DISCRETE AND CONTINUOUS MODELS AND APPLIED COMPUTATIONAL SCIENCE

Volume 33 Number 2 (2025)

Founded in 1993

Founder: PEOPLES' FRIENDSHIP UNIVERSITY OF RUSSIA NAMED AFTER PATRICE LUMUMBA

DOI: 10.22363/2658-4670-2025-33-2

Edition registered by the Federal Service for Supervision of Communications,
Information Technology and Mass Media
Registration Certificate: ПИ № ФС 77-76317, 19.07.2019

ISSN 2658-7149 (Online); 2658-4670 (Print)
4 issues per year.
Language: English.

Publisher

Peoples' Friendship University of Russia named after Patrice Lumumba (RUDN University).

Indexed by

- Scopus (https://www.scopus.com),
- Ulrich's Periodicals Directory (http://www.ulrichsweb.com),
- Directory of Open Access Journals (DOAJ) (https://doaj.org),
- Russian Index of Science Citation (https://elibrary.ru),
- CyberLeninka (https://cyberleninka.ru).

Aim and Scope

Discrete and Continuous Models and Applied Computational Science arose in 2019 as a continuation of RUDN Journal of Mathematics, Information Sciences and Physics. RUDN Journal of Mathematics, Information Sciences and Physics arose in 2006 as a merger and continuation of the series "Physics", "Mathematics", "Applied Mathematics and Computer Science", "Applied Mathematics and Computer Mathematics".

Discussed issues affecting modern problems of physics, mathematics, queuing theory, the Teletraffic theory, computer science, software and databases development.

It's an international journal regarding both the editorial board and contributing authors as well as research and topics of publications. Its authors are leading researchers possessing PhD and PhDr degrees, and PhD and MA students from Russia and abroad. Articles are indexed in the Russian and foreign databases. Each paper is reviewed by at least two reviewers, the composition of which includes PhDs, are well known in their circles. Author's part of the magazine includes both young scientists, graduate students and talented students, who publish their works, and famous giants of world science.

The Journal is published in accordance with the policies of COPE (Committee on Publication Ethics).

The editors are open to thematic issue initiatives with guest editors. Further information regarding notes for contributors, subscription, and back volumes is available at http://journals.rudn.ru/miph

E-mail: miphj@rudn.ru, dcm@sci.pfu.edu.ru

Editorial board

Computer Design: *Anna V. Korolkova, Dmitry S. Kulyabov*

English Text Editors: *Nikolay E. Nikolaev, Ivan S. Zaryadov, Konstantin P. Lovetskiy*

Address of editorial board:
3 Ordzhonikidze St, 115419 Moscow, Russia
+7 (495) 955-07-16, e-mail: publishing@rudn.ru

Editorial office:
+7 (495) 952-02-50, miphj@rudn.ru, dcm@sci.pfu.edu.ru,
site: http://journals.rudn.ru/miph

Paper size 70×108/16. Offset paper. Offset printing. Typeface “Adobe Source”.
Conventional printed sheet 9.28. Printing run 500 copies. Open price. The order 761.
PEOPLES’ FRIENDSHIP UNIVERSITY OF RUSSIA NAMED AFTER PATRICE LUMUMBA
6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

Printed at RUDN Publishing House:
3 Ordzhonikidze St, Moscow, 115419, Russian Federation,
+7 (495) 952-04-41; e-mail: publishing@rudn.ru

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2)
http://journals.rudn.ru/miph

Contents

Editorial

Computer science

Modeling and Simulation

Letters

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 125–129
http://journals.rudn.ru/miph

Editorial

DOI: 10.22363/2658-4670-2025-33-2-125-129

EDN: BLVYBG

Using generative artificial intelligence

Dmitry S. Kulyabov[1, 2], Leonid A. Sevastianov[1, 2]

[1] RUDN University, 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

[2] Joint Institute for Nuclear Research, 6 Joliot-Curie St, Dubna, 141980, Russian Federation

Abstract. The use of generative artificial intelligence (GenAI) in scientific publications requires strict guidelines to ensure transparency, credibility, and ethics of research.

Key words and phrases: generative artificial intelligence, GenAI, taxonomy

For citation: Kulyabov, D. S., Sevastianov, L. A. Using generative artificial intelligence. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 125–129. doi: 10.22363/2658-4670-2025-33-2-125-129. edn: BLVYBG (2025).

1. Introduction

This paper should be considered as a supplement to the description of author's ethics for the journal [1].

2. Key rules for using artificial intelligence

- Disclosure of AI use. In the methods section, specify the details of the AI use: what tools were used, at what stages of the work (text generation, data analysis, visualization), date and version of the model.
- Prohibition on indicating AI as an author. AI cannot be indicated as an author or co-author, since it does not have the ability to accept responsibility for the content of the work, declare conflicts of interest, or sign license agreements.
- Citation requirements When using AI to generate text or images, you must:
  - Cite the tool used.
  - Provide the full AI output as additional material for audit.
- Restrictions on using AI for images. It is prohibited to publish images that are fully or partially created using AI, due to the impossibility of verifying the sources of data.

3. Ethical aspects of the use of artificial intelligence

- Fact-checking. Authors are required to manually check the accuracy of information generated by AI, including citations and statistics.
- Data protection and privacy. There is a risk of data leakage when using cloud services. Authors should avoid entering confidential information and use local solutions for sensitive data.
- Responsibility for plagiarism. AI-generated text may contain fragments from training data without specifying sources.

4. Recommendations for authors

- Use AI as a tool, not as a replacement for critical thinking.
- Avoid completely delegating text generation or results interpretation to AI.
- Keep logs of interactions with AI for audit purposes.

5. Acceptable and unacceptable uses of artificial intelligence

Examples of acceptable use cases for artificial intelligence [2].

- Text Generation
    - Acceptable: Artificial intelligence can help with writer's block or definitions; its use should be based on human critical thinking and judgment.
    - Unacceptable: Using artificial intelligence to generate new text is ethically unacceptable. Academic texts should be original and attributed to human authors.
- Text Translation
    - Acceptable: Using artificial intelligence to translate text from another language into English and vice versa.
    - Unacceptable: Using artificial intelligence to translate previously published work into English without subsequent editing (ethical concerns about self-plagiarism).
- Text editing
    - Acceptable: Artificial intelligence can be used to identify and correct grammatical errors, typos, and other writing errors.
    - Unacceptable: Correcting entire paragraphs using artificial intelligence without human critical thinking and judgment.
- Image creation
    - Acceptable: Artificial intelligence can create images based on text cues. This seems acceptable only if the topic of the article is automatic image generation.
    - Unacceptable: Using artificial intelligence to create visual aids such as charts, graphs, illustrations.
- Paraphrasing
    - Acceptable: Artificial intelligence can help you rephrase sentences or paragraphs to make them clearer.
    - Unacceptable: Paraphrasing without human supervision.

6. Declaration on the use of generative artificial intelligence

- All authors are responsible for providing an accurate description of the use of generative AI.
- A declaration of the use of generative AI must be provided in the article.
- Authors are required to declare and describe in detail the specific contribution of any generative AI tools and services used in the preparation of their work.
- This disclosure must include the following:
    - tools and services: a complete list of all generative AI tools and services used;
    - tool contributions: an accurate description of the specific contribution from each generative AI tool (as per the generative AI use taxonomy).

There is no generally accepted taxonomy of generative artificial intelligence usage. We will use the taxonomy of generative artificial intelligence usage contributions based on [3]. It is also recommended to pay attention to the short taxonomy [4].

7. GenAI usage taxonomy

Drafting content AI can help you write different sections of your paper, such as introductions, literature reviews, or methodology descriptions.

Generate images AI can help you generate images for your paper.

Text translation AI can help you in translating your work or reaching a broader audience.

Generate literature review AI can help you drafting a literature review section starting from a set of relevant papers.

Paraphrase and reword AI can help you express ideas in different ways, ensuring clarity and conciseness.

Improve writing style AI can offer suggestions for sentence structure, word choice, and overall flow.

Abstract drafting AI can draft a concise abstract that captures the gist of your research.

Grammar and spelling check AI can catch errors that you might have missed.

Plagiarism detection AI can help you identify potential plagiarism issues in your own writing.

Citation management AI can help format citations and references according to specific styles.

Formatting assistance AI can ensure your paper adheres to specific formatting guidelines required by journals or institutions.

Peer review simulation AI can simulate peer review by providing feedback on the strengths and weaknesses of your paper.

Content enhancement AI can suggest additional content or research that could strengthen your arguments.

8. Example declaration of use of generative artificial intelligence

The declaration may look like this.

If no artificial intelligence tools was used:

> The authors have not employed any Generative AI tools.

If artificial intelligence tools were used:

> During the preparation of this work, the authors used ChatGPT-4 and Grammarly in order to: grammar and spelling check. Further, the authors used X-AI-IMG for figures 3 and 4 in order to: generate images. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

The source text uses the following LaTeXenvironment:

```
\begin{aideclaration}
...
\end{aideclaration}
```

Author Contributions: The contributions of the authors are equal. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new data were created or analysed during this study. Data sharing is not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Declaration on Generative AI: The authors have not employed any Generative AI tools.

References

1. Kulyabov, D. S. & Sevastianov, L. A. Author's ethics. *Discrete and Continuous Models and Applied Computational Science* **32,** 135–139. doi:10.22363/2658-4670-2024-32-2-135–139 (2024).
2. Salatino, A., Fornari, F. & Zdravkovic, J. *CEUR-WS Policy on AI-Assisting Tools* https://ceur-ws.org/GenAI/Policy.html.
3. Salatino, A., Fornari, F. & Zdravkovic, J. *GenAI Usage Taxonomy* https://ceur-ws.org/GenAI/Taxonomy.html.
4. Hyde, A., Chodacki, J. & Shannon, P. *An Initial Scholarly AI Taxonomy* Apr. 2023. doi:10.54900/6p6re-xyj61.

Information about the authors

Dmitry S. Kulyabov—Professor, Doctor of Sciences in Physics and Mathematics, Professor of Department of Probability Theory and Cyber Security of RUDN University; Senior Researcher of Laboratory of Information Technologies, Joint Institute for Nuclear Research (e-mail: kulyabov-ds@rudn.ru, ORCID: 0000-0002-0877-7063, ResearcherID: I-3183-2013, Scopus Author ID: 35194130800)

Leonid A. Sevastianov—Professor, Doctor of Sciences in Physics and Mathematics, Professor of Department of Computational Mathematics and Artificial Intelligence of RUDN University (e-mail: sevastianov-la@rudn.ru, ORCID: 0000-0002-1856-4643, ResearcherID: B-8497-2016, Scopus Author ID: 8783969400)

DOI: 10.22363/2658-4670-2025-33-2-125-129 EDN: BLVYBG

Использование генеративного искусственного интеллекта

Д. С. Кулябов[1, 2], Л. А. Севастьянов[1, 2]

[1] Российский университет дружбы народов, ул. Миклухо-Маклая, д. 6, Москва, 117198, Российская Федерация

[2] Объединённый институт ядерных исследований, ул. Жолио-Кюри, д. 6, Дубна, 141980, Российская Федерация

Аннотация. Использование генеративного искусственного интеллекта в научных публикациях требует соблюдения строгих правил для обеспечения прозрачности, достоверности и этичности исследований.

Ключевые слова: генеративный искусственный интеллект, таксономия

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 130–143
http://journals.rudn.ru/miph

Research article

UDC 519.[176+178+16]:004.4
PACS 02.10.Ox 89.70.Eg 07.05.Tp
DOI: 10.22363/2658-4670-2025-33-2-130-143

EDN: BMIMTX

A technique of algorithms construction for solving a correlation clustering problem

Ellada I. Ibragimova, Daria V. Semenova, Aleksandr A. Soldatenko

Siberian Federal University, 79 Svobodny pr, Krasnoyarsk, 660041, Russian Federation

(received: January 24, 2025; revised: February 14, 2025; accepted: March 1, 2025)

Abstract. We propose a construction method for network structure based algorithms (NS-algorithms), aimed at solving the correlation clustering problem (CCP) specifically for signed networks. Our model assumes an undirected, unweighted simple signed graph. This problem is considered in optimization form with the error functional as a linear combination of inter-cluster and intra-cluster errors. It is known that this formulation of the problem is NP-hard. The technique of NS-algorithms constructing is grounded on the system approach presented in the form of a general scheme. The proposed scheme comprises six interconnected blocks, each corresponding to a stage in addressing the CCP solution. The main idea of the technique is to combine modules representing each block of the scheme. The proposed approach has been realized as a software package. The paper presents a model NS-algorithm constructed using the proposed technique. To evaluate its performance, computational experiments utilizing synthetic datasets are conducted, comparing the new algorithm against existing methods.

Key words and phrases: signed network, algorithm systematization, network structure based approach, correlation clustering problem

For citation: Ibragimova, E. I., Semenova, D. V., Soldatenko, A. A. A technique of algorithms construction for solving a correlation clustering problem. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 130–143. doi: 10.22363/2658-4670-2025-33-2-130-143. edn: BMIMTX (2025).

1. Introduction

A network is a collection of real objects of an arbitrary nature, in which some pairs of these objects are connected. The network nodes can be objects from social, biological or telecommunications systems [1–3]. Signed networks are defined as an extension of networks that includes additional information about positive and negative links.

The following tasks are related to the analysis and modeling of signed networks: balance recognition; searching for a subnetwork with certain properties; clustering tasks; generating signed network with given properties; signed network mining [4–7].

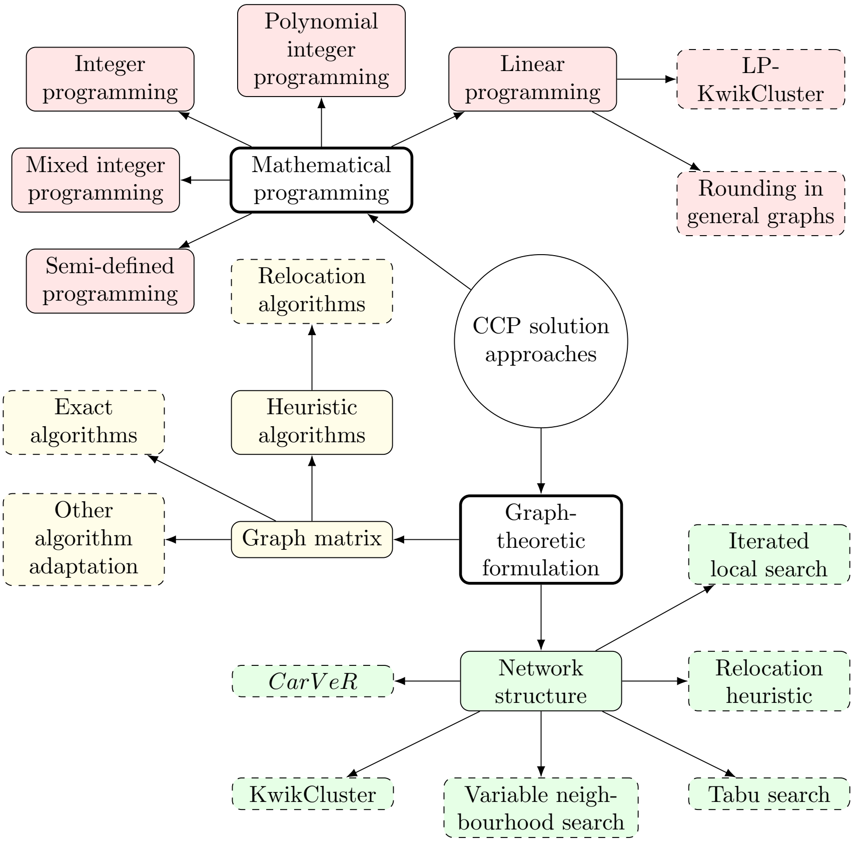


Figure 1. CCP solution approaches

Our research focuses on the Correlation Clustering problem (CCP). A historical overview of the problem is given in [8], an overview of existing solution methods is given in [9]. There are two main approaches to solving the correlation clustering problem [9]. Figure 1 shows the approaches to solving the correlation clustering problem. The first approach is based on the representing of the correlation clustering problem as a mathematical programming problem (e.g., linear, integer, semi-definite, etc.). [9, 10]. The second approach is based on a graph-theoretic representation of the signed network. There are two types of algorithm: those based on the graph's structure (NS-algorithms) [9, 11–13]; and those based on the graph's matrix representation [9, 14]. The following results refer to the former, i. e. NS-algorithms.

The definitions and the formulation of the problem of correlation clustering of signed networks necessary for further exposition are given in section 2. Section 3.1 describes the general scheme of algorithm construction and analysis, as well as possible variants of blocks. Section 3.2 describes the program complex for solving the CCP problem and investigates a new algorithm *CarVeR*. Section 3.3 presents the results of computational experiments for the algorithm built according to the scheme are given.

2. Theoretical Basis

This paper investigates signed networks $\Sigma = (G, \sigma)$, where $G = (V, E)$ is undirected, unweighted graph, with a vertex set V, $|V| = n \geqslant 2$, and a set of edges E, $|E| = m \geqslant 1$. We will consider simple graphs, i.e., graphs without multiple edges and loops. In the graph G each edge is uniquely represented by an unordered pair $e = (u, v)$, where $e \in E$, $u, v \in V$. In this case, edge e is said to be incident to vertices u and v, which are adjacent.

On edges $(u, v) \in E$ of the graph G the sign function $\sigma : E \to \{+, -\}$ is given, which generates a partition of the set of edges of the graph $E = E^+ \cup E^-$, where E^+ is a set of positive edges, E^- is a set of negative edges.

A signed network is k-balanced, if its vertex set can be divided into k pairwise non-intersecting non-empty clusters such that all positive edges lie within clusters and all negative edges lie between them [15].

We denote the system of sets that form a partition of the vertex set V into k subsets as follows:

$$C = \left\{ C_i \subseteq V : \bigcup_{i=1}^{k} C_i = V, C_i \cap C_j = \varnothing, i \neq j;\ i = \overline{1,k} \right\}.$$

It is known that the k-balance property may not be fulfilled for an arbitrary sign network. In this case, it is interesting to find a partition of the vertex set of the network such that changing the sign of the minimum number of edges results in a k-balanced network. This problem is considered as a graph clustering problem with a special kind of error functional. The elements of the partition $C_i \in C$ will be called clusters.

We define the positive error $P(C)$ of a partition (2) as the number of positive edges between subsets $C_1, \dots, C_k$. Note that $P(C)$ is the inter-cluster error calculated by the formula

$$P(C) = \sum_{i=1}^{k} \sum_{u \in C_i} \sum_{v \in V \setminus C_i} [(u, v) \in E^+],$$

where $[\cdot]$ is Iverson bracket [16].

We define the negative error $N(C)$ as the number of negative edges in the subsets of partition (2). Negative error is the intra-cluster error calculated by the formula

$$N(C) = \sum_{i=1}^{k} \sum_{\{u,v\} \subseteq C_i} [(u, v) \in E^-].$$

In [17] the authors propose representing the total error as a convex combination of positive and negative errors depending on the parameter $\alpha \in [0, 1]$:

$$Q_\alpha(C) = \alpha N(C) + (1 - \alpha) P(C).$$

Note that the error functional (2) always satisfies the following inequality

$$0 \leq Q_\alpha \leq \alpha |E^-| + (1 - \alpha) |E^+|.$$

In this paper, the signed network clustering problem is considered in the following formulation [17].

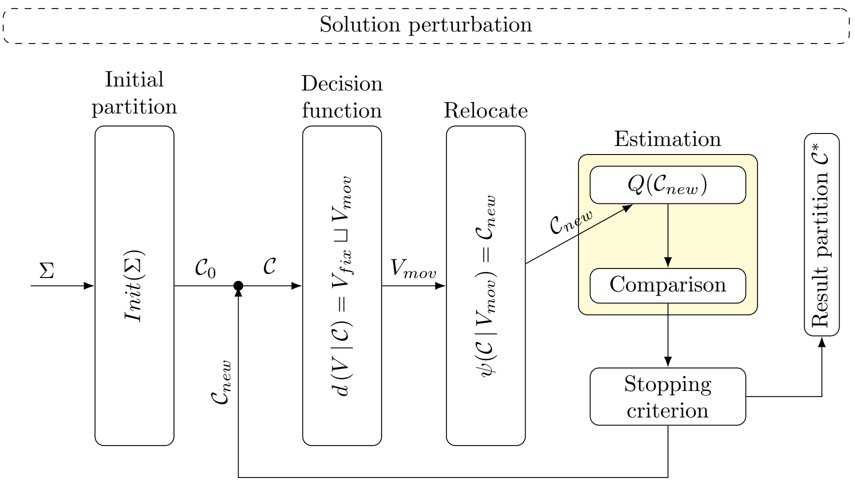


Figure 2. General NS-algorithms scheme

2.1. Correlation Clustering problem (CCP)

Condition: the signed network $\Sigma = (G, \sigma)$ is given, where $G = (V, E)$ is an undirected graph, $n = |V| \geqslant 2$, $m = |E| \geqslant 1$.

Question: for given $\alpha \in [0, 1]$ is required to find a partition $\mathcal{C}$ of vertex set V of signed network Σ with minimum total error $Q_\alpha(\mathcal{C})$.

As shown in [18], the problem of correlation clustering of signed network with an error functional in the form of (2) at $\alpha = 0.5$ is NP-hard.

The solution of the problem is the set of clusters $\mathcal{C}^*$ that provides the minimum of the error functional (2):

$$\mathcal{C}^* = \arg\min_{\mathcal{C}\in\Phi}\big[\alpha N(\mathcal{C}) + (1-\alpha)P(\mathcal{C})\big],$$

where $\Phi = \bigcup_{k=1}^{n} \Phi_k$ is a set of all possible partitions V, Φ_k is a set of partitions into k subsets. The power of the solution space Φ is equal to Bell's number B_n. It should be noted that the solution (2.1) may not be single.

3. Results

3.1. A technique for constructing algorithms for solving CCP

To solve the CCP, a systematic approach involving the construction and analysis of algorithms based on network structure (NS-algorithms) was proposed in [19]. This approach was realized in the form of a general scheme, which is presented in Fig. 2.

The scheme consists of six blocks that are executed sequentially until the stopping condition is satisfied. In [19], five known and two our NS-algorithms for solving the CCP were analyzed using this scheme.

This paper proposes a detailed breakdown of the blocks in the general scheme. The modules that mimic the behavior of known NS-algorithms are highlighted for each block. A brief description of each block and its corresponding modules is then presented. For the modules that can be evaluated, an upper bound on the execution time for a naive implementation is given.

It should be noted that our systematic approach enables us to combine different modules to create new algorithms, and the topology of the overall scheme simplifies the analysis and proof of the computational complexity of the constructed algorithms.

The "Initial partition" block contains the function $Init(\Sigma)$, which performs the initial partitioning of the vertex set V of the signed network Σ according to a certain rule. This block is usually executed once at the start of the algorithm and forms the initial partitioning $C_0 \in \Phi$ in the form (2).

The following modules are highlighted for this block.

- *Trivial partition.* The original graph is randomly partitioned into k non-overlapping subsets. The subsets number k can be either given initially or generated, and $k < |V|$. The module is executed in time $\mathcal{O}(|V|)$.
- *Prepared partition.* The result of another algorithm or an input file is used as the initial partitioning.
- *Connectivity components.* The clusters are the connectivity components of the graph $\Sigma^+ = (V, E^+)$. The module is executed in time $\mathcal{O}(|V| + |E^+|)$.
- *Ranking and cut.* The score $I(v)$ is evaluated for each vertex v, and the cluster to which it should belong is determined. Then, for each selected cluster, all the vertices are ordered according to their score. A random vertex set among the first $\alpha < |V|$ vertices, where α is an integer is selected from the ordered vertex set. This process is repeated until all the vertices have been assigned to a cluster. The module can be completed in time $\mathcal{O}(|V|^2 \cdot \log |V| \cdot |E|)$.
- *KwikCluster.* Partitioning according to the *KwikCluster* algorithm [11]. At each step, a random vertex is selected to form a cluster with all positively connected vertices. This process is repeated until all the vertices have been assigned to a cluster. The module is executed in time $\mathcal{O}(|V| \cdot |E|)$.

The "Decision function" block is preparatory for the "Relocation" block. The decision function divides the set of vertices V taking into account the current partitioning C into two non-intersecting subsets:

$$d(V \mid C) = V_{fix} \sqcup V_{mov},$$

where V_{fix} is the set of vertices blocked for moving between clusters, and V_{mov} is the set of vertices allowed to move between clusters.

The following modules are allocated to this block.

- *Trivial function.* The solving function returns $V_{mov} = V$. The execution time of the module is $\mathcal{O}(1)$.
- *Tabu.* While the algorithm is running, the vertices that were affected are memorized, and are added to the tabu list V_{tabu} for a given number of iterations or until the end of the algorithm. The solving function returns $V_{mov} = V \setminus V_{tabu}$. The execution time of the module is $\mathcal{O}(|V|)$.
- *Negative error.* All vertices that contribute a negative error in (2) are returning. The execution time of the module is $\mathcal{O}(|V| \cdot |E|)$.
- *Potential function.* This block returns the vertices for which the value of some potential function satisfies the given condition.

The "Relocation" block contains the function $\psi(\mathcal{C} \mid V_{mov})$, which is responsible for moving vertices from the set V_{mov} between clusters of the current partition $\mathcal{C}$ and thus forming a new partition $\mathcal{C}_{new} \in \Phi$. It is worth noting that vertices can move not only between existing clusters, but also form new clusters.

The following modules are highlighted for this block. All the described strategies allow for the formation of new clusters.

- *By contribution*. The vertex from the set V_{mov} that makes the maximum contribution to the negative error is selected. A new cluster is chosen so that the target function (2) is minimized when a vertex joins it. The execution time of the module is $\mathcal{O}(|V|^2 \cdot |E|)$.
- *By objective*. A vertex from the set V_{mov} is selected and moved to another existing cluster so that the target function (2) is minimized. For this purpose, all possible vertex moves from the set of allowable vertices are evaluated. The execution time of the module is $\mathcal{O}\left(|V|^2 \cdot |E|\right)$.
- *Relocation of r vertices*. We select $1 \leq r \leq r_{max}$ vertices from the set V_{mov}, which are simultaneously moved to other existing clusters so that the target function (2) is minimized. In this case, r_{max} is set initially. The module is executed in the time $\mathcal{O}\left(|V|^{2r} \cdot |E|\right)$.

The "Evaluation" block consists of two stages. At the first stage, the error functional $Q(\mathcal{C}_{new})$ is calculated.

The second stage compares the current partition with previously found partitions by the value of the error functional. This block will also be present in the algorithm that finds several partitions sequentially or in parallel.

The "Stopping criterion" block allows to exclude the search over the whole Φ set. The stopping criterion can be: time, number of iterations, value of the error functional, residual of the error functional, etc.

The following modules are highlighted for this block:

- *Stop by time*. When the algorithm is started, a running time limit is set. Once the algorithm exceeds the specified time, the process stops and the current partitioning is the result.
- *Stop by iteration*. The number of iterations of the main loop of the algorithm is set when it starts, and the result is the partitioning obtained at the final iteration.
- *Stop by vertices*. The algorithm terminates when the tabu list contains all vertices. The result of the algorithm is the partition obtained at the last iteration.

The "Perturbation" block aims at getting out of the error functional local minimum by shuffling the vertices of the current partition, and the method of shuffling may determine the main idea of the algorithm. This block may follow after any other block of the overall scheme and may be repeated many times.

For this block, only the "Vertex shuffling" module is currently provided. A given number of vertices are moved between clusters randomly. The execution time of the module is $\mathcal{O}(|V|)$.

The table 1 represents known algorithms according to the modules described above. Comparisons of the first seven algorithms on synthetic and real data were presented in the papers [19–21].

The proposed technique allows us to construct both deterministic algorithms and algorithms with an element of random solution search. Analysis of existing NS-algorithms (see Table 1) revealed that elements of randomness appear in the "Initial partition" and "perturbation" blocks. In particular, the algorithms *CarVeR*, *SGClust*$_\alpha$, *Tabu Search* are deterministic, i.e., the same input leads to the same output. In the algorithms *Relocation heuristic*, *Variable neighborhood search* and *Iterated local search*, randomness is present at the initial partitioning stage and in the process of perturbing the solution to escape from local minima. The theoretical derivations and possible limitations of these algorithms are presented in the corresponding articles of the authors presented in the table 1.

The proposed system approach provides the possibility of combining different modules to develop new algorithms. In the last row of the 1 table, a representation of the new deterministic algorithm *Custom* is given. The algorithm *Custom* consists of the following modules according to the scheme in Fig. 2:

Table 1

Representation of algorithms for solving the CCP by scheme phases 2

Algorithm	Initial partition	Decision function	Relocation	Stopping Criterion	Perturbation
Relocation heuristic (*RH*) [17]	*Trivial partition*	*Trivial function*	*By objective*	*Stop by time*	–
Tabu search [12]	*Prepared partition*	*Tabu*	*By objective*	*Stop by time*	–
Variable neighborhood search [12]	*Prepared partition*	*Trivial function*	*By objective*	*Stop by time*	+
KwikCluster [11]	*KwikCluster*	–	–	*Stop by vertices*	–
Iterated local search (*ILS*) [13]	*Ranking and cut*	*Trivial function*	*Relocation of r vertices*	*Stop by iteration, time*	+
SGClust$_\alpha$ [21]	*Connectivity components*	*Negative error*	*By contribution*	*Stop by vertices*	–
CarVeR [19]	*Connectivity components*	*Potential function*	*By contribution*	*Stop by vertices*	–
Custom	*Connectivity components*	*Potential function*	*Relocation of r vertices*	*Stop by iteration, time*	–

- “Connectivity components” of the “Initial partition” block;
- “Relocation of *r* vertices” of the block “Relocation”;
- “Stop by iteration” of the “Stopping criterion” block.

Computational experiments on testing the new algorithm and comparing it with the known ones will be presented in section 3.3.

3.2. CCP solving software package

The technique of building and testing NS-algorithms for the CCP is implemented as software package in Python 3.10.2. The complex consists of three structural elements (Figure 3): NS-algorithms for the CCP solving, methods for signed graphs obtaining, comparison and analysis of the NS-algorithms results.

The first structural element implements the block modules shown in Figure 2. Modules are realized independently and can be combined according to the flow of the scheme taking into account their

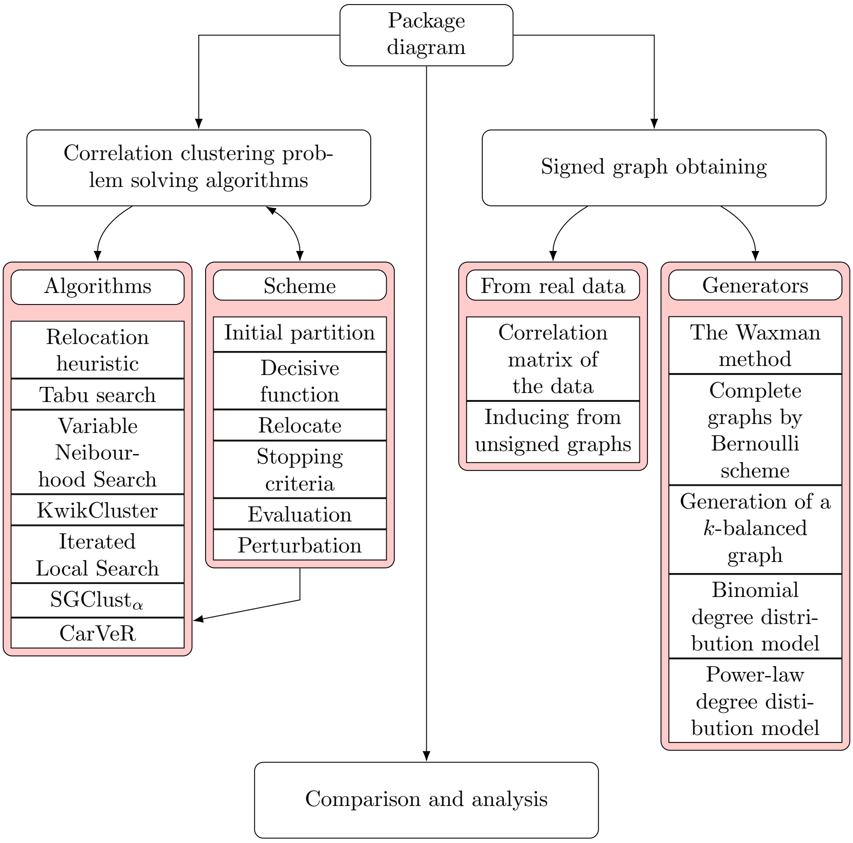


Figure 3. Software package structure

peculiarities. By combining the corresponding modules the known NS-algorithms according to the table 1 and the model algorithm *Custom* are realized in the program complex.

The second structural element implements both generation methods and methods of obtaining signed graphs from real data. The program provides the well-known generation methods, as well as methods for obtaining signed graphs from unsigned graphs described below.

The following methods of signed graph generation are implemented in the program complex.

- *Waxman's method [22].* To generate a graph consisting of n vertices, we generate n points on the two-dimensional plane. For each pair of vertices $\{u, v\}$ the probability of their connection by an edge is calculated by the following formula:

$$\mathbf{P}(\{u, v\}) = \beta \exp \frac{-d(u, v)}{L \cdot \alpha},$$

  where $d(u, v)$ is the distance between vertices u and v, L is the maximum distance between two vertices. The parameter β adjusts the density of edges, and the parameter α adjusts the density of long edges relative to short edges, $\alpha, \quad \beta \in (0, 1]$. The sign of an edge is generated by Bernoulli scheme with a given probability p for a positive sign.
- *Complete sign graphs by Bernoulli scheme [21].* For each pair of vertices of a complete graph, the sign of an edge is generated by Bernoulli scheme with a given probability of positive sign p.

- *k-balanced.* The input to the algorithm is k is the number of clusters, n_k is the number of vertices in a cluster, p is the probability of an edge occurring. The edges inside the clusters are assigned a positive sign, while the edges between them are assigned a negative sign.
- *2-balanced with binomial distribution of vertices [23].* An unsigned graph with binomial distribution of vertices is generated. Then, the set of vertices is randomly divided into two groups. Edges within groups are assigned a positive sign and edges between groups are assigned a negative sign.
- *2-balanced with power law distribution of vertices [23].* An unsigned graph is generated with a degree distribution of vertices. Then, the set of vertices is randomly divided into two groups. The edges within each group are assigned a positive sign, while the edges between the two groups are assigned a negative sign.
- *Binomial distribution of vertices.* An unsigned graph with binomial distribution of vertices is generated. The sign of an existing edge is determined by Bernoulli scheme with parameter p — the probability of positive sign.
- *Power law distribution of vertices.* An unsigned graph with a power law distribution of vertices is generated. The sign of an existing edge is determined by Bernoulli scheme with parameter p — the probability of positive sign.

Some methods of obtaining sign graphs from real data are implemented in the software package.

- *Direct assignment*. The data is represented as a sign graph without modifications.
- *Correlation matrix.* The initial object-property table is considered. The nodes of the graph are the properties of the objects. For each pair of properties, the correlation is calculated and the edge is labeled based on it.
- *Inducing from unsigned graphs*. The papers [24, 25], propose methods for obtaining signed graphs from unsigned graphs. In the proposed methods, existing edges in a graph are assigned a sign based on some vertex-dependent functional. For example, its degree or eccentricity.

The third structural element enables comparative analysis of algorithms. For example, a given number of graphs are generated according to the specified method for the selected NS-algorithms. On the obtained graphs the solution of the CCP is searched by each of the selected algorithms. A summary table containing the averaged results for each algorithm and the characteristics of the graphs is output to a file. For better perception of the experimental results, their visualization is provided.

3.3. Computation experiments

This section presents the results of computational experiments for the new model NS-algorithm *Custom*, which was proposed in section 3.1.

All experiments were carried out on a computer with 8GB RAM, an Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz processor running the Windows 10 operating system using single-threaded mode.

First series. The first series of experiments was conducted for the algorithms $SGClust_\alpha$, $CarVeR$ and *Custom* on 200 graphs of 25 vertices each generated using the Waxman method. All three algorithms were run for each graph and the time taken, as well as the main clustering parameters (number of clusters, positive and negative errors, and total error), were measured. The stopping criterion was applied to the relocation block of the *Custom* algorithm. The set time was equal to 5 s.

Figure 4 compares the algorithms $SGClust_\alpha$, $CarVeR$ and *Custom* by inter-cluster and intra-cluster errors. From the plots, we can conclude that the algorithms produce comparable error rates.

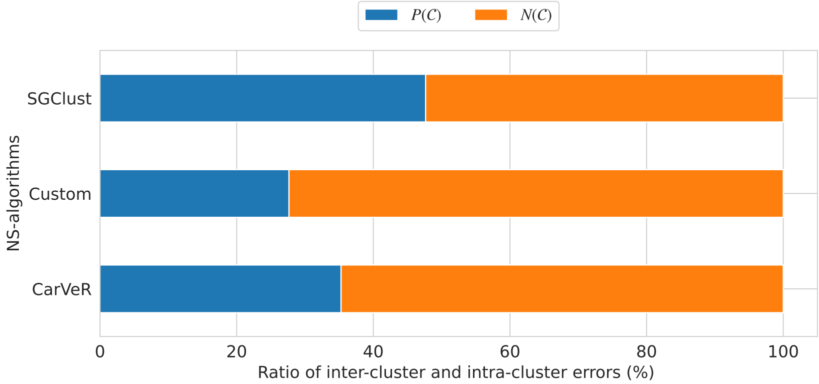


Figure 4. Comparison of NS-algorithms by inter-cluster $P(\mathcal{C})$ and intra-cluster $N(\mathcal{C})$ errors

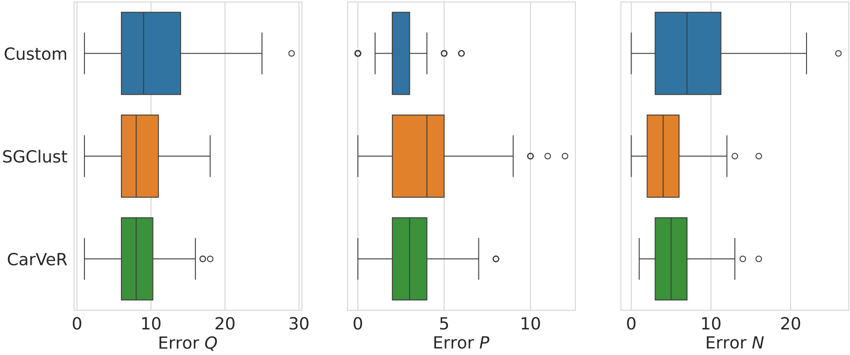


Figure 5. Comparison of algorithms by error functional

are comparable in terms of error.

Figure 5 compares the algorithms $SGClust_\alpha$, $CarVeR$ and $Custom$ in terms of total, positive and negative error values.

Second series. The second series of experiments was performed on the full signed graph, with parameters: $|V| = 20$, $|E| = 190$, $|E^+| = 94$, $|E^-| = 96$. The *Custom* algorithm was run with different specified times ranging from the running time of the *CarVeR* algorithm to 15 seconds with a step of 3 seconds. Some of the results are presented in the table 2. As can be seen from the table, the error decreases as the running time of the *Custom* algorithm increases. Therefore, when using time-dependent algorithms, it is necessary to select the running time.

4. Discussion

This paper builds upon our ongoing research into the development and analysis of heuristic NS-algorithms for solving the correlation clustering problem for signed networks. We have developed a technique for constructing NS-algorithms for solving CCP based on a systematic approach, which is

Table 2

Comparison of algorithms at different runtimes

Algorithm	Cluster count	Error	Negative error	Positive error	Time, s
Custom	2	84	72	12	2,828
Custom	2	72	52	20	3,464
Custom	3	59	33	26	6,045
$SGClust$	4	59	21	38	$< 0,001$
$CarVeR$	3	59	33	26	$< 0,001$

presented in the form of a generalized scheme comprising six interrelated components. The key idea of the proposed technique is a modular architecture, which enables the combination of different algorithmic solutions of existing and new NS-algorithms at each stage. This provides a flexible platform for the development and analysis of NS-algorithms.

The proposed technique has been implemented as a Python program. This complex provides the possibility of constructing new NS-algorithms, applying existing ones and comparing the results of their execution. The modular structure of the complex provides flexibility of approach: different modules can be more effective for certain types of networks. Due to this, searching for a solution by different NS algorithms can help to find a solution with better performance (lower error, in less time, etc.). Note that some of the modules used can be resource-intensive, which limits their application to high dimensional networks. Besides, the complex supports generation of sign networks with specified parameters, the induction of signed networks from unsigned graphs, construction of signed networks from real data. This enables the formation of a library of signed networks of varying complexity, which is necessary for testing and the comparative analysis of NS algorithms.

Based on the proposed approach, a model NS-algorithm was constructed, and its efficiency was tested in two series of computational experiments using synthetic data. In the first series of experiments, the model algorithm was compared with $CarVeR$ and $SGClust_\alpha$ algorithms in terms of error functional. The obtained results indicate that the value of the error functional for the model algorithm is comparable to the results of known algorithms, which confirms its competitiveness. In the second series of experiments, the effect of the stopping criterion on the quality of clustering was studied. In particular, it was found that the use of the time stopping criterion requires careful tuning, since the limitation of the algorithm running time can significantly affect the quality of the obtained solution. It should be noted that the constructed *Custom* algorithm is not intended to compete with existing NS-algorithms. Our objective was to showcase the potential of the proposed approach for developing such algorithms. Therefore, we can conclude that the *Custom* algorithm can be used as a basis for NS-algorithms creating.

5. Conclusion

A promising area for further research is the development of a formal method for obtaining an upper estimate of the computational complexity of algorithms constructed on the basis of the proposed scheme. It is also advisable to extend the software system's functionality by including other approaches to solving CCP, developing new methods for generating signed networks, adding new metrics for evaluating graph properties and clustering, and adapting the proposed technique to solve other problems on signed networks. Specific examples of extensions to the software package include: adding alternative error functionals to the "Evaluation" block, including different potential functions in the "Decision Function" block, adding new generation algorithms to the "Signed graph obtaining" block, and introducing different metrics for analyzing the sign network properties and clustering quality in the "Comparison and Analysis" block.

Author Contributions: Conceptualization, D.V. and A.A.; methodology, D.V. and A.A.; Software, E.I.; validation A.A. and E.I.; formal analysis D.V. and A.A.; investigation E.I.; data curation, E.I.; writing - original draft, E.I.; writing - review & editing, D.V. and A.A.; visualization, A.A. and E.I.; supervision, D.V.; project administration, D.V.; funding acquisition, D.V. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Krasnoyarsk Mathematical Center and financed by the Ministry of Science and Higher Education of the Russian Federation (Agreement No. 075-02-2024-1429).

Data Availability Statement: Data are openly available in a repository. Data supporting this study are openly available from Zenodo at 10.5281/zenodo.15206744 and 10.5281/zenodo.15206779.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Maatouk, A., Hajri, S. E., Assaad, M. & Sari, H. On optimal scheduling for joint spatial division and multiplexing approach in fdd massive mimo. *IEEE Trans Signal Process* **67,** 1006–10213 (2018).
2. Arinik, N., Figueiredo, R. & andLabatut, V. Multiple partitioning of multiplex signed networks: Application to European parliament votes. *Social Networks* **60,** 83–102 (2020).
3. Abdelnasser, A., Hossain, E. & Kim, D. I. Clustering and resource allocation for dense femtocells in a two-tier cellular OFDMA network. *IEEE Trans Wireless Communications* **13,** 1628–1641 (2014).
4. Hüffner, F., Betzler, N. & Niedermeier, R. *Optimal Edge Deletions for Signed Graph Balancing* in *Experimental Algorithms* (ed Demetrescu, C.) (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007), 297–310.
5. Figueiredo, R. & Frota, Y. *An improved Branch-and-cut code for the maximum balanced subgraph of a signed graph* 2013. ArXiv: abs/1312.4345.
6. Figueiredo, R. & Frota, Y. The maximum balanced subgraph of a signed graph: Applications and solution approaches. *European Journal of Operational Research* **236** (2014).
7. Tang, J., Chang, Y., Aggarwal, C. C. & Liu, H. A Survey of Signed Network Mining in Social Media. *ACM Computing Surveys CSUR* **49,** 1–37 (2015).
8. Il'ev, V., Il'eva, S. & Kononov, A. *Short Survey on Graph Correlation Clustering with Minimization Criteria* in *Lecture Notes in Computer Science* (Germany, Heidelberg, Springer-Verlag, 2016), 25–36.
9. Wahid, D. F. & Hassini, E. A Literature Review on Correlation Clustering: Cross-disciplinary Taxonomy with Bibliometric Analysis. *Oper. Res. Forum* **47** (2022).
10. Queiroga, E., Subramanian, A., Figueiredo, R. & Frota, Y. T. Integer programming formulations and efficient local search for relaxed correlation clustering. *Journal of Global Optimization* **81,** 919–966 (2021).
11. Ailon, N., Charikar, M. & Newman, A. *Aggregating inconsistent information: ranking and clustering* in *Symposium on the Theory of Computing* (2005).
12. Brusco, M. J. & Doreian, P. Partitioning signed networks using relocation heuristics, tabu search, and variable neighborhood search. *Social Networks* **56,** 70–80 (2019).
13. Levorato, M., Figueiredo, R., Frota, Y. & Drummond, L. Evaluating balancing on social networks through the efficient solution of correlation clustering problems. *EURO Journal on Computational Optimization* **5,** 467–498 (2017).
14. Doreian, P. & Mrvar, A. Partitioning signed social networks. *Soc. Networks* **31,** 1–11 (2009).

15. Harary, F. Structural Balance: A Generalization of Heider's Theory. *Psychological Review* **63,** 227–293 (1956).
16. Graham, R. L., Knuth, D. E. & Patashnik, O. *Concrete Mathematics: A Foundation for Computer Science 2nd edn.* 657 pp. (Addison-Wesley, Massachusetts, USA, 1994).
17. Doreian, P. & Mrvar, A. A partitioning approach to structural balance. *Social Networks* **18,** 149–168 (1996).
18. Bansal, N., Blum, A. & Chawla, S. Correlation Clustering. *Machine Learning* **56,** 89–113 (2002).
19. Soldatenko, A. A., Semenova, D. V. & Ibragimova, E. I. An approach to analyzing and constructing algorithms for solving a one clustering problem on signed graphs. *Applied discrete mathematics,* 112–127 (2024).
20. Ibragimova, E. I., D. V. Soldatenko, A. A. & Semenova. *Comparison of Two Heuristic Algorithms for Correlation Clustering Problem Solving* in *2023 5th International Conference on Problems of Cybernetics and Informatics (PCI)* (IEEE, 2023), 1–4.
21. Soldatenko, A., Semenova, D. & Ibragimova, E. *On heuristic algorithm with greedy strategy for the Correlation Clustering problem solution* in *Lecture Notes in Computer Science* (Germany, Heidelberg, Springer-Verlag, 2023), 462–477.
22. Waxman, B. M. Routing of multipoint connections. *IEEE Journal on Selected Areas in Communications* **6,** 1617–1622 (1988).
23. Ciotti, V., Bianconi, G., Capocci, A., Colaiori, F. & Panzarasa, P. Degree correlations in signed social networks. *Physica A: Statistical Mechanics and its Applications,* 25–39 (2015).
24. Aniyan, A. & Naduvath, S. Induced signed graphs of some classes of graphs. *Proceedings of the Jangjeon Mathematical Society* **23,** 283–291 (2020).
25. Sudheer, N., George, A. C., Aniyan, A. & Naduvath, S. Some new results on colour-induced signed graphs. *Acta Univ. Sapientiae Informatica* **14,** 338–353 (2022).

Information about the authors

Ellada I. Ibragimova—Assistant of Department of Higher and Applied Mathematics, Siberian Federal University (e-mail: EIbragimova@sfu-kras.ru, ORCID: 0000-0002-6852-2281, Scopus Author ID: 58766400800)

Daria V. Semenova—Candidate in Physics and Mathematics, Associate Professor of the Department of Higher and Applied Mathematics, Siberian Federal University (e-mail: DVSemenova@sfu-kras.ru, ORCID: 0000-0002-8670-2921, ResearcherID: O-3107-2019, Scopus Author ID: 49261191500)

Aleksandr A. Soldatenko—Candidate in Physics and Mathematics, Associate Professor of the Department of Higher and Applied Mathematics, Siberian Federal University (e-mail: ASoldatenko@sfu-kras.ru, ORCID: 0000-0001-6708-4753, ResearcherID: O-8550-2019, Scopus Author ID: 57194285182)

УДК 519.[176+178+16]:004.4
PACS 02.10.Ox 89.70.Eg 07.05.Tp
DOI: 10.22363/2658-4670-2025-33-2-130-143 EDN: BMIMTX

О методе построения алгоритмов для решения задачи корреляционной кластеризации

Э. И. Ибрагимова, Д. В. Семенова, А. А. Солдатенко

Сибирский федеральный университет, д. 79, пр. Свободный, г. Красноярск, 660041, Российская Федерация

Аннотация. Развивается техника построения алгоритмов, основанных на структуре сети (NS-алгоритмы), для решения задачи корреляционной кластеризации (CCP) для знаковых сетей. Моделью знаковой сети выступает неориентированный и невзвешенный простой знаковых графов. Эта задача рассматривается в оптимизационной форме с функционалом ошибки в виде линейной комбинации межкластерной и внутрикластерной ошибок. Известно, что в данной постановке задача является NP-трудной. Техника построения NS-алгоритмов основана на системном подходе, представленном в виде общей схемы. Схема состоит из шести взаимосвязанных блоков, каждый из которых отражает основные этапы решения CCP. Основная идея техники заключается в комбинировании модулей, представляющих каждый блок схемы. Предложенный подход реализован в виде программного комплекса. В работе представлен модельный NS-алгоритм, построенный с помощью предлагаемой техники. Проведены вычислительные эксперименты на синтетических данных по сравнению модельного алгоритма с уже известными.

Ключевые слова: знаковая сеть, задача корреляционной кластеризации, NS-алгоритм, техника построения

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 144–156
http://journals.rudn.ru/miph

Research article

UDC 519.872
PACS 07.05.Tp
DOI: 10.22363/2658-4670-2025-33-2-144-156

EDN: BMHYOY

Asymptotic analysis of multiserver retrial queueing system with π-defeat of negative arrivals under heavy load

Natalya P. Meloshnikova, Ekaterina A. Fedorova

National Research Tomsk State University, 36 Lenin Ave, Tomsk, 634050, Russian Federation

(received: April 30, 2025; revised: May 10, 2025; accepted: May 14, 2025)

Abstract. The paper studies a multiserver retrial queuing system with π-defeat as a mathematical model of cloud services. The arrival processes of "positive" calls are Poisson. The system has a finite number of servers and the service time for calls at the servers is exponentially distributed. When all servers are busy, calls entering the system transfer to an orbit, where they experience a random delay. After the delay, calls from the orbit attempt to access the service unit according to a multiple access policy. The system also receives a stream of negative calls. Negative calls do not require the service. An negative call "deletes" a random number of calls is the service unit. For the considered model, the Kolmogorov equations are written in the steady state. The method of asymptotic analysis under a heavy load condition is applied for deriving the stationary probability distribution of the number of calls in the orbit. The results of the numerical analysis are presented.

Key words and phrases: mathematical modelling, retrial queueing system, negative calls, asymptotic analysis, heavy load

For citation: Meloshnikova, N. P., Fedorova, E. A. Asymptotic analysis of multiserver retrial queueing system with π-defeat of negative arrivals under heavy load. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 144–156. doi: 10.22363/2658-4670-2025-33-2-144-156. edn: BMHYOY (2025).

1. Introduction

Cloud technologies represent a model for providing computing resources on demand over the Internet, where infrastructure, software, and data are located on remote servers (in the "cloud") rather than on the user's local devices. Users access these resources via the Internet using various devices, such as computers, smartphones, and tablets, and pay only for the resources actually consumed, making cloud technologies a flexible, scalable, and cost-effective solution.

Cloud technologies encompass a wide range of services delivered over the internet, including Infrastructure as a Service (IaaS) from AWS, Azure, and Google Cloud; Platform as a Service (PaaS) such as Heroku and App Engine; Software as a Service (SaaS) like Microsoft 365 and Salesforce; Functions as a Service (FaaS) such as AWS Lambda; as well as Database as a Service (DBaaS) and cloud storage solutions such as Amazon S3, Azure Blob Storage, and Google Cloud Storage, providing users with flexible, scalable, and cost-effective computing resources [1].

Cloud technologies, as and development of new telecommunication networks, remains a priority for science and technology, reflected in diverse approaches ranging from analytical methods [2],the development of architectures for managing network slicing [3] and the performance modeling of mmWave networks [4], requiring the advancement of analytical tools.

Mathematical modeling is critically important for the optimization of costs, the enhancement of performance, and the assurance of reliability by predicting resource consumption, identifying bottlenecks, and planning for scaling. It is particularly important that modeling allows for the consideration of potential negative factors, such as software failures, cyberattacks, and accidents, in order to develop effective protection and redundancy strategies, guaranteeing the uninterrupted operation of the system.

In this paper, we present a mathematical model of a cloud in the form of a multi-server retrial queueing system (RQ system) with negative calls.

Retrial queueing systems are mathematical models of queueing theory widely used to analyze and optimize various telecommunications systems, mobile communication networks, and call centers [5–7]. The main feature of such models is the presence of repeated calls to the server after an unsuccessful attempt to receive the service. There is not a queue in the system, unserved calls go to an orbit (some virtual place), where they perform a random delay. There is a random access protocol for all calls in an orbit.

J. Artalejo and A. Gomez-Corral [5], G. Falin and J. Templeton [6] offered comprehensive treatments in retrial queueing systems, establishing the groundwork for analyzing queues with repeated calls. T. Phung-Duc [7] provides a survey of retrial queueing models, highlighting their theoretical developments and diverse applications.

The concept of negative calls was pioneered by E. Gelenbe [8]. He introduced negative signals that can remove calls from the system, providing a framework for modeling complex interactions in various systems. This concept was further explored in [9, 10], solidifying the mathematical foundation for this area, such models began to be called as G-networks and G-systems. Do [11] offers a valuable bibliography on G-networks, negative calls, and their applications. M. Caglayan [12] highlighted the applications of G-networks to machine learning and energy packet networks.

Later research has expanded upon these foundations, considering various aspects of G-networks and retrial queues with negative calls. P. Bocharov and V. Vishnevsky [13] discussed the development of the theory of multiplicative networks in the context of G-networks. Y. Shin [14] investigated multiserver retrial queues with both negative calls and disasters, while R. Razumchik [15] studied queueing systems with negative arrivals, a “bunker” for displaced calls, and varying service intensities. M. Matalytski and V. Naumenko [16] analyzed queueing networks with bounded waiting times for both positive and negative calls under non-stationary conditions. Further contributions to this field include research on related queuing models. Liu et al. [17] explore a multiserver two-way communication retrial queue subject to disasters and synchronous working vacations, offering insights into the impact of disruptive events and service strategies on system performance. A. Melikov [18] considers inventory queuing systems with negative arrivals. E. Lisovskaya et al. [19] investigate a resource retrial queue with two orbits and negative customers. These works demonstrate a great interest and expansion of the theoretical understanding and practical applications of G-systems with negative calls.

In this paper, we consider a multiserver retrial queuing system with π-defeat as a model for cloud services. In terms of queueing theory, data flow in cloud services is described as an arrival process in multiserver system, which defines a cloud node. The orbit is the storage location for so-called “sleeping” requests, negative arrivals define hacker attacks.

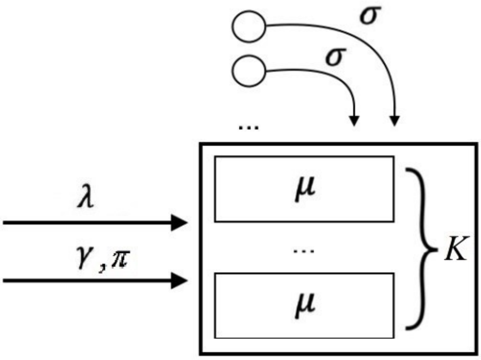


Figure 1. Multiserver retrial queueing systems system with π-defeat

The article is structured as follows. Section 2 describes a multiserver queueing system with retrials and catastrophes in the service unit, formulates the problem statement, and writes down the system of Kolmogorov equations. Section 3 is devoted to the asymptotic analysis method under heavy load conditions, which is applied to solve the system of equations. In Section 4, we demonstrate some numerical experiments that show the accuracy of the asymptotic results. Section 5 is devoted to some concluding remarks.

2. Mathematical model

In this paper, we consider a multiserver retrial queueing systems (Figure 1). The arrival process is Poisson with parameter λ, we will call these calls as "positive." Positive calls arrive into a service unit (which has K servers), until all servers become busy. The service time of each call is distributed exponentially with parameter μ. If all servers are busy, an arrival call goes to an orbit, where it performs a random delay. The delay duration has an exponential distribution with parameter σ. From the orbit, the call again turns to the service unit. If there is a free server, the call begins its service; otherwise, it returns to the orbit to make a next attempt. We suppose that there is a multiple access policy in the orbit.

In addition, negative calls arrive into the system according to Poisson arrival process with parameter γ. A negative call does not need the service; it negatively affects the system. Here, we consider a general model with negative arrivals – π-defeat. It means that a negative call deletes k servicing calls with probability π_k, where $k = \overline{1, K}$. For the probability distribution of the number of deleted calls, the normalization condition holds $\sum_{k=1}^{K} \pi_k = 1$.

Special cases of the presented model are considered in the previous studies, such as RQ with disasters in the service unit ($\pi_k = 0$ for $k = \overline{1, K-1}$ and $\pi_K = 1$) [20], the model with a single destruction ($\pi_1 = 1$ and $\pi_k = 0$ for $k = \overline{2, K}$) [21].

Denote a random process of the number of calls in the orbit by $i(t)$. The problem of obtaining the stationary probability distribution of the number of calls in the orbit is posed.

Due to $i(t)$ is not Markov process, we introduce process $k(t)$ determined states of the service unit as follows:

$$k(t) = \begin{cases} 0, \text{if all servers are free,} \\ 1, \text{if one server is busy,} \\ \ldots \\ k, \text{if } k \text{ servers are busy,} \\ \ldots \\ K, \text{if all servers are busy.} \end{cases}$$

Obviously, process $\{k(t), i(t)\}$ is a continuous-time Markov chain. Let $P\{k(t) = k, i(t) = i\} = P(k, i)$ be the stationary probabilities that the service unit is in state k and there are i calls in the orbit. The system of Kolmogorov equations in the steady state has the following form

$$\begin{cases} -P(0,i)(\lambda + i\sigma) + P(1,i)\mu + \sum\limits_{k=1}^{K} P(k,i)\gamma \sum\limits_{v=k}^{K} \pi_v = 0, \\ -P(k,i)(\lambda + k\mu + i\sigma + \gamma) + P(k-1,i)\lambda + P(k-1,i+1)\sigma(i+1) + \\ \qquad + P(k+1,i)(k+1)\mu \sum\limits_{v=k+1}^{K} P(v,i)\gamma\pi_{v-k} = 0, \text{ with } 1 \leqslant k \leqslant K-1, \\ -P(K,i)(\lambda + K\mu + \gamma) + P(K-1,i)\lambda + P(K,i-1)\lambda + P(K-1,i+1)\sigma(i+1) = 0. \end{cases} \tag{1}$$

In addition, the normalization condition must be taken into account:

$$\sum_{k=0}^{K} \sum_{i=0}^{\infty} P(k,i) = 1.$$

Let us introduce the partial characteristic functions:

$$H_k(u) = \sum_{i=0}^{\infty} e^{jui} P(k,i).$$

Then we rewrite System (1) as follows

$$\begin{cases} -H_0(u)\lambda + j\sigma\frac{\partial H_0(u)}{\partial u} + H_1(u)\mu + \gamma \sum\limits_{k=1}^{K} H_k(u) \sum\limits_{v=k}^{K} \pi_v = 0, \\ -H_k(u)(\lambda + k\mu + \gamma) + j\sigma\dfrac{\partial H_k(u)}{\partial u} - j\sigma e^{-ju}\dfrac{\partial H_{k-1}(u)}{\partial u} + +H_{k-1}(u)\lambda + \\ \qquad + H_{k+1}(u)(k+1)\mu + \gamma \sum\limits_{v=k+1}^{K} H_v(u)\pi_{v-k} = 0, \text{ with } 1 \leqslant k \leqslant K-1, \\ -H_K(u)(\lambda(1 - e^{ju}) + K\mu + \gamma) + H_{K-1}(u)\lambda - j\sigma e^{-ju}\frac{\partial H_{K-1}(u)}{\partial u} = 0. \end{cases} \tag{2}$$

Summing up all equations of System (2), we obtain an additional equation

$$j\sigma \sum_{k=0}^{K-1} \frac{\partial H_k(u)}{\partial u} + H_K(u)\lambda e^{ju} = 0. \tag{3}$$

3. Asymptotic analysis

System (2)-(3) is a differential equations system of K functions, thus it is necessary to apply specific methods for its solving. We propose the method of the asymptotic analysis under a heavy load condition in this paper.

Let us denote the load parameter

$$\rho = \frac{\lambda}{K\mu}.$$

The steady state condition is written $\rho < S$, where S is an upper limit value of the load parameter, such as the system being unstable for $\rho \geq S$. Sometimes, S is called a throughput parameter. Note that an expression for S is unknown for the considered model, but further we will derive it.

The asymptotic condition of a heavy load is described as $\rho \to S$ or $\varepsilon \to 0$, where $\varepsilon = S - \rho$.

The method of the asymptotic analysis consists of several steps. First of all, we introduce the asymptotic notations:

$$\lambda = (S-\varepsilon)K\mu,\ u = \varepsilon w,\ H_k(u) = \varepsilon F_k(w,\varepsilon),\ H_K(u) = F_K(w,\varepsilon).$$

From system (2)–(3), the following system of the asymptotic equation is obtained

$$\begin{cases} -F_0(w,\varepsilon)\varepsilon(S-\varepsilon)K\mu + j\sigma\frac{\partial F_0(w,\varepsilon)}{\partial w} + F_1(w,\varepsilon)\varepsilon\mu + F_K(w,\varepsilon)\gamma\pi_K + \gamma\sum\limits_{k=1}^{K-1}F_k(w,\varepsilon)\varepsilon\sum\limits_{v=k}^{K}\pi_v = 0, \\ -F_k(w,\varepsilon)\varepsilon((S-\varepsilon)K\mu + k\mu + \gamma) + j\sigma\dfrac{\partial F_k(w,\varepsilon)}{\partial w} + F_K(w,\varepsilon)\gamma\pi_{K-k} - j\sigma e^{-j\varepsilon w}\dfrac{\partial F_{k-1}(w,\varepsilon)}{\partial w} + \\ \qquad + F_{k-1}(w,\varepsilon)\varepsilon(S-\varepsilon)K\mu + F_{k+1}(w,\varepsilon)\varepsilon(k+1)\mu + \gamma\sum\limits_{v=k+1}^{K-1}F_v(w,\varepsilon)\varepsilon\pi_{v-k} = 0,\ 1 \leqslant k \leqslant K-2, \\ -F_{K-1}(w,\varepsilon)\varepsilon((S-\varepsilon)K\mu + (K-1)\mu + \gamma) + j\sigma\dfrac{\partial F_{K-1}(w,\varepsilon)}{\partial w} - j\sigma e^{-j\varepsilon w}\dfrac{\partial F_{K-2}(w,\varepsilon)}{\partial w} + \\ \qquad + F_{K-2}(w,\varepsilon)\varepsilon(S-\varepsilon)K\mu + F_K(w,\varepsilon)K\mu + F_K(w,\varepsilon)\gamma\pi_1 = 0,\ k = K-1, \\ -F_K(w,\varepsilon)((S-\varepsilon)K\mu(1-e^{j\varepsilon w}) + K\mu + \gamma) + F_{K-1}(w,\varepsilon)(S-\varepsilon)K\mu\varepsilon - j\sigma e^{-j\varepsilon w}\frac{\partial F_{K-1}(w,\varepsilon)}{\partial w} = 0, \end{cases} \tag{4}$$

and

$$j\sigma\sum_{k=0}^{K-1}\frac{\partial F_k(w,\varepsilon)}{\partial w} + F_K(w,\varepsilon)(S-\varepsilon)K\mu e^{j\varepsilon w} = 0. \tag{5}$$

Under $\varepsilon \to 0$ in system (4)–(5), we have

$$\begin{cases} j\sigma F_0'(w) + F_K(w)\gamma\pi_K = 0, \\ j\sigma F_k'(w) - j\sigma F_{k-1}'(w) + F_K(w)\gamma\pi_{K-k} = 0,\ 1 \leqslant k \leqslant K-2, \\ j\sigma F_{K-1}'(w) - j\sigma F_{K-2}'(w) + F_K(w)K\mu + F_K(w)\gamma\pi_1 = 0, \\ -j\sigma F_{K-1}'(w) - F_K(w)(K\mu + \gamma) = 0, \end{cases} \tag{6}$$

and

$$j\sigma\sum_{k=0}^{K-1}\frac{\partial F_k(w)}{\partial w} + F_K(w)SK\mu = 0. \tag{7}$$

From system (6), it can be obtained the following expressions:

$$\begin{cases} j\sigma F_0'(w) = -F_K(w)\gamma\pi_K, \\ j\sigma F_k'(w) = -F_K(w)\gamma(\pi_K + \pi_{K-1} + ... + \pi_{K-k}) = 0, \ 1 \leqslant k \leqslant K-2, \\ j\sigma F_{K-1}'(w) = -F_K(w)(K\mu+\gamma) = 0. \end{cases} \tag{8}$$

From (8), let us express:

$$j\sigma \sum_{k=0}^{K-1} \frac{\partial F_k(w)}{\partial w} = -F_K(w)K\mu - F_K(w)\gamma(K\pi_K + (K-1)\pi_{K-1} + ... + 2\pi_2 + \pi_1). \tag{9}$$

Substituting (9) into (7), we obtain:

$$S = \frac{b\gamma}{K\mu} + 1,$$

where $b = \sum_{k=1}^{K} k\pi_k$ is a mean of the number of servicing calls deleted by negative arrivals.

In the next step of the asymptotic analysis, the following expansions for functions $F_k(w,\varepsilon)$ will be used:

$$F_k(w,\varepsilon) = F_k(w) + \varepsilon f_k(w) + O(\varepsilon^2). \tag{10}$$

Substituting (10) into system (4)–(5) and taking into account (6)–(7), we derive

$$\begin{cases} -SK\mu\varepsilon F_0(w) + j\sigma F_0'(w) + j\sigma\varepsilon f_0'(w) + \mu\varepsilon F_1(w) + \\ \quad + \gamma\sum_{k=1}^{K-1} \varepsilon F_k(w) \sum_{v=k}^{K-1} \pi_v + \gamma\pi_K F_K(w) + \gamma\varepsilon\pi_K F_K(w) = 0, \\ -(k\mu+\gamma)\varepsilon F_k(w) - SK\mu\varepsilon F_k(w) + j\sigma F_k'(w) + j\sigma\varepsilon f_k'(w) - \\ \qquad - j\sigma F_{k-1}'(w) + j\sigma(j\varepsilon w)F_{k-1}'(w) - j\sigma\varepsilon f_{k-1}' + SK\mu\varepsilon F_{k-1}(w) + \\ \quad + (k+1)\mu\varepsilon F_{k+1}(w) + \gamma\sum_{v=k+1}^{K-1} \varepsilon\pi_{v-k} F_v(w) + \gamma\varepsilon\pi_{K-k} f_K(w) + \gamma\pi_{K-k} F_K(w) = 0, k = 1...K-2, \\ -SK\mu\varepsilon F_{K-1}(w) - ((K-1)\mu+\gamma)\varepsilon F_{K-1}(w) + j\sigma F_{K-1}'(w) + \\ \qquad + j\sigma\varepsilon f_{K-1}'(w) - j\sigma F_{K-2}'(w) + j\sigma(j\varepsilon w)F_{K-2}'(w) - j\sigma\varepsilon f_{K-2}' + \\ \quad + SK\mu\varepsilon F_{K-2}(w) + K\mu F_K(w) + K\mu\varepsilon f_K(w) + \gamma\pi_1 F_K(w) + \gamma\varepsilon\pi_1 f_K(w) = 0, k = K-1, \\ -SK\mu(j\varepsilon w)F_K(w) - (K\mu+\gamma)F_K(w) - (K\mu+\gamma)\varepsilon f_K(w) + \\ \quad + SK\mu\varepsilon F_{K-1}(w) - j\sigma(1-j\varepsilon w)F_{K-1}'(w) + j\sigma(j\varepsilon w)F_{K-1}'(w) - j\sigma\varepsilon f_{K-1}'(w) = 0, \end{cases} \tag{11}$$

and

$$j\sigma \sum_{k=0}^{K-1} \frac{\partial f_k(w)}{\partial w} + SK\mu(jw)F_K(w) - K\mu F_K(w) + SK\mu f_K(w) = 0. \tag{12}$$

Under limit $\varepsilon \to 0$, system (11)–(12) is written as

$$\begin{cases} -SK\mu F_0(w) + j\sigma f_0'(w) + \mu F_1(w) + \gamma \sum\limits_{k=1}^{K-1} F_k(w) \sum\limits_{v=k}^{K} \pi_v + \gamma \pi_K f_K(w) = 0, \\ -(SK\mu + k\mu + \gamma)F_k(w) + j\sigma f_k'(w) + j\sigma jw F_{k-1}'(w) - j\sigma f_{k-1}'(w) + SK\mu F_{k-1}(w) + \\ \qquad + (k+1)\mu F_{k+1}(w) + \gamma \sum\limits_{v=k+1}^{K-1} \pi_{v-k} F_v(w) + \gamma \pi_{K-k} f_K(w) = 0, \ k = 1 ... K-2, \\ -(SK\mu + (K-1)\mu + \gamma)F_{K-1}(w) + j\sigma f_{K-1}'(w) + j\sigma jw F_{K-2}'(w) - \\ - j\sigma f_{K-2}'(w) + SK\mu F_{K-2}(w) + K\mu f_K(w) + \gamma \pi_1 f_K(w) = 0, \\ -jwSK\mu F_K(w) - (K\mu + \gamma) f_K(w) + SK\mu F_{K-1}(w) + j\sigma jw F_{K-1}'(w) - j\sigma f_{K-1}'(w) = 0, \end{cases} \tag{13}$$

and

$$j\sigma \sum_{k=0}^{K-1} \frac{\partial f_k(w)}{\partial w} + K\mu(Sjw - 1)F_K(w) + SK\mu f_K(w) = 0. \tag{14}$$

Let us express $j\sigma f_0'(w)$ from the first equation of System (13):

$$j\sigma f_0'(w) = SK\mu F_0(w) - \mu F_1(w) - \gamma \sum_{k=1}^{K-1} F_k(w) \sum_{v=k}^{K} \pi_v - \gamma \pi_K f_K(w).$$

From system (13), the following expressions can be derived

$$\begin{aligned} j\sigma f_k'(w) = SK\mu F_k(w) - j\sigma jw \sum_{v=0}^{k-1} F_v'(w) - ((k+1)\mu + \gamma)F_{k+1}(w) - \\ - \gamma \sum_{m=k+2}^{K-1} F_m(w) \sum_{v=m-k}^{K} \pi_v - \gamma f_K(w) \sum_{v=K-k}^{K} \pi_v \ \text{ for } k = \overline{1, K-2}, \end{aligned} \tag{15}$$

and

$$j\sigma f_{K-1}'(w) = SK\mu F_{K-1}(w) - j\sigma jw \sum_{v=0}^{K-2} F_v'(w) - (K\mu + \gamma) f_K(w). \tag{16}$$

Let us denote $g_k = \sum\limits_{v=k}^{K} \pi_v$. Taking into account (15)–(16), we derive

$$\begin{aligned} j\sigma \sum_{k=0}^{K-1} f_k'(w) = -\gamma \sum_{k=2}^{K-1} F_k(w)(g_k + 1) + \sum_{k=0}^{K-1} SK\mu F_k(w) - \sum_{k=1}^{K-1} k\mu F_k(w) - \\ - \gamma F_1(w) - \gamma \sum_{k=1}^{K-3} \sum_{m=k+2}^{K-1} F_m(w) g_{m-k} - j\sigma jw \sum_{k=1}^{K-1} \sum_{v=0}^{k-1} F_v'(w) - f_K(w) SK\mu. \end{aligned}$$

The last step is substituting all derived expression into Equation (14):

$$\begin{aligned} -\gamma \sum_{k=2}^{K-1} F_k(w)(g_k + 1) + SK\mu \sum_{k=0}^{K-1} F_k(w) - \mu \sum_{k=1}^{K-1} kF_k(w) - \gamma F_1(w) - \\ - \gamma \sum_{k=1}^{K-3} \sum_{m=k+2}^{K-1} F_m(w) g_{m-k} - jw\gamma F_K(w) \sum_{v=2}^{K} (v-1) g_v + K\mu(Sjw - 1)F_K(w) = 0. \end{aligned}$$

Substituting (8), we obtain

$$\frac{\gamma}{j\sigma}(K\mu+\gamma)F_K(w)(g_{K-1}+1)+\frac{\gamma}{j\sigma}\sum_{k=2}^{K-2}F_K(w)g_{K-k}(g_k+1)-\frac{SK\mu}{j\sigma}(K\mu+\gamma)F_K(w)-\frac{SK\mu}{j\sigma}\sum_{k=0}^{K-2}\gamma F_K(w)g_{K-k}+$$
$$+\frac{\mu(K-1)}{j\sigma}(K\mu+\gamma)F_K(w)+\frac{\mu}{j\sigma}\sum_{k=1}^{K-2}k\gamma F_K(w)g_{K-k}-\frac{\gamma}{j\sigma}\gamma F_K(w)g_{K-1}-$$
$$-\gamma\sum_{k=1}^{K-3}\Big(-\frac{(K\mu+\gamma)}{j\sigma}F_K(w)g_{K-1-k}-\frac{\gamma}{j\sigma}\sum_{m=k+2}^{K-2}F_K(w)g_{K-m}g_{m-k}\Big)+$$
$$+j\gamma F_K(w)\sum_{v=2}^{K}(v-1)g_v+jw\gamma F_K'(w)\sum_{v=2}^{K}(v-1)g_v+KS\mu jF_K(w)+K\mu(Sjw-1)F_K'(w)=0.$$

After some mathematical transformations, we obtain a differential equation for function $F_K(w)$ of the following form:

$$F_K(w)\alpha+j\sigma(1-jw\beta)F'(w)=0, \tag{17}$$

where

$$\begin{gathered}\alpha=\sigma\beta+\mu+\gamma\Big(b+\frac{(K-1)^2}{K}+2\pi_{K-1}+3\pi_K+\frac{d}{K}\Big)-\frac{\gamma}{K\mu}(f_2+b+\\ +\pi_{K-1}+2\pi_K+1)+\frac{\gamma^2}{K\mu}\Big(K(b-1)+3\pi_{K-1}+4\pi_K+2-C\Big),\\ \beta=S+\frac{\gamma}{K\mu}\sum_{k=2}^{K}(k-1)g_k,\ f_n=\sum_{k=n}^{K-2}g_{K-k}g_k, g_k=\sum_{v=k}^{K}\pi_v,\\ d=\sum_{k=0}^{K-2}kg_{K-k}, c_{k+2}=\sum_{m=k+2}^{K-2}g_{K-m}\sum_{n=m-k}^{m-1}\pi_n,\ C=\sum_{k=1}^{K-3}(f_{k+2}+c_{k+2}).\end{gathered} \tag{18}$$

The solution to the equation (17) is:

$$F_K(w)=\Big(1-jw\beta\Big)^{-\frac{\alpha}{\sigma}}.$$

Turning up the asymptotic notation, the characteristic function of the number of calls in the orbit

$$H(u)=F_K\Big(\frac{u}{\varepsilon}\Big)+O(\varepsilon).$$

In this way, we finally conclude that the asymptotic characteristic function of the number of calls in the orbit of the considered model under a heavy load condition

$$h(u)=F_K\Big(\frac{u}{S-\rho}\Big)=\Big(1-\frac{ju\beta}{S-\rho}\Big)^{-\frac{\alpha}{\sigma}}, \tag{19}$$

has the form of the characteristic function of the gamma distribution with parameters α and β defined by (18).

3.1. Special cases

As mentioned above, the studied retrial queueing system with π-defeat generalizes models considered previously in [20, 21]. So we can compare the asymptotic results in two special cases:

1. Model with single destruction: $\pi_1=1$ and $\pi_k=0$ for $k=\overline{2,K}$.
2. Model with disasters in the service unit: $\pi_K=1$ and $\pi_k=0$ for $k=\overline{1,K-1}$.

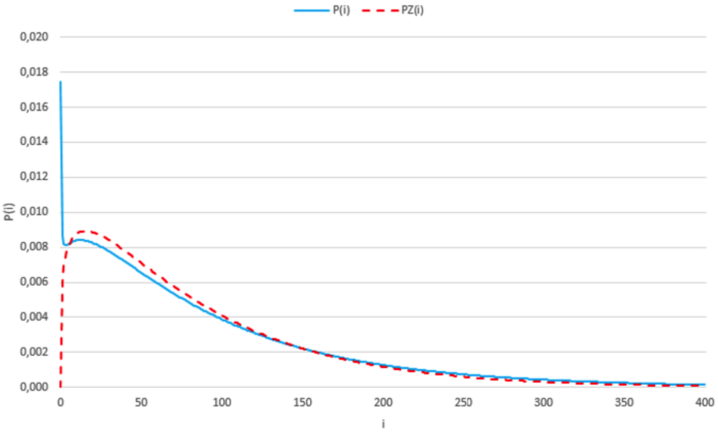


Figure 2. Exact and asymptotic probability distributions for $\rho = 0.99$

By substituting the corresponding values of π_k into (19) the following corollaries can be formulated (which coincide with [20] and [21]).

Corollary 1. The asymptotic characteristic function of the probability distribution of the number of calls in orbit in RQ M/M/K with single destruction of negative calls has the form of the gamma distribution function (19) with parameters $\alpha = \mu + \gamma + \sigma,\ \beta = 1,\ S = 1 + \dfrac{\gamma}{K\mu}$.

Corollary 2.It can be derived from (19) that the asymptotic characteristic function of the probability distribution of the number of calls in orbit in RQ M/M/K with disasters in the service unit has the form of the gamma distribution function (19) with parameters $\beta = 1 + \frac{\gamma(K+1)}{2\mu}$, $S = 1 + \frac{\gamma}{\mu}$,

$$\alpha = \sigma\beta + \mu + \gamma(2K-1) - \tfrac{\gamma(K-1)(2+\gamma(K-2))}{2K} + \tfrac{\gamma^2(K^2+K-4)}{2K\mu}.$$

Note that we have proved that the asymptotic distribution under a heavy load has a gamma form for all cases.

4. Numerical analysis

To analyze the range of applicability of the proposed asymptotic method, we numerically compare asymptotic distribution $PZ(i)$ and exact distribution $P(i)$ obtained using a numerical algorithm for various values of the system parameters.

As a measure of the asymptotic method accuracy, we use the Kolmogorov distance:

$$\triangle = \left| \sum_{n=0}^{i} PZ(n) - P(n)) \right|.$$

As an example, we present the calculation for the following values of system parameters:

$$\rho = 0.99 \cdot S,\ \mu = 1,\ \gamma = 0.01,\ \sigma = 5,\ K = 5,\ \pi = [0.247\ 0.603\ 0.101\ 0.045\ 0.004]$$

From Figure 4, we can see that the exact distribution has the value of the zero state probability ($P\{i(t) = 0\}$) much greater than others $P(i)$. This is the main feature of the model for quite large values of γ (negative calls arrive more frequently and more calls are deleted).

Table 1

Kolmogorov distances for various values of the parameter ρ

	$\gamma = 0.01$	$\gamma = 0.001$	$\gamma = 0.0001$
$\rho = 0.99 \cdot S$	0.120	0.026	0.016
$\rho = 0.98 \cdot S$	0.120	0.041	0.034
$\rho = 0.97 \cdot S$	0.06	0.053	0.052

In this way the approximation has a quite big error in the point $i(t) = 0$ ($\Delta = 0.120$, Table 1), while if we analyze the entire remaining range of $i(t) > 0$ (excluding the zero point), the Kolmogorov distance does not $\sum_{i \neq 0} \Delta < 0.01$. Unfortunately, it is not possible to analytically estimate the probability P_0 from System 1 or by the asymptotic analysis.

The results of the comparison of distributions for different values of are presented in Table 1 and Figure 4.

From Table 1, it can be concluded that the accuracy of the approximation increases as the system load increases and a negative arrival rate decreases.

5. Conclusion

In this study, a multiserver RQ-system with π-defeat is considered as a mathematical model of cloud services. The asymptotic analysis method under a heavy load condition is applied. It is proven that the asymptotic characteristic function of the distribution of the number of calls in the orbit has the form of the gamma distribution function with the obtained parameters. A formula for the system throughput is obtained. Numerical analysis is presented, demonstrating the accuracy of the approximation.

Author Contributions: Conceptualization, N. Meloshnikova and E. Fedorova; methodology E. Fedorova; software, N. Meloshnikova; validation, N. Meloshnikova; formal analysis, N. Meloshnikova and E. Fedorova; investigation, N. Meloshnikova and E. Fedorova; writing—original draft preparation, N. Meloshnikova; writing—review and editing, E. Fedorova. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing is not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kavis, M. J. *Architecting the Cloud: Design Decisions for Cloud Computing Service Models* 224 pp. (Wiley; 1st edition, 2014).
2. Boccardi, F., Heath, R. W., Lozano, A., Marzetta, T. L. & Popovski, P. Five disruptive technology directions for 5G. *IEEE Communications Magazine* **52,** 74–80 (May 2014).
3. Ferrus, R., Sallent, O., Pérez-Romero, J. & Agustí, R. Management of Network Slicing in 5G Radio Access Networks. *Functional Framework and Information Models* (2018).

4. Wu, J., Wang, M., Chan, Y. C., Wong, E. W. M. & Kim, T. Performance Evaluation of 5G mmWave Networks with Physical-Layer and Capacity-Limited Blockingm. *IEEE 21st International Conference on High Performance Switching and Routing (HPSR),* 1–6 (May 2020).
5. Artalejo, J. R. & Gomez-Corral, A. *Retrial Queueing Systems* 318 pp. (Springer Berlin, 2008).
6. Falin, G. & Templeton, J. *Retrial Queues* 320 pp. (Taylor & Francis, 1997).
7. Phung-Duc, T. Retrial Queueing Models: A Survey on Theory and Applications. *Stochastic Operations Research in Business and Industry,* 1–26 (May 2017).
8. Gelenbe, E. Product-form queueing networks with negative and positive customers. *Journal of Applied Probability* **7,** 656–663 (1991).
9. Gelenbe, E., Glynn, P. & Sigman, K. Queues with Negative Arrivals. *J. Appl. Probab* **28,** 245–250 (1991).
10. Gelenbe, E. G-networks with signals and batch removal. *Probability in the Engineering and Informational Sciences Vol. 7,* 335–343 (1993).
11. Do, T. Bibliography on G-networks, negative customers and applications. *Mathematical and Computer Modelling,* 205–212 (2011).
12. Caglayan, M. G-networks and their applications to machine learning, energy packet networks and routing: introduction to the special issue. *Probability in the Engineering and Informational Sciences* **31,** 381–395 (2017).
13. Bocharov, P. & Vishnevsky, V. G-networks: development of the theory of multiplicative networks. Russian. *Automation and Telemechanics Vol. 5,* 70–74 (2003).
14. Shin, Y. Multi-server retrial queue with negative customers and disasters. *Queueing Syst,* 223–337 (2007).
15. Pechinkin, A. & Razumchik, R. A Method for Calculating Stationary Queue Distribution in a Queuing System with Flows of Ordinary and Negative Claims and a Bunker for Superseded Claims. *J. Commun. Technol. Electron.* **57,** 882–891 (2012).
16. Matalytski, M. & Naumenko, V. Analysis of the queueing network with a random bounded waiting time of positive and negative customers at a non-stationary regime. *Journal of Applied Mathematics and Computational Mechanics* **16,** 97–108 (2017).
17. Liu, T., Hsu, H. & Chang, F. Multi-Server Two-Way Communication Retrial Queue Subject to Disaster and Synchronous Working Vacation. *Algorithms* **18,** 24 (May 2025).
18. Melikov, A., Poladova, L., Edayapurath, S. & Sztrik, J. Single-Server Queuing-Inventory Systems with Negative Customers and Catastrophes in the Warehouse. *Mathematics* **11** (2023).
19. Lisovskaya, E., Fedorova, E., Salimzyanov, R. & Moiseeva, S. Resource Retrial Queue with Two Orbits and Negative Customers. *Mathematics* **10,** 321 (May 2022).
20. Meloshnikova, N. & Fedorova, E. *Asymptotic Analysis of a multiserver RQ-system with disasters in a service block under heavy load* Russian. in *Information and telecommunication technologies and mathematical modeling of high-tech systems: Proceedings of the All-Russian conference with international participation* (RUDN, Moscow, Russian Federation, April 7–11, 2025), (in print).
21. Meloshnikova, N. *Study Retrial Queue $M|M|N$ with negative calls under heavy load* Russian. in *Proceedings of Tomsk State University. Physics and Mathematics Series: Mathematical and Software Support of Information, Technical, and Economic Systems: Materials of the IX International Scientific Conference.* **307** (Tomsk State University Publishing House, Tomsk, Russian Federation, May 26–28, 2021), 165–168.

Information about the authors

Meloshnikova, Natalya P.—PhD-student, Junior researcher of Laboratory of queueing theory and teletraffic theory of National Research Tomsk State University (e-mail: meloshnikovana@gmail.com, phone: +7 (913) 213-83-71, ORCID: 0000-0002-8708-124X, ResearcherID: MTF-1866-2025, Scopus Author ID: 58304893200)

Fedorova, Ekaterina A.—PhD in Physical and Mathematical Sciences, Associate Professor of Department of Probability Theory and Mathematical Statistic of National Research Tomsk State University (e-mail: ekat_fedorova@mail.ru, phone: +7 (952) 801-93-94, ORCID: 0000-0001-8933-5322, ResearcherID: E-3161-2017, Scopus Author ID: 56439120600)

УДК 519.872
PACS 07.05.Tp
DOI: 10.22363/2658-4670-2025-33-2-144-156 EDN: BMHYOY

Асимптотический анализ многолинейной RQ-системы с π-поражением в условии большой загрузки

Н. П. Мелошникова, Е. А. Фёдорова

Национальный исследовательский Томский государственный университет, пр. Ленина, д. 36, Томск, 634050, Российская Федерация

Аннотация. В работе исследуется многолинейная RQ-система с π-поражением как математическая модель облачных сервисов. На вход системы поступает простейший поток «положительных» заявок. В системе конечное число обслуживающих приборов, время обслуживания заявок на приборах распределено по экспоненциальному закону. Когда все приборы заняты, заявки поступающие в систему переходят на орбиту, где осуществляют случайную задержку. После осуществления задержки, заявки с орбиты обращаются к блоку обслуживания согласно политике множественного доступа. Также в систему поступает поток так называемых «отрицательных» заявок. Отрицательная заявка не нуждается в обслуживании: при поступлении она удаляет случайное число обслуживаемых заявок. Для рассматриваемой модели записаны уравнения Колмогорова в стационарном режиме. Предлагается метод асимптотического анализа в условии большой загрузки для нахождения стационарного распределения вероятностей числа заявок на орбите. Представлены результаты численного анализа.

Ключевые слова: математическое моделирование, система массового обслуживания с повторными вызовами, отрицательные заявки, асимптотический анализ, большая загрузка

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 157–171
http://journals.rudn.ru/miph

Research paper

UDC 378.1:519.876.5:519.872
PACS 07.05.Tp
DOI: 10.22363/2658-4670-2025-33-2-157-171

EDN: MMCFUG

Business process analysis of university admissions: Combining TM Forum's eTOM framework, discrete-event simulation, and queuing theory

Konstantin M. Terentyev[1], Leyla D. Abuzyarova[1],
Irina A. Kochetkova[1, 2], Konstantin E. Samouylov[1, 2]

[1] RUDN University, 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

[2] Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, 44-2 Vavilova St, Moscow, 119133, Russian Federation

Abstract. The increasing complexity of university admissions requires efficient, standardized processes to manage large volumes of applications and changing regulatory requirements. To address this, the paper applies the TM Forum's Business Process Framework (eTOM) from the telecommunications industry, a standard for modeling and optimizing academic admissions workflows. Using RUDN University as a case study, the entire admissions process is formalized into a hierarchical model that aligns with the eTOM level 2 processes. The approach integrates discrete-event simulation (DES) and queueing network analysis, providing detailed process modeling and analytical solutions for assessing the average execution time. DES replicates the dynamic interactions between applicants and staff. Queueing analysis provides mathematical model to analyze the average execution times for each step in the process. Together, these techniques help optimize the admissions process and ensure efficient management of large volumes of applications. Through this approach, we aim to streamline processes, increase transparency, and support digital transformation efforts within universities.

Key words and phrases: university admissions, business process, TM Forum, business process framework, eTOM, discrete-event simulation, DES, queuing network, RUDN University

For citation: Terentyev, K. M., Abuzyarova, L. D., Kochetkova, I. A., Samouylov, K. E. Business process analysis of university admissions: Combining TM Forum's eTOM framework, discrete-event simulation, and queuing theory. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 157–171. doi: 10.22363/2658-4670-2025-33-2-157-171. edn: MMCFUG (2025).

1. Introduction

The organization of university admissions represents a complex and challenging task that requires the integration of various administrative, academic, and regulatory processes [1]. This process involves several stages, including document verification, entrance exams, applicant ranking, and final enrollment, all governed by strict deadlines and compliance requirements. With the advent of digital platforms like Public Services Portal of the Russian Federation, universities face increasing pressure to align their internal processes with these government platforms [2]. This creates a dual imperative for both efficiency and transparency in the admissions process. However, due to the lack of standardized methods for modeling and optimizing workflows, different institutions often adopt fragmented approaches that fail to scale during high-volume admission periods.

The need to modernize admissions processes is emphasized by the increasing competition among universities for students [3]. For instance, at RUDN University, delays in document verification during the 2023 admissions cycle resulted in a 22% dropout rate among applicants, directly affecting institutional revenue and reputation. Globally, similar issues persist, with manual processes and isolated departmental operations contributing to inefficiencies. A 2022 UNESCO report found that 65% of universities in emerging economies lack digitalized admission systems, leading to extended processing times and dissatisfied applicants [4]. Addressing these challenges is crucial not only for institutional competitiveness but also for integrating higher education into broader digital transformation initiatives [5].

Current approaches to admissions optimization, such as BPMN (Business Process Model and Notation) for workflow visualization or retrospective statistical analyses, offer limited predictive power for complex, high-volume scenarios. While industry frameworks like ITIL (Information Technology Infrastructure Library) are effective in IT service management, they lack domain-specific adaptations for educational institutions [6]. Crucially, these methods do not provide quantitative tools for identifying real-time bottlenecks or allocating resources during peak loads. Moreover, the lack of standardized reference processes hinders cross-institutional benchmarking and perpetuates inefficiencies.

To address these gaps, our study combines three complementary methodologies: the TM Forum's Business Process Framework (also known as eTOM) [7], a hierarchical process map designed for telecommunications operations [8–10]; discrete-event simulation (DES), a computational technique that models process flows as sequences of events; and queuing network analysis, a mathematical framework that quantifies service delays, resource utilization, and bottlenecks [11, 12]. By adapting eTOM for university admissions, we aim to bridge the gap between educational process engineering and advanced computational modeling [13, 14]. This study aims to establish a unified reference framework and provide actionable insights for optimizing admissions processes based on data from RUDN University [15–17].

The main contributions of our study are as follows:

- An adaptation of the eTOM framework for university admissions, which maps activities of the university admissions process to level 2 and 3 eTOM processes.
- A DES model that allows for scenario testing under different load conditions, such as applicant surges.
- A queuing network model that can be used for analytical analysis of the university admissions process, particularly during peak periods.

The rest of the paper is organized as follows. Section 2 deconstructs RUDN University's current admissions process and identifies potential pain points. Section 3 aligns these processes with eTOM, providing a framework for best practices. Section 4 describes the simulation design and analysis. Section 5 presents the queuing model. Section 6 concludes with a discussion of policy implications, limitations, and future directions for research.

2. Business process of university admissions

The organization of university admissions is a critical and complex process that requires the seamless integration of various administrative, academic, and regulatory processes. The goal of this process is to evaluate and admit applicants while ensuring compliance with strict deadlines, standards, and institutional priorities. Drawing on the example of RUDN University, this section will outline the general structure of the admissions process. This structure can be applied to higher education institutions at various levels (bachelor's, master's, and doctoral programs) and with different funding models (public – state-funded or private – contract-based).

The admissions process begins with document submission and verification. Applicants provide essential materials, such as academic records, identification documents, and test scores. For Russian institutions, this includes the Unified State Exam (USE) or internal entrance exams specific to the university. Admission officers carefully review these documents to verify their authenticity, completeness, and compliance with program requirements. Any discrepancies or missing information trigger notifications to the applicant, requiring revisions or additional documentation. This stage can be time-consuming, as manual verification of physical documents is still common, particularly in institutions without fully digital processes. Delays at this step can impact subsequent stages, creating bottlenecks in the admissions timeline.

After document verification, applicants may be required to take additional assessments, such as university-specific entrance exams, for competitive programs like medicine, engineering, or the creative arts. For programs that rely on the USE, results are automatically retrieved from federal databases. Internal exams are administered and graded by departmental faculty, and the outcomes are compiled into ranked lists that determine eligibility for state-funded or contract-based enrollment. A critical challenge at this stage is synchronizing data from different sources, including federal systems for USE scores and internal databases for exam results. Manual updates to spreadsheets with applicant information can also cause discrepancies and delays in data integration, leading to inaccuracies in rankings and disadvantages for applicants. These issues can strain institutional credibility.

The final stage, eligibility evaluation and enrollment, involves formalizing admission decisions and executing administrative procedures. Successful candidates are notified of their placement and asked to submit original documents in order to confirm their enrollment. For candidates funded by the state, this process is often straightforward, depending on meeting the published score thresholds. However, contract-based candidates need to go through additional steps, such as negotiating and signing tuition agreements, processing payments, and issuing enrollment orders. These orders are then published on institutional platforms and sent to federal education systems to finalize the applicant's status as a student. Despite its clear process, this phase can be hindered by last-minute withdrawals, delayed submissions of documents, and resource constraints, especially during peak enrollment times.

While the admissions process is well-structured, its implementation reveals several systemic vulnerabilities that are common across higher education institutions. Manual workflows: A heavy reliance on paper-based document checks and manual data entry lengthens processing times and increases error rates. Disconnected IT systems: Exam results, applicant profiles, and financial agreements are stored in separate systems, making it difficult to track the end-to-end process. Annual applicant surges: Exceeding 40,000 applicants at RUDN University overwhelm existing staff and infrastructure, leading to delays and dissatisfaction among applicants. Regulatory requirements: Evolving privacy laws and reporting standards demand continuous adjustments to the admissions process, but often without corresponding resource allocations for compliance.

These challenges highlight the need for a comprehensive re-evaluation of admission processes, balancing efficiency and transparency. The following sections of this paper address these issues through a structured approach, combining process standardization, computational, and mathematical modeling. By breaking down the process into its individual stages, this work aims to provide a repeatable framework for universities to navigate the complexities of contemporary student recruitment and enrollment.

3. Applying the eTOM framework

In this section, we formalize the university admissions process using the TM Forum's eTOM, a hierarchical framework for standardizing business operations. By mapping RUDN University's workflows to the eTOM's level 2-3 processes, we have established a reference model for benchmarking and optimizing cross-institutional operations.

3.1. Process with detailed activities

The admissions process consists of 14 steps, each with a unique identifier to align with the eTOM hierarchical structure (see Figure 1). These steps are numbered and described below, along with concise titles that are used in Table 1.

- Step 1: Online Admission Portal. Prospective students access RUDN University's online admission portal to create a personal account, submit an application, and track the progress. This digital platform provides guidelines, application deadlines, and automatic validation for required documents such as diplomas and identification.
- Step 2: Student Guidance & Career Counselling. Specialized advisors assist students in choosing programs that align with their academic profile and career goals. They help clarify admission criteria, scholarship opportunities, and pathways after graduation.
- Step 3: Application Submission & Data Verification. Students upload personal information and relevant documents through the platform. Admissions staff verify the completeness of the data and initiate automatic background checks (such as plagiarism detection for admission essays).
- Step 4a: Document Verification and Notification (Manual). After completing all steps, students receive a notification about the status of their application. If everything is in order, they are invited to the next stage of the admission process. Officers manually verify physical or scanned documents (such as diplomas and exam certificates) to ensure their authenticity. Applicants are notified automatically about acceptance or rejection via email or SMS.
- Step 4b: Entrance Exam Scheduling. If a program requires internal exams (for example, medicine), the system creates personalized schedules based on applicant preferences and faculty availability.
- Step 5a: Exam Administration. Exams are proctored either on-site or remotely. Results are digitized and stored in the RUDN University database. Applicants are notified about the results within 48 hours through automated alerts.
- Step 5b: Result Integration. The USE results are retrieved from federal systems and combined with internal scores to create a unified applicant profile.
- Step 6a: Eligibility Evaluation. Applicants are ranked based on their composite scores using algorithms. Their scores are compared to program-specific admission thresholds for budget and contracts.
- Step 6b: Document Request. Candidates qualifying for budget seats are required to submit original diplomas within 7 days. If they fail to comply, they will be shifted to contract-based pools.
- Step 7: Contract Offer for Non-Qualified Applicants. Applicants who do not meet the budget threshold will receive formal offers for enrollment based on a contract, detailing tuition fees, payment plans, and academic requirements.
- Step 8: Contract Review & Signature. Legal teams will draft contracts, which applicants will review and electronically sign via a secure platform. Digital signatures will be timestamped and archived.

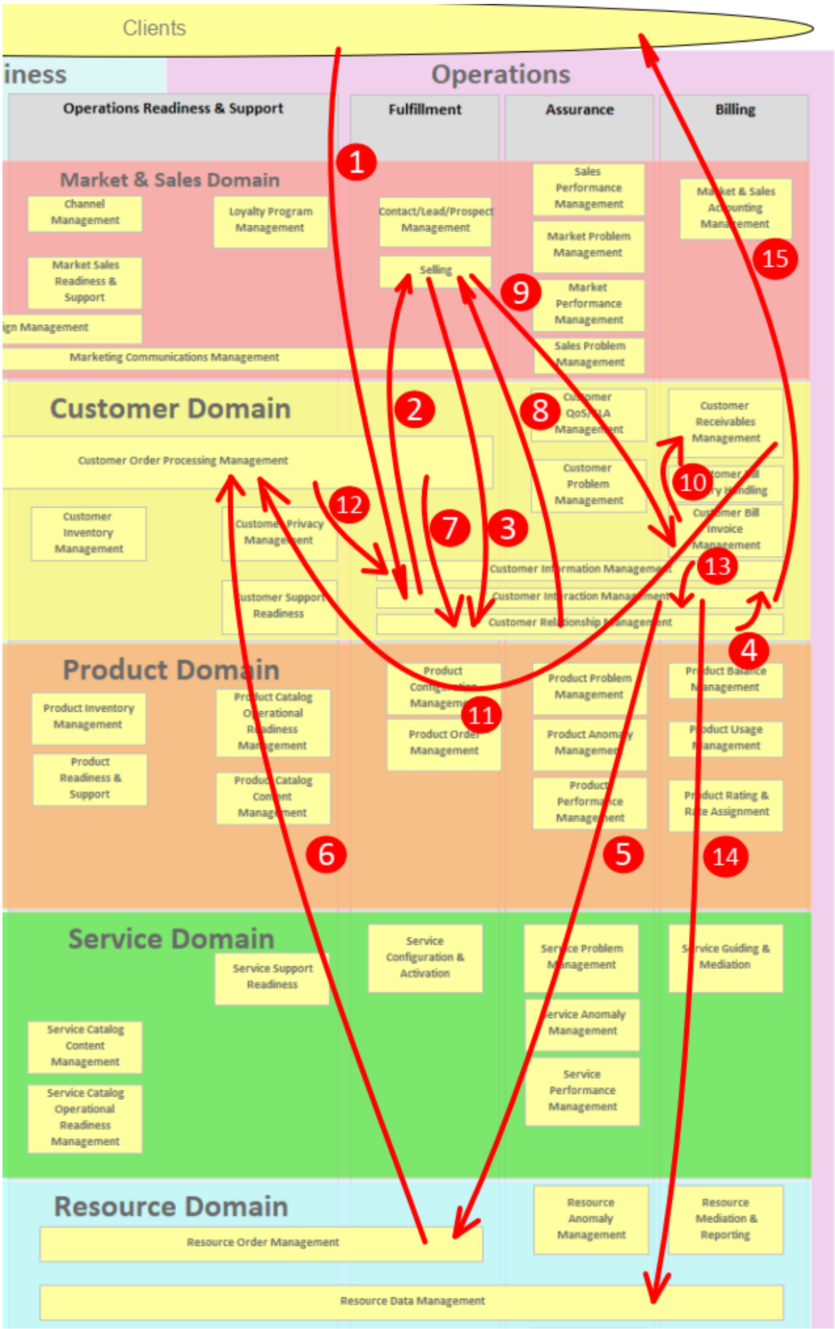

Clients
iness
Operations
Operations Readiness & Support
Fulfillment
Assurance
Billing
Market & Sales Domain
Channel Management
Loyalty Program Management
Contact/Lead/Prospect Management
Sales Performance Management
Market & Sales Accounting Management
Market Sales Readiness & Support
Selling
Market Problem Management
Market Performance Management
Sales Problem Management
ign Management
Marketing Communications Management
Customer Domain
Customer QoS/SLA Management
Customer Receivables Management
Customer Order Processing Management
Customer Problem Management
Customer Bill Inquiry Handling
Customer Bill Invoice Management
Customer Inventory Management
Customer Privacy Management
Customer Information Management
ustomer Support Readiness
Customer Interaction Management
Customer Relationship Management
Product Domain
Product Configuration Management
Product Problem Management
Product Balance Management
Product Inventory Management
Product Catalog Operational Readiness Management
Product Order Management
Product Anomaly Management
Product Usage Management
Product Readiness & Support
Product Catalog Content Management
Product Performance Management
Product Rating & Rate Assignment
Service Domain
Service Support Readiness
Service Configuration & Activation
Service Problem Management
Service Guiding & Mediation
Service Anomaly Management
Service Catalog Content Management
Service Catalog Operational Readiness Management
Service Performance Management
Resource Domain
Resource Order Management
Resource Anomaly Management
Resource Mediation & Reporting
Resource Data Management
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

Figure 1. University admissions detailed activities on the eTOM framework

Table 1

Mapping of university admissions detailed activities to eTOM level 2 processes

No. step	eTOM level 2 process	University admissions detailed activities
1	Customer Interaction Management	Online Admission Portal
2	Selling	Student Guidance & Career Counselling
3	Customer Relationship Management	Application Submission & Data Verification
4a	Customer Interaction Management	Document Verification and Notification (Manual)
4b	Customer Interaction Management	Entrance Exam Scheduling
5a	Resource Order Management	Exam Administration
5b	Resource Order Management	Result Integration
6a	Customer Order Processing	Eligibility Evaluation
6b	Customer Order Processing	Document Request
7	Customer Relationship Management	Contract Offer for Non-Qualified Applicants
8	Selling	Contract Review & Signature
9	Customer Bill Management	Invoice Creation & Payment
10	Customer Receivables Management	Admitted Student List Preparation
11	Customer Order Processing	Enrollment Preparation
12	Customer Information Management	Enrollment Finalization
13	Customer Interaction Management	Enrollment Publication
14	Resource Data Management	Enrollment Archiving

- Step 9: Invoice Creation & Payment. Systems will automatically generate invoices reflecting the contractual terms, which will be processed through integrated banking gateways with real-time updates on payment status.
- Step 10: Admitted Student List Preparation. Staff will compile final lists of enrolled students, generating unique enrollment orders for federal reporting purposes.
- Steps 11-14: Publication & Archiving of Enrollment Records. Enrollment records will be published and archived according to the established process. Orders are published on the RUDN University website, emailed to applicants, and stored in accordance with data retention policies.

3.2. Mapping to eTOM framework

Figure 1 shows the end-to-end alignment of RUDN University's admissions process with the eTOM's level 2 processes. Table 1 provides a detailed mapping, showing how each university activity (Steps 1-14) corresponds to eTOM's standardized categories (for example, Customer Interaction Management and Resource Order Management).

Note that Steps 4a-5b (Document Verification to Exam Result Integration) correspond to eTOM's Resource Order Management, highlighting the importance of resource coordination. Steps 7-9 (Contract Proposal to Invoice Generation) reflect the focus on Customer Relationship Management, which is centered around the applicants. Parallel processes such as budget/contract enrollment are unified under the hierarchical levels of eTOM, allowing for scalable optimization.

4. Discrete-event simulation model

In this section, we describe the development and results of a DES model that was designed to analyze the efficiency of the admissions process at RUDN University. By translating the 14-step process of the institution into a computationally manageable model, the simulation identified bottlenecks, quantified delays, and proposed data-driven optimization strategies.

4.1. Process with aggregated activities

To strike a balance between granularity and computational feasibility, the original eTOM-aligned process (see Section 3) was simplified into seven aggregated stages (Table 2). This simplification prioritized tasks that depend on staff, excluding automated steps or those that are driven by applicants (e.g., document uploads and email notifications). For instance, Stages 1-4a of the original process, which involved document verification and notifications to applicants, were combined into Stage A (Document Processing) to reflect the shared responsibility of admission officers.

The normalized BPMN model (Figure 2) was implemented using the BIMP platform [18], selected for its ability to simulate stochastic events and scale to 10,000 concurrent users, which represents 25% of RUDN University's 2023 applicant volume due to free-tier constraints. Input parameters were derived from historical data. Activity durations followed normal or exponential distributions, calibrated to 2023 operational timestamps (Table 3). Branching probabilities (Table 4) governed decision points, such as document verification success (92.61%) or budget eligibility (25.6%). Resource limits mirrored actual staffing levels, with 15 admissions officers and 8 commercial team members.

4.2. Numerical results

The simulation replicated a scaled-down version of RUDN University's 2023 admission process, revealing critical inefficiencies. The average admission process took 14.2 days, and Stage A (Document Processing) accounted for 68% of the delays (Figure 3). Manual verification of physical documents, modeled as $\mathcal{N}(15, 9)$ minutes per application, created queues during peak periods. Non-working days exacerbated delays, increasing the total cycle time by 22% (Figure 4). Admission officers operated at 89% capacity during peak loads, resulting in applicant queues up to 1,240 people (Figure 5). In contrast, commercial teams remained underutilized (52%), highlighting imbalances in workforce allocation. Stages A (Document Processing) and D (Original Document Request) emerged as primary bottlenecks. At Stage A, 45% of applicants experienced wait times exceeding 2 hours due to manual checks. At Stage D, delays were caused by applicants' delayed submission of original documents.

Table 2

University admissions aggregated activities for simulation

Stage	Aggregated activity	No. steps (Table 1)
A	Document Processing: Process and verify documents	1-4a
B	Entrance Exam Management: Prepare lists and conduct the entrance exam	4b-5a
C	Exam Result Integration: Request and process the results of the USE through Super Service	5b-6a
D	Original Document Request: Request for the original documents of previous education	6b
E	Contract-Based Enrollment: Offer the option of enrolling on a contractual basis	7
F	Contract Finalization: Sign the contract and pay the bill	8-10
G	Enrollment Publication: Form and publish an order for enrollment	11-14

Table 3

University admissions aggregated activities duration

Stage	Resource	Duration	SLA threshold
A	Admissions Officer	$\mathcal{N}(15, 9)$ min	60 min
B	Admissions Officer	$\mathcal{N}(120, 15)$ min	300 min
C	Admissions Officer	$\mathcal{N}(15, 25)$ min	60 min
D	Admissions Officer	Exp(15) min	180 min
E	Commercial Team	$\mathcal{N}(10, 4)$ min	30 min
F	Commercial Team	$\mathcal{N}(5, 2.25)$ min	15 min
G	Admissions Officer	60 min	120 min

The simulation results suggested three targeted strategies to improve operations. First, automated document verification could be implemented to replace manual checks with standardized digital workflows. This would reduce Stage A processing time by 40% and lower the average duration to 9 minutes. This adjustment would shorten the overall cycle time by approximately 4 days. Second, dynamic staff reallocation could be used to shift 3 admissions officers from Stage G (enrollment publication) to Stage A during peak periods. This would reduce officer utilization by 72% and alleviate queues by 35%. Third, e-signature integration could be implemented in Stage F to digitize contract signing. This would cut processing time to 2 minutes and accelerate contract finalization by 62%. It would also reduce commercial team idle time.

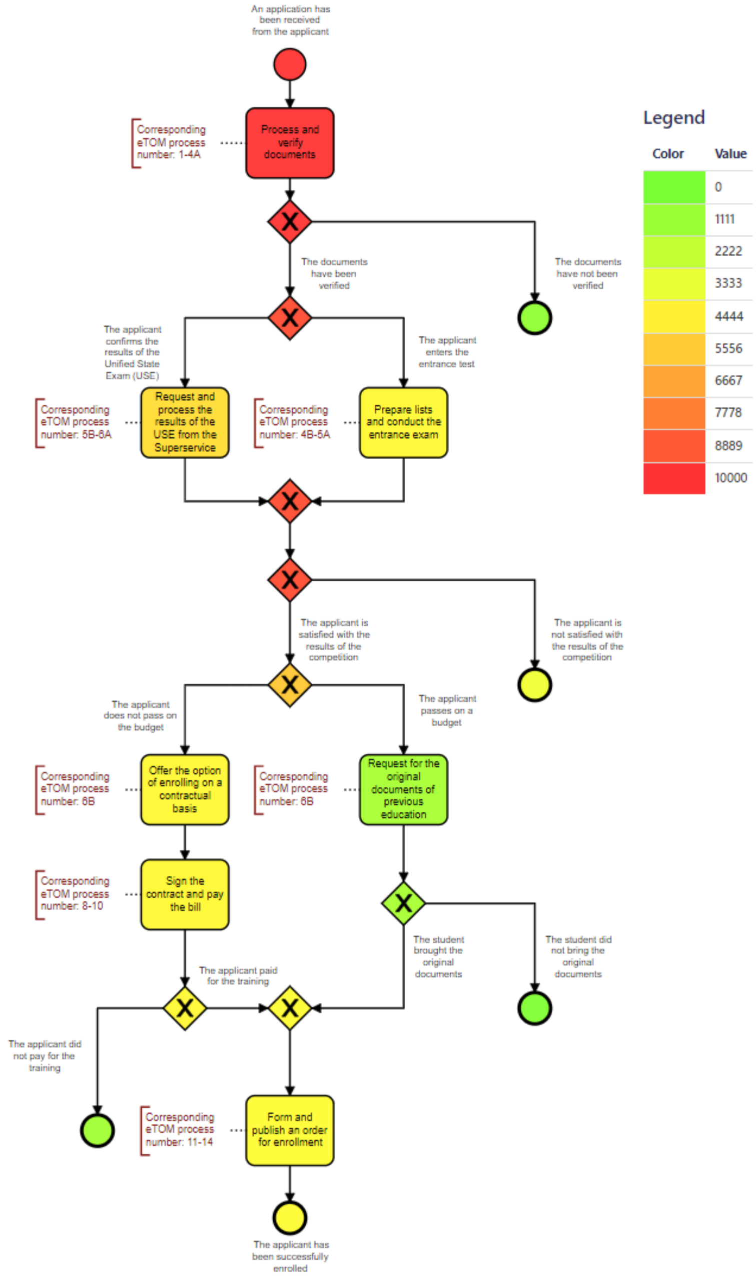

An application has been received from the applicant
Corresponding eTOM process number: 1-4A
Process and verify documents
The documents have been verified
The documents have not been verified
The applicant confirms the results of the Unified State Exam (USE)
The applicant enters the entrance test
Corresponding eTOM process number: 5B-6A
Request and process the results of the USE from the Superservice
Corresponding eTOM process number: 4B-6A
Prepare lists and conduct the entrance exam
The applicant is satisfied with the results of the competition
The applicant is not satisfied with the results of the competition
The applicant does not pass on the budget
The applicant passes on a budget
Corresponding eTOM process number: 6B
Offer the option of enrolling on a contractual basis
Corresponding eTOM process number: 6B
Request for the original documents of previous education
Corresponding eTOM process number: 8-10
Sign the contract and pay the bill
The student brought the original documents
The student did not bring the original documents
The applicant paid for the training
The applicant did not pay for the training
Corresponding eTOM process number: 11-14
Form and publish an order for enrollment
The applicant has been successfully enrolled
Legend
Color
Value
0
1111
2222
3333
4444
5556
6667
7778
8889
10000

Figure 2. University admissions process for simulation

5. Queuing model

We formalize the university admissions process as an open Jackson network, leveraging queuing theory to derive analytical expressions for average execution time. The model extends the DES (Section 4) by providing a mathematical framework for stability analysis and scenario testing.

Table 4

University admissions aggregated activities branching probabilities

Gate	Decision node	Scenario 1	Scenario 2
1	Document verification	Verified (92.61%)	Not verified (7.39%)
2	Exam preference	Internal Exam (45.53%)	USE confirmation (54.47%)
3	Competition satisfaction	Dissatisfied (84.55%)	Satisfied (15.45%)
4	Budget eligibility	Budget admitted (25.6%)	Not admitted (74.4%)
5	Document submission	Submitted (77.9%)	Not submitted (22.1%)
6	Payment status	Paid (69.36%)	Unpaid (30.64%)

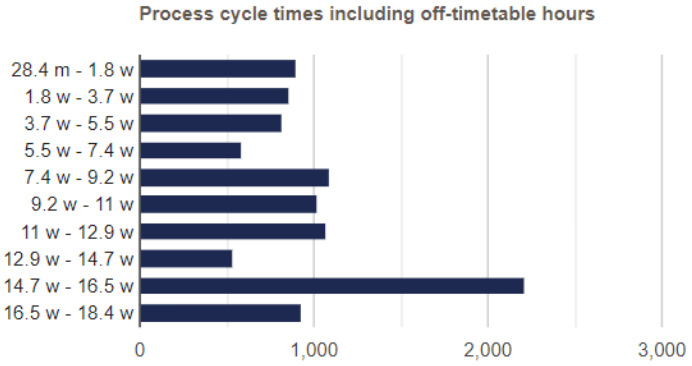


Figure 3. University admissions duration, including non-working hours

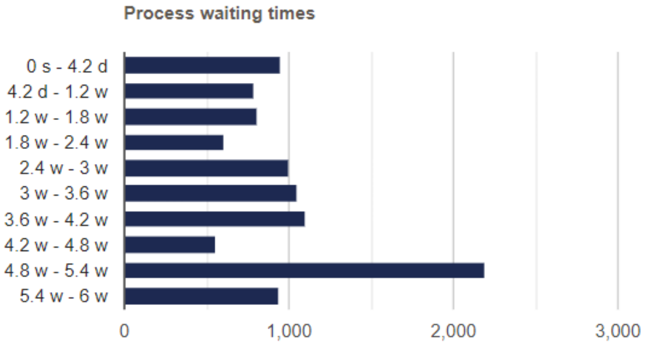


Figure 4. University admissions waiting times before activities start

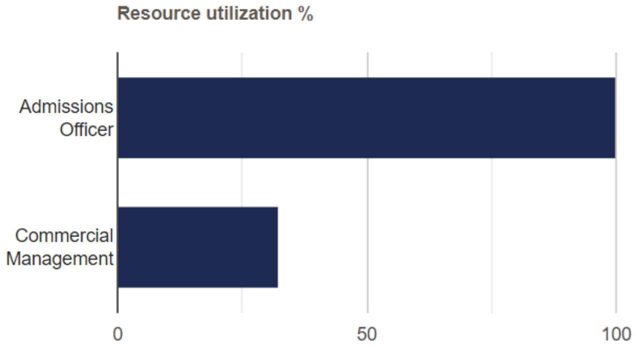


Figure 5. Resource capacity

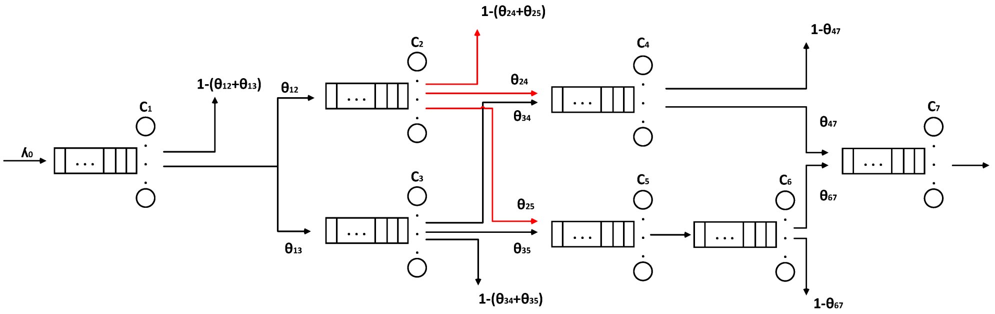


Figure 6. University admissions process as a queuing network

The admission workflow is modeled as a network of $m = 7$ service nodes corresponding to the stages in Table 2. See Figure 6 for a visual representation. For example, node 1 is Document Processing and node 7 is Enrollment Publication. Applicants enter the system at node 1 following a Poisson process with a rate of λ_0. Each node processes applications at a rate μ_i, derived from empirical data in Table 3. After processing at node i, applications move to node j with probability θ_{ij}, as defined in routing matrix (see Table 5).

The effective arrival rate λ_i at node i is governed by the traffic equations:

$$\begin{aligned}
&\lambda_1 = \lambda_0,\\
&\lambda_2 = \lambda_0\theta_{12},\\
&\lambda_3 = \lambda_0\theta_{13},\\
&\lambda_4 = \lambda_0(\theta_{12}\theta_{24} + \theta_{13}\theta_{34}),\\
&\lambda_5 = \lambda_0(\theta_{12}\theta_{25} + \theta_{13}\theta_{35}),\\
&\lambda_6 = \lambda_0(\theta_{12}\theta_{25} + \theta_{13}\theta_{35}),\\
&\lambda_7 = \lambda_0(\theta_{12}\theta_{24}\theta_{47} + \theta_{13}\theta_{34}\theta_{47} + \theta_{12}\theta_{25}\theta_{67} + \theta_{13}\theta_{35}\theta_{67}).
\end{aligned}$$

Table 5

Routing matrix for queuing network

	0	1	2	3	4	5	6	7
0	0	1	0	0	0	0	0	0
1	$1-(\theta_{12}+\theta_{13})$	0	θ_{12}	θ_{13}	0	0	0	0
2	$1-(\theta_{24}+\theta_{25})$	0	0	0	θ_{24}	θ_{25}	0	0
3	$1-(\theta_{34}+\theta_{35})$	0	0	0	θ_{34}	θ_{35}	0	0
4	$1-\theta_{47}$	0	0	0	0	0	0	θ_{47}
5	0	0	0	0	0	0	1	0
6	$1-\theta_{67}$	0	0	0	0	0	0	θ_{67}
7	1	0	0	0	0	0	0	0

We consider that each node operates as an Erlang–C model – $M|M|C_i|\infty$ queue. The network is stable if $\rho_i < C_i$ for all nodes i, where $\rho_i = \frac{\lambda_i}{\mu_i}$ is the total offered traffic and C_i is the number of servers at node i. The average time W_i spent at node i is calculated by the formula

$$\begin{aligned} W_i &= \frac{\rho_i^{C_i}}{C_i!}\frac{\rho_i C_i}{(C_i-\rho_i)^2}\cdot\left(\sum_{n=0}^{C_i-1}\frac{\rho_i^n}{n!}+\frac{\rho_i^{C_i}}{C_i!}\frac{C_i}{C_i-\rho_i}\right)^{-1}+\frac{1}{\mu_i}, && C_i > 1,\\ W_i &= \frac{1}{\mu_i(1-\rho_i)}, && C_i = 1. \end{aligned}$$

The total average execution time aggregates delays across all nodes

$$T = W_1 + \theta_{12}W_2 + \theta_{13}W_3 + (\theta_{24}+\theta_{34})W_4 + (\theta_{25}+\theta_{35})W_5 + W_6 + (\theta_{47}+\theta_{67})W_7.$$

6. Conclusion

Motivated by the need to standardize and optimize university admissions processes in the face of increasing digitalization, this study adapted the TM Forum's eTOM framework, originally designed for telecommunications, to academic operations. By integrating real-world admission data from 2023 and regulatory requirements, we evaluated the effectiveness of the framework through a hybrid methodology that combined DES and queuing theory.

We have shown that integrating eTOM with simulation modeling identifies critical bottlenecks such as manual document verification (Stage A), which is responsible for 68% of delays. The queuing network model predicts an average cycle time. Digitizing document checks and contract signing (e.g., e-signatures) could reduce processing times, particularly during peak loads. Reassigning staff members to Stage A could decrease officer utilization and shorten queues. Automating contract signing (Stage F) could be useful under fluctuating workloads.

For future directions, we consider developing a centralized database for historical admission metrics for predictive analytics, as well as creating software to automatically generate queuing models from BPMN diagrams to streamline bottleneck analysis. The proposed framework could be adapted to other university workflows, such as dormitory allocation and academic advising, ensuring standardization and transparency.

Author Contributions: Conceptualization, supervision, K.S.; methodology, I.K.; validation, resources, K.T.; software, visualization, L.A.; formal analysis, investigation, writing—original draft preparation, K.T. and L.A.; writing—review and editing, I.K. and K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing is not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kiaghadi, M. & Hoseinpour, P. University admission process: A prescriptive analytics approach. *Artificial Intelligence Review* **56,** 233–256. doi:10.1007/s10462-022-10171-y (2023).
2. Roudnitski, A. University admissions mechanism: Theoretical analysis with application to Russian universities. *HSE Economic Journal* **27,** 594–610. doi:10.17323/1813-8691-2023-27-4-594-610 (2023).
3. Shao, L., Levine, R. A., Hyman, S., Stronach, J. & Fan, J. A combinatorial optimization framework for scoring students in university admissions. *Evaluation Review* **46,** 296–335. doi:10.1177/0193841X221082887 (2022).
4. For Europe, U. N. E. C. *Sustainable Development Goals* https://w3.unece.org/SDG/en. Accessed: 2025-03-01.
5. Tampieri, A. University admission: Is achievement a sufficient criterion? *B.E. Journal of Economic Analysis and Policy* **24,** 799–833. doi:10.1515/bejeap-2023-0260 (2024).
6. Mengash, H. A. Using data mining techniques to predict student performance to support decision making in university admission systems. *IEEE Access* **8,** 55462–55470. doi:10.1109/ACCESS.2020.2981905 (2020).
7. Forum, T. *Business Process Framework (eTOM)* https://www.tmforum.org/oda/business-architecture-framework/process-framework-etom/. Accessed: 2025-03-01.
8. Horiuchi, S. & Tayama, K. Latest activities in TM Forum. *NTT Technical Review* **19,** 75–80. doi:10.53829/ntr202107gls (2021).
9. Suzuki, A., Kashibuchi, K. & Nakamura, T. Activities toward TM Forum Frameworx 17.0 and TM Forum Live! 2017 report. *NTT Technical Review* **16.** doi:10.53829/ntr201801gls (2023).
10. Horiuchi, S. & Tayama, K. Efforts by TM Forum, an operation standards organization. *NTT Technical Review* **21,** 53–58. doi:10.53829/ntr202307gls (2023).
11. Samouylov, K., Gaidamaka, Y. & Zaripova, E. Analysis of business process execution time with queueing theory models. *Communications in Computer and Information Science* **638,** 315–326. doi:10.1007/978-3-319-44615-8_28 (2016).
12. Yarkina, N., Popovskaya, N., Khalina, V., Gaidamaka, A. & Samouylov, K. Modeling end-to-end business processes of a telecom company with a BCMP queueing network. *Communications in Computer and Information Science* **800,** 279–296. doi:10.1007/978-3-319-68069-9_23 (2017).
13. Fernandes, C. & Cruz, R. A business processes model for the integration of over-the-top platforms in communications service providers operations. *Journal of Information Systems Engineering and Management* **6.** doi:10.29333/jisem/9570 (2021).
14. Dachyar, M., Zagloel, T. Y. M. & Saragih, L. R. Enterprise architecture breakthrough for telecommunications transformation: A reconciliation model to solve bankruptcy. *Heliyon* **6.** doi:10.1016/j.heliyon.2020.e05273 (2020).
15. University, R. *Admission committee* https://admission.rudn.ru/. Accessed: 2025-03-01.
16. University, R. *Electronic examination support system* https://esse.rudn.ru/?lang=en. Accessed: 2025-03-01.

17. Ivleva, M., Nezhnikova, E. & Safronova, N. The study of the impact of coaching on the efficiency of learning in the higher education. *RUDN Journal of Sociology* **24,** 523–538. doi:10.22363/2313-2272-2024-24-2-523-538 (2024).
18. Of Tartu, U. *BIMP simulator* https://bimp.cs.ut.ee/simulator/. Accessed: 2025-03-01.

Information about the authors

Terentyev, Konstantin M.—Ph.D. student of the Department of Probability Theory and Cyber Security, RUDN University; Director of the Department of Digitalization and Digital Transformation, RUDN University (e-mail: terentyev-km@rudn.ru, ORCID: 0009-0005-4379-3017)

Abuzyarova, Leyla D.—M.Sc. student of the Department of Probability Theory and Cyber Security, RUDN University (e-mail: d.abuzyarova@yandex.ru, ORCID: 0009-0007-0084-0201)

Kochetkova, Irina A.—Candidate of Physical and Mathematical Sciences, Assosiate Professor of the Department of Probability Theory and Cyber Security, RUDN University; Senior Researcher, Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (e-mail: kochetkova-ia@rudn.ru, ORCID: 0000-0002-1594-427X, ResearcherID: E-3806-2014, Scopus Author ID: 35332169400)

Samouylov, Konstantin E.—Professor, Doctor of Technical Sciences, Head of the Department of Probability Theory and Cyber Security, RUDN University; Senior Researcher, Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (e-mail: samuylov-ke@rudn.ru, ORCID: 0000-0002-6368-9680, ResearcherID: E-9966-2014, Scopus Author ID: 14009785000)

УДК 378.1:519.876.5:519.872
PACS 07.05.Tp
DOI: 10.22363/2658-4670-2025-33-2-157-171 EDN: MMCFUG

Анализ процесса приема в университет: применение карты TM Forum eTOM, имитационного моделирования и сети массового обслуживания

К. М. Терентьев[1], Л. Д. Абузярова[1], И. А. Кочеткова[1, 2], К. Е. Самуйлов[1, 2]

[1] Российский университет дружбы народов имени Патриса Лумумбы, ул. Миклухо-Маклая, д. 6, Москва, 117198, Российская Федерация

[2] Федеральный исследовательский центр «Информатика и управление» Российской академии наук, ул. Вавилова, д. 44, кор. 2, Москва, 119333, Российская Федерация

Аннотация. С каждым годом процессы поступления в университеты становятся все более сложными. Для управления большим количеством заявок и постоянно меняющимися требованиями законодательства необходимы эффективные методы. В статье используется модель бизнес-процессов Forum eTOM, которая изначально была разработана для телекоммуникационной отрасли, для моделирования и оптимизации приемной кампании. На примере Российского университета дружбы народов имени Патриса Лумумбы (РУДН) показано, как можно формализовать весь процесс поступления в виде иерархической модели, соответствующей уровням eTOM. Методология объединяет два подхода: дискретно-событийное моделирование позволяет детально анализировать динамические взаимодействия между абитуриентами и сотрудниками, а сеть массового обслуживания дает возможность оценить среднее время выполнения процесса. Проведен анализ реального сценария обработки заявок, учитывая ограничения ресурсов. Совместное использование этих методов помогает выявить узкие места, такие как ручная проверка документов, и предложить решения для их устранения. Предложенный подход способствует стандартизации процессов, повышает прозрачность операций и поддерживает цифровую трансформацию университетов. Его можно адаптировать для других учебных заведений, обеспечивая эффективное управление приемными кампаниями в условиях растущих требований к автоматизации и масштабируемости.

Ключевые слова: университет, приемная кампания университета, бизнес-процесс, TM Forum, карта процессов, eTOM, имитационное моделирование, сеть массового обслуживания, РУДН

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 172–183
http://journals.rudn.ru/miph

Research article

UDC 004.62:004.822
PACS 07.05.Mh, 07.05.Kf
DOI: 10.22363/2658-4670-2025-33-2-172-183

EDN: MGCVKV

Predictive diagnostics of computer systems logs using natural language processing techniques

Vladislav A. Kiriachek, Soltan I. Salpagarov

RUDN University, 6, Miklukho-Maklaya St, Moscow, 117198, Russian Federation

(received: December 7, 2024; revised: January 10, 2025; accepted: January 20, 2025)

Abstract. This study aims to develop and validate a method for predictive diagnostics and anomaly detection in computer system logs, using the Vertica database as a case study. The proposed approach is based on semi-supervised learning combined with natural language processing techniques. A specialized parser utilizing a semantic graph was developed for data preprocessing. Vectorization was performed using the fastText NLP library and TF-IDF weighting. Empirical validation was conducted on real Vertica log files from a large IT company, containing periods of normal operation and anomalies leading to failures. A comparative assessment of various anomaly detection algorithms was performed, including k-nearest neighbors, autoencoders, One Class SVM, Isolation Forest, Local Outlier Factor, and Elliptic Envelope. Results are visualized through anomaly graphs depicting time intervals exceeding the threshold level. The findings demonstrate high efficacy of the proposed approach in identifying anomalies preceding system failures and delineate promising directions for further research.

Key words and phrases: machine learning, natural language processing, log analysis, anomaly detection, predictive diagnostics

For citation: Kiriachek, V. A., Salpagarov, S. I. Predictive diagnostics of computer systems logs using natural language processing techniques. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 172–183. doi: 10.22363/2658-4670-2025-33-2-172-183. edn: MGCVKV (2025).

1. Introduction

Effective monitoring and analysis of events occurring in computer systems are key factors in ensuring their reliable, secure and uninterrupted operation. The main source of information on the functioning of such systems are log files - text files containing structured and unstructured data on a wide range of events, including normal activity, warnings, errors and anomalies. Due to the rapid growth of the volume of generated data, measured in millions and billions of lines daily, as well as the diversity of sources and log formats, their systematic manual analysis becomes a virtually impossible task even for teams of qualified specialists. Failure of critical computer systems, such as databases, can lead to the collapse of other systems that rely on them. For example, the operation of web analytics products (dashboards) depends on the operation of databases, and their shutdown due to a database failure costs companies millions in losses. This raises the problem of searching for anomalies in order to prevent failures in computer systems using their logs.

In this regard, the development of intelligent systems for automatic log analysis based on Natural Language Processing (NLP) and machine learning is of paramount importance. Such systems are capable of extracting structured information from unstructured text, identifying patterns and anomalies, and generating reports and alerts automatically. The use of NLP-techniques, in particular, text classification algorithms, clustering, entity and relationship extraction, as well as modern deep learning models based on transformers, opens up new opportunities for intelligent big data analysis in the IT sector.

Having a certain structure, log files contain information about various system events, such as errors, warnings, and other incidents. They record the time and date of the event, as well as its type or importance level, designated by special tags (e.g. `<INFO>`, `<ERROR>`, `<FATAL>`). In addition, logs contain a significant amount of volatile data, including hash-sums, process identifiers, network addresses, etc. This data can be generated dynamically and, as a rule, is not repeated in future records, which requires the use of special preprocessing methods for their normalization and anonymization before further analysis. The literature discusses many parsers built on various architectures, such as Drain [1], Spell [2], and others. However, to meet the requirements of production tasks, which impose additional restrictions in the form of the need for integration into existing software and the ability to flexibly configure individual system components, it was decided to develop our own specialized log parser.

Anomalies in computer systems are events characterized by outlier values of their features and sharply contrasting with typical modes of operation of such systems during periods of their normal operation. Anomalous behavior of systems is often rare and unpredictable, deviating from established patterns based on previous observations. Therefore, the developed anomaly detection approach is based on semi-supervised learning techniques together with NLP algorithms such as fastText [3] and TF-IDF, which does not require labeling of training data. It is only necessary to know the periods of normal, uninterrupted operation of the system.

An analysis of existing solutions in the subject area under consideration reveals a wide variety of approaches to solving the problem of anomaly detection in computer systems, among which a significant share is made up of techniques based on the supervised learning paradigm LogRobust [4], CNN [5] as well as semi-supervised learning techniques, such as DeepLog [6], LogAnomaly [7], LogBERT [8], PLELog [9] or unsupervised approaches, such as Logsy [10].

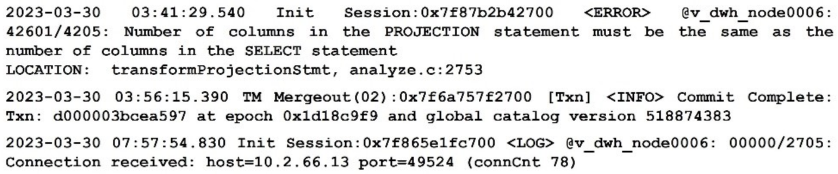
The most widely used ML architectures include Recurrent Neural Networks (RNN) [4, 6, 7, 11], Convolutional Neural Networks (CNN) [5, 12], Transformers (TF) [8, 10], Graph Neural Networks (GNN) [13], as well as approaches that can do without labeling, such as Autoencoders (AE) [11, 14], Variational Autoencoders (VAE) [12] and classical machine learning techniques such as One Class SVM [15], Isolation Forest [16], Local Outlier Factor [17], Elliptic Envelope [18], k-nearest neighbors, tested in this work.

The developed approach is based on several successive stages: log preprocessing, vectorization and anomaly detection (predictive diagnostics).

2. System description

2.1. Description of the source data format

Before we talk about log preprocessing, let's define the concept of a log. A log is a text file with a certain structure containing information about system events, such as errors, the time and date of these events, and the event tag itself (e.g. `<INFO>`, `<ERROR>`, `<FATAL>`). The logs of various computer systems contain a lot of variable information, such as hash-sums, process IDs, timestamps, etc.



```
2023-03-30   03:41:29.540   Init   Session:0x7f87b2b42700   <ERROR>   @v_dwh_node0006:
42601/4205: Number of columns in the PROJECTION statement must be the same as the
number of columns in the SELECT statement
LOCATION:  transformProjectionStmt, analyze.c:2753
2023-03-30 03:56:15.390 TM Mergeout(02):0x7f6a757f2700 [Txn] <INFO> Commit Complete:
Txn: d000003bcea597 at epoch 0x1d18c9f9 and global catalog version 518874383
2023-03-30 07:57:54.830 Init Session:0x7f865e1fc700 <LOG> @v_dwh_node0006: 00000/2705:
Connection received: host=10.2.66.13 port=49524 (connCnt 78)
```

Figure 1. Example of raw Vertica database log data

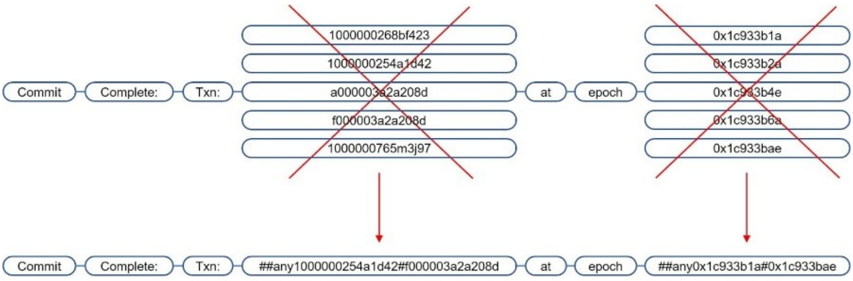


Figure 2. Parser diagram

In the Vertica database, considered in the work as a typical example of a computer system, the logs have a such structure. Examples of several events are shown in Figure 1. First comes information about the date and time of the event, then system information with the process id, the event tag is specified in curly brackets <...>, after which comes the text of the event itself, containing variable information, i.e. id, hosts, ports, etc. It is worth noting that most computer systems, not only databases, have a similar structure.

Log file sizes can be large enough for manual analysis. For example, in the Vertica database considered in the study, log files collected over one day have an average weight of about 400 MB and contain an average of over 10 million events. Thus, the size of the entire training set for 2 months of non-stop operation was 23 GB and over 830 million events.

2.2. Description of the proposed approach

Among the features typical for logs of any database, and not just the Vertica database considered in the work, it is worth noting the presence of SQL-queries in the logs themselves. This data can be useful, since a suboptimally written query can lead to problems in the operation of the database and even its failure. The use of existing log parsers is impossible, since they delete a lot of useful information, including SQL-queries, which was the reason for developing our own log parser, the architecture of which is based on the use of a semantic graph. The general scheme of the proposed approach is illustrated in Figure 2. The key idea is to build a semantic graph in the process of learning on a training data set, where individual lexical units correspond to graph vertices. When the number of graph branches becomes large enough, the graph collapses as shown in Figure 2, and frequent words are replaced with special words containing the constant part ##any and an added part consisting of a range of replaceable words. This is done, firstly, so that different special words appear in different places in the graph, and secondly, so that by looking at this word one can understand its approximate meaning and characteristic values.

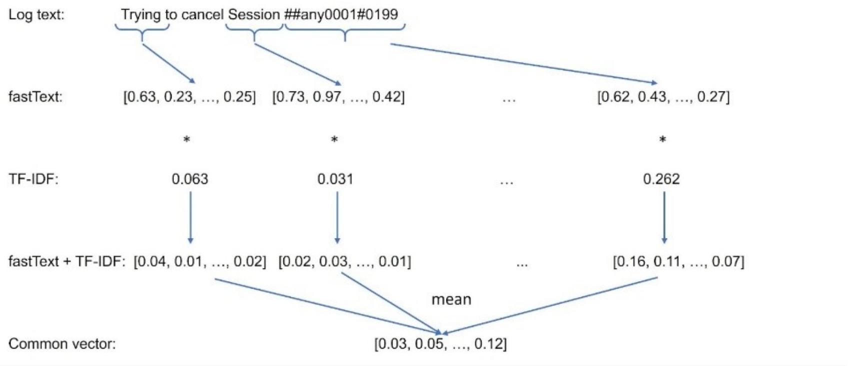


Figure 3. Log vectorization scheme

After preprocessing the data using a parser, the text information is converted into numeric information using the fastText NLP-model [3], which is trained on a training dataset. A special feature of this model is that previously unheard of words will still be assigned vectors using the so-called subword model. This advantage distinguishes this model from models such as Word2Vec [19] or Glove [20].

Since tens of thousands of individual events can be received in the logs per minute, they must first be grouped. In the work, this was done using a sliding window with a step equal to the window size. In this way, the logs are divided into equal time intervals with different numbers of events falling into them. The procedure for averaging vectors within one time interval is as follows: first, the vector of one event is calculated as the arithmetic mean of all words included in it, and then the arithmetic mean of all events in one interval is found. As a result, one time interval corresponds to one vector of the dimension specified during training of the fastText model. Words found in logs are not equivalent, so the ability to weight them using TF-IDF was also added. To do this, for each word within a time interval, its TF-IDF value is calculated, and then this value is multiplied by its vector. The same scheme was done for entire logs, which after the parser were combined into patterns. This approach allows us to reduce the importance of common words (log patterns) and increase it for rare ones. The general scheme of vectorization and averaging is shown in Figure 3. It is worth noting that the use of TF-IDF for weighting words and log patterns is optional and can be disabled. The results of the computational experiment show graphs both with and without weighting using TF-IDF.

Once the vectors for each time interval for both the training and test periods have been obtained, we can proceed to the problem of anomaly detection. In this setting, the time periods are known when there is confidence that the computer system operates without anomalies, and there is no other data labeling. Typically, as noted earlier, approaches with partial teacher involvement are used in such cases. Among the techniques tested in the work, the following algorithms should be noted: One Class SVM, Isolation Forest, Local Outlier Factor, Elliptic Envelope, k-nearest neighbors, and Autoencoder.

For each time period characterized by a vector, it is also possible to determine the contribution of log patterns to the total vector, i.e. decompose the vector into the sum of its subvectors. To do this, the projection of the vector of each pattern is found, and then it is normalized by the length of the total vector so that the sum of their contributions gives one. This allows us to find out which events had the greatest impact on the state vector and take the necessary measures to eliminate them. Figure 4 shows a diagram of this simple but useful interpretation.

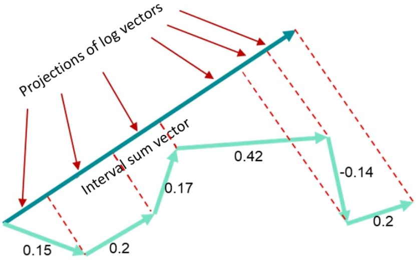


Figure 4. Scheme for determining the contributions of log patterns to the total vector

3. Results of the computational experiment

This section will discuss the experiments performed, the hyperparameters chosen, and the results.

Let's first look at the training data. As noted above, our own dataset, collected from the Vertica database logs, was used to train the model. The training data is 2 months of uninterrupted database operation in March and April, consisting of more than 800 million events. There are two test periods - late October - early November and late December - early January. Both test periods contain anomalies and database failures.

The first stage of building the model is to configure the parser on training data by training the semantic graph. The number of graph branches required for collapse, depending on the position of the token in the log (the closer to the beginning of the log, the more branches there can be), was selected analytically by selecting and evaluating various values, but in the future, automation of this process is planned.

Next, the fastText NLP-model is trained on the logs pre-processed by the parser with a given vector dimension of 100, since varying this parameter did not lead to significant changes in the results.

Along with the fastText model, TF-IDF is also trained, where time intervals act as document context. TF-IDF is trained separately for both words and log patterns. Several experiments were conducted in which TF-IDF was used to weight either only words, only log patterns, or both words and patterns.

Figure 5 shows the anomaly graphs for the test anomaly period with different set parameters: with TF-IDF calculation for log patterns only (Figure 5a), for words only (Figure 5b) and for log patterns and words simultaneously (Figure 5c). The units of anomaly measurement depend on the detection method used. Since the k-nearest neighbors method was used in Figure 5, the anomaly measure is the average distance to a given number of vectors from the training data set. Exceeding the threshold value is indicated by a scarlet indicator, and a long-term excess of this threshold is indicated by a red indicator. The threshold value is estimated based on the training data, for which the anomaly coefficient is also estimated and outliers are removed, for example, using quantiles. For example, in the work, the threshold value was equal to 0.99 quantiles of the anomaly coefficient for the training data set consisting of 830 million events. This is done to filter out single anomalies. You can also filter out outliers in the training dataset using other unsupervised learning techniques, for example, you can use Isolation Forest or Local Outlier Factor.

All graphs clearly show the occurrence of anomalies in the middle of the test period on October 28, which led to the failure on October 30, but other anomalous zones are also highlighted.

Figure 6 shows the same calculations for another test period with an anomaly that caused the failure on January 4. In these graphs, anomalies immediately before the failure are recorded only for the model with the log-only TF-IDF calculation (Figure 6a).

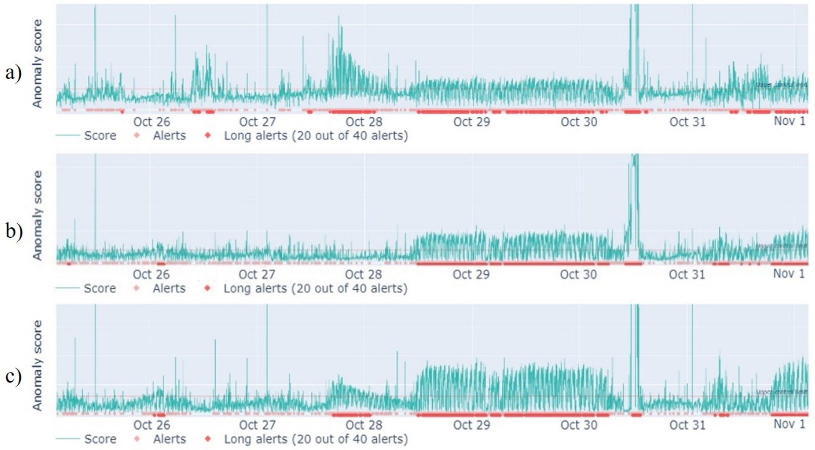


Figure 5. Anomaly graphs for the 1st test period with TF-IDF calculation: a) only for log patterns; b) only for words; c) for log patterns and words simultaneously

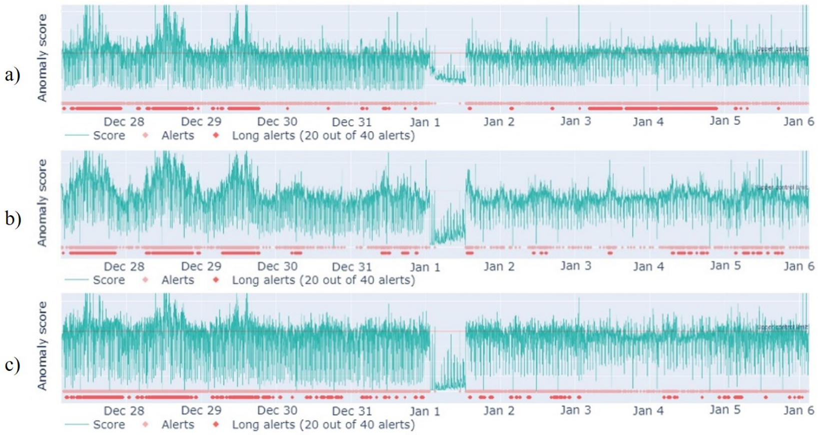


Figure 6. Anomaly graphs for the 2nd test period with TF-IDF calculation: a) only for log patterns; b) only for words; c) for log patterns and words simultaneously

The two cases considered show that calculating TF-IDF only for log patterns allows detecting anomalies before failure better than in other cases, but this is not the only advantage of this approach, which we will follow further.

It is also important to note that each log contains a tag, such as `<INFO>`, `<ERROR>`, etc., which were not involved in training the model, but the results clearly show the moment of database failure in the middle of the day on October 30 (Figure 5a), characterized by a surge in the number of fatal errors, shown in Figure 7a. But the graphs in Figure 5a, where TF-IDF was calculated only for log patterns, clearly show a correlation with the graph of the number of `<ERROR>` errors (Figure 7b), although information about them was not involved in the model. This once again proves the correctness of choosing the TF-IDF calculation method as the best model.

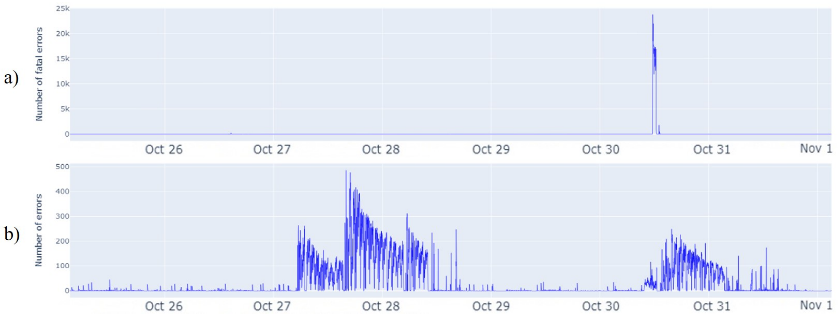


Figure 7. Fatal errors (a) and errors (b) graphs

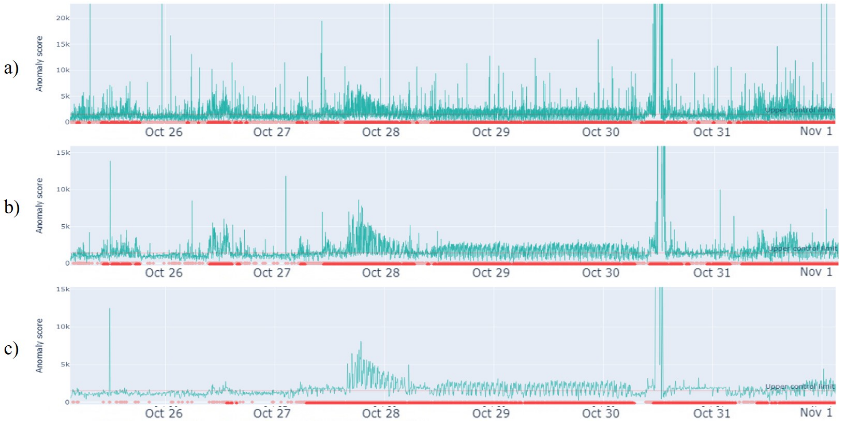


Figure 8. Anomaly graphs for different time interval sizes: 0.2 min (a), 1 min (b) and 5 min (c)

In the previous figures, the time interval size of one minute was the same everywhere. This hyperparameter was also changed to 0.2 and 5 minutes (Figure 8) for the model from Figure 5a. It is worth noting that when the interval size decreases, the computational complexity increases, therefore, the used physical and RAM memory increases, and the calculation time increases. The minimum possible interval size should also not be less than the estimated inference time of the model. A significant increase in the interval leads to excessive smoothing of the resulting vector due to averaging, so there is a high probability of missing an abnormal period.

As can be seen from the graphs, decreasing the interval size resulted in a larger number of anomaly bursts in Figure 8a compared to the one-minute interval (Figure 8b), while increasing it to 5 minutes (Figure 8c) resulted in some anomalies at the beginning of the test period disappearing due to averaging. Thus, it can be concluded that the selection of the time interval size should be left to the user, since it is necessary to find a balance between the accuracy of the model, its computational complexity and interpretability.

Figure 9 shows the graphs for the 1st test anomaly period with a comparison of the methods described in the introduction for the model with TF-IDF calculation only for log patterns (Figure 5a). It is worth noting that the k-nearest neighbors method, despite its simplicity, showed itself to be no worse than the other methods. All graphs clearly show the occurrence of anomalies in the middle of the test period, which led to failure.

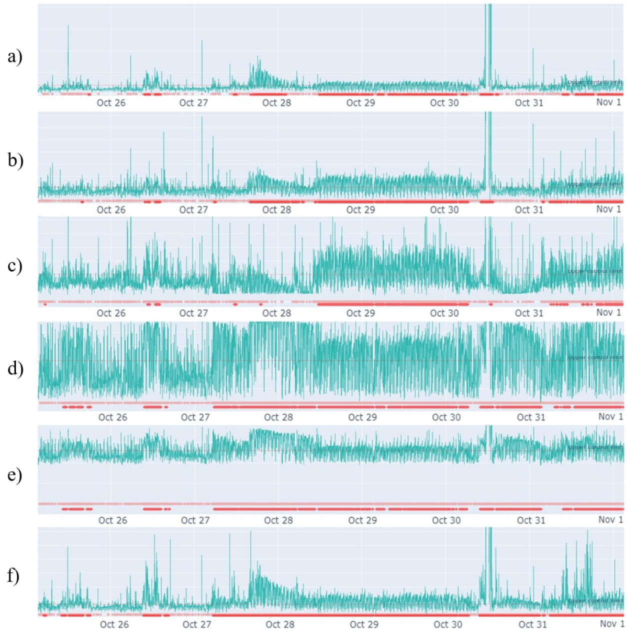


Figure 9. Anomaly plots constructed by the following methods: a) k-nearest neighbors, b) Elliptic Envelope, c) Local Outlier Factor, d) One Class SVM, d) Isolation Forest, e) Autoencoder

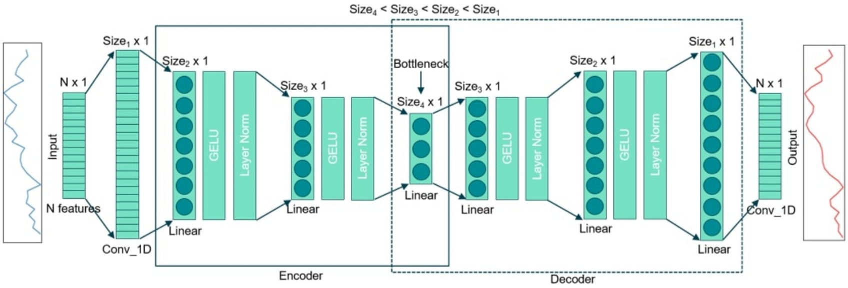


Figure 10. Schematic diagram of the Autoencoder used

The schematic of the Autoencoder used to calculate the anomaly of the test period shown in Figure 9f is shown in Figure 10. At the beginning of this neural network there are three consecutive encoder layers, transforming the original 100-dimensional vector space into a latent 10-dimensional one, called bottleneck. Then there are 3 decoder layers, returning to the original 100-dimensional vector space. The activation function was used by GeLU. The loss function during training of the autoencoder is set as Huber, combining the advantages of MSE and MAE.

As can be seen from Figure 9, the graphs turned out to be quite different, but they all highlight the period of database failure in the middle of the day on October 30, as well as the abnormal period preceding the failure. However, the correlation with the error graph (Figure 7b) is most clearly traced for the k-nearest neighbors method, so this method was chosen as the base one, and it is also easier to interpret and calculate.

Table 1

The importance of patterns in anomalies for one time interval

Log pattern	Pattern importance	Log tag	Number of patterns in interval
`Init Session ##any:0x7ea7e976a#:0x7f8eab7f0 [Basics] <WARNING> AuditedMemPool system tables deparse expression cannot reserve 100 MB of memory for planning [a0000043031afd,1]`	0.68719	`<WARNING>`	8122
`Init Session ##any:0x7ea7e976a#:0x7f8eab7f0 [Basics] <WARNING> AuditedMemPool system tables deparse expression Cannot release memory for [a0000043031afd,1], ResourceManager claims ...`	0.31267	`<WARNING>`	8122
`Init Session ##any:0x7ea7e976a#:0x7f8eab7f0 [Basics] <WARNING> MemoryPool static OPT::Plan* OPT::OptimizerInterface::makePlan (CAT::VQuery*, OPT::OptimizerConfig&) is using more mem...`	7.7252e-06	`<WARNING>`	103
`Init Session ##any:0x7ea7e976a#:0x7f8eab7f0 <NOTICE> @v_dwh_node0004: 00000/2001: NOTICE OF LICENSE NON-COMPLIANCE Continued use of this database is in violation of the...`	4.7842e-06	`<NOTICE>`	43
`Init Session ##any:0x7ea7e976a#:0x7f8eab7f0 <LOG> @v_dwh_node0004: 00000/6433: TLS session started for server`	3.8770e-06	`<LOG>`	40

In the future, it is planned to collect datasets for calculating metrics to determine the quality of the model and select optimal hyperparameters. At the moment, based on the results obtained from the two considered real cases of failure, further research has optimistic forecasts.

Let us consider one anomalous minute interval separately in more detail. As noted earlier, the vector of one interval can be decomposed into the sum of the vectors of its components with an assessment of their contribution. Table 1 shows an example of such a decomposition for the anomalous period from `2023-10-30 11:27` to `2023-10-30 11:28`. As can be seen from the table, the greatest contribution to the total vector was made by patterns with the `<WARNING>` tag, which indicate previously unseen events.

4. Conclusion

In this study, we demonstrated the results of applying semi-supervised learning techniques to solve the problem of anomaly detection in computer systems using the Vertica database as an example. The work was aimed at studying the possibility of using predictive maintenance approaches typical for technical equipment in relation to computer systems. It should be noted that even relatively simple algorithms demonstrated satisfactory efficiency in solving the problem. However, in future studies, we plan to test more complex anomaly detection techniques, as well as improved text data vectorization algorithms. In addition, we plan to expand the experimental base by collecting additional database failure incidents, conduct a comprehensive quality assessment using relevant metrics, and select the most optimal algorithm and its hyperparameters.

Author Contributions: Conceptualization, Vladislav A. Kiriachek and Soltan I. Salpagarov; methodology, Soltan I. Salpagarov; software, Vladislav A. Kiriachek; validation, Vladislav A. Kiriachek and Soltan I. Salpagarov; formal analysis, Soltan I. Salpagarov; investigation, Vladislav A. Kiriachek; resources, Vladislav A. Kiriachek; data curation, Vladislav A. Kiriachek; writing—original draft preparation, Vladislav A. Kiriachek; writing—review and editing, Vladislav A. Kiriachek and Soltan I. Salpagarov; visualization, Vladislav A. Kiriachek; supervision, Soltan I. Salpagarov; project administration, Soltan I. Salpagarov. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing is not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. He, P., Zhu, J., Zheng, Z. & Lyu, M. R. Drain: An online log parsing approach with fixed depth tree. *IEEE International Conference on Web Services (ICWS),* 33–40. doi:10.1109/ICWS.2017.13 (2017).
2. Du, M. & Li, F. Spell: Streaming parsing of system event logs. *2016 IEEE 16th International Conference on Data Mining (ICDM),* 859–864. doi:10.1109/ICDM.2016.0103 (2016).
3. Bojanowski, P., Grave, E., Joulin, A. & Mikolov, T. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics* **5,** 135–146. doi:10.1162/tacl_a_00051 (2017).
4. Zhang, X. *et al.* Robust log-based anomaly detection on unstable log data. *ESEC/FSE,* 807–817. doi:10.1145/3338906.3338931 (2019).
5. Lu, S., Wei, X., Li, Y. & Wang, L. Detecting anomaly in big data system logs using convolutional neural network. *In 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech),* 151–158. doi:10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00037 (2018).
6. Du, M., Li, F., Zheng, G. & Srikumar, V. DeepLog: Anomaly detection and diagnosis from system logs through deep learning. *In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security,* 1285–1298. doi:10.1145/3133956.3134015 (2017).
7. Meng, W. *et al.* LogAnomaly: Unsupervised Detection of Sequential and Quantitative Anomalies in Unstructured Logs. *In IJCAI* **7,** 4739–4745. doi:10.24963/ijcai.2019/658 (2019).
8. Guo, H., Yuan, S. & Wu, X. LogBERT: Log Anomaly Detection via BERT. *In 2021 international joint conference on neural networks,* 1–8. doi:10.48550/arXiv.2103.04475 (Mar. 2021).
9. Yang, L., Chen, J., Wang, Z., Wang, W., Jiang, J., Dong, X. & Zhang, W. Semi-Supervised Log-Based Anomaly Detection via Probabilistic Label Estimation. *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE),* 1448–1460. doi:10.1109/ICSE43902.2021.00130 (2021).
10. Nedelkoski, S., Bogatinovski, J., Acke, A., Cardoso, J. & Kao, O. Self-attentive classification-based anomaly detection in unstructured logs. *In 2020 IEEE international conference on data mining,* 1196–1201. doi:10.1109/ICDM50108.2020.00148 (2020).
11. Farzad, A. & Gulliver, T. A. Unsupervised log message anomaly detection. *ICT Express,* 229–237. doi:10.1016/j.icte.2020.06.003 (2020).

12. Wang, Q., Zhang, X., Wang, X. & Cao, Z. Log Sequence Anomaly Detection Method Based on Contrastive Adversarial Training and Dual Feature Extraction. *Entropy* **24,** 69. doi:10.3390/e24010069 (Dec. 2021).
13. Wan, Y., Liu, Y., Wang, D. & Wen, Y. *GLAD-PAW: Graph-Based Log Anomaly Detection by Position Aware Weighted Graph Attention Network* in (May 2021). doi:10.1007/978-3-030-75762-5_6.
14. Catillo, M., Pecchia, A. & Villano, U. AutoLog: Anomaly detection by deep autoencoding of system logs. *Expert Systems with Applications* **191.** doi:10.1016/j.eswa.2021.116263 (2022).
15. Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J. & Williamson, R. C. Estimating the support of a high-dimensional distribution. *Neural Computation* **13(7),** 1443–1471. doi:10.1162/089976601750264965 (2001).
16. Liu, F. T., Ting, K. M. & Zhou, Z. H. Isolation Forest. *2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy,* 413–422. doi:10.1109/ICDM.2008.17 (2008).
17. Breunig, M., Kröger, P., Ng, R. & Sander, J. LOF: Identifying Density-Based Local Outliers. *ACM Sigmod Record* **29,** 93–104. doi:10.1145/342009.335388 (June 2000).
18. Rousseeuw, P. J. & Van Driessen, K. A fast algorithm for the minimum covariance determinant estimator. *Technometrics* **41(3),** 212. doi:10.1080/00401706.1999.10485670 (1999).
19. Mikolov, T., Chen, K., Corrado, G. & Dean, J. Efficient estimation of word representations in vector space. doi:10.48550/arXiv.1301.3781 (2013).
20. Pennington, J., Socher, R. & Manning, C. D. Glove: Global vectors for word representation. *In Proceedings of the 2014 conference on empirical methods in natural language processing,* 1532–1543. doi:10.3115/v1/D14-1162 (2014).

Information about the authors

Kiriachek, Vladislav A.—PhD student of Department of Computational Mathematics and Artificial Intelligence of Peoples' Friendship University of Russia (RUDN University) (e-mail: w.a.kiryachok@mail.ru, phone: +7(968)8521346, ORCID: 0009-0002-9692-0225, Scopus Author ID: 57220041155)

Salpagarov, Soltan I.—Candidate of Physical and Mathematical Sciences, associate Professor of Department of Computational Mathematics and Artificial Intelligence of Peoples' Friendship University of Russia (RUDN University) (e-mail: salpagarov-si@rudn.ru, phone: +7(903)2716575, ORCID: 0000-0002-5321-9650, Scopus Author ID: 57201380251)

УДК 004.62:004.822

PACS 07.05.Mh, 07.05.Kf

DOI: 10.22363/2658-4670-2025-33-2-172-183 EDN: MGCVKV

Предиктивная диагностика логов компьютерных систем с помощью методов обработки естественного языка

В. А. Кирячёк, С. И. Салпагаров

Российский университет дружбы народов, ул. Миклухо-Маклая, д. 6, Москва, 117198, Российская Федерация

Аннотация. Данное исследование направлено на разработку и валидацию метода предиктивной диагностики и детекции аномалий в логах компьютерных систем, используя в качестве примера базу данных Vertica. Предложенный подход основан на обучении с частичным привлечением учителя в сочетании с методами обработки естественного языка. Для предварительной обработки данных разработан специализированный парсер, использующий семантический граф. Векторизация осуществлялась с применением NLP-библиотеки fastText и взвешивания TF-IDF. Эмпирическая валидация проводилась на реальных лог-файлах Vertica крупной IT-компании, содержащих как периоды нормального функционирования, так и аномалии, приведшие к сбоям. Проведена сравнительная оценка эффективности различных алгоритмов обнаружения аномалий, включая метод k-ближайших соседей, автоэнкодеры, One Class SVM, Isolation Forest, Local Outlier Factor и Elliptic Envelope. Результаты визуализированы посредством графиков аномальности, отражающих временные интервалы с превышением порогового уровня. Полученные результаты демонстрируют высокую эффективность предложенного подхода в идентификации предшествующих сбоям аномалий и определяют перспективные направления дальнейших исследований.

Ключевые слова: машинное обучение, методы обработки естественного языка, анализ логов, детекция аномалий, предиктивная диагностика

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 184–198
http://journals.rudn.ru/miph

Research article

UDC 533.6.011, 662.612, 517.95
PACS 47.70.Nd, 82.20.–w, 02.60.Lj,
DOI: 10.22363/2658-4670-2025-33-2-184-198

EDN: BPOFHS

Interval models of nonequilibrium physicochemical processes

Alexander Yu. Morozov[1, 2], Dmitry L. Reviznikov[1, 2], Vladimir Yu. Gidaspov[2]

[1] Federal Research Center Computer Science and Control of the Russian Academy of Sciences, 44-2 Vavilova St, Moscow, 119333, Russian Federation

[2] Moscow Aviation Institute (National Research University), 4 Volokolamskoe Highway, Moscow, 125993, Russian Federation

(received: November 15, 2024; revised: December 20, 2024; accepted: January 10, 2025)

Abstract. The paper discusses the application of the adaptive interpolation algorithm to problems of chemical kinetics and gas dynamics with interval uncertainties in reaction rate constants. The values of the functions describing the reaction rate may differ considerably if they have been obtained by different researchers. The difference may reach tens or hundreds of times. Interval uncertainties are proposed to account for these differences in models. Such problems with interval parameters are solved using the previously developed adaptive interpolation algorithm. On the example of modelling the combustion of a hydrogen-oxygen mixture, the effect of uncertainties on the reaction process is demonstrated. One-dimensional nonequilibrium flow in a rocket engine nozzle with different nozzle shapes, including a nozzle with two constrictions, in which a standing detonation wave can arise, is simulated. A numerical study of the effect of uncertainties on the structure of the detonation wave, as well as on steadyystate flow parameters, such as the ignition delay time and the concentration of harmful substances at the nozzle exit, is performed.

Key words and phrases: chemical kinetics, gas dynamics, interval parameters, interval velocity constants, nozzle, rocket engine, standing detonation wave, adaptive interpolation algorithm

For citation: Morozov, A. Y., Reviznikov, D. L., Gidaspov, V. Y. Interval models of nonequilibrium physicochemical processes. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 184–198. doi: 10.22363/2658-4670-2025-33-2-184-198. edn: BPOFHS (2025).

1. Introduction

In order to simulate gas-phase chemical transformations it is necessary to know the kinetic mechanism and the rates of the reactions taking place. As a rule, the dependences that describe the rates are obtained experimentally, often giving only approximate values [1]. The values of the functions approximating the rate of the same reaction, but obtained by different researchers, may differ by tens or hundreds of times. To account for these differences, we propose to use the interval apparatus [2–5]. In this case interval parameters are introduced into the model and simulation results are interval estimates for the values of interest.

The previously developed adaptive interpolation algorithm [6–9] is used to solve such problems with interval parameters. The algorithm belongs to the methods that determine an explicit dependence of the solution of the problem on the values of interval parameters. Two subgroups can be distinguished in this group: methods using symbolic expressions [10–12] and methods representing the solution as a polynomial with respect to interval parameters [13, 14]. The adaptive interpolation algorithm belongs to the latter subgroup.

This algorithm has a theoretical justification. It consists in constructing a polynomial for each moment of time which interpolates the dependence of the problem solution on the values of the parameters in a given area of uncertainty. The interpolation polynomial is constructed on the basis of a set of nodes that form a grid. At each step of the algorithm, values in the nodes of the grid are updated, and then adaptation is made depending on the interpolation error. New nodes are added in places with a large error, and nodes are removed in places with a small error. The classical version of the algorithm uses interpolation on complete meshes, which limits its application to systems with a small number of interval parameters. However, two approaches, sparse meshes [15–17] and tensor trains [18, 19], have been applied in [20–22], which extend the application of the algorithm to dynamic systems with a large number of interval parameters.

The paper deals with the problems of chemical kinetics and gas dynamics. The simulation of combustion of a mixture of hydrogen and oxygen in the presence of interval uncertainties in the reaction rate constants has been carried out. A one-dimensional mathematical model describing chemical nonequilibrium flows in a nozzle of a given shape with uncertainties in the reaction rate constants is presented. Results of a numerical study of the effect of uncertainties on the structure of the detonation wave, as well as on steady-state flow parameters, such as the ignition delay time and the concentration of harmful substances at the nozzle exit, are presented.

2. Model of chemical kinetics

Here is a description of the basic relations. A multicomponent system of a variable composition of N substances, in which N_r reactions takes place, has the form [23]:

$$\sum_{i=1}^{N} \vec{\upsilon}_i^{(r)} M_i \underset{\overleftarrow{W}^{(r)}}{\overset{\vec{W}^{(r)}}{\rightleftarrows}} \sum_{i=1}^{N} \overleftarrow{\upsilon}_i^{(r)} M_i,\ \overset{\leftrightarrow}{q}^{(r)} = \sum_{i=1}^{N} \overset{\leftrightarrow}{\upsilon}_i^{(r)},\ r = \overline{1,\ N_r},$$

where M_i — symbols for molecules or atoms of chemical components, $\overset{\leftrightarrow}{q}^{(r)}$ — molecularity of elementary reactions, r — ordinal number of reaction, $\overset{\leftrightarrow}{\upsilon}_i^{(r)}$ — stoichiometric coefficients, $\overleftrightarrow{W}^{(r)}$ — the forward and reverse r-reaction rates.

The rate $\overleftrightarrow{W}^{(r)}$ is defined as the product of the reaction rate constant $\overleftrightarrow{K}^{(r)}(T)$ and the volume concentrations of the components:

$$\overleftrightarrow{W}^{(r)} = \overleftrightarrow{K}^{(r)}(T) \prod_i (\rho\gamma_i)^{\overleftrightarrow{\upsilon}_i^{(r)}},$$

where γ_i — molar-mass concentration of the i-th component, ρ — density.

The temperature dependence of the direct reaction rate constant is approximated by the generalised Arrhenius formula:

$$\vec{K}(T) = AT^n \exp\left(-\frac{E}{T}\right),$$

where A, n, E — some constant values for each specific reaction. It is in these quantities that the uncertainty may be contained.

The rate constants of reverse reactions are calculated using the equilibrium constant:

$$\overleftarrow{K}^{(r)}(T) = \overrightarrow{K}^{(r)}(T)\exp\left[\sum_{i=1}^{N}\left(\overleftarrow{\upsilon}_i^{(r)} - \overrightarrow{\upsilon}_i^{(r)}\right)\left(\frac{G_i^0(T)}{RT} + \ln\frac{RT}{P_0}\right)\right],$$

where $G_i^0(T)$ — the standard molar Gibbs potential of the i-component, which is given by using polynomials from the handbook [24], R — universal gas constant, $P_0 = 101325$ Pa — standard pressure.

The rate of formation of the i-component is as follows:

$$W_i = \sum_{r=1}^{N_r}\left(\overrightarrow{\upsilon}_i^{(r)} - \overleftarrow{\upsilon}_i^{(r)}\right)\left(\overleftarrow{W}^{(r)} - \overrightarrow{W}^{(r)}\right), i = \overline{1,N}.$$

All thermodynamic quantities for a mixture of ideal gases are expressed in terms of the standard Gibbs molar potential. Here are some basic relations:

- Specific Gibbs potential: $G\left(T,\, P,\, \overline{\gamma}\right) = \sum\limits_{i=1}^{N}\gamma_i\left[G_i^0(T) + RT\ln\left(P\gamma_i/P_0\sum\limits_{j=1}^{N}\gamma_j\right)\right]$;
- Entropy: $S\left(T,\, P,\, \overline{\gamma}\right) = \sum\limits_{i=1}^{N}\gamma_i\left[-\frac{dG_i^0(T)}{dT} - R\ln\left(P\gamma_i/P_0\sum\limits_{j=1}^{N}\gamma_j\right)\right]$;
- Enthalpy (caloric equation): $H\left(T,\, \overline{\gamma}\right) = \sum\limits_{i=1}^{N}\gamma_i\left(G_i^0(T) - T\frac{dG_i^0(T)}{dT}\right)$;
- Internal energy (caloric equation): $U\left(T,\, \overline{\gamma}\right) = H\left(T,\, \overline{\gamma}\right) - RT\sum\limits_{i=1}^{N}\gamma_i$;
- Isobaric heat capacity: $C_p\left(T,\, \overline{\gamma}\right) = \frac{\partial H(T,\, \overline{\gamma})}{\partial T}$;
- Molar heat capacity: $C_\upsilon\left(T,\, \overline{\gamma}\right) = C_p\left(T,\, \overline{\gamma}\right) - R\sum\limits_{i=1}^{N}\gamma_i$;
- Specific heat ratio: $\varkappa\left(T,\, \overline{\gamma}\right) = \frac{C_p(T,\, \overline{\gamma})}{C_\upsilon(T,\, \overline{\gamma})}$;
- Sound speed: $a\left(T,\, \overline{\gamma}\right) = \sqrt{\varkappa\left(T,\, \overline{\gamma}\right)RT\sum\limits_{i=1}^{N}\gamma_i}$;
- Equation of state for a mixture of ideal gases (thermal equation):

$$P = \rho RT\sum_{i=1}^{N}\gamma_i;$$

where $\overline{\gamma} = (\gamma_1, \gamma_2, ..., \gamma_N)$.

The kinetic mechanisms from [25] (table 1) and [26] (table 2) have been considered to demonstrate the uncertainties associated with the rate constants of chemical reactions. The equilibrium constant was used to compare the reactions. Figure 1 shows the temperature dependences of the rate constants for different mechanisms in accordance with [20]. The strongest mismatch of the curves is observed for reactions 1, 3 and 8 from mechanism 2 which was taken into account in the corresponding interval coefficients (table 3).

Thus, the value in the Arrhenius formula becomes interval, and as a consequence the temperature dependence of the rate constants also becomes interval.

Table 1

The first combustion mechanism of the mixture $H_2 - O_2$

№	Reaction	A, m, mol, s	n	E , K
1.	$O_2 + H \to OH + O$	2.0×10^8	0.0	8455
2.	$H_2 + O \to OH + H$	5.06×10^{-2}	2.67	3163
3.	$H_2 + OH \to H_2O + H$	1.0×10^2	1.6	1659
4.	$OH + OH \to H_2O + O$	1.5×10^3	1.14	50
5.	$H + H + M \to H_2 + M$	1.8×10^6	−1.0	0
6.	$O + O + M \to O_2 + M$	2.9×10^5	−1.0	0
7.	$H + OH + M \to H_2O + M$	2.2×10^{10}	−2.0	0

Table 2

The second combustion mechanism of the mixture $H_2 - O_2$

№	Reaction	A, m, mol, s	n	E , K
1.	$H_2O + H \to OH + H_2$	8.4×10^7	0	10116
2.	$O_2 + H \to OH + O$	2.2×10^8	0	8455
3.	$H_2 + O \to OH + H$	1.8×10^4	1	4480
4.	$O_2 + M \to 2O + M$	5.4×10^{12}	−1	59400
5.	$H_2 + M \to 2H + M$	2.2×10^8	0	48300
6.	$H_2O + M \to OH + H + M$	10^{18}	−2.2	59000
7.	$HO + M \to O + H + M$	8.5×10^{12}	−1	50830
8.	$H_2O + O \to 2OH$	5.8×10^7	0	9059

Table 3

The interval part of the mixture combustion mechanism $H_2 - O_2$

№	Reaction	A, m, mol, s	n	E , K
1.	$H_2O + H \to OH + H_2$	$\left[8.4 \times 10^7,\ 4.2 \times 10^8\right]$	0	10116
3.	$H_2 + O \to OH + H$	$\left[1.8 \times 10^4,\ 9 \times 10^4\right]$	1	4480
8.	$H_2O + O \to 2OH$	$\left[5.8 \times 10^7,\ 2.9 \times 10^8\right]$	0	9059

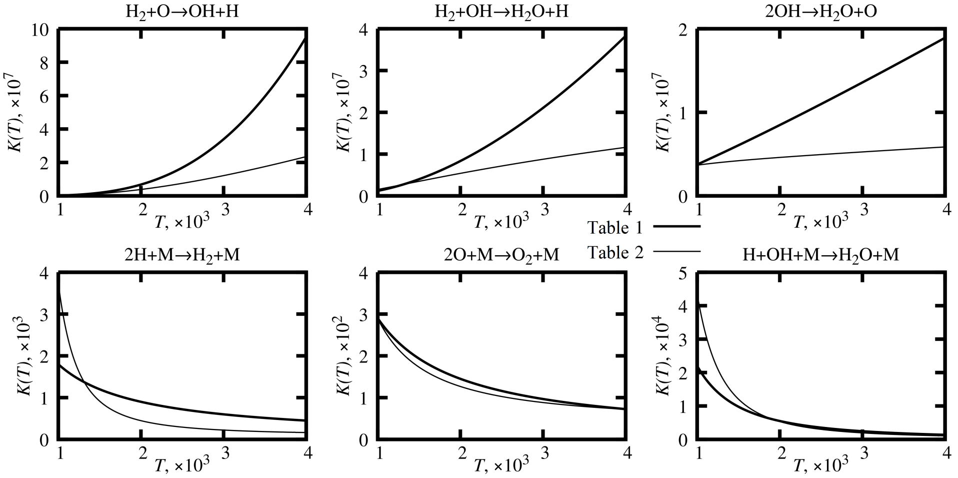


Figure 1. Comparison of different mixture combustion mechanisms $H_2 - O_2$

Let us simulate the combustion of a mixture of hydrogen and oxygen in the presence of uncertainties in the reaction rate constants according to [20]. Consider a stoichiometric mixture of hydrogen and oxygen at initial temperature $T = 1200$ K, constant density $\rho = 0.122$ kg / m^3 and constant internal energy $U = 1.48$ MJ / kg. A chemical kinetics model is given by a system of six components (H_2O, OH, H_2, O_2, H, O) in which eight reactions occur (table 2 and table 3). Here, the ODE system is written as follows:

$$\frac{d\gamma_i}{dt} = \frac{1}{\rho}\sum_{r=1}^{N_r}\left(\vec{\upsilon}_i^{(r)} - \overleftarrow{\upsilon}_i^{(r)}\right)\left(\overleftarrow{W}^{(r)} - \vec{W}^{(r)}\right),\ i = \overline{1,\ N}$$

— and is complemented by the equation of conservation of internal energy of the system:

$$U(T,\ \overline{\gamma}) = H(T,\ \overline{\gamma}) - RT\sum_{j=1}^{N}\gamma_i.$$

For each calculation of the right-hand side of the ODE system, the internal energy equation is solved relatively T using Newton's method.

Initial conditions: $\gamma_{H_2} = 55.50868$ mol / kg, $\gamma_{O_2} = 27.75434$ mol / kg, $\gamma_{H_2O} = \gamma_{OH} = \gamma_H = \gamma_O = 0$ mol / kg.

To integrate the resulting rigid ODE system, the implicit Rosenbrock method with a frozen Jacobi matrix was used [27]. Figure 2 shows the upper and lower estimates of the concentrations of all components in the mixture. The uncertainty in the reaction rate constants leads to an uncertainty in the ignition delay time between 22 and 29 μs.

Note that in this model the equilibrium state is reached regardless of the values of the reaction rate constants. This is confirmed by the fact that after a certain point in time, the upper estimates of the concentrations coincide with the lower estimates. In addition, note that the results obtained are in agreement with those obtained earlier in [20], in which the differences in the above kinetic mechanisms were accounted for in a different way.

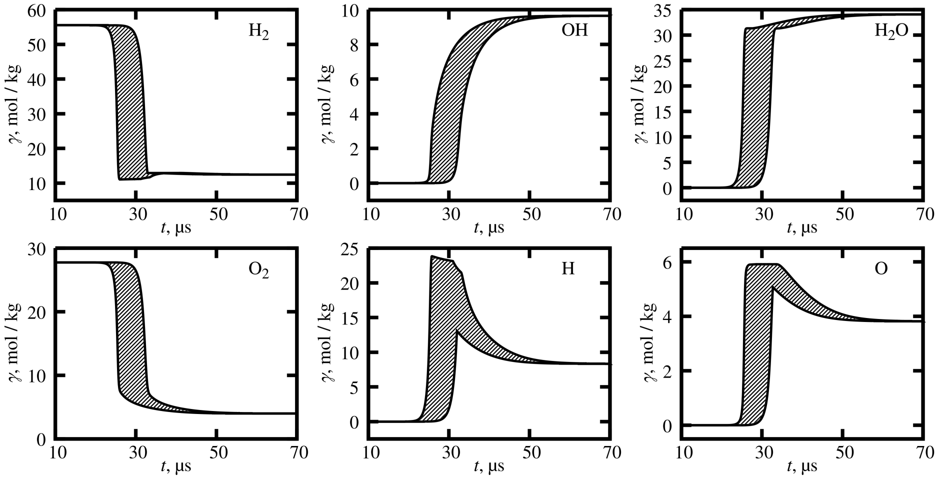


Figure 2. Dependence of mole-mass concentrations on time

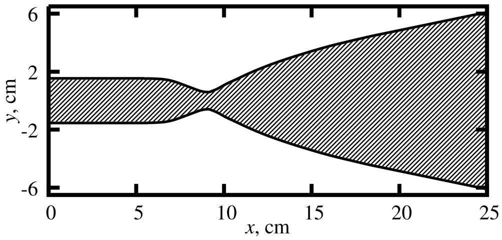


Figure 3. Nozzle profile

3. Chemical non-equilibrium flow in a nozzle

When simulating the flow in a liquid rocket engine (LRE) nozzle, ambiguity in the kinetic constants leads to uncertainties in all macro-parameters, such as thrust, Mach number, etc. Also the parameters of freezing values of concentrations of toxic combustion products become ambiguous, which is important from the point of view of ecology. For such problems, it is a natural need to determine interval estimates of solutions from known interval values of initial data. In practice, the simulation of the flow in an LRE nozzle is reduced to solving rigid ODE systems, which can be reintegrated by an adaptive interpolation algorithm.

A one-dimensional flow in the nozzle of a liquid rocket engine (figure 3) running on asymmetric dimethylhydrazine $(CH_3)_2N_2H_2$ and nitrogen tetraxide N_2O_4 is considered. The pressure in the combustion chamber is $P = 100$ atm, the oxidant excess ratio is $\alpha = 1$, the enthalpy is $H = 42.57$ kJ / kg. The concentrations in the combustion chamber were calculated from the condition of chemical equilibrium. The chemical processes were modelled by kinetic mechanism[28] involving 15 reactions and where 12 components are participating (table 4). Interval uncertainties were introduced in the rates of reactions 1 and 15.

Table 4

Kinetic mechanism for the system C-O-H-N

№	Reaction	A, m, mol, s	n	E , kJ / mol
1.	$CO + O + M \leftrightarrow CO_2 + M$	$[3.5 \times 10^2, 3.5 \times 10^3]$	0	1.06
2.	$OH + H + M \leftrightarrow H_2O + M$	1.2×10^8	−1	0
3.	$O + N + M \leftrightarrow NO + M$	3.3×10^3	0	0
4.	$H + H + M \leftrightarrow H_2 + M$	1.4×10^8	−1.5	0
5.	$O + O + M \leftrightarrow O_2 + M$	5.5×10^5	−0.87	0
6.	$N + N + M \leftrightarrow N_2 + M$	2.7×10^4	−0.5	0
7.	$H + O + M \leftrightarrow OH + M$	3.3×10^6	−0.5	0
8.	$H_2 + OH \leftrightarrow H_2O + H$	1.1×10^8	0	4.33
9.	$H_2 + O \leftrightarrow OH + H$	1.3×10^7	0	4.96
10.	$O_2 + H \leftrightarrow OH + O$	2.2×10^8	0	8.3
11.	$O_2 + N_2 \leftrightarrow NO + NO$	5.2×10^7	0	53.85
12.	$NO + N \leftrightarrow N_2 + O$	3×10^7	0	0.1
13.	$NO + O \leftrightarrow O_2 + N$	1.1×10^7	0	20.97
14.	$OH + OH \leftrightarrow H_2O + O$	1.0×10^7	0	0.6
15.	$CO + OH \leftrightarrow CO_2 + H$	$[2.5 \times 10^6, 2.5 \times 10^7]$	0	2.57

Gas flow without viscosity, thermal conductivity and diffusion is considered. Such gas flows in channels with gentle walls in continuous flow regions are given by equations which have the following divergent form:

$$\begin{cases} \frac{\partial}{\partial t}\rho F + \frac{\partial}{\partial x}\rho u F = 0, \\ \frac{\partial}{\partial t}\rho u F + \frac{\partial}{\partial x}\left(\rho u^2 + P\right) F = P\frac{\partial F}{\partial x}, \\ \frac{\partial}{\partial t}\rho\left(e + \frac{u^2}{2}\right) F + \frac{\partial}{\partial x}\rho u\left(e + \frac{P}{\rho} + \frac{u^2}{2}\right) F = 0, \\ \frac{\partial}{\partial t}\rho\gamma_i F + \frac{\partial}{\partial x}\rho u\gamma_i F = F W_i, \quad i = \overline{1, N}. \end{cases} \quad \left| \begin{array}{l} W_i = \sum\limits_{r=1}^{N_r}\left(\vec{v}_i^{(r)} - \overleftarrow{v}_i^{(r)}\right)\left(\overleftarrow{W}^{(r)} - \overrightarrow{W}^{(r)}\right), \\ \overleftrightarrow{W}^{(r)} = \overleftrightarrow{K}^{(r)}(T)\prod\limits_i (\rho\gamma_i)^{\overleftrightarrow{v}_i^{(r)}}, \\ \vec{K}(T) = AT^n \exp\left(-\frac{E}{T}\right). \end{array} \right.$$

Here first three equations are equations of conservation of mass (continuity), momentum and energy respectively; last equations are equations describing change of chemical composition; $F = F(x)$ - dependence of channel area on longitudinal coordinate; N- number of components in mixture. The system of equations is closed by thermal and caloric equations of state: $\rho = \rho\left(T, P, \overline{\gamma}\right)$, $e = e\left(T, P, \overline{\gamma}\right)$.

The finite volume method with TVD monotonization was used for modeling the flow, and the Harten-Lax-van Leer scheme was used for calculations of flows through cell boundaries. Since the resulting ODE system is rigid, it was integrated in two steps: at each step of the Runge-Kutta second-order method, which integrated the gas-dynamic equations, several steps were performed by the

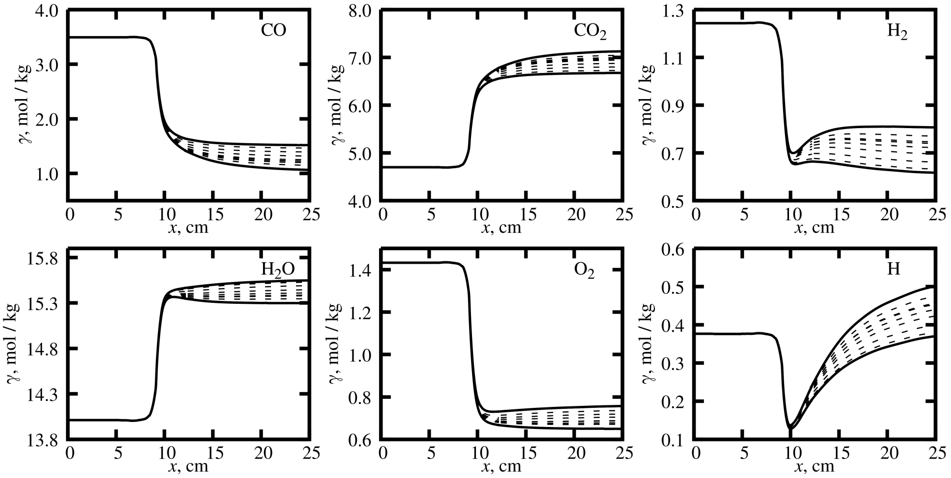


Figure 4. Mole-mass concentration distribution of the components in the nozzle

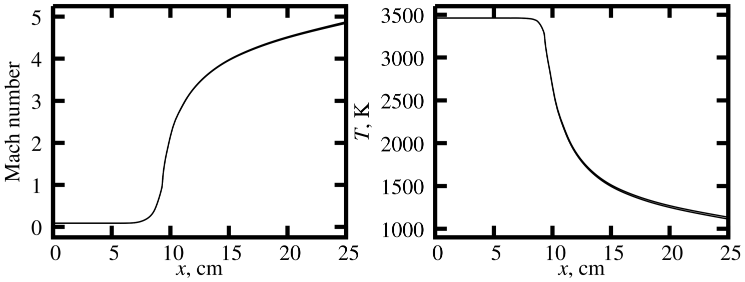


Figure 5. Macroparameter distribution in the nozzle

implicit Rosenbrock method with a frozen Jacobi matrix to integrate the chemical kinetics equations in each cell.

Figure 4 shows the concentration distributions of some mixture components in the nozzle after the flow has been established. Uncertainties in the reaction rate constants affect the freezing of the mixture components, which in turn affects the environmental performance of the engine. The dashed lines in this figure show a number of Monte Carlo runs. These are all contained in the resulting estimates. Figure 5 shows the interval estimates of the macro parameters.

Unlike concentrations, uncertainties in rate constants have much less effect on Mach number, temperature, pressure, etc.

4. Standing detonation wave

According to classical theory, a detonation wave (DW) propagating through an explosive mixture is a combination of a shock wave (SW) and an adjacent thin zone in which exothermic chemical reactions take place, ending in chemical equilibrium, with the zone of chemical transformations propagating at the SW.

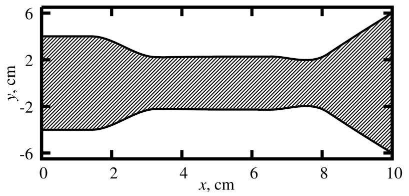


Figure 6. Nozzle profile with two constrictions

In this section, we consider one-dimensional supersonic flow in a nozzle, which is characterized by the occurrence of a standing detonation wave. Of interest is the case when a standing DW is realized up to the critical section (CS) and then the flow accelerates again to supersonic velocity at the transition through the CS. According to the classical theory [29], standing DW in the narrowing part of the channel is unstable, so for its long-term stable existence the nozzle shape with two constrictions is used. Mathematical model is a system of equations written in the previous section.

First consider a nozzle shape in which the first contraction is weakly pronounced (figure 6). The range in which the radius at the expansion section is located is chosen from the solution of the corresponding steady-state equilibrium problem. In this range a standing DW is guaranteed, assuming that all chemical transformations occur infinitely fast. In the previous problem it was obtained that the uncertainties in the reaction rate constants affect the macro parameters insignificantly, and in order to enhance their effect, the section of the nozzle where the DW should be established is made very shallow.

A mixture of H_2, O_2, N_2, Ar is fed to the nozzle inlet with velocity $u = 2750$ m / s, pressure $P = 1$ atm and temperature $T = 400$ K in the ratio $42 : 21 : 78 : 1$ [30]. Four interval uncertainties are introduced into the kinetic mechanism (table 5), which slow down the corresponding reactions by a hundred thousand times.

The flow establishment was carried out in several stages. In the first stage, only supersonic flow of unreacted gas was obtained. In the second stage, a SW was artificially induced. At the moment when it was at the required section of the nozzle, the non-interval kinetic mechanism was engaged. In the third stage, after a standing DW had been established, the interval kinetic mechanism was used.

Figure 7 shows the distributions of component concentrations in the nozzle. The presence of uncertainties in the kinetic mechanism strongly affected the width of the interval concentration estimates, in contrast to the width of the temperature estimates (figure 8). The concentration spikes in the last three plots are not anomalous and are explained by the fact that basically all chemical transformations take place in the DW region.

The following example takes a closer look at the structure of the standing DW and how it is affected by small uncertainties in the reaction rate constants. As before, a nozzle with two constrictions is used here (figure 9) and the same mixture of H_2, O_2, N_2, Ar is fed to the inlet of the nozzle.

At the point where the DW is roughly to be established, a strong compaction of the computational grid is performed so that the DW can be considered in detail. Given the constructed spatial grid, the resulting ODE system contains over 30,000 equations. The nozzle inlet velocity $u = 2500$ m / s, pressure $P = 1.38$ atm and temperature $T = 391$ K. We replace the four interval coefficients in table 5 with two according to table 6. Figure 10 shows the distributions of the interval estimates of the concentrations of all substances in the nozzle in the vicinity of the standing DW for several substances.

Figure 11 shows the distribution of Mach number and temperature in the nozzle. Practically invisible, both temperature and Mach number are interval.

Table 5

Kinetic mechanism for the system $H_2 - O_2$

№	Reaction	A, m, mol, s	n	E , K
1.	$H_2O + H \leftrightarrow OH + H_2$	$[8.4 \times 10^2, 8.4 \times 10^7]$	0	10116
2.	$O_2 + H \leftrightarrow OH + O$	$[2.2 \times 10^3, 2.2 \times 10^8]$	0	8455
3.	$H_2 + O \leftrightarrow OH + H$	1.8×10^4	1	4480
4.	$O_2 + M \leftrightarrow 2O + M$	5.4×10^{12}	−1	59400
5.	$H_2 + M \leftrightarrow 2H + M$	2.2×10^8	0	48300
6.	$H_2O + M \leftrightarrow OH + H + M$	$[1 \times 10^{13}, 1 \times 10^{18}]$	−2.2	59000
7.	$HO + M \leftrightarrow O + H + M$	8.5×10^{12}	−1	50830
8.	$H_2O + M \leftrightarrow 2OH$	$[5.8 \times 10^2, 5.8 \times 10^7]$	0	9059
9.	$H + O_2 + M \leftrightarrow HO_2 + M$	3.5×10^4	−0.41	−565
10.	$H_2 + O_2 \leftrightarrow H + HO_2$	7.39×10^{-1}	0.6	26926
11.	$H_2O + O \leftrightarrow H + HO_2$	4.76×10^5	2.43	28743
12.	$H_2O + O_2 \leftrightarrow OH + HO_2$	1.5×10^9	0.372	36600
13.	$2OH \leftrightarrow H + HO_2$	1.2×10^7	0.5	20200
14.	$OH + O_2 \leftrightarrow O + HO_2$	1.3×10^7	0	28200
15.	$H + H_2O_2 \leftrightarrow H_2 + HO_2$	1.6×10^6	0	1900
16.	$H + H_2O_2 \leftrightarrow H_2O + OH$	5×10^8	0	5000
17.	$2HO_2 \leftrightarrow H_2O_2 + O_2$	1.8×10^7	0	500
18.	$HO_2 + H_2O \leftrightarrow H_2O_2 + OH$	1.8×10^7	0	15100
19.	$OH + HO_2 \leftrightarrow H_2O_2 + O$	5.2×10^4	0.5	10600

Table 6

The modified reaction coefficients from table 5

№	Reaction	A, m, mol, s	n	E , K
1.	$H_2O + H \leftrightarrow OH + H_2$	8.4×10^7	0	10116
2.	$O_2 + H \leftrightarrow OH + O$	2.2×10^8	0	8455
6.	$H_2O + M \leftrightarrow OH + H + M$	1×10^{18}	−2.2	59000
8.	$H_2O + M \leftrightarrow 2OH$	5.8×10^7	0	9059
16.	$H + H_2O_2 \leftrightarrow H_2O + OH$	$[2.5 \times 10^8,\ 5 \times 10^8]$	0	5000
19.	$OH + HO_2 \leftrightarrow H_2O_2 + O$	$[5.2 \times 10^4,\ 2.6 \times 10^5]$	0.5	10600

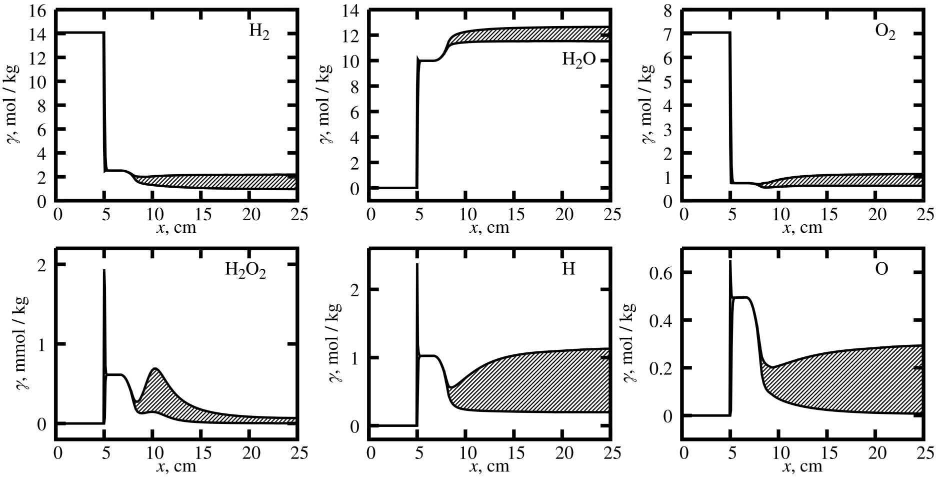


Figure 7. Mole-mass concentration distribution of the components in the nozzle

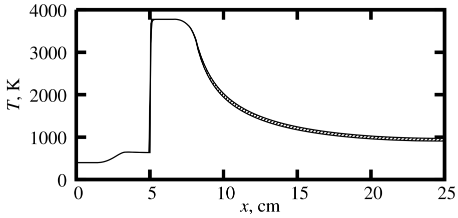


Figure 8. Temperature distribution in the nozzle

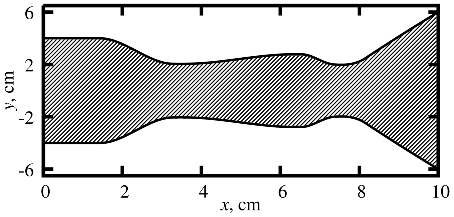


Figure 9. Nozzle profile with two pronounced constrictions

When the x-axis is zoomed in 2500 times (inset in the graphs), the structure of the DW becomes visible. The temperature plot clearly shows the SW followed by a zone of exothermic chemical reactions. Here there is a slight shift of the DV front and a slight increase in the ignition delay time.

5. Conclusion

A simulation of combustion of a hydrogen-oxygen mixture in the presence of uncertainties in the reaction rate constants has been carried out. A mathematical model of nonequilibrium flows has been developed, taking into account uncertainties in the values of reaction rate constants. Numerical studies of the effect of uncertainties on the structure of the detonation wave, as well as on the parameters of the steady-state flow, such as the ignition delay time and the concentration of harmful substances at the nozzle exit, have been performed. It is obtained that the uncertainties mainly affect the chemical composition at the nozzle exit and, to a lesser extent, the temperature, Mach number, and detonation wave. All obtained results do not contradict the already known solutions and coincide with the solutions obtained by the Monte Carlo method. The solved problems further confirm the efficiency and universality of the adaptive interpolation algorithm developed earlier

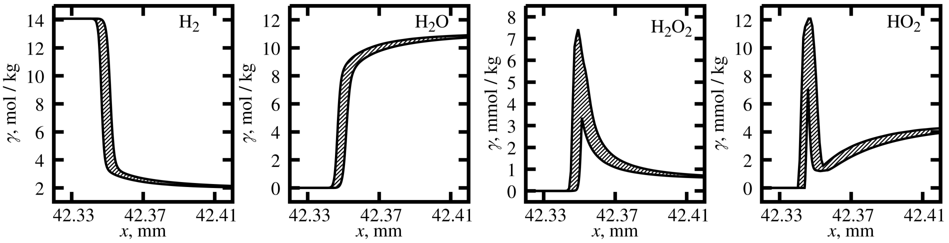


Figure 10. Distribution of component concentrations in the standing DW region

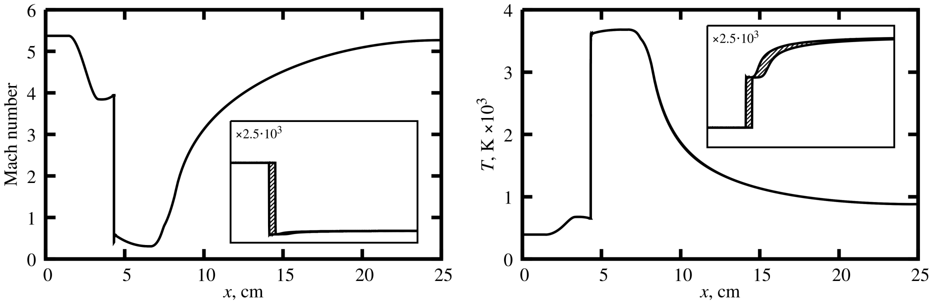


Figure 11. Mach number and temperature distribution in the standing DV region

Author Contributions: Methodology, software, investigation, A.Yu.M.; conceptualization, methodology, supervision, D.L.R.; methodology, validation, V.Yu.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Vaitiev, V. A. & Mustafina, S. A. Searching for uncertainty regions of kinetic parameters in the mathematical models of chemical kinetics based on interval arithmetic. Russian. *Bulletin of the South Ural State University. Series Mathematical Modelling, Programming & Computer Software* **7,** 99–110. doi:10.14529/mmp140209 (2014).
2. Moore, R. E. *Interval analysis* 145 pp. (Prentice-Hall, New Jersey, Englewood Cliffs, 1966).
3. Moore, R. E., Kearfott, R. B. & Cloud, M. J. *Introduction to Interval Analysis* 223 pp. doi:10.1137/1.9780898717716 (Society for Industrial and Applied Mathematics, 2009).
4. Bazhenov, A. N., Zhilin, S. I., Kumkov, S. I. & Shary, S. P. *Processing and analysis of interval data* Russian. 356 pp. (Institute of Computer Research, Izhevsk, 2024).
5. Dobronec, B. S. *Interval mathematics* Russian. 287 pp. (SibFU, Krasnoyarsk, 2007).
6. Morozov, A. Y. & Reviznikov, D. L. Adaptive Interpolation Algorithm Based on a kd-Tree for Numerical Integration of Systems of Ordinary Differential Equations with Interval Initial Conditions. *Differential Equations* **54,** 945–956. doi:10.1134/S0012266118070121 (2018).

7. Morozov, A. Y. & Reviznikov, D. L. Adaptive sparse grids with nonlinear basis in interval problems for dynamical systems. *Computation* **11.** doi:10.3390/computation11080149 (2023).
8. Morozov, A. Y., Zhuravlev, A. A. & Reviznikov, D. L. Analysis and Optimization of an Adaptive Interpolation Algorithm for the Numerical Solution of a System of Ordinary Differential Equations with Interval Parameters. *Differential Equations* **56,** 935–949. doi:10.1134/s0012266120070125 (2020).
9. Morozov, A. Y., Reviznikov, D. L. & Gidaspov, V. Y. Adaptive Interpolation Algorithm Based on a KD-Tree for the Problems of Chemical Kinetics with Interval Parameters. *Mathematical Models and Computer Simulations* **11,** 622–633. doi:10.1134/S2070048219040100 (2019).
10. Makino, K. & Berz, M. *Verified Computations Using Taylor Models and Their Applications* in *Numerical Software Verification* **10381** (Springer International Publishing, Heidelberg, Germany, July 22–23, 2017), 3–13. doi:10.1007/978-3-319-63501-9_1.
11. Neher, M., Jackson, K. & Nedialkov, N. On Taylor model based integration of ODEs. *SIAM Journal on Numerical Analysis* **45,** 236–262. doi:10.1137/050638448 (2007).
12. Rogalev, A. N. Guaranteed Methods of Ordinary Differential Equations Solution on the Basis of Transformation of Analytical Formulas. Russian. *Computational technologies* **8,** 102–116 (2003).
13. Fu, C., Ren, X., Yang, Y., Lu, K. & Qin, W. Steady-state response analysis of cracked rotors with uncertain-but-bounded para-meters using a polynomial surrogate method. *Communications in Nonlinear Science and Numerical Simulation* **68,** 240–256. doi:10.1016/j.cnsns.2018.08.004 (2018).
14. Fu, C., Xu, Y., Yang, Y., Lu, K., Gu, F. & Ball, A. Response analysis of an accelerating unbalanced rotating system with both random and interval variables. *Journal of Sound and Vibration* **466.** doi:10.1016/j.jsv.2019.115047 (2020).
15. Smoliak, S. A. Quadrature and Interpolation Formulae on Tensor Products of Certain Classes of Functions. Russian. *Dokl. Akad. Nauk. Sssr* **148,** 1042–1045 (1963).
16. Bungatrz, H.-J. & Griebel, M. Sparse grids. *Acta Numerica* **13,** 147–269 (2004).
17. Bungatrz, H. J. *Finite Elements of Higher Order on Sparse Grids* 127 pp. (Shaker Verlag, Germany, Duren/Maastricht, 1998).
18. Oseledets, I. V. Tensor-train decomposition. *SIAM Journal on Scientific Computing* **33,** 2295–2317. doi:10.1137/090752286 (2011).
19. Oseledets, I. & Tyrtyshnikov, E. TT-cross approximation for multidimensional arrays. *Linear Algebra and its Applications* **432,** 70–88. doi:10.1016/j.laa.2009.07.024 (2010).
20. Gidaspov, V. Y., Morozov, A. Y. & Reviznikov, D. L. Adaptive Interpolation Algorithm Using TT-Decomposition for Modeling Dynamical Systems with Interval Parameters. *Computational Mathematics and Mathematical Physics* **61,** 1387–1400. doi:10.1134/S0965542521090098 (2021).
21. Morozov, A. Y., Zhuravlev, A. A. & Reviznikov, D. L. Sparse Grid Adaptive Interpolation in Problems of Modeling Dynamic Systems with Interval Parameters. *Mathematics* **9.** doi:10.3390/math9040298 (2021).
22. Morozov, A. & Reviznikov, D. Adaptive Interpolation Algorithm on Sparse Meshes for Numerical Integration of Systems of Ordinary Differential Equations with Interval Uncertainties. *Differential Equations* **57,** 947–958. doi:10.1134/S0012266121070107 (2021).
23. Gidaspov, V. Y. & Severina, N. S. *Elementary Models and Computational Algorithms in Physical Fluid Dynamics. Thermodynamica and Chemical Kinetics* Russian. 84 pp. (Faktorial, Moscow, 2014).
24. Glushko, V. P., Gurvich, L. V., Veits, I. V. & et al. *Thermodynamic Properties of Some Substances* Russian (Nauka, Moscow, 1978).
25. Warnatz, J., Maas, U. & Dibble, R. *Combustion. Physical and Chemical Fundamentals, Modelling and Simulation, Experiments, Pollutant Formation* 378 pp. doi:10.1007/978-3-540-45363-5 (Springer, Berlin, Heidelberg, 2006).

26. Starik, A. M., Titova, N. S., Sharipov, A. S. & Kozlov, V. E. Syngas oxidation mechanism. *Combustion, Explosion, and Shock Waves* **46,** 491–506. doi:10.1007/s10573-010-0065-x (2010).
27. Novikov, E. A. & Golushko, M. I. (m, 3) third-order method for stiff nonautonomous systems of ODEs. Russian. *Computational technologies* **3,** 48–54 (1998).
28. Pirumov, U. G. & Roslyakov, G. S. *Gas dynamics of nozzles* Russian. 368 pp. (Nauka, Moscow, 1990).
29. Cherny, G. G. *Gas dynamics* Russian. 424 pp. (Nauka, Moscow, 1988).
30. Zhuravskaya, T. A. & Levin, V. A. Stabilization of detonation combustion of a high-velocity combustible gas mixture flow in a plane channel. *Fluid Dynamics* **50,** 283–293. doi:10.1134/S001546281502012X (2015).

Information about the authors

Morozov, Alexander Yu.—Doctor of Physical and Mathematical Sciences, Senior Researcher, Department of Mathematical Modeling of Heterogeneous Systems, Federal Research Center Computer Science and Control of the Russian Academy of Sciences; Associate Professor of the Department of Computational Mathematics and Programming, Moscow Aviation Institute (National Research University) (e-mail: morozov@infway.ru, ORCID: 0000-0003-0364-8665, ResearcherID: ABC-7836-2021, Scopus Author ID: 57203389215)

Reviznikov, Dmitry L.—Doctor of Physical and Mathematical Sciences, Professor, Leading Researcher, Department of Mathematical Modeling of Heterogeneous Systems, Federal Research Center Computer Science and Control of the Russian Academy of Sciences; Professor of the Department of Computational Mathematics and Programming, Moscow Aviation Institute (National Research University) (e-mail: reviznikov@mai.ru, ORCID: 0000-0003-0998-7975, ResearcherID: T-4571-2018, Scopus Author ID: 6602701797)

Gidaspov, Vladimir Yu.—Doctor of Physical and Mathematical Sciences, Associate Professor, Professor of the Department of Computational Mathematics and Programming, Moscow Aviation Institute (National Research University) (e-mail: gidaspov@mai.ru, ORCID: 0000-0002-5119-4488, ResearcherID: B-4572-2019, Scopus Author ID: 6506396733)

УДК 533.6.011, 662.612, 517.95
PACS 47.70.Nd, 82.20.–w, 02.60.Lj,
DOI: 10.22363/2658-4670-2025-33-2-184-198 EDN: BPOFHS

Интервальные модели неравновесных физико-химических процессов

А. Ю. Морозов[1, 2], Д. Л. Ревизников[1, 2], В. Ю. Гидаспов[2]

[1] Федеральный исследовательский центр “Информатика и управление” Российской академии наук, ул. Вавилова, д. 44, кор. 2, Москва, 119333, Российская Федерация

[2] Московский авиационный институт (национальный исследовательский университет), Волоколамское шоссе, д. 4, Москва, 125993, Российская Федерация

Аннотация. В данной работе рассматривается применение алгоритма адаптивной инетрполяции к задачам химической кинетики и газовой динамики с интервальными неопределенностями констант скоростей реакций. Значения функций, описывающих скорость реакции, могут значительно различаться, если они были получены разными исследователями. Разница может достигать десятков или сотен раз. Для учета данных различий в моделях предлагается использовать интервальные неопределенности. Решение таких задач с интервальными параметрами выполняется с помощью ранее разработанного алгоритма адаптивной интерполяции. На примере моделирования горения смеси водорода и кислорода демонтируется влияние неопределенностей на процесс протекания реакций. Моделируется одномерное неравновесное течение в сопле ракетного двигателя с разной формой сопла, включая сопло с двумя сужениями, в котором может возникать стоячая детонационная волна. Выполняется численное исследование влияния неопределенностей на структуру детонационной волны, а так же на параметры установившегося течения, такие как время задержки воспламенения и концентрация вредных веществ на выходе из сопла.

Ключевые слова: химическая кинетика, газовая динамика, интервальные параметры, интервальные константы скоростей, сопло, ракетный двигатель, стоячая детонационная волна, алгоритм адаптивной интерполяции

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 199–213
http://journals.rudn.ru/miph

Research article

UDC 519.854.2

PACS 03.67.Ac, 02.10.Ox

DOI: 10.22363/2658-4670-2025-33-2-199-213

EDN: BOMUBB

Evaluating quantum-classical heuristics for traveling salesman problem

Mariia A. Makarova[1, 2], Sergey V. Fedorov[1], Anna V. Titova[1],
Alexander A. Khomich[1], Alexander S. Rumyantsev[1, 2]

[1] Petrozavodsk State University, 33 Lenina Pr, Petrozavodsk, 185910, Russian Federation

[2] Institute of Applied Mathematical Research of the KarRC RAS, 11 Pushkinskaya St, Petrozavodsk, 185910, Russian Federation

(received: February 18, 2025; revised: March 1, 2025; accepted: March 10, 2025)

Abstract. In this paper, we develop and evaluate a hybrid quantum-classical heuristic approach to solving the Traveling Salesman Problem. This approach uses exhaustive enumeration of the starting paths and optimizes the remainder of the route using quantum computing. For quantum co-processing, we use either the Variational Quantum Eigensolver or the Quantum Annealing. Results of evaluation of the approach on several datasets including TSPLIB and touristic data for Petrozavodsk and Karelia Republic, both in simulation and in hardware, are presented. Issues of practical applicability are also discussed.

Key words and phrases: hybrid quantum-classical heuristics, traveling salesman problem, variational quantum eigensolver, quantum annealing

For citation: Makarova, M. A., Fedorov, S. V., Titova, A. V., Khomich, A. A., Rumyantsev, A. S. Evaluating quantum-classical heuristics for traveling salesman problem. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 199–213. doi: 10.22363/2658-4670-2025-33-2-199-213. edn: BOMUBB (2025).

1. Introduction

Traveling Salesman Problem (TSP) is a challenging combinatorial problem which is NP-hard [1], that is, the exact optimal solution in general cannot be obtained in polynomial time (w.r.t. the size of input data). The problem implies a search for the shortest possible (cyclic) route that visits (exactly once) a set of cities and returns to the starting city. Since the number of feasible routes increases exponentially with the number of cities [2], it is computationally hard to solve for large TSP instances using traditional algorithms such as brute force and the Held–Karp algorithm [3] (which, however, works well on a small scale).

Despite its difficulty, the TSP has many practical applications in various fields such as logistics [4], transportation, and network design. Finding efficient solutions to TSP can help industries optimize delivery routes, reduce costs, and improve overall efficiency. Due to this practical importance, various heuristic and approximation algorithms were introduced to find suboptimal solutions in a reasonable time, such as the nearest-neighbor search [5] or k-opt [6] for the Lin–Kernighan algorithm [7]. The search for effective algorithms continues, and new promising efforts are made in the direction of novel computing hardware such as quantum computing (QC).

This research is focused on solving the TSP with QC techniques. QC is a promising research field for solving large-scale problems due to the so-called *quantum supremacy*. The latter is a theoretical concept that exploits quantum bit (qubit) properties such as quantum *superposition* (co-existence of the qubit in multiple states at once) and *entanglement* (dependence of several qubits) to outperform traditional algorithms. Some examples of effective quantum algorithms include the celebrated Shor algorithm [8] and Deutsch–Josa algorithm [9], the latter being a key example of the so-called *quantum parallelism* [10]. Performance of such effective schemes is, however, limited due to the need of a quantum-classical input-output interface and imperfections of the present QC hardware such as limited coherence time due to environmental noise, high error rates and a relatively small number of qubits, those being problems of the so-called Noisy intermediate-scale quantum (NISQ) computers [11].

One of the natural classes of problems that can be addressed by QC is optimization, and in many cases the QC techniques can either offer speedup to the classical algorithms, or deliver efficient heuristics. A detailed description of state-of-the-art quantum optimization (QO) methods, both from the perspective of complexity and practical applicability, is given in the review paper [2]. Two important QO techniques are the Variational Quantum Algorithms (VQA) [12, 13] and Quantum Annealing (QA) implemented on hardware [14]. While the former allows one to use gate and circuit-based QC hardware (known to be *universal* i.e. capable of running any algorithm), the latter suits only for QA machines such as the D-Wave, useful only for a specific problem set. However, both options can be effectively used to tackle the class of Quadratic Unconstrained Binary Optimization (QUBO) problems where the cost function is quadratic w.r.t. the binary unknowns.

Within QO, TSP is a prominent example of a problem in which complexity depends on specific requirements and restrictions (such as the type of state space and the goal), and heuristic QO can provide meaningful results even if quantum speedups seem unattainable [2]. Thus, many researchers focus on solving TSP using various heuristic QO algorithms, including QA and VQA. We briefly summarize these two key approaches to TSP below, and refer the interested reader to a survey [15] that outlines general logistics and supply chain management problems treated by QC.

In [16], a comparative study of Quantum Approximate Optimization Algorithms (QAOA), a specific variant of VQA, is performed. Various aspects of the approach, such as sensitivity to the TSP graph, numerical accuracy and effect of noise, are studied. Performance of QAOA implementation for TSP with 5 cities is studied in [17]. In [18], comparative study of solution encoding is performed for the TSP problem to be solved using VQA methods. Results are presented for problem sizes of up to 6 cities. In [19], Variational Quantum Eigensolver (VQE), which is another representative of VQA class, and QAOA schemes are compared to the approach based on the so-called Grover adaptive search, and numerical results are presented for TSP with up to 7 cities. The effect of the various configurations of the parametrized quantum circuits within the VQE approach to TSP is demonstrated in [20], and numerical results are presented for 4-cities TSP.

In [21], QA was used to tackle TSP with four smallest problems from TSPLIB including Burma14 by the D-Wave using Kerberos and LeapHybridSampler solvers, and reasonable solution quality was reported with relatively less optimal solutions found for bigger size problems. Note, however, that the solvers used are hybrid (quantum-classical) and remain trade secrets of the D-Wave [21]. In contrast, D-Wave's 5,000-qubit Advantage 1.1 quantum annealer is useful only for solving TSP with 8 or fewer cities, as reported in [22]. A groundbreaking approach in solving TSP by minimal QC resources was introduced in [23], where a single-qubit scheme with a careful encoding of TSP using Bloch sphere was used to obtain solutions with problem size up to 9. We also mention that solving TSP by QA can be a useful part of a hybrid approach to more general real-time routing problem [24].

In summary, both in the VQA and QA approaches to TSP the size of the problem that is solvable by QC is rather limited, which is in part explained by the imperfections of the NISQ hardware [25]. In most cases reported in the literature reviewed above, the size of the TSP solved by the QC hardware is at most 9 cities. At the same time, hybrid approaches such as the one in [21] are useful for studying larger problems. This motivates us to develop a hybrid quantum-classical approach to TSP that can be used both within the QA and VQA frameworks. To do so, we use the QUBO representation of the TSP. The key ingredients of our approach are the classical enumeration of the beginning of the route followed by QC counterpart used as the co-processor to optimize the route ending. This novel combination is the key contribution of the present paper.

Following [25], we use VQE as the VQA class technique, and we use D-Wave in case of the QA. We demonstrate the capabilities of our approach using TSPLIB example and apply the method to optimize tours between points of interest (POI) within Karelia Republic and Petrozavodsk city.

The rest of the paper is organized as follows. We give a necessary background on the QC, VQE and QA, describe the QUBO representation of the TSP in Section 2. The hybrid approach for solving TSP in QUBO form is proposed and evaluated (using the TSPLIB dataset Burma14 and a few others) in Section 3. The paper ends up with a discussion on possible complications with solving TSP in QUBO form in Section 4.

2. Necessary Background

In this section we briefly review the QC and QO fundamentals, TSP and its QUBO representation. This information is useful for the development of the hybrid approach. More detailed information on QC can be found in the celebrated monograph [26], and QO is reviewed in [2]. We also refer the reader to the book [27] for a more concise view of TSP.

2.1. QC Fundamentals

A qubit is a basic information unit in QC. It is a quantum system with two basis states $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ living in a 2-dimensional Hilbert space. Any qubit state $|\psi\rangle$ can be in *superposition* of the basis states, meaning that

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \quad \alpha, \beta \in \mathbb{C}, \;\; |\alpha|^2 + |\beta|^2 = 1.$$

The coefficients α and β hold a defined physical significance, as observing a qubit would result in the state $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$. However, a qubit can only be measured once as the measurement process causes the qubit to collapse to one of the binary states, preserving that state in subsequent measurements. (Using Dirac notation, $|\psi\rangle$ is a column, and $\langle\psi|$ is a row vector of appropriate size.)

To compose multiple independent qubits, the tensor (Kronecker) product $\otimes$ is typically used. For example, the product $\psi \otimes \varphi$ of two single-qubit states $|\psi\rangle$ and $|\varphi\rangle$ results in a qubit pair state represented as a vector in $\mathbb{C}^4$, and in general n qubits can be represented as a vector in $\mathbb{C}^{2^n}$. Such a representation can be written as

$$|\varphi\rangle = \sum_{i=0}^{2^n} \alpha_i \, |i\rangle \, ,$$

where $\sum_{i \leq 2^n} |\alpha_i|^2 = 1$ and $\alpha_i \in \mathbb{C}$. The basis vectors $|i\rangle$ can also be written in the form of binary representation of i from the set $\{0,1\}^n$, say, $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ instead of $|0\rangle, \ldots, |3\rangle$ for $n = 2$. However, algebraically, the size of $|\phi\rangle$ and, correspondingly, $|i\rangle$ is 2^n, where we can write

$$|i\rangle = (x_0, \ldots, x_{2^n}), \quad x_j = \delta_{i,j}, \tag{1}$$

with $\delta_{i,j} = 1$ for $i = j$ and 0 otherwise (i.e. Kronecker delta function).

Note that states of qubits that can be represented as the tensor product are independent. On the contrast, *entanglement* allows qubits to interact regardless of their distance, creating a strong connection between them. When qubits are entangled, they cannot be treated separately and their states become dependent on each other. The Bell state of a 2-qubit system

$$|\varphi\rangle = \frac{|00\rangle + |11\rangle}{2}$$

is an example where the state cannot be broken down into two separate single-qubit states.

2.2. QC Models

There are two widely used models of QC. The universal model (that can represent any computation) is constructed by using the *quantum gates* which act on individual qubits and their groups. In QC hardware, these gates usually operate with single qubits and pairs. Such operations can be interpreted as matrix multiplications from the left by the unitary matrix U. A matrix U is unitary if its conjugate (Hermitian) transpose equals its inverse, $U^* = U^{-1}$. Note that such a transformation preserves matrix norms, and thus keeps the unit sum of probabilities for the qubit states after transpose. Using quantum gates and qubits, the quantum program is often represented as the so-called quantum circuit which describes the sequence of gates applied to various *qubit lines* (acting as variables, e.g. inputs or outputs of the program). In such a model, input is transformed into output by a sequence of gates in a discrete way.

Another universal model is the so-called adiabatic QC (AQC). On the contrast to the circuit model, in AQC the transformation of the qubit state to the desired solution of the problem is performed continuously, by slow change in the parameters of the system Hamiltonian $H(t)$ at time t (in AQC hardware it can be implemented, say, by a slowly changing magnetic field). Due to a slow change, the qubits remain in the so-called *ground state* (i.e. state with minimal energy) which corresponds to the system Hamiltonian $H(t)$. Thus, to obtain the desired solution, problem is encoded in a Hamiltonian H_P whose minimal energy state is the desired solution, and the transformation is organized by a scheme

$$H(t) = (1 - s(t))H_I + s(t)H_P, \ \ t \in [0, \tau], \tag{2}$$

where $s : [0, \tau] \to [0, 1]$ is a smooth monotone function such that $s(0) = 0$ and $s(\tau) = 1$, H_I is some initial Hamiltonian which is easy to construct, and τ is the final time of computation. It can be noted that the solution of a problem in AQC is obtained in finite time τ irregardless of the properties of H_P.

2.3. VQA and VQE

VQA is a class of algorithms that adopt hybrid quantum-classical approach to obtain quantum advantage on the NISQ devices [12]. This set is universal in a sense that arbitrary quantum circuit can be represented by an appropriate VQA problem [28]. The approach is based on defining a cost function $C(\theta)$ which encodes the problem and estimating $C(\theta)$ by QC, whereas classically optimizing θ, to find $\theta^* = \operatorname{argmin}_\theta C(\theta)$ (encoding the solution). Dependence on θ is parametrized by the so-called ansatz operator $U(\theta)$, a specific quantum operator which is then optimized iteratively.

A specific version of VQA is VQE aimed at finding the ground state energy of quantum systems [29]. This makes it particularly valuable in quantum chemistry [30], materials science [31], and chemical engineering [32]. The algorithm is based on AQC approach [33] and the variational principle. According to the latter, for any system represented by the Hamiltonian H and the quantum state $|\psi\rangle$, the expectation value $\langle\psi|H|\psi\rangle$ (corresponding to the expected energy value of the system) is equal to or greater than the ground state energy of the system [34]. It is interesting to illustrate, why $\langle\psi|H|\psi\rangle$ is the expected value of the energy that is obtained after measurement. Indeed, take $|\psi\rangle = \sum_{i=0}^{2^n} \alpha_i |i\rangle$ and use the so-called Hermitian decomposition [35] to obtain $H = \sum_{i=0}^{2^n} \lambda_i |i\rangle\langle i|$, where λ_i are eigenvalues corresponding to eigenvectors $|i\rangle$ and $\langle i|$ (these correspond to energy states of the appropriate basis states). Note that due to (1), the basis $|0\rangle, \dots, |2^n\rangle$ is orthonormal. Then, recalling that $\langle x|y\rangle$ is the complex scalar product that requires complex conjugate to be used on the first argument, observe

$$\langle\psi| H |\psi\rangle = \sum_{i=0}^{2^n} \lambda_i \langle\psi|i\rangle \langle i|\psi\rangle = \sum_{i=0}^{2^n} \lambda_i \alpha_i^* \alpha_i. \tag{3}$$

Recalling that $\alpha_i^*\alpha_i = |\alpha_i|^2$ is the probability of measuring the basis state $|i\rangle$, it is clear from (3) that the value observed corresponds to the average energy of the system.

Since in VQE the value ψ depends on the parameter vector θ, after measuring the average energy $C(\theta) = \langle\psi(\theta)| H |\psi(\theta)\rangle$, the value θ is iteratively modified so as to minimize $C(\theta)$ and, accordingly, minimize the energy of the ground state of the system [12, 29]. To achieve this, VQE uses QC to initialize quantum register ψ_0, apply anzatz to get $\psi(\theta) = U(\theta)\psi_0$, then apply H and measure $C(\theta)$, and optimize the circuit's parameter θ variationally (using classical optimizer) by repeating these steps accordingly.

2.4. QA

A particular case of AQC, which is not universal, but rather useful in QO, is QA. Historically the QA term was also used to denote heuristic optimization method [36] (which is somewhat similar to the so-called simulated annealing). To be analyzed by QA, the (optimization) problem needs to be encoded by a Hamiltonian H_P in such a way that the ground state (with minimal energy) of a system characterized by H_P corresponds to the desired solution. Then the solution is obtained using the sufficiently slow evolution of the ground state from the one encoded by initial Hamiltonian H_I following (2). The initial Hamiltonian H_I is selected in such a way that the ground state is relatively straightforward, say, when all qubits are in a superposition state $|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. Hardware implementations such as the D-Wave machine in the majority of cases require H_P to represent quadratic model, that is, to have the Ising form or QUBO form. In the former case,

$$H_P = \sum_{i=1}^{n} h_i {\sigma_z}^i + \sum_{i>j} J_{ij} {\sigma_z}^i {\sigma_z}^j, \tag{4}$$

where ${\sigma_z}^i = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ is the so-called Pauli-Z matrix acting on qubit $i = 1, \dots, n$, $J_{ij} = J_{ji}$ is the so-called coupling strength of the coupled qubits i and j and h_i is the so-called qubit bias (on-site energy) acting on qubit i. The matrix ${\sigma_z}^i$ has eigenvalues $\{-1, 1\}$ and thus, using QA, the Ising cost function is minimized [37]:

$$\mathcal{E}() = \sum_{i=1}^{n} h_i z_i + \sum_{i>j} J_{ij} z_i z_j, \tag{5}$$

where $\in \{-1, 1\}^n$. In D-Wave systems, $h_i \in [-2, 2]$ and $J_{ij} \in [-1, 1]$ (these ranges, however, can be extended by autoscaling). When running the optimization problem on a QA machine, the topology of possible coupling between qubits needs to be taken care of, which can be done automatically by the so-called minor embedding. This process, however, may dramatically upscale the problem in terms of the number of physical qubits required.

In summary, the optimization problem is encoded by the vector $(h_1, \dots, h_n)$ and the matrix $||J_{i,j}||_{i,j\in\{1,\dots,n\}}$, and the solution needs to be decoded from the vector $\vec{z} \in \{-1, 1\}^n$ that minimizes the Ising cost $\mathcal{E}(\vec{z})$ given in (5).

An equivalent representation of the problem is in QUBO format, where the solution is encoded by a binary vector. In such a case, QUBO objective function has the following form:

$$\mathcal{E}(\vec{x}) = \vec{x}^T Q \vec{x} = \sum_{i=1}^{n} Q_{ii} x_i + \sum_{i>j} Q_{ij} x_i x_j, \tag{6}$$

where $\vec{x} \in \{0, 1\}^n$. Therefore, the optimal solution in QUBO format is $\operatorname{argmin}_{\vec{x}} \mathcal{E}(\vec{x})$, where $\vec{x}$ is a binary vector encoding the solution, and Q is a symmetric matrix of real values encoding the problem.

2.5. TSP and its QUBO Representation

There are many ways to define the problem in TSP class, depending on the restrictions imposed, goals to optimize and properties of the travel costs. In simpler case, the cost of travel from city $i = 1, \dots, n$ to $j = 1, \dots, n$ is symmetric, i.e. equals the cost from j to i, and these constitute a matrix $M = ||M_{ij}||_{i,j\leq n}$, where $M_{ij} = M_{ji}$. Then it is straightforward to define the cost of a tour with the help of a binary matrix $X = ||X_{ij}||_{i,j\leq n}$, where $X_{ij} = 1$ if i-th city is traversed at j-th place. As such, the cost of travel can be given as

$$\sum_{k=1}^{n-1} \sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij} X_{ik} X_{j(k+1)} + \sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij} X_{in} X_{j1}. \tag{7}$$

Unrolling the matrix X columnwise into a column vector $\vec{x} = (x_1, \dots, x_{2^n})$ by a transformation

$$x_{n(k-1)+i} = X_{ik}, \quad i, k = 1, \dots, n, \tag{8}$$

and defining

$$Q = J \otimes M, \text{ where } J = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix},$$

it is easy to see that (7) can be transformed into the form (6), and the matrix Q has almost block-upper diagonal form.

At the same time, in order to solve the QUBO representation of TSP, it is not enough just to encode the cost function, but we also need to guarantee that X (and hence its vector representation $\vec{x}$) encodes a feasible solution. To do so, one-hot type restrictions are imposed on columns and rows of X, that is,

$$\sum_{i=1}^{n} X_{ij} = \sum_{j=1}^{n} X_{ij} = 1,$$

for all $i, j = 1, \dots, n$. To embed these restrictions into the cost function, quadratic terms are introduced into cost function with (appropriately large) Lagrange coefficients L_{1i}, L_{2i}, $i = 1, \dots, n$. That is, the cost of violating feasibility of the solution (which is zero for a feasible X) is given as

$$\sum_{j=1}^{n} L_{1j} \left(1 - \sum_{i=1}^{n} X_{ij}\right)^2 + \sum_{i=1}^{n} L_{2i} \left(1 - \sum_{j=1}^{n} X_{ij}\right)^2 .$$

Taking for simplicity $L_{1i} = L_{2i} = L$ as in `traveling_salesperson_qubo` function of D-Wave `networkx.algorithms` package, noting that the constant $\sum_{j=1}^{n} L_{1j} + \sum_{i=1}^{n} L_{2i} = 2nL$ can be omitted from the optimization function, recalling that (due to binary nature of the unknowns) $X_{ij}^2 = X_{ij}$ and using the transformation (8), after some straightforward algebra, the constraints can be given in QUBO form as

$$\vec{x}^T(-2L\vec{I} + 2L(\vec{U} - \vec{I}))\vec{x},$$

where $\vec{I}$ is the identity matrix and $\vec{U}$ is the matrix of ones. Finally, this gives the QUBO representation of TSP in the form (6) as follows,

$$\vec{x}^T \left(J \otimes M + 2L(\vec{U} - \vec{I}) - 2L\vec{I}\right) \vec{x} \to \min,$$

where, recall, M is the given matrix of distances. Note that at implementation phase, care must be taken, since some QA frameworks may require Q to be upper-triangular.

3. Proposed Approach and its Evaluation

In this section we present a hybrid quantum-classical approach to tackle the TSP problem in QUBO form, both by using QA and VQE techniques. Results of numerical experiments both on the input file from the well-known TSPLIB data source and two small-scale problems related to touristic POI in Karelia Republic are presented.

3.1. Hybrid Approach in Solving TSP in QUBO Form

A representation of the TSP as QUBO problem, as can be seen from the results presented in Section 2, is useful both for the classical (gate-based) QC and for QA. Indeed, solving QUBO problem by VQE may be done by imposing restrictions on the problem Hamiltonian (3) so as to describe interactions only up to pairs of qubits, in a similar way like the Ising form (4) does.

Based on this similarity, it seems fruitful to state the TSP in QUBO form and use either QA or VQE to obtain the (approximate) solution of the optimization problem. This, however, comes at a price of a dramatic increase of the state space and decrease in “density” of feasible solutions. We elaborate more on this in Section 4. To mitigate this effect, we propose the following two-stage hybrid scheme.

At the *first stage*, we define the size $n_q \le n$ of the subproblem that is to be solved by QC in QUBO form. Fixing the first city (say, city 1) of the tour, the initial path of $n - 1 - n_q$ cities is enumerated exhaustively. For each such path $x \in \{2, \dots, n\}^{n-1-n_q}$ (having unique cities), we define a submatrix

$$M_x = ||M_{ij}||_{i,j \in \{2,\dots,n\} \setminus x}.$$

At the *second stage*, the solution of a TSP problem with a set of remaining cities $\{2, \dots, n\} \setminus x$ with the costs defined by M_x is obtained by QC. The solution y is then concatenated with the beginning x, and the overall price of the cycle (x, y) is calculated. The result is then obtained after exhaustive enumeration over x is complete. We note that this heuristic is partially inspired by the celebrated Karp's partitioning algorithm, see [38] on the detailed probabilistic analysis of the accuracy of the latter. Note that, due to the nature of QC, the result of the second stage may be suboptimal.

To implement this solution, we used `Qiskit` and `DWave` frameworks for classical QC and QA, respectively. In both cases, at the second stage, the TSP was solved in QUBO form. In the former case, using `qiskit_optimization.applications` library, TSP in QUBO form was augmented with the so-called TwoLocal ansatz, whereas Simultaneous Perturbation Stochstic Approximation `SPSA` optimzier from `qiskit_algorithms.optimizers` library was used to optimize the ansatz. Sampling of VQE was done with the help of `SamplingVQE` function from `qiskit_algorithms` library using the standard Sampler from `qiskit.primitives`. In the latter case, the result of the second stage was obtained by `traveling_salesperson` function from `dwave_networkx.algorithms` library using `SimulatedAnnealingSampler` or `DWaveSampler` from `dwave.samplers` library.

3.2. Experimental Evaluation

In order to validate the proposed approach, a number of numerical experiments were conducted. Those included solving TSP on a few examples; in all cases, the problem had a symmetric distance matrix, and the distances were geographical. The first dataset is `Burma14` from the TSPLIB, which is a standard set of benchmarks and algorithms related to TSP. Two more datasets included distances between POIs for Petrozavodsk city (walking distance) and Karelia Republic (driving/linear distance).

For the experiments in simulation mode, we used Linux-based machine (AMD Ryzen 9 7900X, 16 Gb memory) and Linux-based nodes of the high-performance computing cluster of Karelian Research Center (two Intel Xeon Silver 4215R, 128 Gb memory). Both machines were running Python 3.12, the following versions of libraries were used: `dwave_networkx 0.8.15`, `dwave-samplers 1.4.0`, `qiskit 1.2.0`, `qiskit-algorithms 0.3.0`, `qiskit-optimization 0.6.1`. A small set of experiments was also performed using the trial cloud access to the D-Wave machine.

The first set of benchmarks was used to determine computational capabilities in simulation mode. During these experiments, a *problem of size 8* was solved using *Petrozavodsk* dataset. 8 POIs located within Petrozavodsk city were: ”Molecule”, ”Soldier, Woman and Child” and ”Fishers” fountains, Sister-cities gallery, Karelian State Philarmony, Peter the Great monument, Alexander Nevsky Cathedral and National Museum. The walking distances in the dataset were acquired through public maps API.

For the VQE-based solver on Petrozavodsk dataset, the size of QC subproblem for the second stage was set to $n_q = 4$ due to the fact that `SamplingVQE` function required over 1 TB memory for $n_q = 6$, occasionally $n_q = 5$ also produced memory overflow, whereas the TSP problem for 3 cities (for $n_q = 3$) is trivial. The VQE approach reached the globally optimal solution with length 8985 (steps). For QA approach, larger size of QC subproblem can be used (in simulation mode). However, due to heuristic nature of the result, in order to increase the sample set (and hence the quality) $n_q = 4$ was used for QA approach as well. The solution returned for QA after several runs had suboptimal length of 9155. It is important to note that the runtime for QA is dramatically shorter than for VQE (minutes vs. days),

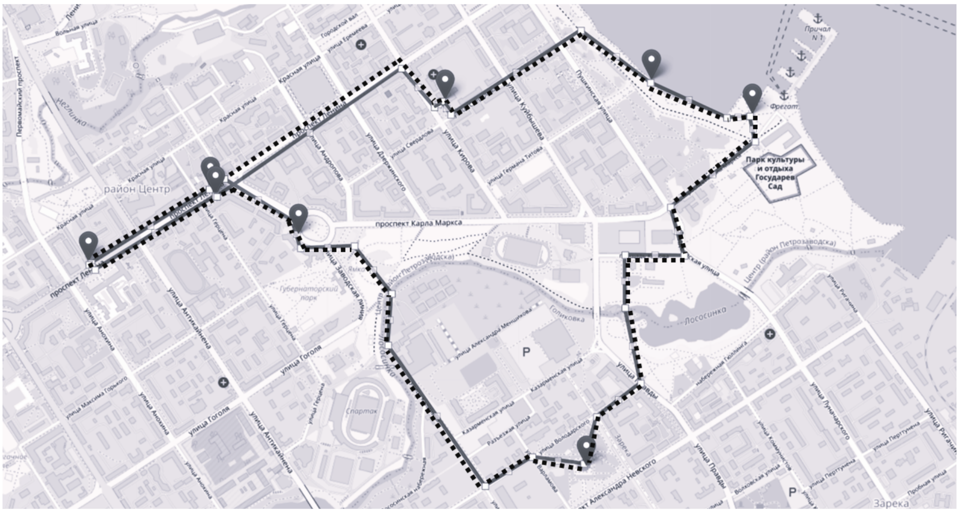


Figure 1. Solutions for the TSP on the *Petrozavodsk* dataset by the two heuristic approaches with $n_q = 4$, dotted line: VQE (tour length 8985 steps), solid line: QA (9155 steps)

which, as we can see, compromises the quality. The best solutions for both approaches obtained in simulation mode are depicted in Figure 1.

The second set of benchmarks was based on the Burma14 dataset. For the VQE approach (with $n_q = 4$), the overall computation time was around 65 hours, whereas for QA in simulation mode it took only several minutes (with $n_q = 8$) to derive an approximate solution. Both approaches performed reasonably well, with the best solution for VQE having tour length 3499 and the best tour for QA in simulation mode having length 3795, compared to the best known result of length 3323 [39]. The best solutions for VQE and QA in simulation mode are depicted in Figure 2.

Running the algorithm on the Burma14 dataset on D-Wave with quantum engines required $n_q \leq 7$ to obtain feasible solutions. Indeed, due to a limited computational time, only 3717 solutions (for the case $n_q = 7$) were derived, and out of those only 140 were feasible, whereas for $n_q = 8$ we did not find any feasible solution out of 935. QA on quantum machine performed slightly worse with tour length 4641. We elaborate more on the reasons for this suboptimality in Section 4.

Due to a reasonable quality of the solutions obtained at higher speed, we used QA approach in simulation mode with $n_q = 4$ to solve the TSP for *Karelia* dataset, where the following 10 natural and cultural POIs were listed: Martsial'nye Vody (Marcial Waters resort), Ruskeala mountain park, Kizhi island, Kiwatsch waterfall, Sortavala city, Syamozero lake, Valaam monastery, Petrozavodsk city, Old Ladoga, and Karelian Zoo. After five runs, the globally optimal solution of length 1676 (kilometers) was obtained, which is depicted in Figure 3. A larger dataset of 18 POIs in Petrozavodsk (including 8 earlier used) was processed by QA approach in simulation mode with $n_q = 12$. The runtime was around 4 hours, and the route including 47488 steps is depicted in Figure 4.

4. Conclusion and Discussion

In this paper we presented a hybrid quantum-classical approach for solving TSP in QUBO form using the VQE and QA as the quantum counterpart, and exhaustive enumeration of the starting path for the

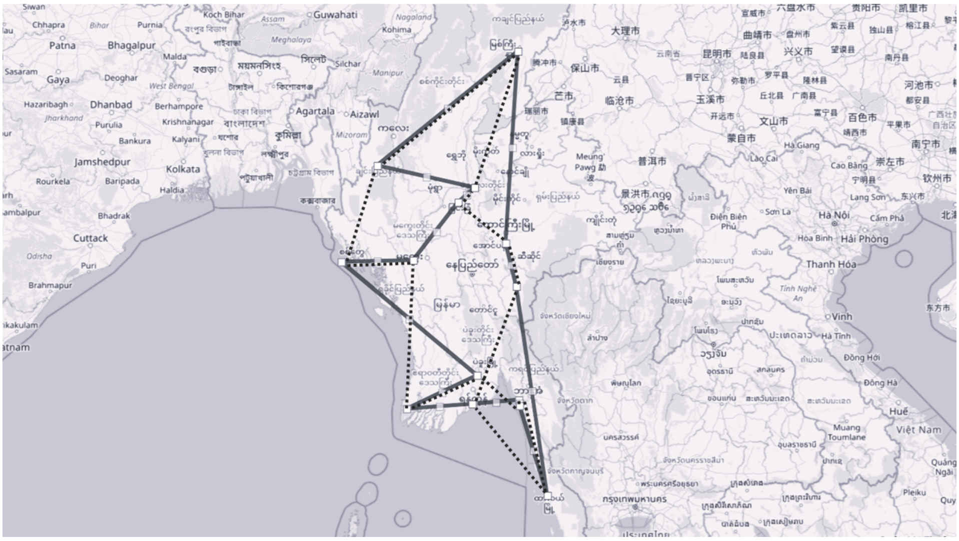


Figure 2. Solutions for the TSP on the Burma14 dataset by the two heuristic approaches, dotted line: VQE ($n_q = 4$, tour length 3499 km), solid line: QA ($n_q = 8$, 3795 km)

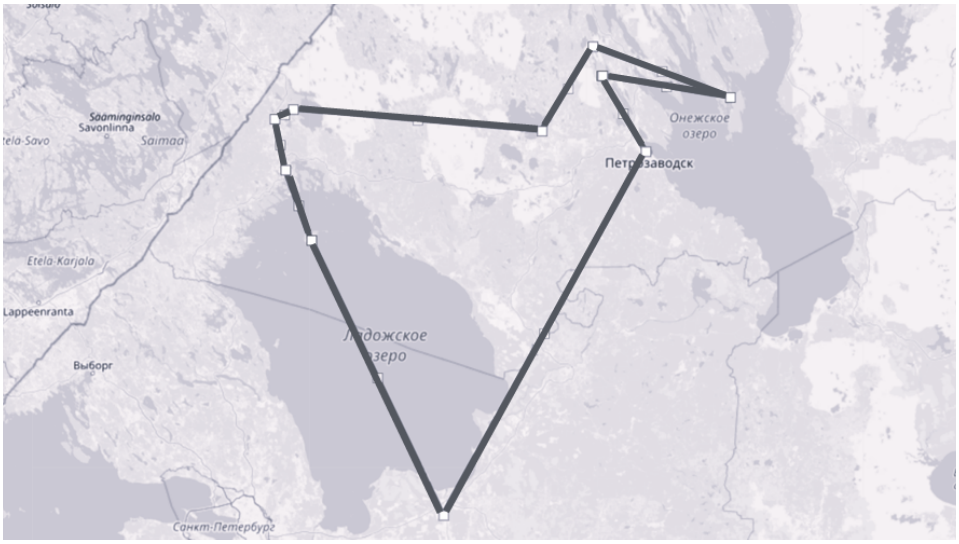


Figure 3. Solutions for the TSP on the *Karelia* dataset (10 POIs) by the QA approach with $n_q = 4$, tour length 1676 km

classical counterpart. The approach performed reasonably well for both QC techniques, although in simulation mode the quality of solutions obtained by VQE was better than those obtained with QA, at a price of dramatically higher computing time.

As for the QA run at the D-Wave machine, there were several problems that allowed us to find feasible solutions only for subproblem size (solved at quantum machine) $n_q \leq 7$. Moreover, for $n_q = 10$ using the `MinorEmbedding`, the cloud was not able to find a quantum engine to run the problem

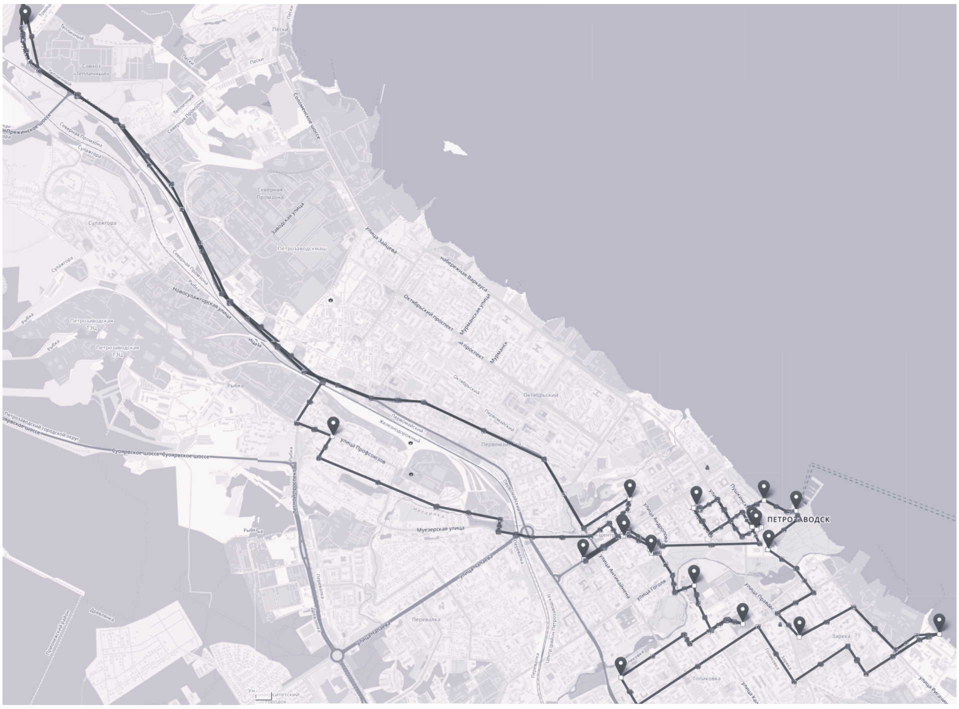


Figure 4. Solutions for the TSP on the *Petrozavodsk* dataset (18 POIs) by the QA approach with $n_q = 12$, tour length 47488 steps

with. There are several reasons for that. Firstly, as reported in [22], out of m physical qubits only around $\sqrt{m}$ logical are available, thus resulting in about 73 *logical* qubits for the D-Wave computing unit having around 5000 *physical* qubits. Secondly, due to the one-hot encoding (8), a problem of size n requires at least n^2 *logical* qubits. This essentially means that TSP of size n requires around n^4 physical qubits which explains the limitation $n_q \leq 8$ for our case, see also the discussion on the embedding capabilities in [40].

Another problem comes from QUBO encoding (8) is the density of the feasible solutions. It is shown in [18] that such density decreases dramatically for TSP QUBO with increasing size. Indeed, this can be intuitively explained by the fact that the set of n^2-component binary vectors has 2^{n^2} elements, whereas the set of feasible solutions for TSP problem contains only $n!$ (for simplicity, we do not take into account symmetries and fixed starting city). Asymptotic analysis using the celebrated Stirling's formula shows that the ratio of the feasible solutions to the state space of QUBO problem vanishes for large n,

$$\frac{n!}{2^{n^2}} \sim \frac{\sqrt{2\pi n}(n/e)^n}{e^{n^2 \log 2}} \sim e^{n \log n - n + 0.5 \log n - n^2 \log 2} \to 0, \quad n \to \infty.$$

It is rather simple to check that even for the case $n = 6$ the frequency of feasible solution is of order 10^{-8}, see also [18]. This explains the low density of feasible solutions that was evidenced when running QA experiments on hardware with $n_q = 7$, see Section 3.2.

One of the perspective approaches to overcome this "curse of dimensionality" is to find a different encoding for the TSP. In particular, permutation encoding may be promising [18] for the QO approaches that do not rely on Hamiltonian. Other approach is introduced in [23] by converting the TSP problem into a classical Brachistochrone problem and using the single-qubit scheme with rotations. Other techniques such as Grover adaptive search can also be adopted [19]. However, all those possibilities need careful exploration that seems promising for future research.

Author Contributions: Conceptualization, M.M. and A.R.; methodology, M.M and A.R.; software, S.F., A.K., M.M., A.R.; validation, S.F., A.K.and A.R.; formal analysis, M.M. and A.R.; investigation, A.T.; resources, A.R.; data curation, A.T.; writing—original draft preparation, M.M. and A.R.; writing—review and editing, M.M.; visualization, A.T.; supervision, A.R.; project administration, S.F., M.M. and A.R.; funding acquisition, S.F. All authors have read and agreed to the published version of the manuscript.

Funding: The research described in this publication was made possible in part by R&D Support Program for undergraduate and graduate students and postdoctoral researchers of PetrSU, funded by the Government of the Republic of Karelia. Experiments were conducted with the help of high-performance computing cluster of the Center for Collective Use “High-Performance Data Center” of the Karelian Research Center of RAS.

Data Availability Statement: Data can be sent by the authors on reasonable request.

Acknowledgments: Authors thank the referees for careful reading and suggestions.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Jünger, M., Reinelt, G. & Rinaldi, G. *Chapter 4 The traveling salesman problem* in *Network Models* (Elsevier, 1995). doi:10.1016/s0927-0507(05)80121-5.
2. Abbas, A. *et al.* Challenges and opportunities in quantum optimization. *Nature Reviews Physics* **6,** 718–735. doi:10.1038/s42254-024-00770-9 (Dec. 2024).
3. Held, M. & Karp, R. M. A Dynamic Programming Approach to Sequencing Problems. *Journal of the Society for Industrial and Applied Mathematics* **10,** 196–210. doi:10.1137/0110015 (Mar. 1962).
4. Filip, E. & Otakar, M. The travelling salesman problem and its application in logistic practice. **8,** 163–173 (Oct. 2011).
5. Kizilateş, G. & Nuriyeva, F. *On the Nearest Neighbor Algorithms for the Traveling Salesman Problem* in *Advances in Computational Science, Engineering and Information Technology* (Springer International Publishing, 2013). doi:10.1007/978-3-319-00951-3_11.
6. Helsgaun, K. General k-opt submoves for the Lin–Kernighan TSP heuristic. *Mathematical Programming Computation* **1,** 119–163. doi:10.1007/s12532-009-0004-6 (Oct. 2009).
7. Lin, S. & Kernighan, B. W. An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations Research* **21,** 498–516. doi:10.1287/opre.21.2.498 (Apr. 1973).
8. Shor, P. *Algorithms for quantum computation: discrete logarithms and factoring* in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (1994), 124–134. doi:10.1109/SFCS.1994.365700.
9. Deutsch, D. & Jozsa, R. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* **439.** Publisher: Royal Society, 553–558. doi:10.1098/rspa.1992.0167 (Jan. 1997).
10. Markidis, S. *What is Quantum Parallelism, Anyhow?* en. in *ISC High Performance 2024 Research Paper Proceedings (39th International Conference)* (IEEE, Hamburg, Germany, May 2024), 1–12. doi:10.23919/ISC.2024.10528926.
11. Cheng, B. *et al.* Noisy intermediate-scale quantum computers. *Frontiers of Physics* **18,** 21308. doi:10.1007/s11467-022-1249-z (Mar. 2023).
12. Cerezo, M. *et al.* Variational quantum algorithms. *Nature Reviews Physics* **3,** 625–644. doi:10.1038/s42254-021-00348-9 (Sept. 2021).

13. Grange, C., Poss, M. & Bourreau, E. An introduction to variational quantum algorithms for combinatorial optimization problems. en. *Annals of Operations Research* **343,** 847–884. doi:10.1007/s10479-024-06253-5 (Dec. 2024).
14. Mohseni, N., McMahon, P. L. & Byrnes, T. Ising machines as hardware solvers of combinatorial optimization problems. *Nature Reviews Physics* **4,** 363–379. doi:10.1038/s42254-022-00440-8 (June 2022).
15. Phillipson, F. *Quantum Computing in Logistics and Supply Chain Management an Overview* en. arXiv:2402.17520 [quant-ph]. Feb. 2024. doi:10.48550/arXiv.2402.17520.
16. Qian, W., Basili, R. A. M., Eshaghian-Wilner, M. M., Khokhar, A., Luecke, G. & Vary, J. P. Comparative Study of Variations in Quantum Approximate Optimization Algorithms for the Traveling Salesman Problem. en. *Entropy* **25,** 1238. doi:10.3390/e25081238 (Aug. 2023).
17. Ruan, Y., Marsh, S., Xue, X., Liu, Z. & Wang, J. The Quantum Approximate Algorithm for Solving Traveling Salesman Problem. en. *Computers, Materials & Continua* **63,** 1237–1247. doi:10.32604/cmc.2020.010001 (2020).
18. Schnaus, M., Palackal, L., Poggel, B., Runge, X., Ehm, H., Lorenz, J. M. & Mendl, C. B. *Efficient Encodings of the Travelling Salesperson Problem for Variational Quantum Algorithms* in *2024 IEEE International Conference on Quantum Software (QSW)* (IEEE, Shenzhen, China, July 2024), 81–87. doi:10.1109/QSW62656.2024.00022.
19. Zhu, J., Gao, Y., Wang, H., Li, T. & Wu, H. *A Realizable GAS-based Quantum Algorithm for Traveling Salesman Problem* en. arXiv:2212.02735 [quant-ph]. Dec. 2022. doi:10.48550/arXiv.2212.02735.
20. Matsuo, A., Suzuki, Y., Hamamura, I. & Yamashita, S. Enhancing VQE Convergence for Optimization Problems with Problem-specific Parameterized Quantum Circuits. en. *IEICE Transactions on Information and Systems* **E106.D.** arXiv:2006.05643 [quant-ph], 1772–1782. doi:10.1587/transinf.2023EDP7071 (Nov. 2023).
21. Warren, R. H. *Solving combinatorial problems by two D_Wave hybrid solvers: a case study of traveling salesman problems in the TSP Library* Version Number: 1. 2021. doi:10.48550/ARXIV.2106.05948.
22. Jain, S. Solving the Traveling Salesman Problem on the D-Wave Quantum Computer. en. *Frontiers in Physics* **9,** 760783. doi:10.3389/fphy.2021.760783 (Nov. 2021).
23. Goswami, K., Veereshi, G. A., Schmelcher, P. & Mukherjee, R. *Solving The Travelling Salesman Problem Using A Single Qubit* en. arXiv:2407.17207 [quant-ph]. Dec. 2024. doi:10.48550/arXiv.2407.17207.
24. Mario, S. S., Pothamsetti, P., Antony, L., Vishwanath, T., Ha, D. S., Ahmed, A., Sinno, S., Thuravakkath, S. & M, D. S. *Quantum Annealing Based Hybrid Strategies for Real Time Route Optimization* en. 2024. doi:10.2139/ssrn.4970901.
25. Khumalo, M. T., Chieza, H. A., Prag, K. & Woolway, M. An investigation of IBM quantum computing device performance on combinatorial optimisation problems. en. *Neural Computing and Applications* **37,** 611–626. doi:10.1007/s00521-022-07438-4 (Jan. 2025).
26. Nielsen, M. A. & Chuang, I. L. *Quantum computation and quantum information* 10th anniversary ed. en (Cambridge University Press, Cambridge ; New York, 2010).
27. Jünger, M., Reinelt, G. & Rinaldi, G. en. *Chapter 4 The traveling salesman problem* in *Handbooks in Operations Research and Management Science* 225–330 (Elsevier, 1995). doi:10.1016/S0927-0507(05)80121-5.
28. Biamonte, J. Universal variational quantum computation. en. *Physical Review A* **103,** L030401. doi:10.1103/PhysRevA.103.L030401 (Mar. 2021).
29. Tilly, J. *et al.* The Variational Quantum Eigensolver: A review of methods and best practices. *Physics Reports* **986,** 1–128. doi:10.1016/j.physrep.2022.08.003 (Nov. 2022).

30. Deglmann, P., Schäfer, A. & Lennartz, C. Application of quantum calculations in the chemical industry—An overview. *International Journal of Quantum Chemistry* **115,** 107–136. doi:10.1002/qua.24811 (Nov. 2014).
31. Lordi, V. & Nichol, J. M. Advances and opportunities in materials science for scalable quantum computing. *MRS Bulletin* **46,** 589–595. doi:10.1557/s43577-021-00133-0 (July 2021).
32. Cao, Y. *et al.* Quantum Chemistry in the Age of Quantum Computing. *Chemical Reviews* **119,** 10856–10915. doi:10.1021/acs.chemrev.8b00803 (Aug. 2019).
33. Vesely, M. *Finding the Optimal Currency Composition of Foreign Exchange Reserves with a Quantum Computer* 2023. arXiv: 2303.01909 [econ.GN].
34. Zhu, L., Liang, S., Yang, C. & Li, X. *Optimizing Shot Assignment in Variational Quantum Eigensolver Measurement* 2024. arXiv: 2307.06504 [quant-ph].
35. Gantmacher, F. *The Theory of Matrices* (Chelsea Publishing Company, 1980).
36. De Falco, D., Apolloni, B. & Cesa-Bianchi, N. *A numerical implementation of quantum annealing* in (July 1988).
37. Su, J. Towards Quantum Computing: Solving Satisfiability Problem by Quantum Annealing (2018).
38. Karp, R. M. Probabilistic Analysis of Partitioning Algorithms for the Traveling-Salesman Problem in the Plane. en. *Mathematics of Operations Research* **2,** 209–224 (1977).
39. Karagul, K., Aydemir, E. & Tokat, S. Using 2-Opt based evolution strategy for travelling salesman problem. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)* **6,** 103–113. doi:10.11121/ijocta.01.2016.00268 (Mar. 2016).
40. *Estimating Quantum Volume for Advantage* https://support.dwavesys.com/hc/en-us/community/posts/360051945133-Estimating-Quantum-Volume-for-Advantage.

Information about the authors

Makarova, Mariia A.—junior researcher, PhD student, SMITS Lab, Institute of Applied Mathematical Research of the Karelian Research Center of Russian Academy of Sciences; lecturer, Petrozavodsk State University (e-mail: masha.mariam.maltseva@mail.ru, ORCID: 0000-0002-7982-0209, ResearcherID: AAZ-7810-2021, Scopus Author ID: 57204436887)

Fedorov, Sergey V.—Bachelor's degree student of Petrozavodsk State University (e-mail: sergey-fedorov-04@mail.ru)

Titova, Anna V.—Bachelor's degree student of Petrozavodsk State University

Khomich, Alexander A.—Bachelor's degree student of Petrozavodsk State University

Rumyantsev, Alexander S.—Doctor of Physical and Mathematical Sciences, Leading researcher, SMITS Lab, Institute of Applied Mathematical Research of the Karelian Research Center of Russian Academy of Sciences; professor, Petrozavodsk State University (e-mail: ar0@krc.karelia.ru, ORCID: 0000-0003-2364-5939, ResearcherID: L-1354-2013, Scopus Author ID: 36968331100)

УДК 519.854.2

PACS 03.67.Ac, 02.10.Ox

DOI: 10.22363/2658-4670-2025-33-2-199-213 EDN: BOMUBB

Исследование квантово-классической эвристики для задачи коммивояжера

М. А. Макарова[1, 2], С. В. Федоров[1], А. В. Титова[1], А. А. Хомич[1], А. С. Румянцев[1, 2]

[1] Петрозаводский государственный университет, пр. Ленина, д. 33, Петрозаводск, 185910, Российская Федерация

[2] Институт прикладных математических исследований КарНЦ РАН, ул. Пушкинская, 11, Петрозаводск, 185910, Российская Федерация

Аннотация. В статье разрабатывается и оценивается гибридный квантово-классический эвристический подход к решению задачи коммивояжера. Этот подход использует исчерпывающий перебор начальных путей и оптимизирует оставшуюся часть маршрута с помощью квантовых вычислений. Для обработки на квантовой машине используется либо вариационный квантовый собственный решатель, либо квантовый отжиг. Представлены результаты оценки предложенного подхода на нескольких наборах данных, включая TSPLIB и туристические данные для Петрозаводска и Республики Карелия, как в режиме имитации, так и на соответствующей квантовой машине. Обсуждаются вопросы практической применимости.

Ключевые слова: гибридная квантово-классическая эвристика, задача коммивояжера, вариационный квантовый собственный решатель, квантовый отжиг

Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (Online), 2658-4670 (Print)

2025, 33 (2) 214–225
http://journals.rudn.ru/miph

Research article

UDC 519.7
PACS 07.05.Tp
DOI: 10.22363/2658-4670-2025-33-2-214-225

EDN: MKTGEJ

On modelling multi-agent systems based on large language models

Eugeny Yu. Shchetinin[1], Tatyana R. Velieva[2], Lyubov A. Yurgina[3], Anastasia V. Demidova[2]

[1] Financial University under the Government of the Russian Federation, 49, Leningradsky Pr, Moscow, 125993, Russian Federation

[2] RUDN University, 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

[3] Sochi Branch of RUDN University, 32 Kuibyshev St, Sochi, 354340, Russian Federation

(received: March 1, 2025; revised: March 14, 2025; accepted: March 20, 2025)

Abstract. The article studies the effectiveness of implementation of multi-agent systems based on large language models in various spheres of human activity, analyses their advantages, problems and challenges. The results of the research have shown that multi-agent systems based on large language models have significant potential and wide opportunities in modelling various environments and solving various tasks.

Key words and phrases: multi-agent systems, large language models, society modeling

For citation: Shchetinin, E. Y., Velieva, T. R., Yurgina, L. A., Demidova, A. V. On modelling multi-agent systems based on large language models. *Discrete and Continuous Models and Applied Computational Science* **33** (2), 214–225. doi: 10.22363/2658-4670-2025-33-2-214-225. edn: MKTGEJ (2025).

1. Introduction

Artificial intelligence has undergone significant changes in the last few years, rapidly moving from basic automation to sophisticated generative models. Large language models (LLMs) such as GPT-4o or Gemini, which can understand and create human-like text, have pushed the boundaries of MAShine learning, enabling unprecedented applications in content creation, customer support, and other areas. These advances have shown how AI can process and interpret vast amounts of data, but it operates within a set of constraints, often lacking the flexibility to adapt in real time [1, 2].

A new paradigm, known as agent-based artificial intelligence (AI), is emerging that has the potential to revolutionize the role of AI: from simply interpreting and responding to prompts to autonomously managing complex tasks. This leap promises a future in which AI not only improves productivity and decision-making, but also autonomously navigates dynamic environments, making it a transformative technology for all fields of application.

One of the most promising and rapidly growing areas of AI today is the field of multi-agent systems (MAS), as they have the potential to rethink the way MAShines interact and solve complex problems. These systems decline from traditional approaches to AI, allowing autonomous agents to collaborate, communicate, and coordinate their actions to achieve common goals in dynamic environments. From healthcare to finance, logistics to entertainment, MAS are poised to revolutionize a variety of industries through the use of collective intelligence and adaptive behavior.

Traditional large language models, while powerful, have limitations. First, LLMs are trained on huge amounts of data that represent a snapshot in time, limiting their knowledge to the date the training data was collected. Consequently, they are unaware of events and information that have emerged since the last update to the training data, limiting their usefulness in scenarios that require up-to-date data. Additionally, adapting monolithic models like LLMs is a complex and resource-intensive task, as they require significant amounts of data and computational power to fine-tune their behavior.

Another key limitation is that standard LLMs operate without context-sensitive access to user information or databases. For example, if someone queries a model to determine available vacation days, a typical LLM cannot retrieve this information because it does not have access to personalized, secure data. While traditional LLMs can help with tasks such as document summarization or text composition, they are insufficient when users need specific answers based on real-time or customized data sources.

In addition, the structure of LLMs makes it difficult to add components in order to dynamically verify or validate results. Although they can generate answers based on patterns in their data, they cannot autonomously access external databases or tools to validate or augment their answers. This limits their accuracy and flexibility when solving complex problems that may require multi-step solutions using real-world data. Agent-based AI overcomes these limitations by integrating modular components (real-time search tools, logic verifiers, and specialized databases) that extend the capabilities of LLMs beyond static answers. With the flexibility to access external resources and perform complex tasks, agent-based AI combines the language processing capabilities of LLMs with the precision of rule-based programming. This enables AI systems to generate accurate, context-aware answers in a wide range of applications, opening the way to new levels of usefulness and responsiveness [3–5].

The article analyzes problems and challenges of development and implementation of distributed intelligence based on LLM in various social spheres.

2. Multi-agent systems based on large language models

Multi-agent systems (MAS) are a paradigm in which different agents, each with specialized roles, collaborate to achieve common goals. This approach improves problem solving efficiency by providing robustness and adaptability that are difficult to achieve with individual agents. In MASs, agents can be programmed to perform specific tasks and interact with each other, allowing for a decentralized method of solving complex problems. These systems work well in environments where tasks can be distributed among agents with different expertise, allowing for simultaneous execution of tasks and real-time problem solving.

The collaborative nature of MASs is particularly useful in scenarios that require distributed intelligence and decision making. Each agent in the system acts autonomously, but their actions are coordinated to achieve the overall goals of the system. This structure provides flexibility and resilience, as the system can continue to function effectively even if individual agents fail or are removed. In addition, MASs can adapt to changing conditions and requirements, making them suitable for dynamic and complex applications such as logistics, autonomous vehicles, and resource management.

At its core, a MAS consists of multiple autonomous entities, or agents, each with unique capabilities, knowledge, and goals. These agents interact with each other and with the environment, often making decisions based on local observations and rules. Unlike centralized AI systems, where one person makes all decisions, in a MAS, intelligence is distributed across multiple agents, allowing for decentralized decision making and emergent behavior that can adapt to changing conditions [6].

Agent-based AI belongs to systems that have some degree of autonomy, capable of making decisions, planning actions, and learning from experience to achieve specific goals without ongoing human intervention. Agent-based AI systems are designed to possess "agency", meaning they can autonomously perform tasks on behalf of users or other systems by developing their own workflows and using available tools. Unlike traditional large language models that respond to queries within given parameters, agent-based AI can adapt and learn from new data and user interactions, plan and execute complex, multi-step tasks, interact with external tools and databases, and improve over time through feedback and continuous learning [7, 8].

3. Multi-agent systems based on LLM

The integration of LLM into MAS marks a significant advance in the development and capabilities of MAS. LLM-based MAS leverage LLM's reasoning, planning, and decision-making capabilities, enabling agents to perform tasks that require a high degree of cognitive processing. These systems can understand and generate human-like text, making them ideal for applications involving natural language processing and interaction. LLM can help agents better understand each other's goals and strategies, facilitating smoother collaboration and efficient task execution.

LLM-based agents can conduct complex dialogues, interpret contextual information, and make informed decisions based on vast amounts of data. This ability is especially useful in scenarios where agents must understand subtle instructions or collaborate on tasks that require deep reasoning. By using LLM, MAS can improve their ability to solve complex and dynamic problems, providing more robust and adaptive solutions than traditional approaches. This level of collaboration is essential in environments where precise coordination and adaptive response are critical to success [9].

3.1. Topology of multi-agent systems

Topology in the context of MAS refers to the arrangement or structure of different agents in the system and how they interact with each other. This concept is important to understand the communication patterns, coordination mechanisms and overall effectiveness of MAS. Here is a deeper understanding of what topology means and what its implications are. MASs can be divided into different topological types depending on the roles and interactions of the agents. The topology of MAS influences on how agents communicate, coordinate and perform tasks, which affects the effectiveness and reliability of the system as a whole.

3.2. Peer-to-peer topology

In a peer-to-peer topology, agents operate at the same hierarchical level. Each agent has its own role and strategy, but no agent has a hierarchical advantage over others. This topology supports cooperation and competition in achieving common goals without centralized leadership. Decentralized decision-making allows agents to work independently, but still contribute to achieving the overall goals of the system. This approach ensures equal status for all agents, which is especially suitable for problems that require collective decision-making and shared responsibility.

3.3. Hierarchical topology

Hierarchical topologies include a leader and one or more followers. The leader agent is responsible for direction, planning, and strategic decision making, while the follower agents carry out tasks based on the leader's instructions. This centralized decision-making process allows for a clear separation of roles and responsibilities, making it effective in scenarios that require coordinated efforts under the direction of a central authority. The role of the leader agent is critical to ensuring that the overall strategy is consistent and that tasks are aligned with the system's goals.

3.4. Nested topology

A nested topology combines peer-to-peer and hierarchical structures to create a flexible system in which agents can form substructures to solve complex problems. These substructures operate either through peer-to-peer interactions or through hierarchical command chains within the larger system. This dual approach allows agents to leverage the benefits of both peer-to-peer and hierarchical mechanisms to facilitate efficient task execution and coordination. The flexibility of nested topologies allows agents to dynamically create subsystems as needed, increasing the overall adaptability and efficiency of the system [10].

3.5. Dynamic topology

Dynamic topologies are characterized by the roles, relationships, and number of agents changing over time. This inherent adaptability of the system allows it to reconfigure itself in response to changing tasks and conditions. This flexibility is essential for maintaining efficiency and effectiveness in rapidly changing environments. Agents within a dynamic topology can join or leave the system as needed, ensuring the system's resilience and ability to adapt to new tasks.

This high degree of flexibility and adaptability makes dynamic topologies particularly suitable for environments in which task requirements change frequently. For example, in a multi-agent system designed to respond to natural disasters, the number and roles of agents can be changed depending on the situation. This capability allows the system to deploy additional resources or reassign tasks as the disaster scenario evolves, ensuring an effective and coordinated response. The ability to dynamically adapt to new information and circumstances increases the overall stability and effectiveness of the system in managing complex, unpredictable situations [11, 12].

4. Planning in multi-agent systems

Planning in MASs is a critical process that includes both global and local planning. Effective planning ensures that agents can work together effectively, adapt to dynamic conditions, and ultimately achieve the system's goals [13].

4.1. Global planning

Global planning involves understanding the overall task and developing work processes that take advantage of each agent's specialization. This high-level planning phase is necessary to establish the basis for the system operation and ensure that all agents work in concert to achieve the overall goal.

- Task Decomposition: The first step in global planning is to decompose the main task into smaller, more manageable sub-tasks. This requires analyzing the task requirements and breaking them down into components that can be performed by individual agents or groups of agents.
- Workflow Design: Once the tasks have been decomposed, work processes are designed to define how the tasks will be performed. This includes defining the sequence of activities, communication protocols, and the roles of each agent. The goal is to maximize the efficiency and capabilities of each agent by making effective use of their specialized skills.
- Task Allocation: Assigning tasks to agents based on their specialization and capabilities is a critical part of global planning. This ensures that each agent works on tasks that best suit their strengths, resulting in improved overall system performance.
- Alignment with overall goals: Ensuring that individual tasks are aligned with the overall goals of the system is critical. This alignment helps maintain consistency in the system actions and effectively achieve desired results.

4.2. Local planning

Local planning focuses on decomposing the tasks assigned to each agent into manageable subtasks. This detailed level of planning ensures that agents can effectively complete their tasks and contribute to the collective goal.

- Subtask decomposition: Each agent takes its assigned tasks from the global plan and further decomposes them into specific actions that it can perform. This may include detailed planning of the steps, resources needed and expected results for each subtask.
- Execution strategies: Agents develop strategies for effectively completing their subtasks. This includes planning the sequence of actions, optimizing resource use, and scheduling tasks to meet deadlines or milestones.
- Goal alignment: Even at the local level, it is important for agents to ensure that their actions are consistent with the collective goals. Agents must continually adapt and update their plans based on feedback and changes in the environment to stay aligned with the overall goal.

4.3. Planning challenges

To ensure effective planning in MAS, several challenges must be addressed:

- Designing effective workflows: Designing effective and flexible workflows is challenging, especially in a dynamic environment. Workflows must be robust enough to handle uncertainty and adapt to changes in the system or task requirements.
- Iterative processes: Managing iterative processes to improve intermediate results is essential. This includes refining plans based on feedback, optimizing performance through iterative cycles, and ensuring continuous improvement in task execution.
- Coordination and communication: Ensuring effective coordination and communication between agents is critical to the success of the system. Agents must be able to share information, agree on roles, and synchronize their actions to achieve a collective goal.

- Scalability: As the number of agents and the complexity of tasks increases, maintaining effective planning becomes increasingly difficult. The system must scale efficiently to handle a large number of tasks without sacrificing performance and coordination [14].

5. Agent memory and information retrieval

Effective memory management is critical to the operation of MAS. The ability to store, manage, and retrieve information allows agents to work more intelligently and collaboratively. Here, we take a closer look at the types of memory used in MAS and the challenges associated with them.

5.1. Types of memory

Short-term memory is used during the ongoing interaction, is transient, and does not persist after the conversation ends. This type of memory helps agents keep track of the immediate context and perform tasks in real time. For example, during negotiations between agents, short-term memory is used to store the most recent offers and counteroffers, allowing agents to make informed decisions based on the most recent information received from the exchange.

Long-term memory stores historical requests and responses, providing context for future interactions. This type of memory helps agents recall and learn from past experiences, thereby improving their performance over time. For example, a customer service agent can use long-term memory to recall previous interactions with a customer, allowing them to provide more personalized and effective assistance based on the customer's history and preferences.

Integrating large language models with external databases improves the accuracy and relevance of MAS responses. This type of memory allows agents to access vast amounts of information that is not stored internally. For example, agents can use external stores to obtain updated weather or stock price information that is important for decision making but is not stored in their internal memory. With this capability, agents can provide informed and timely responses using comprehensive and relevant data sources [15].

Episodic memory captures interactions within a system and allows agents to refer to past interactions to improve problem solving. This type of memory is similar to human episodic memory, which recalls specific events or episodes. For example, in a multi-agent system designed to respond to emergency situations, agents can use episodic memory to recall past emergency scenarios and apply the lessons learned to current situations. This ability allows agents to use historical data to improve decision making and performance in a dynamic and complex environment.

The consensus memory in a MAS serves as a single source of common information for agents, ensuring consistency and smooth interaction. This is essential for maintaining a common understanding and coordinated actions across the entire system. For example, in a collaborative robotics system, the consensus memory ensures that all robots have the same map of the environment and understand their respective tasks and positions. This common knowledge base allows agents to work together to make informed decisions that contribute to the overall efficiency and effectiveness of the system [16].

5.2. Memory management issues

- Hierarchical Storage: Managing a hierarchical storage where different agents have different levels of access and needs can be complex. Ensuring that sensitive information is protected while providing the necessary access to the relevant data requires robust access control mechanisms. Implementing multi-level access control and encryption can help manage multi-level storage effectively and ensure that only authorized agents have access to sensitive information. This approach ensures the protection of critical data while maintaining the efficiency and functionality of the system by providing agents with appropriate access according to their roles and requirements.

- Maintaining the integrity of the consensus memory: Ensuring the integrity of the consensus memory is very important as any tampering or unauthorized changes can lead to system failures. Maintaining a consistent and accurate consensus ledger is essential for effective collaboration between agents. The use of distributed ledger systems can ensure a secure and immutable record of shared information, thereby enhancing the integrity of the consensus memory. This approach ensures that all agents have access to reliable and tamper-proof data, which is necessary for coordinated and efficient operations in the system.
- Effective communication and information sharing: Ensuring effective communication and information sharing between agents is necessary for coordinated actions, since miscommunication or data loss can disrupt the operation of the system. Developing reliable communication protocols and redundancy mechanisms can ensure reliable information sharing and reduce the risk of miscommunication. This approach helps maintain the integrity and efficiency of the system, allowing agents to effectively collaborate and achieve common goals despite possible communication problems.
- Use of episodic memory: Effective use of episodic memory to improve responses to new queries requires retrieving and using contextually relevant past interactions, which requires sophisticated algorithms to accurately match current queries with past experiences. Advanced machine learning techniques, such as memory-enhancing neural networks, can help agents effectively recall and apply past experiences to new situations. This approach allows agents to improve their decision-making and problem-solving capabilities by drawing on a rich history of relevant interactions, resulting in more informed and effective responses.
- Scalability and efficiency: As the number of agents and task complexity increases, maintaining effective memory management becomes increasingly challenging. The system must scale to handle larger volumes of data and more complex interactions without degrading performance. Scalable storage architectures and distributed storage solutions can support the growth of MAS, ensuring that the system remains efficient and effective as it expands. This approach allows the system to cope with increased demands while maintaining high performance and reliability, ensuring smooth operation in increasingly complex and data-rich environments [17, 18].

6. Bringing agent-based AI to industry

One of the main reasons why agent-based AI is a breakthrough in AI research is its ability to combine flexibility with precision. LLMs excel at processing and generating human-like text, making it easy for users to interact with AI using natural language commands. They can generate responses or actions based on subtle, context-sensitive understanding, which is useful in scenarios where traditional programming may not be able to handle all possible situations. Traditional programming, on the other hand, offers structured, deterministic algorithms that are ideal for tasks that require precision, repeatability, and verifiability. Agent-based AI systems leverage the flexibility of LLMs to solve problems that require dynamic responses, while relying on traditional programming to provide rigorous rules, logic, and performance. This combination allows AI to be both intuitive and precise [18].

Autonomy is another distinctive feature of agent-based AI systems. They can operate independently and autonomously perform certain tasks without the need for constant human supervision, enabling continuous operation in environments where human supervision is limited or unnecessary. Autonomous systems can support long-term goals, manage multi-stage tasks, and track progress over time.

6.1. Autonomous management of marketing campaigns

An agent-based AI tasked with managing a marketing campaign can autonomously perform various stages of campaign management, from content creation to performance analysis. For example, an agent might start by developing targeted content based on audience data and recent engagement trends. It would then distribute that content across multiple platforms–social media, email, or websites – and track the campaign's performance in real time. As engagement data comes in, the AI can dynamically adjust its strategy, such as shifting budget to more effective channels or changing the tone of messaging to resonate with a specific demographic. If a certain ad format is underperforming, the AI can replace it with a more effective one. This capability not only optimizes campaign results, but also allows marketers to focus on creative strategy and innovation. The agent can work continuously, ensuring that the campaign always aligns with business goals and adapting in real time to maximize ROI.

6.2. Monitoring health status and adjusting treatment in real time

In healthcare, agent-based AI could be a game changer in patient care by continuously monitoring a patient's vital signs, medication adherence, and other health metrics. For example, an agent could track data from wearable devices, such as heart rate, blood pressure, and oxygen levels, and compare these metrics with historical data and clinical guidelines. If any metrics fall outside safe parameters, the agent could alert medical staff or automatically adjust the treatment plan, such as increasing medication dosage or recommending diagnostic testing.

For chronic conditions such as diabetes, an agent can continuously analyze blood glucose levels and suggest dietary adjustments to the patient in real time. By autonomously monitoring patient data and providing clinicians with actionable insights, agent-based AI supports proactive and responsive treatment, potentially improving patient outcomes and reducing hospital visits. It allows healthcare professionals to focus on complex and sensitive care while the AI handles routine monitoring and adjustments [19].

6.3. Autonomous cybersecurity surveillance and threat response

In the cybersecurity space, agent-based AI can act as a vigilant and tireless guard, constantly monitoring network traffic, system logs, and user behavior for potential threats. For example, an agent can monitor access patterns to a company's systems, identifying any unusual activity that might indicate phishing attacks, malware intrusions, or unauthorized access attempts. If a particular user's behavior deviates significantly from the norm—such as multiple failed login attempts or accessing sensitive data outside of normal hours—the AI can flag this as suspicious and immediately initiate security protocols.

Additionally, AI can analyze incoming network traffic for signs of distributed denial of service (DDoS) attacks, blocking potentially malicious IP addresses in real time. When a threat is detected, AI can autonomously deploy countermeasures, such as isolating affected servers or restricting access to compromised accounts. This level of autonomous threat response allows organizations to maintain robust security even in the face of rapidly evolving cyber threats, while requiring minimal human intervention for routine security monitoring [12, 20].

7. Discussion

However, as exciting as the prospects of agent-based AI are, they are not without problems. Ethical considerations, such as ensuring that these systems make decisions in line with human values, are paramount. The complex nature of AI models can make decision-making processes difficult to understand and interpret, creating accountability and trust issues, especially in high-stakes applications. There is also the question of liability – who is responsible if an agent-based AI makes a mistake? Data privacy and security are another major concern as these systems become increasingly autonomous and handle increasingly sensitive information. Robust safeguards are needed to protect against misuse or security breaches.

While agent-based AI promises transformative applications across industries, it also poses challenges related to trustworthiness. The accuracy and reliability of the information provided by AI agents ideally requires human oversight, but with agent-based AI, this oversight can be compromised. This challenge arises from the probabilistic nature of large language models, which are inherently non-deterministic and can sometimes produce “hallucinations” – answers that seem plausible but are actually incorrect. If data sources are limited or if the wrong processing step occurs, agent-based AI can generate unfounded assumptions or misleading answers. These inaccuracies are especially problematic in sensitive areas such as healthcare or legal advice, where accurate and verified information is important for ensuring safety, compliance, and overall trust.

8. Conclusion

Multi-agent systems exhibit remarkable collective intelligence and the ability to adapt to changing conditions, making them useful for solving complex problems and modeling social processes. Despite the existing problems, the potential benefits of agent-based AI are too great to ignore. As research in this area advances, we can expect to see increasingly sophisticated AI agents that can collaborate with humans in ways we have only seen in science fiction. The key to harnessing the full potential of agent-based AI lies in striking the right balance between autonomy and human control. Thoughtful design of such systems and careful consideration of ethical implications will enable us to create AI agents that complement human capabilities rather than replace them.

Author Contributions: Conceptualization, Shchetinin E.; methodology, Shchetinin E., Yurgina L.; writing—review and editing Shchetinin E., Velieva T., Demidova A.; supervision, Demidova A., Velieva T.; project administration, Shchetinin E., Yurgina L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing is not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Minsky, M. *Society of mind* (Simon and Schuster, 1988).
2. Russell, S. J. *Artificial intelligence a modern approach* (Pearson Education, Inc., 2010).
3. Ginsberg, M. L. *Essentials of Artificial Intelligence* (Morgan Kaufmann, 1993).
4. Wooldridge, M. & Jennings, N. R. Intelligent agents: theory and practice. *The Knowledge Engineering Review* **10,** 115–152. doi:10.1017/S0269888900008122 (1995).
5. Linardatos, P., Papastefanopoulos, V. & Kotsiantis, S. Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy* **23,** 18. doi:10.3390/e23010018 (2021).
6. Han, S., Zhang, Q., Yao, Y., Jin, W., Xu, Z. & He, C. *LLM Multi-Agent Systems: Challenges and Open Problems* 2024.
7. Wang, L. *et al. A survey on large language model based autonomous agents* 2023.
8. Andreas, J. *Language Models as Agent Models* in *Findings of the Association for Computational Linguistics: EMNLP 2022* (eds Goldberg, Y., Kozareva, Z. & Zhang, Y.) (Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, Dec. 2022), 5769–5779. doi:10.18653/v1/2022.findings-emnlp.423.
9. Brooks, R. A. *The artificial life route to artificial intelligence* in. Chap. Intelligence without reason (Routledge, 2018).
10. *Openai: Introducing chatgpt. Website* 2022.
11. Thirunavukarasu, A., Ting, D., Elangovan, K., Gutierrez, L., Tan, T. & Ting, D. Large language models in medicine. *Nature Medicine* **29,** 1930–1940. doi:10.1038/s41591-023-02448-8 (2023).
12. Chan, C.-M., Chen, W., Su, Y., Yu, J., Xue, W., Zhang, S., Fu, J. & Liu, Z. *ChatEval: Towards Better LLM-based Evaluators through Multi-Agent Debate* 2023.
13. Cai, T., Wang, X., Ma, T., Chen, X. & Zhou, D. *Large language models as tool makers* 2023.
14. Wu, T., Terry, M. & Cai, C. J. *AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts* in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (Association for Computing Machinery, New York, NY, USA, 2022), 385. doi:10.1145/3491102.3517582.
15. Mohtashami, A. & Jaggi, M. *Landmark attention: Random-access infinite context length for transformers* 2023.
16. Bar, A., Gandelsman, Y., Darrell, T., Globerson, A. & Efros, A. *Visual Prompting via Image Inpainting* in *Advances in Neural Information Processing Systems* (eds Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K. & Oh, A.) **35** (Curran Associates, Inc., 2022), 25005–25017.
17. Mustakim, M., Pratama, A. R., Ahmad, I., Arifianto, T., Sussolaikah, K. & Sepriano, S. *Image classification of corn leaf diseases using CNN architecture ResNet-50 and data augmentation* in *2024 International Conference on Decision Aid Sciences and Applications (DASA)* (2024), 1–6.
18. Shchetinin, E. Y. On Using Computer Linguistic Models in the Classification of Biomedical Images. *Mathematical Models and Computer Simulations* **2** (2024).
19. Shchetinin, E. Y., Glushkova, A. G. & Demidova, A. V. Developing a computer system for student learning based on vision-language models. *Discrete and Continuous Models and Applied Computational Science* **32,** 234–241 (2024).
20. Wu, Q. *et al. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation* 2023.

Information about the authors

Shchetinin, Eugeny Yu.—Doctor of Physical and Mathematical Sciences, Lecturer of Artificial Intelligence Department (e-mail: riviera-molto@mail.ru, ORCID: 0000-0003-3651-7629, ResearcherID: O-8287-2017, Scopus Author ID: 16408533100)

Velieva, Tatyana R.—Candidate of Physical and Mathematical Sciences, Senior lecturer of Department of Probability Theory and Cyber Security (e-mail: velieva-tr@rudn.ru, ORCID: 0000-0003-4466-8531)

Yurgina, Lyubov A.—Ph.D. of Pedagogical Sciences, Head of the Department of Mathematics and Information Technology of the Sochi branch of RUDN University (e-mail: yurgina_la@pfur.ru, ORCID: 0009-0004-4661-5059)

Demidova, Anastasia V.—Candidate of Physical and Mathematical Sciences, Assistant Professor of Department of Probability Theory and Cyber Security (e-mail: demidova-av@rudn.ru, ORCID: 0000-0003-1000-9650)

УДК 519.7

PACS 07.05.Tp

DOI: 10.22363/2658-4670-2025-33-2-214-225 EDN: MKTGEJ

О моделировании мультиагентных систем на основе больших языковых моделей

Е. Ю. Щетинин[1], Т. Р. Велиева[2], Л. А. Юргина[3], А. В. Демидова[2]

[1] Финансовый университет при Правительстве Российской Федерации, Ленинградский проспект, д. 49, Москва, 125993, Российская Федерация

[2] Российский университет дружбы народов, ул. Миклухо-Маклая, д. 6, Москва, 117198, Российская Федерация

[3] Сочинский институт (филиал) Российского университета дружбы народов, ул. Куйбышева, д. 32, Сочи, 354340, Российская Федерация

Аннотация. В статье изучается эффективность внедрения мультиагентных систем на основе больших языковых моделей в различных сферах человеческой деятельности, анализируются их преимущества, проблемы и задачи. Результаты исследования показали, что мультиагентные системы на основе больших языковых моделей обладают значительным потенциалом и широкими возможностями в моделировании различных сред и решении различных задач.

Ключевые слова: искусственный интеллект, финансовые риски, большие данные