centering”.
- Technological transition No. 2 – “Hole drilling  $\phi 12$ ”.
- Technological transition No. 3 – “Hole milling  $\phi 25$ ”.
- Technological transition No. 4 – “Thread milling M27x2-6H”.

The technological route for processing a group of threaded holes M27x2-6N is multi-stage. In this regard, individual optimization problems were identified for each technological transition:

- Technological transition No. 1: reduction of labor intensity of the forming process by at least 20%, while the increase in the processing error should not exceed 15%.
- Technological transition No. 2: reduction of labor intensity of the forming process by at least 15%, while the increase in the processing error should not exceed 20%.
- Technological transition No. 3: reduction of the processing error by at least 15%, while the increase in labor intensity should not exceed 25%.
- Technological transition No. 4: reduction of the processing error by at least 15%, while the increase in labor intensity should not exceed 30%.

The set of cutting tools used as part of the technological processing route and the corresponding ranges of cutting modes are presented in Table 2.

Table 2

List of cutting tools and cutting mode ranges

| The type of cutting tools                               | Cutting modes |     |              |      |          |     |
|---|---------------|-----|--------------|------|----------|-----|
|   | $V$ , m/min   |     | $S$ , mm/vol |      | $t$ , mm |     |
|   | min           | max | min          | max  | min      | max |
| Centering $\phi 8$ A1174-8                              | 85            | 115 | 0.15         | 0,25 | –        | –   |
| Drill $\phi 12$ A3299XPL-12                             | 160           | 180 | 0.35         | 0.45 | –        | –   |
| Mill $\phi 10$ 1P251-1000-XA 1630                       | 80            | 110 | 0.5          | 0.75 | 1.3      | 0.5 |
| Thread milling cutter<br>P = 2 326R08-B251100VM-TH 1025 | 425           | 450 | 2            |      | 0.1      | 0.5 |

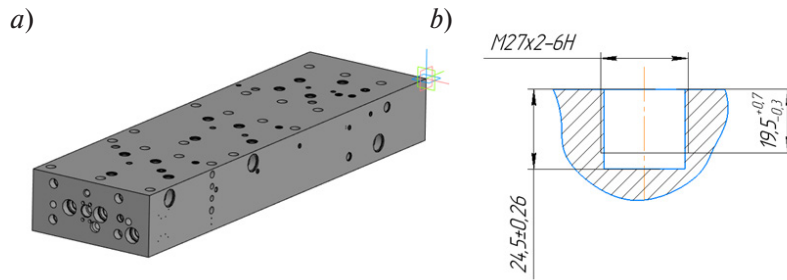


Fig. 6. a) solid model of the “Plate”; b) the sketch of threaded hole

The following optimization criteria are defined:

- The first processing stages:  $(T_{bas}^{PrSt})_1 \rightarrow \min; (T_{bas}^{TechTr})_{1/1} \rightarrow \min; (T_{bas}^{TechTr})_{2/1} \rightarrow \min; T_{1/1}^{WSt} \rightarrow \min; T_{1/2}^{WSt} \rightarrow \min.$
- The second processing stages:  $Er_2^{PrSt} \rightarrow \min; Er_{1/2}^{TechTr} \rightarrow \min; Er_{1/1...1/1}^{WSt} \rightarrow \min.$
- The third processing stages:  $Er_3^{PrSt} \rightarrow \min; Er_{1/3}^{TechTr} \rightarrow \min; Er_{1/4...4/1}^{WSt} \rightarrow \min.$

Tables 3 and 4 present a comparative analysis of target values and processing modes before and after the optimization process. Fig. 8 shows a comparative analysis of the efficiency of implementing a processing route for a group of holes before and after the optimization process.

Table 3

Value of the cut mode settings before and after the optimization process

| Processing stage | Technological transition                                   | Cutting parameter   |              |          |                    |              |          |
|------------------|--|---|--------------|----------|--------------------|--------------|----------|
|                  |  | Before optimization   |              |          | After optimization |              |          |
| 1                | Technological transition No. 1 – “Centering”               | $V$ , m/min   | $s$ , mm/rev | $t$ , mm | $V$ , m/min        | $s$ , mm/rev | $t$ , mm |
|                  |  | Technological transition No. 2 – “Hole drilling $\phi 12$ ” | 90           | 0.17     | –                  | 105          | 0.2      |
| 2                | Technological transition No. 3 – “Hole milling $\phi 25$ ” | 160   | 0.12         | –        | 175                | 0.19         | –        |
| 3                | Technological transition No. 4 – “Thread milling M27x2-6H” | 110   | 2            | 1.0      | 95                 | 2            | 0.8      |
|                  |  | 220   |              | 0.4      | 230                |              | 0.4      |
|                  |  | 220   |              | 0.4      | 230                |              | 0.3      |
|                  |  | 215   |              | 0.2      | 230                |              | 0.2      |
|                  |  |   |              |          | 215                |              | 0.1      |
|                  |  |   |              |          |                    |              |          |

### Conclusions

The following results were obtained:

1. Technological transition No. 1 “Centering”: increase of center hole processing error by 10.2%, while reducing labor intensity by 30%.
2. Technological transition No. 2 – “Hole drilling  $\phi 12$ ”: increase of processing error by 18%, while reducing labor intensity by 23.2%.
3. Technological transition No. 3 – “Hole milling  $\phi 25$ ”: reduction in processing error by 25%, while increasing labor intensity by 23.9%.
4. Technological transition No. 4 – “Thread milling M27x2-6H”: reduction in processing error by 22.2% while increasing labor intensity by 24.32%.

Table 4

Value of targets for process transitions before and after the optimization process

| Processing stage | Technological transition                                   | Targets      |       |                          |       |                         |      |                           |   |                          |      |                           |      |                              |      |                           |      |                           |      |
|------------------|--|--------------|-------|--------------------------|-------|-------------------------|------|---------------------------|---|--------------------------|------|---------------------------|------|------------------------------|------|---------------------------|------|---------------------------|------|
|                  |  | $T_{r^{31}}$ |       | $T_{r^{32}}$             |       |                         |      |                           |   |                          |      | $T_{r^{33}}$              |      |                              |      |                           |      |                           |      |
|                  |  | before       | after | $(C_{cr}^{TechTr})_{mn}$ |       | $(C_{v}^{TechTr})_{mn}$ |      | $(C_{req}^{TechTr})_{mn}$ |   | $(C_{am}^{TechTr})_{mn}$ |      | $(C_{rep}^{TechTr})_{mn}$ |      | $(C_{di.oh.}^{TechTr})_{mn}$ |      | $(T_{bas}^{TechTr})_{mn}$ |      | $(T_{aux}^{TechTr})_{mn}$ |      |
| 1                | Technological transition No. 1 – “Centering”               | 49           | 54    | 2.77                     | 1.94  | 6.8                     | 5.01 | –                         | – | 1.28                     | 0.96 | 0.04                      | 0.03 | 0.06                         | 0.04 | 0.5                       | 0.35 | 0.52                      | 0.41 |
|                  |  | 177          | 209   | 3.1                      | 2.38  | 7.2                     | 5.6  | –                         | – | 1.36                     | 1.05 | 0.05                      | 0.04 | 0.065                        | 0.5  | 0.56                      | 0.43 | 0.52                      | 0.41 |
| 2                | Technological transition No. 3 – “Hole milling φ25”        | 24           | 18    | 3.26                     | 1.76  | 11                      | 12.1 | –                         | – | 2.01                     | 2.28 | 0.11                      | 0.13 | 0.13                         | 0.16 | 1.13                      | 1.4  | 0.52                      | 0.41 |
|                  |  | 18           | 14    | 16.18                    | 20.11 | 22.9                    | 26.9 | –                         | – | 4.33                     | 5.1  | 0.29                      | 0.35 | 0.34                         | 0.43 | 2.92                      | 3.63 | 0.52                      | 0.41 |
| 3                | Technological transition No. 4 – “Thread milling M27x2-6H” | 18           | 14    | 16.18                    | 20.11 | 22.9                    | 26.9 | –                         | – | 4.33                     | 5.1  | 0.29                      | 0.35 | 0.34                         | 0.43 | 2.92                      | 3.63 | 0.52                      | 0.41 |
|                  |  | 18           | 14    | 16.18                    | 20.11 | 22.9                    | 26.9 | –                         | – | 4.33                     | 5.1  | 0.29                      | 0.35 | 0.34                         | 0.43 | 2.92                      | 3.63 | 0.52                      | 0.41 |

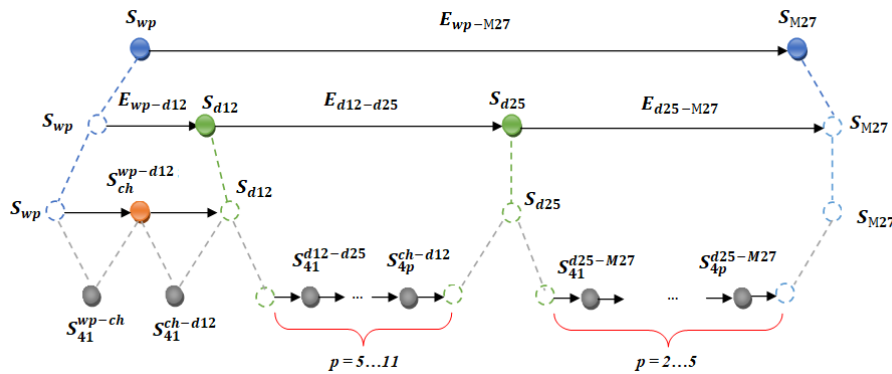


Fig. 7. Structural model of the threaded hole machining process

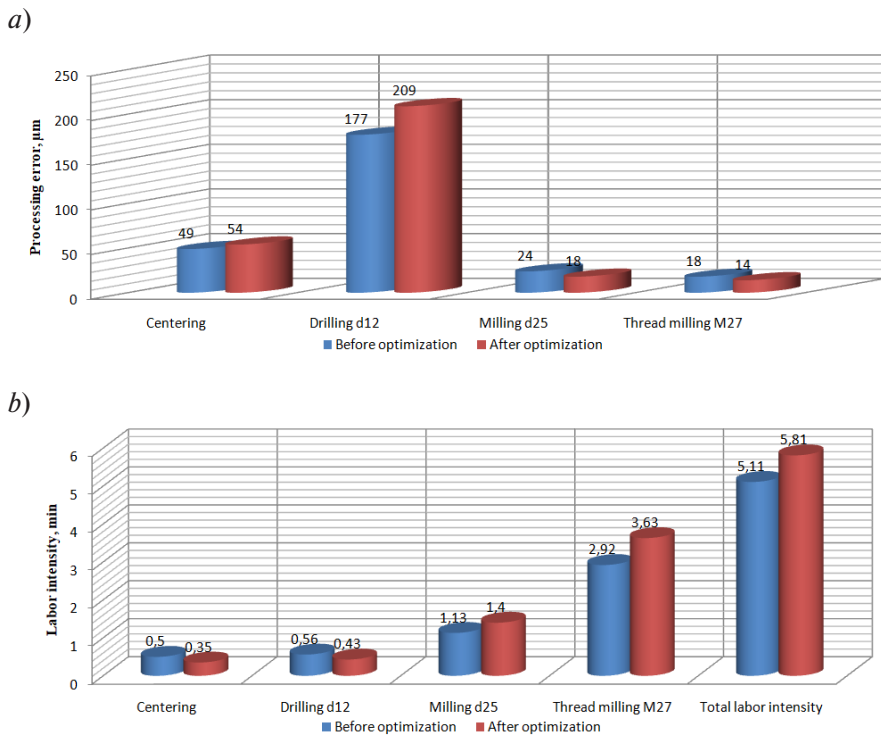


Fig. 8. Benchmarking targets before and after optimization.

- a) comparative analysis of processing errors generated as part of the technological transition;
- b) comparative analysis of labor intensity of technological transitions

The total labor intensity of the process of manufacturing a group of threaded holes increased by 13.69%. The results obtained correspond to the optimization condition, based on which it can be concluded that the goal of the work has been achieved.

The model of optimization of the parameters of the process of forming threaded holes presented in the work can be considered as a basic element of a complex model of optimization of the parameters of the technological process of manufacturing a product. This model can be used to optimize the parameters of the process of forming geometric elements of various types that form the structure of the product. For this, it is necessary to clarify the calculation formulas for a group of target indicators characterizing the accuracy of the parameters describing the configuration of a geometric element.

## REFERENCES

1. **Khrustaleva I.N., Larionova T.A., Lyubomudrov S.A., Chernykh L.G., Stepanov S.N.** Automating production engineering for custom and small-batch production on the basis of simulation modeling. *Journal of Physics: Conference Series*, 2021, Vol. 1753, Art. no. 012047. DOI: 10.1088/1742-6596/1753/1/012047
2. **Kostenko D., Shkodyrev V., Onufriev V.** Solving multicriteria optimization problem for an oil refinery plant. *Proceedings of International Scientific Conference on Telecommunications, Computing and Control*, 2021, Vol. 220, Pp. 131–140. DOI: 10.1007/978-981-33-6632-9\_11
3. **Kostenko D.A., Onufriev V.A., Shkodyrev V.P.** Multicriterial optimisation of rectificational process based on SPEA2 algorithm. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems*, 2019, Vol. 12, No. 2, Pp. 39–49. DOI: 10.18721/JCSTCS.12204
4. **Kostenko D., Arseniev D., Shkodyrev V., Onufriev V.** Pareto optimization in oil refinery. *Data Mining and Big Data (DMBD 2020)*, 2020, Vol. 1234, Pp. 26–33. DOI: 10.1007/978-981-15-7205-0\_3
5. **Yassine H.M., Shkodyrev V.P.** The intelligent control system of optimal oil manufacturing production. *Proceedings of the 2020 3<sup>rd</sup> International Conference on Computational Intelligence and Intelligent Systems*, 2020, Pp. 131–135. DOI: 10.1145/3440840.344084
6. **Ivanov V.K.** Modeling and control integrated system of the enterprise facilities. *Avtomatizatsiia i Sovremennye Tekhnologii [Automation and Modern Technologies]*, 2012, Vol. 7, Pp. 34–39.
7. **Terekhova A.S., Pervov M.L.** Concept of a technique for the optimum technology selection considering provisions of a specific production division. *Vestnik RGATU imeni P.A. Solov'eva [Bulletin of RSATU named after P. A. Solovyov]*, 2018, Vol. 46, No. 3, Pp. 88–93.
8. **Krotov K.V.** A branch and bound algorithm for the optimization of batch scheduling in pipeline systems. *Information and Control Systems*, 2023, Vol. 2, Pp. 15–26. DOI: 10.31799/1684-8853-2023-2-15-26
9. **Ivanov A.A., Bochkarev P.Yu.** Formalising of the optimal scheduling problem in multinomenclature production of machining process. *Vestnik of Saratov State Technical University*, 2011, Vol. 56, No. 2, Pp. 61–69.
10. **Golovina L., Bugaenko M.V., Timokhin D.V., Popova G.I.** Using the Technological cycle closure model for digital modeling of the economics of production processes. *Procedia Computer Science*, 2022, Vol. 213, Pp. 808–815. DOI: 10.1016/j.procs.2022.11.138
11. **Radicić D., Petković S.** Impact of digitalization on technological innovations in small and medium-sized enterprises (SMEs). *Technological Forecasting & Social Change*, 2023, Vol. 191, Art. no. 122474. DOI: 10.1016/j.techfore.2023.122474
12. **Xu Y., Yuan L., Khalfaoui R., Radulescu M., Mallek S., Zhao X.** Making technological innovation greener: Does firm digital transformation work? *Technological Forecasting & Social Change*, 2023, Vol. 197, Art. no. 122928. DOI: 10.1016/j.techfore.2023.122928
13. **Martyn Y., Liaskovska S., Gregus M., Izonin I., Velyka O.** Optimization of technological's processes Industry 4.0 parameters for details manufacturing via stamping: Rules of queuing systems. *Procedia Computer Science*, 2021, Vol. 191, Pp. 290–295. DOI: 10.1016/j.procs.2021.07.036
14. **Qin J., Liu Y., Grosvenor R.** A categorical framework of manufacturing for Industry 4.0 and beyond. *Procedia CIRP*, 2016, Vol. 52, Pp. 173–178. DOI: 10.1016/j.procir.2016.08.005
15. **ElMaraghy H., Schuh G., ElMaraghy W., Piller F., Schönsleben P., Tseng M., Bernard A.** Product variety management. *CIRP Annals*, 2013, Vol. 62, No. 2, Pp. 629–652. DOI: 10.1016/j.cirp.2013.05.007
16. **Tao F., Qi Q., Wang L., Nee A.Y.C.** Digital twins and cyber–physical systems toward smart manufacturing and Industry 4.0: Correlation and comparison. *Engineering*, 2019, Vol. 5, No. 4, Pp. 653–661. DOI: 10.1016/j.eng.2019.01.014
17. **Kang K., Zhong R.Y.** A methodology for production analysis based on the RFID-collected manufacturing big data. *Journal of Manufacturing Systems*, 2023, Vol. 68, Pp. 628–634. DOI: 10.1016/j.jm-sy.2023.05.014



18. Ma S., Zhang Y., Lv J., Ge Y., Yang H., Li L. Big data driven predictive production planning for energy-intensive manufacturing industries. *Energy*, 2020, Vol. 211, Art. no. 118320. DOI: 10.1016/j.energy.2020.118320
19. Zhang Y., Zhang R., Wang Y., Guo H., Zhong R.Y., Qu T., Li Z. Big data driven decision-making for batch-based production systems. *Procedia CIRP*, 2019, Vol. 83, Pp. 814–818. DOI: 10.1016/j.procir.2019.05.023
20. Pan J.Z., Vetere G., Gomez-Perez J.M., Wu H. Exploiting linked data and knowledge graphs in large organisations. Cham: *Springer*, 2017. DOI: 10.1007/978-3-319-45654-6
21. Fensel D., Şimşek U., Angele K., Huaman E., Kärle E., Panasiuk O., Toma I., Umbrich J., Wahler A. How to use a knowledge graph. In: *Knowledge Graphs*. Cham: Springer, 2020. DOI: 10.1007/978-3-030-37439-6\_3
22. Jawad M.S., Dhawale C., Bin Ramli A.A., Mahdin H. Adoption of knowledge-graph best development practices for scalable and optimized manufacturing processes. *MethodsX*, 2023, Vol. 10, Art. no. 102124. DOI: 10.1016/j.mex.2023.102124

#### INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

**Khrustaleva Irina N.**

**Хрусталева Ирина Николаевна**

E-mail: irina.khrustaleva@mail.ru

ORCID: <https://orcid.org/0000-0001-6425-9749>

**Shkodyrev Viacheslav P.**

**Шкодырев Вячеслав Петрович**

E-mail: shkodyrev@mail.ru

**Khokhlovskiy Vladimir N.**

**Хохловский Владимир Николаевич**

E-mail: 78v.kh77@gmail.com

**Chernyh Larisa G.**

**Черных Лариса Георгиевна**

E-mail: 2904180@mail.ru

**Stepanov Sergey N.**

**Степанов Сергей Николаевич**

E-mail: stepanov56@mail.ru

*Submitted: 14.09.2024; Approved: 22.01.2025; Accepted: 18.02.2025.*

*Поступила: 14.09.2024; Одобрена: 22.01.2025; Принята: 18.02.2025.*





Research article

DOI: <https://doi.org/10.18721/JCSTCS.18110>

UDC 622.279.8



## TECHNOLOGICAL PROCESS CONTROL OF OIL GAS ABSORPTION

*D.A. Novak*<sup>1</sup> , *N.A. Nushtaev*<sup>2</sup> ,  
*Yu.N. Kozhubaev*<sup>1</sup>  

<sup>1</sup> Empress Catherine II Saint Petersburg Mining University,  
St. Petersburg, Russian Federation;

<sup>2</sup> Group of Companies “Avtomatika” LLC,  
St. Petersburg, Russian Federation

 [y.n.kozhubaev@gmail.com](mailto:y.n.kozhubaev@gmail.com)

**Abstract.** The main purpose of the paper is to study the optimization of the technological process of oil gas absorption. For this purpose, a complete analysis of the technological process was made with the identification of automation tasks: ensuring a stable temperature of absorbent in the circuit; ensuring a stable temperature of the cooling circuit; filtration of absorbent; ensuring a stable gas pressure in the system; free flow of absorbent between the tanks; accounting of purified gas. The process of selecting equipment for development of a three-level automated control system for oil gas absorption was investigated. The system has 35 discrete signals and 17 analog signals. The measuring devices that should be responsible for collecting and transmitting process information to the logic controller model were selected, the actuators that directly interact with the gas absorption process were selected. Based on the selected sensors, the type of sensor signal, its name and the required number for the possible realization of the automated system were specified. The industrial logic controller, which meets all the requirements of the technological process, was selected. The article provides a rationale for the choice made.

**Keywords:** oil gas absorption, resource processing, technological process, automated control system, SCADA system

**Citation:** Novak D.A., Nushtaev N.A., Kozhubaev Yu.N. Technological process control of oil-based gas absorption. Computing, Telecommunications and Control, 2025, Vol. 18, No. 1, Pp. 130–140. DOI: 10.18721/JCSTCS.18110





Научная статья

DOI: <https://doi.org/10.18721/JCSTCS.18110>

УДК 622.279.8



## УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ МАСЛЯНОЙ АБСОРБЦИИ ГАЗА

Д.А. Новак<sup>1</sup> , Н.А. Нуштаев<sup>2</sup> ,  
Ю.Н. Кожубаев<sup>1</sup>  

<sup>1</sup> Санкт-Петербургский горный университет императрицы Екатерины II,  
Санкт-Петербург, Российская Федерация;

<sup>2</sup> ООО «ГК АВТОМАТИКА», Санкт-Петербург, Российская Федерация

 [y.n.kozhubaev@gmail.com](mailto:y.n.kozhubaev@gmail.com)

**Аннотация.** Основная цель данной статьи заключается в исследовании оптимизации технологического процесса масляной абсорбции газа. Для этого был произведен полный анализ технологического процесса с выявлением задач автоматизации: обеспечение стабильной температуры абсорбента в контуре; обеспечение стабильной температуры холодильного контура; фильтрация абсорбента; обеспечение стабильного давления газа в системе; свободное протекание абсорбента между баками; учет очищенного газа. Был исследован процесс выбора оборудования для реализации трехуровневой автоматизированной системы управления масляной абсорбции газа. Система насчитывает 35 дискретных сигналов и 17 аналоговых. Выбраны измерительные устройства, которые должны отвечать за сбор и передачу информации о процессе к моделируемому логическому контроллеру, выбраны исполнительные устройства, которые непосредственно взаимодействуют на процесс абсорбции газа. Исходя из выбранных датчиков, указан тип сигнала датчика, его наименование и необходимое количество для возможной реализации автоматизированной системы. Выбран промышленный логический контроллер, который отвечает всем требованиям технологического процесса. В статье приведено обоснование проведенного выбора.

**Ключевые слова:** масляная абсорбция газа, переработка ресурсов, технологический процесс, автоматизированная система управления, SCADA-система

**Для цитирования:** Novak D.A., Nushtaev N.A., Kozhubaev Yu.N. Technological process control of oil-based gas absorption // Computing, Telecommunications and Control. 2025. Т. 18, № 1. С. 130–140. DOI: 10.18721/JCSTCS.18110

### Introduction

Oil absorption of gas is one of the most common and used methods of extracting natural gasoline, as it is the most cost-effective and less resource-intensive [1, 2]. The technological process takes place in hard-to-reach areas, in other words, directly in the gas or gasoline field, which increases the risks to human life.

Based on the above, there is a need to introduce an automated process control system (APCS), which increases the quality of purified gas and gasoline, while constantly maintaining the quality of gas, optimizes the operation of the technological process, maintaining the specified values of the parameters of the plants, and provides an opportunity to reduce risks to human life.

Process control systems are divided into automated and automatic. An automated control system implies human participation in the direct control of the technological process, for example, an operator or a dispatcher [3]. An automatic control system consists of a control object and a control device that operate independently [4, 5].

An automated control system is categorized into several levels of automation, where humans occupy the top level. Each automated control system has its own requirements for speed, safety and labor

protection. The main task of an automated system is to collect and process information that helps to optimize the control of the technological process.

The subject of the study is the methods and ways of controlling the technological process of oil gas absorption.

The purpose of the study is the approaches to the automation of the technological process of oil gas absorption. Achieving this goal is ensured by solving a number of problems:

- to conduct a complete analysis of the technological process with the identification of automation tasks;
- to analyze the technical characteristics of equipment for the implementation of automated control system for oil gas absorption;
- to model software and hardware for the process of creating a human-machine interface [6, 7].

### **Analysis of the technological process**

Gas absorption is a method for drying and purifying gas from heavy hydrocarbons using liquid absorbent. The temperature at which the process takes place and the purity of the liquid reagent play an important role in absorption [8].

Absorption is divided into two processes: physical and chemical. The first one is characterized by achieving equilibrium between interacting gas and liquid flows due to diffusion (transfer) of substance from one phase to another. It can also be noted that physical absorption is a reversible process, therefore absorption-desorption plants are used to reduce the cost of absorbents. The desorber is used to regenerate the liquid reactant, whereby the absorbed component is released. While sorption requires high pressure and low temperature, the reverse process (desorption) requires high temperature and low pressure [9].

The absorber is selected according to the following criteria: absorption capacity; dependence of absorption capacity on changes in thermobarometric characteristics; selectivity to the selected substance; cost; possibility of regeneration [10, 11].

The process of oil gas absorption is presented in the form of a process flow diagram (Fig. 1).

After compression, the oil gas from the second and third stages of separation passes the cooler No. 2 and is transported through a pipeline to the absorber. In the process of gas rising up the absorber, the absorbent absorbs heavy hydrocarbons, which flows down the plates from the upper part of the column. The stripped gas first passes through the mist eliminator, where the absorbent carried away by the gas will be captured, then it enters the dehydration unit, and then it is sent to the main gas pipeline or to the consumer [12].

The “fat” absorbent, saturated with heavy hydrocarbon vapors, is discharged through the level regulator (not shown in Fig. 1) from the lower part of the absorber and enters the weathering unit. Since the pressure in this unit is somewhat lower than in absorber, most of the methane and ethane dissolved in the absorbent is released from the “fat” absorbent [13].

From the weathering unit, the “fat” absorbent is first sent to heat exchanger No. 8, where it is preheated with “lean” absorbent coming from the lower part of desorption column, and then to furnace. In the furnace, the “fat” absorbent is heated to a temperature of about 250°C, after which it enters the middle part of desorber, where intensive release of hydrocarbons from the saturated absorbent occurs due to high temperature and significant decrease in pressure reduction. To intensify the desorption process, gas is fed to the lower part of the desorber from the weathering unit, which is preheated in the heat exchanger No. 5 due to the heat of the hot absorbent coming from the lower part of the desorber. Heavy hydrocarbon vapors from the upper part of the desorber together with the weathering gases, are sent to the cooler No. 12, where they are condensed. The condensate together with the weathering gas enters the separator, from where part of the condensate is taken by the pump No. 16 and sent for irrigation to the desorber. The other part enters the tank with unstable condensate. The hot absorbent from the lower part of the desorber, as already mentioned above, passes sequentially through the heat exchangers No. 5 and 8, then enters the cooler No. 7, where its temperature decreases to approximately 38°C.

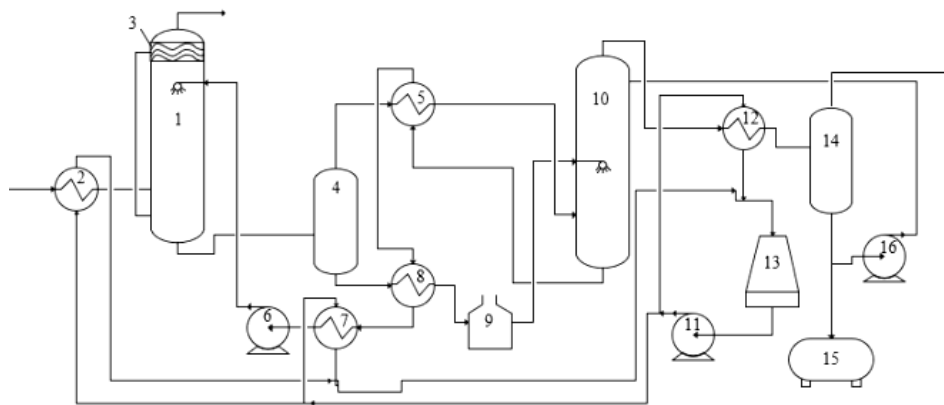


Fig. 1. Process flow diagram of oil gas absorption (compiled by the authors):

1 – absorber; 2, 7, 12 – cooler; 3 – mist eliminator; 4 – weathering unit; 5, 8 – heat exchanger; 9 – furnace; 10 – desorber; 11, 16 – pump; 13 – cooling tower; 14 – separator; 15 – tank with unstable condensate

The cooled absorbent is fed to the suction by the unit No. 6, which pumps it into the upper part of absorber for irrigation, and the cycle of movement of the “lean” absorbent is repeated. Cooling of the hot absorbent in cooler No. 7 and condensation in coolers No. 12 of heavy hydrocarbon vapors released from the “fat” absorbent in desorber is carried out in this plant as a result of closed circulation of water cooled in the cooling tower and pumped by the pump No. 11.

The oil gas absorption plant performs purification and production in two stages, where the first stage is the purification of gas directly from the extraction site, and the second stage involves obtaining gas from the absorbent. The process itself has two circuits, where the main control parameter is temperature. The system controls several parameters, namely: the filling level of the tanks; the pressure in the pipes before and after the tanks; the pressure drop across the filters; the temperature of the absorbent in the tanks [14]. The process is controlled directly by gate valves, starters, and a furnace that heats the absorbent.

For ease of monitoring the process and reducing the risk of an emergency, it is necessary to develop an automatic workstation (AWS) for the operator, who makes adjustments to the process, prevents accidents and monitors technological parameters of the process.

When analyzing the process, seven main tasks of automation of oil gas absorption were identified:

1. Ensuring a stable temperature of the absorbent in the circuit.
2. Ensuring a stable temperature of the cooling circuit.
3. Filtration of the absorbent.
4. Ensuring a stable gas pressure in the system.
5. Free flow of absorbent between tanks.
6. Accounting of the purified gas.

The system has several types of signals: discrete input (DI), discrete output (DO), analog input (AI), analog output (AO). Discrete inputs are signals from limit sensors, differential pressure sensors. Analog input signals are temperature, liquid level, and system pressure sensors. Discrete outputs in the system control the starters and gate valves of the technological process. Analog outputs are required to control the temperature and position of the valve for opening/closing the gas supply to the plant for purification.

From the entire process description, the total number of signals in the system was determined to be 52, including 17 DI, 18 DO, 15 AI, 2 AO.

#### Justification of the selection of technical automation equipment

In order to correctly select the technical automation equipment, let us consider the architecture of automated process control system (Fig. 2).

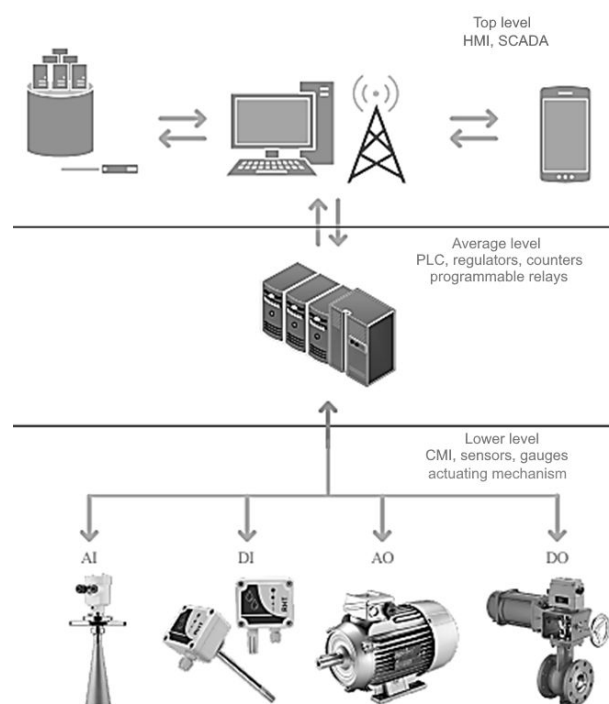


Fig. 2. Architecture of automated process control system (compiled by the authors)

The upper level (SCADA level) of the automated process control system is the level where a person can influence the system [15–17]. This level also implies the presence of a real-time database management system, where the operator's screen displays current information about the system and the interface of interacting with the process (buttons for opening/closing the gate valves, setting the temperature, engine speed etc.) [18–20]. At the moment, such a system can imply remote control at a distance of tens of kilometers [21, 22].

The middle level of the automated process control system includes the functions of measurement, control, protection, blocking, regulation [23, 24]. Here are the algorithms for the operation of the technological process and communication with the upper level, to which all the information collected from sensors is sent. The middle level implies the presence of industrial logic controller, which performs all functions.

At the very bottom of the hierarchy are the actuators that directly affect the technological process (switching on/off electric engines, as well as regulating the position of gate valves, process temperature etc.), and all kinds of sensors that collect data on the process (water temperature, pressure in pipes, engine speed etc.) [25].

The main factor for selecting a differential pressure sensor is the maximum possible differential pressure [26, 27]. A comparative analysis showed the advantage of the IFM PK5522 sensor, since its price can compete with other differential pressure sensors ROSMA RPD-D and Huba Control 652.

Based on the maximum measured temperature plus a 10% reserve, we will select a sensor for the desorber and the furnace, where the maximum temperature reaches 250°C. It should be understood that for both the furnace and the desorber, the measured medium will be the absorbent that flows in the system. Comparison of high-temperature temperature sensors OWEN DTS035M-100P.0,5.400.MG.I and TURCK TTMS-206A-CF-LIUPN-H1140-L150 showed the first of them to be the leader in the price segment.

The choice of temperature sensor for an absorber and cooling tower was between Balluff BFT 6050-DX001-R02A0A-S4, TURCK TP-103A-G1/8-H1141-L013 and OWEN DTPK084-00.250/3K. Despite the more protected versions of Balluff and TURCK, the OWEN sensor is more profitable. When

measuring the temperature of a cooling tower, there is no need for precise temperature measurement, therefore the error fades into the background compared to the price of the sensor.

The gas creates the pressure at the absorber inlet, therefore it is worth using a pressure sensor for natural gas, which can withstand up to 2 MPa, to obtain and record pressure values during overloads. For comparison, the following sensors were selected: OWEN PD100-DI2,5-111-0,5, ADZ-SML-10.0 0/20 BAR, ROSMA RPD-I 0...2.5 MPa 4...20 mA. There is no significant difference in technical characteristics of the sensors, therefore we will focus on the economic component, where ROSMA stands out. The sensor is universal for both gases and liquids, and it can be used for absorbent pressure in pipes.

The studied automated process control system includes two types of level sensors: high-temperature and conventional. The first one is used for the desorber, because the temperature in it is maintained at 250°C, and can even go beyond the limits, in other words, the maximum temperature that the sensor must maintain should not be lower than 300°C. The process takes place under pressure, where the maximum pressure value reaches 2 MPa. In addition to the level gauge, it is also possible to use a level switch, but at least three of them will be needed to know the lower limit, working limit and upper limit of the permissible values of the absorbent level in the desorber. For comparison, three sensors were selected: RIZUR-2030, SIRUR-03B and VEGASWING 66. The first one is the most economical and most functional.

To measure the level in the absorber, weathering and unstable condensate tanks, conventional float sensors are suitable, which must only meet the temperature conditions that do not exceed 70°C, and contact with chemically aggressive liquids (RIZUR-NMT-G sensor is suitable for this).

The flowmeter in the technological process is not subject to certain temperature limitations only because the gas at the absorber inlet and separator outlet has a temperature of  $20 \pm 10^\circ\text{C}$  and its pressure is limited by the outlet of 1.5 MPa. In this case, we select the flow transducer EMIS-VIKHR 200 (EV-200).

Based on the information about the selected sensors, Table 1 with their number and type of signal was created.

Table 1

**Summary table of control and measuring instruments (compiled by the authors)**

|   | Product Name                      | AI | DI | Quantity |
|---|-----------------------------------|----|----|----------|
| 1 | IFM PK5522                        |    | +  | 2        |
| 2 | OWEN DTS035M-100P.0,5.400.MG.I    | +  |    | 2        |
| 3 | OWEN DTPK084-00.250/3K            | +  |    | 2        |
| 4 | ROSMA RPD-I 0...2,5 MPa 4...20 mA | +  |    | 4        |
| 5 | RIZUR-2030                        | +  |    | 1        |
| 6 | RIZUR-NMT-G                       | +  |    | 3        |
| 7 | EV-200                            | +  |    | 2        |

The system uses 16 electric drives with 3-phase power supply of 380 V, a power of 90 W and a discrete input for their control (GZ-OF 200/7M) as actuators (Table 2).

For the automated process control system, it is optimal to use three starters with a rated voltage 380 V and a rated operating current of 25 A, as well as a discrete signal for control (EKF PML-2160DM 25A 230V Basic).

For the correct operation of the automated system, one thyristor power controller is required, applicable to the heating element capable of heating up to the temperature of 250°C (SIPIN W5-TP4V030-24J).

As a programmable logic controller an industrial logic controller that meets all the requirements of the technological process (industrial modular logic controller REGUL R200) was selected. At the same

time, it has a good price-quality ratio in terms of functional and technical capabilities. It was equipped with the following modules: power supply, processor, analog input, discrete input, analog output, discrete output.

Table 2

**Summary table of actuators (compiled by the authors)**

|   | Product Name                  | AO | DI | DO | Quantity |
|---|-------------------------------|----|----|----|----------|
| 1 | GZ-OF-200/7M                  |    |    | +  | 16       |
| 2 | GZ-OF-200/7M (limit switches) |    | +  |    | 16       |
| 3 | EKF Basic PML-2160DM          |    |    | +  | 3        |
| 4 | SIPIN W5-TP4V-030-24J         | +  |    |    | 1        |

### Software tools of the automated control system

For the operation of the automated process control system, two interconnected blocks are required, namely: a control program that will be entered into the industrial logic controller; an operator interface that visualizes all the information about the ongoing process in real time and sends control signals to the industrial logic controller [28–30], for example, starting and stopping the process.

The hardware configuration and adjustment of the Regul family industrial logic controller is carried out in the Epsilon LD software, which supports five languages from the IEC 61131-3 standard list, and also has a number of functional capabilities for programming.

Since it is physically impossible to create a control program, it was decided to create a program in a similar free programming environment CoDeSys V2.3. Therefore, the entire created control program is implemented in a free version, which is identical for loading it into Epsilon LD.

According to the description of the technological process and to its settings for physical values, it is necessary to create a program that performs a number of the following functions:

- starting and stopping the oil gas absorption system;
- controlling the absorbent level in the tanks;
- controlling the temperature in the cooling and absorbent circuit;
- controlling the gas pressure at the process inlet as well as the absorbent pressure in the pipes;
- controlling the filter clogging;
- starting and stopping the cooling circuit;
- selecting the “passive” manual operation mode and “active” automated mode.

For this purpose, a flow chart of the technological process of oil gas absorption was developed and algorithms were created on its basis, which keep the technological parameters of the system within the established limits.

All code was created in the Sequential Function Chart (SFC) language, as the most convenient when comparing the program with the flow chart. SFC is convenient in that several states can be active at once on parallel branches, which monitor the process. Moreover, each of the states can be written in a language convenient to the programmer.

To check the program operation, a visualization was created in the internal CoDeSyS software [31] (Fig. 3).

The purpose of modelling an AWS is to provide convenience in analyzing the technological process and making quick decisions.

The main screen contains dashboards with the main process parameters and navigation between the windows. The second-level windows contain control of individual parts of the technological process and more detailed information; it is also possible to change the process settings.



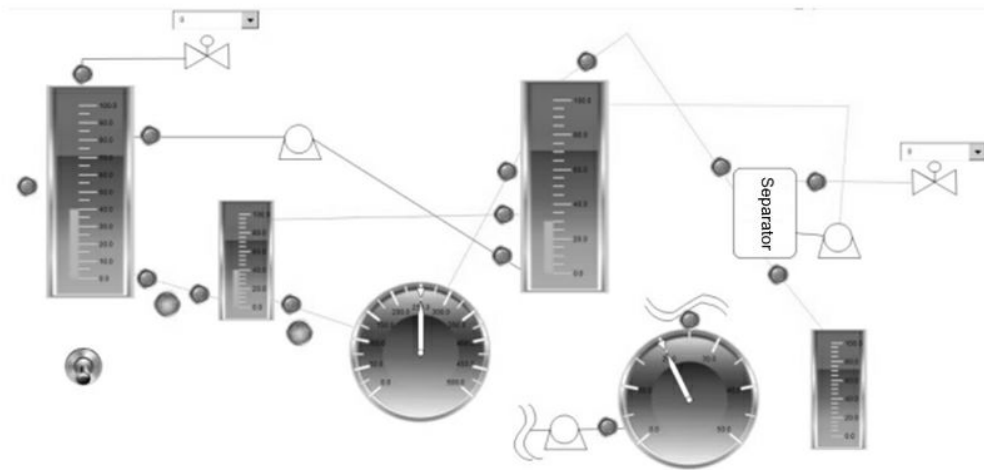


Fig. 3. Visualization of the technological process (compiled by the authors)

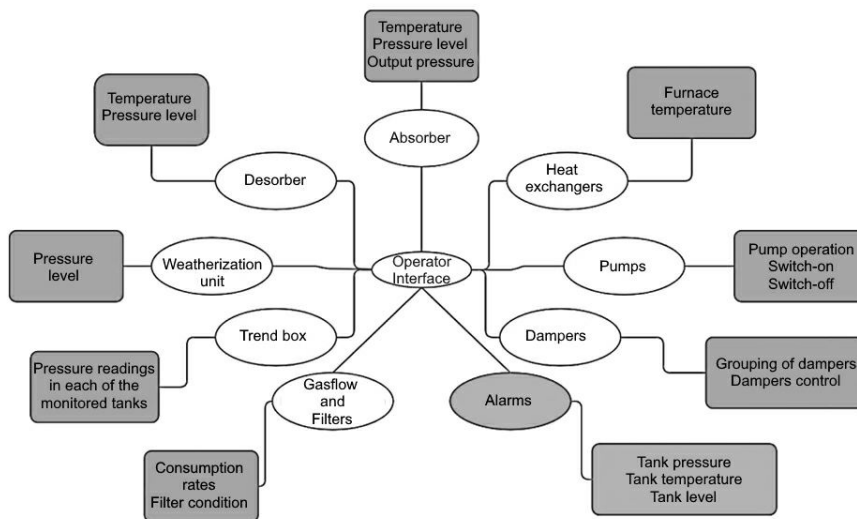


Fig. 4. SCADA screens tree (compiled by authors)

Before we start creating an AWS, let us create a tree of screens in our program, shown in Fig. 4.

Next, an AWS for the operator, who will monitor the process, was developed in SCADA InTouch [32, 33]. All kinds of animations of the program for processing emergencies and operating the system in normal mode were configured. After that, the communication between SCADA system and programmable logic controller program was configured via Matricon Explorer OPC server [34, 35]. Alarms and archiving were configured.

### Conclusions

The study examines and describes the design and development process of the automated control system for the technological process of oil gas absorption. The operation of the aforementioned technological process is described and analyzed as an automation object, with 52 signals involved: 18 DI, 19 DO, 14 AI, 1 AO. The technological diagram of the oil gas absorption plant is presented, where the location of sensors and actuators is indicated.

The following sensors and actuators have been selected for the implementation of the automated process control system: differential pressure sensor on the IFM Electronic PK5522 filter (2 pcs.), high-temperature temperature sensor (OWEN DTS035M-100P.0,5.400.MG.I, 2 pcs.), and temperature sensor (OWEN DTPK084-00.250/3K (2 pcs.), pressure sensor ROSMA RPD-I 0...2.5 MPa 4...20 mA (4 pcs.), level transmitter RIZUR-2030 (1 pc.), level transmitter RIZUR-NMT-G (3 pcs.), flowmeter EV-200 (2 pcs.), quarter-turn electric actuator GZ-OF-200/7M (16 pcs.), starter EKF Basic PML-2160DM (3 pcs.), thyristor power regulator SIPIN W5-TP4V030-24J (1 pc.). The REGUL R200 block-module PLC with the required module configuration was selected: R200 CU 00 031 (1 pc.), R200 AI 04 051 (4 pcs.), R200 AO 02 011 (1 pc.), R200 DI 08 011 (3 pcs.), R200 DO 08 011 (2 pcs.), R200 DO 04 021 (1 pc.).

A program for controlling the technological process of oil gas absorption in SFC language in CoDe-Sys V2.3 software was developed.

An automated operator's workstation in InTouch environment was modelled, thanks to which the operator receives information about the process operation and can perform independent process regulation in the manual mode.

## REFERENCES

1. **Bevzyuk D.V.** Improving the operation of the low-temperature oil absorption unit at the Orenburg gas processing plant. *Oil & Gas Chemistry*, 2022, Vol. 3, Pp. 19–21. DOI: 10.24412/2310-8266-2022-3-19-21
2. **Prokopov A.V., Istomin V.A.** Absorbtsionnye tekhnologii promyslovoi podgotovki gazokondensatnykh gazov [Absorption technologies for industrial treatment of gas condensate gases]. *Vesti Gazovoy Nauki*, 2016, Vol. 26, No. 2, Pp. 165–173.
3. **Abramovich B.N., Bogdanov I.A.** Improving the efficiency of autonomous electrical complexes of oil and gas enterprises. *Journal of Mining Institute*, 2021, Vol. 249, Pp. 408–416. DOI: 10.31897/PMI.2021.3.10
4. **Malarev V., Bogdanov I., Senchilo N.** Algorithm for automatic compensation of voltage dips in power supply of industrial facilities. *Journal of Applied Engineering Science*, 2020, Vol. 18, No. 2, Pp. 173–180. DOI: 10.5937/jaes18-26361
5. **Quiroz Cabascango V.E., Bazhin V.Yu., Martynov S.A., Ojeda Pardo F.R.** Automatic control system for thermal state of reverberatory furnaces in production of nickel alloys. *Metallurgist*, 2022, Vol. 66, Pp. 104–116. DOI: 10.1007/s11015-022-01304-3
6. **Zhukovskiy Y., Koshenkova A., Vorobeva V., Rasputin D., Pozdnyakov R.** Assessment of the impact of technological development and scenario forecasting of the sustainable development of the fuel and energy complex. *Energies*, 2023, Vol. 16, No. 7, Art. no. 3185. DOI: 10.3390/en16073185
7. **Zhukovskiy Yu.L., Kovalchuk M.S., Batueva D.E., Senchilo N.D.** Development of an algorithm for regulating the load schedule of educational institutions based on the forecast of electric consumption within the framework of application of the demand response. *Sustainability*, 2021, Vol. 24, No. 13, Art. no. 13801. DOI: 10.3390/su132413801
8. **Tkachenko A.S.** Primenenie maslianoi i nizkotemperaturnoi absorbtsii dlia ochistki gaza [Application of oil and low temperature absorption for gas purification]. *Alleia Nauki [Science Alley]*, 2019, Vol. 28, No. 1.
9. **Abramkin S.E., Dushin S.E.** Upravlenie tekhnologicheskimi protsessami gazodobyvaiushchikh kompleksov [Management of technological processes of gas production complexes]. *Sistemnyi Sintez i Prikladnaia Sinergetika [System Synthesis and Applied Synergetics]*, 2019, Pp. 430–439. DOI: 10.23683/978-5-9275-3228-5-2019-430-439
10. **Abramkin S.E., Dushin S.E., Pervukhin D.A.** Problems of development of control systems of gas-producing complexes. *Journal of Instrument Engineering*, 2019, Vol. 62, No. 8, Pp. 685–692. DOI: 10.17586/0021-3454-2019-62-8-685-692

11. **Dumont E., Delmas H.** Mass transfer enhancement of gas absorption in oil-in-water systems: A review. *Chemical Engineering and Processing: Process Intensification*, 2003, Vol. 42, No. 6, Pp. 419–438. DOI: 10.1016/S0255-2701(02)00067-3
12. **Ismagilov R.N., Abramkin S.E., Dushin S.E.** Sostoiannie i perspektivy razvitiia avtomatizatsii ustanovok kompleksnoi podgotovki gaza na UNGKM [Status and development prospects of automation of integrated gas treatment plants at the Urengoy oil and gas condensate field]. *Perspektivnye napravleniia razvitiia Urengoiskogo kompleksa [Prospective directions of development of the Urengoy complex]*, 2018, Pp. 271–281.
13. **Parshukov A.N.** Mathematical model of the natural gas absorption technological process. *The Herald of the Siberian State University of Telecommunications and Information Science*, 2022, Vol. 1, Pp. 68–76. DOI: 10.55648/1998-6920-2022-16-1-68-76
14. **Sychev Y.A., Zimin R.Y.** Improving the quality of electricity in the power supply systems of the mineral resource complex with hybrid filter-compensating devices. *Journal of Mining Institute*, 2021, Vol. 247, Pp. 132–140. DOI: 10.31897/PMI.2021.1.14
15. **Yadav G., Paul K.** Architecture and security of SCADA systems: A review. *International Journal of Critical Infrastructure Protection*, 2021, Vol. 34, Art. no. 100433. DOI: 10.1016/j.ijcip.2021.100433
16. **Sheng C., Yao Y., Fu Q., Yang W.** A cyber-physical model for SCADA system and its intrusion detection. *Computer Networks*, 2021, Vol. 185, Art. no. 107677. DOI: 10.1016/j.comnet.2020.107677
17. **Sattari F., Lefsrud L., Kurian D., Macciotta R.** A theoretical framework for data-driven artificial intelligence decision making for enhancing the asset integrity management system in the oil & gas sector. *Journal of Loss Prevention in the Process Industries*, 2022, Vol. 74, Art. no. 104648. DOI: 10.1016/j.jlpi.2021.104648
18. **Šverko M., Galinac Grbac T.** Automated HMI design as a custom feature in industrial SCADA systems. *Procedia Computer Science*, 2024, Vol. 232, Pp. 1789–1798. DOI: 10.1016/j.procs.2024.02.001
19. **Wadinger M., Kvasnica M.** Adaptable and interpretable framework for anomaly detection in SCADA-based industrial systems. *Expert Systems with Applications*, 2024, Vol. 246, Art. no. 123200. DOI: 10.1016/j.eswa.2024.123200
20. **Thepmanee T., Pongswatd S., Asadi F., Ukakimaparn P.** Implementation of control and SCADA system: Case study of Allen Bradley PLC by using WirelessHART to temperature control and device diagnostic. *Energy Reports*, 2022, Vol. 8 (1), Pp. 934–941. DOI: 10.1016/j.egyr.2021.11.163
21. **Beloglazov I.I., Petrov P.A., Bazhin V.Yu.** The concept of digital twins for tech operator training simulator design for mining and processing industry. *Eurasian Mining*, 2020, Vol. 2, Pp. 50–54. DOI: 10.17580/em.2020.02.12
22. **Voytyuk I.N., Kopteva A.V., Skamyin A.N.** “Emergency response plan” automated system for oil production and transportation enterprises. *Journal of Ecological Engineering*, 2021, Vol. 22, No. 1, Pp. 76–82. DOI: 10.12911/22998993/128871
23. **Islamov S., Grigoriev A., Beloglazov I., Savchenkov S., Gudmestad O.T.** Research risk factors in monitoring well drilling — A case study using machine learning methods. *Symmetry*, 2021, Vol. 13, No. 7, Art. no. 1293. DOI: 10.3390/sym13071293
24. **Sultanbekov R., Beloglazov I., Islamov S., Ong M.C.** Exploring of the incompatibility of marine residual fuel: A case study using machine learning methods. *Energies*, 2021, Vol. 24, No. 14, Art. no. 8422. DOI: 10.3390/en14248422
25. **Safiullin R.N., Afanasyev A.S., Rezchenko V.V.** The concept of development of monitoring systems and management of intelligent technical complexes. *Journal of Mining Institute*, 2019, Vol. 237, Pp. 322–330. DOI: 10.31897/PMI.2019.3.322
26. **Urazakov K.R., Belozarov V.V., Latypov B.M.** Study of the dynamics for gas accumulation in the annulus of production wells. *Journal of Mining Institute*, 2021, Vol. 250, Pp. 606–614. DOI: 10.31897/PMI.2021.4.14
27. **Belousov A.E., Ovchinnikov E.S.** Mathematical modeling of the operation of an expander-generator pressure regulator in non-stationary conditions of small gas pressure reduction stations. *Mathematics*, 2022, Vol. 10, No. 3, Art. no. 393. DOI: 10.3390/math10030393

28. **Erushin E.Y., Kostyukova N.Y., Boyko A.A., Miroshnichenko I.B., Sherstov I.V., Kolker D.B.** Widely tunable automatic system for multiple gas detection. *2022 IEEE 23<sup>rd</sup> International Conference of Young Professionals in Electron Devices and Materials (EDM)*, 2022, Pp. 356–359. DOI: 10.1109/EDM55285.2022.9855152
29. **Figueiredo J., Ayala Botto M., Rijo M.** SCADA system with predictive controller applied to irrigation canals. *Control Engineering Practice*, 2013, Vol. 21, No. 6, Pp. 870–886. DOI: 10.1016/j.conengprac.2013.01.008
30. **Celentano L.** A fast design technique for robust industrial controllers. *Journal of the Franklin Institute*, 2023, Vol. 360, No. 8, Pp. 5689–5727. DOI: 10.1016/j.jfranklin.2023.03.033
31. **Simon H., Triefenbach L., Kowalewski S.** Structural concolic testing for sequential function chart. *IFAC-PapersOnLine*, 2018, Vol. 51, No. 7, Pp. 422–427. DOI: 10.1016/j.ifacol.2018.06.335
32. **Alanazi M., Mahmood A., Chowdhury M.J.M.** SCADA vulnerabilities and attacks: A review of the state of the art and open issues. *Computers & Security*, 2023, Vol. 125, Art. no. 103028. DOI: 10.1016/j.cose.2022.103028
33. **Zhu Q., Zhang G., Luo X., Gan C.** An industrial virus propagation model based on SCADA system. *Information Sciences*, 2023, Vol. 630, Pp. 546–566. DOI: 10.1016/j.ins.2022.12.119
34. **Neis P., Wehrmeister M.A., Mendes M.F., Pesente J.R.** Applying a model-driven approach to the development of power plant SCADA/EMS software. *International Journal of Electrical Power & Energy Systems*, 2023, Vol. 153, Art. no. 109336. DOI: 10.1016/j.ijepes.2023.109336
35. **Rodríguez F., Guzmán J.L., Castilla M., Sánchez-Molina J.A., Berenguel M.** A proposal for teaching SCADA systems using Virtual Industrial Plants in Engineering Education. *IFAC-PapersOnLine*, 2016, Vol. 49, No. 6, Pp. 138–143. DOI: 10.1016/j.ifacol.2016.07.167

#### INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

**Novak Diana A.**

**Новак Диана Александровна**

E-mail: novak\_da@pers.spmi.ru

ORCID: <https://orcid.org/0009-0000-2651-899X>

**Nushtaev Nikita A.**

**Нуштаев Никита Андреевич**

E-mail: um-urii@mail.ru

ORCID: <https://orcid.org/0009-0002-7544-9155>

**Kozhubaev Yuriy N.**

**Кожубаев Юрий Нурғалиевич**

E-mail: y.n.kozhubaev@gmail.com

ORCID: <https://orcid.org/0009-0006-1822-7117>

*Submitted: 27.11.2024; Approved: 12.02.2025; Accepted: 13.03.2025.*

*Поступила: 27.11.2024; Одобрена: 12.02.2025; Принята: 13.03.2025.*