МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

ВІСНИК
ЧЕРКАСЬКОГО ДЕРЖАВНОГО ТЕХНОЛОГІЧНОГО УНІВЕРСИТЕТУ

Науковий збірник

Том 28,
№ 4. 2023

ЧЕРКАСИ
2023

ISSN: 2306-4412
E-ISSN: 2708-6070

Засновник:
Черкаський державний технологічний університет

Рік заснування: 1996

Рекомендовано до друку та поширення
через мережу Інтернет Вченою радою
Черкаського державного технологічного університету
(протокол № 5 від 18 грудня 2023 р.)

Свідоцтво про державну реєстрацію
друкованого засобу масової інформації
серії КВ 6061 від 16 квітня 2002 р.

Журнал входить до переліку наукових фахових видань України
Категорія «Б». Спеціальність:
122 – Комп'ютерні науки, 123 – Комп'ютерна інженерія, 125 – Кібербезпека,
126 – Інформаційні системи та технології, 151 – Автоматизація та комп'ютерно-інтегровані технології, 172 – Телекомунікації та радіотехніка
(наказ МОН України від 28 грудня 2019 року № 1643)

Журнал представлено у міжнародних наукометричних базах даних, репозитаріях та пошукових системах: Index Copernicus Journals Master List, BASE, CiteFactor, Google Scholar, Crossref, Directory of Open Access Journals (DOAJ), Eurazian Scientific Journal Index (ESJI), Ulrich's Periodicals Directory, WorldCat, Open Ukrainian Citation Index

Адреса редакції:
Черкаський державний технологічний університет
18006, бульв. Шевченка, 460, м. Черкаси, Україна
E-mail: info@bulletin-chstu.com.ua
https://bulletin-chstu.com.ua/uk

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
CHERKASY STATE TECHNOLOGICAL UNIVERSITY

BULLETIN
OF CHERKASY STATE TECHNOLOGICAL UNIVERSITY

Scientific Collection

Volume 28,
No. 4. 2023

CHERKASY
2023

ISSN: 2306-4412
E-ISSN: 2708-6070

Founders:
Cherkasy State Technological University

Year of foundation: 1996

Recommended for printing and distribution
via the Internet by the Academic Council
of Cherkasy State Technological University
(Minutes 5 of December 18, 2023)

Certificate of state registration
of the print media
Series KV No. 6061 of April 16, 2002

The journal is included in the list of Professional Scientific Publications of Ukraine
Category "B". Branch of sciences – Technical.
Specialty: 122 – Computer Science, 123 – Computer Engineering, 125 – Cybersecurity,
126 – Information Systems and Technologies, 151 – Automation and Computer-Integrated
Technologies, 172 – Telecommunications and Radio Engineering
(order of the Ministry of Education and Science of Ukraine No. 1643, dated December 28, 2019)

The journal is presented international scientometric databases, repositories
and scientific systems: Index Copernicus Journals Master List, BASE, CiteFactor, Google Scholar,
Crossref, Directory of Open Access Journals (DOAJ), Eurazian Scientific Journal Index (ESJI),
Ulrich's Periodicals Directory, WorldCat, Open Ukrainian Citation Index

Editors office address:
Cherkasy State Technological University
18006, 460 Shevchenko Blvd., Cherkasy, Ukraine
E-mail: info@bulletin-chstu.com.ua
https://bulletin-chstu.com.ua/en

Редакційна колегія

Editorial Board

Зміст / Contents

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 004.91
DOI: 10.62660/2306-4412.4.2023.10-18

Review of joint text editing algorithms Conflict-free Replicated Data Types (CRDT)

Yurii Rabeshko*
Postgraduate Student
National University of Water Management and Nature Management
33000, 11 Soborna Str., Rivne, Ukraine
https://orcid.org/0009-0002-0763-7138

Yurii Turbal
Doctor of Technical Sciences, Professor
National University of Water Management and Nature Management
33000, 11 Soborna Str., Rivne, Ukraine
https://orcid.org/0000-0002-5727-5334

Abstract. Analysing and selecting algorithms for collaborative text editing, especially implementing conflict-free replicated data types, is critical to understanding how modern systems can achieve real-time collaboration while ensuring data integrity. The purpose of the study is to review various collaborative editing algorithms and conduct a comparative analysis to understand their advantages, disadvantages, and applications. Statistical methods, methods for analysing algorithms and their use in real-world scenarios are used. The study results showed that the use of collaborative text editing algorithms contributes to solving important tasks and challenges in the modern world of information technology. Collaborative editing algorithms are determined to facilitate real-time communication and information exchange. This is especially important in the context of remote work and communication, which have become the standard for many organisations. Collaborative text editing is widely used in various fields and industries where teamwork, real-time collaboration, and document sharing are essential, such as scientific research, education, software development, book and manuscript editing, legal cooperation, contract draughting, medical reports, etc. In addition, using optimised collaborative editing algorithms helps reduce the time required to process data and create text materials. Collaborative editing algorithms have a wide range of applications in research, business, and education. They allow teams and individual users to solve problems more efficiently and work together on projects. The ability to use the results obtained in practical activities will allow using algorithms for joint text editing for further innovation and development of information technologies, which will allow working together and exchanging information with the whole world in real-time

Keywords: distributed systems; key operations; Logoot; Astrong; Logarithmic Sequence; Replicated Growable Array

Article's History: Received: 04.09.2023; Revised: 10.11.2023; Accepted: 18.12.2023.

INTRODUCTION

The relevance of this study lies in the development of conflict-free replicated data type algorithms (CRDT), which are becoming an essential part of digital life and have a wide range of applications in various industries. The use of CRDT has become an important tool for editing text and large amounts of information, and working together on a document. In addition, the use of collaborative text editing algorithms can help increase efficiency and productivity. Due to CRDT, users can work together on texts in real-time, make joint

Suggested Citation:
Rabeshko, Yu., & Turbal, Yu. (2023). Review of joint text editing algorithms Conflict-free Replicated Data Types (CRDT). *Bulletin of Cherkasy State Technological University,* 28(4), 10-18. doi: 10.62660/2306-4412.4.2023.10-18.

*Corresponding author

edits and changes. This is especially important in teams and groups of people working on joint projects or documents from a distance. Research on collaborative text editing algorithms is important because it helps understand how these algorithms work, their advantages, and limitations. This opens up the possibility of developing new and improved collaborative editing techniques that help improve the efficiency and productivity of working with texts. Such investigations can also have practical applications in various fields, including education, scientific, business, etc.

The main problem of the study is the preservation of consistency in distributed systems, as they are becoming more common in the modern digital world, and the emergence of conflict situations when trying to update data simultaneously. This problem is particularly relevant in modern digital world, where working together on documents and data in real-time is becoming the standard. Investigating collaborative text editing algorithms allows understanding how these algorithms help solve this problem and what aspects of distributed systems and conflicting text editing affect their consistency and efficiency. The study results can have important practical applications for developing systems that provide reliable and conflict-free collaborative text editing, which is becoming an increasingly important aspect in the modern world of information technology.

According to N. Saquib *et al.* (2022), CRDT (Conflict-free Replicated Data Types) are abstract data types that provide a principled approach to asynchronous data mismatch matching. The researchers noted that this is very important for modern distributed systems, where data access must be reliable and fast, even in possible data discrepancies. The paper does not provide detailed instructions on the practical implementation of algorithms. A study on cloud services that support real-time document collaboration was conducted by O. Medvedovska and V. Yatsenko (2021). They examined the functionality of individual applications, such as Google Docs and Google Drive, that use CRDT structures. The results obtained proved that the use of a tool that supports joint work on the document showed labour efficiency. The study does not consider collaborative text editing algorithms as an object of research, but the effectiveness of using these algorithms can be seen in the example of the Google Docs cloud environment since collaboration on a document is conducted without conflicts.

F. Hu and R.H. Trivedi (2020) used the RGA (Replicated Growable Array) algorithm to position a hotel brand and map the competitive landscape using text-based content analysis. This approach demonstrates the importance of collaborative text editing algorithms in various areas, including analysing large amounts of text information for making business decisions and improving brand strategies. The use of algorithms such as RGA helps in effective data management and analysis and in solving tasks related to text editing and collaboration on documents and information. However, the study does not include a comparative analysis of the RGA algorithm with other text analysis methods.

According to M. Kleppmann *et al.* (2021), replicated tree data structures are the fundamental building block of distributed file systems. Their paper introduced the LSEQ (Logarithmic Sequence) algorithm, which handles arbitrary simultaneous modifications on trees while ensuring that the tree structure remains valid. However, this paper does not provide detailed guidance on the practical implementation of the LSEQ algorithm. O. Shchetynina *et al.* (2022) reviewed the Trello multi-platform project management system. In the study, an experiment was conducted that proved the effectiveness of using Trello. The paper did not consider CRDT as an object of research, but an experiment and the use of the Trello platform demonstrated the effectiveness of using these algorithms. From a study by H.H. Kirychek and A.I. Chubich (2020), this platform uses a project management paradigm known as kanban. This paradigm uses methods that prevent overloading team members.

According to D. Brahneborg *et al.* (2022), most CRDTs are based on some variant of atomic translation, as this allows them to maintain causal relationships between updates of multiple objects. However, the overhead of this atomic translation is unnecessary in systems that process only independent CRDT objects. The researchers identified a set of use cases for tracking resource usage, where there is a need for a replication mechanism with less complexity and network usage compared to using atomic translation. The study does not include a comparative analysis of CRDT with other data replication methods, including traditional approaches.

The purpose of the study is to review and compare CRDTs, such as Logoot, RGA, LSEQ, Astrong, and understand the advantages and disadvantages of each of the algorithms. The following tasks were formed in the examination of joint text editing algorithms to achieve this goal: detailed analysis of joint text editing algorithms, in particular, Logoot, RGA, LSEQ, and Astrong, with the definition of their basic principles and functionality; identifying the advantages and disadvantages of each of these algorithms to evaluate their effectiveness and suitability for specific use cases; considering the problems of conflicts and consistency that may arise when using these algorithms and suggesting possible strategies for solving them. This study can help to understand the algorithms used in text editing and create new approaches to collaborative text editing.

MATERIALS AND METHODS

By analysing the papers of researchers such as: B. Nédelec *et al.* (2013), M. Nicolas *et al.* (2020) and F. Jacob *et al.* (2021), data containing information about the performance and efficiency of various algorithms and their functioning under different usage conditions were selected. This analysis allowed obtaining

objective information on how different text editing algorithms work in real-world scenarios. Using this, the advantages and disadvantages of each of the algorithms under study, their performance, and their interaction with other parts of the system were established. This approach collecting objective data that plays an important role in making informed decisions when choosing and configuring algorithms for specific applications.

In addition, statistical methods were used for a detailed examination of joint text editing algorithms. Using these methods allowed conducting a quantitative analysis of data to determine the statistical importance of various parameters and factors affecting the performance of algorithms. Statistical methods also helped in the comparative analysis of various algorithms, identifying their advantages and disadvantages in terms of statistical indicators. This approach allowed obtaining objective data on the effectiveness and performance of various CRDTs, which is important for making informed decisions when choosing a specific algorithm for a specific application. This analysis also allowed identifying opportunities for improving and optimising existing algorithms to improve their efficiency.

In the course of the study, the method of analysing algorithms was used. This technique involved a detailed review and analysis of the operation of specific algorithms for joint text editing and their theoretical foundations. The analysis of algorithms helped to reveal the specifics and features of each algorithm, understand exactly how they solve the problem of joint text editing and what aspects are important for their effective work. In addition, using the analysis, it was possible to determine exactly how each algorithm solves the problem of joint text editing. The analysis method thoroughly investigated the mechanisms underlying various CRDTs and allowed understanding how they handle and synchronise text editing between users.

Algorithms such as Logoot, RGA, LSEQ, and Astrong were also considered. These algorithms were chosen for examination and comparison because of their relevance and importance in the field of collaborative text editing. The selected algorithms are representative examples of different approaches to solving the problem of joint text editing, and each of them has its own unique features and advantages. A comparative analysis of these algorithms allowed identifying which specific algorithms were more effective under certain conditions. A detailed analysis of the data identified the strengths and weaknesses of each algorithm, their speed and stability under different conditions of use. Implementation methods were considered for some of the above algorithms. This enabled a greater understanding of how these algorithms can be implemented in practise. In addition, the implementation of algorithms allowed clearly understanding the functioning of the algorithm and determining how they interact with real data and users in specific conditions.

RESULTS

Detailed overview of CRDT collaborative text editing algorithms and their functions. For a better understanding of collaborative text editing algorithms, it is necessary to get familiarised with the theoretical aspects and usage examples. Notably, an important aspect in examining of CRDT is the understanding of conflict-proof replicated data structures that underlie many collaborative text editing algorithms. In this section, the algorithms that are widely used in practise are discussed. Logoot is a CRDT algorithm for supporting collaborative editing on distributed systems. It was presented by S. Weiss *et al.* (2009). Logoot was designed to provide better performance than previous CRDT algorithms such as Treedoc and WOOT (WithOut Operational Transforms). Logoot is based on the idea of encoding each element in a data structure using a sequence of integers, enabling the efficient insertion and deletion operations.

The Logoot algorithm presented an editable document as a sequence of lines, where each line is a sequence of characters. Each character was represented as a tuple (i, c), where i – the unique identifier for that character and c – its value. Logoot gave each character a unique identifier, encoding its position in the document using a sequence of integers. Logoot used a tree-based data structure, which allowed efficient character insertion and deletion operations and fast search and retrieval. Figure 1 shows part of the Logoot algorithm implementation code.

```
import Logoot exposing (empty, insert, toList)
pid = ([(2, 3)], 0)
logoot1 = empty
    |> insert pid "hey!"
    |> remove pid "hey!"
    |> toList
logoot2 = empty
    |> remove pid "hey!"
    |> insert pid "hey!"
    |> toList
result = logoot1 == logoot2 -- True
```

Figure 1. Implementation of part of the Logoot algorithm in the Elm programming language
Source: GitHub (2016)

The presented code demonstrated the "commutativity" property of the Logoot algorithm when inserting and deleting characters. Notably, Logoot ensures that character insertion and deletion operations are commuted, meaning that the order in which they are performed does not affect the final result. Logoot supports three key operations: insert, delete, and merge. Logoot first generates a unique identifier for that character based on its position in the document to insert a new character into a document. The ID consisted of a sequence of integers that indicated a place in the document

structure. The algorithm then found a suitable place to insert a new character in the document, considering the order of identifiers. This allowed inserting characters in the correct order and ensuring that their IDs are unique. After that, the document structure was updated accordingly. Logoot simply marks the character as deleted, setting its value to null to remove a character from a document. The deletion operation does not remove the character from the document but marks it as deleted, which allows saving the document's editing history and resolving contradictions between different versions. Merging was used to resolve contradictions between different versions of a document that were edited simultaneously by different users. Logoot has integrated two versions of the document, applying changes from each version in the order they are executed. This maintains the consistency of the document and avoids conflicts.

The algorithm showed good performance in terms of scalability and memory usage. It can process documents of any size, and its memory usage increases logarithmically with the document size. Logoot also proved to be faster than previous CRDT algorithms, such as Treedoc and WOOT, especially for large documents. In one study, Logoot was compared with other CRDT algorithms for collective real-time editing. The results showed that Logoot has the best performance in terms of speed with low latency and high throughput while maintaining consistency and correctness (Karayel & Gonzàlez, 2022). RGA is a CRDT algorithm for supporting list-like data structures in distributed systems. It was introduced by Hyun-Gul Roh, Myeongjae Jeon, Jin-Soo Kim, and Joonwon Lee in 2011. RGA was designed to address the limitations of previous CRDT algorithms for list-like data structures such as WOOT and Treedoc. These algorithms were inefficient for large lists or for many simultaneous edits. RGA is based on a sequence of positions and characters, similar to Logoot, but uses the latest method for processing simultaneous edits (Roh *et al.,* 2011).

The algorithm represented a list as a sequence of positions and symbols. Each position represented the location of a character in the list, and each character had a unique identifier and value. Unlike Logoot and LSEQ, RGA did not use a hierarchical data structure but presented the list as a sequence of nodes connected by pointers. Figure 2 shows part of the implementation code for this algorithm. The above code is a representation of the sequence of operations for editing a document in joint text editing. RGA supports two main operations: insert and delete. Each character in the document has its own unique identifier, which is generated based on its value and position in the document. This allowed accurately defining each character and avoiding conflicts when inserting at the same time. RGA generates a unique identifier based on the previous and next characters in the sequence and assigned the character a position between them to insert a new character. This allowed inserting characters at any point in the sequence. RGA marked it as deleted, setting its value to null to delete the character. Deleted characters are not removed from the data structure, but they are marked as deleted and excluded from subsequent operations. If there are conflicts where two or more users delete or insert characters in the same place, the RGA uses the "bookmarks" technique (special markers that indicate that the character has been deleted). They are distributed to other nodes in the system to ensure consistency.

```
{
  agent: 'abc',
    seq: 10,
    ops: [{
      type: 'insert', // <-- This insert is ('abc', 10)
      content: 'x',
    }, {
      type: 'delete', // <-- deletes don't bump the version
    }, {
      type: 'insert', // <-- This insert is ('abc', 11)
      content: 'y',
  }]
}
```

Figure 2. Implementation of part of the RGA algorithm in the Rust programming language
Source: GitHub (2021)

The RGA algorithm showed good results in terms of scalability and performance. It can handle large lists, and a large number of simultaneous edits, and memory usage increases linearly with the size of the list. RGA also performed well in real-world applications, such as collective text editing and chat applications. V.B. Gomes *et al.* (2017) compared RGA with other CRDT algorithms for list-like data structures in a collective text editor. The results showed that RGA had better performance than other algorithms, with lower latency and higher throughput. RGA also had better convergence properties, which means it handled conflicts better and provided consistency when editing simultaneously. LSEQ is a CRDT algorithm for maintaining list data structures in distributed systems. It was developed by B. Nédelec *et al.* (2013). LSEQ was designed to improve the scalability and performance of previous CRDT algorithms for list data structures such as Treedoc and RGA. These algorithms had limitations in terms of scalability and efficiency, especially when working with large lists or high levels of simultaneous editing. LSEQ is based on a sequence of integers, similar to Logoot, but uses a more efficient encoding scheme.

The algorithm represented a list as a sequence of integers, where each number represents the position of a character in the list. The encoding scheme used by LSEQ enabled the efficient insertion and deletion of characters and quick searching for and bypassing them. LSEQ uses the binary search tree data structure to store integers, which allows efficiently combining different versions of the list. Each character in the document is represented by an integer corresponding to its position in the document. These numbers are encoded to

be unique and allow efficient insertion and deletion of characters. One of the critical features of LSEQ is the ability to effectively combine different versions of the list that occur when users edit simultaneously. This is done using a binary search tree and a special join algorithm. The main goal of LSEQ is to ensure efficient insertion and removal of characters in a text document and avoid conflicts between users who are editing the document at the same time.

LSEQ supports two key operations: insert and delete. A unique integer is generated for to insert a new character into the LSEQ list. This integer determines the position of the character in the document. Unique numbers are generated in such a way that they are unique and ensure that the characters are sorted. A character is removed from the LSEQ list by marking the character as deleted and setting its value to null. Deleted characters remain in the tree, but they are not counted in further editing operations. When conflicts occur during simultaneous edits, LSEQ uses a technique known as "path copying" to resolve these conflicts. This means creating a new subtree that represents the version of the list with the changes made. The path is copied from the root of the binary search tree to the position of the edited character.

The LSEQ algorithm showed good results in terms of scalability, performance, and memory usage. It is capable of handling very large lists and high levels of simultaneous edits, with memory usage increasing logarithmically as the list size increases. LSEQ also has good convergence properties, maintaining consistency and correctness even under simultaneous editing conditions. In the study of by M. Kleppmann *et al.* (2021), LSEQ was compared with other CRDT algorithms for list-like data structures in distributed text editors. The results showed that LSEQ has the best performance in terms of low-latency and high-throughput speeds, while maintaining consistency and correctness. LSEQ was identified to be more efficient than previous CRDT algorithms such as Treedoc and RGA, especially for large lists.

Astrong is a CRDT algorithm for supporting set-like data structures in distributed systems. It was presented by H. Attiya *et al.* (2016). Astrong is a CRDT algorithm designed to perform efficient and scalable operations on large sequences of ordered elements. It was proposed as a variant of the LSEQ algorithm to improve its scalability and performance characteristics. Astrong was specifically designed for use cases where the sequence size can be very large, such as in text editors or document management systems. The algorithm uses a multi-level tree structure to represent an ordered sequence of elements. At each level of the tree, the sequence is divided into several segments, each of which is represented by a unique identifier. These segment IDs are used to efficiently localise and manipulate individual elements in a sequence. Astrong uses a combination of linear and logarithmic data structures to represent segments and corresponding identifiers, allowing efficient indexing and manipulation of large sequences.

It provides efficient and scalable operations for adding and removing elements from a sequence. When an element is added to a sequence, it is assigned a unique identifier based on its position in the sequence. This ID specifies exactly where to insert an element into the data structure. Because the main data structure in Astrong is a multi-level tree, and due to the unique identifier, insertion was performed efficiently and quickly. This ID is then used to find the corresponding segment in a multi-level tree, where the element can be inserted without the need for any additional restructuring or restructuring of the data structure. Similarly, when an element is removed from a sequence, it is simply marked as deleted, without the need to actually delete data from the main data structure. This allows performing deletion operations without substantially affecting performance and rebuilding the structure. The Astrong algorithm was evaluated in various contexts and use cases, shown to provide efficient and scalable operations on large sequences of ordered elements. Compared to other CRDT algorithms, Astrong showed improved performance characteristics, especially when working with very large sequences or a large number of concurrent users. Astrong's data structure and implementation were carefully designed to balance the trade-off between space complexity, time complexity, and scalability, allowing efficient handling of a wide range of use cases.

Comparison of CRDT algorithms for collaborative text editing in distributed systems. Logoot and RGA use different data structures to represent an ordered sequence. Logoot uses a tree data structure, while RGA uses a sequence of nodes connected by pointers. LSEQ and Astrong use hierarchical trees to represent the sequence, but they differ in implementation details. The different data structures used by these algorithms have different trade-offs between space-time complexity and scalability (Karayel & Gonzàlez, 2022). All CRDT algorithms provide some level of support for concurrent editing of ordered sequences but differ in resolving conflicts that occur when multiple users edit the same part of the sequence simultaneously. Logoot and RGA use the "last editor wins" strategy to resolve conflicts, while LSEQ and Astrong use a more sophisticated conflict resolution engine that considers the unique identifiers assigned to each element in the sequence. This mechanism allows LSEQ and Astrong to resolve conflicts more efficiently than Logoot and RGA.

The scalability and performance of these CRDT algorithms depend on a variety of factors, including sequence size, number of users, and frequency of simultaneous edits. Logoot and RGA are known to have scalability and performance limitations, especially when working with large sequences or a large number of users. LSEQ and Astrong were designed to address some of these limitations and demonstrated good performance in various scenarios (Kleppmann *et al.*, 2019). Each of these CRDT algorithms has its own unique

advantages and disadvantages, which makes them more suitable for certain use cases. Logoot and RGA are suitable for applications that require low space complexity and simple conflict resolution mechanisms but may have scalability and performance issues in more complex scenarios. LSEQ is suitable for applications that require efficient indexing and more advanced conflict resolution mechanisms but may have a greater time complexity. Astrong is designed for use cases that require efficient processing of very large sequences but can have greater space complexity.

The main results showed the advantages and disadvantages of each of the algorithms and demonstrated the importance of collaborative text editing algorithms. For example, Logoot is characterised by good performance of add and remove operations but it can cause more conflicts when making simultaneous changes. On the other hand, LSEQ provides a low probability of conflicts but may be less efficient when inserting new characters. The study showed that collaborative text editing algorithms are of great importance for distributed systems. They allow users to work together on documents regardless of their physical location and network availability. This is especially important for teamwork and collaborative editing of real-time text documents. The study also focused on evaluating the performance and performance of each algorithm under different usage conditions. This helps to determine which algorithm is best suited for specific tasks and use cases. It is important to understand that different algorithms may be optimal for different application cases. During the study, it was determined that some algorithms may be more difficult to implement than others. This may affect the speed of development and support of the system. In addition, when multiple users work on the same document simultaneously, conflicts may occur. Research on collaborative editing algorithms helps determine which algorithms better resolve these conflicts and ensure that changes are properly integrated.

Thus, collaborative text editing algorithms are important components for modern distributed systems and collaborative text document editing. They help solve problems related to teamwork on real-time documents and ensure simultaneous availability and consistency of data in the face of network separation and other failures. The main task of the study was also to identify how each algorithm affects the performance and efficiency of working with distributed text documents. The results demonstrate how each algorithm affects the performance and efficiency of working with distributed text documents. This information helps developers choose the best algorithm for specific tasks and use cases. In general, collaborative text editing algorithms are important components for building applications and systems that allow users to effectively collaborate on real-time text documents. Their research and improvement are of great importance for the further development of distributed systems and tools for collaborative text editing.

DISCUSSION

In their paper on the examination of various routing protocols, N. Alsulami *et al.* (2022) used the Logoot algorithm to develop algorithms for data replication and document synchronisation in distributed systems for collective editing. The analysis of the study revealed new opportunities and advantages of using the Logoot algorithm outside of its general application in collective editing. This may encourage other researchers to develop and expand this algorithm in other areas of information technology. The use of data replication algorithms in network protocols can help ensure data consistency and synchronisation between nodes, which is an important aspect in distributed systems. The similarity of the paper lies in the analysis of distributed data replication methods in the context of distributed systems. In addition, both studies address the problem of data synchronisation between replicas or nodes in a distributed system. However, the paper examines new features and benefits of using the Logoot algorithm beyond collective editing.

W. Cai *et al.* (2022) investigated accelerating consistency support algorithms based on a commutative replicated data type using multi-core processors, examined the problem using the example of the RGA algorithm and parallel to RGA. Experimental results showed that when processing millions of simultaneous updates, PRGA achieves a performance increase of about four times compared to RGA on a conventional quad-core processor. Paper analysis can make important contributions to the scientific and practical fields of distributed systems and processing simultaneous data updates. Determining the increase in performance when using the PRGA algorithm compared to RGA on a multi-core processor is an important indicator. This can help developers of distributed systems improve the performance and speed of processing simultaneous data updates in such systems. This study can serve as an example of the efficient use of multi-core processors to speed up data processing in distributed systems. It can be useful for other researchers and developers working on similar tasks. The results obtained can be important for understanding and developing the theoretical foundations of CRDT and computational theory. They can help extend and improve existing approaches to handling concurrent updates. Although both papers focus on similar aspects, they have different research objects: the first paper focuses more on CRDT algorithms for collaborative text editing, while the second paper focuses on speeding up algorithms on multi-core processors using commutative replicated data.

M. Nicolas *et al.* (2020) devoted their study to the issue of effective renaming in the context of CRDTs for sequences and considered the issue of effective

renaming of objects or text parts in CRDTs sequences based on the LSEQ algorithm. Renaming is an important operation that allows correctly handling insertion, deletion, and editing operations in such sequences. Researchers offer effective renaming algorithms to ensure the correctness and speed of performing operations. The analysis of the study will improve CRDTs algorithms for sequences. Effective renaming is an important operation that affects the performance and correctness of data insertion, deletion, and editing operations in replicated sequences. The results of the study can help developers of CRDT structures expand their capabilities and applications. Efficient renaming algorithms make CRDTs more suitable for a variety of use cases, including collective editing of text data. Both studies aim to improve CRDTs to ensure correct and efficient operation in distributed systems, and their results are essential for developers and researchers working on distributed systems and collective data editing.

I. David and E. Syriani (2022) dedicated their paper to the implementation of distributed, collaborative modelling at different levels (multi-level modelling) using CRDT, which means data types that provide conflict-safe operations for distributed systems, discussed how using CRDT helps resolve conflicts that arise when editing a document in real-time, ensuring consistency and integration of changes made by users at different levels of modelling. CRDTs performing operations on distributed data without the need for manual conflict resolution. Applying CRDTs to implement distributed collaborative modelling can be useful in many areas, including software development, systems analysis, and modelling, and in any field where it is necessary to work together on complex models and data in a distributed environment. The analysis of the study will help to improve methods and approaches for implementing distributed collaborative modelling at different levels. Using CRDT to work with data allows resolving conflicts and ensuring real-time data consistency. The results of the study can be useful in software development, systems analysis, and modelling, in any field where collaborative modelling and collaboration on data in a distributed environment is important. CRDTs can be used in collective editing, work on models, document management, and other areas. The examination of CRDTs and their application in the implementation of distributed collaborative modelling can make important contributions to replicated data theory and the field of distributed systems. Both papers address the importance of using CRDTs in distributed systems to resolve conflicts and ensure data consistency. However, the paper highlights the use of CRDTs to implement collaborative modelling at different levels.

F. Jacob *et al.* (2021) discussed the use of CRDTs in Byzantine systems that may face attacks and abuse. The Byzantine model allows considering cases when some nodes of the system may act maliciously or incorrectly (equivocation). Their examination aims to explore how CRDTs can detect and solve problems of equivocation occurrence in the Byzantine environment, ensuring correct and reliable data synchronisation in the face of attacks and abuses. The use of CRDTs in Byzantine systems can provide increased resistance to abuse and attacks. Performance analysis can reveal how CRDTs can detect and solve equalisation problems in such systems, ensuring correct data synchronisation even when attackers are active. The work can contribute to the development of methods and approaches for data protection and ensuring their conflict-proof replication in Byzantine systems. This can be useful for system developers operating in vulnerable or critical environments. The study by F. Jacob *et al.* (2021) may contribute to increased awareness of security in distributed systems and increased attention to security issues in the Byzantine environment. The similarity of the papers lies in the study of CRDT algorithms, and the fact that they indicate the practical importance of using CRDTs in real-world conditions. However, the study focuses on using CRDTs to improve the resilience and security of data in Byzantine systems that may face attacks and abuse.

Thus, examining CRDT algorithms requires a deep understanding of their theory and applications in various distributed system scenarios. Each of the CRDT algorithms presented in this paper has its own unique advantages and disadvantages, and the correct choice will depend on the specific requirements of the application. By carefully evaluating the trade-offs of various algorithms, developers can choose the right CRDT algorithm to maintain consistency and availability in their distributed systems.

CONCLUSIONS

Research in the field of collaborative text editing algorithms shows that CRDT structures are becoming increasingly popular in the context of building distributed systems. This approach allows creating systems that ensure simultaneous consistency and availability of data, even in the event of network splits or other failures. Collaborative text editing algorithms such as Logoot, RGA, LSEQ, and Astrong use CRDT structures to achieve this goal. The main task of these algorithms is to create a mechanism that allows users to work together on documents and edit them, regardless of their physical location and availability on the network. CRDT structures allow avoiding conflicts and automatically combining changes made by users in different parts of the system. This greatly simplifies working with collaborative text editing, making it more efficient and complete. Each of these algorithms has its advantages and disadvantages, and the correct choice will depend on the specific requirements of the application.

The main problems with using the above algorithms are the complexity of implementation, which may require a deep understanding of the algorithm itself and the specifics of working with distributed systems and networks. The implementation of the algorithm must be efficient and reliable to ensure the quality of service for users. In addition, the problem may be

managing conflicts that may occur when multiple users change simultaneously. This is important to ensure that the data is correct and consistent. Different algorithms have different conflict resolution strategies, and it is vital to determine which approach is best suited for a particular use. Notably, working with CRDT structures may require additional resources, especially in conditions of a large number of users or a large amount of data. This can affect system performance and scalability. The results showed that choosing Logoot and RGA is a good option for supporting collaborative text editing, and RGA has the advantage of better solving concurrent operations. LSEQ is a good choice for supporting scalable and efficient indexing of large data sets. Astrong is a good choice for efficient and scalable processing of large sequences. One of the main advantages of CRDT algorithms is that they can provide high availability and low latency without complex coordination or centralised management mechanisms. However, they have some limitations, especially regarding the types of operations they support and the effectiveness of those operations. Therefore, it is essential to carefully evaluate the trade-offs of different CRDT algorithms when choosing the right one for a particular use case. The improvement and development of new algorithms and methods for joint text editing based on CRDT structures remains a relevant area. Research also can be focused on solving conflict problems in CRDT structures and developing algorithms that can resolve these conflicts more efficiently and automatically synchronise data between different users. Another important area is the development of interfaces and tools that make it easier for users to edit texts together based on CRDT structures. This includes creating user-friendly editors that are intuitive and easy to use and developing mechanisms for syncing data between devices and platforms. In addition, analysing use cases and identifying optimal approaches to implementing CRDT-based collaborative text editing systems in various fields, such as education, business, research, and many others, is relevant.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Alsulami, N., Cherif, A., & Imine, A. (2022). Collaborative editing over opportunistic networks. *International Journal of Ad Hoc and Ubiquitous Computing*, 39(3), 141-156. doi: 10.1504/IJAHUC.2022.121121.

[2] Attiya, H., Burckhardt, S., Gotsman, A., Morrison, A., Yang, H., & Zawirski, M. (2016). Specification and complexity of collaborative text editing. In *Proceedings of the 2016 ACM symposium on principles of distributed computing* (pp. 259-268). New York: Association for Computing Machinery. doi: 10.1145/2933057.2933090.

[3] Brahneborg, D., Afzal, W., & Mubeen, S. (2022). Resilient conflict-free replicated data types without atomic broadcast. In *Proceedings of the 17th international conference on software technologies* (pp. 516-523). Lisbon: SciTePress. doi: 10.5220/0011314500003266.

[4] Cai, W., He, F., & Lv, X. (2022). Multi-core accelerated CRDT for large-scale and dynamic collaboration. *The Journal of Supercomputing*, 78, 10799-10828. doi: 10.1007/s11227-022-04308-7.

[5] David, I., & Syriani, E. (2022). Real-time collaborative multi-level modeling by conflict-free replicated data types. *Software and Systems Modeling*, 22, 1131-1150. doi: 10.1007/s10270-022-01054-5.

[6] GitHub. (2016). Retrieved from https://github.com/hugooliveirad/elm-logoot.

[7] GitHub. (2021). Retrieved from https://github.com/josephg/simple-crdt-text.

[8] Gomes, V.B., Kleppmann, M., Mulligan, D.P., & Beresford, A.R. (2017). Verifying strong eventual consistency in distributed systems. *Proceedings of the ACM on Programming Languages*, 1, article number 109. doi: 10.1145/3133933.

[9] Hu, F., & Trivedi, R.H. (2020). Mapping hotel brand positioning and competitive landscapes by text-mining user-generated content. *International Journal of Hospitality Management*, 84, article number 102317. doi: 10.1016/j.ijhm.2019.102317.

[10] Jacob, F., Bayreuther, S., & Hartenstein, H. (2021). On conflict-free replicated data types and equivocation in byzantine setups. *arXiv - CS - Distributed, Parallel, and Cluster Computing*, 1. doi: 10.48550/arXiv.2109.10554.

[11] Kyrychek, H.H., & Chubich, A.I. (2020). Kanban-method use for software development organization. *Scientific notes of Taurida National V.I. Vernadsky University. Series: Technical Sciences*, 31(6), 78-82. doi: 10.32838/TNU-2663-5941/2020.6-1/13.

[12] Kleppmann, M., Mulligan, D.P., Gomes, V.B.F., & Beresford, A.R. (2021). A highly-available move operation for replicated trees. *IEEE Transactions on Parallel and Distributed Systems*, 33(7), 1711-1724. doi: 10.1109/TPDS.2021.3118603.

[13] Kleppmann, M., Wiggins, A., van Hardenberg, P., & McGranaghan, M. (2019). Local-first software: You own your data, in spite of the cloud. In *Proceedings of the 2019 ACM sigplan international symposium on new ideas, new paradigms, and reflections on programming and software* (pp. 154-178). New York: Association for Computing Machinery. doi: 10.1145/3359591.3359737.

[14] Medvedovska, O., & Yatsenko, V. (2021). Cloud services for organization of joint work on documents in real time mode. *Bulletin of the Cherkasy Bohdan Khmelnytsky National University. Series "Pedagogical Sciences"*, 1, 112-121. doi: 10.31651/2524-2660-2021-1-112-121.

[15] Nédelec, B., Molli, P., Mostefaoui, A., & Desmontils, E. (2013). LSEQ: An adaptive structure for sequences in distributed collaborative editing. In *Proceedings of the 2013 ACM symposium on document engineering* (pp. 37-46). New York: Association for Computing Machinery. doi: 10.1145/2494266.2494278.
[16] Nicolas, M., Oster, G., & Perrin, O. (2020). Efficient renaming in sequence CRDTs. In *proceedings of the 7th workshop on principles and practice of consistency for distributed data* (pp. 1-8). New York: Association for Computing Machinery. doi: 10.1145/3380787.3393682.
[17] Karayel, E., & Gonzàlez, E. (2022). Strong eventual consistency of the collaborative editing framework WOOT. *Distributed Computing*, 35, 145-164. doi: 10.1007/s00446-021-00414-6.
[18] Roh, H.G., Jeon, M., Kim, J.S., & Lee, J. (2011). Replicated abstract data types: Building blocks for collaborative applications. *Journal of Parallel and Distributed Computing*, 71(3), 354-368. doi: 10.1016/j.jpdc.2010.12.006.
[19] Saquib, N., Krintz, C., & Wolski, R. (2022). Ordering operations for generic replicated data types using version trees. In *Proceedings of the 9th workshop on principles and practice of consistency for distributed data* (pp. 39-46). New York: Association for Computing Machinery. doi: 10.1145/3517209.3524038.
[20] Shchetynina, O., Kravchenko, N., Horbatiuk, L., Alieksieieva, H., & Mezhuyev, V. (2022). Trello as a tool for the development of lifelong learning skills of senior students. *Postmodern Openings*, 13(2), 143-167. doi: 10.18662/po/13.2/447.
[21] Weiss, S., Urso, P., & Molli, P. (2009). Logoot: A scalable optimistic replication algorithm for collaborative editing on P2P networks. In *29th IEEE international conference on distributed computing systems* (pp. 404-412). Montreal: IEEE. doi: 10.1109/ICDCS.2009.75.

Огляд алгоритмів спільного редагування тексту Conflict-free Replicated Data Types (CRDT)

Юрій Олександрович Рабешко
Аспірант
Національний університет водного господарства та природокористування
33000, вул. Соборна, 11, м. Рівне, Україна
https://orcid.org/0009-0002-0763-7138

Юрій Васильович Турбал
Доктор технічних наук, професор
Національний університет водного господарства та природокористування
33000, вул. Соборна, 11, м. Рівне, Україна
https://orcid.org/0000-0002-5727-5334

Анотація. Аналіз та вибір алгоритмів для спільного редагування тексту, особливо впровадження безконфліктних реплікованих типів даних, має вирішальне значення для розуміння того, як сучасні системи можуть досягти співпраці в режимі реального часу, забезпечуючи при цьому цілісність даних. Метою дослідження був огляд різних алгоритмів спільного редагування, а також проведення порівняльного аналізу для розуміння їхніх переваг, недоліків та сфер застосування. Серед використаних методів можна зазначити статистичні методи, методи аналізу алгоритмів та їх використання у реальних сценаріях. Результати дослідження показали, що використання алгоритмів спільного редагування тексту сприяє вирішенню важливих завдань і викликів у сучасному світі інформаційних технологій. Виявлено, що алгоритми спільного редагування сприяють полегшенню комунікації та обміну інформацією в реальному часі. Це особливо важливо в умовах віддаленої роботи та спілкування, які стали стандартом для багатьох організацій. Спільне редагування текстів широко використовується в різних сферах і галузях, де командна робота, співпраця в режимі реального часу та обмін документами є важливими, наприклад: дослідницька діяльність, наукові роботи, освіта, розробка програмного забезпечення, редагування книг та рукописів, юридична співпраця, складання контрактів, медичні звіти, тощо. Крім того, використання оптимізованих алгоритмів спільного редагування допомагає зменшити час, необхідний для обробки даних та створення текстових матеріалів. Алгоритми спільного редагування мають широкий спектр застосувань у наукових дослідженнях, бізнесі та освіті. Вони дозволяють командам та індивідуальним користувачам більше ефективно вирішувати завдання, та спільно працювати над проектами. Можливість використання отриманих результатів в практичній діяльності дозволить використовувати алгоритми спільного редагування тексту для подальших інновацій та розвитку інформаційних технологій, що дасть можливість спільно працювати та обмінюватися інформацією з усім світом в реальному часі

Ключові слова: розподілені системи; ключові операції; Logoot, Astrong; Logarithmic Sequence; Replicated Growable Array

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 004.45
DOI: 10.62660/2306-4412.4.2023.19-27

Comparative analysis of frameworks for mobile application development: Native, hybrid, or cross-platform solutions

Oleksii Zarichuk*
Computer and Information Systems Manager
LLC "Fides"
02000, 11a E. Sverstyuk Str., Kyiv, Ukraine
https://orcid.org/0009-0009-0771-8465

Abstract. In the modern digital world, mobile application development is a key area of information technology, and choosing the optimal approach to their development is crucial for effective market implementation. The purpose of this study is to conduct a comparative analysis of various frameworks for mobile application development: native, hybrid, and cross-platform solutions. To achieve this purpose, methods of analysis, synthesis, and comparison were used. Characteristics of different frameworks for mobile application development, including their performance, cost, and access to device capabilities, were analysed. The study disclosed that native frameworks are distinguished or their highest performance and the ability to provide a maximally native look and functionality of the application. However, this approach has limitations as it requires separate development for each platform, leading to increased time and resource costs. Hybrid solutions proved to be cost-effective, allowing the use of a single codebase for creating applications for different platforms. This simplifies the development and maintenance process. Nevertheless, hybrid applications may have limited performance due to the use of WebView for interface display and restricted access to device capabilities. Cross-platform frameworks, on the other hand, provide a balance between performance and resource efficiency. They allow using a single codebase for creating applications for multiple platforms and can achieve satisfactory performance. However, they may have limited access to certain device capabilities and application appearance. This study makes a new contribution to science by providing a detailed comparative analysis of different approaches to mobile application development and the frameworks used for their creation. The results obtained can be used to make informed decisions regarding the choice of a framework for mobile application development

Keywords: Java; software development; performance; native look; codebase

Article's History: Received: 29.08.2023; Revised: 03.11.2023; Accepted: 18.12.2023

INTRODUCTION

In the modern information society, mobile application development has become one of the most important and relevant fields of software engineering. With each passing year, the number of mobile devices increases, along with the demand for high-quality and effective applications. Choosing the right approach to mobile application development is a critically important task for developers and businesses. The issue under study involves determining which type of framework for mobile application development is the most effective and meets the needs of the modern software market. This issue becomes increasingly relevant due to the growing number of frameworks and approaches available to developers, expanding the spectrum of possibilities for mobile applications.

In several studies, various aspects of mobile application development have been analysed and compared, including the choice between native and hybrid approaches, defining best practises in cross-platform development, and exploring methods for choosing the

Suggested Citation:
Zarichuk, O. (2023). Comparative analysis of frameworks for mobile application development: Native, hybrid, or cross-platform solutions. *Bulletin of Cherkasy State Technological University,* 28(4), 19-27. doi: 10.62660/2306-4412.4.2023.19-27.

*Corresponding author

optimal platform for cross-platform development. In their study, H.O. Kozub and Yu.H. Kozub (2022) explored the development of cross-platform mobile applications using Kotlin Multiplatform and Jetpack Compose for different operating systems, including Android, Windows, Linux, and macOS. They express the importance of these tools and the potential for reducing development time and preventing errors through their use. In addition, the authors emphasise the necessity of employing a declarative approach for creating user interfaces.

O. Karatanov *et al.* (2021) conducted a comparative analysis of two important frameworks for modular testing in the Java programming language – JUnit and TestNG. They identified the main functions and advantages of both frameworks, noting that TestNG has more extensive functionality, making it more flexible for complex projects. In the study by M. Singh and G. Shobha (2021), the relevance of cross-platform mobile application development was highlighted. Various frameworks for this purpose were discussed, and a comparative analysis of their functionality was conducted. The main conclusions of the study emphasise the need for a well-founded framework choice based on the specific project requirements, and there is no universal framework for all situations.

In the paper by T. Zohud and S. Zein (2021), the authors investigate the approaches of development teams to cross-platform mobile application development. They use qualitative research, including the case study method, interviews, and group discussions, to gather information from four software development companies in Palestine. The results of the study show that developer experience is a key factor in the development process. The React Native framework is recognised as promising and dominant. A. Kaczmarczyk *et al.* (2022) considered a comparison between native and hybrid mobile applications for the Android operating system, with a focus on using BLE (Bluetooth Low Energy) and Wi-Fi (Wireless Fidelity). It was noted that mobile applications are becoming increasingly popular in the era of smartphones and tablets. The authors conducted a comparative analysis aimed at determining the efficiency of data processing in both technologies. The study by P. Lachgar *et al.* (2022) solves the problem of developing cross-platform mobile applications due to the growing popularity of mobile devices. The authors propose a new framework for choosing the optimal platform for cross-platform development, using multi-criteria decision-making methods such as the Analytic Hierarchy Process (AHP) and the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS).

After analysing studies in the field of mobile application development, the need for research comparing different frameworks, including native, hybrid, and cross-platform solutions, was identified. The purpose of the study was to analyse approaches to mobile application development and compare them, considering their limitations and advantages. The most popular frameworks and tools for each approach were also considered to provide objective information that would help developers and organisations make informed choices in mobile application development.

MATERIALS AND METHODS

This study conducted a comparative analysis of frameworks for developing mobile applications using native, hybrid, and cross-platform solutions. To achieve the purpose, the following methods were used: analysis, synthesis, and comparison. The use of these methods contributed to clarifying the advantages and limitations of each of the examined approaches in mobile application development.

As part of the study, two main mobile platforms were identified for analysis: Android and iOS, which are leaders in the modern mobile device market. Three main approaches to mobile application development were considered during the analysis: native, hybrid, and cross-platform. Specialised integrated development environments (IDEs) were used for native application investigation, such as Android Studio for Android and Xcode for iOS. For the study of hybrid applications, frameworks based on web technologies were used, including Apache Cordova, Ionic, and React Native. Cross-platform application frameworks such as Flutter, Xamarin, and Appcelerator Titanium were reviewed. One of the main criteria for selecting frameworks was their popularity and relevance in the mobile application development market.

The study was based on an analysis of open scientific literature that was available on the Internet. Various works, studies, and publications that detailed different approaches and frameworks for mobile application development were used (Biørn-Hansen *et al.*, 2020; Zohud & Zein, 2021; Lachgar *et al.*, 2022). This literature provided the foundation for the analysis and comparison of the selected approaches. The analysis focused on several key aspects, including development productivity, the quality and performance of created applications, development cost, support for different mobile platforms, extensibility, and other important factors. For each of the examined approaches (native, hybrid, and cross-platform solutions), their advantages and limitations were identified and analysed, contributing to the creation of an objective comparative analysis. The synthesis method involved gathering information about each of the reviewed frameworks, including programming languages used, development tools availability, platform support, application quality, extensibility, and other characteristics. This method provided a comprehensive understanding of each approach.

Comparison was conducted by juxtaposing the obtained results to determine the advantages and disadvantages of each approach. Factors considered in the comparative analysis included development productivity, development cost, application quality and performance, support for different mobile platforms, memory

consumption, extensibility, and other parameters. Based on this comparative analysis, the strengths and weaknesses of each of the examined approaches to mobile application development were identified. The choice of analysis, synthesis, and comparison methods for this research is justified by their ability to provide a comprehensive and objective analysis of different approaches to mobile application development. The analysis method allowed for a deeper examination of each approach, identifying its features, technical parameters, and other characteristics. Synthesis combined the gathered information to create a comprehensive understanding of each approach, which was useful for further comparison. Comparison identified the advantages and disadvantages of different approaches, considering various aspects such as productivity, development cost, application quality and performance, support for different mobile platforms, and other factors.

RESULTS

Native frameworks for mobile development. Native frameworks for mobile platforms are specially designed tools that allow developers to create applications optimised for a specific platform or operating system. These frameworks provide access to native capabilities and features of the platform, resulting in the development of high-performance and efficient software products. The key feature of these frameworks is their focus on development for a specific platform, such as iOS or Android. This means that developers have the opportunity to fully leverage all the capabilities of the given platform to create applications that work optimally and reliably. Typically, for iOS, this involves the Swift programming language and the Xcode development tool, while for Android, it includes the Kotlin or Java programming language and the Android Studio development tool.

Xcode is an integrated development environment (IDE) designed for creating software for iOS and macOS platforms. It is specifically tailored for programming languages recommended by Apple, including C, C++, Objective-C, Swift, Java, AppleScript, Python, and Ruby. Xcode offers advanced capabilities and tools that facilitate the development process, including code refactoring features (Tkachuk & Bulakh, 2022). The modern Swift programming language was developed by Apple for iOS, macOS, watchOS, and tvOS platforms. It aims to provide safety, performance, and code expressiveness, offering a more convenient and efficient alternative to the Objective-C language (Fojtik, 2019). Swift comes integrated with the Xcode development environment in the macOS operating system. This integration allows developers to develop, test, and debug their Swift programmes in a unified environment. Xcode provides a range of tools and features that ease the development process, such as a visual interface builder, debugger, and simulator for testing applications on various devices (Ziyodullayevich *et al.,* 2019).

The advantage of using Xcode and Swift to develop iOS applications is the ability to use native features and functions of Apple devices. Native development allows creating applications optimised for specific iOS hardware and software, resulting in improved performance and user experience. However, it requires developers to have knowledge of different programming languages and development environments (Biørn-Hansen *et al.,* 2020). The primary programming languages for Android application development include Java and Kotlin. Java is a traditional programming language for Android, while Kotlin is a modern alternative that provides efficiency and convenience for developers. Android Studio serves as an integrated development environment (IDE) specifically designed for creating Android applications. It supports both major programming languages, Java and Kotlin, and provides developers with the necessary tools for application development and debugging. In addition, for scenarios requiring maximum performance, the Native Development Kit (NDK) is available, allowing developers to use native code, such as C and C++, in their applications. The Android Software Development Kit (SDK) offers tools and resources for application development, testing, and documentation for Android. It also includes emulators for testing applications on various devices and Android versions (Sun *et al.,* 2021). All these components of the Android application development infrastructure enable developers to create efficient and reliable applications for different devices, working at an optimal level of performance and utilising the powerful capabilities of the Android platform.

The development of native mobile applications has several key features that distinguish it from other development approaches. One of the main advantages is the ability to harness the full potential of the underlying operating system and device hardware. Native applications have direct access to device-specific features and APIs (application programming interfaces), enabling developers to create optimised and productive applications. This level of control allows the development of multifunctional and engaging user interfaces with smooth animations, fast responsiveness, and access to advanced functionalities (Dittrich *et al.,* 2023). Another key feature of native application development is the ability to follow specific design recommendations for user interface components. Native applications can provide a consistent user experience by adhering to design principles and templates of the target operating system (Thamutharam *et al.,* 2021). This ensures that the application looks and feels like it is native to the platform, improving user experience and satisfaction.

Native application development also offers better integration with the device ecosystem. Developers can easily access device capabilities such as the camera, GPS, accelerometer, and push notifications (Raeesi *et al.,* 2022). This enables the creation of applications that can fully utilise the device's potential, providing features like location-based services, augmented reality, and

real-time functionality. In addition, compared to other development approaches, native applications typically offer better performance. Since they are built using native programming languages and tools, they can operate more efficiently. However, native application development also has limitations and challenges. One of the main drawbacks is the need to develop separate codebases for each target platform (Masaad Alsaid *et al.,* 2021). This can increase development time and costs. Moreover, updates and bug fixes may require separate deployments for each platform, leading to increased maintenance efforts.

Hybrid mobile app development frameworks. Hybrid development of mobile applications allows developers to create mobile applications using web development technologies, such as HTML (Hybrid Mobile App Development Frameworks), CSS (Cascading Style Sheets), and JavaScript. This method combines the advantages of other approaches and offers a compromise between native applications and web applications, allowing the production of applications that simultaneously provide high functionality for a specific platform and compatibility with different platforms (Wu *et al.,* 2022).

One of the most popular tools for hybrid development is Apache Cordova. It initialises a native application using WebView, an embedded web browser. It acts as a bridge between native code and application web components, enabling developers to write business logic in JavaScript and create user interfaces using HTML and CSS. This approach allows the development of applications that can work on various platforms, including Android and iOS, using a single codebase. Hybrid application development frameworks provide a wide range of features and capabilities to simplify the development process. These frameworks often include components and libraries that allow developers to create visually appealing and fast user interfaces (Singh & Shobha, 2021). They also provide access to device features and APIs through plugins, allowing developers to use native functionality in their hybrid applications.

In addition to Apache Cordova, other popular hybrid frameworks include Ionic (which uses Angular) and React Native (based on React). Each of these frameworks has its unique features and applications (Table 1), allowing developers to choose the one that best suits their needs.

Table 1. Comparative characteristics of popular hybrid frameworks

Framework	Description	Programming language	Main features
Apache Cordova	Uses web technologies to develop mobile applications.	HTML, CSS, JavaScript	Supports many plugins for device functionality.
Ionic	A hybrid framework based on Angular. Designed specifically for creating beautiful and functional mobile applications	HTML, CSS, JavaScript (with Angular)	Pre-built components and tools for the user interface.
React Native	Allows using React to create mobile applications with the appearance and functionality of native applications.	JavaScript (with React)	Ability to reuse code between platforms and high performance.

Source: compiled by the author

One of the main advantages of hybrid development is the ability to leverage existing skills and resources of web developers. Developers proficient in web technologies can easily transition to hybrid development as they can use their knowledge of HTML, CSS, and JavaScript. This can lead to shortened development timelines and reduced costs compared to developing separate native applications for each platform. However, hybrid development also has its limitations. Since hybrid applications rely on WebView for displaying the user interface, they may not achieve the same level of performance as native applications (Hu *et al.,* 2023). In addition, using plugins may be necessary to access certain device features and APIs, introducing additional complexity and potential compatibility issues. Overall, hybrid mobile application development offers a compromise between native and web applications. It allows developers to create applications for different platforms using web technologies while providing access to native device features. Apache Cordova is a popular tool for hybrid development, enabling the use of HTML, CSS, and JavaScript to create applications that can work on various platforms. Although hybrid applications may not have the same performance as native applications, they offer advantages in terms of development efficiency and cost-effectiveness.

Efficient Cross-Platform Software Development Solutions. Cross-platform mobile application development is a modern approach in the field of mobile device software. It involves creating applications that can run on different platforms, such as Android and iOS, using a unified codebase and other shared resources. Cross-platform development has gained popularity due to its ability to effectively utilise shared code for creating applications on different platforms. This allows developers to save time and resources by avoiding the need to maintain separate code for each platform.

The main advantage of cross-platform development is the reduction of costs for application development and maintenance since shared code can be used across all platforms. In addition, developers can use a single programming language to create application

functionality. Cross-platform frameworks like Flutter, Xamarin, and others (Table 2) provide tools for developing mobile applications with appearances and behaviours similar to native applications on each platform (Martínez, 2019). This ensures high quality and consistent functionality on different operating systems.

Table 2. Overview of popular cross-platform frameworks

Framework	Programming language	Description
Flutter	Dart	A framework from Google for creating beautiful mobile applications with a single code base and using its own rendering engine.
Xamarin	C#	A framework from Microsoft that allows developers to use the C# programming language to create mobile applications for Android and iOS.
Appcelerator Titanium	JavaScript	A framework that uses JavaScript to develop cross-platform mobile applications and provides access to native features.

Source: compiled by the author

Cross-platform frameworks for mobile application development, while offering advantages such as a unified codebase and reduced development costs, also come with a set of drawbacks. Firstly, one of the main drawbacks is the limited access to native functions and platform APIs (Application Programming Interface). Cross-platform frameworks attempt to abstract the device's functionality, but sometimes they may not fully replicate all the capabilities available on a specific platform. The second disadvantage is performance and speed. Mobile applications developed using cross-platform frameworks may run slower and require more resources compared to native applications. In addition, delays in updates may lead to difficulties in utilising new features. Finally, cross-platform frameworks may not be as well-adapted to the specific look and behaviour of certain platforms, which may necessitate additional customisation and adaptation. Therefore, cross-platform frameworks, while allowing for time and resource savings in mobile application development, have limitations and drawbacks that need to be considered when choosing a development approach.

Comparison of the Effectiveness of Native, Hybrid, and Cross-Platform Solutions. In the modern context of mobile application development, choosing the optimal approach is crucial. Developers face different capabilities and limitations when choosing between native, hybrid, and cross-platform solutions. Table 3 provides a comparative analysis of these development approaches.

Table 3. Comparative analysis of native, hybrid, and cross-platform mobile application development solutions

Parameter	Native applications	Hybrid applications	Cross-platform applications
Performance and speed	High performance, optimised speed using native components and programming languages (Java/Kotlin for Android, Swift/Objective-C for iOS).	Moderate performance and speed due to using WebView for UI display and some abstraction.	Moderate performance but can be improved with specialised cross-platform frameworks (e.g., Flutter).
Cost and development time	High costs and longer development time due to the need to create separate code for each platform.	Lower costs and faster development by using a single codebase for both platforms. Some applications may require adjustments for a specific platform.	Lower costs and less time, but development may take slightly longer compared to hybrid solutions due to choosing the right cross-platform framework and team training.
Access to device capabilities	Full access to all device capabilities and native APIs.	Limited access, requires the use of plugins to access specific device functions.	Average access; cross-platform frameworks provide APIs for simplified access to device capabilities but may not always have a full range of features.
Visual appearance	Native appearance on each platform, ensuring high-quality and polished design.	Less native appearance; the interface may be less attractive and less aligned with the design standards of each platform.	Typically less native appearance, but the ability to use specialised frameworks to enhance design.
Platform support	Platform-specific (Android and iOS).	Both platforms (Android and iOS) from a single codebase.	Both platforms (Android and iOS) from a single codebase.
Updates and support	Separate for each platform, separate updates and maintenance.	Updates and maintenance from a single codebase, but there may be a need for updates for each OS (Operating System) separately to support new platform features.	Updates and maintenance from a single codebase, the possibility of additional work to support new features.

Source: compiled by the author

Native applications are characterised by high performance and speed due to the use of native components and programming languages for each platform separately. For developing native applications on the Android platform, programming languages such as Java or Kotlin are used, while on the iOS platform, Swift or Objective-C are utilised. This approach provides full access to all device capabilities and native APIs, allowing the creation of applications with excellent performance and speed. However, it requires more resources and development time due to the need for separate code for each platform, leading to increased costs and project duration. On the other hand, hybrid applications allow cost and development time reduction by using a single codebase for both platforms. They often rely on web technologies such as HTML, CSS, and JavaScript, simplifying development for web-experienced developers. One popular platform for hybrid development is Apache Cordova, which initialises a native application using WebView, an embedded web browser. Hybrid applications may exhibit lower performance due to the use of WebView and limited access to device capabilities. Plugins can be used to access specific device functions, such as the camera or geolocation.

Cross-platform applications combine the advantages of both previous approaches, enabling the use of a single codebase for both platforms. With the right choice of a cross-platform framework, satisfactory performance can be achieved. One such framework is Flutter, developed by Google, which uses the Dart programming language and its own engine for UI rendering. Another popular option is Xamarin, a framework from Microsoft that allows the use of the C# programming language. Cross-platform applications may also use specialised frameworks to enhance the design and appearance of applications. However, they may look less native and have limited access to platform-specific features. The choice of a development approach should be well-founded and depends on the specific project needs, budget constraints, resources, and the importance of performance, speed, and native application appearance.

DISCUSSION

As a result of the study of three main approaches to mobile application development, namely native, hybrid, and cross-platform solutions, a detailed analysis of each of these approaches was conducted, considering the use of different frameworks. The overall analysis of the findings indicates that the choice of a specific approach to mobile application development should be based on the specific requirements of the project, budget constraints, and resources, as well as the importance of performance, speed, and the native look of the application. Adequate evaluation of these factors before choosing an approach is a crucial stage in mobile application development.

The study by S.R. Uplenchwar *et al.* (2022) focuses on exploring Flutter as a cross-platform framework for mobile app development, the Dart programming language, and Firebase technology. The author notes that the development of wireless technologies and mobile devices has a significant impact on everyday life. Consequently, many aspects of life become digital, and to reduce manual work, more tasks are performed using mobile applications. Flutter, as mentioned by the author, is a popular User Interface (UI) framework for mobile application development from Google. It provides a set of user interface elements such as sliders, buttons, and text fields. Developers building mobile applications using Flutter use the Dart programming language. The author also emphasises Firebase technology, which provides tools for tracking analytics, app crash reports, and conducting marketing experiments. This article explores the possibility of using Flutter, Dart, and Firebase for mobile application development to reduce time and resource costs and to ensure consistent functionality on both Android and iOS platforms. The general conclusion from the study by S.R. Uplenchwar *et al.* (2022), which can be agreed upon, is that Flutter, Dart, and Firebase are powerful tools for mobile application development. Flutter allows the creation of beautiful and functional applications with a single codebase for both platforms. Dart is a programming language specifically designed for working with mobile applications and has its package manager for convenient dependency management. Firebase provides extensive capabilities for analysing and enhancing applications. Together, these technologies can significantly simplify and improve the process of mobile application development.

A study by N. Varghese and N. Medina-Medina (2021) aimed to develop a new methodology for mobile application development, referred to as "Agile Beeswax." The authors argue for the necessity of such a methodology due to the specificities of mobile application development and the requirements for rapid response to changes in this field. The key elements of the Agile Beeswax methodology include an incremental and iterative approach to development, consisting of two main iteration cycles (sprints): the incremental design cycle and the incremental development cycle. These two cycles are connected by a bridge. The Agile Beeswax methodology is divided into six phases, including strategy and idea, user experience design, user interface design, design to development, handover and technical decisions, development and deployment, and monitoring. The authors indicate that one of the main advantages of their methodology is that it is designed to encompass both academic and business-oriented perspectives, aiming to unite these two communities. The findings show that the Agile Beeswax methodology is focused on addressing specific problems and challenges that arise in the process of mobile application development. It combines Agile and Scrum practises, engineering technical practises,

and operational practises to achieve greater efficiency in this field. The authors emphasise the importance of responding quickly to changes in the market and acknowledge that Agile methodologies are the best approach for mobile application development, as they allow for faster changes and improved product quality. It can be agreed that the Agile Beeswax methodology represents an interesting approach to mobile application development that can be beneficial for both the academic and business-oriented communities, addressing specific problems associated with this type of development.

As a result of the study, it was established that each mobile application development method has its advantages and limitations. Native applications provide high performance and a native look but require more resources and development time. Hybrid applications save time and costs but may have limited access to device functions. Cross-platform applications combine the benefits of both approaches, but the choice of the right framework is crucial. The study by S.R. Uplenchwar *et al.* (2022) underscores the importance of selecting the right cross-platform framework, such as Flutter, for creating applications with a single codebase. Dart and Firebase also play a significant role in easing the development process. The study by H.A. Alrabaiah and N. Medina-Medina (2021) proposes the Agile Beeswax methodology for mobile application development, uniting the best practises of Agile and Scrum. This methodology is aimed at improving mobile application development and can be beneficial for both academic and business-oriented communities. The choice of approach and tools for mobile application development should be based on the specific needs of the project, limitations, and important aspects of productivity and application appearance. The discussed methodologies and frameworks can be useful in achieving the goals of mobile application development. In particular, the choice between native, hybrid, and cross-platform applications should be determined by considering the project's needs for performance, access to device capabilities, and constraints on budget and resources. Research shows that for better mobile application development, it is important to consider the individual requirements and specificity of the project, as well as use appropriate methodologies and tools.

CONCLUSIONS

In the study, various approaches to the development of mobile applications were considered, such as native, hybrid, and cross-platform solutions. Native applications, developed for a specific platform, provide the highest performance and access to device capabilities but require considerable effort and resources. Hybrid applications reduce costs and time by using a shared codebase but may have limited access to device functions. Cross-platform applications, combining the advantages of both approaches, can be effective with the right choice of a framework. Different frameworks for mobile application development were analysed in the study, including Flutter, Xamarin, Apache Cordova, React Native, and Ionic. Each has its own features and advantages. Flutter, developed by Google, allows creating high-quality interface and performance applications with a single codebase. Xamarin by Microsoft uses the C# programming language and integrates with the Microsoft ecosystem. Apache Cordova is based on web technologies and initialises a native application through WebView. React Native from Facebook enables creating applications with a native look and fast coding using React and JavaScript. Ionic uses HTML, CSS, and JavaScript for hybrid development with the option to use Angular. The results of the study show that the choice of a framework should be based on the specific project requirements, considering performance, budget, and resources. Each framework has its advantages and limitations, and it is essential to consider them in the context of a particular project.

Recommendations include selecting the development approach for a mobile application based on specific project requirements, financial constraints, and resources. It is crucial to evaluate the performance, speed, and native appearance of the application correctly to achieve successful results in mobile application development. The obtained results can be useful for mobile application developers and organisations planning to create mobile applications, helping them choose the optimal approach and frameworks for development. Further research can focus on a detailed analysis and comparison of different cross-platform frameworks for mobile application development, exploring the impact of the development approach choice on the quality and performance of applications, and developing new methodologies and tools to optimise the mobile application development process, considering current technological and business needs.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Alrabaiah, H.A., & Medina-Medina, N. (2021). Agile beeswax: Mobile app development process and empirical study in real environment. *Sustainability*, 13(4), article number 1909. doi: 10.3390/su13041909.

[2] Biørn-Hansen, A., Rieger, C., Grønli, T.M., Majchrzak, T.A., & Ghinea, G. (2020). An empirical investigation of performance overhead in cross-platform mobile development frameworks. *Empirical Software Engineering*, 25, 2997-3040. doi: 10.1007/s10664-020-09827-6.

[3] Dittrich, F., Albrecht, U.V., Scherer, J., Becker, S.L., Landgraeber, S., Back, D.A., Fessmann, K., ... Klietz, M.L. (2023). Development of open backend structures for health care professionals to improve participation in app developments: Pilot usability study of a medical app. *JMIR Formative Research*, 7, article number e42224. doi: 10.2196/42224.
[4] Fojtik, R. (2019). Swift a new programming language for development and education. In T. Antipova & Á. Rocha (Eds.), *Digital Science 2019* (vol. 1114; pp. 284-295). Cham: Springer. doi: 10.1007/978-3-030-37737-3_26.
[5] Hu, J., Wei, L., Liu, Y., & Cheung, S.C. (2023). ωTest: Webview-oriented testing for android applications. In *ISSTA 2023: Proceedings of the 32nd ACM SIGSOFT international symposium on software testing and analysis* (pp. 992-1004). New York: Association for Computing Machinery. doi: 10.1145/3597926.3598112.
[6] Kaczmarczyk, A., Zając, P., & Zabierowski, W. (2022). Performance comparison of native and hybrid android mobile applications based on sensor data-driven applications based on Bluetooth low energy (BLE) and Wi-Fi communication architecture. *Energies*, 15(13), article number 4574. doi: 10.3390/en15134574.
[7] Karatanov, O., Yena, M., Bova, Y., & Ustymenko, O. (2021). Comparison of popular test frameworks JUnit and TestNG. *Young Scientist*, 5(93), 164-170. doi: 10.32839/2304-5809/2021-5-93-31.
[8] Kozub, H., & Kozub, Yu. (2022). Declarative method for creating multiplatform applications. *Bulletin of the Eastern Ukrainian National University named after Volodymyr Dahl*, 5(275), 10-15. doi: 10.33216/1998-7927-2022-275-5-10-15.
[9] Lachgar, M., Hanine, M., Benouda, H., & Ommane, Y. (2022). Decision framework for cross-platform mobile development frameworks using an integrated multi-criteria decision-making methodology. *International Journal of Mobile Computing and Multimedia Communications*, 13(1), 1-22. doi: 10.4018/ijmcmc.297928.
[10] Martínez, M. (2019). Two datasets of questions and answers for studying the development of cross-platform mobile applications using Xamarin framework. In *2019 IEEE/ACM 6th international conference on mobile software engineering and systems (MOBILESoft)* (pp. 162-173). Piscataway: Institute of Electrical and Electronics Engineers. doi: 10.1109/mobilesoft.2019.00032.
[11] Masaad Alsaid, M.A.M., Ahmed, T.M., Sadeeq, J., Khan, F.Q., Mohammad, & Khattak, A.U. (2021). A comparative analysis of mobile application development approaches: Mobile application development approaches. *Proceedings of the Pakistan Academy of Sciences: A. Physical and Computational Sciences*, 58(1), 35-45. doi: 10.53560/PPASA(58-1)717.
[12] Raeesi, A., Khajouei, R., & Ahmadian, L. (2022). Evaluating and rating HIV/AIDS mobile apps using the feature-based application rating method and mobile app rating scale. *BMC Medical Informatics and Decision Making*, 22, article number 281. doi: 10.1186/s12911-022-02029-8.
[13] Singh, M., & Shobha, G. (2021). Comparative analysis of hybrid mobile app development frameworks. *International Journal of Soft Computing and Engineering*, 10(6), 21-26. doi: 10.35940/ijsce.f3518.0710621.
[14] Sun, C., Ma, Y., Zeng, D., Tan, G., Ma, S., & Wu, Y. (2021). μDep: Mutation-based dependency generation for precise taint analysis on android native code. In *IEEE transactions on dependable and secure computing*, 20(2), 1461-1475. Piscataway: Institute of Electrical and Electronics Engineers. doi: 10.1109/TDSC.2022.3155693.
[15] Thamutharam, Y.N., Mustafa, M.B.P., Musthafa, F.N., & Tajudeen, F.P. (2021). Usability features to improve mobile apps acceptance among the senior citizens in Malaysia. *ASM Science Journal*, 16. doi: 10.32802/asmscj.2021.686.
[16] Tkachuk, A., & Bulakh, B. (2022). Research of possibilities of default refactoring actions in swift language. *Technology Audit and Production Reserves*, 5(2(67)), 6-10. doi: 10.15587/2706-5448.2022.266061.
[17] Uplenchwar, S.R., Denge, U.S., Bajoriya, A.S., & Bachwani, S.A. (2022). Review on detail information about flutter cross platform. *International Journal for Research in Applied Science and Engineering Technology*, 10(1), 1016-1022. doi: 10.22214/ijraset.2022.39977.
[18] Wu, C., Pérez-Álvarez, J.M., Mos, A., & Carroll, J.M. (2022). Codeless app development: Evaluating a cloud-native domain-specific functions approach. In *Proceedings of the 56th annual Hawaii international conference on system sciences, HICSS 2023* (pp. 6904-6913). Washington: IEEE Computer Society. doi: 10.48550/arxiv.2210.01647.
[19] Ziyodullayevich, A.Q., Babakulovich, Z.R., Bakhodirovna, M.Z., Bekmurodovich, S.A., & Akif-ogli, M.R. (2019). Efficient and convenient application to determine the functions and analysis of the reliability of the device. *International Journal of Innovative Technology and Exploring Engineering*, 9(2), 1804-1809. doi: 10.35940/ijitee.b7323.129219.
[20] Zohud, T., & Zein, S. (2021). Cross-platform mobile app development in industry: A multiple case-study. *International Journal of Computing*, 20(1), 46-54. doi: 10.47839/ijc.20.1.2091.

Порівняльний аналіз фреймворків для розробки мобільних програм: рідні, гібридні чи крос-платформні рішення

Олексій Геннадійович Зарічук
Менеджер комп'ютерних та інформаційних систем
ТОВ «Фідес»
02000, вул. Є. Сверстюка, 11а, м. Київ, Україна
https://orcid.org/0009-0009-0771-8465

Анотація. У сучасному цифровому світі розробка мобільних додатків є ключовою галуззю інформаційних технологій, а вибір оптимального підходу до їх розробки має вирішальне значення для ефективного впровадження на ринку. Метою цієї наукової роботи було проведення порівняльного аналізу різних фреймворків для розробки мобільних додатків: рідних, гібридних і крос-платформних рішень. Для досягнення поставленої мети були використані методи аналізу, синтезу та порівняння. Були проаналізовані характеристики різних фреймворків для розробки мобільних додатків, включаючи їх продуктивність, вартість та доступ до пристроєвих можливостей. Під час проведення дослідження встановлено, що рідні фреймворки вирізняються найвищою продуктивністю та можливістю забезпечити максимально нативний вигляд та функціональність додатку. Однак цей підхід має свої обмеження, оскільки вимагає окремої розробки для кожної платформи, що призводить до збільшення витрат часу та ресурсів. Гібридні рішення виявилися економічно вигідними, оскільки вони дозволяють використовувати єдину кодову базу для створення додатків для різних платформ. Це спрощує процес розробки та підтримки. Однак гібридні додатки можуть мати обмежену продуктивність через використання WebView для відображення інтерфейсу та обмежений доступ до пристроєвих можливостей. Крос-платформні фреймворки, у свою чергу, забезпечують баланс між продуктивністю та ефективністю витрат ресурсів. Вони дозволяють використовувати одну кодову базу для створення додатків для кількох платформ і при цьому можуть досягати задовільної продуктивності. Однак вони можуть мати обмежений доступ до деяких пристроєвих можливостей та вигляду додатку. Це дослідження робить новий внесок у науку шляхом детального порівняльного аналізу різних підходів до розробки мобільних додатків і фреймворків, які використовуються для їх створення. Отримані результати можна використовувати для прийняття інформованих рішень щодо вибору фреймворку для розробки мобільного додатку

Ключові слова: Java; розробка програмного забезпечення; продуктивність; нативний вигляд; кодова база

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 628.164-926.41
DOI: 10.62660/2306-4412.4.2023.28-35

Study of possibilities of cleaning of mechanical filters of baromembrane plants

Margarita Karpenko*
Postgraduate Student
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
03056, 37 Beresteyskyi Ave., Kyiv, Ukraine
https://orcid.org/0000-0003-3237-4130

Abstract. The appearance of contamination on the membrane and filters, which significantly reduces the efficiency of operation (separation efficiency, water permeate flow, salt rejection) and the service life of a reverse osmosis plant, is one of the most urgent problems that arise during water treatment using this plant. Therefore, the purpose of the work is to find optimal methods for cleaning and regeneration of mechanical filters of reverse osmosis plants. The solution to this problem is carried out by means of an empirical analysis of available scientific information and conducting laboratory studies on cartridge cleaning and infiltrate processing. A cycle of experiments on cleaning of used mechanical filters of reverse osmosis in a specially assembled plant with a gradual increase in the concentration of sulphuric acid has been conducted. According to the observations during laboratory research, effective removal of dissolved iron from the solution used to wash contaminated polystyrene filters begins at pH = 4 and up to pH = 10, but at pH = 4 the settling and filtering take 24 hours, and at pH = 10 this process takes no more than an hour. As a result of further research, the most effective hydrogen indicator for settling and almost complete removal of iron from the solution is 4.5. Further increase of the hydrogen indicator to 10 is ineffective. In addition to the chemical method of neutralising the mother solution, the use of electrodialysis is also advisable, while the use of electrolyser with a lead anode would be the best option. Thus, after cleaning of one mechanical filter, almost 80 g of pure gypsum is obtained. This gypsum can serve as a high-quality cement additive or be used for the production of internal cladding plates. Clean water obtained during the process can be used for subsequent cycles of other mechanical filters cleaning

Keywords: reverse osmosis; cartridges; polystyrene filters; regeneration; carbon filters; concentrate

Article's History: Received: 06.09.2023; Revised: 11.11.2023; Accepted: 18.12.2023.

INTRODUCTION

Reverse osmosis (RO) as a method of purification of water of any content to drinking standards is a popular and widespread technology in the world. The possibility of obtaining clean water easily and without unnecessary costs eliminates environmental risks and threats that these plants bring to the environment. Manufacturers of reverse osmosis plants and variable components for them (polystyrene filters, carbon cartridges, membranes, mineralisers and ultraviolet disinfectants) are not interested in investigating the damage that their business is doing to the environment. Scientists from all over the world are focused on the study of optimisation of osmotic and energy processes at plants, modification of membranes, the development of the latest antiscalants and aggressive chemical membrane cleaners that extend their service life. However, the huge complex of damage caused by uncontrolled and growing use of such plants in everyday life, despite their benefits for humans, is not taken into account. In addition to the fact that in the

Suggested Citation:
Karpenko, M. (2023). Study of possibilities of cleaning of mechanical filters of baromembrane plants. *Bulletin of Cherkasy State Technological University,* 28(4), 28-35. doi: 10.62660/2306-4412.4.2023.28-35.

*Corresponding author

process of water purification, 1/3 of the inlet water for purification is discharged into the sewer, hundreds of thousands of used polypropylene filters, mineralisers, plastic cartridges with activated carbon, ultraviolet disinfectants and the membranes themselves, contaminated with products of cleaning, are thrown into the environment every year without processing. There is no statistical information on the damage caused, just as there are no environmentally safe methods to regenerate the components of filters and blocks. Therefore, this issue is relevant and still not resolved.

The use of baromembrane technologies for water purification is one of the most effective and widespread technologies in the world, according to an analysis by X. Zhang and Y. Liu (2021). According to D.M. Warsinger *et al.* (2018) and M. Qasim *et al.* (2019), the appearance of membrane and filter contamination, which limits the efficiency of operation (separation efficiency, water permeate flow, salt rejection) and the lifetime of the reverse osmosis plant is one of the most serious problems that arise in RO water treatment. If the membrane can be used for 1-2 years, then mechanical filters must be replaced every three months. Untimely replacement of filters can lead to scale formation on the membrane, and the failure of the entire system, according to T.M. Joseph *et al.* (2023). Contamination and scale formation reduce the hydrophobicity of the membrane and deform the pores, which leads to wetting of the membrane – a critical drawback in the plant, when the nutrient solution flows through the pores in the liquid phase, which threatens the rejection of the membrane and inefficient water purification.

In the literature, quite a lot of methods and ways of membrane protection are described, but almost no attention is paid to pre-membrane cartridges (filters), since they are quite cheap, and their regeneration does not have a significant economic effect. However, from an ecological point of view, the accumulation of such waste is quite critical and dangerous. The availability of carbon filters significantly complicates the regeneration of filters for mechanical (pre-membrane) cleaning. In their research, I.I. Radovenchyk *et al.* (2023) studied the problem of filter disposal and environmental contamination due to the use of reverse osmosis systems. They noted that after 3-6 months, such filters are thrown out with the usual garbage. This leads to environmental contamination, which carries with it a number of threats. The authors point out that there are no enterprises in Ukraine dealing with the disposal of such filters. Researchers M.V. Kravchenko *et al.* (2021) proved that contamination and untimely replacement of filters precede membrane contamination and reduce its service life by 30-35%. Preventive and corrective measures are required for continuous operation of the plant. These corrective and preventive measures are necessary according to the properties of the outlet water as well as the properties of purified water, for example, in purified water after desalination, an environment is formed which can cause corrosion in the corresponding equipment. These measures can be implemented in pre-treatment methods, internal therapy modification, and post-treatment requirements. Periodic measurements can be used to analyse productivity.

Scientists X. Zhang and Y. Liu (2021) noted that the quantity and quality of the outlet water can change when the properties of the inlet water and operating parameters change. Scale formation on the membrane is caused by settling on the surface of the membrane (heterogeneous nucleation) or settling in the bulk solution (homogeneous nucleation) followed by settling on the membrane. In their study, C. Skuse *et al.* (2021) noted that calcium and silicon are the two ions that are usually responsible for membrane scale formation.

Scientists M. Jafari *et al.* (2020) described in detail an important indicator for detecting the contamination of membranes and filters – the silt density index (SDI, an indicator of the potential of the outlet water contamination). This is an on-site calculation of colloidal and suspended particles. This is a test to analyse the productivity of equipment used for pre-treatment of water. Silt density index measurements should be performed using pre-cleaning cartridge filters. The value of SDI should be as minimal as possible, which can be achieved by operating the filter material at its calculated pressure drop (ΔP) and proper backwashing of mechanical filters. SDI can also be minimised by using auxiliary filters in the system of polystyrene and carbon cartridges, but in Ukraine this practise has not yet been introduced into mass production.

V. Radovenchyk *et al.* (2023) studied the possibility of mechanical filters cleaning with sulphuric acid solutions and showed positive results of this method. At the same time, oxides and hydroxides of iron (III) pass into the solution and release the pores of the cartridge. However, the use of highly concentrated sulphuric acid is possible only in industrial conditions.

Therefore, the purpose of the study was to find, substantiate and confirm by laboratory experiments ecologically safe and affordable methods for regeneration of polystyrene filters of baromembrane water purification plants in household conditions.

MATERIALS AND METHODS

To perform the task of assessing the possibility of regeneration of used filters of reverse osmosis plants, the method of analysis in laboratory conditions was used, and the method of chemical analysis was used to evaluate the change in the hydrogen index and concentrations of contaminants. The method of functional analysis was used to observe the processes of settling, which made it possible to determine the optimal indicators of settling of the suspension. To determine the content of iron as the main contaminant of mechanical filters, the sulphate salicylic method was used according to DSTU 7262:2012 "Chemical reagents. Methods of determination of iron impurities" (2013).

The experiments were conducted on the basis of the laboratory of the Department of Ecology and Technology of Plant Polymers of the Faculty of Engineering and Chemistry of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". A cycle of experiments on cleaning of used mechanical filters of reverse osmosis (Ecosoft, Ukraine) in a specially assembled plant with a gradual increase in the concentration of sulphuric acid was conducted, the processes of cleaning the pores of the filters from iron compounds were carried out according to formula (1):

$$2Fe(OH)_3 + 3H_2SO_4 \rightarrow Fe_2(SO_4)_3 + 3H_2O. \quad (1)$$

Measurements of concentrations and the hydrogen indicator were made every 8 min after circulation of the solution in a purification plant. Different doses of sulphuric acid were used and the hydrogen indicator was monitored using a pH metre when the acid was gradually introduced into the cleaning solution. The optical density was also measured on a photocolorimeter using the sulphate salicylic method and the content of iron in the solution was determined. The introduction of sulphuric acid was stopped when the concentration of iron ions remained unchanged, that is, when the filters were completely cleaned.

To determine the optimal hydrogen indicator, at which complete neutralisation of sulphuric acid in the solution and effective filtering of the obtained suspension would occur, a number of experiments on model solutions were conducted. 26.9 g of $FeCl_3*6H_2O$ were dissolved in distilled water, the obtained solution was brought to 1000 ml with the pH = 2.1. Five samples of 100 ml were taken. With the help of NaOH, the pH was gradually increased by 0.5 to 4. With the help of measuring cylinders, the intensity of settling and distribution into suspension and liquid phase were observed. Then the resulting solution was settled for a day and filtered.

RESULTS

The results of measurements of the intensity of settling and filtration of the suspension obtained in the process of mechanical filters cleaning using sulphuric acid solution of different concentrations (the effect of concentration is noted on the hydrogen indicator for $FeCl_3*6H_2O$) are shown in Table 1.

Table 1. Results of measurements of settling intensity and suspension filtration

pH $FeCl_3*6H_2O$	**2.6**	**3.1**	**3.6**	**4.03**	**10**
V (NaOH), ml	2.5	2.62	2.75	2.78	2.8
The ratio of the liquid phase to the suspension, % during settling for a day	100/0	100/0	98/2	80/20	23/77
The volume of filtrate in the cylinder when filtering the suspension during the day	100	100	72	48	100
Notes*	The filter is coloured but clean, the filtrate is coloured, all iron is in the filtrate	Filtering is very slow, there is little sediment on the filter, the filtrate is coloured, most of iron is in the filtrate	Filtering is slow. The filtrate is transparent, the concentration of iron in the filtrate is 1.6 mg/l	The filtrate is transparent, the concentration of iron in the filtrate is 0.2 mg/l	Filtering is fast, iron is not detected in the filtrate (the reaction to sulphite salicylic acid and ammonia)

Source: developed by the author based on the results obtained

As can be seen from the research results shown in Table 2, the content of iron in the most contaminated filter is 6.5 g/l, with a norm of 0.2 g/l according to State sanitary standards and rules "Hygienic requirements for drinking water intended for human consumption" (2022). At the same time, 60 ml of sulphuric acid with a concentration of 98% has been used to clean one contaminated filter.

Table 2. Results of iron concentration measurements during cleaning of three mechanical polystyrene filters with a gradual increase in the dose of sulphuric acid

Hydrogen indicator, pH	**Concentration of iron ions, g/l**	**Hydrogen indicator, pH**	**Concentration of iron ions, g/l**	**Hydrogen indicator, pH**	**Concentration of iron ions, g/l**
Filter 1		Filter 1, 2		Filter 3	
pH (1) = 4.5	0.001	pH (0) = 5	0	pH (0) = 5	0
pH (2) = 4.0	0.02	pH (1) = 4.5	0.1	pH (1) = 4.5	0.002
pH (3 = 3.4	0.3	pH (2) = 3.8	0.9	pH (2) = 4.0	0.1
pH (4) = 3.2	0.6	pH (3) = 3.3	3.8	pH (3) = 3.5	1.5
pH (5) = 2.5	3.8	pH (4) = 2.7	5.8	pH (4) = 2.5	5.9
pH (6) = 1.1	6	pH (5) = 1.1	5.8	pH (5) = 1.0	6.5

Source: developed by the author based on the results obtained

The next task, after cleaning the filters, is the neutralisation of products of cleaning, because their discharge into the sewage system is unacceptable, since pH = 1. According to observations, effective removal of dissolved iron from the cleaning solution begins at pH = 4 and continues until pH = 10. However, at pH = 4 settling and filtering require 24 hours, while at pH = 10 this process takes no more than an hour. Additional studies confirm that 4.5 is the most effective pH indicator for settling and almost complete removal of iron from the solution. Further increase in the hydrogen indicator to 10 turns out to be impractical. The research data obtained are used to neutralise sulphuric acid, which, together with dissolved iron removed from the used filter, is in the mother solution.

The volume of the mother solution, which has been formed as a result of cleaning of contaminated mechanical filter of the return osmosis plant, is 1 l, its pH = 0.95, the solution contains 65 ml of sulphuric acid, this volume has been used in previous studies for complete cleaning of mechanical filter. A neutralisation reaction (2) is carried out in order to remove sulphuric acid, which is dangerous for discharge, from the used solution. The calculated amount of 37.42 g of CaO is added, bringing the pH to 5.3. At this level of the hydrogen indicator, the most efficient settling of gypsum ($CaSO_4$) and separation of solid and liquid phases occur:

$$CaO + H_2SO_4 \rightarrow CaSO_4 + H_2O. \quad (2)$$

The filtration of the obtained suspension on paper filters “blue tape” is the next stage. The volume of the filtrate is 776 ml. The mass on the filter before drying is 265.61 g. After complete drying in the drying chamber, the obtained substance is brought to a constant mass of 79.50 g (Fig. 1).

Figure 1. Cement additive obtained as a result of neutralisation of the mother solution from the contaminated filter
Source: developed by the author

This method of regeneration of contaminated filters (using sulphuric acid solution) can be used in the first and third stages of filtration if the first and third filters are made of polystyrene with a corresponding pore diameter of 10 and 5 μm, respectively. By visual examination of the used filters, it has been determined that iron is the most problematic and main contaminant of filters, and even with a small amount of it, it is very difficult to get rid of reddish colour after several stages of cleaning with an acid solution. Therefore, the pH of the cleaning solution should be brought up to 1, while the norm in tap water is 5-6, as shown by the analysis of water samples by the sulphate salicylic method in the laboratory. Figure 2 shows sets of clean (a) and used (b) mechanical filters of the 1st, 2nd and 3rd stages of water purification in reverse osmosis plants.

Figure 2. View of Ecosoft filters before and after use in a typical reverse osmosis plant for three months
Source: developed by the author

The set includes two polystyrene filters, which, unlike the carbon one, can be easily cleaned of iron using sulphuric acid. Such a combination of filters for reverse osmosis plant is designated in the international classification as PP, CTO, PP (polypropylene 10 μm, carbon, polypropylene 5 μm). However, sometimes there are other combinations described by F.M. Idrees (2020), such as PP, UDF, CTO (polypropylene 10 μm, from

granular activated carbon, pressed carbon cartridge). In general, the approach to cleaning these types of filters does not change, because cartridges are based on coal as the main filler in various modifications.

Purification with a sulphuric acid solution can be used in the first and third stages of filtration if the first and third filters are made of polystyrene. In addition to the chemical method of neutralising the mother solution, the electrodialysis method can be used, while an electrolyser with a lead anode would be the best option. In the process of cleaning the filters, a suspension, from which iron is settled with CaO, filtered and dried, has been obtained. After cleaning of one mechanical filter, almost 80 g of pure gypsum has been obtained. This gypsum can serve as a binding cement additive or be used for the production of internal cladding plates, and clean water can be used for subsequent cycles of mechanical filters cleaning with the subsequent addition of sulphuric acid to it. According to laboratory studies on model solutions with different iron content (from 5.38 to 16.4 grams per 1 litre of solution), the effect of sulphuric acid on iron solubility is shown in Table 2.

As a result of the experiment on cleaning of used mechanical filters, it is found that the content of iron in the most contaminated filter is 6.5 g/l. At the same time, 60 ml of sulphuric acid with a concentration of 98% is used to clean one contaminated filter. From the results of the measurements, it can be seen that with gradual addition of sulfuric acid to the cleaning solution, iron compounds begin to actively dissolve at pH = 3.3, and the maximum efficiency of filters cleaning from iron compounds is achieved at pH = 1.6. The introduction of sulphuric acid has been stopped at an unchanged indicator of iron ions concentration, in other words, at complete cleaning of filters, which occurs at pH = 1.6. These results have been also confirmed during repeated naturalistic experiments with real contaminated filters, removed from users of household reverse osmosis systems in Kyiv. Since the naturalistic experiment involved filters of various degrees of contamination, which depends on the intensity of use, the number of people in the family, the season and the period of use (from 3 to 7 months), the necessary volume of sulphuric acid, needed for filter cleaning, was in the range from 20 to 78 ml per 1 litre of the cleaning solution.

DISCUSSION

According to the results of the experiment, used polypropylene filters can be regenerated and reused in baromembrane water purification plants. Bacteriological safety of using such filters was confirmed by scientists T.Yu. Nyzhnyk *et al.* (2019). However, the reuse of mechanical polystyrene filters was confirmed only by bacteriological indicators. The conducted research also proved the possibility of regeneration for the main chemical contaminant – iron compounds. Therefore, the conducted studies complement and do not contradict the obtained results. In their work, S.V. Huliienko *et al.* (2020) noted that salt rejection and permeate flow were key factors for analysing the productivity of reverse osmosis plants. Membrane productivity can also be analysed by the salt rejection rate of the membrane system.

In the course of research, it has been determined that an aqueous acidic environment (having a pH from 1 to 4) is the best cleaner for replaceable blocks (filters and membranes) of reverse osmosis plants. Increase in acidity, i.e. bringing the pH closer to 1, ensures the dissolution of water-insoluble calcium and silica salts, iron compounds and organic contaminants. Positive effect of the hydrogen indicator on the rejection of salts from reverse osmosis filters and membranes was theoretically described in the works of S.V. Huliienko and I.V. Syman (2018) and A.P. Safonyk *et al.* (2020). In the conducted research, it is possible to prove this theory in practise and to determine specific pH limits, in which the acidity of the environment will be effective. It has been proven that a further decrease in pH < 1 is impractical, because undissolved iron on the filters can no longer exist at this state of the environment. According to the results of the experiments, increase in the hydrogen indicator to 10 is impractical. The obtained research data are used to neutralise sulphuric acid, which, together with dissolved iron extracted from the used filter, is in the mother solution. Researchers M.V. Kravchenko *et al.* (2021) concluded that the settling on the surface of membranes and cartridges of some calcium salts, such as gypsum, is also influenced by high temperatures, as they reversely dissolve with increasing temperature. Gypsum crystals can block and deform the pores, which leads to a rapid decrease in water flow and wetting of the membrane. Calcium ions can also precipitate as calcite or in a complex with organic compounds and cause severe flow reduction, a phenomenon described in detail by S. Virych (2023).

As a result of examination of the content and analysis of the used carbon filler, it is found that it is impossible to chemically regenerate carbon filters. According to the conducted studies, even with a high level of purity of water supplied to activated carbon filters and regular thermal sanitisation (washing with water at 95°C for 2.5 hours), it is impossible to completely remove biological contamination from the cartridge filler. During the process of hot water sanitisation, surviving single planktonic cells begin an active process of microcolonies formation and biofilm creation as a survival mechanism in extreme conditions. Already three days after heat treatment, the formation of a biofilm is observed on the surface of activated carbon granules, which leads to a phenomenon known as biological activation of coal. In her work, M.V. Kravchenko (2022) consistently investigated the behaviour of coal cartridges and proved that due to the peculiarity of coal bulk material, it cannot be restored by detaching contaminant ions. Researchers Ch.Ch. Wu *et al.* (2019), describing the reproduction of bacteria in

contaminated carbon filters, came to the same conclusion about the impossibility of regeneration of these filters precisely because of sorption-biological interaction of the sorbent and the contaminant. The results of the observations refute the possibility of creating cartridges with a separate zone for the accumulation of waste contaminants, since in practise the carbon filter behaves differently than in the theoretical work of N.O. Matlakh *et al.* (2022), where methods of modifying polypropylene and carbon filters are shown, with structural construction of filter sections so that sediment accumulates at the bottom of the sections, the bottom of the section must be unscrewed so that the sediment can be removed, and new filters must be easily inserted into the section. The work includes measures to control contamination of pre-membrane filters (titanium plates and ammeter). This solution can extend the service life of mechanical filters, but will not solve the issue of their regeneration and disposal.

Silica scale formation is slower than that of calcium salts such as gypsum because silicic acid polymerises more slowly than gypsum crystallises. Processes of silica scale formation described by I.A. Tereshchenko (2021) proved a high level of threat of silica influence on cleaning efficiency at neutral pH ranges. Even NaCl solutions can cause flow reduction if available in supersaturating concentrations. A. Kayvani Fard *et al.* (2018) investigated that scale mitigation by methods that include volume filtration and antiscalant addition was a rather fictitious way to extend the service life of the membrane and the plant as a whole.

However, the use of a standard set of cartridge filters helps to minimise flow reduction by removing precipitated salts before they reach the membrane. Therefore, timely replacement of filters makes it impossible to damage the membrane with subsequent loss of its efficiency. However, the problem of environmental contamination with such filters has still not been solved globally, because most of them are thrown away either in the trash or in special containers. The achievement of more sustainable and environmentally safe alternatives becomes a critical task for preserving natural resources and maintenance of sustainable development.

CONCLUSIONS

The use of carbon filters complicates the process of restoring filters for mechanical (pre-membrane) cleaning. Iron is the most significant and main contaminant for mechanical polystyrene filters. Even with small iron content, after several stages of cleaning with an acid solution, it is very difficult to remove the reddish colour. Therefore, in order to achieve the effectiveness of the solution cleaning, it is necessary to reduce its pH to the level of 1. This deviation from the usual value of tap water, which is 5-6, has been confirmed by the analysis of water samples using the sulphate salicylic method in the laboratory. After that, filters must undergo forced washing in the circulation network to normalise the hydrogen indicator in the thickness of material and be safe for further reuse in the reverse osmosis plant after regeneration.

It can be seen from the results of measurements that during gradual introduction of sulfuric acid into the cleaning solution, iron compounds begin to actively dissolve at pH = 3.3, and the highest effectiveness of filters cleaning from iron compounds is achieved at pH = 1.6. The addition of sulphuric acid is stopped when the concentration of iron ions remains stable. In general, complete cleaning of filters takes place at pH = 1.6. The obtained results are confirmed during repeated natural experiments with real contaminated filters, which have been taken from users of household reverse osmosis systems in the city of Kyiv.

If polystyrene filters can be completely regenerated with sulphuric acid and after further washing can be used again in the system, then carbon filters are almost impossible to regenerate. The reason is that contaminated coal filler is in a non-separable structure and the very structure of the filler does not allow the contaminant to be removed from the thickness of the coal layer. The case is not disassembled and after removing the carbon filler from it, it must also be disposed of. In Ukraine and in the world, such cartridges are simply thrown away. Regardless of whether they have fillers or have already had them removed, the amount of waste remains the same. This method of regeneration of contaminated filters can be used in the first and third stages of filtration if the first and third filters are made of polystyrene. And reverse osmosis carbon filters, unlike mechanical ones, cannot be cleaned.

In conclusion, it can be noted that reverse osmosis carbon filters, unlike mechanical ones, cannot be processed. The author came to this in the result of a dissection of the cartridges containing these filters. The case is not disassembled and after removing the carbon filler from it, it must also be disposed of. According to the examination of the content and analysis of the used carbon filler, it becomes clear that it is impossible to chemically regenerate carbon filters. According to research, it is impossible to completely clean the cartridge filler, activated carbon, from biological contaminants. Even with high indicators of purity of water entering carbon filters, and periodic thermal sanitisation (washing with water at 95°C for 2.5 hours), it turns out to be impossible to stop microflora development for a long time. During the sanitisation treatment of the filter with hot water, surviving individual plankton cells begin the process of active microcolonies formation and biofilm creation as a survival mechanism in extreme conditions. Already three days after heat treatment, biofilm formation is observed on the surface of activated carbon granules, which leads to the so-called biological activation of coal.

Therefore, the most effective method to recycle carbon filters is to develop a collapsible design of the

cartridge, which can be filled with fresh activated carbon, and the used carbon can be burned at high temperatures or sent to the production of coal briquettes. Such briquettes are completely safe for the environment and can be used as solid fuel. The identified and researched methods of regeneration and recycling of mechanical and carbon pre-membrane filters can be applied on an industrial scale with the condition of their larger-scale testing. This technology is inexpensive and allows polystyrene filters to be fully regenerated and reused and it is possible to obtain fuel briquettes from carbon filters during further research on improving the recycling scheme. Among the directions for further research, the following ones can be highlighted: design development and improvement; study and use of new technologies for filters cleaning; energy efficiency, etc. In addition to the mentioned directions, the following aspects are important for study: persistence of contaminations and efficiency of removal of various types of them, materials and membranes, etc.

ACKNOWLEDGEMENTS

The author expresses her gratitude to the staff of the Department of Ecology and Technology of Plant Polymers of the Faculty of Engineering and Chemistry of the National Technical University of Ukraine "Igor Sikorskyi Kyiv Polytechnic Institute" for consultations and assistance in conducting laboratory research. Thanks are also expressed to the residents of Kyiv who agreed to participate in the experiment and provided statistical information, used cartridges and membrane blocks of reverse osmosis plants (Olena Kyrylenko, Oleksii Antonov, Olga Petryshyna, Alina Sobol, etc.).

CONFLICT OF INTEREST

None.

REFERENCES

[1] DSTU 7262:2012. (2013). *Chemical reagents. Methods of determination of iron impurities*. Retrieved from https://online.budstandart.com/ua/catalog/doc-page?id_doc=54398.

[2] Huliienko, S.V., & Syman, I.V. (2018). Membrane contamination and methods of their regeneration: A critical review. *International Scientific Journal "Internauka"*, 5, 51-56.

[3] Huliienko, S.V., Korniienko, Y.M., Metlina, M.S., Tereshenko, I.Y., & Kaminskyi, V.S. (2020). The correction of the dimensionless equation for the mass transfer coefficient estimation during the membrane modules regeneration. *Journal of Engineering Sciences*, 7(2), 24-29. doi: 10.21272/jes.2020.7(2).f4.

[4] Idrees, F.M. (2020). Performance analysis and treatment technologies of reverse osmosis plant – a case study. *Case Studies in Chemical and Environmental Engineering*, 2, article number 100007. doi: 10.1016/j.cscee.2020.100007.

[5] Jafari, M., Zlopasa, J., D'haese, A., Cornelissen, E.R., Vrouwenvelder, J.S., Verbeken, K., Verliefde, A., van Loosdrecht, M.C.M., & Picioreanu, C. (2020). A comparison between chemical cleaning efficiency in lab-scale and full-scale reverse osmosis membranes: Role of extracellular polymeric substances (EPS). *Journal of Membrane Science*, 609, article number 118189. doi: 10.1016/j.memsci.2020.118189.

[6] Joseph, T.M., Al-Hazmi, H.E., Śniatała, B., Esmaeili, A., & Habibzadeh, S. (2023). Nanoparticles and nanofiltration for wastewater treatment: From polluted to fresh water. *Environmental Research*, 238(1), article number 117114. doi: 10.1016/j.envres.2023.117114.

[7] Kayvani Fard, A., McKay, G., Buekenhoudt, A., Al Sulaiti, H., Motmans, F., Khraisheh, M., & Atieh, M. (2018). Inorganic membranes: Preparation and application for water treatment and desalination. *Materials*, 11(1), article number 74. doi: 10.3390/ma11010074.

[8] Kravchenko, M.V. (2022). Reverse osmosis – as a method of purification of drinking water at the local level. In *Current issues, priorities and development strategies of Ukraine: Abstracts of the IV International scientific-practical online conference* (pp. 113-116). Kyiv: ITTA.

[9] Kravchenko, M.V., Vasylenko, L.O., & Voloshkina, O.S. (2021). Application of the reverse osmosis method for the purification of drinking water. *Environmental Safety and Natural Resources*, 40(4), 32-45. doi: 10.32347/2411-4049.2021.4.32-45.

[10] Matlakh, N.O., Trushakov, D.V., Kozlovskyi, O.A., & Fedotova, M.O. (2022). Creation of a multi-section water purification filter with automated control of membrane pollution. *Design, Production and Operation of Agricultural Machines*, 52, 166-177. doi: 10.32515/2414-3820.2022.52.166-177.

[11] Nyzhnyk, T.Yu., Nyzhnyk, Yu.V., Strykalenko, T.V., & Marievsky, V.F. (2019). Disinfectant filter material from polypropylene filter production waste. In *10th All-Ukrainian scientific and practical conference of young scientists, graduate students and students "Water in the food industry"* (pp. 72-74). Odesa: ONUT.

[12] Qasim, M., Badrelzaman, M., Darwish, N.N., Darwish, N.A., & Hilal, N. (2019). Reverse osmosis desalination: A state-of-the-art review. *Desalination*, 459, 59-104. doi: 10.1016/j.desal.2019.02.008.

[13] Radovenchyk, I.I., Karpenko, M.I., & Ivanenko, O. (2023). Removal of iron compounds from mechanical filters of domestic reverse osmosis water treatment systems. *Power Engineering: Economics, Technique, Ecology*, 2, 127-132. doi: 10.20535/1813-5420.2.2023.279720.

[14] Radovenchyk, V., Radovenchyk, I., Ivanenko, O., & Karpenko, M. (2023). Removal of iron compounds from mechanical filters of household reverse osmosis systems water purification. *Ecological Engineering & Environmental Technology*, 24(6), 163-172. doi: 10.12912/27197050/168097.
[15] Safonyk, A.P., Targoniy, I.M., & Hrytsiuk, I.M. (2020). Development and research of the electromagnetic installation for purification of process water from ferromagnetic impurities as an object of automation. *Electronic Modeling*, 42(4), 87-102. doi: 10.15407/emodel.42.04.087.
[16] Skuse, C., Gallego-Schmid, A., Azapagic, A., & Gorgojo, P. (2021). Can emerging membrane-based desalination technologies replace reverse osmosis? *Desalination*, 500, article number 114844. doi: 10.1016/j.desal.2020.114844.
[17] State sanitary standards and rules "Hygienic requirements for drinking water intended for human consumption" (St.SanStandart 2.2.4-171-10). 2022, March. Retrieved from https://zakon.rada.gov.ua/laws/show/z0452-10#n25.
[18] Tereshchenko, I.A. (2021). The process of mass exchange during the removal of sediments from the surface of roll membrane modules. (Master's thesis, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine).
[19] Virych, S. (2023). Patterns of concentration polarization during reverse osmosis. (Master's dissertation, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine).
[20] Warsinger, D.M., Chakraborty, S., Tow, E.W., Plumlee, M.H., Bellona, Ch., Loutatidou, S., … Lienhard , J.H. (2018). A review of polymeric membranes and processes for potable water reuse. *Progress in Polymer Science*, 81, 209-237. doi: 10.1016/j.progpolymsci.2018.01.004.
[21] Wu, Ch.Ch., Love, N.G., & Olson, T.M. (2021). Bacterial transmission and colonization in activated carbon block (ACB) point-of-use (PoU) filters. *Environmental Science: Water Research & Technology*, 7, 1114-1124. doi: 10.1039/D0EW00982B.
[22] Zhang, X., & Liu, Y. (2021). Reverse osmosis concentrate: An essential link for closing loop of municipal wastewater reclamation towards urban sustainability. *Chemical Engineering Journal*, 421(2), article number 127773. doi: 10.1016/j.cej.2020.127773

Дослідження можливостей очищення механічних фільтрів баромембранних установок

Маргарита Вікторівна Карпенко

Аспірант
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»
03056, пр. Берестейський, 37, м. Київ, Україна
https://orcid.org/0000-0003-3237-4130

Анотація. Однією з найбільш актуальних проблем, які виникають при обробці води за допомогою установки зворотного осмосу, є поява забруднення на мембрані та фільтрах, що суттєво зменшує ефективність роботи (продуктивність розділення, потік пермеату води, відторгнення солі) і термін служби установки. Тому метою роботи став пошук оптимальних методів очищення та регенерації механічних фільтрів установок зворотного осмосу. Розв'язання цієї проблеми проводилося шляхом емпіричного аналізу наявної наукової інформації та проведення лабораторних досліджень з очищення картриджів та обробки інфільтрату. Було проведено цикл дослідів з очищення використаних механічних фільтрів зворотного осмосу в спеціально змонтованій установці з поетапним збільшенням концентрації сірчаної кислоти. Згідно зі спостереженнями під час проведення лабораторних досліджень, ефективне виведення розчиненого заліза з розчину, яким проводилось промивання забруднених полістирольних фільтрів, починається при pH = 4 та до pH = 10, проте при pH = 4 відстоювання і фільтрування потребує 24 години часу, а при pH = 10 цей процес займає не більш ніж годину. В результаті подальших досліджень найефективніший водневий показник для осадження і практично повного виведення заліза з розчину становить 4,5. Подальше підвищення водневого показника до 10 є недоцільним. Крім хімічного методу нейтралізації маточного розчину, ефективним варіантом є застосування електродіалізу, при цьому оптимальним вибором є використання електролізера з анодом зі свинцю. Отже, після очищення одного механічного фільтра було отримано майже 80 г чистого гіпсу. Цей гіпс може слугувати як високоякісна добавка для цементу або використовуватися при виготовленні плит для внутрішнього облицювання. Чисту воду, яку отримано під час процесу, можна використовувати для наступних етапів очищення інших механічних фільтрів

Ключові слова: зворотний осмос; картриджі; полістирольні фільтри; регенерація; вугільні фільтри; концентрат

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 004.9
DOI: 10.62660/2306-4412.4.2023.36-49

Improving sip trunking services with cloud-based SBC: Performance analysis and optimisation strategies

Oleksandr Pidpalyi*
Postgraduate Student
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
03056, 37 Beresteiskyi Ave., Kyiv, Ukraine
https://orcid.org/0009-0007-6852-7959

Abstract. In modern telecommunications networks, session initiation protocol trunking is an important technology for voice and video communication over Internet protocol networks. Cloud-based session boundary controllers play a key role in ensuring security and quality of service but there are challenges to their performance and optimisation, especially as data volume and users grow. The purpose of this study is to develop a strategy for achieving efficient and scalable session initiation protocol trunking services in cloud environments. Methods of theoretical analysis and practical experiments are used. Analysis of the performance of cloud controllers identifies important analytical conclusions about their capabilities, and the developed optimisation strategies provide practical recommendations for improving the provision of cloud trunking services. This analysis also shows that performance can vary depending on the implementation and settings. It is established that the main factors affecting performance are network latency, bandwidth, and configuration features of controllers. Based on this information, an optimisation strategy is developed, including implementing quality of service for prioritising session initiation protocol traffic, methods for automatically scaling resources, redundancy, and optimising network protocols. It is demonstrated that the use of Session Initiation Protocol-trunking and cloud-based Session Border Controller can substantially improve the quality of service in telecommunications systems, which allows optimising call routing and ensuring the security of communications. Evaluating the performance of the Session Border Controller cloud provided valuable insight into their capabilities, and the proposed optimisation strategies can become specific recommendations for improving the provision of Session Initiation Protocol trunking services in cloud environments. The results of the study can serve as valuable recommendations for telecommunications professionals who want to deploy efficient and scalable trunking solutions in cloud networks

Keywords: session initiation protocol; communication service quality; session boundary controllers; network delays; resource scalability

Article's History: Received: 25.08.2023; Revised: 31.10.2023; Accepted: 18.12.2023.

INTRODUCTION

Cloud-based Session Border Controller (SBC) is a critical element of modern telecommunications networks, especially in the context of providing Session Initiation Protocol (SIP) services-trunking over IP (Internet Protocol) networks. SIP trunking is defined as a technology that allows transmitting voice and multimedia traffic over the internet protocol instead of conventional telephone networks and cloud-based SBC are key elements in modern SIP trunking networks because of their ability to ensure the security, quality of service, and reliability of communication services. However, with the constant growth of data volumes and the number of users on networks, cloud-based SBC is facing new challenges and tasks. Modern cloud-based SBC systems are complex systems with many settings and capabilities. However, their performance can vary

Suggested Citation:
Pidpalyi, O. (2023). Improving sip trunking services with cloud-based sbc: Performance analysis and optimisation strategies. *Bulletin of Cherkasy State Technological University,* 28(4), 36-49. doi: 10.62660/2306-4412.4.2023.36-49.

*Corresponding author

substantially depending on the implementation and configuration features. One of the key challenges that need to be addressed is improving the efficiency and scalability of these controllers to provide high-quality user service in growing networks. The relevance of the research topic is determined by the need to improve and optimise cloud SBC to improve the quality and efficiency of SIP trunking services in cloud environments.

There is an extensive body of research in the field of SIP trunking and cloud session boundary controllers. For example, P.I. Romaniuk (2023) investigated the use of cloud technologies in the educational process. The author provided an overview of popular cloud platforms and considered their prospects in the educational process, and also noted that the use of cloud technologies helps to increase efficiency and reduce costs in the field of higher education, provides fast and secure access to the necessary services for training. V. Mazharenko (2020) highlighted that SIP trunking creates the ability to connect a local telephone system to the Internet instead of a regular telephone network, simplifying the provision of Voice Over Internet Protocol (VoIP) services to companies located on the host. The author examines the possibilities and methods of integrating IP telephony in the areas of providing telephone services, evaluates the quality of use of the developed SIP client, and comes to the conclusion about the purposefulness of using IP telephony on web resources. A paper of A. Praveen *et al.* (2022) demonstrates the importance of testing SBC functionality in modern unified communications systems, modernisation of contact centres, and cloud communication services. The researchers note that manual testing of various SBC call scenarios is complex, time-consuming, and error-prone. However, this problem can be solved by creating an automated approach for testing SBC, simulating a real-world production environment, and using SIP scripts to test all important VoIP call scenarios. They offer a methodology for testing SBC to ensure the reliability of its operation in enterprises and service operators. Authors S. Kaul and A. Jain (2019) discuss the possibilities of using cloud session boundary controllers (C-SBC) in cloud-based unified communications using virtualisation. They emphasise that with the advent of a new generation of communication tools, the volume of real-time communication and intelligent traffic is growing every day, which makes communication in enterprises quite difficult. The proposed solution helps enterprises in the field of next-generation internet protocol networking, media processing and transcoding, which is usually used for various next-generation services.

T. Smirnova *et al.* (2022) focused their investigation on selecting the optimal technology for a cloud-based information and communication system to optimise production processes. It includes the analysis and selection of communication technologies and the development of a multi-criteria optimisation model to achieve this goal. The results of the study confirm the feasibility of using 5G cellular networks for automating production processes and provide a methodology for choosing optimal solutions using these technologies. Authors M. Skulysh *et al.* (2018) and A. Bondarchuk and V. Zhebka (2023) in their papers highlighted the principles of optimising the service of hybrid telecommunications services in a heterogeneous environment. The basic idea is that subscriber process management functions can be migrated to the cloud environment, including subscriber search and physical element management. All mobile communication subsystems are controlled by controllers located in the data centre, and interaction between these controllers is conducted within the centre, which simplifies the management of the telecommunications network and reduces the number of service flows.

In the study by I. Tas *et al.* (2020), a new SIP attack, SR-DRDoS, is presented, which uses lesser-known SIP capabilities along with IP address spoofing techniques and DDoS attack logic. A SIP-based DoS/DDoS attack simulator was developed to investigate the SR-DRDoS attack. The attack showed the ability to increase the CPU load of the SIP server from 0 to 100% in just 4 minutes. In the context of protecting against such attacks, it is noted that many SBCs leave organisations with a false sense of security due to insufficient ability to detect and track such attacks. M.R.S. Al Saidat (2019) offers a secure SIP-based distributed architecture model that can be deployed in a service provider's data centres to ensure availability, scalability, and security. The proposed model is designed to reduce SIP attempts on session initiation controllers. The study includes testing the model in a real-world service provider environment, the results of which showed the ability of the improved session protocol backup initiation model to reduce attempts to initiate the protocol via SIP trunking.

Researchers T.M. Tardaskina and T.V. Hryshchuk (2021) examined the cloud technology market in Ukraine and determined the role of telecommunications operators and their interaction with consumers. The study results indicated that Ukrainian telecommunication operators have a small share in the cloud technology market, but large companies can increase their presence in this sector by adding cloud services to their portfolio, acting as intermediaries between providers and users. L. Tokar (2022) examined the advantages of cloud-based automated telephone systems (ATC) over traditional methods of organising telephone communication in companies and offices. The study highlights the capabilities of cloud ATC systems, such as multi-channel numbers, intelligent call forwarding, flexible analytics, and a number of other features that contribute to the quality and efficiency of telephone communication. The main benefits are cost savings and the ability to quickly scale resources in a cloud ATC. Unlike the above-mentioned studies, which focus on cloud technologies in the field of telecommunications, this

paper is aimed specifically at examining and improving the provision of SIP trunking services through the introduction of cloud-based SBC.

The purpose of this study was to analyse the performance of cloud SBC and identify possible limitations to propose optimisation strategies to improve performance and reliability in the field of SIP trunking services.

MATERIALS AND METHODS

Methods of theoretical analysis and practical experiments were used to achieve a comprehensive understanding of SBC clouds and their impact on SIP trunking, forming an empirical analysis approach. The theoretical analysis method was used to examine in detail the theoretical aspects of cloud SBC, their role in SIP trunking, and analyse various papers on a given topic. Theoretical analysis allowed understanding the basic principles of SBC cloud operation and their potential impact on trunking services. This method helped to identify such basic aspects as the principles of SBC cloud operation, the role of SIP trunking, the potential impact on trunking services, and comparison with other solutions. The basic principles of functioning of cloud-based SBC are studied, including their ability to route SIP traffic, authenticate, transform, and ensure communication security. The important role of cloud-based SBC in improving the quality of SIP trunking and ensuring reliable delivery of SIP requests and responses between different networks is evaluated. The study examines how the implementation of cloud-based SBC can affect the quality of SIP trunking service, reduce latency, and provide greater scalability of services. Compare cloud-based SBC with other methods of trunking and routing SIP traffic, highlighting their advantages and possible disadvantages.

Real-world implementations of SBC cloud providers such as Amazon Chime Voice Connector, Twilio Elastic SIP Trunking, Azure Communication Services, Dialogflow Telephony Gateway, and Oracle Cloud Infrastructure were used to conduct practical experiments. These implementations were chosen because of their popularity and prevalence in the market, which allowed conducting an objective comparative analysis of their characteristics and performance. Practical experiments included modelling test SIP sessions, generating traffic, creating block diagrams and tables, outputting calculations, etc. A simple programme was written to create test SIP Sessions to simulate real-world communication events between users, which also generated traffic transmitted through cloud-based SBC's to measure their performance and traffic handling capabilities. The results obtained were processed to compare the characteristics of certain cloud SBC's, determine the quality of service, and assess the impact on performance and call quality. Graphs that showed a comparative analysis of SIP trunking services in different regions to visualise the results were constructed. During the experiments, various tasks and situations occurred that required solving and analysing in the context of SBC cloud functionality. Formulae are also used to calculate call quality, relative performance gain, loss reduction, bandwidth redundancy, and network load calculations.

The IntelliJ IDEA environment and the Java programming language were used for software implementation. The Cross-platform File Format Apps platform is selected for building block diagrams. MS Excel is used to plot a comparative analysis of SIP trunking services in different regions. Call setup time, call quality assessment, resource usage and scalability assessment, network latency, bandwidth, availability, communication quality, usage scale, regulatory support, infrastructure, etc. were used as performance metrics and evaluation criteria. Formula (1) is used to evaluate the call quality:

$$MOS = 1 + (R - 1.5 * A) + (A * (A + 1) * B * C), \quad (1)$$

where R – voice quality rating (usually from 1 to 5); A – jitter (change the delay between packets); B – packet loss (percentage loss); C – delay (the sum of delays in milliseconds).

The relative productivity gain can be calculated using Formula (2):

$$(\%) = ((Y - X)/X) * 100, \quad (2)$$

where X – number of sessions before SBC implementation; Y – number of sessions after SBC implementation.

The relative loss reduction can be calculated as follows (3):

$$(\%) = ((L1 - L2)/L1) * 100, \quad (3)$$

where $L1$ – number of lost sessions before SBC implementation; $L2$ – number of lost sessions after SBC implementation.

When calculating bandwidth redundancy (4):

$$R(\%) = ((TB - UB)/TB) * 100), \quad (4)$$

where R – reservation; TB – total bandwidth; UB – used bandwidth.

In addition, to calculate network load (5):

$$L(\%) = ((TV/B) * 100), \quad (5)$$

where L – load; TV – traffic volume; B – bandwidth.

RESULTS

Evaluation and comparison of cloud SBC in the context of SIP trunking. Cloud-based SBCs can be compared by analysing call setup time, call quality score (MOS (Mean Opinion Score)), and resource usage and scalability score (Table 1).

Table 1. SBC cloud comparison

	Call setup time	Call quality assessment (MOS Score)	Evaluating resource usage and scalability
Amazon Chime Voice Connector	Fast	High	High
Twilio Elastic SIP Trunking	Fast	High	Good
Azure Communication Services	Fast	High	High
Dialogflow Telephony Gateway	Fast	Good	Good
Oracle Cloud Infrastructure	Fast	High	High

Source: compiled by the author

To get the call setup time, it is necessary to calculate the average time required to set up a call on each of the SBC clouds under consideration. The call quality score (MOS Score) is a widely used metric that can be calculated based on various factors such as jitter, subprime, delay, and other parameters (1). Calculating resource usage can be measured as the percentage of resources used during call processing. It is necessary to use formulas that will help calculate how easily you can expand the system by adding new resources to assess scalability.

Calculating key performance indicators is an equally important part of evaluating the benefits of using SIP trunking services. A possible approach to mathematical calculations to determine the improvements that SBC can provide includes performance improvements or loss reductions. Metrics such as the number of successful SIP sessions per unit of time before and after implementing SBC can be used to measure performance (2). Considering the number of lost SIP sessions before and after SBC implementation can be used to determine the impact of SBC on loss reduction (3).

Examples include Amazon Chime Voice Connector, Twilio Elastic SIP Trunking, Azure Communication Services, Dialogflow Telephony Gateway, and Oracle Cloud Infrastructure. Amazon Chime Voice Connector is a cloud-based SBC that provides the ability to connect internal IP phone systems to Amazon Chime cloud communication services. This provides fast call setup, high call quality, and the flexibility to scale resources. Twilio Elastic SIP Trunking allows connecting IP telephony to the global Twilio infrastructure using SIP trunking. It has fast call setup, high call quality, and good scalability. Azure Communication Services provides a cloud-based SBC for connecting IP telephony to Microsoft Teams and Azure Communication Services cloud services. Provides reliability, high call quality, and zoom support. Dialogflow Telephony Gateway is a cloud-based platform for developing and connecting voice bots to the telephone network via SIP trunking. It has fast call setup, good call quality, and advanced zoom capabilities. Oracle Cloud Infrastructure provides a cloud-based SBC for security and control of communication flows in a cloud environment. Provides fast call setup, high call quality, and scalability. Thus, among the listed cloud SBCs, Amazon Chime Voice Connector, Azure Communication Services, and Oracle Cloud Infrastructure are the most optimal. In general, this paper is considered on the basis of the following network elements: users, corporate IP network, cloud SBC, and the Internet (Fig. 1).

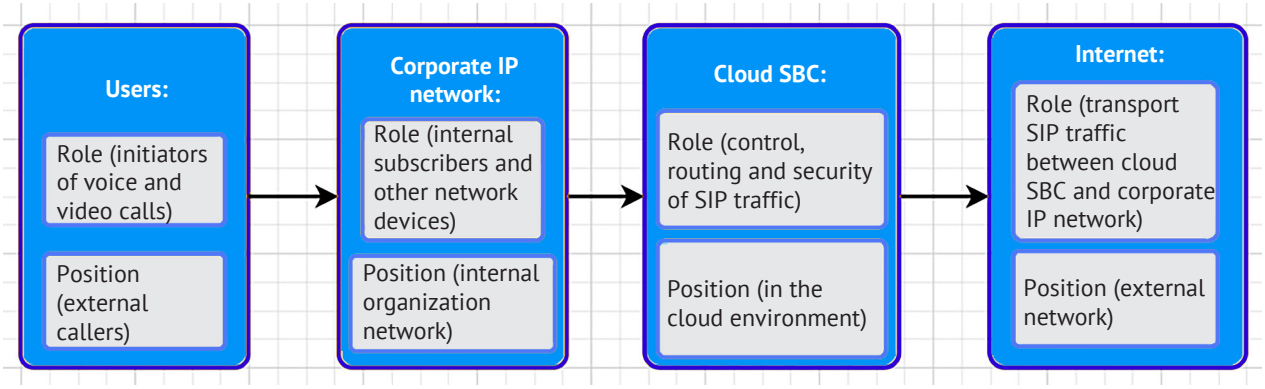

Figure 1. Block diagram of network elements

Source: compiled by the author

Users represent external subscribers who initiate voice and video calls. A corporate IP network indicates the organisation's internal network, where internal subscribers and other network devices are located. Cloud-based SBC is a session boundary controller that monitors, routes, and ensures the security of SIP traffic. The Internet is an external network through which SIP traffic is transported between the cloud-based SBC and the corporate IP network. Cloud-based SBC can perform the following functions: session monitoring (provides

management and control of voice and video traffic sessions); routing (defines routing paths for session traffic); security (performs security functions such as traffic filtering, attack protection, and encryption); transcoding (can perform media conversion); QoS (Quality of Service) (allows prioritising voice traffic to ensure high quality of service); signalling (provides exchange of alarm messages between session agents).

SBC cloud optimisation: the impact of network latency, bandwidth, and regional features on SIP trunking services: a comparative analysis and recommendations. Network latency affects the latency of data transmission between two points in the network. High network latency can lead to delays in the transmission of audio and video data, which can negatively affect the quality of the service. If there is a large network latency, real-time latency issues may occur, which may affect the quality of voice communication and other services that use SIP trunking. Network bandwidth is also an important factor for the performance of SBC clouds. It defines the maximum amount of data that the network can transmit over a certain period of time. Insufficient bandwidth can lead to network congestion and reduced service quality. In addition, high bandwidth can provide the ability to transfer more data simultaneously, which contributes to high performance.

The Amazon Chime Voice Connector may be sensitive to network latency, as the distance between users and Amazon servers may affect data latency. And with more bandwidth, the Amazon Chime Voice Connector can handle more simultaneous calls and deliver high performance. Network latency can affect the quality of calls via Twilio Elastic SIP Trunking. High latency can cause delays in voice transmission. High throughput is also important for Twilio Elastic SIP Trunking, as this service handles a large amount of traffic. Due to network latency, there may be a delay in processing calls through Azure Communication Services, especially if there is a long distance between Azure clients and servers. In addition, to ensure high performance, a large network bandwidth is important. Network latency can affect the delay in voice transmission via the Dialogflow Telephony Gateway. However, with its high bandwidth, Dialogflow Telephony Gateway can handle many simultaneous calls and deliver high performance. In Oracle Cloud Infrastructure, network latency can affect call delivery times, and high network bandwidth is important for ensuring Oracle Cloud Infrastructure performance. It is necessary to analyse the impact of network latency and bandwidth on this performance (Table 2) to identify the performance limitations of some cloud-based SBCs.

Table 2. Network latency and bandwidth of SBC clouds

	Network latency	Bandwidth
Amazon Chime Voice Connector	Sensitive	High
Twilio Elastic SIP Trunking	Affects call quality	High
Azure Communication Services	Affects delays	High
Dialogflow Telephony Gateway	Affects delays	High
Oracle Cloud Infrastructure	Affects delivery time	High

Source: compiled by the author

Providers with lower network latency and higher bandwidth (Amazon Chime Voice Connector and Twilio Elastic SIP Trunking) can be a good choice for organisations looking to provide high-quality voice communication and high performance. Cloud-based SBCs with higher network latency (Azure Communication Services and Dialogflow Telephony Gateway) can also be effective, but they require additional attention to the network infrastructure to reduce latency. Oracle Cloud Infrastructure can be an effective choice for organisations that value high performance and are willing to optimise their network to compensate for delays.

When identifying performance limitations, SBC configuration, and optimisation problems and the impact of geographical distribution on the provision of SIP trunking services are also justified. It is parameters such as network latency and bandwidth that can limit the performance of the current SBC configuration. In addition, these criteria can be supplemented by security, processor, and memory. If the current SBC configuration does not meet the bandwidth requirements of a particular network or standards, this may lead to performance limitations. In this case, considering increasing SBC throughput by configuring hardware or software resources, or using more powerful hardware is necessary. If the network latency in the current SBC configuration is high, this may affect performance.

Latency may depend on the location of the SBC, network quality, and other factors. Optimising network configuration is advised to improve performance, for example by using optimal routes, increasing network throughput, or using latency reduction technologies such as QoS or WAN (Wide Area Network) optimisation. In addition, incorrect SBC security configuration can lead to performance limitations. For example, insufficient protection against attacks or incorrect security settings can lead to delays or data loss. It is crucial to review the current SBC security configuration to make sure that it meets the requirements of this network and security standards. If the current hardware running SBC

has limited CPU or memory capacity, this may affect performance. Consider upgrading the hardware or configuring SBC so that it uses resources more efficiently is necessary in this case.

Optimisation of SBC can include various aspects, depending on the specific provider or platform. For the Amazon Chime Voice Connector, bandwidth configuration and network optimisation are recommended. For Twilio Elastic SIP Trunking – correct SIP Trunking configuration and network optimisation. For Azure Communication Services – server placement and monitoring. For Dialogflow Telephony Gateway – checking the phone gateway settings and network optimisation. For Oracle Cloud Infrastructure – hardware resources and security configuration. The provision of SIP trunking services may vary on each continent and in different countries. However, a general comparative analysis can be performed for countries in North America, South America, Europe, Asia, Africa, Australia, and Oceania (Fig. 2).

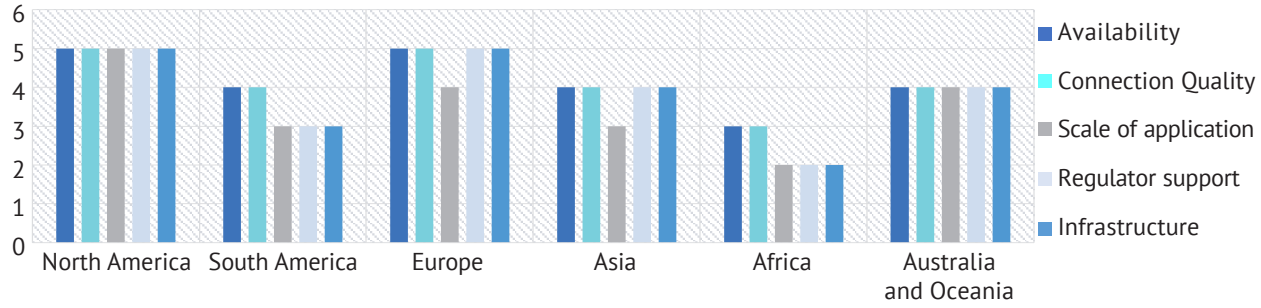

Figure 2. Comparative analysis of SIP trunking services in different regions
Source: compiled by the author

North America has a fairly developed communication infrastructure and high availability of SIP trunking services. Many providers and carriers offer extensive capabilities, and popular SIP trunking systems are available in many regions, especially in the United States and Canada. South America has different levels of availability and development of SIP trunking services. There are countries where access to SIP trunking services may be limited or where the infrastructure is not as developed as in North American regions. However, in some South American countries, such as Brazil, Argentina, and Colombia, well-developed networks and SIP trunking providers that offer high-quality services are also present. Europe also has a well-developed network infrastructure and availability of SIP trunking services. Most European Union countries have a regulatory framework that promotes the expansion of SIP trunking services. In addition, many European countries have a number of local providers that provide these services. Asia is a broad region with diverse markets and varying levels of communication development. Some countries, such as Japan, South Korea, and Singapore, have very developed networks and a wide range of SIP trunking services. Other countries may have limited availability or regulatory restrictions.

The availability of SIP trunking services may be limited in many African countries, especially in less developed areas. However, the growth of internet infrastructure and the growing popularity of mobile communications are driving the development of SIP trunking services in some regions of Africa. Australia and some Oceania countries, such as New Zealand, have developed communication networks and the availability of SIP trunking services. The western part of Oceania may have a less developed infrastructure, and the availability of services may be limited. Based on this comparative analysis, it can be concluded that North America and Europe are the most developed regions for providing SIP trunking services, while Africa is the least developed region. The regions of Asia, Australia, and Oceania have their own characteristics on a case-by-case basis.

Optimising the quality of service in cloud-based SBC: implementing SIP traffic and resource scaling and allocation strategies. Implementing quality of service for prioritising SIP traffic involves setting priorities and service parameters for SIP voice traffic (Fig. 3). This helps ensure stable and high-quality voice communication on the network, even in the presence of other types of traffic.

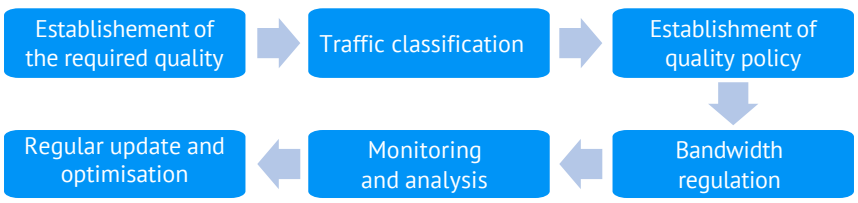

Figure 3. Sequence of actions for implementing the quality of SIP Traffic Service
Source: compiled by the author

Resource allocation and automatic scaling methods in cloud-based SBC are essential in ensuring efficient system operation and optimising its performance. Using the following methods and strategies is advisable: horizontal, vertical, and automatic scaling; load distribution; resource planning and redundancy; monitoring and analysis. Horizontal scaling involves adding new SBC instances to the system to distribute traffic. This is especially useful when the system exceeds its current bandwidth or requires more call processing. Horizontal scaling allows expanding system resources as the load increases. Vertical scaling involves extending resources to existing SBC servers. For example, adding additional processors or memory to an existing server to improve its performance. It can be useful for optimising performance in cases where more resources are required on a separate server. The auto-scaling strategy provides for automatic detection of changes in load and automatic scaling of resources according to the need. This can be achieved by using a variety of technologies, such as containers, orchestras, or cloud services that automatically respond to load changes and scale resources up or down.

Load distribution involves distributing traffic between different SBC instances to balance the load. It can be implemented at the network level, using load balancers, or at the application level, using various routing algorithms. The resource planning method can help avoid having to over-scale the system. It includes load analysis and forecasting to correctly determine how many resources need to be allocated for optimal SBC performance. When reserving resources, providing backup resources for emergencies is an important aspect of optimising cloud SBC. This allows the system to withstand the load in the event of a failure of some resources and ensures continuous operation. In addition, constant monitoring and analysis of SBC performance allows identifying problems and opportunities for optimisation in time. Monitoring also helps make decisions about scaling and resource allocation. The use of these methods and strategies helps to optimise the operation of cloud SBC, ensuring high quality of service and efficient use of resources. It is important to pay attention to redundancy and optimisation of network protocols by performing brief calculations (Formulae 4, 5). In the context of cloud-based SBC, network protocol redundancy and optimisation play a key role in ensuring the reliability, efficiency, and quality of service of SIP trunking. These strategies aim to ensure optimal use of network resources and improve performance.

Performance validation and testing: modelling and optimising SIP sessions. A simple programme is implemented using the Java language and the JAIN-SIP library to simulate test SIP sessions and generate traffic (Fig. 4). This code creates a single SIP session and sends an INVITE request to the server.

```
import javax.sip.*;
import javax.sip.address.*;
import javax.sip.header.*;
import javax.sip.message.*;
import java.util.*;

public class SipSessionExample {
 public static void main(String[] args) {
 try {
 // Creating a SIP factory and configuration
 SipFactory sipFactory = SipFactory.getInstance();
 sipFactory.setPathName("gov.nist");

 Properties properties = new Properties();
 properties.setProperty("javax.sip.IP_ADDRESS", "your_local_ip");
 properties.setProperty("javax.sip.STACK_NAME", "stack");
 SipStack sipStack = sipFactory.createSipStack(properties);

 ListeningPoint listeningPoint = sipStack.createListeningPoint("your_local_ip", 5060, "udp");
 SipProvider sipProvider = sipStack.createSipProvider(listeningPoint);
 sipProvider.addSipListener(new SipListenerImpl());

 // Creating a SIP factory for address and communication
 AddressFactory addressFactory = sipFactory.createAddressFactory();
 MessageFactory messageFactory = sipFactory.createMessageFactory();
 HeaderFactory headerFactory = sipFactory.createHeaderFactory();

 // Creating SIP addresses for communication
 Address fromAddress = addressFactory.createAddress("sip:caller@yourdomain.com");
 Address toAddress = addressFactory.createAddress("sip:callee@remote_domain.com");

 // Creating a sip request INVITE
 CallIdHeader callIdHeader = sipProvider.getNewCallId();
 CSeqHeader cSeqHeader = headerFactory.createCSeqHeader(1, Request.INVITE);
```

Continued Figure 4

```
MaxForwardsHeader maxForwards = headerFactory.createMaxForwardsHeader(70);
Request request = messageFactory.createRequest(
toAddress.getURI(),
Request.INVITE,
callIdHeader,
cSeqHeader,
fromAddress,
toAddress,
new ViaHeader(),
maxForwards
);

// Adding a SIP request to the queue
ClientTransaction clientTransaction = sipProvider.getNewClientTransaction(request);
clientTransaction.sendRequest();

} catch (Exception e) {
e.printStackTrace();
}
}

static class SipListenerImpl implements SipListener {
public void processRequest(RequestEvent requestEvent) {
// Processing incoming SIP requests
}

public void processResponse(ResponseEvent responseEvent) {
// Processing responses to SIP requests
}

public void processTimeout(TimeoutEvent timeoutEvent) {
// Processing timeouts
}

public void processIOException(IOExceptionEvent ioExceptionEvent) {
// Imput and output error handling
}

public void processTransactionTerminated(TransactionTerminatedEvent transactionTerminatedEvent) {
// Processing of completed transactions
}

public void processDialogTerminated(DialogTerminatedEvent dialogTerminatedEvent) {
// Processing completed dialogues
}
}
}
```

Figure 4. Implementation of SIP session simulation and traffic generation in Java with JAIN-SIP
Source: compiled by the author

This code creates a single SIP session and sends an INVITE request to the server.

The user can change addresses and other parameters to suit their needs. However, this is just a basic example, and to generate more sessions and process results, it is necessary to implement a more complex and larger programme. The code result may look as follows (Fig. 5).

```
Sending an INVITE request to sip.example.com...

Response from the server.
HTTP/1.1 200 OK
Server: MySIPServer
Content-Length: 256
...

INVITE request successfully processed. The session is established.
```

Figure 5. Programme result
Source: compiled by the author

The experimental results obtained confirm the high level of performance of cloud SBC and show that the optimisation strategies considered in this study can improve the quality of calls and ensure efficient use of resources.

As a real example, Microsoft can be considered (Fig. 6). It requires an SBC certification that combines Teams and a telephone network. The session restriction controller performs security, compatibility, connecting to legacy systems, migration, and emergency switching functions. Security acts as a firewall that recognises voice and encrypts traffic, protecting both the voice traffic itself and preventing intruders from entering the data network via the voice network. Since not all SIP trunking providers are identical, Ribbon SBC can configure key parameters to simplify and seamlessly establish communication. Many organisations still have analogue telephones, elevator telephones, door telephones, and fax machines. The company's SBC hardware has analogue ports, which allow maintaining the operation of these tools for many more years. Also, organisations often want to switch to Teams, leaving their outdated ATC for a long time. Ribbon's SBC can provide simultaneous call forwarding, allowing Teams and the old ATC to share connections. SBC data supports certain Microsoft hardware to provide basic call services, even if the Microsoft 365 cloud is not available. In addition, Ribbon supports backup network management and several other options for managing service interruptions, depending on the deployment model. These tools allow keeping a specific organisation connected, even in the event of unforeseen circumstances. Compare the network performance using SBC with and without optimised settings is also necessary (Table 3).

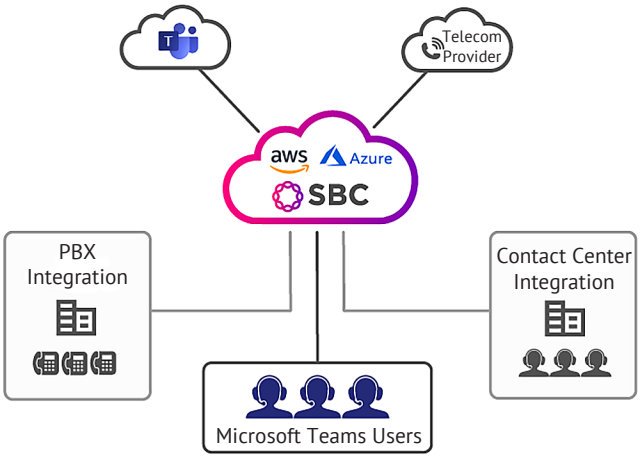

Figure 6. The role of SBC in direct routing
Source: Azure & AWS hosted SBC for Microsoft Teams (n.d.)

Table 3. Comparison table for network operation using SBC

Parameter/ Characteristics	Using SBC (with optimised settings)	Without using SBC
Security	Secures traffic and prevents intruders from entering the data network	Increased threats to voice traffic security
Compatibility	Easy setup and smooth operation with various SIP trunking providers	Requires more complex compatibility settings
Connecting to legacy systems	Support for Analogue Devices (faxes, analogue phones)	Limited connectivity to legacy systems
Migration	Support for simultaneous connection of Teams and the old ATC	Difficult migration and disabling of the old ATC
Emergency switching	Support for basic call services even if the Microsoft 365 cloud is unavailable	Lack of basic services in case of unavailability of the Microsoft 365 cloud

Source: compiled by the author

This table shows the advantages of using SBC with optimised settings over network operation without using SBC. It is SBC that provides increased security, ease of configuration and support for legacy systems, simplified migration and resilience in the event of a Microsoft 365 cloud failure. This comparative analysis is suitable not only for Microsoft, but also for cloud SBC of various corporations. It is worth giving examples of successful improvements in SBC performance and evaluating the impact of optimisation on call quality and resource usage. An example of successful SBC performance improvement can be divided into scenarios, actions and optimisation, and results. Scenario: a large company used cloud-based

SBC to handle a large volume of calls. As a result of performance analysis, it was identified that the system often reached its bandwidth, which led to call rejection and reduced service quality. Actions and optimisation: the team decided to optimise SBC using resource allocation methods and improved routing algorithms. Monitoring and automatic scaling have also been configured for more efficient use of resources. Results: after optimisation, SBC performance improved substantially. The number of rejected calls decreased, and the quality of service improved. The system has become more scalable and reliable.

Evaluation of the impact of optimisation on call quality and resource usage is also divided into scenarios, actions, optimisation, and results. Scenario: in this experiment, two SBC options are installed – with and without optimised settings. A series of calls were made, communication quality and resource usage were measured for both options. Actions and optimisation: the optimised settings included the use of QoS (Quality of Service) to prioritise SIP traffic, reserve bandwidth, and improve routing algorithms. Results: call quality assessment showed that the optimised version has fewer delays and higher audio quality. In addition, resource usage was optimised, and more calls could be processed simultaneously while maintaining a high quality of service.

Optimisation strategies and key practises for cloud SBC: selecting, configuring, and ensuring efficiency and security. When choosing cloud-based SBC, considering several main aspects, including SBC features such as security, compatibility, quality of service, and session routing (Fig. 7). Firstly, analysing one's own or company's needs and requirements is necessary. This means determining the amount of voice traffic, the number of simultaneous calls, and other parameters that are important for a particular organisation. Next, considering the quality of service factors and the geographical location of users and partners is necessary, as this may affect the effectiveness of the service. It is also important to evaluate the ability of cloud-based SBC to integrate with existing systems and applications.

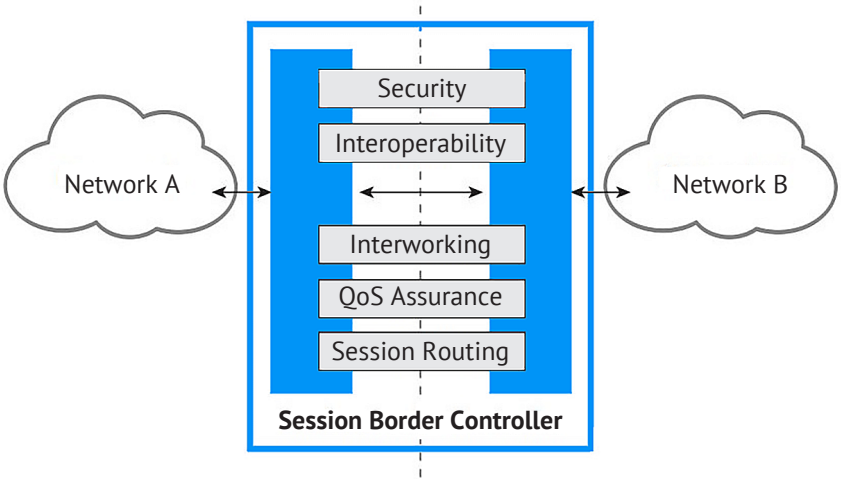

Figure 7. SBC functions

Source: What is a Session Border Controller (SBC)? (n.d.)

There are several strategies to achieve the best performance with a particular cloud-based SBC. It is necessary to prioritise processing different types of traffic, establish methods for efficient resource allocation, and consider scaling the system as needed. Setting up a cloud-based SBC with these aspects in mind can substantially improve performance and quality of service. Optimisation strategies for efficient SIP trunking services in cloud environments: requirements and needs analysis, cloud SBC selection, cloud SBC configuration, scalability planning, security, debugging, and improvement.

Analysis of requirements and needs. Examining the company's needs for processing voice traffic in the cloud is necessary. After that, it is important to determine the amount of voice traffic, the number of simultaneous calls, and other parameters that are important for the organisation. Choosing a cloud-based SBC. One should choose the cloud-based SBC that best meets the quality of service needs and requirements. After, it is critical to ensure that the selected cloud-based SBC supports QoS to prioritise voice traffic. Setting up a cloud-based SBC. Prioritising processing different types of traffic, such as voice, video, and data are important. Configuration of resource allocation methods, such as QoS is conducted afterwards to ensure efficient use of available bandwidth. Scalability planning. Considering horizontal (adding new servers) and vertical (increasing the capacity of existing servers) scaling options is recommended. After that, backup clusters or servers are configured to ensure high availability in the event of a failure. Ensuring security. Using traffic encryption to protect against unauthorised access and data interception is critical. This includes configuring firewalls, intrusion detection, and prevention systems (IDS/IPS)

to monitor and protect the network. Updating SBC software regularly is required to fix any vulnerabilities found. Debugging and improvement. It is necessary to constantly monitor and analyse the performance of the cloud SBC and improve the settings in accordance with changes in traffic needs and volume. This optimisation strategy provides an action plan to improve the delivery of SIP trunking services in cloud environments and can help achieve efficiency and scalability in this area.

Among the best practises for each step of the service optimisation strategy is to conduct a voice traffic audit to determine the bandwidth consumed and traffic patterns. It also involves analysing traffic priorities and determining which type of traffic (voice, video, data) requires the highest priority in the organisation. The next practise is to evaluate various SBC cloud providers based on their reputation, experience, and QoS support. Next, comparison of the scalability and high availability capabilities offered by various cloud-based SBC solutions is conducted. It also sets priorities for different types of traffic in the SBC cloud, considering the analysis results. Resource allocation mechanisms, such as QoS, are used to ensure efficient use of bandwidth. It is necessary to note such practises as exploring the possibility of horizontal and vertical scaling of the selected cloud SBC and choosing the one that best suits the specific needs. It is vital to consider scheduling redundancy and network infrastructure to ensure high availability. The next practise is to use modern traffic encryption methods to protect against unauthorised access. The latter can be distinguished by configuring firewalls and intrusion detection and prevention systems to effectively monitor and protect the network.

Thus, the optimisation strategy includes best practises for each step, which helps to achieve efficient and scalable SIP trunking services in cloud environments. Horizontal and vertical scaling, redundancy, and network infrastructure planning must be considered to ensure efficient scaling and high availability of cloud-based SBC. The company should choose the scaling strategy that best suits their specific organisation and consider backup options to ensure system reliability. Planning the network infrastructure so that it can support traffic growth when scaling is important. Ensuring security in cloud-based SBC includes the use of encryption, configuring firewalls and intrusion detection and prevention systems, and timely software updates. Traffic encryption is used to protect against unauthorised access and data interception. Firewalls and intrusion detection and prevention systems must be used to monitor and protect the network. In addition, regularly update SBC software to fix any detected vulnerabilities is also necessary.

DISCUSSION

F. Abdallah (2019) performed VoIP security analysis using the ZRTP (composed of Z and Real-time Transport Protocol) method on SIP trunking using SBC. Some attacks that can easily undermine VoIP security and various ways to establish VoIP security were investigated. The paper considered the use of security based on ZRTP, which is a modern and updated way to ensure security since it uses a key for security between communicating clients. A common aspect between this study and the one mentioned is that both consider the use of SIP trunking and SBC. However, the 2019 study focuses more on VoIP and ZRTP, while this study specifically reviews SBC and SIP trunking.

Author P. German (2016) emphasised that security cannot be maintained over time, as all security tools become vulnerable as threats evolve. Any security solution can have serious drawbacks. Therefore, successful organisations regularly update their antivirus and anti-malware software, strengthen the security of their infrastructure, and review their policies. It addresses SIP security issues based on a one-time session boundary controller (SBC) implementation. What the two studies have in common is that both of them consider SIP and SBC. However, a paper of P. German (2016) focuses more on the security of these technologies, and this study focuses on improving the provision of SIP trunking services.

A study by U. Nakorji *et al.* (2023) shows the vulnerability of VoLTE voice networks to attacks on the SIP session initiation protocol. DDoS attacks on SIP can include modifying SIP messages, which leads to a restart of SIP devices, or flooding devices with requests that cause resource exhaustion and denial to real users. A technique for detecting and preventing attacks on SIP registration with a high detection accuracy of more than 96.67% is proposed. Like the mentioned paper, this study focuses on the session initiation protocol and voice communication network. However, the above study uses VoLTE technology, and this one – cloud-based SBC. In addition, this study focuses more on analysing works and optimising services, while the other focuses more on countering DDoS attacks. P. German (2017) noted that organisations actively use session boundary controllers (SBC) to ensure the security of SIP trunking. However, most SBC's are considered not only one-time investments, but also one-time deployments. Despite this, SIP trunking has built-in vulnerabilities that constantly change and worsen, including denial-of-service attacks and call fraud. The author concluded that, in their current form, most SBCs create a false sense of security in various organisations.

In the paper of C. Yin *et al.* (2023), a system for evaluating the quality of cloud service composition is being created, including the degree of service compliance, composition synergy, and other metrics such as service time, cost, and reliability. A two-criterion composition selection model is then created, which is solved by an improved ant colony algorithm. It can be concluded that both studies describe services in cloud environments using different building blocks. However, the criteria for comparative analysis differ, as this study uses call time and quality, resource usage and scalability,

network latency and bandwidth, and another – the degree of compliance and synergy of service composition, service time and cost, and reliability.

Z. Hou *et al.* (2023) explore the session initiation protocol (SIP), which is one of the most popular multimedia communication protocols and has many interesting features, such as advanced and open communication. However, with the rapid development of this protocol, load-balancing issues and security risks have become apparent. This study uses sequential process modelling (CSP) to model SIP and the model analysis (PAT) tool to test the internal and secure properties of a given model. The test results show that SIP has a load-balancing problem and cannot guarantee data security in the event of malicious users. Therefore, the authors improved the model by adding a software-defined network (SDN) architecture and an in-process authentication mechanism. Based on the results of the new test, it becomes evident that the improved model can satisfy internal properties and has improved the security of the SIP model. Like this study, the above-mentioned paper focuses on the protocol of initiating sessions and ensuring their security. However, this study uses cloud-based SBC, while another – tools such as CSP, PAT, and SDN.

S. Wang *et al.* (2022) provide a comprehensive overview of analysis and optimisation performance for stochastic cloud service requests. Analysing the main entities and actions in general performance analysis procedures, a general performance analysis model that includes five main characteristics is proposed: query, sequence, queue, distribution, and services. Practical factors of each characteristic are analysed. The authors discuss the impact of each characteristic on optimisation goals, including costs, profit, response time, and energy consumption. Both studies focus on analysing and optimising service on cloud platforms. However, unlike this study, the 2022 study does not use session initiation protocol trunking and cloud-based session boundary controllers.

In turn, V. Kumar and O. Roy (2020) review security issues in the VoIP voice internet protocol system and reviews several security attacks. For communication between nodes in a VoIP network, a SIP signal protocol is used. The authors continue to review all issues that have a substantial impact on VoIP security and issues, such as data manipulation, replacement attacks, power supply, jitter, and latency. Both studies consider the SIP protocol. However, the 2020 study focuses more on protocol security, and this on more – on improving services using these protocols.

Analysing all the listed papers and comparing them with this one, several key aspects that make this study unique and important in the context of research in the field of cloud technologies can be identified. Firstly, this study tried to cover a wider range of aspects related to the quality of services in cloud systems, including performance analysis, optimisation, security, and sustainability. Compared to the individual aspects discussed in other studies, this paper has provided more completeness and depth in understanding problems. Secondly, SIP trunking and cloud session boundary controllers (SBC) were actively used in this study, which can be an important aspect in the context of improving the quality of Service and security of telecommunications technologies on cloud platforms. This allows submitting new solutions and approaches to improve the quality of communication. Thirdly, the study actively discusses methods for optimising the quality of service in cloud systems, considering various factors such as response time, costs, and energy consumption. This can be useful for practical applications and businesses, as improving productivity and reducing costs are always important. Fourthly, the study evaluated security aspects in telecommunications and cloud systems, considering possible attacks and methods for detecting and preventing them. This makes the study more comprehensive in the form of a discussion of security measures. Lastly, this study actively includes comparative analysis with the work of other researchers, which helped identify common aspects and differences in the papers. This provides a deeper understanding of the context and importance of the work. Overall, this study can make an important contribution to improving the listed technologies and the quality of user communication.

CONCLUSIONS

This study was aimed at analysing the performance of SBC clouds, identifying possible limitations, and proposing optimisation strategies to improve the overall performance, reliability, and user experience of SIP trunking services. Performance research and development of optimisation strategies are important aspects when providing SIP trunking services using cloud-based SBC. This subject is quite relevant in the modern telecommunications industry, as it allows increasing the efficiency and scalability of trunking services in cloud environments. It also allows discovering the benefits and solving problems associated with using cloud-based SBC, helping to improve the user experience and ensuring network reliability. In general, cloud-based SBCs are solutions that ensure the security and control of communication flows in VoIP (voicemail over the Internet) networks. Cloud-based SBC allows connecting IP phone networks to various cloud services, and their main goal is to provide reliable, secure, and efficient routing and processing of session traffic in voice and video communication networks that use the SIP protocol.

Certain cloud-based SBCs, which are quite common were used to implement the study. Practical experiments consisted of modelling SIP test sessions, graphs, diagrams, tables, formulae, and a general application. Based on the investigation and its results, recommendations were made for improving the SBC cloud architecture, using optimisation strategies, exploring alternative solutions, expanding research, practical implementation, monitoring, and support. It is worth

implementing effective monitoring and logging systems to quickly identify and solve problems. It is important to support users and operators in solving problems and providing the necessary assistance. The results of the study are of practical importance for developers and operators of telecommunications systems and cloud services. The recommendations obtained can be used to improve the efficiency and safety of various systems. Given the complexity and rapid development of technologies, an important aspect is the possibility of further research in this area in according to the dynamics of the emergence of new technologies.

Possible areas of further research and development may consist of expanding the functionality of SBC, data security and protection, the use of artificial intelligence and data analysis, international aspects and global networks, energy efficiency, standardisation and regulation, integration with other technologies. Exploring opportunities to further expand the functionality of cloud-based SBC, such as integration with other cloud services, video traffic analysis, WebRTC support, etc., is necessary.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Abdallah, F. (2019). *VoIP security analysis using the ZRTP method in SIP trunking which uses SBC*. Retrieved from https://repository.mercubuana.ac.id/67267/.

[2] Al Saidat, M.R.S. (2019). A design of an enhanced redundant SIP model for securing SIP-based networks. In *16th international conference on computer systems and applications (AICCSA)* (pp. 1-6). Abu Dhabi: IEEE. doi: 10.1109/AICCSA47632.2019.9035304.

[3] Azure & AWS hosted SBC for Microsoft Teams. (n.d.). Retrieved from https://ribboncommunications.com/solutions/enterprise-solutions/microsoft-solutions/azure-aws-hosted-sbc-microsoft-teams.

[4] Bondarchuk, A., & Zhebka, V. (2023). Protection of a heterogeneous telecommunication network from the influence of destabilizing factors. *Telecommunication and Information Technologies*, 78(1), 4-16. doi: 10.31673/2412-4338.2023.010416.

[5] German, P. (2016). The SIP security fallacy. *Network Security*, 2016(6), 18-20. doi:10.1016/S1353-4858(16)30060-5.

[6] German, P. (2017). Is your Session Border Controller providing a false sense of security? *Network Security*, 2017(1), 14-16. doi: 10.1016/S1353-4858(17)30007-7.

[7] Hou, Z., Jiaqi, Y., Zhu, H., & Chen, N. (2023). Formalization and verification of SIP using CSP. In H. Takizawa, H. Shen, T. Hanawa, P. Hyuk, J. Park, H. Tian & R. Egawa (Eds.), *Parallel and distributed computing, applications and technologies* (pp. 158-171). Cham: Springer. doi: 10.1007/978-3-031-29927-8_13.

[8] Kaul, S., & Jain, A. (2019). Study on the future of enterprise communication by cloud session border controllers (SBC). In G. Singh Tomar, N.S. Chaudhari, J.L.V. Barbosa & M.K. Aghwariya (Eds.), *International conference on intelligent computing and smart communication* (pp. 407-414). Singapore: Springer. doi: 10.1007/978-981-15-0633-8_39.

[9] Kumar, V., & Roy, O. (2020). Security and challenges in voice over internet protocols: A survey. *IOP Conference Series: Materials Science and Engineering*, 1020, article number 012020. doi: 10.1088/1757-899X/1020/1/012020.

[10] Mazharenko, V. (2020). *Study of the peculiarities of the implementation of IP telephony services in Internet resources*. (Master's thesis, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine).

[11] Nakorji, U., Bello-Salau, H., Adedokun, E., Mustafa, M., & Salami, O.W. (2023). Detection and prevention of SIP register injection attack on a VoLTE network. *Journal of Science, Engineering and Technology*, 2(1), 39-44. doi: 10.59568/KJSET-2023-2-1-06.

[12] Praveen, A., Kumar, S., & Rajendran, S. (2022). Automated testing of session border controller in simulated production environment. *International Journal for Research in Applied Science and Engineering Technology*, 10(6), 4608-4614. doi: 10.22214/ijraset.2022.45035.

[13] Romaniuk, P.I. (2023). Cloud technologies: Analysis, perspectives, implementations. *Computer-Integrated Technologies: Education, Science, Production*, 50, 108-113 doi: 10.36910/6775-2524-0560-2023-50-16.

[14] Skulysh, M., Romanov, O., & Globa, L. (2018). Principle of flows service in the heterogeneous telecommunication environment. *Academic Notes of TNU named after V.I. Vernadskyi*, 29(2), 92-97.

[15] Smirnova, T., Motorin, Y., Buravchenko, K., Bochulia, T., & Kovalenko, O. (2022). Selection of optimal technology of construction of cloud information and communication system of automation of production processes. *Measuring and Computing Devices in Technological Processes*, 1, 15-26. doi: 10.31891/2219-9365-2022-69-1-3.

[16] Tardaskina, T.M., & Hryshchuk, T.V. (2021). Features of use and development of cloud technologies in the digital economy. Biznesinform, 3, 254-260. doi: 10.32983/2222-4459-2021-3-254-260.

[17] Tas, I., Unsalver, B., & Baktir, S. (2020). A novel SIP based distributed reflection denial-of-service attack and an effective defense mechanism. *IEEE Access*, 8, 112574-112584. doi: 10.1109/ACCESS.2020.3001688.

[18] Tokar, L. (2022). Features of building virtual PBX. *Radiotekhnika*, 1(208), 55-64. doi: 10.30837/rt.2022.1.208.06.
[19] Wang, S., Li, X., Sheng, Q., & Beheshti, A. (2022). Performance analysis and optimization on scheduling stochastic cloud service requests: A survey. *IEEE Transactions on Network and Service Management*, 19(3), 3587-3602. doi: 10.1109/TNSM.2022.3181145.
[20] What is a Session Border Controller (SBC)? (n.d.). Retrieved from https://ribboncommunications.com/company/get-help/glossary/session-border-controller-sbc.
[21] Yin, C., Li, S., & Li, X. (2023). A matching-synergy degree-based optimization selection method for cloud manufacturing service composition. *The International Journal of Advanced Manufacturing Technology*. doi: 10.21203/rs.3.rs-2566065/v1.

Покращення надання послуг sip-транкінг за допомогою хмарних SBC: аналіз продуктивності та стратегії оптимізації

Олександр Іванович Підпалий

Аспірант
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»
03056, пр. Берестейський, 37, м. Київ, Україна
https://orcid.org/0009-0007-6852-7959

Анотація. У сучасних телекомунікаційних мережах транкінг протоколу ініціації сесій є важливою технологією для голосових та відеокомунікацій через мережі Інтернет-протоколу. Хмарні контролери кордонів сесій грають ключову роль у забезпеченні безпеки та якості обслуговування, але виникають виклики щодо їхньої продуктивності та оптимізації, особливо при зростанні обсягу даних та користувачів. Метою статті є розроблення стратегії для досягнення ефективних та масштабованих послуг транкінгу протоколу ініціації сесій в хмарних середовищах. Були використані методи теоретичного аналізу та практичних експериментів. Аналіз продуктивності хмарних контролерів розкрив важливі аналітичні висновки стосовно їх можливостей, а розроблені стратегії оптимізації надали практичні рекомендації для поліпшення надання послуг хмарного транкінгу. Даний аналіз також показав, що продуктивність може варіюватися в залежності від реалізації та налаштувань. Виявлено, що основними факторами, що впливають на продуктивність, є мережева латентність, пропускна спроможність та конфігураційні особливості контролерів. На основі цих відомостей, розроблено стратегію оптимізації, включаючи реалізацію якості обслуговування для пріоритизації трафіку протоколу ініціації сесій, методи автоматичного масштабування ресурсів, резервування та оптимізацію мережевих протоколів. Було продемонстровано, що використання Session Initiation Protocol-транкінгу та хмарних Session Border Controller може значно покращити якість обслуговування в телекомунікаційних системах, що дає можливість оптимізувати маршрутизацію дзвінків та забезпечити безпеку комунікацій. Оцінка продуктивності хмарних Session Border Controller надала цінне уявлення про їхні можливості, а пропоновані стратегії оптимізації можуть стати конкретними рекомендаціями для поліпшення надання послуг Session Initiation Protocol-транкінгу в хмарних середовищах. Результати дослідження можуть служити цінними рекомендаціями для фахівців у галузі телекомунікацій, які прагнуть розгортати ефективні та масштабовані рішення транкінгу в хмарних мережах

Ключові слова: протокол ініціації сесій; якість обслуговування комунікацій; контролери кордонів сесій; мережеві затримки; скалабельність ресурсів

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 621.791.7:004.932.046.78
DOI: 10.62660/2306-4412.4.2023.50-58

Features of application in critical structures of products made by fused deposition using 3D printing technology

Dmytro Tychkov*

Researcher
State Scientific Research Institute of Armament and Military Equipment Testing and Certification
18000, 164 Vyacheslav Chornovil Str., Cherkasy, Ukraine
https://orcid.org/0000-0003-1741-9329

Yuliia Bondarenko

PhD in Technical Sciences, Professor, Leading Researcher
State Scientific Research Institute of Armament and Military Equipment Testing and Certification
18000, 164 Vyacheslav Chornovil Str., Cherkasy, Ukraine
https://orcid.org/0000-0002-5179-8329

Vyacheslav Tuz

PhD in Technical Sciences, Associate Professor
Cherkasy State Technological University
18006, 460 Shevchenko Blvd., Cherkasy, Ukraine
https://orcid.org/0000-0003-2969-5080

Serafym Sapozhnikov

Researcher
State Scientific Research Institute of Armament and Military Equipment Testing and Certification
18000, 164 Vyacheslav Chornovil Str., Cherkasy, Ukraine
https://orcid.org/0000-0003-4653-203X

Daria Shapovalova

Junior researcher – engineer
State Scientific Research Institute of Armament and Military Equipment Testing and Certification
18000, 164 Vyacheslav Chornovil Str., Cherkasy, Ukraine
https://orcid.org/0000-0001-9161-1103

Abstract. The technology of hot-melt 3D printing has prospects for military equipment and other special applications, provided that necessary requirements for the quality, strength and durability of plastic components are met. However, there is a problem of insufficient accuracy and unexplored patterns of change in technological parameters during 3D printing, which makes it necessary to manufacture critical structures using plasma

Article's History: Received: 01.09.2023; Revised: 08.11.2023; Accepted: 18.12.2023.

Suggested Citation:
Tychkov, D., Bondarenko, Yu., Tuz, V., Sapozhnikov, S., & Shapovalova, D. (2023). Features of application in critical structures of products made by fused deposition using 3D printing technology. *Bulletin of Cherkasy State Technological University,* 28(4), 50-58. doi: 10.62660/2306-4412.4.2023.50-58.

*Corresponding author

deposition technology. The main purpose of the article is to study and evaluate the features of using the fused deposition method in 3D printing technology for the manufacture of products that are critical special-purpose structures and require increased strength and durability. To achieve this purpose, the technology and means of 3D printing by fused deposition method (filament feed control system with a circular encoder) have been improved. It has been established that the use of the filament feed control system can reduce the number of visually detected defects on the surface of printed products by an average of 71.7%, but increase the amount of wire consumed by 13% and the average printing time by 15-17%, which is due to the lack of supporting structures for the resulting surfaces. The research conducted in this paper has shown an increase in delamination strength of elements of critical structures, such as brackets for mounting a laser rangefinder to the body of an unmanned aerial vehicle. With the help of the filament feed control system, the strength of these elements has increased from 27.4 (tensile) and 32.1 (compressive) to 38 (tensile) and 44.3 MPa (compressive). There is also a more than 2.7-fold increase in the number of dynamic load cycles during endurance tests. This study indicates greater dynamic stability and less fatigue of the elements manufactured by fused 3D printing using a filament feed control system. The results will have practical applications in various fields, including military applications

Keywords: additive technology; print quality; strength; endurance

INTRODUCTION

Gradual growth of product quality requirements is becoming an integral part of a competitive market. This relevance, however, is gaining new dimensions due to the use of non-standard technical and technological solutions that are becoming more common in modern manufacture of products. These non-standard solutions cannot always be effectively implemented using traditional methods and technologies. There is a tendency to use innovative approaches, such as 3D printing and other advanced technologies. These methods open up great opportunities for creating unique and complex products, but at the same time pose new challenges for manufacturers in terms of researching and optimising process parameters. Despite significant progress in this area, important patterns of change in technological parameters of the printing process and their impact on product quality and mechanical and operational characteristics remain insufficiently understood.

Traditional technologies are increasingly replaced by additive technologies and rapid prototyping methods that can be used for mass manufacture of products without the need for complex production planning and specialised equipment, which greatly simplifies the process of product formation due to more flexible approaches to production management, which has been written about by B.A. Praveena *et al.* (2022) and G. Prashar *et al.* (2023). Authors I. Gibson *et al.* (2021) noted that 3D printing technologies are increasingly being used in various fields of science and technology. For example, the introduction of such technologies in modern production allows to manufacture products of varying complexity and configuration, including fragments of various energy, aviation, rocket and space, and military equipment (e.g., quadcopters, unmanned vehicles, etc.) operating under high climatic and mechanical loads, according to T.D. Ngo *et al.* (2018). Existing 3D printing technologies often do not meet high accuracy requirements for products, especially those that are components of critical structures. This is due to the limited research, rapid prototyping technologies, and lack of information on methods for controlling the accuracy of parts during their manufacture, as pointed out by J. Kiendl and G. Chao (2020).

In general, despite the prevalence of 3D printing methods and tools that have been developed recently, the study of their performance in the manufacture of critical products remains relevant to many leading world-class scientists. For example, a team under the guidance of J. Huang (Huang *et al.,* 2022) proposed an algorithm for implementing a unidirectional slicing strategy that slices the product model in the Z-axis direction, perpendicular to the 3D printer platform. This algorithm, although simple to implement, requires a significant amount of time to complete. Due to the disadvantages of this algorithm, which also include high computational complexity and inaccuracy, it is proposed to use a direct slicing strategy using analytical models. This strategy slices the model without converting it from a surface model to a mesh one. This can improve the accuracy of cut contours and correct the shortcomings of the mesh model. These algorithms usually require the support of a commercial software kernel for computer modelling, as written about in the works of such researchers as R.P. Ferreira and A. Scotti (2021), F. Wang *et al.* (2021) and S. Sagar *et al.* (2021).

Despite all the achievements and clear progress of this technology, there are currently a number of problems that limit the use of 3D printing technologies for the manufacture of high-precision and reliable products of critical structures. These problems include the lack of approaches to maintaining technological quality of moulded surfaces due to the lack of feedback in most 3D printing tools to determine the actual consumption

of consumables, which can lead to a decrease in printing quality and the emergence of a number of defects (microcracks, non-compliance with the thickness of product elements, etc.) and, as a result, deterioration of mechanical performance of these products, according to A. Jennings (2022).

For this reason, solving the issue of improving such characteristics in critical structures of products manufactured by fused deposition using 3D printing technology is an urgent task to which this article is devoted. To achieve this goal, the article solves the following tasks:

- to analyse existing solutions among the methods and systems of fused 3D printing of critical products for military equipment and other special-purpose applications;
- to compare the results of the proposed method of product printing using an improved filament feed control system and the method of traditional fused deposition by 3D printing;
- to test the products obtained using different technologies for compliance with their mechanical strength and endurance under dynamic loads.

The main purpose of this article was to study and evaluate the features of applying the fused deposition method in 3D printing technology for the manufacture of products of critical special-purpose structures that meet increased requirements for their high strength and durability. Scientific novelty of this work is that the technology and means of 3D printing by fused deposition method, in which a filament feed control system, including a circular encoder, is used, and the use of which can significantly improve the quality of 3D printing of products, increase their mechanical strength and endurance to external dynamic loads, have been improved.

MATERIALS AND METHODS

Technological experiment to obtain prototypes of products was carried out on a fusion 3D printer developed and improved with the participation of the authors in the laboratory of additive technologies in instrumentation, created on the basis of the department of instrumentation of mechatronics and computerised technologies of Cherkasy State Technological University. Testing and verification of the results obtained for the selected printing modes were carried out on the Ender-3 V3 SE device (Creality, China) at the State Scientific Research Institute of Armament and Military Equipment Testing and Certification. Experimental verification of research results with the aim of their comparison was carried out using both proposed technologies (traditional one and authors' technology using the filament feed control system). Polylactic acid (PLA) material was used to print the products. The average feed rate of a wire with a diameter 1.75 mm was 3.0 (for traditional technology) and 1.8 mm/s (for the authors' technology), and the extruder movement speed was 30 mm/s. The data obtained were calculated using 3D printing software, Cura (Vijay *et al.*, 2022). The printing was performed with a layer of 0.6 mm, with 20% filling without supports.

Analysing the available solutions among the methods and systems for fused deposition of 3D printing, it should be noted that methods such as FDM (Fused Deposition Method) or FFF (Fused Filament Fabrication Method) are very common in the additive manufacturing industry. This is due to their ease of equipment modification, affordability, wide range of consumables, and the possibility of quick training, according to T.S. Xiong *et al.* (2022). When manufacturing products with a complex geometric structure or there is a need to meet special reliability requirements (especially for components of critical structures) using FDM method, some defects may occur, for example, the effect of segmented texture (Fig. 1). This problem manifests itself when the angle of inclination of the surface relative to the printing plane decreases, and the formed surface becomes less smooth.

Figure 1. Exterior view of the product (Boeing Insitu ScanEagle (USA) UAV compressor bushing) printed by fused deposition using 3D printing technology with a distinct segmented texture effect

Source: compiled by the authors

In addition to the aesthetic problem, the effect of segmented texture also negatively affects mechanical properties of products, such as strength, resistance to harsh environmental conditions, aerodynamics, resistance to perpendicular loads, and other aspects. To solve this issue, a reduction in the height of the vertical layer can be used, but this approach leads to an exponential increase in printing time. Therefore, another method is proposed, which is also considered to be one of promising areas for the development of FDM technology in general – a method of manufacturing curved layers, known as Curved-Layer Manufacturing (CLM) (Cao *et al.*, 2023). Compared to the standard fused deposition process, in which parts are manufactured layer by layer, curved layer method allows to avoid the effect of segmented texture on surfaces lying close to the horizontal by printing these areas using continuous solid layers (Fig. 2).

Figure 2. Exterior view of the product (Boeing Insitu ScanEagle (USA) UAV compressor bushing) printed by the curved layer method using solid layer 3D printing technology
Source: compiled by the authors

The method of non-planar 3D printing allows to reduce or even eliminate the need to print supporting structures necessary for the construction of overhanging parts of the product, which leads to additional material and time consumption for the manufacture of parts. To improve operational and mechanical (strength) characteristics of products manufactured by the fused deposition method using 3D printing technology, it is proposed to use a filament feed control system, the exterior view of which is shown in Figure 3.

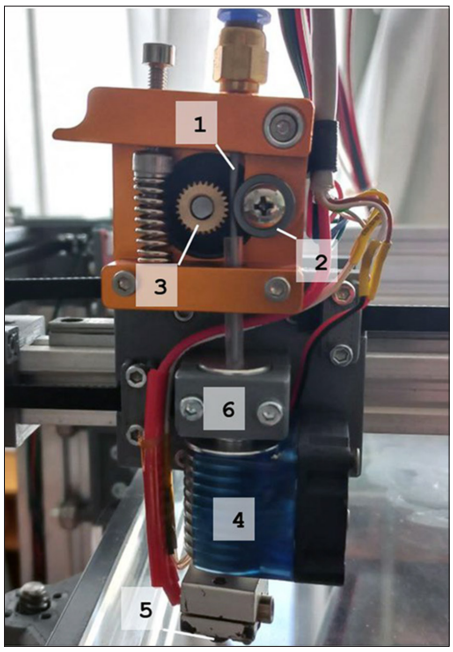

Figure 3. Exterior view of the filament feed control system together with the extruder
Notes: 1 – filament, 2 – measuring rollers, 3 – toothed filament feed rollers, 4 – heating unit, 5 – nozzle, 6 – circular encoder mechanism
Source: compiled by the authors

The principle of operation of the control system is as follows: filament 1 is fed from the bay (not shown in the figure) through special toothed rollers that act as measuring elements and activate the circular encoder mechanism 6 of the filament flow metring unit. When a command is received to start 3D printing on the device, the filament preheated by the heating unit 4 is directed through the nozzle 5 to the area where the 3D product is formed using the feed motor (not shown). Simultaneously with the start of the feed motor, toothed filament feed rollers 3 and measuring rollers 2 start feeding the filament 1 through the measuring device. If the filament slips in feeding devices into the heating and printing zone, a reduction in the required filament volume is taken into account using encoder 6. In this case, the control programme recalculates to reduce the filament feed rate, which ensures the required operational and mechanical characteristics of elements of critical structures.

RESULTS AND DISCUSSION

The strength and durability of the obtained products under dynamic loading were evaluated by testing the developed device in real conditions. For this purpose, the consumption metring mechanism of consumable material – filament – was proposed and tested (based on the developed automatic control system (Andriienko *et al.*, 2019)). This mechanism made it possible to control the supply of consumables to the working area with high accuracy (over 95%) and speed (the reaction time to a change in the wire feed rate is 250-480 ms). A comparison of the results of the proposed method of printing the product using the filament feed control system and the method of traditional fused deposition by 3D printing is shown in Table 1.

Table 1. Comparison of printing process parameters using the proposed technology with the filament feed control system and traditional technology

Printing process settings	By technology using the filament feed control system	Using traditional technology
Number of samples / printed surface area, m^2	12/0.34	12/0.34
Number of visually detected surface defects	61	216
Length of wire consumed, m	230.8	201
Weight of consumed material, kg	0.292	0.247
Printing time, min	128.2	111.6

Source: compiled by the authors

Table 1 shows that with the use of the control system, the number of visually detected defects decreases by 71.7%, from 216 to 61. At the same time, the amount of wasted wire increases by 13%, which for these products amounts to 247 (for products manufactured with traditional technology) and 292 grams (for products manufactured using the technology with a filament feed control system). Since there are no supporting structures for resulting product surfaces, the total length of printing segments (and, consequently, the wire used) also increases, which leads to an increase in the average printing time by 15-17%.

Figure 4 shows the results of test printing of products of critical structures, namely mounting brackets made of PLA plastic, obtained at a print speed of 60 mm/s using a 0.2 mm nozzle. When using the filament feed control system (Fig. 4b) with a 0.2 mm nozzle and a higher printing speed (80 mm/s), the quality of the solid surface is much better, and printing artefacts observed when using traditional technology are almost completely eliminated. The authors suggest that obtaining high-quality and reliable products is possible with a further increase in printing speed under conditions of controlled filament supply and adjustment of its operating temperature. The decrease in the quality of printing the edges of a product with a sharply defined geometry in 3D printing using traditional technology is associated with the occurrence of internal mechanical stresses due to faster cooling of the filament (associated with a lower printing speed) and, as a result, peeling and cracking of the applied solid material. In contrast to traditional technologies, 3D printing with the filament feed control system, which is carried out at a 30-35% higher speed, provides better adhesion between the layers of the applied material and, thus, greater adhesion strength and fewer internal and surface defects.

a  b 

Figure 4. Exterior view of samples (brackets for mounting the laser rangefinder to the UAV body)
Notes: a – printed using traditional technology, b – printed using the technology with the filament feed control system
Source: compiled by the authors

The authors D. Chakraborty *et al.* (2022) presented a surface-based parametric tool generation method for rapid prototyping technology, known as layer-by-layer modelling with curved layer deposition. This method allows the deposition of curved layers instead of traditional flat layers in FDM (Fused Deposition Method), which can improve the quality of thin curved parts. However, this method, according to the authors, is developed mainly for thin parts, and the algorithms presented in this paper are theoretical. Researchers Y. Li *et al.* (2022) used the method of slicing with curved layers to preserve the features of a complex surface. Genetic algorithms that passed curved layers through clusters of object features to reduce the number of curved layers were proposed. This method is mainly suitable for thin parts and hollow shells, so no experiments were conducted with products. Thus, in contrast to the applied method of 3D printing using the filament feed control system, the use of the methods proposed above is impractical due to practical unconfirmation of the results of the proposed solutions, as well as the lack of information (even theoretical one) on how to improve mechanical characteristics of parts printed using the proposed technologies.

Mechanical characteristics of parts (in particular, strength and endurance to dynamic mechanical loads) manufactured using the fused deposition method by 3D printing were investigated by N. Elmrabet and P. Siegkas (2020), M. Algarni (2021) and M. Doshi *et al.* (2022). At the same time, studies on the compliance with tensile and compressive strength of the printed product according to the methodology proposed by J. Sedlak *et al.* (2023) showed the following. According to classical technology, the delamination strength of bracket samples was 27.4 (tensile) and 32.1 MPa (compressive), while according to the technology using the filament feed control system, these values were 38 and 44.3 MPa, respectively, and approached the ultimate tensile and compressive strength of a solid material (for PLA plastics, the latter are 50-70 and about 60 MPa, respectively), which was a positive result. Experimental results indicate a significant improvement in mechanical properties of such parts compared to parts made by layer-by-layer surfacing.

When these products were tested for endurance under dynamic mechanical loads at load amplitude of 30 MPa and cyclic frequency of 5 Hz, the following was established. The first signs of fracture (local delamination and cracking in the product) appeared on the samples of brackets produced by traditional 3D printing technology after 4.680 load cycles, while the samples produced by the technology proposed in this paper withstood 12.960 cycles before the appearance of signs of fracture, which proves greater dynamic stability and lower fatigue of the products manufactured by this technology.

The hot melt technology of 3D printing opens up opportunities for its use in military equipment and other specialised industries, provided that it meets the required standards of quality, strength and durability of plastic parts. At the same time, there is a problem of insufficient accuracy and unexplored patterns of change in technological parameters during 3D printing. This problem can be solved by creating critical structures using plasma deposition technology. One way to overcome these difficulties is to conduct a detailed study and optimise printing process parameters to achieve the highest quality and efficiency of manufactured products. Such an approach will simultaneously expand the scope of the technology and ensure its successful use in highly demanding military and special applications.

Identified positive aspects can be applied to improve technical characteristics of other products where it is important to ensure dynamic stability and durability. The results of the study can become the basis for further research aimed at improving and optimising 3D printing processes to increase the productivity of manufacturing high-quality parts. It should be borne in mind that the implementation of the findings can contribute to the development of innovative solutions in manufacturing, in particular in the field of new materials and technologies. Thus, this study opens up wide opportunities for the use of advanced approaches in the production and development of high-tech products.

CONCLUSIONS

The article analyses the existing solutions among the methods and systems of fused 3D printing of critical structures for military equipment and other special-purpose applications, which shows the prospects of using this technology provided that necessary requirements for manufacturing quality, mechanical strength, and endurance to external dynamic loads of elements of critical plastic structures are met. It has been established that with the application of the technology using the filament feed control system, the number of visually detectable defects on the surface of the printed product is reduced by an average of 71.7%. At the same time, the amount of wire consumed increases by 13%, and due to the lack of supporting structures for the resulting product surfaces, the total length of printing segments (and, accordingly, the wire consumed) also increases, which leads to an increase in the average printing time by 15-17%. An increase in delamination strength of critical structural elements (for example, brackets for mounting a laser rangefinder to the UAV body) from 27.4 (tensile) and 32.1 MPa (compressive) when such elements are produced using the classical technology to 38 (tensile) and 44.3 MPa (compressive) when producing such elements using the technology with the filament feeding control system, as well as an increase of more than 2.7 times in dynamic load cycles (load amplitude is 30 MPa; cyclic frequency is 5 Hz) for elements of critical structures obtained using the control system during their endurance testing are shown. This proves greater dynamic stability and lower fatigue of such elements and opens up the prospect of using products made by

fused deposition by means of 3D printing technology with the filament feed control system in critical structures, including military applications. Further promising research in scientific direction stated in this paper may be devoted to the study of tactical, technical and operational characteristics of the resulting products when they are actively used under critical loads and effects of aggressive environments.

ACKNOWLEDGEMENTS

The authors thank the journal's anonymous reviewers and the handling editor for helpful comments on an earlier draught. Content and any remaining errors are the sole responsibility of the authors.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Algarni, M. (2021). The influence of raster angle and moisture content on the mechanical properties of PLA parts produced by fused deposition modeling. *Polymers,* 13(2), article number 237. doi: 10.3390/polym13020237.

[2] Andriienko, O., Bondarenko, M., & Antonyuk, V. (2019). Automated system for controlling the characteristics of microsystem equipment devices. In *Quality, standardization, control: Theory and practice: Abstracts of the 19th international scientific and practical conference* (pp. 26-28). Kyiv: ATM of Ukraine.

[3] Cao, A., Filippo, B., & Chao, G. (2023). Curved layer fused deposition modeling method. In *Structural integrity of additively manufactured materials* (p. 33). Timisoara: SIAMM21.

[4] Chakraborty, D., Tirumala, T., Chitral, S., Sahoo, B.N., Kiran, D.V., & Kumar, P.A. (2022). The state of the art for wire arc additive manufacturing process of titanium alloys for aerospace applications. *Journal of Materials Engineering and Performance,* 31, 6149-6182. doi: 10.1007/s11665-022-07128-1.

[5] Doshi, M., Mahale, A., Singh, S.K., & Deshmukh, S. (2022). Printing parameters and materials affecting mechanical properties of FDM-3D printed parts: Perspective and prospects. *Materials Today: Proceedings,* 50(5), 2269-2275. doi: 10.1016/j.matpr.2021.10.003.

[6] Elmrabet, N., & Siegkas, P. (2020). Dimensional considerations on the mechanical properties of 3D printed polymer parts. *Polymer Testing,* 90, article number 106656. doi: 10.1016/j.polymertesting.2020.106656.

[7] Ferreira, R.P., & Scotti, A. (2021). The concept of a novel path planning strategy for wire+arc additive manufacturing of bulky parts: Pixel. *Metals,* 11(3), article number 498. doi: 10.3390/met11030498.

[8] Gibson, I., Rosen, D., Stucker, B., & Khorasani, M. (2021). *Additive manufacturing technologies*. Cham: Springer. doi: 10.1007/978-3-030-56127-7.

[9] Huang, J., Qin, Q., Wen, Ch., Chen, Zh., Huang, K., Fang, X., & Wang, J. (2022). A dynamic slicing algorithm for conformal additive manufacturing. *Additive Manufacturing,* 51, article number 102622. doi: 10.1016/j.addma.2022.102622.

[10] Jennings, A. (2022). *3D printing troubleshooting: All problems & solutions*. Retrieved from https://all3dp.com/1/common-3d-printing-problems-troubleshooting-3d-printer-issues/.

[11] Kiendl, J., & Chao, G. (2020). Controlling toughness and strength of FDM 3D-printed PLA components through the raster layup. *Composites Part B: Engineering,* 180, article number 107562. doi: 10.1016/j.compositesb.2019.107562.

[12] Li, Y., He, D., Yuan, Sh., Tang, K., & Zhu, J. (2022). Vector field-based curved layer slicing and path planning for multi-axis printing. *Robotics and Computer-Integrated Manufacturing,* 77, article number 102362. doi: 10.1016/j.rcim.2022.102362.

[13] Ngo, T.D., Kashani, A., Imbalzano, G., Nguyen, K.T.Q., & Hui, D. (2018). Additive manufacturing (3D printing): A review of materials, methods, applications and challenges. *Composites Part B: Engineering*, 143, 172-196. doi: 10.1016/j.compositesb.2018.02.012.

[14] Prashar, G., Vasudev, H., & Bhuddhi, D. (2023). Additive manufacturing: Expanding 3D printing horizon in industry 4.0. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 17(5), 2221-2235. doi: 10.1007/s12008-022-00956-4.

[15] Praveena, B.A., Lokesh, N., Buradi, A., Santhosh, N., Praveena, B.L., & Vignesh, R. (2022). A comprehensive review of emerging additive manufacturing (3D printing technology): Methods, materials, applications, challenges, trends and future potential. *Materials Today: Proceedings*, 52, 1309-1313. doi: 10.1016/j.matpr.2021.11.059.

[16] Sagar, S., Sharma, S.K., & Rathod, D.W. (2021). A review on process planning strategies and challenges of WAAM. *Materials Today: Proceedings,* 47(19), 6564-6575. doi: 10.1016/j.matpr.2021.02.632.

[17] Sedlak, J., Joska, Z., Jansky, J., Zouhar, J., Kolomy, S., Slany, M., Svasta, A., & Jirousek, J. (2023). Analysis of the mechanical properties of 3D-printed plastic samples subjected to selected degradation effects. *Materials,* 16(8), article number 3268. doi: 10.3390/ma16083268.

[18] Vijay, S., Ganesh, G., Navaneeth, G., Naidu, A.V., & Kumar, G.A. (2022). Optimal surface finish of material extrusion 3D printed products using Ultimaker Cura interface. *Advances in Science and Technology,* 120, 111-116. doi: 10.4028/p-hep857.
[19] Wang, F., Zheng, J., Wang, G., Jiang, D., & Ning, F. (2021). A novel printing strategy in additive manufacturing of continuous carbon fiber reinforced plastic composites. *Manufacturing Letters,* 27, 72-77. doi: 10.1016/j.mfglet.2020.12.006.
[20] Xiong, T.S., Fang, G., Dong, Q., Shen, X.Zh., & Wang, F.Y. (2022). A feedback-based print quality improving strategy for FDM 3D printing: An optimal design approach. *The International Journal of Advanced Manufacturing Technology,* 120, 2777-2791. doi: 10.1007/s00170-021-08332-4.

Особливості застосування у відповідальних конструкціях виробів виготовлених методом осадження плавленням за технологією 3D-друку

Дмитро Володимирович Тичков
Науковий співробітник
Державний науково-дослідний інститут випробувань і сертифікації озброєння та військової техніки
18000, вул. В'ячеслава Чорновола, 164, м. Черкаси, Україна
https://orcid.org/0000-0003-1741-9329

Юлія Юріївна Бондаренко
Кандидат технічних наук, професор, провідний науковий співробітник
Державний науково-дослідний інститут випробувань і сертифікації озброєння та військової техніки
18000, вул. В'ячеслава Чорновола, 164, м. Черкаси, Україна
https://orcid.org/0000-0002-5179-8329

Вячеслав Валерійович Туз
Кандидат технічних наук, доцент
Черкаський державний технологічний університет
18006, бульв. Шевченка, 460, м. Черкаси, Україна
https://orcid.org/0000-0003-2969-5080

Серафим Кирилович Сапожніков
Науковий співробітник
Державний науково-дослідний інститут випробувань і сертифікації озброєння та військової техніки
18000, вул. В'ячеслава Чорновола, 164, м. Черкаси, Україна
https://orcid.org/0000-0003-4653-203X

Дар'я Юріївна Шаповалова
Молодший науковий співробітник-інженер
Державний науково-дослідний інститут випробувань і сертифікації озброєння та військової техніки
18000, вул. В'ячеслава Чорновола, 164, м. Черкаси, Україна
https://orcid.org/0000-0001-9161-1103

Анотація. Технологія термоплавкого 3D-друку має перспективи для військової техніки та інших спеціальних застосувань за умови дотримання необхідних вимог до якості, міцності та довговічності пластикових деталей. Однак існує проблема недостатньої точності та недосліджених закономірностей зміни технологічних параметрів під час 3D-друку, що зумовлює необхідність виготовлення відповідальних конструкцій з використанням технології плазмового осадження. Основною метою статті є вивчення та оцінка особливостей використання методу осадження плавленням в технології 3D-друку для виготовлення виробів, які відносяться до відповідальних конструкцій спеціального призначення і вимагають підвищеної міцності та довговічності. Для досягнення поставленої мети було вдосконалено технологію та засоби 3D-друку методом плавкого осадження

(система контролю подачі філаменту з круговим енкодером). Встановлено, що використання системи контролю подачі філаменту дозволяє зменшити кількість візуально виявлених дефектів на поверхні надрукованих виробів в середньому на 71,7%, але збільшити кількість витраченого дроту на 13 % і середній час друку на 15-17 %, що пов'язано з відсутністю опорних конструкцій для одержуваних поверхонь. Дослідження, проведені в даній роботі, показали підвищення міцності на розшарування елементів відповідальних конструкцій, таких як кронштейни для кріплення лазерного далекоміра до корпуса безпілотного літального апарату. За допомогою системи керування подачі філаменту міцність цих елементів зросла з 27,4 МПа (при розтягуванні) і 32,1 МПа (при стисненні) до 38 МПа (при розтягуванні) і 44,3 МПа (при стисненні). Також більш ніж у 2,7 рази збільшилася кількість циклів динамічного навантаження під час випробувань на витривалість. Це дослідження свідчить про більшу динамічну стійкість і меншу втому елементів, виготовлених методом термоплавкого 3D-друку з використанням системи керування подачі нитки. Результати матимуть практичне застосування в різних галузях, в тому числі у військовій сфері

Ключові слова: адитивна технологія; якість друку; міцність; витривалість

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 004.421.22
DOI: 10.62660/2306-4412.4.2023.59-69

Deduplication of error reports in software malfunction: Algorithms for comparing call stacks

Serhii Pavlenko*
Postgraduate Student
Taras Shevchenko National University of Kyiv
01033, 60 Volodymyrska Str., Kyiv, Ukraine
https://orcid.org/0000-0003-4095-3925

Petro Kuliabko
PhD in Physical and Mathematical Sciences, Associate Professor
Taras Shevchenko National University of Kyiv
01033, 60 Volodymyrska Str., Kyiv, Ukraine
https://orcid.org/0000-0001-5411-6592

Abstract. In the software industry, the standard recognises automatic fault monitoring systems as mandatory for implementation. Considering the constant development of technologies and the high complexity of programmes, the importance of optimising processes for detecting and eliminating errors becomes a relevant task due to the need for reliability and stability of software. The purpose of this study is to conduct a detailed analysis of existing deduplication algorithms for reports from automatic systems collecting information about software failures. Among the algorithms considered were: the longest common subsequence method, Levenshtein distance, deep learning methods, Siamese neural networks, and hidden Markov models. The results obtained indicate a great potential for optimising processes of error detection and elimination in software. The developed comprehensive approach to the analysis and detection of duplicates in call stacks in failure reports allows for effectively addressing issues. The deep learning methods and hidden Markov models have demonstrated their effectiveness and feasibility for real-world applications. Effective methods for comparing key parameters of reports are identified, which contributes to the identification and grouping of recurring errors. The use of call stack comparison algorithms has proven critical for accurately identifying similar error cases in products with large audiences and high parallelism conditions. Siamese neural networks and the Scream Tracker 3 Module algorithm are used to determine the similarity of call stacks, including the application of recurrent neural networks (long short-term memory, bidirectional long short-term memory). Optimisation of report processing and clustering particularly enhances the speed and efficiency of responding to new failure cases, allowing developers to improve system stability and focus on high-priority issues. The study is useful for software developers, software development companies, system administrators, research groups, algorithm and tool development companies, cybersecurity professionals, and educational institutions

Keywords: automatic monitoring; fault detection systems; duplication removal; computer failures; analysis of interacting context structures

Article's History: Received: 31.08.2023; Revised: 16.11.2023; Accepted: 18.12.2023.

Suggested Citation:
Pavlenko, S., & Kuliabko, P. (2023). Deduplication of error reports in software malfunction: Algorithms for comparing call stacks. *Bulletin of Cherkasy State Technological University,* 28(4), 59-69. doi: 10.62660/2306-4412.4.2023.59-69.

*Corresponding author

INTRODUCTION

With the increasing complexity and volume of code in modern software systems, the probability of errors occurring during their operation is increasing. Even when using secure software development methods, errors can occur under inappropriate or unforeseen execution conditions, and reproducing them during testing is not always possible. This can have a negative impact on users and increase costs for further software maintenance. In particular, the growing importance of ensuring the security and stability of programmes indicates the importance of studying and optimising processes for detecting and fixing errors. However, with a large audience and complexity of programmes, there is a problem of effectively managing the flow of error reports coming from automatic monitoring systems. This creates the need for the development and implementation of effective methods and tools for analysing, filtering, and resolving these issues. Considering that modern software systems are used in various fields, from enterprises and e-commerce to medicine and finance, the stability and security of programmes are of great importance to end users and the business environment as a whole. Accurate resolution of this issue is a key aspect of ensuring the stability and efficiency of software development.

G.R. Sinha *et al.* (2021) and D. Feng (2022) investigated the issue of data deduplication. From the results of their study, it can be learnt that data deduplication is a data compression technology without loss that involves removing redundant data. This process involves removing duplicates, leaving only one physical copy that other copies can refer to. This leads to reduced storage costs and network bandwidth optimisation. Data deduplication methods are used to reduce the number of duplicates and repeated records. This technique improves the efficiency of data storage and usage and contributes to optimising data transmission over the network. Effective replacement of redundant fragments is achieved by identifying unique fragments and using the appropriate approach to removing duplicates.

V.S. Yakovyna and B.V. Uhrynovsky (2019) conducted an analysis of literary sources, which confirmed that the ageing of software leads to deterioration of performance and increased failures, negatively affecting its reliability and user satisfaction. The authors identified key characteristics of the phenomenon, such as effects, factors, metrics, and classification of ageing factors. The analysis of ageing modelling methods has shown that the development of hybrid approaches, combining analytical and measurement models, is a promising area for further research in this area. D. Medzatyi *et al.* (2023) emphasised that many software security approaches are aimed at avoiding complete failures, but the identification and classification of individual failures and vulnerabilities remains relevant. The literature review indicated that existing methods are not always suitable for classifying failures and vulnerabilities. Overall, this study aimed to develop a technology for identifying and classifying failures and vulnerabilities using surveys and rules based on response analysis to improve software security.

O.V. Shmatko and M.I. Myronenko (2018) discussed a testing technology that automatically sorts error reports in software. The uniqueness of this approach lies in the development of a software system for monitoring and detecting errors, which analyses error reports, automatically sorts them, and evaluates the reliability of the software. The paper by O.G. Trofymenko *et al.* (2023) emphasised that numerous risks accompany software development, and to maintain competitiveness, technology companies must effectively manage these risks. Software development projects are characterised by fast-paced and numerous changes, and risk classification helps to manage them effectively. Insufficient risk classification can lead to unforeseen problems and conflicts in risk management strategies, so adapted classifications that consider the specifics of software development and cybersecurity risks should be used.

The purpose of this study was to review existing software systems for optimisation tasks and determine their feasibility in the educational process, and to investigate systems that automatically collect data on software malfunction cases. The originality of the study lies in the examination of software for optimisation and justification of its application in educational processes. The feasibility of using technologies for detecting software vulnerabilities is generalised from the analysis of various studies, but this problem lacks a complete solution, namely due to the lack of generalised methods aimed at avoiding complete software ageing, deterioration of its performance, and increased failures, and insufficient risk management in the software development process.

MATERIALS AND METHODS

The study used the methods of longest common subsequence, Levenshtein, deep learning, Siamese neural networks and hidden Markov models. The longest common subsequence method was one of the first to be used to determine a similarity between call stack fragments (Castelluccio *et al.*, 2017). It allowed identifying sequences of functions present in both call stacks, regardless of their order. These algorithms can be used to determine the degree of similarity of programme call stacks, which is an important criterion for establishing relationships between programmes, detecting anomalies, and in other areas of software analysis.

The Levenshtein method was the next step as an improvement over the longest common subsequence method. Its logical use is due to the fact that the call stack is an ordered set of frames, and is represented as a sequence of insertion, replacement, and deletion operations. The Levenshtein algorithm is modified to account for the peculiarities of call stacks, in particular, the costs of operations are statistically determined based on the training data set.

Deep learning has shown potential for use in determining the similarity of call stacks in crash reports (Ebrahimi *et al.*, 2016). It effectively considers the context of the frame in the call stack, which allows for a more accurate determination of similarity between them. One of the prospects of this approach is the ability to automatically extract key features, facilitating efficient processing and measurement of similarity. In addition, deep learning algorithms have shown robustness to noise and data variability, providing accurate measurements of similarity in diverse and unpredictable data sets.

Siamese neural networks have proven to be an effective approach to determining the similarity of call stacks in crash reports using deep learning (Dang *et al.*, 2012). Using a Siamese architecture, the method evaluated the similarity of stacks, considering their internal structure. Key elements include converting the call stack into a vector form, using a recurrent neural network to obtain a transformed vector representation, and using a classifier to assess the similarity between stacks. With the Siamese architecture, this method provided an effective assessment of call stack similarity, with the major advantages being a smaller training sample size and the ability to automatically extract key features for more accurate comparison.

The hidden Markov model method was an approach to detecting duplicates in crash reports (van Tonder *et al.*, 2018). This algorithm was based on the idea of creating a separate hidden Markov model for each cluster, allowing for efficient determination of a specific stack belonging to a particular cluster. Each stack was considered as a discrete sequence of states or function calls, and the determination of the number of observed and hidden states was done using unique frames of the stacks in the cluster.

The general matrix H for the entire set of call stacks is calculated using the formula (Castelluccio *et al.*, 2017) (1):

$$H_{i,j} = max \begin{cases} H_{i-1,j-1} + S_{i,j} \\ H_{i-1,j} - d \\ H_{i,j-1} - d \end{cases}, \quad (1)$$

where $H(i, j)$ – optimal value of the algorithm for the i and j elements of the respective subsequences; $S(i, j)$ – value of comparing these elements; d – fixed penalty for a missing or extra function.

To calculate the result of the comparison, it is necessary to consider how important a given function $P(f)$ is (Castelluccio *et al.*, 2017) (2):

$$P_1(f) = 1 - \frac{Number\ of\ stacks\ containing\ f}{Total\ number\ of\ stacks}. \quad (2)$$

It is also necessary to consider the position of the sought function a_i in the stack, relative to the top, i.e., $P(a_i)$. The closer it is to the top, the higher the weight (Castelluccio *et al.*, 2017) (3):

$$P_2(a_i) = 1 - \frac{\text{i}}{Total\ number\ of\ frames}, \quad (3)$$

where i – interval between a_i – i-th frame of the stack being compared to b_j – j-th frame of the stack, which is introduced in the next formula (Castelluccio *et al.*, 2017) (4):

$$P_3(a_i, b_j) = e^{-|i-j|}. \quad (4)$$

The comparison function looks as follows (Castelluccio *et al.*, 2017) (5):

$$S_{i,j} = max \begin{cases} P_1(f) * P_2(a_i) * P_3(a_i, b_j), if\ a_i = b_j = f \\ 0, otherwise \end{cases}. \quad (5)$$

The weight calculation of the function has the following form (Rosenberg & Moonen, 2018) (6, 7):

$$\omega(f_i) = l\omega_\alpha(f_i) * g\omega_{\beta\gamma}(f_i), \quad (6)$$

where $\omega(f_i)$ – weight of function f_i; $g\omega_{\beta\gamma}(f_i)$ – global weight of function f_i among all stacks accumulated in the base; α, β, γ – coefficients for adjusting the algorithm.

$$l\omega_\alpha(f_i) = \frac{1}{i^\alpha}, \quad (7)$$

where i – frame number.

The calculation of the total weight is done using the *TF – IDF* ranking function (Rosenberg & Moonen, 2018) (8):

$$g\omega_{\beta\gamma}(f_i) = \frac{1}{1+e^{-x}} * (\beta * (IDF(f_i) - \gamma), \quad (8)$$

where $TF(f_i) = 1$, $IDF(f_i)$ – inverse document frequency; β, γ – parameters for tuning $IDF(f_i)$.

The calculation of the feature vector based on the two obtained vectors is (Dang *et al.*, 2012) (9):

$$features(v_1, v_2) = (|v_1 - v_2|, \frac{(v_1+v_2)}{2}, v_1 \times v_2), \quad (9)$$

where $v_1 = biLSTM(C1)$, $v_2 = biLSTM(C2)$, despite the fact that $C1$,$C2$ are call stacks.

The stack as a discrete sequence of states or function calls (van Tonder *et al.*, 2018) (10):

$$S = \{f_1, f_2, \ldots, f_m\}. \quad (10)$$

The structure of the hidden Markov model is determined as follows (van Tonder *et al.*, 2018) (11):

$$\underline{V} = \{v_1, \ldots, v_M\}, \quad (11)$$

where M – number of observable states.

The set of observations is the set of hidden states of the model (van Tonder *et al.*, 2018) (12):

$$S = \{s_1, \ldots, s_N\}, \quad (12)$$

where N – number of hidden states.

RESULTS

Collection and analysis of error reports in software. Modern information systems are rapidly increasing in code volume and complexity, and most coding errors and unforeseen conditions are not detected during the testing phase. After software release, its further maintenance requires significant resources to adapt to new operating systems, fix critical errors, and expand functionality. User reports are not efficient enough due to objective limitations. Thus, automatic error report collection is a necessary part of software development, provided by systems that monitor programmes for failures and send reports to the server when they occur (Brodie *et al*., 2005). These reports include memory dumps, log files, configuration files, and other characteristics of software operation, allowing for quicker and more effective problem resolution, collecting data about the software usage environment, measuring its stability, and prioritising failures. The intensity of the flow of error reports depends on the number of software users and can reach millions per day. The task of deduplicating reports arises from the large number of duplicates and outdated reports. Accurate deduplication is essential for correct statistics and prioritisation of failures and allowing engineers to use their time more efficiently by eliminating the need to review recurring failures and control defect fixes.

Overall, a failure is defined as the inability of software or its component to perform its functions or meet defined criteria, such as speed or resource usage. Meanwhile, a thread's call stack represents a data structure that contains information about the sequence of function calls made by a single execution thread in a programme. The thread's call stack stores information about the current state of each function in that thread, including the location of the programme counter and the values of local variables (Bartz *et al*., 2008). It can also contain information about the return point within the thread, allowing the programme to correctly resume after the function completes. In a programme with parallel execution, there may be multiple thread call stacks, each operating independently, but in case of an error or failure, tracing the call stack of that thread indicates the sequence of functions that caused the problem. In addition, there is the concept of a programme memory dump, which is a file that captures the state of the programme's memory at a specific moment in time (Wrembel, 2022). It can include the entire memory area the programme uses at that particular moment.

When it comes to existing algorithms for clustering crash reports, they can be divided into two categories depending on the information they use. In the first group, algorithms use metadata about crashes, while in the second group, they use the call stack of the thread where the crash occurred. For a better understanding of the crash report clustering processes, it is worth using the schematic diagram presented in Figure 1.

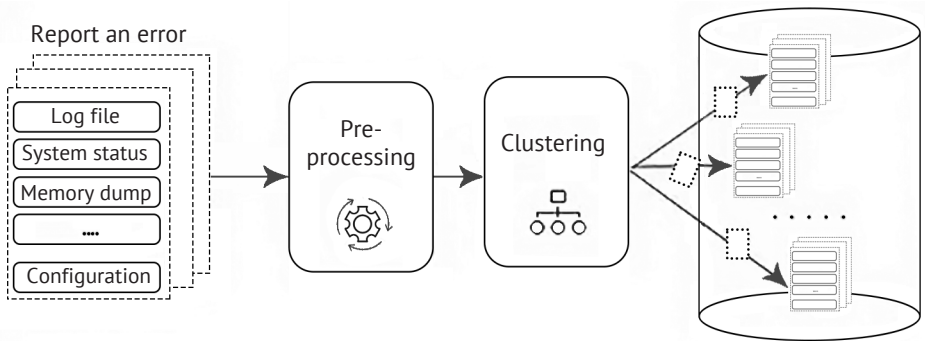

Figure 1. Schematic representation of the crash report clustering process
Source: compiled by the authors

Both approaches have their advantages and disadvantages. Among the disadvantages, the need to collect memory dumps can be highlighted, which takes time and increases the size of the report. Furthermore, the large sizes of dumps and symbol files, their complex restoration process, and possible impact on the computational power of the user's computer are disadvantages. On the other hand, the call stack provides comprehensive information about the cause of the crash and allows for additional investigation of the programme's state. In any case, both methods can be used separately or in combination. A typical example of a call stack for Linux systems can also be considered (Table 1). Frames 1 to 9 describe the programme's logic, while frames 10 and 11 represent system calls for thread initialisation.

Table 1. Example of a crash call stack in Mozilla

Frame	Module	Function signature	Path to the source file
1	libxul.so	mz::lrs::NativeCA::NotifySurfaceReady()	RenderComposNative.cpp:291
2	libxul.so	mz::lrs::NativeOGL::Unbind()	RenderComposNative.cpp:558

Continued Table 1

Frame	Module	Function signature	Path to the source file
3	webrender.so	webrender::Renderer::draw_frame	renderer/mod.rs:4412
4	libxul.so	nsThread::ProcessNextEvent(bool, bool*)	threads/nsThread.cpp:1233
5	libxul.so	MessageLoop::RunInternal()	base/message_loop.cc:381
6	libxul.so	MessageLoop::RunHandler()	base/message_loop.cc:374
7	libxul.so	nsThread::ThreadFunc(void*)	threads/nsThread.cpp:391
8	libxul.so	std::sys::unix::Thread::thread_start	sys/unix/thread.rs:108
9	firefox-bin	set_alt_start(PthreadCreateParams*)	pthread_interposer.cpp:80
10	libc.so.6	start_thread	
11	libc.so.6	clone3	

Notes: Frame 10 and frame 11 are standard functions, and their path to the source file is not important for this study.
Source: compiled by the authors

Before any actions with the call stack, a very important step is preprocessing, the main task of which is to remove insignificant data from the stack. Preprocessing may include actions such as partially removing frames of recursive calls, identifying runtime and system modules, removing fault handling frames, marking frames according to module versions, identifying insignificant or "reliable" functions, and formulating a "dictionary" of functions in the call stack. Another important criterion is that the function where the crash occurred should be identical. Nevertheless, identifying this function is not always straightforward and may not be at the top of the stack due to error handling, platform or runtime code, or changes in the product's version. In any case, the further away a function is from the top of the stack, the less likely it is to be relevant to the crash, and the more times it may appear in the reports. For example, thread initialisation functions (Fig. 2).

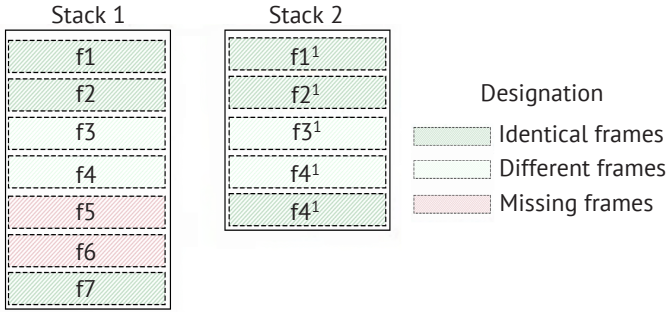

Figure 2. Example of two call stacks that are compared
Source: compiled by the authors

In addition to the function name, another important aspect is parameters such as translation unit, implementation location, and data types, which also affect comparison and evaluation. There may be multiple stacks that lead to a crash in the same function. Detecting such groups of stacks is important for comprehensive information and effective fault removal.

Algorithms for processing text information and measuring similarity. Solving the task of comparing call stacks is a relatively new topic that uses various adapted algorithms. Initial attempts were based on simple string comparison algorithms, from straightforward sequence comparison to using regular expressions. Subsequently, ranking-based algorithms were introduced, which proved to be more flexible and accurate. In addition, attempts have been made to use graph similarity algorithms. Overall, in recent years, there has been a significant variety of algorithms, including those based on machine learning methods.

There is an algorithm that uses a modified longest common subsequence (LCS) method. However, its application also requires using a variation of the Needleman-Wunsch algorithm, which allows any possible gaps but does not allow frame replacements. Considering that the function where the failure occurred is most likely to be at the top of the stack, it is recommended to introduce a parameter that determines the position relative to the top. This algorithm uses dynamic programming and calculates a common matrix for the entire set of call stacks (Formula 1).

The next stage in improving the previous method is the ReBucket algorithm. Its main principle is also based on finding LCS, but it differs from the previous one in the approach to the comparison function Si, j and the

absence of fixed penalties *d*. Its modification is known as the position-dependant model (PDM). The key idea is that functions located far from the top of the stack should have less influence on the comparison result. The algorithm allows adjusting the weight of the influence using respective parameters. The concept of "Alignment Offset" is also introduced to account for missing or extra functions.

To express the difference between two call stacks as a number, an adapted Levenshtein distance algorithm can also be used. Its use for this task is logical, as the call stack is an ordered set of frames. The basis is the classic Levenshtein distance calculation algorithm, which uses only modifications of insertion, deletion, and replacement. For the given stacks, transposition is not considered since the order of frames is important. The result of the Levenshtein distance calculation will be the minimum number of modifications. The maximum length of the two stacks will be used to normalise the obtained value so that the result is within 0 to 1.

Two essential modifications to the algorithm should be mentioned. First, unlike the classical version of the Levenshtein algorithm, where the cost of deletion, replacement, and insertion operations is considered the same, this cost should be determined separately for each operation using a statistical model built on the training data. Second, in the case of text processing, there is a constant set of characters, while when working with call stacks, a frame is an analogue of a character. The frame includes the module, address, and function signature. These parameters are also considered when determining operations with corresponding costs. An example of a frame could look like this: "Module: example_module; Function address: 0x12345678; Function signature: void example_function(int a, char b)".

Methods of deep learning are widely used for analysing various data, which makes them potentially effective for determining the similarity of call stacks in crash reports. Although these methods are actively used for crash report deduplication tasks, their application for measuring the similarity of call stacks is less common. It is worth noting the promising prospects of such an approach. For example, the ability to consider the frame context. Deep learning can effectively evaluate the relationships between different call stack frames, which allows for more accurate determination of similarity. An important criterion is the automatic extraction of key features. Deep learning models can automatically determine important features from data, providing more efficient processing and consideration of similarity. In addition, there is robustness to noise and data variability. Deep learning algorithms demonstrate greater robustness to noise and data variability, allowing for precise similarity measurements even in complex or unpredictable data sets (Yang *et al*., 2023). It is worth using feature embedding, considering the possibility of deriving semantic similarity between call stack frames. Flexibility is also an important aspect, as deep learning algorithms can be adapted to different types of data, such as text, images, and sounds, making them versatile for various applications.

Siamese neural networks, the S3M method, and hidden Markov models. The Scream Tracker 3 Module (S3M) can be considered as the first effective example of using deep learning for calculating the similarity of call stacks. In general, the Siamese architecture consists of at least two identical convolutional neural networks and an evaluation module. Here, identity means that the networks have the same configuration and weight coefficients. Convolutional networks receive call stacks as input and return feature vectors, which are passed to the evaluation module, which returns a number characterising their similarity in the range from 0 to 1.

Classical neural networks are also used to solve clustering tasks, but for this, predefined clusters and a significant training sample are required. In the case of Siamese neural networks, their task is to estimate the similarity of data from a specific domain, so they require a smaller training sample that should consider the diversity of data with which the network will work (Esteves *et al*., 2023). As for the S3M algorithm itself, its first step is to convert the call stack into a vector form. To do this, functions in frames are cropped and tokenised. Cropping is used to minimise the size of the token vocabulary, for example, module or class names can be truncated. Minimising the vocabulary helps improve comparison. Furthermore, having a vector representation of stacks, it is worth using recurrent neural networks (RNN) to obtain a transformed vector representation. Since this process requires substantial resources, the obtained representation is stored in the system, and for further comparisons, it is necessary to compute only for the input call stack.

The next step is to use a classifier for the pair of transformed call stack representations. Since these stacks can be quite large, the algorithm uses an architecture called long short-term memory (LSTM) in the RNN. This network iteratively works with data (tokens), changing its internal state at each iteration and generating an output vector-result. This allows the RNN to "remember" and consider the previous context, processing each subsequent token. Generally, RNN and its LSTM architecture were initially developed for working with text but later found application in other areas.

The classical LSTM architecture has a drawback in that the network considers only tokens that are located before the token with which it is currently working at a specific time (Islam *et al*., 2021). However, in the case of call stacks, frames $f_{k+1...n}$ located after the current frame in the stack also have a significant impact. The situation arises as follows: the direct order of frames is important for the overall assessment of stack similarity, and the reverse order is important for assessing the impact of a specific frame on the entire call stack. To consider this feature, a modification of LSTM called

bidirectional LSTM (BiLSTM) should be used. In general, this is simply two classical LSTMs, where one processes the data in the forward direction, and the other in the reverse. The result is obtained by concatenating the two vector-results of these two LSTM networks.

The next step is to calculate a feature vector based on the two obtained vectors (9), which should consider the common and different aspects of the two vectors. The final step is that the obtained result is processed by a two-layer neural network with a rectified linear unit activation function. Considering the system's ability to store computed feature vectors for already processed stacks, the amortised time complexity of the algorithm is equal to $O(dh)$, where d and h represent the size of the input and hidden layers of the two-layer neural network that performs the final similarity estimation. A drawback of this approach, like other deep learning methods, is the need for a significant training sample.

Another interesting method is the hidden Markov model (HMM). Typically, HMM is widely used for speech recognition, DNA sequence analysis, and text processing, demonstrating impressive results in learning data sequences. An example of HMM is shown below (Fig. 3).

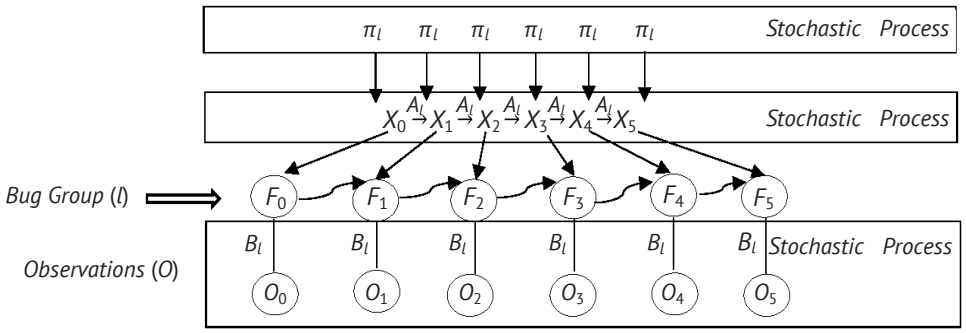

Figure 3. Detecting duplicate crash reports using HMM

Source: N. Ebrahimi *et al.* (2016)

This algorithm is based on the idea of creating a separate hidden Markov model for each cluster, which allows determining which cluster the given stack belongs to. The stack can be represented as a discrete sequence of states or function calls (10). To determine the number of observable states (11), all unique frames of stacks for all reports in the cluster should be found. The number of hidden states (12) is determined by the method of separate selection for each case within $N = [5, 10, 15, \ldots, 50]$. Other elements should also be considered. For example, A is a matrix of transition probabilities for a set of states S of size $N \times N$. Besides, B is a matrix of observation probability distribution of size $N \times M$. A value $\pi = \{\pi i\}$ is the initial distribution. The HMM model is denoted as $\lambda = (A, B, \pi)$, where A represents the probability distribution of states and transitions of the system in a Markov process. Training HMM based on discrete observations $O(O_0, O_2, \ldots, O_{T-1})$ is the maximisation of the probability $P(O|\lambda)$ among the parameters A,B,π.. The mentioned model is built for all clusters. One important feature of this algorithm is the need for historical data. It is interesting to note that this algorithm is an improvement of CrashAutomata, built on the generation of finite automata.

DISCUSSION

It is worth considering other studies in this area for a more detailed analysis of detecting duplicate error reports in software and algorithms for comparing call stacks. For example, R. Van Tonder *et al.* (2018) considered automated dynamic testing tools, including fuzzers that generate a large number of erroneous or boundary input data to provoke crashes. To reduce the number of duplicate crash reports, they use various heuristics, such as stack hashes, etc. However, after removing duplicates, there remains a substantial number of unique crashes that may correspond to the same error (have a common cause). The paper proposes a method of semantic clustering of crashes, which uses programme transformations to accurately group crashes. Instead of fast but inaccurate heuristics, the method uses approximate bug fixes through minor modifications of the programme's source code based on patch patterns and semantic feedback. This approach is suitable for general classes of bugs and outperforms built-in deduplication methods for next-generation fuzzers. Like in this study, the aforementioned study addresses the issue of deduplicating error reports. Nevertheless, this study concerns deduplicating crash reports received from users during normal usage, while in the other case, it deals with general reports of system malfunctions and their detection during testing stages. Furthermore, in the former study, unlike this one, semantic clustering of crashes is used. In this study, crash grouping is based on the similarity of call stacks.

C.M. Rosenberg and L. Moonen (2018) focused on the problem of grouping logs of software system operations related to identical faults. They investigated whether such an approach could be applied to logs of a system operating in real conditions. They also evaluated the impact of dimensionality reduction and cluster merging criteria on the quality of automatic clustering.

The authors used a method that replicates and extends the work on clustering of system logs. They considered the inclusion of dimensionality reduction methods and three cluster merging criteria. The results showed that fault detection in continuous deployment logs through clustering is reasonable, and fault detection through automatic clustering improves when dimensionality reduction is applied, which increases its robustness to different input data. The common aspects are the consideration of clustering methods. However, the methods used to achieve the purpose differ in the studies and they use different data to reach it.

Furthermore, S. Gupta and S. Gupta (2021) emphasised that defects are an integral part of software projects. In open-source projects, bug reports are stored in open bug repositories. When a new report arrives, a person called a "triager" analyses it before assigning it to the responsible developer, trying to determine if it is not a duplicate. The problem of duplicates complicates the software maintenance process, and the paper investigates the issue of detecting duplicate defect reports. It also classifies and analyses existing research on duplicate defect report detection, noting the advantages, limitations, and defining key areas for future investigation in this area. It can be concluded that both studies analyse the problem of deduplicating crash and software defect reports. However, in this study, algorithms are considered that allow automating the clustering of reports based on call stacks, rather than descriptions, unlike the other study. In addition, this study not only focuses on detecting duplicate error reports but also proposes various ways to address this issue.

S. Mukhtar *et al.* (2023) argued that defect reports can vary in granularity: some are exhaustive, while others are quite brief. In such cases, duplicate defect reports can be a useful resource for enriching defect descriptions. Nevertheless, existing methods for summarising descriptions mainly focus on individual defect reports. In this paper, the authors explored the summarisation of duplicate software defect reports to obtain an informative summary, reducing redundant phrases. They applied various text summarisation methods to duplicate defect reports, comparing their effectiveness and determining the best method. The experimental results confirm that the extractive multi-document method is the most effective, providing detailed summaries of duplicate defect reports. This helps advance defect report summarisation techniques based on unique information from duplicates. Thus, both studies address the issue of duplicate crash and software defect reports. However, this study examines clustering methods based on the similarity of call stacks and failure metadata obtained automatically, while the other study focuses on a multi-document method based on defect reports. The idea of complementing information based on duplicates is also interesting.

S. Jahan and M.M. Rahman (2022) analysed a large number of defect reports from three open-source systems, identified textually similar and distinct duplicates, and evaluated the effectiveness of existing duplicate detection methods. The authors discovered that existing methods perform poorly for textually distinct duplicates, as they often miss important components, leading to low efficiency. They also used domain-specific embeddings to improve the efficiency of traditional duplicate detection methods. The results obtained showed that such an approach of combining traditional methods with modern large language models achieves new record results, highlighting the potential of such a combination to improve software engineering tasks. The commonality in both studies lies in the use of methods for detecting duplicate error reports. Nonetheless, the difference lies in the approaches to detecting these duplicates and the data used from the reports for this purpose.

S. Qian *et al.* (2023) noted that a large number of new software defects constantly arise, creating a workload for those who fix them. Therefore, effective defect deduplication is of great importance in software development. This study begins with an examination of the available literature, including an analysis of research trends, mathematical models, methods, and widely used datasets. Further, the authors generalised the overall process of using methods, analysing them in terms of information obtained during execution and information taken from error reports. They also provide a detailed review of the methods used in relevant works. Ultimately, a detailed comparison of practical findings obtained in different studies is provided, based on indicators such as usage methods, datasets, accuracy, and completeness, among others. The main common aspect between the two studies is that they both address deduplication of errors in software. Both studies analyse different mathematical models and methods. However, the metrics, results, and the actual methods of detecting duplicates in the studies differ.

T. Zhang *et al.* (2023) focused on duplicate bug report detection (DBRD), which is a problem in science. The researchers proposed various methods for more accurate duplicate detection using both conventional and deep approaches. The study showed that deep models are not always more effective, but conventional methods are limited in that they do not capture the semantics of error reports. To overcome these limitations, they proposed using a large language model to enhance the conventional DBRD approach. This approach, called Cupid, combines the best conventional method with the ChatGPT (Generative Pre-trained Transformer) large language model, which was used to extract important information from error reports. In experiments, Cupid showed new best results, indicating the potential of combining large language models to improve performance in software engineering tasks. The common

aspects include obtaining methods for detecting duplicate error reports. Nevertheless, unlike this study, the other study uses large language models (specifically ChatGPT) and has a specific approach to detecting duplicate reports. In addition, the authors' approach is based on defect reports, whereas this study uses automatically generated crash reports.

Thus, like in this study, the above-mentioned studies address the deduplication of crash and software defect reports. They also share certain methods for detecting such duplicates and clustering. However, in each study, the detection of duplicate reports and recommendations for their resolution have their own features and differences.

CONCLUSIONS

Research on automatic detection and correction of errors in software has proven to be important in the context of the modern industry. This study examined the achievement of this goal using methods such as Longest Common Subsequence, Levenshtein distance, deep learning, Siamese neural networks, and hidden Markov models. Each of the described algorithms has its advantages and disadvantages. LCS-based algorithms determine the longest sequence of method calls present in both call stacks, regardless of their order, and are characterised by their simplicity and efficiency. Another algorithm used in many approaches for measuring the similarity of call stacks is Levenshtein distance. It calculates the minimum number of insertions, deletions, and substitutions required to transform one call stack into another. Hidden Markov models are probabilistic models that can be used to represent the sequence of method calls in a call stack. Siamese neural networks have been applied to measure the similarity of call stacks and have shown promising results. Using deep learning approaches can help identify complex dependencies between call stacks and provide more accurate clustering results.

Overall, the results indicate a substantial increase in interest in this issue, and the use of recent advances in deep learning opens up promising ways to detect and deduplicate errors. The developed comprehensive approach to the analysis and detection of duplicate call stacks considers the advantages and limitations of different methods, providing the opportunity to choose the optimal approach for a specific task. Considering the conclusions drawn during the study, several practical recommendations can be made for detecting and correcting errors in software, and for further development in this field. The methods and algorithms discussed highlight the importance of using a comprehensive approach to error analysis. Given the effectiveness of deep learning methods in measuring the similarity of call stacks, further improvement of these methods and their consideration in software lifecycle management is recommended.

It is recommended to develop universal algorithms that accurately identify similar error cases in runtime conditions on different operating systems and contribute to further optimisation of crash report processing. In addition, exploring the possibilities of using hybrid clustering models that combine stack-based approaches and metadata is recommended. This can help accommodate changes in implementation between different versions of the software. Deep learning methods are promising in terms of their ability to learn during use, which will allow the system to adapt to a specific product. Among the further research areas, the following can be highlighted: development of methods to ensure confidentiality and integration of deduplication into security systems; examination of algorithms that consider the specifics of different types of programmes and their calls; creation of effective filters to identify critical errors and filter out irrelevant or duplicate reports, etc.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Bartz, K., Stokes, J.W., Platt, J.C., Kivett, R., Grant, D., Calinoiu, S., & Loihle, G. (2008). Finding similar failures using callstack similarity. In *Proceedings of the third conference on tackling computer systems problems with machine learning techniques (Sysml'08)*. Berkeley: USENIX Association. doi: 10.5555/1855895.1855896.

[2] Brodie, M., Ma, S., Lohman, G., Mignet, L., Wilding, M., Champlin, J., & Sohn, P. (2005). Quickly finding known software problems via automated symptom matching. In *Second international conference on autonomic computing (ICAC'05)* (pp. 101-110). Seattle: Institute of Electrical and Electronics Engineers. doi: 10.1109/ICAC.2005.49.

[3] Castelluccio, M., Sansone, C., Verdoliva, L., & Poggi, G. (2017). Automatically analyzing groups of crashes for finding correlations. In *ESEC/FSE 2017: Proceedings of the 2017 11th joint meeting on foundations of software engineering* (pp. 717-726). New York: Association for Computing Machinery. doi: 10.1145/3106237.3106306.

[4] Dang, Y., Wu, R., Zhang, H., Zhang, D., & Nobel, P. (2012). ReBucket: A method for clustering duplicate crash reports based on call stack similarity. In *2012 34th international conference on software engineering (ICSE)* (pp. 1084-1093). Zurich: Institute of Electrical and Electronics Engineers. doi: 10.1109/ICSE.2012.6227111.

[5] Ebrahimi, N., Islam, S., Hamou-Lhadj, A., & Hamdaqa, M. (2016). An effective method for detecting duplicate crash reports using crash traces and hidden Markov models. In *CASCON '16: Proceedings of the 26th annual international conference on computer science and software engineering* (pp. 75-84). Riverton: IBM Corp. doi: 10.5555/3049877.3049885.
[6] Esteves, J., Costa, R., Zhou, Y., & Almeida, A. (2023). An exploratory analysis of methods for real-time data deduplication in streaming processes. In *DEBS '23: Proceedings of the 17th ACM international conference on distributed and event-based systems* (pp. 91-102). New York: Association for Computing Machinery. doi: 10.1145/3583678.3596898.
[7] Feng, D. (2022). *Data deduplication for high performance storage system*. Singapore: Springer. doi: 10.1007/978-981-19-0112-6_2.
[8] Gupta, S., & Gupta, S. (2021). A systematic study of duplicate bug report detection. *International Journal of Advanced Computer Science and Applications*, 12(1), 578-589. doi: 10.14569/IJACSA.2021.0120167.
[9] Islam, S., Hamou-Lhadj, A., Koochekian Sabor, K., Hamdaqa, M., & Cai, H. (2021). EnHMM: On the use of ensemble HMMs and stack traces to predict the reassignment of bug report fields. In *2021 IEEE international conference on software analysis, evolution and reengineering (SANER)* (pp. 411-421). Honolulu: Institute of Electrical and Electronics Engineers. doi: 10.1109/SANER50967.2021.00045.
[10] Jahan, S., & Rahman, M.M. (2022). Towards understanding the impacts of textual dissimilarity on duplicate bug report detection. In *2023 IEEE international conference on software analysis, evolution and reengineering (SANER)* (pp. 25-36). Taipa: Institute of Electrical and Electronics Engineers. doi: 10.1109/SANER56733.2023.00013.
[11] Medzatyi, D., Voichur, Yu., & Voichur, O. (2023). Technology of identification and classification of software failures and vulnerabilities. *Measuring and Computing Devices in Technological Processes*, 1, 53-57. doi: 10.31891/2219-9365-2023-73-1-8.
[12] Mukhtar, S., Primadani, C.C., Lee, S., & Jung, P. (2023). A comparison of summarization methods for duplicate software bug reports. *Electronics*, 12(16), article number 3456. doi: 10.3390/electronics12163456.
[13] Qian, C., Zhang, M., Nie, Y., Lu, S., & Cao, H. (2023). A survey on bug deduplication and triage methods from multiple points of view. *Applied Sciences*, 13(15), article number 8788. doi: 10.3390/app13158788.
[14] Rosenberg, C.M., & Moonen, L. (2018). Improving problem identification via automated log clustering using dimensionality reduction. In *ESEM '18: Proceedings of the 12th ACM/IEEE international symposium on empirical software engineering and measurement* (pp. 1-10). New York: Association for Computing Machinery. doi: 10.1145/3239235.3239248.
[15] Shmatko, O.V., & Myronenko, M.I. (2018). Information technology of depending of errors software. *Scientific Works of Kharkiv National Air Force University*, 2(56), 120-125. doi: 10.30748/zhups.2018.56.17.
[16] Sinha, G.R., Thwel, T.Th., Mohdiwale, S., & Shrivastava, D.P. (2021). Introduction to data deduplication approaches. In T.Th. Thwel & G.R. Sinha (Eds.), *Data deduplication approaches. Concepts, strategies, and challenges* (pp. 1-15). Cambridge, Massachusetts: Academic Press. doi: 10.1016/C2020-0-00104-0.
[17] Trofymenko, O.G., Loginova, N.I., Teslenko, P.O., Savielieva, O.S., & Poliakov, V.M. (2023). Classification of software project risks. *Visnyk of Kherson National Technical University*, 3(86), 119-128. doi: 10.35546/kntu2078-4481.2023.3.15.
[18] van Tonder, R., Kotheimer, J., & Le Goues, C. (2018). Semantic crash bucketing. In *ASE '18: Proceedings of the 33rd ACM/IEEE international conference on automated software engineering* (pp. 612-622). New York: Association for Computing Machinery. doi: 10.1145/3238147.3238200.
[19] Wrembel, R. (2022). Data integration, cleaning, and deduplication: Research versus industrial projects. In E. Pardede, P.D. Haghighi, I. Khalil & G. Kotsis (Eds.), *Proceedings of the 24th international conference "Information integration and web intelligence"* (pp. 3-17). Cham: Springer. doi: 10.1007/978-3-031-21047-1_1.
[20] Yakovyna, V.S., & Uhrynovskyi, B.V. (2019). Software aging in the context of its reliability: A systematic review. *Scientific Bulletin of UNFU*, 29(5), 123-128. doi: 10.15421/40290525.
[21] Yang, C., Chen, J., Fan, X., Jiang, J., & Sun, J. (2023). Silent compiler bug de-duplication via three-dimensional analysis. In *ISSTA 2023: Proceedings of the 32nd ACM SIGSOFT international symposium on software testing and analysis* (pp. 677-689). New York: Association for Computing Machinery. doi: 10.1145/3597926.3598087.
[22] Zhang, T., Irsan, I.C., Thung, F., & Lo, D. (2023). Cupid: Leveraging ChatGPT for more accurate duplicate bug report detection. *Cornell University*, 37(4), article number 1. doi: 10.48550/arXiv.2308.10022.

Дедублікація звітів про помилки в роботі програмного забезпечення: алгоритми порівняння стеків викликів

Сергій Васильович Павленко
Аспірант
Київський національний університет імені Тараса Шевченка
01033, вул. Володимирська, 60, м. Київ, Україна
https://orcid.org/0000-0003-4095-3925

Петро Петрович Кулябко

Кандидат фізико-математичних наук, доцент
Київський національний університет імені Тараса Шевченка
01033, вул. Володимирська, 60, м. Київ, Україна
https://orcid.org/0000-0001-5411-6592

Анотація. В індустрії системи автоматичного моніторингу збоїв у програмному забезпеченні визнані обов'язковим для впровадження стандартом. Враховуючи постійний розвиток технологій і високу складність програм, важливість оптимізації процесів виявлення та усунення помилок стає актуальним завданням завдяки потребі у надійності та стабільності програмного забезпечення. Мета даного дослідження полягає в детальному аналізі існуючих алгоритмів дедублікації звітів систем автоматичного збору інформації про збої у роботі програмного забезпечення. Серед розглянутих алгоритмів, були наступні: метод найдовшої спільної підпослідовності, відстань Левенштейна, методи глибинного навчання, сіамські нейронні мережі та метод прихованих марковських моделей. Отримані результати свідчать про великий потенціал оптимізації процесів виявлення та усунення помилок в програмному забезпеченні. Розроблений комплексний підхід до аналізу та виявлення дублікатів стеків викликів у звітах про збої дозволяє ефективно вирішувати проблеми. Використані методи глибинного навчання та прихованих марковських моделей проявили свою ефективність та можливість використання в реальних умовах. Зазначено ефективні способи порівняння ключових параметрів звітів, що сприяє ідентифікації та групуванню повторюваних помилок. Використання алгоритмів порівняння стеків викликів виявилося критичним для точного виявлення схожих випадків помилок у продуктах з великою аудиторією та умовами високої паралельності. Сіамські нейронні мережі та алгоритм Scream Tracker 3 Module використовуються для визначення подібності стеків викликів, зокрема, застосовуються рекурентні нейронні мережі (long short-term memory, bidirectional long short-term memory). Оптимізація обробки та кластеризації звітів значно підвищує швидкість та ефективність реагування на нові випадки збоїв, дозволяючи розробникам удосконалити стабільність системи та зосередитися на проблемах високого пріоритету. Дослідження корисне для розробників програмного забезпечення, компаній з розробки ПЗ, системних адміністраторів, дослідницьких груп, компаній з розробки алгоритмів та інструментів, фахівців у галузі кібербезпеки, а також освітніх установ

Ключові слова: автоматичний моніторинг; системи виявлення недоліків; усунення повторів; комп'ютерні збої; аналіз структури взаємодіючих контекстів

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 661.94
DOI: 10.62660/2306-4412.4.2023.70-81

Kinetic models of ozone decomposition mechanisms in aqueous solutions as sources of oxygen-containing $HO_2^{\bullet}$ and $HO^{\bullet}$ radicals

Vitalii Viazovyk*

Doctor of Technical Sciences, Associate Professor
Cherkasy State Technological University
18000, 460 Shevchenko Blvd., Cherkasy, Ukraine
http://orcid.org/0000-0001-7113-9892

Dmytro Shynkarenko

Head of the Sector
Cherkasy Scientific Research and Forensic Centre of the Ministry of Internal Affairs
18009, 104 Pasterivska Str., Cherkasy, Ukraine
https://orcid.org/0000-0003-1400-6131

Alona Kurochka

Senior Forensic Expert of the Sector
Cherkasy Scientific Research and Forensic Centre of the Ministry of Internal Affairs
18009, 104 Pasterivska Str., Cherkasy, Ukraine
https://orcid.org/0009-0002-7354-5127

Abstract. Oxygen-containing $HO_2^{\bullet}$ and $HO^{\bullet}$ radicals are one of the most effective and environmentally friendly oxidants, one of the sources of which are ozonated aqueous solutions, the mechanism of the processes occurring in them has not yet been precisely established. The purpose of the work is to compare, to confirm or refute, mechanisms proposed by S.D. Razumovsky and H.S. Stolyarenko as the most modern ones based on the processes of suppressing the formation of "thermal" nitrogen(II) oxides during fuel combustion. For this, changes in the concentration of all compounds involved in ozone destruction processes in aqueous solutions and the effect of these compounds on the formation of "thermal" nitrogen oxides have been investigated using a kinetic mathematical model. During the study, kinetic mathematical models of the process of suppressing the formation of "thermal" nitrogen(II) oxides during fuel combustion using the process of ozone destruction in aqueous solutions have been obtained according to mechanisms proposed by S.D. Razumovsky and H.S. Stolyarenko. Based on the results of the comparison of both mechanisms, it can be stated that the mechanism of H.S. Stolyarenko more realistically reflects the process of decomposition of ozone in aqueous solutions. The dependences of changes in the concentration of substances that participate in the process of decomposition of ozone in aqueous solutions on time, as well as the amount of ozone consumed for this process have been determined. Establishing of the mechanism of ozone destruction in the same solutions and the obtained dependencies will make it possible to more freely use oxygen-containing $HO_2^{\bullet}$ and $HO^{\bullet}$ radicals, which are formed during the destruction of ozone in

Article's History: Received: 07.09.2023; Revised: 14.11.2023; Accepted: 18.12.2023.

Suggested Citation:
Viazovyk, V., Shynkarenko, D., & Kurochka, A. (2023). Kinetic models of ozone decomposition mechanisms in aqueous solutions as sources of oxygen-containing $HO_2^{\bullet}$ and $HO^{\bullet}$ radicals. *Bulletin of Cherkasy State Technological University,* 28(4), 70-81. doi: 10.62660/2306-4412.4.2023.70-81.

*Corresponding author

the same solutions, for the processes of gas flows cleaning, especially for their denitrification, as well as for the intensification of oxidation processes in chemical industry, for example, in the production of nitric acid

Keywords: hydroxyl radical; peroxide radical; destruction of ozone; nitrogen(II) oxides; oxidiser

INTRODUCTION

In the case of chemodestruction or removal of toxic impurities from process gas streams or exhaust gases, high reactivity of chemical compounds, as well as the intensification of diffusion processes are of great importance. The use of highly reactive reagents – radicals is the most effective method of accelerating chemical processes. Oxygen-containing $HO_2^{\bullet}$ and $HO^{\bullet}$ radicals are found in various chemical processes: they play a significant role in the combustion of any fuel; are synthesised during the formation of ozone holes; are formed in various chemical processes in which ozone participates. The processes of synthesis of oxygen-containing $HO_2^{\bullet}$ and $HO^{\bullet}$ radicals during the destruction of ozone in aqueous solutions are of great interest. In most cases, the presence and high reactivity of oxygen-containing $HO_2^{\bullet}$ and $HO^{\bullet}$ radicals during the destruction of ozone in aqueous solutions has been experimentally proven. Ozonated aqueous solutions are one of the widespread sources of oxygen-containing radicals. However, there is still no precisely formed mechanism for this process. The decomposition of ozone in water environments has been studied for quite a long time, and already in early works attempts were made to explain its kinetics and mechanism.

The authors A.B. Murphy *et al.* (2023), F.G. Beltrán and F.J. Rivas (2023) studied and described radical mechanism of ozone decomposition in water environments, which was first proposed by D. Weiss in 1935. In the mechanism of D. Weiss, for the first-time the existence of $H_2O - O_3$ in the system was noted for superoxide radical, the formation of which was explained by the absorption band in the blue region of the spectrum during ozonation of a 40% KOH solution at a temperature of -40°C. In this mechanism, ion-radical mechanism of ozone decomposition and spontaneous interaction of radicals after their occurrence in the solution were shown. In this, the main role is given to $HO_2^{\bullet}$ radical. It has been established that the absorption band in the region of 430 nm is due to the formation of $O_3^{\bullet-}$ ozonide ion-radical under the same conditions as those used by D. Weiss.

Scientists such as A. Mojiri *et al.* (2019), W. Fan *et al.* (2020) and M. Rodríguez-Peña *et al.* (2021) in their works studied the mechanism of ozone decomposition proposed in 1950 by M. Alder and G. Hill. The scientists noted that this mechanism is more prone to neutral and acidic environments. The rate of ozone decomposition according to this mechanism is determined by the reactions of the formation of $HO_3^{\bullet}$ radical and decomposition of $HO^{\bullet}$ radical. The authors A.B. Murphy *et al.* (2023), as well as F.G. Beltrán and F.J. Rivas (2023), in their works described the mechanism proposed by E. Abel in 1955. The authors noted that E. Abel tried to take into account all intermediate complex compounds that are formed according to this mechanism and are determined spectrally: $O^{\bullet-}O^{\bullet}O^{\bullet}OH^{\bullet}$, O_2^-, $HO_2^{\bullet}$, $O^{\bullet-}O^{\bullet}O^{\bullet}$, O^{2-}. In the mechanism of E. Abel, the formation of an intermediate O $O^{\bullet}O^{\bullet}$ particle, which is essentially a $O_3^{\bullet-}$ ion-ozonide radical with an electron localised on a certain oxygen atom, was proposed for the first time.

Researchers A. Mojiri *et al.* (2019) and W. Fan *et al.* (2020) in their studies, among more modern mechanisms, described the mechanism of ozone destruction in aqueous solutions proposed by S.D. Razumovsky, and the mechanism proposed by H.S. Stolyarenko (2021; 2023). However, both of these mechanisms due to short existence time of compounds participating in the process (existence time is 10^{-4}-10^{-5} s) and therefore the impossibility of determining them by analytical methods, have not received exact confirmation or refutation.

The purpose of this study was to theoretically confirm the mechanism of ozone decomposition in aqueous solutions based on laboratory studies of suppressing the formation of "thermal" nitrogen oxides during the combustion of gaseous fuel using ozonised solutions and to create kinetic models of the investigated process based on the mechanisms of ozone destruction in aqueous solutions, proposed by S.D. Razumovsky, and the mechanism, proposed by H.S. Stolyarenko. For this purpose, the following tasks were set:

- to investigate the process of suppressing the formation of "thermal" nitrogen oxides during the combustion of gaseous fuel using ozonated solutions;
- to develop kinetic mathematical models based on the results of research into the process of ozone destruction in jet streams of aqueous solutions.

MATERIALS AND METHODS

Two mechanisms have been singled out, for which kinetic mathematical models were compiled and solved: the mechanism proposed by S.D. Razumovsky (Mojiri *et al.*, 2019; Fan *et al.*, 2020) (hereinafter the first method) and the mechanism proposed by H.S. Stolyarenko (2021; 2023) (hereinafter the second method). The purpose of this mathematical model, based on literature data and the results of studies of the intensification of combustion processes, is to select from these two mechanisms the one that would correspond to real results of the studies. Research on the intensification of combustion process consists in the use of oxygen radicals to suppress the formation of "thermal" nitrogen oxides during the combustion of gaseous fuel. The results of these researches are compared with the

results of mathematical modelling and it is determined which of the two mechanisms best corresponds to the ongoing process.

The exit of radicals is determined by changing the concentration of "thermal" nitrogen(II) oxide in flue gases. By simulating the process of the formation of "thermal" nitrogen(II) oxide before and after the introduction of radicals into the combustion zone, it is possible to calculate theoretical values of the concentration of the radical in the combustion zone. By comparing the results of modelling by both mechanisms with the results of research, the mechanism of the process can be established. In laboratory studies, the degree of suppression of the formation of nitrogen(II) oxides is achieved by 80% (Stolyarenko, 2023). Ya.B. Zeldovich proposed a chain scheme of nitrogen(II) oxidation, in which free oxygen and nitrogen atoms play an active role according to reactions (1-2) (Cellek, 2022; Gao, 2022):

$$O_2 + M \rightarrow O + O + M - 494 \text{ kJ/mol (chain initiation)}; \quad (1)$$

$$O + N_2 \rightarrow NO + N - 314 \text{ kJ/mol}; \quad (2)$$

$$N + O_2 \rightarrow NO + O + 134 \text{ kJ/mol}; \quad (3)$$

$$O + O + M \rightarrow O_2 + M + 494 \text{ kJ/mol (chain break)}. \quad (4)$$

The concentration of oxygen in combustion products can be determined from the empirical equation (Cellek, 2022; Gao, 2022)

$$CO_2 = 0.21 \frac{\alpha L_o - L_o}{1 + \alpha L_o}, \quad (5)$$

where α is the coefficient of excess air; L_o is theoretically necessary amount of air, m³/m³.

Initial concentration of atomic oxygen in the flame during the combustion with an excess of oxidiser according to the formula:

$$C_o^0 = [K]\sqrt{C_{O_2}}, \quad (6)$$

where *[K]* is the equilibrium constant (Cellek, 2022; Gao, 2022).

$$[K] = 1{,}3{*}10^4 \, exp(-\frac{61000}{RT}). \quad (7)$$

The reaction rate for each component taking into account the decomposition reaction of nitrogen oxide according to the reaction $2NO \xrightarrow{k_5} N_2 + O_2$:

$$\frac{dC_{O_2}}{d\tau} = 0; \quad (8)$$

$$\frac{dC_{N_2}}{d\tau} = -k_2 C_{N_2} C_O + k_5 (C_{NO})^2; \quad (9)$$

$$\frac{dC_N}{d\tau} = k_2 C_O C_{N_2} - k_3 C_N C_{O_2}; \quad (10)$$

$$\frac{dC_O}{d\tau} = C_o^0 - k_2 C_{N_2} C_O + k_3 C_N C_{O_2} - k_4 (C_O)^2; \quad (11)$$

$$\frac{dC_{NO}}{d\tau} = k_2 C_O C_{N_2} + k_3 C_{O_2} C_N - k_5 (C_{NO})^2. \quad (12)$$

Reaction rate constants k_2, k_3, k_4, k_5 according to equations (Surl *et al.,* 2021; Cellek, 2022; Gao, 2022):

$$k_2 = 5{,}01 \cdot 10^{13} \, exp(-\frac{316500}{RT}); \quad (13)$$

$$k_3 = 3{,}8 \cdot 10^{13} \, exp(-\frac{33500}{RT}); \quad (14)$$

$$k_4 = 11039\sqrt{T}; \quad (15)$$

$$k_5 = 3 \cdot 10^6 \, exp(-\frac{385000}{RT}). \quad (16)$$

Solving the system of equations (8-12) by the Runge-Kutta method at T=1300 K, α=1.18, L_o=8.83 m³/m³, CO_3 = 2.8 mg/dm³ and pH = 11, the dependence of the change in the concentration of nitrogen(II) oxide on time is obtained (Chauhan & Srivastava, 2019; Ince *et al.,* 2021). The formation time of the equilibrium concentration of nitrogen(II) oxide is taken to be 0.3 s.

The process of suppressing the formation of nitrogen(II) oxide during the combustion of gaseous fuel is based on the process of binding of atomic oxygen by $HO_2^{\bullet}$ radical according to the reaction (Ragupathi *et al.,* 2023)

$$HO_2^{\bullet} + O^{\bullet} \xrightarrow{k_6} HO_2^{\bullet} + O_2. \quad (17)$$

According to the chain scheme of nitrogen oxidation, proposed by Ya.B. Zeldovich, the formation of nitrogen(II) oxide decreases with a decrease in the concentration of atomic oxygen. Then the system of equations taking into account $HO_2^{\bullet}$ radical will take the form (18-22):

$$\frac{dC_{O_2}}{d\tau} = 0; \quad (18)$$

$$\frac{dC_{N_2}}{d\tau} = -k_2 C_{N_2} C_O + k_5 (C_{NO})^2; \quad (19)$$

$$\frac{dC_N}{d\tau} = k_2 C_O C_{N_2} - k_3 C_N C_{O_2}; \quad (20)$$

$$\frac{dC_O}{d\tau} = C_o^0 - k_2 C_{N_2} C_O + k_3 C_N C_{O_2} - k_4 (C_O)^2 - k_6 C_O C_{HO_2^{\bullet}}; \quad (21)$$

$$\frac{dC_{NO}}{d\tau} = k_2 C_O C_{N_2} + k_3 C_{O_2} C_N - k_5 (C_{NO})^2. \quad (22)$$

Each mechanism of ozone decomposition in aqueous solutions is considered. The first mechanism of ozone decomposition process in aqueous solutions is presented in reactions (23-29). Reaction rate for O_3, OH^-, $HO^{\bullet}$, H^+, $O_3^{\bullet}$, $HO_2^{\bullet}$, $HO_3^{\bullet}$ taking into account the reaction $H_2O \xrightarrow{k_{13}} H^+ + HO^-$ and assuming that $CH_2O = 1$:

$$\frac{dC_{O_3}}{d\tau} = 0; \quad (23)$$

$$\frac{dC_{OH^-}}{d\tau} = -k_7 C_{O_3} C_{OH^-} - k_{11} C_{OH^-} C_{HO^{\bullet}} - k_{13}; \quad (24)$$

$$\frac{dC_{HO^{\bullet}}}{d\tau} = k_7 C_{O_3} C_{OH^-} - k_{10} C_{O_3} C_{OH^-} - k_{11} C_{OH^-} C_{HO^{\bullet}}; \quad (25)$$

$$\frac{dC_{H^+}}{d\tau} = -k_8 C_{O_3^-} C_{H^+} + k_{13}; \quad (26)$$

$$\frac{dC_{O_3^{\bullet -}}}{d\tau} = k_7 C_{O_3} C_{OH^-} - k_8 C_{O_3^{\bullet -}} C_{H^+}; \quad (27)$$

$$\frac{dC_{HO_2^{\bullet}}}{d\tau} = k_{10} C_{O_3} C_{HO^{\bullet}}; \quad (28)$$

$$\frac{dC_{HO_3^{\bullet}}}{d\tau} = k_2 C_{O_3^{\bullet -}} C_{H^+} - k_9 C_{HO_3^{\bullet}}. \quad (29)$$

Reaction rate constants: $k_7 = 1.9 \cdot 10^5$, $k_8 = 1.9 \cdot 10^5$, $k_9 = 1 \cdot 10^7$, $k_{10} = 2 \cdot 10^8$, $k_{11} = 1.8 \cdot 10^5$, $k_{12} = 3 \cdot 10^5$, $k_{13} = 2.86 \cdot 10^{-4}$ s^{-1} (Feinberg, 2019; Tan *et al.*, 2019; Liu *et al.*, 2023). The concentration of ozone at the phase boundary is calculated according to the expression (Stolyarenko, 2023):

$$\frac{C_0 - C}{C_0 - C_p} = 11 - \frac{6}{\pi^2} exp(-4\pi^2 F_0'), \quad (30)$$

where $F_0' = \frac{D\theta}{d^2}$ is the Fourier diffusion criterion; C_0, C are initial concentration and concentration at the moment of time θ; C_p is the concentration at the phase boundary.

RESULTS AND DISCUSSION

The idea of using a kinetic mathematical model to confirm or disprove one or another mechanism of ozone decomposition in aqueous solutions belongs to the authors of this article. This article presents the results of these studies without comparison with the results of other researchers, which are presented in this form for the first time.

Reaction rate for O_3, OH^-, $HO_2^{\bullet}$, $O_2^{\bullet -}$, $HO^{\bullet}$, $O_3^{\bullet -}$, $O^{\bullet}$ assuming that $CH_2O = 1$:

$$\frac{dC_{O_3}}{d\tau} = 0; \quad (31)$$

$$\frac{dC_{OH^-}}{d\tau} = -k_{14} \cdot C_{OH^-} + k_{15} \cdot C_{O_2^{\bullet -}} - -k_{16} \cdot C_{O_3} \cdot C_{OH^-} + k_{21} \cdot C_{O_3^{\bullet -}}; \quad (32)$$

$$\frac{dC_{HO_2^{\bullet}}}{d\tau} = k_{14} \cdot C_{O_3} \cdot C_{OH^-} + k_{15} \cdot C_{O_2^{\bullet -}} - -k_{17} \cdot C_{HO_2^{\bullet}} \cdot C_{O_3} - k_{19} \cdot C_{HO^{\bullet}} \cdot C_{HO_2^{\bullet}} + +k_{20} \cdot C_{HO^{\bullet}} \cdot C_{O_3} + k_{22} \cdot C_{HO^{\bullet}} C_{O^{\bullet}}; \quad (33)$$

$$\frac{dC_{O_2^{\bullet -}}}{d\tau} = k_{14} \cdot C_{O_3} \cdot C_{OH^-} - k_{15} \cdot C_{O_2^{\bullet -}}; \quad (34)$$

$$\frac{dC_{OH^{\bullet}}}{d\tau} = k_{16} \cdot C_{O_3} \cdot C_{OH^-} + k_{17} \cdot C_{HO_2^{\bullet}} \cdot C_{O_3} - -k_{18} \cdot C^2_{HO^{\bullet}} - k_{19} \cdot C_{HO^{\bullet}} \cdot C_{HO_2^{\bullet -}} - -k_{22} \cdot C_{HO^{\bullet}} \cdot C_{O^{\bullet}}; \quad (35)$$

$$\frac{dC_{O_3^{\bullet -}}}{d\tau} = k_{16} \cdot C_{OH^-} \cdot C_{O_3} - k_{21} \cdot C_{O_3^{\bullet -}}; \quad (36)$$

$$\frac{dC_{O^{\bullet}}}{d\tau} = k_{18} \cdot C_{OH^{\bullet}} \cdot C_{HO^{\bullet}} - k_{22} \cdot C_{O^{\bullet}} \cdot C_{HO^{\bullet}}. \quad (37)$$

Reaction rate constants: $k_{14} = 1.2 \cdot 10^4$, $k_{15} = 1.2 \cdot 10^4$, $k_{16} = 1.9 \cdot 10^5$, $k_{17} = 8.4 \cdot 10^6$, $k_{18} = 6.67 \cdot 10^{10}$, $k_{19} = 7.5 \cdot 10^{12}$, $k_{20} = 2 \cdot 10^8$, $k_{21} = 9.05 \cdot 10^{11}$, $k_{22} = 1.99 \cdot 10^{12}$ s^{-1} (Tan *et al.*, 2019; Murphy *et al.*, 2023). As a result of calculating the system of equations (23-29) and substituting the obtained concentration of $HO_2^{\bullet}$ radical into the system (18-22), the dependence of the concentration of nitrogen(II) oxide on time in the process of supressing the formation of nitrogen(II) oxide with the synthesis of $HO_2^{\bullet}$ radical according to the first mechanism is established (Fig. 1). The results of solving the system of equations (31-37) are presented in Figure 1, the dependence of the concentration of nitrogen(II) oxide on time in the process of supressing the formation of nitrogen oxide with the synthesis of $HO_2^{\bullet}$ radical according to the second mechanism is also shown in Figure 1.

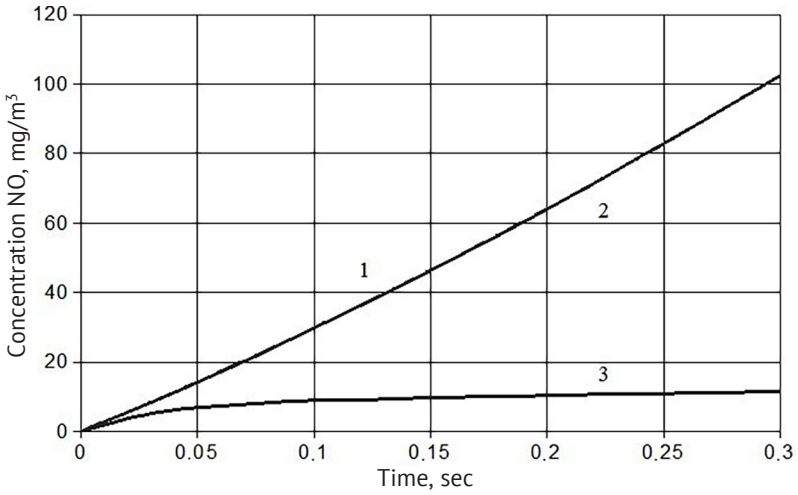

Figure 1. Dependence of the formation of nitrogen(II) oxide on time

Notes: 1 – before the introduction of radicals; 2 – according to the first mechanism; 3 – according to the second mechanism

Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

Figure 2 shows the dependence of the degree of suppressing the formation of nitrogen(II) oxide on time by both mechanisms. The results of the calculation of the system of equations (23-29) are presented in Figure 2. The second process mechanism is presented in reactions (1-5).

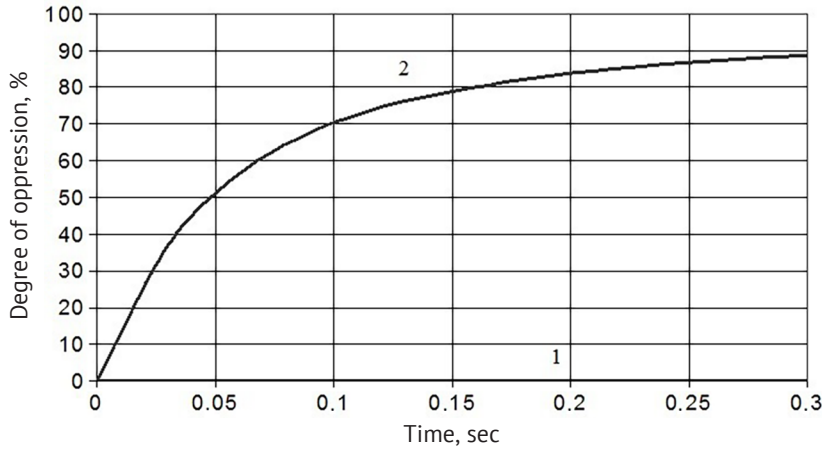

Figure 2. Dependence of the degree of suppressing the formation of nitrogen(II) oxide on time
Notes: 1 – according to the first mechanism; 2 – according to the second mechanism
Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

As can be seen from Figure 2, suppression of the formation of nitrogen(II) oxide with the synthesis of $HO_2^{\bullet}$ radical by the first mechanism almost does not occur, because the concentration of $HO_2^{\bullet}$ radical is synthesised from $HO^{\bullet}$ radical. On the other hand, the influence of $HO^{\bullet}$ radical is impossible due to relatively low quenching rate of $HO^{\bullet} + O^{\bullet}$. In Figure 1, the curve of the change in the concentration of nitrogen(II) oxide over time when using suppression of the formation of nitrogen(II) oxide with the synthesis of $HO_2^{\bullet}$ radical according to the first mechanism coincides with the curve of the change in the concentration of nitrogen(II) oxide before the introduction of radicals, which also indicates a low degree suppression of nitrogen(II) oxide. Suppression of the formation of nitrogen(II) oxide with the synthesis of $HO_2^{\bullet}$ radical by the second mechanism reaches almost 80% of the suppression of "thermal" oxides. Figure 3 shows the dependences obtained for the main compounds of the investigated process according to the first mechanism.

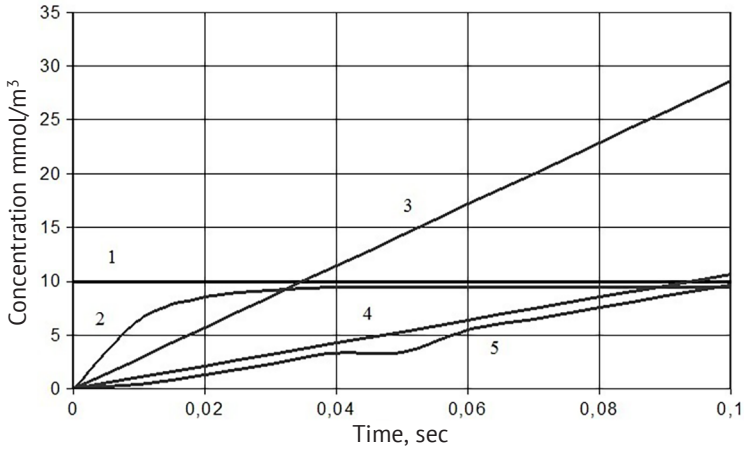

Figure 3. Dependence of changes in concentrations of substances over time according to the first mechanism
Notes: 1 – $100°C_{OH^-}$; 2 – $100°C_{HO^{\bullet}}$; 3 – $10^6°C_{H^+}$; 4 – $10 \cdot C_{O_3^{\bullet-}}$; 5 – $1000 \cdot C_{HO_2^{\bullet}}$
Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

The exit of $HO_2^{\bullet}$ radical according to the second mechanism is much higher than by the previous one (Fig. 4). Comparison of Figure 3 and Figure 4 shows that the exit of $HO_2^{\bullet}$ radical by the first mechanism is an order of magnitude lower than the formation by the second mechanism. At the same time, the formation of $HO^{\bullet}$

radical according to the first mechanism is 20 times lower than according to the second mechanism, and several times more ozonide radical is formed than according to the second mechanism. In the first mechanism, there is no superoxide radical, which plays an essential role in the formation of oxygen-containing radicals.

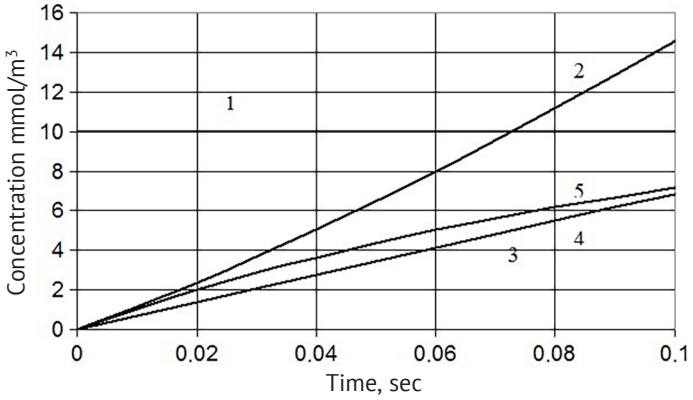

Figure 4. Dependence of changes in concentrations of substances over time according to the second mechanism
Notes: 1 – 100°C_{OH^-}; 2 – 10 · $C_{HO^\bullet}$; 3 – 10^6°$C_{O_2^{\bullet-}}$; 4 – 100 · $C_{HO_2^\bullet}$ (coincides with $C_{O_2^{\bullet-}}$); 5 – $10C_{O_3^{\bullet-}}$
Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

Due to the fact that only the concentration of "thermal" nitrogen(II) oxides decreases and the concentration of "fast" nitrogen(II) oxides remains constant, it is impossible to achieve 100% suppression. Thus, the results of the experiments on suppression of the formation of nitrogen(II) oxides, given in Table 1, and the results of the simulation of the process of suppressing the formation of nitrogen(II) oxide are adequate. It can be concluded that the second mechanism of $HO_2^\bullet$ radical synthesis more realistically reflects the processes that take place during the decomposition of ozone with increasing pH of the solution.

Table 1. Ozone concentration after passing through alkaline solutions at a contact time, τ = 006 s

OAM costs, dm^3/min	Ozone concentration, mg/dm^3			
	0.1 n	0.01 n	0.001 n	To the solution
0.5	2.5	2.55	2.61	2.8
1	1.6	2.3	2.35	2.4
2	1.7	1.9	1.95	2.1

Source: compiled by the authors based on G.S. Stolyarenko (2023)

Based on the data given in Tables 1-3, the dependence of the change in the concentration of $HO_2^\bullet$ and $HO^\bullet$ radicals on time for a 0.1 n solution is calculated at different ozone-air mixture (OAM) rates.

Table 2. Ozone concentration after passing through alkaline solutions at a contact time, τ = 0.12 s

OAM costs, dm^3/min	Ozone concentration, mg/dm^3			
	0.1 n	0.01 n	0.001 n	To the solution
0.5	2.5	2.6	2.8	2.8
1	2.0	2.05	2.1	2.4
2	1.65	1.85	2.0	2.1

Source: compiled by the authors based on G.S. Stolyarenko (2023)

Table 3. Ozone concentration after passing through alkaline solutions at a contact time, τ = 0.18 s

OAM costs, dm^3/min	Ozone concentration, mg/dm^3			
	0.1 n	0.01 n	0.001 n	To the solution
0.5	2.3	2.6	2.7	2.8
1	1.6	2.25	2.28	2.4
2	1.55	1.8	1.9	2.1

Source: compiled by the authors based on G.S. Stolyarenko (2023)

Figure 5 illustrates the formation of $HO_2^{\bullet}$ radical at a specific time point, with the time parameter denoted as t = 0.06 s. This graph provides a snapshot of the concentration of $HO_2^{\bullet}$ radical at this precise moment during the reaction or process under consideration.

The figure captures a moment in the kinetic process, allowing for a focused analysis of radical formation at the specified time of 0.06 s. The shape and trajectory of the curves in Figure 5 convey information about the rate of radical formation or depletion at this particular time instance. Upon examining Figure 6, an analysis of the behaviour of $HO_2^{\bullet}$ radical at the specific time instance is possible. The features observed, including the curve's shape, its slope, and any distinctive patterns present, provide valuable insights into the rate of radical formation or depletion during this particular period.

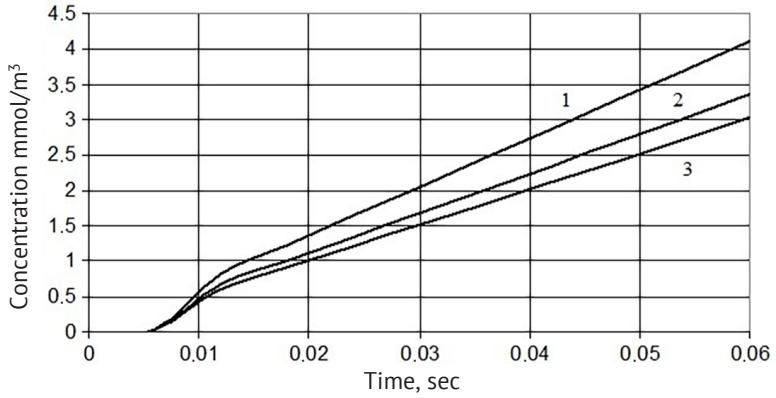

Figure 5. Formation of $HO_2^{\bullet}$ radical, t = 0.06 s

Notes: 1 – OAM consumption is 0.5 dm ^{3}/min; 2 – OAM consumption is 1 dm^3/min; 3 – OAM consumption is 2 dm^3/min

Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

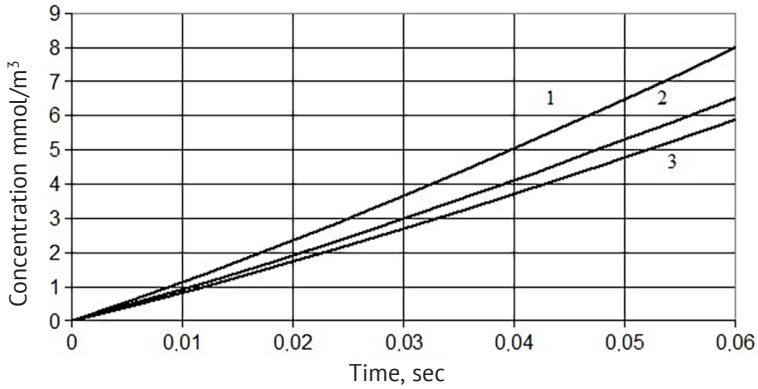

Figure 6. Formation of $HO_2^{\bullet}$ radical, τ = 0.12 s

Notes: 1 – OAM consumption is 0.5 dm^3/min; 2 – OAM consumption is 1 dm^3/min; 3 – OAM consumption is 2 dm^3/min

Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

Figure 7 illustrates the formation of $HO^{\bullet}$ radical at a specific time. The analysis of Figure 7 enables to examine the behaviour of $HO^{\bullet}$ radical precisely at the time point τ = 0.12 s. The features of the graph, including the curve's shape, its slope, and any distinctive patterns, provide valuable information regarding the rate of radical formation or depletion during this particular time segment.

In Figure 8, the graphs depict the relationships governing the formation of oxygen-containing radicals at different contact times, using kinetic mathematical models to describe the underlying processes. A similar trend is observed in Figure 9, where the concentration varies linearly with the contact time.

Figures 5-8 show the dependences of the formation of oxygen-containing radicals at different contact times according to kinetic mathematical models. As can be seen from Figures 5, 7, 8, the change in the concentration of $HO_2^{\bullet}$ radical is linear and the time of the reaction almost does not affect the course of radical formation. The same is observed in Figures 7 and 9 for $HO^{\bullet}$ radical. The concentration of $HO^{\bullet}$ radical itself is an order of magnitude higher than the concentration of $HO_2^{\bullet}$ radical. The issue of ozone decomposition in aqueous solutions has been discussed in the literature for a long time. Competitive kinetics and scavenging assay are commonly used for radical quantification.

However, the accuracy of the two methods has been challenged in electrochemical advanced oxidation processes since the strong reactivity of electrode against organic indicators may disrupt the quantitative relationship between indicator consumption and radical concentration.

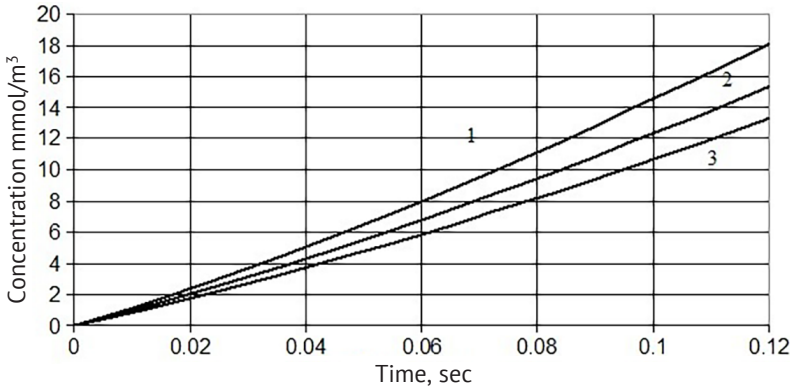

Figure 7. Formation of $HO^{\bullet}$ radical, $\tau = 0.12$ s
Notes: 1 – OAM consumption is 0.5 dm³/min; 2 – OAM consumption is 1 dm³/min; 3 – OAM consumption is 2 dm³/min
Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

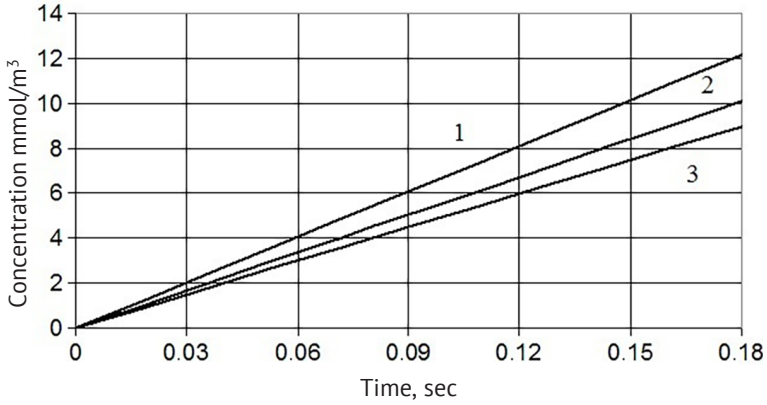

Figure 8. Formation of $HO_2^{\bullet}$ radical, $\tau = 0.18$ s
Notes: 1 – OAM consumption is 0.5 dm³/min; 2 – OAM consumption is 1 dm³/min; 3 – OAM consumption is 2 dm³/min
Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

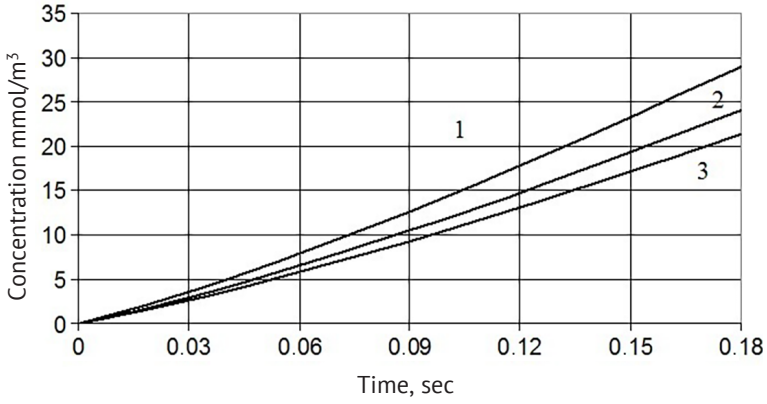

Figure 9. Formation of $HO^{\bullet}$ radical at $\tau = 0.18$ s
Notes: 1 – OAM consumption is 0.5 dm³/min; 2 – OAM consumption is 1 dm³/min; 3 – OAM consumption is 2 dm³/min
Source: compiled by the authors based on compilation and solution of kinetic mathematical models of the ozone destruction process in aqueous solutions

Authors S.Q. Yang *et al.* (2020) in the study focused on screening suitable indicators and developing suitable methods for determining steady-state concentrations of $SO_4^{\bullet-}$ and $HO^{\bullet}$ ($SO_4^{\bullet-}$) and $HO^{\bullet}$ in several EAOPs (Electrochemical Advanced Oxidation Processes) for water treatment based on competitive kinetics and scavenging assay. The applicability of modified methods and available indicators were investigated through experimental and kinetic analysis. Only in anodic process, the competitive kinetics was more appropriate than scavenging assay and benzoic acid met the basic requirement of being a competitor to determine the $HO^{\bullet}$. In the work of M. Nambari *et al.* (2021), a great role was given to the ozone decomposition process. But their attention was paid to catalytic decomposition. The catalytic decomposition approach showed higher efficiency and higher durability with no generation of considerable by-products, particularly manganese oxide (MnOx) based catalysts, which can decompose ozone to oxygen at room temperature.

Researchers M. Whiteside and J.M. Herndon (2022) in their study paid attention to the catalytic effect of various compounds on the destruction of ozone. But in this work, iron is considered as a catalyst. Iron in primary and secondary aerosols plays a crucial role in the formation of ice crystals in cirrus clouds and in polar stratospheric clouds that are involved in the formation of ozone holes. Iron is associated with reactive oxygen species, like $HO^{\bullet}$ hydroxyl radical that destroys ozone in the stratosphere. Iron is known to activate halogens including chlorine, bromine, and iodine.

Authors Y. Wang *et al.* (2021) attributed an important role to the formation of oxygen-containing radicals. But there, more attention was paid to the process of fluorescence assay by gas expansion of these radicals and no attention was paid to other compounds. E.R. Jans *et al.* (2022) also attributed a great role to the processes of the formation of $HO_2^{\bullet}$ radical in diluted mixtures of H_2 of O_2^- Ar single bond, CH_4 of O_2^- Ar single bond, and C_2H_4 of O_2^- Ar single bond excited by a repeated ns pulse discharge in a heated plasma flow reactor, measured by Cavity Ringdown Spectroscopy, a kinetic model was created. But there, more attention was paid to the process of the formation of $HO_2^{\bullet}$ radical and how other compounds of ozone decomposition in the same solutions were not considered in the previous work. In the study of H. Zhang *et al.* (2021), the authors paid attention to the influence of external factors such as ultrasound on the destruction of ozone in the process of wastewater treatment, which also took into account the processes of ozone destruction, but with additional influence of ultrasound. This greatly accelerated the process of wastewater treatment.

However, all of them are based on the results of chemical and instrumental analysis of compounds involved in the process under investigation. As already mentioned, all these compounds exist for a very short time (hundredths and thousandths of a second), they instantly change into one another, and even with modern methods it is impossible to claim with great confidence that one or another compound has been analysed and their concentrations are precisely determined. For this reason, it is proposed to use a fundamentally new approach to research the mechanism of ozone destruction in aqueous solutions, such as kinetic mathematical models, which allow to determine the concentrations and time of appearance of all necessary compounds using a mathematical apparatus and the results of laboratory studies of processes that are well studied and the main thing is that it is possible to accurately analyse the formation of compounds by chemical and instrumental methods.

CONCLUSIONS

Oxygen-containing $HO^{\bullet}$ and $HO_2^{\bullet}$ radicals occur in various chemical processes. The ozonation of aqueous solutions is one of the common sources of oxygen-containing radicals. But until this moment there is no precisely formed mechanism of this process due to its very fast course and short existence time of most of compounds that take part in it. There are many mechanisms of decomposition of ozone in aqueous solutions. All of them contain $HO^{\bullet}$ and $HO_2^{\bullet}$ radicals, but each mechanism has a different role. The presence of other radicals has been proven in all mechanisms. Even in D. Weiss's mechanism, the presence of superoxide radical was proven using spectral analysis. In some mechanisms (for example, E. Abel's mechanism), the authors try to take into account complex compounds formed during the decomposition of ozone and determined spectrally. Among the more modern mechanisms of ozone destruction in aqueous solutions, there are mechanisms proposed by S.D. Razumovsky and H.S. Stolyarenko.

To establish the detailed mechanism of these reactions, kinetic mathematical models were compiled and solved. With the help of mathematical models, it is established that the formation of nitrogen(II) oxides when using the first mechanism almost coincides with the formation without the introduction of radicals into the fuel-air mixture. This indicates that this mechanism of ozone decomposition does not provide such an amount of oxygen-containing radicals, which is necessary for the degree of suppressing the formation of nitrogen(II) oxides. According to experimental data, the formation of nitrogen(II) oxides when using the second method is significantly less than without the introduction of radicals. The degree of suppressing the formation of nitrogen(II) oxides, obtained in the mathematical model and experimentally, is almost the same. It is established that the degree of suppressing the formation of nitrogen(II) oxides by both mechanisms is very different. According to the first mechanism, it is no more than 1%, which is very different from the experimental data, while according to the second mechanism, the degree of suppression almost coincides with the experimental data.

Based on all of the above, it can be stated that the second mechanism more realistically reflects the process of decomposition of ozone in aqueous solutions. Based on the results of mathematical modelling, time dependences of changes in the concentration of substances that participate in the decomposition of ozone in aqueous solutions have been established. The obtained results of research on the determination of the mechanism of ozone destruction in aqueous solutions will allow this process to be more widely used in chemical industry in the future as a source of environmentally friendly oxidants, which are oxygen-containing radicals. Using a kinetic model of the ozone destruction process in aqueous solutions, it is possible to determine more accurate concentrations of oxygen-containing radicals at any moment of the process, which in turn allows to design equipment (for example, injection systems) for the intensification of oxidation processes when obtaining various chemical compounds and purifying gaseous flows.

Perspective areas of research in the chosen topic are: the study of detailed reaction mechanisms and ways of decomposition of ozone in aqueous solutions; the study of the influence of environmental factors such as temperature, pH, and the presence of other chemicals on the kinetics of ozone decomposition. They also include the study of catalytic effect of certain ions or compounds in aqueous solutions on the ozone decomposition process; the use of quantum chemical calculations to complement experimental data, as well as the development of dynamic system models that consider the interaction between ozone decomposition, the formation of radicals, and other relevant chemical processes, etc.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Beltrán, F.J., & Rivas, F.J. (2023). Do film reactions affect ozone gas–liquid fast-moderate reaction kinetics in the proximity of gas–water interface? *Journal of Industrial and Engineering Chemistry*, 127, 149-160. doi: 10.1016/j.jiec.2023.07.001.

[2] Cellek, M.S. (2022). The decreasing effect of ammonia enrichment on the combustion emission of hydrogen, methane, and propane fuels. *International Journal of Hydrogen Energy,* 47(45), 19916-19934. doi: 10.1016/j.ijhydene.2021.11.241.

[3] Chauhan, V., & Srivastava, P.K. (2019). Computational techniques based on Runge-Kutta method of various order and type for solving differential equations. *International Journal of Mathematical, Engineering and Management Sciences*, 4(2), 375-386. doi: 10.33889/IJMEMS.2019.4.2-030.

[4] Fan, W., An, W., Huo, M., Yang, W., Zhu, S., & Lin, Sh. (2020). Solubilization and stabilization for prolonged reactivity of ozone using micro-nano bubbles and ozone-saturated solvent: A promising enhancement for ozonation. *Separation and Purification Technology,* 238, article number 116484. doi: 10.1016/j.seppur.2019.116484.

[5] Feinberg, M. (2019). *Foundations of chemical reaction network theory*. Cham: Springer. doi: 10.1007/978-3-030-03858-8.

[6] Gao, S., Zhang, X., Chen, L., Cui, Y., Jiang, J., Zhang, Z., Peifeng, Yu., & Wang, C. (2022). Review: Radiation temperature measurement methods for engine turbine blades and environment influence. *Infrared Physics & Technology*, 123, article number 104204. doi: 10.1016/j.infrared.2022.104204.

[7] Ince, C., Ince, K., & Hanbay, D. (2021). Randomness analysis with Runge-Kutta methods. In *Volume: IDAP-2021: 5th international artificial intelligence and data processing symposium issue: Special*, 53-60. doi: 10.53070/bbd.990990.

[8] Jans, E.R., Jones, I.W., Yang, X., Miller, T.A., Stanton, J.F., & Adamovich, I.V. (2022). Time-resolved measurements of HO_2 radical in a heated plasma flow reactor. *Combustion and Flame*, 241, article number 112097. doi: 10.1016/j.combustflame.2022.112097.

[9] Liu, R.Y., Trinh, M.M., Chuang, H.T., & Chang, M.B. (2023). Ozone catalytic oxidation of low-concentration formaldehyde over ternary Mn-Ce-Ni oxide catalysts modified with FeO. *Environmental Science and Pollution Research*, 30, 32696-32709. doi: 10.1007/s11356-022-24543-y.

[10] Mojiri, A., Vakili, M., Farraji, H., & Aziz, Sh.Q. (2019). Combined ozone oxidation process and adsorption methods for the removal of acetaminophen and amoxicillin from aqueous solution; kinetic and optimisation. *Environmental Technology & Innovation*, 15, article number 100404. doi: 10.1016/j.eti.2019.100404.

[11] Murphy, A.B., Boulos, M.I., Fauchais, P.L., & Pfender, E. (2023). Handbook of thermal plasmas: Book review. Springer Nature. *Plasma Chemistry and Plasma Processing,* 43, 1277-1279. doi: 10.1007/s11090-023-10367-2.

[12] Nambari, M., Lee, C.S., & Haghighat, F. (2021). Active ozone removal technologies for a safe indoor environment: A comprehensive review. *Building and Environment*, 187, article number 107370. doi: 10.1016/j.buildenv.2020.107370.

[13] Ragupathi, A., Charpe, V.P., Hwu, J.H., & Hwang, K.C. (2023). Oxidative destruction of chlorinated persistent organic pollutants by hydroxyl radicals *via* ozone and UV light irradiation. *Green Chemistry,* 25(23), 9695-9704. doi: 10.1039/D3GC02365F.
[14] Rodríguez-Peña, M., Barrios Pérez, J.A., Llanos, J., Saez, C., Barrera-Díaz, C.E., & Rodrigo, M.A. (2021). Electrochemical generation of ozone using a PEM electrolyzer at acidic pHs. *Separation and Purification Technology*, 267, article number 118672. doi: 10.1016/j.seppur.2021.118672.
[15] Stolyarenko, H.S. (2023). *Redox systems of synthesis and use oxygen-containing radicals*. Cherkasy: Vertykal.
[16] Stolyarenko, H.S. (2021). *Ozone-radicals processes in technology denitrification gas flows*. Cherkasy: Vertykal.
[17] Surl, L., Roberts, Tj., & Bekki, S. (2021). Observation and modelling of ozone-destructive halogen chemistry in a passively degassing volcanic plume. *Atmospheric Chemistry and Physics*, 21, 12413-12441. doi: 10.5194/acp-21-12413-2021.
[18] Tan, Zh., Lu, K., Hofzumahaus, A., Fuchs, H., Bohn, B., Holland, F., ... Zhang, Yu. (2019). Experimental budgets of OH, HO_2, and RO_2 radicals and implications for ozone formation in the Pearl River Delta in China 2014. *Atmospheric Chemistry and Physics*, 19(10), 7129-7150. doi: 10.5194/acp-19-7129-2019.
[19] Wang, Y., Hu, R., Xie, P., & Chen, H. (2021). Measurement of tropospheric HO_2 radical using fluorescence assay by gas expansion with low interferences. *Journal of Environmental Sciences*, 99, 40-50. doi: 10.1016/j.jes.2020.06.010.
[20] Whiteside, M., & Herndon, J.M. (2022). Destruction of stratospheric ozone: Role of aerosolized coal fly ash iron. *European Journal of Applied Sciences*, 10(4), 143-153. doi: 10.14738/aivp.104.12689.
[21] Yang, S.Q., Cui, Y.H., Li, J.Y., & Xu-Dong, L. (2020). Determination methods for steady-state concentrations of $HO^{\bullet}$ and $SO_4^{\bullet-}$ in electrochemical advanced oxidation processes. *Chemosphere*, 261, article number 127658. doi: 10.1016/j.chemosphere.2020.127658.
[22] Zhang, H., Yiting Zhang, Y., Qiao, T., Hu, S., Liu, J., Zhu, R., ... Zhang, L. (2021). Study on ultrasonic enhanced ozone oxidation of cyanide-containing wastewater. *Separation and Purification Technology*, 303, article number 122258. doi: 10.1016/j.seppur.2022.122258.

Кінетичні моделі механізмів розкладу озону у водних розчинах як джерел кисневмісних радикалів $HO_2^{\bullet}$ і $HO^{\bullet}$

Віталій Миколайович Вязовик
Доктор технічних наук, доцент
Черкаський державний технологічний університет
18000, бульв. Шевченко, 460, м. Черкаси, Україна
http://orcid.org/0000-0001-7113-9892

Дмитро Юрійович Шинкаренко

Завідувач сектору
Черкаський науково-дослідний експертно-криміналістичний центр
Міністерства внутрішніх справ України
18009, вул. Пастерівська, 104, м. Черкаси, Україна
https://orcid.org/0000-0003-1400-6131

Альона Вікторівна Курочка

Головний судовий експерт сектору
Черкаський науково-дослідний експертно-криміналістичний центр
Міністерства внутрішніх справ України
18009, вул. Пастерівська, 104, м. Черкаси, Україна
https://orcid.org/0009-0002-7354-5127

Анотація. Оксигеновмісні радикали $HO_2^{\bullet}$ і $HO^{\bullet}$ є одними з найефективніших і екологічно чистих окисників, одним з джерел яких є озоновані водні розчини, механізм процесів, що відбуваються в них, дотепер точно не встановлений. Мета роботи – порівняти, підтвердити або спростувати механізми, запропоновані С. Д. Разумовським та Г. С. Столяренком як найбільш сучасні, що базуються на процесах пригнічення утворення «термічних» оксидів нітрогену(II) під час спалювання палива. Для цього за допомогою кінетичної математичної моделі досліджено зміну концентрації всіх сполук, що беруть участь у процесах озоноруйнування, у водних розчинах та вплив цих сполук на утворення «термічних» оксидів азоту. В ході дослідження отримано кінетичні математичні моделі процесу пригнічення утворення «термічних» оксидів нітрогену(II) при спалюванні палива з використанням процесу деструкції озону у водних розчинах за механізмами, запропонованими С. Д. Разумовським та Г. С. Столяренком. За результатами порівняння обох механізмів можна стверджувати, що механізм Г. С. Столяренка більш реалістично відображає процес розкладання озону у водних розчинах. Визначено залежності зміни концентрації речовин, які беруть участь у процесі розкладу озону у водних розчинах, від часу, а також кількості озону, що витрачається на цей процес. Встановлення механізму руйнування озону в цих же розчинах і отримані залежності дозволять більш вільно використовувати кисневмісні радикали $HO_2^{\bullet}$ і $HO^{\bullet}$, які утворюються при руйнуванні озону в цих же розчинах, для процесів очищення газових потоків, особливо для їх денітрифікації, а також для інтенсифікації окислювальних процесів у хімічній промисловості, наприклад, у виробництві азотної кислоти

Ключові слова: гідроксильний радикал; пероксидний радикал; руйнування озону; нітроген(II) оксиди; окислювач

BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 28 No. 4. 2023

UDC 004.8
DOI: 10.62660/2306-4412.4.2023.82-90

Integration of artificial intelligence technologies in data engineering: Challenges and prospects in the modern information environment

Vasyl Nesterov*
Data Analyst
Itel Inc.
6676, 107 Corporate Center Pkwy., Jacksonville, United States of America
https://orcid.org/0009-0000-3204-1382

Abstract. The integration of artificial intelligence technologies into data engineering gained significant relevancy in the context of constantly growing volumes and complexity of data, which requires innovative approaches to processing and analysis. The goal of the present study is to conduct a deep analysis of the implementation of artificial intelligence into data engineering with a focus on the challenges occurring and perspectives of this process. Research methods, such as analysis methods, comparison, systematisation, and systemic approach, were used for an objective study of this phenomenon and revealing key aspects of this topic. Analysis revealed key challenges, that include variety and instability of data, the importance of standardisation as well as ensuring security of big data amounts. The importance of ethical aspects is underlined, and perspectives on automation of analytical processes and improving prognostic analysis were also determined. According to the results, employment of common standards improves the consistency of approaches, whereas improved algorithms accelerate the processing of big data amounts. Employment of such technology as Apache Hadoop and Spark for processing big data amounts and step-by-step introduction of artificial intelligence is also useful. Increased decision explication also improves their understanding, simplifying interaction between experts and interested parties, and simultaneously creating conditions for effective implementation and employment of integrated artificial intelligence systems in data engineering. The compilation of ethical standards and legal mechanisms creates an opportunity for responsible and balanced employment of these technologies, ensuring trust and ethical compliance in the process of their implementation into various spheres of human activity. These results determine perspectives for the development of this sphere and highlight its importance in a modern information-based society. Integration of artificial intelligence into data engineering expands capabilities of automating analytical processes, ensuring accurate predictions, and reducing manual labour expenses, creating opportunities for effective management and reasoned decision-making in the data processing sphere

Keywords: intelligent systems; optimisation; security of use; automation of analytical processes; machine learning

Article's History: Received: 28.08.2023; Revised: 30.10.2023; Accepted: 18.12.2023.

INTRODUCTION

In the modern information age, where data volumes are growing exponentially, data engineering is proving to be a key area to ensure effective management and analysis of information. In this context, the integration of artificial intelligence (AI) technologies is becoming a necessity to cope with the complexity and volume of data, which is turning into a real resource, a raw material for strategic decision-making. The use o f machine learning algorithms can greatly facilitate data analysis processes, providing more accurate and faster results. Given the fact that data is becoming an important resource for strategic decision-making, the integration of

Suggested Citation:
Nesterov, V. (2023). Integration of artificial intelligence technologies in data engineering: Challenges and prospects in the modern information environment. *Bulletin of Cherkasy State Technological University,* 28(4), 82-90. doi: 10.62660/2306-4412.4.2023.82-90.

*Corresponding author

AI into data engineering opens up new opportunities for the development of different approaches to data analysis, allowing businesses to adapt to the rapidly changing business environment.

Data engineering and AI, whose interaction is becoming a key factor in development, provide a new level of data analysis and optimisation. High-tech AI algorithms not only accelerate the processing of large amounts of information, but also provide an opportunity to automate and improve processes that require high computing power and analytical skills. Integrating AI technologies into data engineering can improve efficiency and cost-effectiveness in many areas. For example, T. Shmatkovska *et al*. (2023) note in their paper that technologies contribute to improved forecasting accuracy and more informed management decisions based on data analysis. In law enforcement, AI allows automating the processes of verifying individuals, analysing large amounts of data from various sources, and identifying potential risks, which helps to increase the efficiency and speed of response of law enforcement agencies, as pointed out in the article by O. Zachek *et al*. (2023). However, other issues arise here related to the need for changes in legislation on the introduction of AI. M. Ayala-Pazmiño (2023) and U.M. Danchenko (2023) note that in the field of education and science, the use of AI can improve the quality of education and research, contributing to the development of new teaching methods and increasing the productivity of research. A.I. Shevchenko (2023), having conducted an extensive study of this topic, emphasises that the use of this technology allows automating the analysis of large amounts of data, reducing the time and effort required to process them, which contributes to increased efficiency and cost-effectiveness in many industries.

However, it is important to highlight the challenges and prospects of integrating AI into data engineering. From security and ethical issues to optimising data processing and developing new technological solutions, the integration of AI into data engineering is changing the paradigm of information management. Z. Hbur (2022), studying the issues of AI application in Ukraine's information security, emphasises its importance. In particular, it was proved that AI technologies provide an opportunity to develop solutions with higher efficiency, which contributes to the rapid detection of cyberattacks, the selection of optimal strategies for responding to security incidents, automatic assessment of the relevance and consequences of incidents, and timely response. AI's ability to automate complex analytical and decision-making tasks in real time is making significant changes to the way people typically work with large amounts of data. Delving deeper into these aspects allows better understanding the power of AI integration and identifying promising ways to use IT technologies in various industries, from business to science, as studied by V. Khaustova *et al*. (2022). The increased availability of large amounts of data, their complexity, and diversity necessitate the development of new strategies for processing and analysis. The integration of AI allows solving these problems by speeding up decision-making processes and enabling complex analyses that were previously beyond the scope of capabilities.

The research aimed to study in detail the implementation of AI in data engineering, with a focus on identifying the challenges and prospects of this process. The objectives were to highlight key challenges and opportunities, as well as to analyse important aspects that should be taken into account when implementing these technologies in the modern information environment.

MATERIALS AND METHODS

In order to thoroughly study the integration of AI technologies in data engineering and reveal its challenges and prospects, this article used a variety of research methods aimed at an in-depth analysis and objective assessment of this phenomenon. During the research, methods of analysis, comparison, system approach and systematisation were used.

The comparison method included a thorough review of the literature, where different approaches to integrating AI into data engineering were studied. Special emphasis was placed on identifying differences and similarities in methodologies used in known scientific works and projects. An analysis of concepts, theories, and definitions used in relevant scientific sources was carried out to gain a deep understanding of the approaches that define the role of AI in data engineering. This made it possible to establish the common and distinctive features of various concepts and clarify the terminological aspects that were important for the precise definition of the research context. In addition, in the process of comparison, the infrastructure, and tools that contribute to the optimisation of the processing of large volumes of data in the context of AI integration were compared. This included an ecosystem of analytical tools, machine learning platforms and other technical solutions used to ensure the efficiency and speed of data processing in accordance with defined strategies and methods.

During the work, an analytical method was used to examine in detail the key concepts and technical aspects of the integration of AI in data engineering. The analytical approach made it possible to critically evaluate the available data, identify trends and identify gaps in the current understanding of this topic. Reviews of experts actively implementing AI in the practise of their activities were considered in detail. Their experiences and conclusions are analysed, looking at the technical aspects of AI integration, such as the use of programming languages, algorithms, and architectural solutions. This approach provided an opportunity to understand the specific challenges and benefits faced by practitioners, allowing for a detailed examination of their practical experiences. An important stage was the systematisation of the results obtained during the

development of technical aspects of AI integration, which included the use of different programming languages, the deployment of algorithms and the choice of architectural solutions. The systematisation of technical details made it possible to structure information according to the effectiveness and potential limitations of AI integration.

The use of a system approach in the work made it possible to study the object of research holistically, in particular, to highlight a holistic approach to understanding the integration of AI in modern science and practise. This, in turn, made it possible to highlight in more detail the scale of the impact of the AI phenomenon itself, not only on the field of data engineering, but also on society as a whole. During the research, information resources, participation in conferences and interaction with professional communities were used. This allowed following the latest trends and innovations in the field of data engineering and AI, providing the research with relevance and contemporary context.

RESULTS

By analysing various aspects of integrating AI technologies into data engineering of this process, successful strategies and solutions have been identified that affect data security, quality and integrity, standardisation, optimisation of big data processing, and other key aspects (Talib *et al.*, 2021; Gröger, 2021; Ebid, 2021). Table 1 summarises the main challenges of AI integration and their solutions.

Table 1. Challenges and solutions in the context of integrating AI into data engineering

Challenges	Decision
Low data quality and reliability	Implementation of technologies for data cleaning and standardisation, use of quality control algorithms
Ethical and legal issues	Development of ethical standards and legal framework for the use of AI, definition of transparent decision-making processes
High complexity of integration with existing systems	Development of specialised interfaces and protocols for interaction between AI and existing systems
High costs and implementation issues	Study and implement effective cost reduction strategies, careful planning, and phased implementation
Low interpretability of models	Use of methods and algorithms that provide greater comprehensibility and interpretability of AI models
Data security and privacy risks	Applying advanced encryption tools, defining mechanisms to protect privacy, and taking into account ethical aspects of security

Source: developed by the author

One of the challenges is to ensure a high level of data security, as the integration of AI makes systems more susceptible to various cyber threats and attacks. Insufficiently protected infrastructure leads to the leakage of confidential information. While adaptive AI algorithms can detect and respond to new types of threats, they are also subject to targeted attacks that are becoming more sophisticated and difficult to detect. In addition, the increasing level of automation in data engineering, thanks to AI technologies, creates potentially vulnerable access points for unauthorised intrusion. AI technologies used to analyse data can lead to confidential information leaks, especially if the algorithms are not adequately protected against attacks and misuse. Integrating AI can push data processing and decision-making forward. However, this can lead to a large amount of load on the system, which can be used for denial-of-service (DDoS) attacks, and create unpredictable situations in the system. Using incorrect or distorted data to train AI models can lead to inaccurate or dangerous results. Employees and other internal users may attempt to exploit the system by using intelligent algorithms to uncover weaknesses and penetrate the system.

The study found that organisations that successfully integrate AI technologies into data engineering actively use a combination of approaches to ensure the highest level of protection. Elements of successful solutions include encryption and decryption. Successful integrations use modern encryption methods to protect sensitive information in real time. Additionally, they consider effective decryption mechanisms for secure access. The use of intelligent threat detection systems allows automatically detecting and responding to potential cyber threats. Machine learning algorithms are used to continuously improve the system in line with new threats. High-quality integrations include access security mechanisms and two-factor authentication to increase the level of aauthorisation and reduce the risk of unauthorised access. Event monitoring and logging systems allow detecting suspicious activity in real time and monitoring system changes. They use analytical tools to identify normal and abnormal patterns of behaviour. In the context of cloud computing, successful solutions provide strong data privacy protection using advanced encryption and key management techniques.

Another challenge to integrating AI into data engineering is data quality and purity. The use of poor quality data in integration processes leads to incorrect analytical conclusions and decision-making errors. Poor data quality can be caused by input errors, lack of

standards, or outdated information. Duplication, instability, and a variety of data formats and structures arise when integrating data from different sources. This can create contradictions in information, complicating both analysis and decision-making. The issues of data simplification and standardisation are important aspects of solving these problems. The need to use data cleaning and normalisation algorithms arises to ensure that data is homogeneous and complies with established standards. This process is key to identifying and correcting errors. The use of old or outdated data in AI models can lead to incorrect predictions and incorrect analytical results. Updating and regularly reviewing data is essential to ensure that it remains relevant. For effective AI integration, it is necessary to ensure that the data source is reliable and trustworthy. Unreliable sources can introduce errors and distort analysis results, and identifying and correcting errors in large amounts of data can be time-consuming, especially when using machine learning algorithms.

Addressing data quality and integrity issues in the context of integrating AI technologies into data engineering requires a comprehensive approach and a variety of strategies. For example, the introduction of standards for data formats and structures simplifies data integration. Simplifying data, reducing duplication, and using common standards can make processing and analysis easier. Using automated algorithms to clean and normalise data can help identify and correct errors and inconsistencies. Ensuring that data is regularly updated, and monitoring systems are in place, can help identify and correct incorrect and outdated information in a timely manner. Use machine learning algorithms to detect and correct errors and to predict future data quality. Techniques such as classification and clustering can be used for automated data analysis. Using open sources and publicly available databases allows obtaining high-quality and up-to-date data to compare and update one's own data. Error detection and prevention systems can automatically detect and correct data errors. These systems can use business logic rules and machine learning algorithms. Involve staff and specialists to review and correct data, especially in cases where context and details that are difficult to automate need to be considered.

The diversity of data sources and their heterogeneity in formats and structures is a significant barrier to integration. Data can come from a variety of sources, such as databases, sensors, web services, etc. The diversity of data formats and structures leads to heterogeneity in the presentation of information. This may include differences in field names, data types, or even in the definition of key attributes. In some cases, there may be a lack of standardised formats for representing data in a particular industry or context. The lack of commonly accepted standards makes it difficult to unify them. Solving this problem requires standardisation and the development of unified approaches to data processing. Develop a canonical data model that serves as a common and standardised format for all data sources. This unifies the presentation of information and simplifies integration. Defining a data description language (DDL) to standardise the structures and relationships between data sources promotes schema consistency and facilitates integration. A variety of data integration tools and platforms are available that have built-in tools to address heterogeneity, including Extract, Transform, Load (ETL) processes and the use of middleware. Involve industry participants in the development and adoption of standards for data representation in a particular area. Creating common standards helps to unite the industry and facilitates the exchange of data between different systems. Use of Semantic Web technologies, such as the Resource Description Framework (RDF) and the Web Ontology Language (OWL), to create semantic data models. This allows not only coordinating schemas, but also providing meanings and relationships between data.

Processing and analysing large amounts of data can be a time-consuming task, especially when using machine learning algorithms. Challenges stem from computational complexity and the need for powerful computing and storage resources. This leads to increased costs and difficulties in conducting effective maintenance. The transfer of large amounts of data between different systems often encounters network bandwidth limitations and, as a result, delays in the integration process. Ensuring the security of large volumes of data and reducing the risk of leakage or unauthorised access becomes a challenge. The use of technologies such as Apache Hadoop or Apache Spark for distributed processing of large amounts of data helps to divide computing and speed up analytical operations. Using big data storage systems, such as Apache HBase or Amazon S3, provides scalability and efficient management of large amounts of information. Data compression methods to reduce data size without losing information help reduce storage requirements and improve transmission efficiency. The use of cloud computing platforms allows for scalability of resources depending on the needs and high availability. Data preprocessing, such as filtering, weeding out uninformative or duplicate data, helps to reduce the volume of data and improve its quality. Increase network bandwidth and optimise data transfer mechanisms for efficient exchange of large amounts of information.

One important challenge is to understand and explicate the decisions made by an AI system. Failure to define how algorithms make decisions can make them difficult to adopt and implement. The lack of explication of how models work leads to a lack of trust in them, especially in areas where ethical considerations need to be taken into account and decisions need to be explained. In many cases, such as in medicine or finance, it is important not only to obtain a decision, but also to understand how this decision was made to justify its logic and objectivity. Using models that are

easily interpreted allows explicating decision-making processes. This may include the use of simple models that are easy to explain, algorithms that generate explanations or justifications for each decision made. Creating decision support systems that allow users to request and receive detailed explanations for each stage of decision-making.

The use of intelligent systems in the field of data engineering raises ethical issues, in particular, in the areas of privacy, discrimination and responsible use of technology. Bias resulting from improperly trained algorithms can cause discrimination in employment, finance, or justice. The use of AI may violate privacy and personal data protection. In the process of addressing this issue, it is proposed that the creation of independent ethics committees and regulatory bodies that define standards and rules for the use of AI from an ethical and legal perspective will help resolve this issue. It is also worth paying attention to the introduction of algorithms aimed at identifying and reducing bias in decisions, ensuring fairness and equality. Develop and implement mechanisms to ensure confidentiality and protection of personal data, such as encryption and security protocols for information processing. Another solution is to introduce ethical training in the educational programmes of specialists working with artificial intelligence to raise awareness and understanding of ethical requirements. Continuous monitoring and evaluation of the impact of AI on society and individuals to identify ethical issues and address them in a timely manner.

Integrating AI can be costly and require significant investments in hardware and staff retraining. It is important to balance the costs and expected benefits, given that integrating new technologies into existing systems can be complex and time-consuming, especially in companies with large amounts of data and extensive infrastructure. The use of AI requires the availability of qualified specialists who can understand, configure, and manage these technologies, which can be problematic due to the shortage of specialists in the labour market. It is suggested to use a strategy of gradual introduction and scaling of AI technologies, which allows for their gradual integration into existing processes without suddenly affecting the entire business. Consideration may be given to the use of off-the-shelf AI platforms and solutions that can significantly simplify the integration process and reduce costs. Study the experience of other companies and use the best practises in AI implementation, which helps to avoid common mistakes and optimise the process.

High-quality machine learning models can be poorly interpreted, making it difficult to understand their decisions and influence analytical conclusions. Many modern AI models, such as neural networks, have a complex structure, which can make their decisions difficult to interpret and understand for experts and users. Some models may be black boxes, meaning that users cannot access the internal mechanisms and parameters, which affects their openness and verifiability. Hence, when choosing models, give preference to those that are more interpretable and generate explanations for their decisions. Use visualisation techniques to present how the models work and their impacts, which helps to understand the decision-making process. Adding special explanatory modules to the models that create or provide explanations for specific decisions, which helps to make them more interpretable. Use algorithms that not only provide accurate results but also explain how they were arrived at, such as SHapley Additive exPlanations (SHAP) or Local Interpretable Model-agnostic Explanations (LIME). Provide training for professionals and users to understand the principles of the models used. Establish ethical standards regarding model interpretability to ensure trust and transparency in areas where this is important.

DISCUSSION

The introduction of AI into data engineering creates several challenges and opportunities that require careful consideration and a rational approach. Data engineering is a key element in ensuring the success of AI adoption by defining a sustainable framework for clean and organised data. Among the key issues related to the implementation of AI in data engineering are data reliability and quality. Data scientists need to ensure that data is reliable and of high quality to ensure that AI systems operate efficiently. As AI relies heavily on input data, the reliability, and quality of this data becomes key to achieving optimal results. Compared to this study, Y. Mohan and A.P. Singh (2022) take a different approach in their article and consider the key aspects of data management in the modern environment. The article emphasises the need to create reliable data for the effective use of AI, which is also evidenced by the results obtained in this study. A high standard of data quality is a prerequisite for the accuracy and reliability of AI analytical models. Data reliability is determined by its validity and resistance to internal or external influences. If the data is not reliable, there is a risk of incorrect analysis and inaccurate decision-making by AI systems. Also, high data quality means that the data is clean, accurate, and complies with specific standards. In this context, data preparation for machine learning models becomes an important issue, requiring a lot of effort from data scientists and engineers. In addition, as noted by S. Martínez-Fernández *et al.* (2022), research related to software testing and software quality is very common, while areas such as software maintenance are ignored. Comparatively, there is a similarity in the findings, which is that data-related issues are the most persistent problems.

Data development turns out to be key in the process of collecting, cleaning, and transforming data to train AI models. Sources and methods of data collection are identified, including sensors, databases, text sources, and more. Data cleaning includes the process of elimi-

nating errors, identifying and correcting incomplete or incorrect records. This stage is important because the quality of the input data determines the accuracy and reliability of AI models. Data transformation ensures that the data is suitable for use in AI training algorithms. This may include standardising formats, extracting important features, or even creating new characteristics to improve the predictive ability of the models. Compared to the author's results J. Serey *et al.* (2021) also offer an in-depth review of data development strategies and techniques for effective AI use. The paper details data sources, including sensors, databases, text sources, and more, and offers practical tips on how to collect and process them. What is different from this work is that J. Serey *et al*. (2021) focus more specifically on the methods of collecting and processing information.

The prospects for implementing AI in data engineering are significant. For example, improved data discovery, where AI technologies can optimise the data search process, saving time and effort for data consumers and providing access to a wider range of data. However, it is also necessary to keep in mind the security and ethical aspects of using AI in data engineering, as outlined in these results and in the work by S. Bharati and P. Podder (2022). It is important to keep in mind that the growing power of AI technologies should be used responsibly. Referring to the article by A. Baird and B. Schuller (2020), ethical standards should become an integral part of the development and use of integrated AI systems, as noted in the case study. The results have much in common. Ensuring transparency and accountability in the choice of algorithms, data processing, and decision-making is an important condition for the adoption of these technologies. Organisations that integrate AI into data engineering must adhere to ethical principles and implement technical means to ensure confidentiality and protect personal data. When integrating AI into data engineering, the possible implications for data privacy and security should be considered. Data engineers and developers should actively work to develop and refine an ethical framework for the use of AI in this context. The overall goal is to create integrated systems that not only effectively harness the power of AI in data engineering, but also ensure a high level of ethics and security for all users and stakeholders.

Simplified data integration and interoperability is a key benefit of implementing AI in data engineering. AI allows easily and efficiently combining different systems and data formats, removing barriers to interaction between different sources of information. AI systems can use algorithms and methods that automatically adapt to different data formats and understand their structure. One example is the introduction of natural language processing algorithms that allow recognition and adaptation to textual information from various sources, as described by J.M. John-Mathews *et al.* (2022) and F. Fui-Hoon Nah *et al.* (2023). This is interesting to consider in further research, as such approaches create a convenient and flexible mechanism for simplified data integration and interaction. Comparing the results of these authors, it is possible to note a difference in the specifics of the research topic and a deeper approach to the study of a particular object. The integration of AI into data processing services is expected to revolutionise data management, making it more scalable and cost-effective, as indicated by I.H. Sarker (2022). When comparing the results, it is worth noting the similarity in the definition of the revolution in information management.

Automation of data pipelines, enhanced by AI tools, is a strategic direction of data engineering development and means the use of algorithms and intelligent systems to optimise and automate various stages of data processing, which greatly facilitates the work of data developers. AI tools can implement technologies for automatic pattern detection, algorithms, and self-adaptive automated monitoring systems. According to A.R. Patel *et al.* (2021), this allows optimising workflows, reducing labour intensity and risks of errors during data development, which is also indicated in this study. In their work, experts note that the introduction of automation into data pipelines allows developers to focus on more valuable tasks, such as strategic planning, analysis, and improving data integration.

Machine learning and big data analytics play a key role in identifying risks as well as building accurate forecasts, which are necessary for strategic planning in various fields, as noted in the work by C. Zhang and Y. Lu (2021) and the study by J. Jöhnk *et al.* (2021). The difference between the results obtained lies in a clearer focus on the application of AI in the field of data analytics, and at the same time, more accurate results. An important aspect of integrating AI into data engineering is addressing the problem of data heterogeneity and standardisation, which requires the development of universal approaches to processing various formats and sources of information, as confirmed by the results of S.S. Gill *et al.* (2022) and this study. All of this indicates that the integration of AI into data engineering is becoming not only a technical challenge, but also requires a comprehensive approach that takes into account the economic, ethical, and social dimensions of this innovation process.

Thus, the prospects for implementing AI in data engineering include enhanced data discovery, facilitated data integration, improved data management efficiency, and automation of data engineering processes. A successful transition to the use of AI in data engineering involves addressing issues related to data reliability, quality, preparation, and automation. Harnessing the potential of AI can optimise data integration and processing, facilitating the effective use of these technologies in data engineering. The development of AI technologies in data engineering opens up new horizons for working with large volumes and diversity of data. Important aspects such as ensuring data security,

improving the quality and purity of information, and addressing ethical and legal issues are becoming key focuses on modern information architecture. In the context of dynamic monitoring and forecasting of trends in data analytics, the role of AI is critical.

CONCLUSIONS

The integration of AI technologies into data engineering is a relevant and promising area of development in the modern information environment. Despite challenges such as financial costs, complexity of integration, and ethical issues, the benefits of increased efficiency, speed, and the ability to work with large amounts of data create positive prospects for the development of this area. The study results indicate that the integration of AI technologies into data engineering significantly improves various aspects of data processing, analysis, and security, making it an important area for the development of organisations in the modern information environment. First and foremost, the use of AI technologies allows increasing the efficiency of data processing and analysis, simplifying routine tasks and bringing them to a new level of automation.

Integration of machine learning algorithms helps to improve data security by allowing timely detection and prevention of possible threats. It is important to consider the interpretability and openness of models as a key element of trust in the AI systems used. The introduction of AI stimulates the development of new tools and technological solutions that expand the ability to process and analyse large amounts of data. AI-enhanced data engineering systems are becoming more consistent and interoperable, which contributes to more efficient data processing and the creation of information resources. Integration of AI into data engineering can be used to optimise business processes, which means improving efficiency, reducing costs, and improving the quality of decision-making. For example, implementing AI analytical algorithms can improve forecasting of market trends, and automating certain tasks can free up time to tackle more complex tasks, while contributing to increased productivity and competitiveness.

AI can be used to improve medical diagnostics, predict environmental changes, and optimise transport systems. AI technologies can help in a variety of industries, and improved ethical standards will help create safer and fairer use of AI in various fields, setting the stage for sustainable and successful implementation of these technologies. Prospects for further research include considering the impact of dynamic changes in the technological environment on the integration of AI technologies into data engineering, and developing more effective strategies for taking ethical and security aspects into account.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Ayala-Pazmiño, M. (2023). Artificial intelligence in education: Exploring the potential benefits and risks. *593 Digital Publisher CEIT*, 8(3), 892-899. doi: 10.33386/593dp.2023.3.1827.

[2] Baird, A., & Schuller, B. (2020). Considerations for a more ethical approach to data in AI: On data representation and infrastructure. *Frontiers in Big Data*, 3, article number 25. doi: 10.3389/fdata.2020.00025.

[3] Bharati, S., & Podder, P. (2022). Machine and deep learning for IoT security and privacy: Applications, challenges, and future directions. *Security and Communication Networks*, 2022, article number 8951961. doi: 10.1155/2022/8951961.

[4] Danchenko, U.M. (2023). Relevance of the study of artificial intelligence recipients of higher military education. In *Materials of the all-Ukrainian science-pedagogical professional development "Technologies for the benevolent use of artificial intelligence in the fields of education and science"* (pp. 83-85). Odesa: Helvetica.

[5] Ebid, A.M. (2021). 35 Years of (AI) in geotechnical engineering: State of the art. *Geotechnical and Geological Engineering*, 39, 637-690. doi: 10.1007/s10706-020-01536-7.

[6] Fui-Hoon Nah, F., Zheng, R., Cai, J., Siau, K., & Chen, L. (2023). Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, 25(3), 277-304. doi: 10.1080/15228053.2023.2233814.

[7] Gill, S.S., Xu, M., Ottaviani, C., Patros, P., Bahsoon, R., Shaghaghi, A., Golec, M.,... Uhlig, S. (2022). AI for next generation computing: Emerging trends and future directions. *Internet of Things*, 19, article number 100514. doi: 10.1016/j.iot.2022.100514.

[8] Gröger, C. (2021). There is no AI without data. *Communications of the ACM*, 64(11), 98-108. doi: 10.1145/3448247.

[9] Hbur, Z. (2022). Use of artificial intelligence in information security in Ukraine. *Public Administration: Improvement and Development*, 1. doi: 10.32702/2307-2156-2022.1.2.

[10] Jöhnk, J., Weißert, M., & Wyrtki, K. (2021). Ready or not, AI comes – an interview study of organizational AI readiness factors. *Business & Information Systems Engineering*, 63, 5-20. doi: 10.1007/s12599-020-00676-7.

[11] John-Mathews, J.M., Cardon, D. & Balagué, C. (2022). From reality to world. A critical perspective on AI fairness. *Journal of Business Ethics*, 178, 945-959. doi: 10.1007/s10551-022-05055-8.

[12] Khaustova, V., Reshetnyak, O., & Khaustov, M. (2022). Promising areas of IT development in the world. *The Problems of Economy*, 51(1), 3-19. doi: 10.32983/2222-0712-2022-1-3-19.
[13] Martínez-Fernández, S., Bogner, J., Franch, X., Oriol, M., Siebert, J., Trendowicz, A., Vollmer, A.M., & Wagner, S. (2022). Software engineering for AI-based systems: A survey. *ACM Transactions on Software Engineering and Methodology*, 31(2), article number 37e. doi: 10.1145/3487043.
[14] Mohan, Y., & Singh, A.P. (2022). Challenges and opportunities in Big data: A review. *International Journal of Research in Electronics and Computer Engineering*, 10(4), 36-46.
[15] Patel, A.R., Ramaiya, K.K., Bhatia, C.V., Shah, H.N., & Bhavsar, S.N. (2021). Artificial intelligence: Prospect in mechanical engineering field – A review. In K. Kotecha, V. Piuri, H. Shah & R. Patel (Eds.), *Data science and intelligent applications. Lecture notes on data engineering and communications technologies* (pp. 267-282). Singapore: Springer. doi: 10.1007/978-981-15-4474-3_31.
[16] Sarker, I.H. (2022). AI-based modeling: Techniques, applications and research issues towards automation, intelligent and smart systems. *SN Computer Science*, 3, article number 158. doi: 10.1007/s42979-022-01043-x.
[17] Serey, J., Quezada, L., Alfaro, M., Fuertes, G., Vargas, E., Ternero, R., Sabattin, J., Durán, C., & Lillo, S. (2021). Artificial intelligence methodologies for data management. *Symmetry*, 13(11), article number 2040. doi: 10.3390/sym13112040.
[18] Shevchenko, A.I. (Ed.). (2023). *Strategy for artificial intelligence development in Ukraine*. Kyiv: Nauka i Osvita. doi: 10.15407/development_strategy_2023.
[19] Shmatkovska, T., Korobchuk, T., & Borysiuk, O. (2023). Modern information and communication technologies in the system of accounting and analytics for the modeling of business processes. *Economy and Society*, 53. doi: 10.32782/2524-0072/2023-53-68.
[20] Talib, M.A., Majzoub, S., Nasir, Q., & Jamal, D. (2021). A systematic literature review on hardware implementation of artificial intelligence algorithms. *The Journal of Supercomputing*, 77, 1897-1938. doi: 10.1007/s11227-020-03325-8.
[21] Zachek, O., Dmytryk, Y., & Senyk, V. (2023). The role of artificial intelligence in increasing efficiency law enforcement activities. *Scientific Bulletin of the Lviv State University of Internal Affairs*, 3, 148-156. doi: 10.32782/2311-8040/2023-3-19.
[22] Zhang, C., & Lu, Y. (2021). Study on artificial intelligence: The state of the art and future prospects. *Journal of Industrial Information Integration*, 23, article number 100224. doi: 10.1016/j.jii.2021.100224.

Інтеграція технологій штучного інтелекту в інженерію даних: виклики та перспективи у сучасному інформаційному середовищі

Василь Федорович Нестеров
Дата-аналітик
Itel Inc.
6676, бульв. Корпоративний Центр, 107, м. Джексонвіль, Сполучені Штати Америки
https://orcid.org/0009-0000-3204-1382

Анотація. Інтеграція технологій штучного інтелекту в інженерію даних набула значущої актуальності у контексті постійно зростаючого обсягу та складності даних, що вимагає новаторських підходів до їх обробки та аналізу. Метою даної статті є проведення глибокого аналізу процесу впровадження штучного інтелекту в інженерію даних з акцентом на виклики, які виникають, та перспективи цього процесу. Методи дослідження, такі як методи аналізу, порівняння, систематизації, синтезу були використані для об'єктивного вивчення цього явища та для розкриття ключових аспектів даної теми. Аналіз виявив ключові виклики, що включають різноманітність та нестабільність даних, важливість стандартизації, а також забезпечення безпеки великих обсягів інформації. Підкреслюється також важливість етичних аспектів, а також визначені перспективи в автоматизації аналітичних процесів та покращенні прогностичного аналізу. Згідно з результатами, застосування спільних стандартів покращує узгодженість підходів, а вдосконалені алгоритми прискорюють обробку великих обсягів даних. Корисним є використання технологій, таких як Apache Hadoop та Spark, для обробки великих обсягів даних, а також стратегії поступового впровадження штучного інтелекту. Збільшена експлікація рішень сприяє їхньому розумінню, полегшуючи взаємодію між фахівцями та зацікавленими сторонами, та водночас створює умови для ефективного впровадження та використання інтегрованих систем штучного інтелекту в інженерії даних. Вироблення етичних стандартів та правових механізмів відкриває шлях до відповідального та збалансованого використання цих технологій, забезпечуючи довіру та етичність у процесі їх впровадження в різноманітні сфери діяльності. Ці результати визначають перспективи розвитку цієї області та підкреслюють її важливість у сучасному інформаційному середовищі. Інтеграція штучного інтелекту в інженерію даних розширює можливості автоматизації аналітичних процесів, забезпечуючи точні прогнози та зменшуючи витрати на ручну роботу, що відкриває перспективи для ефективного управління та прийняття обґрунтованих стратегічних рішень у галузі обробки інформації

Ключові слова: інтелектуальні системи; оптимізація; безпека користування; автоматизація аналітичних процесів; машинне навчання

ВІСНИК
Черкаського державного технологічного університету

Науковий збірник

Том 28, № 4. 2023

Заснований у 1996 р. Виходить чотири рази на рік

Оригінал-макет видання виготовлено у редакційно-видавничому відділі
Черкаський державний технологічний університет

Відповідальний редактор:
Я. Бурлака

Редагування англомовних текстів:
С. Воровський, К. Касьянов

Комп'ютерна верстка:
О. Глінченко

Підписано до друку 18 грудня 2023 р.
Формат 60*84/8
Умов. друк. арк. 10,7
Наклад 50 прим.

Адреса видавництва:
Черкаський державний технологічний університет
18006, бульв. Шевченка, 460, м. Черкаси, Україна
E-mail: info@bulletin-chstu.com.ua
https://bulletin-chstu.com.ua/uk

BULLETIN
of Cherkasy State Technological University

Scientific Collection

Volume 28, No. 4. 2023

Founded in 1996. Published four times per year

The original layout of the publication is made in the editorial and publishing department of Cherkasy State Technological University

Managing editor:
Ya. Burlaka

Editing English-language texts:
S. Vorovsky, K. Kasianov

Desktop publishing:
O. Glinchenko

Signed for print of December 18, 2023.
Format 60*84/8
Conventional printed pages 10.7
Circulation 50 copies

Editors Office Address:

Cherkasy State Technological University
18006, 460 Shevchenko Blvd., Cherkasy, Ukraine
E-mail: info@bulletin-chstu.com.ua
https://bulletin-chstu.com.ua/en