Proceedings of the Workshop on Data Analytics and Mathematical Modeling Within the framework of XV International Conference on Computer Science and Information Technologies June 26-28, Tbilisi, Georgia

Institute for Informatics and Automation Problems
of the National Academy of Sciences of Armenia

The University of Georgia

Tbilisi 2024

CSIT Workshop Chair:

Yu. Shoukourian (Armenia)

Workshop Co-Chairs:

H. Astsatryan (Armenia)
K. Topuria (Georgia)
M. Topuria (Georgia)

Workshop Program Committee:

L. Aslanyan, IIAP (Armenia)- CoChair
H. Meladze, GTU (Georgia)- CoChair
A. Avetisyan, ISP (Russia)
M. Dayde, CNRS (France)
M.Harutyunyan (Armenia)
E. Khmaladze (Georgia)
V. Kvartskhelia (Georgia)
E. Poghosian (Armenia)
H. Sahakyan (Armenia)
V. Sahakyan (Armenia)
S. Tsiramua (Georgia)

Local Organizing Committee:

Merab Topuria (coChair)
Sulkhan Sulkhanishvili
Tinatin Davitashvili
Tinatin Endeladze
M. Buzhghulashvili
N. Katsitadze
L. Kvirtia

Technical Committee

Sulkhan Sulkhanishvili
Jarji Chikaidze
Guram Rukhadze
Tsotne Berishvili
Lazare Kvirtia

Layout and Cover Designer

Tatia Kvlividze

ISBN 978-9941-9897-9-7

Workshop organized by:

Institute for Informatics and Automation Problems of the National Academy of Sciences of Armenia

University of Georgia

Table of contents

Elisabed Asabashvili 4
Comparative Analysis of Ruby Language Libraries in the Field of Data Science

Levon Aslanyan, Hasmik Sahakyan 7
Graph theoretical resolution of a state-agent-environment model

Agit Atashyan, Artyom Lazyan, Davit Hayrapetyan, Hrachya Astsatryan, Vahagn Poghosyan, Suren Poghosyan 9
Cloud Infrastructure to Perform Distributed Multi user Platform for Self-organizing UAV Swarms

Romik Sargsyan, Roman Sahakyan, Emil Gazazyan, Shushanik Sargsyan, Sevak Sargsyan, Hrachya Astsatryan 13
Neural Networks vs. Traditional Techniques: Differential Equation Approaches in Quantum Gate Analysis

Nikoloz Katsitadze, Adeyinka Olanrewaju Ajagbe 17
Privacy-Preservation and Data-Sharing Security Issues in Banking Applications

Nikoloz Katsitadze, Zurab Chachava 21
Optimizing the Mid-Size Microservice Architecture Ecosystem

Konstantin Bulatov 23
Indirect error difference approximation for selecting the best anytime algorithm for tasks with unrecognizeable quality

Tatiana Kosovskaya 27
Artificial Neural Networks that Change their Configuration

Tatiana Kosovskaya, Zhou Juan 31
Algorithm for Extraction Common Properties of Objects Described in the
Predicate Calculus Language with Several Predicate Symbols

Ketevan Kutkhashvili, Neli Kilasonia 35
Scheduling Theory Problem with an Unfixed Start Time

George Giorgobiani, Vakhtang Kvaratskhelia, Vazha Tarieladze 37
Characterization of Sub-Gaussian Random Elements in Banach Spaces

Mariam Haroutunian, Arsen Margaryan, Karen Mastoyan 39
Secure Online Voting Scheme Using Steganography

Giorgi Baghaturia, Marine Menteshashvili 43
On the Numerical Solution of the Characteristic Problem for one Quasi-Linear Equation

Merab Pkhovelishvili, Natela Archvadze, Zaza Tsiramua, Mariam Pkhovelishvili 47
Overview of "Sufficient" Forecasting Models

Givi Qiria, Sulkhan Sulkhanishvili, Nikoloz Katsitadze 50
Development of examination software for the Faculty of Medicine

Simon Razmadze 52
Scoring the Outcomes of 1-3-3 Multistage Model of Computerized Adaptive Testing

Vladimir Sahakyan, Artur Vardanyan 56
Modeling of a Queueing System with Tasks Requiring Multiprocessor Service

Revaz Kakubava, Nani Salia 59
Analysis of Time Measures for the M/G/1 System in a Random Environment

Sergo Tsiramua, Hamlet Meladze. Tinatin Davitashvili, Aleksandre Benashvili, Lazare Kvirtia, Irakli Basheleishvili 62
Combinatorial Analysis of the Reliability of Multi-core Processors

Mikhail Doryshin, Elena Zamyatina 65
Robots Interaction Algorithms: The Results of Simulation Experiments

Comparative Analysis of Ruby Language Libraries in the Field of Data Science

Elisabed Asabashvili
The University of Georgia
Tbilisi, Georgia
e-mail: z.asabashvili@ug.edu.ge

https://doi.org/10.62343/csit.2024.1

***Abstract* — As is known, there are over 700 programming languages used today. Each of them has its own advantages, disadvantages, and specific capabilities, so the choice of using one or another usually depends on its suitability for a particular task. The aim of this article is to discuss the capabilities of the Ruby language in the field of data science, as well as to present and analyze the advantages and disadvantages of using Ruby language libraries compared to other libraries.**

The Ruby language, its syntax, and features that have made it concise, predictable, and suitable for object-oriented programming have been influenced by Perl and Python languages.

A positive aspect of this language is the ability to accomplish the same task in multiple ways. Ruby programming language is cross-platform due to its interpretability; it can be run on any system and work equally well on all platforms. Thus, it is quite flexible and implements principles that are impossible in compiled languages.

***Keywords* — Ruby language, Data Science, Ruby language libraries.**

I. INTRODUCTION

Programming languages are ranked according to the TIOBE community index, which is an indicator of programming language popularity and is updated monthly. Popular websites such as Google, Amazon, Wikipedia, Bing, and more than 20 others are used to calculate the rating. It is important to note that the TIOBE index does not indicate the best programming language or the language in which most code is written, but it is mainly used to check the relevance of innovations in programming or make strategic decisions [1].

According to this ranking, Python was named the most popular language. The R programming language ranks 21st, Julia ranks 37th, and Ruby ranks 18th in the TIOBE index as of March 2024. It is worth noting that languages such as Julia, Scala, Ruby, MATLAB, Octave, SAS, and others, although not at the top, also deserve attention.

The Ruby programming language began to spread in Europe only in the 2000s. Initially, all Ruby documentation was in Japanese, which was a barrier for both Europeans and American scientists. Now, as we have already mentioned, it ranks 18th in popularity worldwide.

The Ruby language was created in 1995 by Japanese programmer Yukihiro Matsumoto (Matz), and its main philosophy was to create a language that is easy to use, convenient, and comfortable, while saving human time and effort.

Ruby's syntax and features that have made it concise, predictable, and suitable for object-oriented programming have been influenced by Perl and Python languages. Another positive aspect of this language is the ability to accomplish the same task in multiple ways, depending on what is convenient for the developer [2, 3].

Thanks to its interpretation, Ruby is a cross-platform language that can work on any system and performs equally well on all platforms. Therefore, it is quite flexible and implements principles that are impossible in compiled languages.

Although Ruby is primarily used as a server-side language, since it was created as a universal language, it can be used to write programs of any type.

Most often, the server-side part of websites and web applications is written in Ruby using the Ruby on Rails framework. Despite its popularity in this regard, this is not the only area of its use.

Some important programs are also written in Ruby, for example:

Metasploit - for penetration testing (pentest), which involves simulating the actions of a malicious actor trying to gain access to a user's information systems with the aim of compromising data integrity, confidentiality, or availability.

Vagrant - for working with virtual environments.

Homebrew - for installing applications on macOS via the command line.

In other words, code written in Ruby can be found in almost any development area [4].

II. USE OF PROGRAMMING LANGUAGES IN THE PROCESS OF WORKING WITH DATA

It is well-known that knowledge of data analysis/processing methods, probability theory, statistics, mathematics, as well as programming languages, forms the basis of data science. Using programming languages in data and database management processes is crucial because writing programs and scripts is necessary for data analysis, and there are many different programming languages available for this purpose [5].

These languages include languages such as:

Python - the most popular and versatile language. It is mainly used for web development, programming of intelligent devices, and API development. It is especially popular among people working with big data. Python is used to write artificial intelligence and machine learning programs, process large data using ready-made libraries and frameworks, etc. However, its drawback is that some operations are relatively slower compared to some other languages, so it should be noted that this language may not be suitable if you need high speed. Additionally, it should be noted that in this program, a variable receives its type not during creation but during value assignment, which is undesirable and can lead to errors when working with data [5, 6].

R - is best suited for complex analytics, scientific research, and statistical data. It has tools for visualizing graphs, integration with databases, and support for machine learning methods. It contains thousands of ready-made functions but is difficult to master and practically unsuitable for solving other programming tasks. Moreover, knowledge of mathematical analysis, probability theory, and statistical methods is desirable to work with it. It works quite slowly when processing large data sets, which is due to its architectural features.

Java - is a versatile language used for working with big data, creating software, and applications. One of its main advantages is cross-platform compatibility. It is fast and very versatile compared to Python, but it is much harder to learn than other programming languages due to its complex syntax. It has fewer built-in tools than Python and R, and therefore is less useful for data analysis and processing. This language is not as widely used for data science tasks, but it should be noted that it has good libraries for working with data.

Scala - although not very popular and not even in the top 20 of the world ranking, it processes large data quickly thanks to parallel computing and is highly compatible with Java because this language runs on a virtual machine. The Scala programming language is not only difficult to learn but also challenging to read programs written in it, which is undoubtedly considered its drawback.

Go - created by Google specifically for analyzing and processing large data. It is mainly focused on extracting and analyzing information from databases, although it is also used for artificial intelligence and web development. The language is very simple and contains many standard libraries, but it is still very young and not fully matured, so it lacks some features and is not yet suitable for handling large projects. At this stage, programs are written for it only for individual microservices.

MATLAB - is a language designed for numerical computations, developed in the last century and intended for complex mathematical calculations and operations. It is more suitable for calculating indicators for data analysis rather than analyzing the data itself.

Julia - a new language specifically designed for working with data and closely resembling MATLAB. It is mainly used in machine learning and computer modeling. Like the Go language, it is not yet matured, so it has few ready-made functions and poorly structured libraries. It is worth noting that Julia is not an object-oriented language.

C++ - is a very fast general-purpose programming language, usually used for writing games, programs, etc., and is not intended for data analysis. As for data processing, tools such as MapReduce, the Caffe repository, or the Minerva neural network library are used for writing.

III. DISCUSSION OF RUBY LANGUAGE LIBRARIES AND THEIR CAPABILITIES

In a data-driven world, the importance of data science is enormous. Data is of utmost importance in today's digital world.

Thanks to the invention and advancement of mobile technologies – smartphones and tablets, mobile networks and Wi-Fi innovations – the creation and consumption of data is becoming more intense. As data grows, so does the need to process it.

The foundation of data science is knowledge of data processing methods, probability theory, statistics, coding and mathematics.

Querying data from a database, analyzing it, creating complex algorithms and running them using a neural network requires knowledge and use of programming.

Choosing effective programming languages and tools for solving Data Science problems is a very serious and responsible matter.

Data scientists should choose a programming language based on their own solution to the problems, choosing the most convenient of the listed languages.

The goal of this article is to discuss the capabilities of the Ruby language in data science and analyze the advantages and disadvantages of using this language compared to other libraries.

Unfortunately, the active use of this concise and elegant Ruby language is not observed in the field of data science, although it can be very successfully used for data analysis and machine learning.

For data science and machine learning, there is a set of libraries called SciRuby, and for statistical functions, there is DescriptiveStatistics. It is worth noting its integration with databases, specifically with SQL using Active Record and the Mongoid library for working with MongoDB [7].

Overall, the Ruby language is an excellent choice for scientific computations and data visualization. Although some languages may have more libraries, Ruby still has its place in science, and writing code in this language is always a pleasure.

It is worth mentioning that there are significant trends currently focusing on creating scientific computations in the Ruby language - the Ruby Science Foundation.

When it comes to data analysis, the focus is often on languages like Python and R, but there are also libraries in Ruby that make data analysis engaging and efficient. For example:

Pandas Library - This is the most popular Python library for convenient data manipulation and analysis. It uses a powerful tool called DataFrame (df) for this purpose. A DataFrame is a two-dimensional data structure where data is arranged in rows and columns. The Pandas DataFrame stores data in a tabular format, similar to Excel spreadsheets. Although this is a Python library, Ruby developers can access it with a shell after installation, allowing them to analyze data and write code in Ruby [8, 9].

Second frequently requested library NumPy Library - This library is mainly used for creating arrays or matrices and can be used with machine learning (ML) or deep learning (DL) models. Therefore, while Pandas is used for creating two-dimensional data objects, NumPy creates N-dimensional homogeneous objects.

It's worth noting that the Numo library in the Ruby programming language is similar to the NumPy library, and Daru serves as an equivalent to Pandas in the Ruby ecosystem.

Daru is a data analysis library written in Ruby with the tagline "Data Analysis in RUBy". Its data structure is a data frame, similar to an in-memory database table. Data frames consist of rows and columns, with each column assigned a specific data type. It's worth noting the role of the daru-view plugin, which is used in web applications and IRuby notebooks for data visualization, including simple and interactive charts. It can work in any Ruby web application environment, such as Rails, Sinatra, Nanoc, etc. [10]

Daru makes data manipulation simple and intuitive, primarily through two data structures: Daru::DataFrame and Daru::Vector. A vector is a basic one-dimensional structure similar to a labeled array, while a DataFrame is a two-dimensional structure resembling an Excel spreadsheet, used for managing and storing data collections.

Ruby users have access to several libraries, such as Pycall, which allows Ruby specialists to access Python functions. After installing Pandas, Pycall acts as a mediator between the Python and Ruby coding languages.

The ChartKick library is used for building charts in Ruby. Thanks to this library, it's possible to create histograms, pie charts, geographic diagrams, and more with just a single line of code.

The Nmatrix library is part of the SciRuby project, primarily written in C and C++, specializing in solving linear algebra problems. Nmatrix is often compared to the Panda library and the R language in terms of organizing data in matrices. It's worth mentioning that Ruby has the capability to perform simple regression, for which there is a well-known linear algebra library called SimpleLinearReprofit with several useful functions.

IV.CONCLUSION

Data science has found wide application in statistical and scientific computations, machine learning, data processing, storage, text and data analysis, as well as visualization.

As the discussion showed, some of them are more or less suitable for effective data processing due to the availability of a variety of ready-made tools, functions, and libraries. Therefore, to solve a specific data science task, it is necessary to choose a specific programming language, taking into account its pros and cons.

The goal of the article is to popularize the Ruby language in data science, discuss its capabilities, present and analyze the advantages and disadvantages of using this language compared to other libraries.

Unfortunately, the use of this concise and elegant language is not actively observed in the field of data science, although it can be very successfully used in data analysis, working with arrays, hashes, and machine learning.

REFERENCES

1. https://www.tiobe.com/tiobe-index/
2. J. Evans *Polished Ruby Programming*, Birmingham, UK, 2021
3. N. Rappin and D. Thomas, *Programming Ruby 3.3 (5th Edition)*, Publisher: Pragmatic Bookshelf, 2024
4. https://github.com
5. D. P. Kroese, Z. I. Botev, T. Taimre, R. Vaisman, *Data Science and Machine Learning. Mathematical and Statistical Methods*, CRC Press, 2023
6. E. Asabashvili. *Challenges of media digitization in Georgia*. XIII International Conference of the Union of Mathematicians of Georgia. Batumi, Georgia 2023.
7. SciRuby.com
8. E. Asabashvili. *Technological Challenges in Education.* 7th International Conference on Social Research and Behavioral Sciences. Antalya, Turkey, 2020
9. https://github.com/mrkn/pandas.rb
10. https://github.com/SciRuby/daru

Graph Theoretical Resolution of a State-agent-environment Model

Levon Aslanyan
Institute for Informatics and Automation
Problems of NAS RA
Yerevan, Armenia
e-mail: lasl@sci.am

Hasmik Sahakyan
Institute for Informatics and Automation
Problems of NAS RA
Yerevan, Armenia
e-mail: hsahakyan@sci.am

https://doi.org/10.62343/csit.2024.2

Abstract — **This paper investigates a simple state-agent-environment model of Artificial Intelligence theory. The general framework of AI in this part provides search and planning and is based on the development of modelling mathematical formalisms with a consequent validation-check, and pathfinding stages. The application problem and the use case considered in this paper belongs to the domain of personalized medicine. The problem is known as the dynamic treatment regime (DTR), and we demonstrate that in addition to traditional AI techniques for such problems, a simpler graph theoretical development can be incorporated that solves the validation and pathfinding problems with less efforts and complexities. The results obtained are achieved by using graph connectivity checking algorithms, algorithms of pathfinding in rooted trees, and checking additional graph components to be a one dicycle cactus graph; and provide a means of consecutive restructuring the whole graph for the connectivity property that preserves the validated DTR policy.**

Keywords — **Pattern recognition, dynamic treatment regime, target class, graph theory, artificial intelligence**.

State-action-environment (SAE) based models are significant part of the artificial intelligence (AI) [1]. Although the strategies required to solve such problems are well investigated, the model itself is diverse in its nature, being considered either fragmentary, fleetingly, or too abstract to tackle in a systematic and flexible applicable analysis. The most commonly used AI techniques in this area are smart search strategies, constraint solving approaches, and planning strategies and algorithms. Early SAE formalisms and languages such as satisfiability modulo theories (SMT), and PDDL (Planning Domain Definition Language), as well as many later tools that serve these models, allow users to describe algorithms and use a type of exhaustive search and planning strategy, usually breadth first search, to check if that model can solve the defined problem. In formal basis, SMT is a generalization of Boolean satisfiability problem (SAT) to more complex formulas involving not only Boolean, but also real numbers, integers, and/or various data structures such as lists, arrays, bit vectors, and strings.

SAE models are very common. They are specific in game and control theory, optimization and various model validation problems. This part of artificial intelligence considers problems related to the hardness of selection of options and overcoming various intractability of tasks. But it is possible that the task under consideration is not typically difficult, in which case, there is no need to use full AI tools. Theoretical analysis is possible and suffices. Below we look at one problem of this type and show how to model it using graph theory and how to solve it using combinatorial algorithms of acceptable complexity.

Adaptive treatment strategies are emerging as a new paradigm for the treatment and long term management of disorders such as alcoholism, smoking cessation, depression and hypertension [2]. In adaptive treatment strategies, the treatment type is repeatedly adjusted according to the individual's medical conditions. Adaptive treatment strategies are frequently called dynamic treatment regimes. DTR is composed of a sequence of decision rules, one per treatment admission. Currently, scientists try to use hard combination of *clinical experiences, behavioral, psychosocial and biological theories and randomized experimental studies* designed to formulate the decision rules composing adaptive treatment strategies. A general goal of the DTR domain is to obtain optimized adaptive treatment strategies for sets of diseases, that is, to produce a treatment strategies that yield the best mean value of the outcome. In this paper, we consider a completely different objective that is a new research topic in the DTR domain. We continue from the point where the DTR strategy (policy) is already adopted, and the issue is in applying logical analyses, to generate *validation analyses* and *recommended improvement* of the policy when needed.

DTR graph theoretical validation. A SAE model can allow a solution (path) to be discovered, but it can also fail to do so even if the problem has a solution. The reasons why the model may fail are various, starting from an insufficiently detailed representation for a state, not including all required validations and transitions, even up to not considering particular search strategies which may not be able to find a solution even if it exists. We propose validating a DTR model by checking whether three claims (implemented as logical formulas) can be proven as valid:

- Final state validation (FSV): a valid final state exists, which satisfies the conditions for reachability from an arbitrary initial state.
- Path to the final state (PFS): if FSV is true, a path between the initial state and final state is constructed using valid transitions described in the model.
- Analysis and correcting defects (ACD): if FSV is false, generate recommendations in the form of necessary transitions for a true FSV.

Considering the DTR problem in terms of SAE, our work provides results that rely on and expand the following:

Proposition [3]: Transition Graph G of a deterministic SAE for DTR consists of one tree, rooted at the goal vertex v_0, and may have several other connectivity components structured as one cycle directed cactus graphs.

REFERENCES

1. Stuart J Russell and Peter Norvig. Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited,, 2016.
2. Susan A Murphy, “An experimental design for the development of adaptive treatment strategies”, Statistics in medicine, 24(10):1455–1481, 2005.
3. L. Aslanyan, K. Gishyan, and H. Sahakyan, “Deterministic recursion to target class classification,” Pattern Recognition and Image Analysis, 33, 584–598 (2023).

Cloud Infrastructure to Perform Distributed Multi-user Platform for Self-organizing Uav Swarms

Agit Atashyan
Institute for Informatics and Automation
Problems of NAS RA
Excellicon Corporation
Yerevan, Armenia
e-mail: agitatashyan1@gmail.com

Artyom Lazyan
Institute for Informatics and Automation
Problems of NAS RA
Yerevan, Armenia
e-mail: artyomlazyan@gmail.com

Davit Hayrapetyan
Institute for Informatics and
Automation Problems of NAS RA
Yerevan, Armenia
e-mail: hayrapetyan96@gmail.com

Hrachya Astsatryan
Institute for Informatics and Automation
Problems of NAS RA
Yerevan, Armenia
e-mail: hrach@sci.am

Vahagn Poghosyan
Institute for Informatics and Automation
Problems of NAS RA
Synopsys Armenia CJSC
Yerevan, Armenia
e-mail: povahagn@gmail.com

Suren Poghosyan
Institute for Informatics and Automation
Problems of NAS RA
Yerevan, Armenia
e-mail: psuren55@yandex.ru

https://doi.org/10.62343/csit.2024.3

Abstract — **Unmanned aerial vehicle (UAV) swarms offer a resource-effective and time-efficient method of collecting and analyzing data for a range of applications. The paper presents a cutting-edge collective artificial intelligence-powered self-organizing UAV swarm mission planning and preparing platform designed to enhance task performance with a fleet of UAVs and facilitate terrain monitoring. Users can create requests easily from the QT interface due to the cloud-based multi-user platform's interactive capabilities, which enable smooth user collaboration and real-time video viewing for group study of dynamic landscape imagery. Optimizing the behavior and performance of UAV swarm navigation maps, the UAV map configurator makes it easier to create and modify them. Additionally, the QT service layer guarantees safe data transfer to cloud servers, and the parameter gossip system encourages coordination and communication among swarm members. This integrated data sets significant parameters such as the number of swarm participants, starting relative coordinates, and statuses (imager and/or strike), and drives the establishment of critical swarm and target jobs. The server uses complex algorithms, such as the research road graph, which is based on the rotor-router model, and the complete information exchange graph, which uses the gossip/broadcast model, to carry out these tasks. These algorithms complement each other in the server environment, allowing the UAV swarm to plan and coordinate tasks more effectively. In addition, the platform provides the smooth transfer of the created goal tasks to each member of the swarms' memory, benefiting decision-making skills and swarm performance as an entire. The QT-based user interface also provides the capability to modify the specific member task type of UAVs. This implies that UAVs can be used for specific area consideration, information gathering, transport, or, in the case of a military use, as an enemy position attacker. A server environment and QT-based interface can be used as a data sharing transponder to accomplish all of these functional changes. The task categories within the mentioned swarm shift when alterations occur in the initial transfer graph of UAVs, necessitating the inclusion of new locations for completing the modified tasks. Additionally, ensuring a unified primary administrative domain and coordinated data transmission among users across multiple platforms are crucial aspects of the suggested method. This domain ensures secure connections and offers additional security features.**

Keywords — **swarm of UAVs, gossip/broadcast models, rotor-router walk, mathematical models.**

I. INTRODUCTION

In recent years, UAVs (Unmanned Aerial Vehicles) have increasingly taken on diverse industrial and civilian roles, executing tasks that surpass human capabilities across extensive areas. These drones provide high performance, flexibility, stability, and efficiency, while preserving traditional resources. UAV swarms, configured in either ad-hoc or other formations, are essential for achieving high-demand, mission-specific objectives. Each UAV is tailored to its specific mission, offering resilience against individual drone failures, ease of device integration or removal, rapid mission execution due to parallel operations, and self-organizing capabilities.

A comparative analysis of UAVs, highlighting key features and applications of well-known drones, is presented in [1]. This study also examines UAV swarm management and control mechanisms. Protocols for UAV swarm management, including mission coordination and secure drone-to-drone communication, are discussed in [2]. Efficient swarm control methods are explored in [3] and [4]. Security aspects of UAV swarms, such as collision avoidance, self-organization, and swarm intelligence modeling, are detailed in [5] and [6]. The field extensively employs multi-agent system design, collective decision-making, and targeting algorithm development [7-9].

A UAV swarm operates as a mathematical multi-agent system, with each drone functioning independently. A set of simple, logically designed drones, configured as a cellular automaton on a connected graph, establishes a reliable framework for group decision-making. A key feature of swarm intelligence is its ability to exceed the capabilities of individual UAVs. Within a swarm, local uncertainties, errors, or malfunctions are mitigated by transferring responsibilities from compromised drones to neighboring swarm members. UAV swarm modeling, a relatively new and evolving field, investigates the collective behavior in decentralized self-organizing systems. UAV swarms are increasingly employed in topological and video surveillance, agriculture, climate monitoring, disaster management, civil security, and other emergency operations, thereby minimizing the human factor.

UAV swarm missions are divided into task and motion planning problems in robotics, employing strategies to build individual UAV trajectories. This involves collecting data on the surveyed area and identifying points of interest for flights, ranging from simple geometric paths to complex network solutions. The frequency characteristics and spatial location of the UAV swarm can be determined with high accuracy, forming a comprehensive, continuous image of the surveyed area. UAVs can operate independently while targeting and exchanging information with neighboring drones through logical links.

The development of a cloud platform for self-organizing UAV swarms, incorporating multi-agent systems (optimal gossip broadcast schemes, sandpile, and rotor-router models) and algorithms, represents a novel advancement in the field of logically linked, decentralized intelligent networks. Designing a software toolkit for self-organizing UAV swarms is complex and expensive. Therefore, integrating cloud technologies, virtual environments, and computing resources into a single platform offers realistic opportunities to overcome on-field challenges. The platform aims to reduce the time and cost of UAV swarm missions and support autonomous missions across diverse tasks and environments.

This study aims to design and analyze a secure cloud-based mathematical model for self-organizing UAV swarms. The proposed platform is intended to facilitate the deployment of adaptable, self-organizing UAV swarms in real-time, even in dynamic environments. Developing decentralized and self-organizing UAV swarms involves designing optimal and fault-tolerant schemes (gossip/broadcast models) for dynamic snapshotting and comprehensive image exchange of surveilled areas during the swarm's quasi-random walk (rotor-router model). The construction includes essential definitions, concepts, and mathematical models [10].

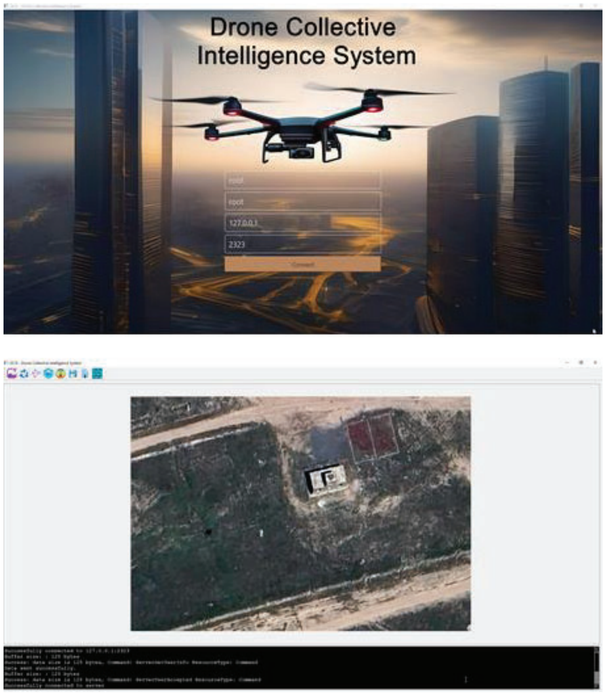
The premises for designing and implementing the cloud platform, in line with modern requirements, are based on our proposed and validated solutions for building high-performance computing infrastructures [11-12], AI-based big data gathering, classification, and processing [13-14], optimizing cloud computing environments [15], reducing energy consumption in electronic infrastructures [16], efficiently using HPC resources in linear arithmetic calculations [17], and providing cloud services [18].

II. PREPARATION ENVIRONMENT

This section discusses the simulation environment provided by the platform, including dynamic scenarios and environmental variables (see fig. 2).

2.1. Generate requests from QT environment

The Applications set is being created using C++/QT library with using Flask API's. So, everything is using QT environment to manage the toolset. The procedure for creating requests from the QT environment to power the platform's functionality is described in this section (see fig 1.).



root
root
127.0.0.1
2323

```
Successfully connected to 127.0.0.1:2323
Buffer size: : 125 bytes
Success: data size is 125 bytes, Command: [illegible] ResourceType: Command
Data sent successfully.
Buffer size: : 125 bytes
Success: data size is 125 bytes, Command: [illegible] ResourceType: Command
Successfully connected to server
```

Fig. 1. User interface for desktop application.

The preparation platform must be started by an active server, and programs must automatically establish connections. Users submit location photographs to the server, which opens an input window with a map based on the coordinates of the image [18-19]. When users enter the necessary coordinates, the system determines the real-world coordinates for each pixel, guaranteeing accuracy when setting up the simulation environment. The QT service layer employs strong encryption to protect against assaults while transferring data to virtual servers in the cloud architecture over secure Internet TCP protocol communication channels. Users can design and manage maps for UAV swarm navigation and tasks with the help of the UAV map graph module. The JSON format of all requests guarantees effective and transparent communication between the platform and the QT environment. Using the computed absolute coordinates of the terrain image's pixels, users create a flight operation network for the UAV swarm. Precise planning for UAV operations is made easier by the classification of vertices into Corner, Side Border, and Inner types. After that, users enter the IP port of the drone to initiate communication and choose particular side vertices for UAV installation. The system creates coordinates and navigation data, which are put into the UAV ground station when the return router algorithm clears any network cycles. With the correct coordinates, the UAVs can navigate the network on their own without assistance from a human. Target locations for UAV strikes can be marked by users on the terrain image, with internal geographic coordinates. Users can change the provided default network topology as needed. All changes are recorded in a thorough log, which instantly notifies the cloud server and keeps all users' graphical interfaces in sync.

2.2. Cloud infrastructure

In the domain of logically interconnected and decentralized intelligent networks, the development of a cloud-based platform for mission preparation for self-organizing UAV swarms utilizing multi-agent systems-such as sandpile models, rotor-router models, and optimal gossip broadcast schemes-represents a significant innovation (see fig 2.).

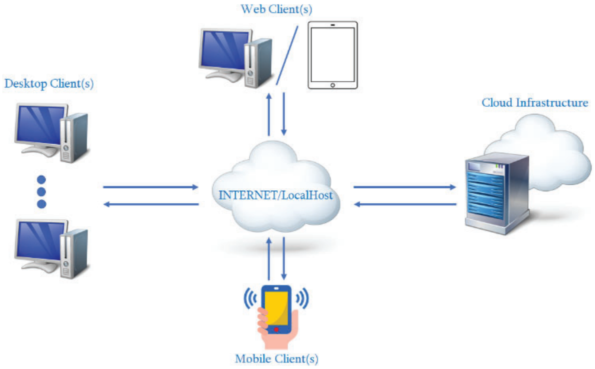


Fig. 2. Cloud infrastructure.

The creation of software toolsets for managing self-organizing UAV swarms is notably challenging and costly. Therefore, integrating virtual environments, cloud technologies, and computational resources into a unified platform presents practical opportunities to address these challenges. The proposed platform aims to enable autonomous mission execution across a diverse array of activities and scenarios, simultaneously reducing the time and cost associated with UAV swarm missions [18-19]. Our proposed and validated solutions for constructing high-performance computing infrastructures form the basis for the design and implementation of this cloud platform, adhering to contemporary standards [11], AI-powered collection, categorization, and processing of massive data [15], improving electronic infrastructure energy usage and cloud computing settings [16], utilizing HPC resources for linear algebra computations in an efficient manner [15] and cloud service disposal. Utilizing scalable computational resources and substantial storage capacity, cloud computing has significantly enhanced the efficiency of UAV image-processing operations [10]. Our approach employs a serverless cloud platform for high-performance computing (HPC), meticulously designed to manage the HPC workloads of the UAV swarm effectively, thereby ensuring the timely execution of swarm operations [16]. This platform leverages Kubernetes and container technologies, such as Docker and Singularity, to deploy and manage containerized applications. This approach facilitates the rapid and efficient completion of resource-intensive operations, eliminating the need for traditional, complex HPC infrastructure installation. By optimizing resource management and minimizing the potential for constraint violations, this method enhances the operational capabilities of the UAV swarm. Within the cloud infrastructure, a server execution environment is established, with a dedicated server allocated for executing swarm or single UAV flight operations. The IP address of this server is recorded in a working log file, which is accessible to users through a graphical user interface. Specific tasks are assigned to each server, initiating data processing and ensuring that results are visualized and synchronized. This configuration enhances the distributed and efficient execution of UAV flight operations and data processing.

III. MISSION PREPARETION HIERARCHICAL SYSTEM

The system is structured with a four-tier security hierarchy, wherein each tier grants specific access permissions and operational functionalities.

Tier 4: Observation-Only Access

User Group: General Clients Permissions: Limited to observing the graph editing workspace. Device Compatibility: Accessible via smartphones, tablets, and computer browsers. Functionality: Users can only view the workspace without altering the graph, ensuring data integrity and preventing unauthorized changes.

Tier 3: Basic Modification Access

User Group: Intermediate Clients Permissions: Authorized to make basic modifications to the graph workspace. Capabilities: Users can add or remove nodes and edges within the graph. Purpose: Facilitates structural adjustments to the graph while maintaining control over modifications.

Tier 2: Advanced Modification Access

User Group: Privileged Clients Permissions: Granted advanced editing capabilities, including strategic modifications. Capabilities: Users can introduce attacking drones and specify their targets, with actions at this level concealed from lower-tier users to uphold strategic confidentiality. Functionality: Supports complex operations and strategic maneuvers within the system, ensuring heightened control and operational security.

Tier 1: Administrative Access

User Group: System Administrators Permissions: Unrestricted root-level access. Capabilities: Administrators can approve all graph changes, upload maps indicating drone placement, and initiate missions. Responsibility: Ensures comprehensive oversight and control over the system, upholding operational integrity and security.

This structured hierarchy delineates distinct roles and responsibilities, bolstering security and operational efficacy. By segregating access levels, the system ensures that sensitive operations and data remain accessible solely to authorized personnel, safeguarding against unauthorized interventions. Each tier builds upon the permissions of the preceding one, establishing a logical progression of access and control. This hierarchical model offers a robust framework for managing user permissions, aligning each level of access with the user's role and responsibilities.

The tiered access system optimizes both security and functionality, preserving the system's integrity while accommodating necessary operational flexibility. This scientific approach to access management ensures precise delineation of capabilities for each user level, fostering a secure and efficient operational environment.

IV. CONCLUSION

Utilizing UAV swarms offers a cost-effective and efficient method for data collection and analysis across various industries. This study introduces a new self-organizing UAV swarm mission preparation platform. Through collaborative data exchange, crucial tasks like determining swarm size and coordinates are efficiently managed. This platform, powered by advanced algorithms, ensures seamless planning and coordination within the UAV swarm.

Similarly, implementing a hierarchical security system with four levels provides a structured approach to access management and operational integrity. From basic observation to advanced administrative control, clear roles and permissions are defined, safeguarding sensitive operations and data. Each level builds upon the previous one, enhancing security and operational efficiency. This approach optimizes security while allowing necessary operational flexibility, promoting a secure and efficient environment for UAV swarm missions.

ACKNOWLEDGMENT

The research was supported by the Science Committee of the Republic of Armenia within the frames of the research project 21AG-1B052.

REFERENCES

1. A. Tahir, J. Böling, M.H. Haghbayan, H.T. Toivonen, J. Plosila,“Swarms of unmanned aerial vehicles - A survey”, Journal of Industrial Information Integration, 16, 100106 (2019).
2. G. Varela, P. Caamaño, F. Orjales, A. Deibe, F. Lopez-Pena, R.J. Duro, “Swarm intelligence-based approach for real time UAV team coordination in search operations”, Third World Congress on Nature and Biologically Inspired Computing, pp. 365-370, IEEE (2011, October).
3. X. Zhu, Z. Liu, J. Yang, “Model of collaborative UAV swarm toward coordination and control mechanisms study”, Procedia Computer Science, 51, pp. 493-502 (2015).
4. L. He, P. Bai, X. Liang, J. Zhang, W. Wang, “Feedback formation control of UAV swarm with multiple implicit leaders”, Aerospace Science and Technology, 72, pp. 327-334 (2018).
5. R.N. Akram, K. Markantonakis, K. Mayes, O. Habachi, D. Sauveron, A. Steyven, S. Chaumette, “Security, privacy and safety evaluation of dynamic and static fleets of drones”, IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), IEEE pp. 1-12 (2017, September).
6. C. Liu, S. Sun, C. Tao, Y. Shou, B. Xu, “Sliding mode control of multi-agent system with application to UAV air combat”, Computers & Electrical Engineering, 96, 107491 (2021).
7. M. Campion, P. Ranganathan, S. Faruque, “A review and future directions of UAV swarm communication architectures”, IEEE international conference on electro/information technology (EIT), pp. 0903-0908, IEEE (2018, May).
8. R.K. Sharma, D. Ghose, “Collision avoidance between UAV clusters using swarm intelligence techniques”, International Journal of Systems Science, 40(5), pp. 521-538 (2009).
9. S. Poghosyan, Y. Alaverdyan, V. Poghosyan, S. Abrahamyan, A. Atashyan, H. Astsatryan, Y. Shoukourian, “Certain methods for investigating epidemics and preventing the spread of viruses in self-organizing systems”, AIP Conference Proceedings (Vol. 2757, No. 1), AIP Publishing, (2023, May)
10. H. Astsatryan, Y. Shoukourian, V. Sahakyan, “The armcluster project: brief introduction. In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications”, PDPTA'04, pp. 1291-1295 (2004).
11. H. Astsatryan, V. Sahakyan, Y. Shoukourian, J. Dongarra, P.H. Cros, M. Dayde, P. Oster, “Strengthening compute and data intensive capacities of Armenia”, 14th RoEduNet International Conference. Networking in Education and Research (RoEduNet NER) pp. 28-33, IEEE (2015, September).
12. H. Astsatryan, H. Grigoryan, A. Poghosyan, R. Abrahamyan, S. Asmaryan, V. Muradyan, G. Giuliani, “Air temperature forecasting using artificial neural network for Ararat valley”, Earth Science Informatics, 14, pp. 711-722 (2021).
13. H. Astsatryan, G. Gevorgyan, A. Knyazyan, A. Mickaelian, G.A. Mikayelyan, “Astronomical objects classification based on the Digitized First Byurakan Survey low-dispersion spectra”, Astronomy and Computing, 34, 100442 (2021).
14. A. Kocharyan, B. Ekane, B. Teabe, G.S. Tran, H. Astsatryan, D. Hagimont, “A remote memory sharing system for virtualized computing infrastructures”, IEEE Transactions on Cloud Computing, 10(3), pp. 1532-1542 (2020).
15. H. Astsatryan, W. Narsisian, A. Kocharyan, G. Da Costa, A. Hankel, A. Oleksiak, “Energy optimization methodology for e-infrastructure providers”, Concurrency and Computation: Practice and Experience, 29(10), e4073 (2017).
16. H. Astsatryan, V. Sahakyan, Y. Shoukouryan, M. Daydé, A. Hurault, R. Guivarch, L. Hovhannisyan, “On the easy use of scientific computing services for large scale linear algebra and parallel decision making with the P-grade portal”, Journal of grid computing, 11, pp. 239-248 (2013).
17. H. Astsatryan, W. Narsisian, S. Asmaryan, “SWAT hydrological model as a DaaS cloud service”, Earth Science Informatics, 9, pp. 401-407 (2016).
18. Poghosyan, V., Poghosyan, S., Lazyan, A., Atashyan, A., Hayrapetyan, D., Alaverdyan, Y., & Astsatryan, H. (2023). Self-Organizing Multi-User UAV Swarm Simulation Platform. Programming and Computer Software, 49(Suppl 1), S7-S15
19. S. Poghosyan, V. Poghosyan, S. Abrahamyan, A. Lazyan, H. Astsatryan, Y. Alaverdyan and K. Eguiazarian, "Cloud-based mathematical models for self-organizing swarms of UAVs: design and analysis," Drone Systems and Applications, vol. 12, pp. 1-12, 2024.

Neural Networks vs. Traditional Techniques: Differential Equation Approaches in Quantum Gate Analysis

Romik Sargsyan
Russian-Armenian University
Yerevan, Republic of Armenia
e-mail: romik.sargsyan@student.rau.am

Roman Sahakyan
Russian-Armenian University,
Yerevan, Republic of Armenia
e-mail: roman.sahakyan@student.rau.am

Emil Gazazyan
Institute for Physical Research of the National Academy of Sciences of the Republic of Armenia
Institute for Informatics and Automation Problems of National Academy of Science,
Yerevan, Republic of Armenia.
e-mail: emil@quopt.net

Shushanik Sargsyan
Institute for Informatics and Automation Problems of the National Academy of Sciences of the Republic of Armenia,
Yerevan, Republic of Armenia.
e-mail: shush_sarg@mail.ru

Sevak Sargsyan
Centre of Advanced Software Technologies of Russian Armenia University,
Yerevan, Republic of Armenia
e-mail: sevak.sargsyan@rau.am

Hrachya Astsatryan
Institute for Informatics and Automation Problems of the National Academy of Sciences of the Republic of Armenia,
Yerevan, Republic of Armenia.
e-mail: hrach@sci.com

https://Doi.org/10.62343/csit.2024.4

***Abstract* — The research evaluates the traditional numerical algorithms and neural network-based methods for solving ordi-nary differential equations. The evaluation covers both mathe-matical and machine learning perspectives. Practical experi-ments are carried out by implementing neural networks and the necessary algorithms using the Julia programming language. These experiments highlight significant advancements in the realm of quantum computing. Additionally, analyzing the effi-ciency metrics of solving differential equations with numerical methods versus neural networks uncovers intriguing results, which vary based on the chosen architecture and network pa-rameters.**

***Keywords* — Neural Network, differential equation, quantum simulation.**

I. INTRODUCTION

Quantum mechanical computing leverages the fundamental principles of quantum mechanics—including superposition, entanglement, tunneling, and annealing—to tackle problems beyond the reach of classical computers [1]. By utilizing quantum bits (qubits) and these quantum phenomena, quantum computers have the potential to revolutionize fields such as material science, cryptography, and optimization with unparalleled efficiency [2, 3]. Notably, quantum computing excels at processing complex differential equations more efficiently than traditional computers, enhancing the numerical solutions for these equations [4]. Integrating neural networks (NN) [5] is pivotal in advancing quantum mechanical computing, as NNs adeptly manage intricate data, optimize parameters, and enhance the precision of quantum simulations.

Neural networks enable cutting-edge technologies to process information at the molecular, atomic, and quantum dot levels. These advancements promise to transform cryptography and computational systems by enabling irreversible data processing, thereby significantly improving information security. Implementing of reversible quantum gates [6] and quantum programmable logic gates [7] supports the development of sophisticated computational architectures, essential for building robust and intricate systems. Quantum gates manipulate qubits through unitary transformations—reversible operations that precisely alter qubit states—crucial for error correction and quantum communication tasks.

A notable advancement in quantum computing is the creation of a three-bit programmable atomic gate on a five-level atom, achieved through the adiabatic transfer of atomic level populations in an M-system [8]. This adiabatic transfer method allows seamless transitions between atomic energy levels while maintaining quantum coherence, a vital requirement for dependable quantum computations. This innovation facilitates the implementation of complex quantum algorithms and the realization of practical quantum technologies. Combined with other non-reversible quantum gates [9], this approach marks a significant progression in the field.

The study focuses on the numerical solutions of non-stationary equations governing the density matrix of this five-level system. Quantum gates play a crucial role in solving complex systems of 25 differential equations, enabling a deeper understanding and modeling of dynamic processes. Specifically, the research examines the relaxation rates of ground state levels in rubidium atomic vapor through fluorescence spectra analysis, a typical regression task [10]. Rubidium vapor is particularly suited for investigating quantum phenomena, atomic clocks, and quantum computing applications. Both linear and nonlinear machine learning (ML) techniques were evaluated for their effectiveness in processing and predicting these physical processes [11]. An optimal regression model was developed, distinguished by its high accuracy and ability to significantly accelerate the modeling of critical functional indicators, thereby enhancing the prediction and analysis of physical phenomena.

The neural network approach for solving differential equations was successfully implemented using TensorFlow [12], addressing both ordinary differential equations (ODE) and partial differential equations (PDE) that do not involve complex numbers. The method demonstrated exceptional convergence, highlighting its advantages and significantly reducing errors compared to traditional techniques such as Runge-Kutta and finite difference methods [13, 14]. Additionally, this neural approach offers superior numerical stability, avoiding the limitations imposed by the Courant-Friedrichs-Lewy condition [15], and can effectively solve PDEs on very coarse grids.

Among traditional numerical methods [4], neural networks [5, 16, 17] stand out as powerful tools for solving differential equations. They are capable of modeling complex dependencies and identifying patterns within data. Applying neural networks to differential equation solving opens new avenues in dynamic system analysis, enabling highly accurate approximate solutions and achieving these results with significantly reduced computational time compared to conventional numerical methods.

The article seeks to assess traditional numerical methods and NN approaches that handle complex numbers for solving ordinary differential equations within a quantum two-level system in quantum mechanics. By comparing the effectiveness of these numerical techniques and NNs, the study highlights notable differences in computational accuracy, emphasizing the potential of neural networks in addressing intricate mathematical challenges, especially in quantum computing. These results carry significant implications for future quantum physics research, particularly in simulating quantum computations where conventional methods might restrict the speed and precision of data processing.

II. NEURAL NETWORKS FOR SOLVING ORDINARY DIFFERENTIAL EQUATION

The study concentrates on the design of the neural network architecture, including activation functions, optimization algorithms, and training strategies, to ensure precise and effective system dynamics modeling. The network architecture is meticulously crafted to manage the complexities of the quantum two-level system under investigation, as depicted in Fig. 1. It comprises an input layer, two hidden layers, and an output layer. The input layer features a single neuron, representing the one-dimensional time input characteristic of dynamic systems. Each hidden layer contains 64 neurons, providing ample capacity to handle the network's information processing demands. The output layer consists of four neurons, each directly corresponding to the four components of the system's state vector, ensuring a clear and direct relationship between the network's output and the physical states of the quantum system.

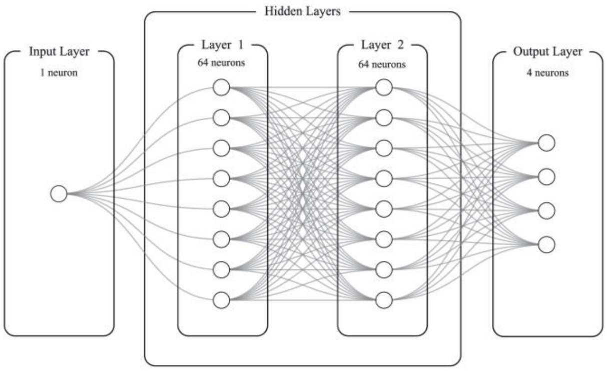


Fig. 1: NN Model Architecture

The hidden layers use a complex sigmoid function for activation, which is crucial for handling complex-valued inputs and outputs. By integrating the real and imaginary components of complex numbers, this function allows the network to accurately simulate the subtle and intricate dynamics characteristic of quantum mechanical systems.

Network optimization employs the Adam optimizer [19, 20], renowned for efficiently navigating complex optimization landscapes and achieving rapid convergence. The learning rate is set at 0.01 to balance the speed and stability of the training process.

The training regime adopts a stochastic approach, introducing randomness in selecting data points to robustly mitigate the risk of overfitting. This randomness prevents the model from excessively memorizing the training data, enhancing its generalization capabilities and ensuring adaptability to various scenarios within the quantum system. The training dataset comprises 300 carefully selected points, balancing computational efficiency with the complexity required to model the dynamic behaviors adequately. Additionally, 30,000 boundary condition points are included to ensure the network rigorously adheres to the physical constraints and boundary conditions essential for obtaining valid and meaningful solutions.

Network parameters are initialized using the kaiming_normal() function [19, 20], which scales the weights based on a calculated standard deviation. This method maintains consistent variances across neuron inputs and outputs, preserving the integrity of signal propagation through the network layers. Such initialization is particularly effective for networks handling complex values, as it helps avoid issues like vanishing or exploding gradients.

Before training, additional setup steps are performed using Lux.setup(rng, chain) [19, 20]. These steps involve crucial tasks such as parameter initialization and system configuration, ensuring that the network is optimally prepared for the training phase. This setup is vital for aligning the network's operational parameters with its architectural and strategic goals, thereby establishing a solid foundation for efficient learning and accurate performance in complex simulation tasks.

In addressing complex-valued Ordinary Differential Equations (ODEs), the study employs the NNODE approach. This method leverages the flexibility and adaptability of neural networks to approximate solutions to differential equations that are otherwise challenging for traditional numerical solvers. Integrating neural networks into the framework of differential equations offers a promising avenue for tackling problems involving complex-valued systems and intricate dynamics.

The NNODE method utilizes a neural network architecture, comprising multiple layers, to represent the solution to an ODE. The network receives the independent variables of the ODE (e.g., time) as inputs and generates predictions for the dependent variables (e.g., state vectors). A loss function, derived from the given differential equation, is defined to train the network and optimize its parameters (weights and biases). This loss function assesses how well the neural network's output satisfies the ODE across a range of input values.

The essence of the NNODE approach lies in its iterative training process, wherein the network minimizes the discrepancy between its predicted solutions and the true solutions of the ODE. The training involves three key components: the optimization algorithm [21], the loss function, and the training strategy. Various strategies can be employed to determine the points at which the ODE is evaluated during training. Techniques such as stochastic training introduce randomness in the selection process, thereby enhancing the network's generalization capabilities and preventing overfitting to specific data points or patterns.

The L2 loss function (1), commonly used in training neural networks with the NeuralPDE.jl package, is calculated based on the squared L2 norm between the network's predictions and the target values:

$$L2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad (1)$$

where y_i is the target value, $\hat{y}_i$ is the corresponding predicted value, and n is the number of observations.

III. NEURAL NETWORKS FOR QUANTUM SIMULATIONS

The study investigates the dynamics of a two-level quantum system, which is distinguished by its capacity to exist in a superposition of two distinct quantum states. These systems, referred to as qubits, are fundamental for comprehending the principles of quantum mechanics. To analyze their behavior, mathematical methods based on linear differential equations in two-dimensional spaces are employed. Specifically, the optical Bloch equations [22, 23] are utilized to model the dynamics of atoms and quantum bits under various external field interactions. These equations incorporate the effects of field interactions alongside quantum mechanical properties such as coherence and superposition.

The optical Bloch equations governing the system are expressed as follows:

$$\begin{cases} \frac{d\rho_{11}}{dt} &= \frac{i\Omega}{2}(\rho_{12} - \rho_{21}) + \Gamma\rho_{22} \\ \frac{d\rho_{22}}{dt} &= -\frac{i\Omega}{2}(\rho_{12} - \rho_{21}) - \Gamma\rho_{22} \\ \frac{d\rho_{12}}{dt} &= -(\gamma + i\Delta)\rho_{12} - \frac{i\Omega}{2}(\rho_{22} - \rho_{11}) \\ \frac{d\rho_{21}}{dt} &= -(\gamma - i\Delta)\rho_{21} + \frac{i\Omega}{2}(\rho_{22} - \rho_{11}) \end{cases} \qquad (2)$$

where the matrix elements ρ_{ij} represent the density operator of the system, Ω denotes the Rabi frequency, Γ is the natural decay rate of the excited state, indicating how swiftly the system returns to its ground state in the absence of external disturbances. γ describes the transverse decay rate, capturing the effects of environmental noise and interactions that cause the degradation of quantum information stored within the system. Δ is the detuning parameter, representing the difference between the external field frequency and the natural frequency of the quantum transition.

Tables 1, 2, and 3 provide detailed data on the values of the loss function (L2) and correlation coefficients when solving differential equations using neural networks (NN) for various parameters Δ and Ω. The loss function values reflect the error between predicted and actual solutions, while the correlation coefficients measure the similarity between the predicted and actual quantum states. Varying the values of Δ and Ω allows for the examination of how different parameters affect the accuracy and correlation of neural networks in quantum simulations.

	Δ = -1	Δ = 0	Δ = 1
Loss function (L2)(2)	0.0000875	0.000805	0.0000869
The correlation coefficients between ρ_{11}^{n} and ρ_{11}^{p}	0.99829	0.9993768	0.9990431
The correlation coefficients between ρ_{22}^{n} and ρ_{22}^{p}	0.99834	0.9994245	0.9990781

Table 1 For Ω=3Γ and Δ loss function and correlation

	Δ = -1	Δ = 0	Δ = 1
Loss function (L2)(2)	0.0003614	0.00028854	0.000315279
The correlation coefficients between ρ_{11}^{n} and ρ_{11}^{p}	0.99613571	0.99775968	0.996612334
The correlation coefficients between ρ_{22}^{n} and ρ_{22}^{p}	0.99637116	0.99783555	0.996645611

Table 2 Ω=4Γ and Δ loss function and correlation

	Δ = -1	Δ = 0	Δ = 1
Loss function (L2)(2)	0.001605311	0.001269584	0.0012727
The correlation coefficients between ρ_{11}^{n} and ρ_{11}^{p}	0.989457997	0.990168838	0.9909499
The correlation coefficients between ρ_{22}^{n} and ρ_{22}^{p}	0.989868546	0.990391949	0.9910183

Table 3 Ω=5Γ and Δ loss function and correlation

These data enable the assessment of the accuracy and efficiency of NN methods compared to traditional numerical methods, such as the Runge-Kutta method. The loss function (L2) is used to measure the accuracy of the neural network's predictions relative to benchmark values, where low values of the loss function indicate high prediction accuracy. In the presented tables, the loss function values vary depending on the parameters Δ and Ω. Correlation coefficients are used to evaluate the degree of correspondence between the values obtained using NNs (ρ_{11}^{p} and ρ_{22}^{p}) and the values obtained by traditional numerical methods (ρ_{11}^{n} and ρ_{22}^{n}). High correlation coefficients indicate a strong relationship between the two data sets and confirm that neural networks can accurately model the behavior of complex systems.

The obtained data demonstrate the high accuracy of NN methods in solving differential equations and show a strong correspondence with the results of traditional numerical methods. This indicates that neural networks are a reliable tool for modeling complex quantum systems, providing high accuracy and efficiency compared to classical approaches.

IV. CONCLUSION

This work presents an in-depth comparative analysis of algorithms from traditional numerical methods and methods using NNs that handle complex numbers and the choice of activation functions for solving ordinary differential equations in quantum computing. Experiments were conducted during the study, demonstrating the efficiency of both approaches depending on the characteristics of the architecture and network parameters. The efficiency metrics of numerical methods and NNs indicate significant differences in accuracy and computational speed, highlighting the potential of NNs in solving complex mathematical These findings could have substantial implications for future research in the fields of physics and engineering, where traditional problems, especially in the field of quantum computing. Methods may limit the speed and accuracy of data processing.

ACKNOWLEDGMENT

The experiments utilized computational resources from the Armenian cloud infrastructure [24]. Armenian cloud provided access to nearly 700 vCPUs. The Higher Education and Science Committee of RA supported the work in the frames of projects N 1-6/IPR and IIAP 1-8/24-I/IIAP

REFERENCES

1. R. P. Feynman, “Quantum mechanical computers,” Foundations of Physics, vol. 16, pp. 507–531, Jun 1986.
2. U. R. Karpuzcu, ‘Special Issue on Quantum Computing’, *IEEE Micro*, vol. 41, no. 5, pp. 6–7, 2021.
3. D. Richart, W. Laskowski, Y. Fischer, and H. Weinfurter, ‘Experimental analysis of qudit entangled states using the time-energy degree of freedom’, in 2013 Conference on Lasers & Electro-Optics Europe & International Quantum Electronics Conference CLEO EUROPE/IQEC, 2013, pp. 1–1.
4. C. Rackauckas and Q. Nie, “DifferentialEquations.jl–a performant and feature-rich ecosystem for solving differential equations in Julia,” Journal of Open Research Software, vol. 5, no. 1, 2017.
5. K. Zubov, Z. McCarthy, Y. Ma, F. Calisto, V. Pagliarino, S. Azeglio,L. Bottero, E. Luj´an, V. Sulzer, A. Bharambe, N. Vinchhi, K. Balakrishnan, D. Upadhyay, and C. Rackauckas, “Neuralpde: Automating physics-informed neural networks (pinns) with error approximations,” 2021.
6. G. Grigoryan, V. Chaltykyan, E. Gazazyan, and O. Tikhova, “All-optical four-bit toffoli gate with possible implementation in solids,” Proc SPIE, vol. 8772, 05 2013.
7. E. A. Gazazyan, G. G. Grigoryan, V. O. Chaltykyan, and D. Schraft, “Implementation of all-optical toffoli gate in λ-systems,” Journal of Contemporary Physics (Armenian Academy of Sciences), vol. 47, pp. 216–221,Sep 2012.
8. A. Y. Aleksanyan and E. A. Gazazyan, “Realization of the programmable logical atomic gate,” Journal of Contemporary Physics (Armenian Academy of Sciences), vol. 53, pp. 205–211, Jul 2018.
9. E. A. Pogosyan and E. A. Gazazyan, “Programmable quantum gate operations using qutrit quantum dots,” Optical Memory and Neural Networks, vol. 32, pp. S396–S401, Dec 2023.
10. A. Aleksanyan, S. Shmavonyan, E. Gazazyan, A. Khanbekyan, H. Azizbekyan, M. Movsisyan, and A. Papoyan, “Fluorescence of rubidium vapor in a transient interaction regime,” Journal of the Optical Society of America B, vol. 37, p. 203, 12 2019.
11. A. A. Sargsyan, A. Y. Aleksanyan, S. A. Petrosyan, E. A. Gazazyan, A. V. Papoyan, and H. V. Astsatryan, “Prediction of atomic ground state relaxation rate from fluorescence spectra using machine learning,” Journal of Contemporary Physics (Armenian Academy of Sciences), vol. 56, pp. 285–290, Oct 2021.
12. F. Sim, E. Budiarto, and R. Rusyadi, “Comparison and analysis of neural solver methods for differential equations in physical systems,” ELKHA: Jurnal Teknik Elektro, vol. 13, no. 2, pp. 134–140, 2021.
13. Ascher, Uri M.; Petzold, Linda R. (1998), Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations, Philadelphia: Society for Industrial and Applied Mathematics, ISBN 978-0-89871-412-8.
14. Atkinson, Kendall A. (1989), An Introduction to Numerical Analysis (2nd ed.), New York: John Wiley & Sons, ISBN 978-0-471-50023-0.
15. R. Courant, K. Friedrichs, and H. Lewy, “ ¨Uber die partiellen differen-zengleichungen der mathematischen physik,” Mathematische Annalen, vol. 100, pp. 32–74, Dec 1928.
16. K. Zubov, Z. McCarthy, Y. Ma, F. Calisto, V. Pagliarino, S. Azeglio, L. Bottero, E. Luj´an, V. Sulzer, A. Bharambe, et al., “Neuralpde: Automating physics-informed neural networks (pinns) with error approximations,” arXiv preprint arXiv:2107.09443, 2021.
17. E. Shi and C. Xu, “A comparative investigation of neural networks insolving differential equations,” Journal of Algorithms & Computational Technology, vol. 15, p. 1748302621998605, 2021.
18. J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, “Julia: A fresh approach to numerical computing,” SIAM Review, vol. 59, no. 1, pp. 65–98, 2017.
19. M. Innes, “Flux: Elegant Machine Learning with Julia”, Journal of Open Source Software, 2018.
20. M. Innes et al., ‘Fashionable Modelling with Flux’, CoRR, vol. abs/1811.01457, 2018.
21. V. K. Dixit and C. Rackauckas, “Optimization.jl: A unified optimization package,” Mar. 2023.
22. C. J. Foot, Atomic physics. Oxford: Oxford University Press, 2007.
23. R. Loudon, The Quantum Theory of Light. Oxford science publications, Clarendon Press, 1973.
24. [24] D. Petrosyan, H. Astsatryan. “Serverless high-performance computing over cloud." Cybernetics and Information Technologies vol. 22, no. 3, pp. 82-92, 2022.

Privacy-Preservation and Data-Sharing Security Issues in Banking Applications

Nikoloz Katsitadze
University of Georgia
Tbilisi, Georgia
e-mail: n.katsitadze@ug.edu.ge

Adeyinka Olanrewaju Ajagbe
University of Georgia
Tbilisi, Georgia
e-mail: ajagbe.adeyinka@yahoo.com

https://doi.org/10.62343/csit.2024.5

***Abstract* — The digital transformation of the banking sector has brought forth unprecedented convenience for users but has also exposed financial institutions to a myriad of cyber threats. Maintaining user data security and privacy has become critical as digital banking develops. This research conducts a comprehensive analysis of security incidents in prominent digital banking institutions. Common patterns and distinctive features surrounding the preservation of user data privacy are identified through a cross-examination of these cases.**

The study unveils recurring themes and unique characteristics in these security incidents, and a subsequent cross-case analysis underscores that security breaches often implicate the three integral components of institutions: People, Processes, and Systems.

This research contributes to the ongoing discourse surrounding digital banking security by providing valuable insights into prevalent vulnerabilities. Moreover, it proposes avenues for the formulation of robust security strategies, accompanied by recommendations for further research. As digital banking continues to play a crucial role in financial transactions, the findings aim to inform future practices that prioritize the protection of user data and the integrity of digital banking systems.

***Keywords* — Privacy-preservation, data-sharing security issues, banking applications, *digital transformation*.**

I. INTRODUCTION

Financial services have rapidly evolved and moved from the era of historical brick-and-mortar institutions to digital finance. This change has led to a change in the way individuals and businesses manage their finances. With the increasing transition to online platforms, ensuring the confidentiality of the users' information is paramount. Modern banking applications offer convenient services such as funds transfer, bill payment, accounts management, finance management, etc. thereby transforming the financial space into a seamless and user-friendly experience.

The advent and convenience of digital banking via applications have birthed lots of security challenges. Cyber threats, ranging from phishing attacks to sophisticated malware, data breaches because of unsecured applications, and non-adherence to regulatory framework and privacy policies, pose a constant risk to user data and financial assets. The importance of robust security measures is magnified as financial transactions are being consummated via networks, emphasizing the need for encryption, secure authentication mechanisms, and proactive threat detection.

II. PRIVACY CONCERNS AND REGULATORY FRAMEWORK

Privacy is integral to banking applications as they handle sensitive user information, including personal details and financial histories. The regulatory landscape, marked by stringent data protection laws and financial regulations, necessitates a robust framework for ensuring user privacy. Compliance with regulations such as GDPR (for countries in the European Union-EU), CCPA, and financial industry standards is not only a legal obligation but also a means of building user trust.

Security breaches and privacy lapses can have severe consequences for banking institutions. Users entrust their financial well-being to these applications, making trust a currency as valuable as the transactions themselves. A secure and privacy-respecting banking application not only safeguards users but also contributes to the institution's reputation and customer loyalty.

While stringent security measures are imperative, achieving a balance with a seamless user experience is equally crucial. Cumbersome security processes may deter users, highlighting the importance of implementing solutions that are both robust and user-friendly. Multi-factor authentication, biometric verification, and continuous monitoring are examples of technologies that enhance security without compromising usability.

III. RATIONALE FOR THE STUDY

The rationale for this study is rooted in the points highlighted below:

I. Increased focus on data privacy: Data privacy has become a primary concern for financial institutions due to increased awareness from regulatory bodies, general awareness, and cyber-attacks.

II. Balancing data sharing and security: Open banking raises concerns about the amount of data to be shared with other parties, and it is important to balance the importance of providing relevant data for services and reducing unwanted data sharing.

III. Transparency in data usage: Open banking necessitates transparency in how customer data is utilized, stored, and shared. Financial institutions and third-party providers must be honest about their practices, policies, and security measures in place to secure customer data.

IV. RESEARCH OBJECTIVES

For the case of this research, the objectives and scope of this project are outlined below.

- Identify and analyze the privacy and data sharing security issues in both open banking and digital banking.
- Compare the security and privacy challenges faced by financial institutions in implementing these two banking models, by Identifying patterns, similarities, and differences, and providing a comprehensive view of their respective strengths and weaknesses.
- Recommend solutions to address the identified security and privacy issues in open banking and digital banking.
- Provide insights and recommendations into how financial institutions can strike a balance between data sharing for innovative services and maintaining customer privacy.

Help people comprehend the significance of data privacy in relation to banking technology changes and laws, like the General Data Protection Regulation (GDPR).

V. SCOPE AND LIMITATIONS OF THE RESEARCH

Scope of the Study:

- Comparison of privacy and data sharing security issues in open banking and digital banking.
- An examination of the privacy and security issues that financial organizations encounter when putting these two banking models into practice.
- The identification of viable remedies to security and privacy concerns in digital and open banking.
- Exploration of the balance between data sharing for innovative services and maintaining customer privacy

Limitations of the Study:

- Limited Access to Internal Information
- Reliance on Historical Data
- Assumption of Accurate Public Information
- Methodological shortcomings
- User-Specific Context

VI. RESEARCH METHODOLOGY

This study will be built upon the application of qualitative research methodology. To support the perceptions, information from libraries and the internet will be used as a secondary data collecting type in this study.

One essential component of the process is the instances to be used. The cases will center on privacy and security, and the best ways to protect them in financial applications. This methodology attempts to produce an inventive, context-rich inquiry that leads to a thorough understanding of Privacy-Preservation and Data-Sharing Security Issues in Banking Applications by utilizing the strengths of the case study method.

Case Study Research Design: The cases used for this project provide ample information on the Privacy-Preservation and Data-Sharing Security Issues in Banking Applications, security, and policy concerns, and many more. The importance of privacy preservation cannot be over-emphasized, as its relationship ensuring the security of users' data.

Techniques for Gathering Data: Examining various data breach scenarios can reveal trends and similarities throughout firms, along with the opportunities and problems they encountered.

Approaches to Data Analysis: Upon completion of the study, a cross-case analysis will be conducted to determine whether existing security vulnerabilities are similar or different, and to determine the best ways to protect users' privacy when using banking applications.

VII. COMPARATIVA ANALYSIS

Upon cross-examination of the cases of Westpac, Bank of Ireland, Cash App, Flagstar Bank, UniCredit, and JPChase Morgan and Co., we uncovered common patterns, as well as distinctive features concerning the security issues bordering the preservation of the privacy of users' data that occurred during the incidences highlighted in this research work.

Common Themes:

- Personal Identifiable Information (PII) Exposure: In all the cases reviewed, significant amounts of personally identifiable information (PII) were exposed, ranging from names, addresses, phone numbers, and email addresses, investment portfolios, etc. This raised concerns about the potential for identity theft and other malicious activities.
- Data Privacy and Protection: Every example highlights how important it is to have strong security measures in place to safeguard personal data. The dangers of gathering and keeping a lot of personal data were made clear by the hacks.
- Reputational Damage: The breaches caused harm to the organizations' reputations, which persists to some extent even now.
- Lack of encryption for sensitive data: Sensitive data as personally identifiable information (PII), should be encrypted. This would, as much as possible, ensure that bad actors, even if they get access to it, may not be able to make any use of it.
- Failure to Use multi-factor authentication (MFA): MFA is a security feature that provides an additional degree of protection. The security breaches made clear how crucial it is to use multi-factor authentication (MFA) to stop unwanted access even in the case that credentials are compromised.
- Financial losses to the companies and the customers involved: In all the cases reviewed, there were financial implications in the form of loss of funds to the customers and fines incurred by the companies.

Distinctive Features:

- Type of Data Stolen: In the Bank of Ireland case, the data revealed was not to bad actors. The internal issues in the system made the wrong customer’s details appear when a user logs into their banking application, which has the potential for fraud to occur. All the other cases, however, had some form of users’ private information exposed.
- Insider Threats: The Cash App and Desjardins' cases stood out as those that emanated from the inside, as against the others which were attempts from external parties. These kinds of threats are the hardest to identify.

- Third-Party Vulnerabilities: Westpac's case was because the third-party company, PayID was compromised, which by extension affected them.
- Incorrect Customer Information Management System Update: The flaw that existed in the customer information system of the Bank of Ireland made a different customer's information appear on the bank's mobile application when a customer logged into it.
- Access Control policy lapses: Still in Westpac's case, even if the ex-staff had access to the system, access to personal information such as an investment portfolio should require levels of access to view and authorization to download.
- Regulatory Impact: The Bank of Ireland case violated data privacy policies, which helped the European Union implement the General Data Protection Regulation (GDPR). They were heavily fined for the lapses as well.

Cross-case analysis. A review of all these cases brings us to the understanding of the following:

- All institutions are made of three components: People, Processes, and Systems.
- A great deal of breaches are because of the people (i.e. human errors). However, it is important to build systems that are easy to use and ensure that robust training policies are in place to ensure that this is curbed or reduced to the barest minimum. Most times, it is these lapses that bad actors (aka hackers) prey on to gain access to the system.
- All the security issues/ threats can either emanate internally or externally.
- Internally, the threats that emanated were from a disgruntled staff that had access to the system, while the other case was that of an ex-staff that still had access to the banking system, even after leaving the system.
- The external threats in this review were via phishing attempts, which enabled bad actors to

VIII. SUMMARY

This research examined security issues in digital banking, analyzing cases from Westpac, Bank of Ireland, Cash App, Flagstar Bank, UniCredit, and JPChase Morgan and Co. Common themes included PII exposure, data privacy concerns, reputational damage, lack of encryption, failure to implement MFA, and financial losses. Distinctive features included the type of data stolen, insider threats, third-party vulnerabilities, access control policy lapses, and regulatory impact.

Cross-Case Analysis Findings:

- Components of Breaches: Breaches often involve the three components of institutions: People, Processes, and Systems. Human errors are significant contributors, emphasizing the importance of user-friendly systems and robust training policies.
- Internal and External Threats: Breaches may originate internally (disgruntled staff, ex-employees with access) or externally (phishing attempts). A multi-layered security strategy is vital, involving access control, encryption, internal audits, human resource management, and continuous employee training.

CONCLUSION

Safeguarding digital banking requires a comprehensive approach. The identified security issues underscore the critical need for institutions to address vulnerabilities in their people, processes, and systems. A multi-layered security strategy, including access control, encryption, and continuous training, is crucial. Rapid incident response plans and learning from past mistakes are essential for minimizing the impact of data breaches. As the digital banking space evolves, maintaining a cyber-aware office environment is paramount to ensuring the security and privacy of customer transactions.

REFERENCES

1. Ahirrao, S. (2023, May 2). What are the Major Security and Privacy Challenges in Open Banking? Ardent Privacy. https://www.ardentprivacy.ai/blog/what-are-the-major-security-and-privacy-challenges-in-open-banking/
2. Arabia, S. (2023, August 30). 5 winning strategies for digital transformation journey of banks. https://www.linkedin.com/pulse/5-winning-strategies-digital-transformation-journey-banks/
3. Biggest Data Breaches in US History (Updated 2024) | UpGuard. (n.d.). https://www.upguard.com/blog/biggest-data-breaches-us#:~:text=The%20data%20breach%20of%20Yahoo,over%20the%20next%20three%20years.
4. Burgess, M. (2020, March 24). What is GDPR? The summary guide to GDPR compliance in the UK. WIRED UK. https://www.wired.co.uk/article/what-is-gdpr-uk-eu-legislation-compliance-summary-fines-2018
5. Challenges and opportunities of digitising the customer relationship. (n.d.). https://blog.cenareo.com/en/digitising-the-customer-relationship
6. Chaurasia, N. (2023, December 5). The 10 Biggest Data Breaches of the 21st Century: Lessons in Cybersecurity. Sprintzeal.com. https://www.sprintzeal.com/blog/biggest-data-breaches
7. Coventry, S. (2023, November 6). Security Measures for Protecting Mobile Banking Users. https://www.poundsterlinglive.com/central-news-feed/148-opinion-news-feed/19470-mobile-banking-protection-11-key-security-measures
8. Data Privacy in Digital Banking | Meniga. (n.d.). https://www.meniga.com/blog/data-privacy-in-digital-banking/
9. Degrees, S. (2023, August 10). Five Biggest Data Breaches in Financial Service History: Lessons to Be Learnt. Six Degrees. https://www.6dg.co.uk/blog/biggest-data-breaches-financial-services/
10. DPC (Ireland) - IN-20-7-2. (n.d.). GDPRhub. https://gdprhub.eu/index.php?title=DPC_(Ireland)_-_IN-20-7-2
11. Fintechly (2021, July 4). Digital or Open Banking. Finextra Research. https://www.finextra.com/blogposting/20470/digital-or-open-banking
12. Fintechly (2022, June 22). Discover How Fintech's Shaping the Future of Banking. Fintechly. https://fintechly.com/financing/how-fintech-is-shaping-the-future-of-banking/
13. Frankenfield, J. (2020, November 11). General Data Protection Regulation (GDPR) Definition and Meaning. Investopedia. https://www.investopedia.com/terms/g/general-data-protection-regulation-gdpr.asp
14. GDPR Data Protection: Definitions and Practical Measures. (2023, September 27). Cloudian.

https://cloudian.com/guides/data-protection/gdpr-data-protection/

15. Issues and Challenges Associated with Data Sharing. (2016, April 29). Principles and Obstacles for Sharing Data From Environmental Health Research - NCBI Bookshelf. https://www.ncbi.nlm.nih.gov/books/NBK362423/
16. Khan, H. (2023, October 17). 8 Effective Digital Transformation Strategies in Banking. Wavetec. https://www.wavetec.com/blog/banking/digital-transformation-tips-for-banks/
17. Khon, S., Nizam, A., Tan, D., & Taraporevala, Z. (2022, July 13). Best of both worlds: Balancing digital and physical channels in retail banking. McKinsey & Company. https://www.mckinsey.com/industries/financial-services/our-insights/best-of-both-worlds-balancing-digital-and-physical-channels-in-retail-banking
18. Koch, R. (2023, September 14). Everything you need to know about GDPR compliance. GDPR.eu. https://gdpr.eu/compliance/
19. Lake, R. (2021, April 5). Increase In Digital Banking Raises Consumer Data Privacy Concerns: How To Protect Yourself. Forbes Advisor. https://www.forbes.com/advisor/banking/digital-banking-consumer-data-privacy-concerns/
20. Leuillier, A. (2023, January 20). What Is the Impact of Digitalization on Customer Service? Vocalcom. https://www.vocalcom.com/blog/impact-digitalization-customer-service/
21. Ovington, T. (2023, May 21). Digital transformation in banking: A complete guide - WalkMe Blog. WalkMe Blog. https://www.walkme.com/blog/digital-transformation-in-banking/
22. Privacy Laws for Apps: How to Protect User Data? (2023, May 23). https://cookie-script.com/blog/privacy-laws-for-apps-how-to-protect-user-data
23. Sadik, N. (2023, April 26). Revolutionizing Customer Relationship Management: The Impact of Digital Transformation and AI. https://www.linkedin.com/pulse/revolutionizing-customer-relationship-management-impact-nassim-sadik/
24. Security, I. (2023, November 13). How can you overcome security challenges in the digital era? www.linkedin.com. https://www.linkedin.com/advice/1/how-can-you-overcome-security-challenges-digital
25. Temple, A. (2023, October 25). Navigating the security challenges of digital commerce. Commercetools. https://commercetools.com/blog/navigating-the-security-challenges-of-digital-commerce
26. The digital customer relationship in financial service - Livework. (2023, September 15). Livework - English. https://liveworkstudio.com/insight/the-digital-customer-relationship-in-financial-service/
27. The Impact of Digital Transformation & Customer Experience. (2023, May 18). https://www.linkedin.com/pulse/impact-digital-transformation-customer-experience/
28. UniCredit 2015 data incident. (2019, October 28). UnicreditGroup.eu. https://www.unicreditgroup.eu/en/press-media/press-releases/2019/unicredit-2015-data-incident.html
29. What is Data Protection and Privacy? (2023, October 27). Cloudian. https://cloudian.com/guides/data-protection/data-protection-and-privacy-7-ways-to-protect-user-data/
30. Wolford, B. (2023, September 14). What is GDPR, the EU’s new data protection law? GDPR.eu. https://gdpr.eu/what-is-gdpr/
31. Yin, R. (2014). Case Study Research Design and Methods. Canadian Journal of Program Evaluation, 282.

Optimizing the Mid-Size Microservice Architecture Ecosystem

Nikoloz Katsitadze
University of Georgia
Tbilisi, Georgia
e-mail: n.katsitadze@ug.edu.ge

Zurab Chachava
University of Georgia
Tbilisi, Georgia
e-mail: chachavazuka@gmail.com

https://doi.org/10.62343/csit.2024.6

***Abstract* — Starting a new IT project requires careful planning, including understanding the business, analyzing data, and making smart decisions about how to build the project. While many are drawn to using microservices for their scalability, they can be complex, costly, and time-consuming. On the other hand, traditional monolithic architectures are simpler but can be hard to scale and maintain as the project grows.**

A good compromise is the modular monolith approach. It breaks the project into smaller parts with clear boundaries, making it easier to manage and reuse code. By using a simple communication method between these parts, called RPC-like communication, we can keep things efficient and avoid complicating the developers' work with networking details. If needed, we can later transition smoothly from a modular monolith to microservices, saving time and avoiding many bugs.

Overall, choosing practical architectural approaches like modular monoliths helps keep projects straightforward while preparing them for future growth and upkeep.

***Keywords* — Microservice architecture, optimization, monolith architecture.**

I. INTRODUCTION

Planning modern IT projects requires hard work. Understanding business domain, making analytics, choosing relevant technical project architecture are the key concepts of IT project management. The work becomes harder when the team is trying to find the best architecture solutions while dealing with business problems. Most of the time modern architect guru's advice to use microservice architecture, but is it really a good idea?

II. WHY MICROSERVICES ARE NOT THE BEST OPTION AT THE BEGINNING.

Even though microservices are one of the most scalable architectural solutions, but as always great power comes with great responsibilities. While it is a practical choice, it has many disadvantages as well: Technical complexity, high cost, time intensive.

When talking about technical complexity, it refers to following problems:

- Complexity in communication: microservices should communicate to each other, which introduces complex APIs, network congestion, latencies and so on;
- Data Management: managing data consistency across services and handling transactions among these processes can be challenging;
- Service Discovery: Keeping track of numerous services and instances for service discovery can be complex in a microservices architecture;
- Security: Ensuring secure communication between services and managing the increased attack surface due to exposed APIs requires more work;
- Monitoring and Troubleshooting: Monitoring and troubleshooting a microservices-based application can be more complex than a monolithic one due to the different programming languages and environments used for each service.

According to these problems, using microservice architecture from the beginning of the project, especially in startups, can cause fatal issues and most probably the project will not see light of the sun.

"You shouldn't start a new project with microservices, even if you're sure your application will be big enough to make it worthwhile." - Martin Fowler.

III. PROBLEM WHILE WORKING WITH MONOLITH

Another common architectural approach can be "good old" monolith. monolith refers to a software application that is built and deployed as a single unit. Therefore, all the components of the application, including the user interface, business logic, and data storage, are tightly integrated and run in a single process. This solution is not as bad as modern developers say. It gives us simplicity because all components are part of a single codebase. Most of the time, it is more performant due to avoiding additional network communication. Debugging and monitoring are also simplified when using single process.

Nothing is perfect, so that monolithic architecture also has disadvantages:

- Scalability: Monolithic applications can become difficult to scale, as the entire application needs to be scaled, regardless of the specific component that requires more resources.
- Deployment: Deploying a monolithic application can be time-consuming, as the entire application needs to be built, tested, and deployed as a single unit.
- Maintenance: As the application grows, it can become more difficult to maintain and update, as changes to one component can potentially affect other components, leading to unintended consequences.

At the end of the day, we get a big giant codebase, which hates changes, and any attempt causes big trouble and pain to the developers.

IV. THE GOLDEN EDGE SOLUTION – MODULAR MONOLITH

A modular monolith is an architectural approach that divides an application's domain into smaller, more manageable components or modules, providing a balance between monolithic and microservices architectures.

The project is segmented, and each piece contains individual features and business logic. This kind of approach promotes high cohesion and low coupling. Modules in a modular monolith communicate through well-defined interfaces, which refers to lose coupling and independent development of functionalities.

Benefits:

- High Reusability: Logic encapsulation enables high reusability while maintaining data consistency and simple communication patterns.
- Simplicity: Easier to manage than multiple microservices, keeping infrastructural complexity and operational costs low.
- Easy to migrate modular monoliths can be separated without much effort. This benefit can be attractive to startups, who seek simplicity and future improvements.

Challenges:

- Limited Technological Diversity: Modular monoliths restrict the use of diverse technologies and languages compared to microservices due to executing code within a single runtime.
- Scalability Constraints: Scaling a modular monolith can face limitations as the entire application needs to be scaled together, potentially leading to inefficient resource utilization.

V. MODULAR MONOLITH INTERNALS

It is simple to imagine how modules are defined in the project. Each module has its own project layers such as domain, application, infrastructure and so on. But what happens when communication between modules is required. There are several approaches, such as sending data through a service bus or declaring module interfaces.

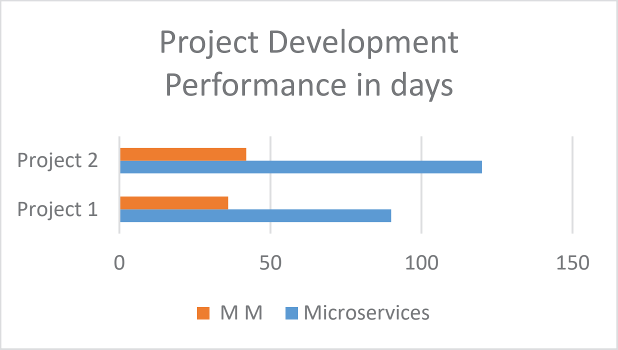
According to my experience, using a service bus gives us a clear border between modules and transferring modular monolith into microservices requires less effort. But it mostly misses the main goal, which is to build a project simply. Service bus is an additional infrastructure unit, and it increases project complexity.

Using interface is easy and performant because direct function calls are always faster than network calls. Unfortunately, standard interface implementation still pairs modules tightly. But what about using the rpc approach? especially grpc which is well supported by Google. RPCs always look like a normal function, even thou they make network calls. What if we use RPC-like structure to call directly referenced modules. When using this approach developer is step ahead and if in the future the project is scaled, fake RPC functions will be replaced by real RPC calls.

It is fact, that transferring from fake RPCs to real RPCs is not straightforward, but this process is easy. According to my bank experience building projects for experimental products using modular monolith and RPC approach gives better development performance, because dev team members do not need to worry about sending messages, network failures and so on. The development time was mostly reduced about 30%-40%, which is pretty much attractive for businesses people and product managers.

VI. SPEED DIFFERENCE BETWEEN MONOLITH AND MODULAR MONOLITH WHILE SEPARATING INTO MICROSERVICES

Transferring into microservices became a smooth process. The transferring time was also reduced by 50%-70% and not many bugs were developed during this period. Modules are easily decomposed as a bunch of units. Units became docker containers and were deployed in the cloud.



Picture 1

The following chart displays the development performance of two real financial projects. According to the results, using modular monolith significally reduces the development time compared to microservices. This big difference is caused by avoiding additional infrastructural components.

VII. SUMMARISE

As reality has shown us, using the most advanced architectural approaches for newly started projects is not good decision. Choosing easy structures is the best way to build new digital products. But developers must be wise and plan the project like this so that in the future it will be easily maintainable scalable and reliable. Modular monoliths are one of the best options to achieve this goal.

REFERENCES

1. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. First edition. Martin Kleppmann, 2017
2. System Design Interview – An insider's guide volume 1 – An insider's guide. Alex Xu, 2020
3. System Design Interview – An insider's guide volume 2 – An insider's guide. Alex Xu, Sahn Lam 2022
4. Domain-Driven Design: Tackling Complexity in the Heart of Software 1st Edition. Eric Evans 2003
5. Building Microservices 1st Edition – Sam Newman 2015

Indirect Error Difference Approximation for Selecting the Best Anytime Algorithm for Tasks with Unrecognizeable Quality

Konstantin Bulatov
IIAP NAS RA, Yerevan, Armenia
e-mail: hpbuko@gmail.com

https://doi.org/10.62343/csit.2024.7

***Abstract* — In this paper a preliminary study is described of the method of selecting the best anytime algorithm out of two. Given two algorithms which solve the same problem generally improve their results over time, but for which the current quality of the result is now known, the method tries to estimate the difference between their error levels. The method performs its estimation indirectly – first, it constructs a method to approximate the absolute value of the difference between distances from two potential answers to an unknown correct answer, then it uses this approximation to estimate the error value of an anytime algorithm down to an additive constant, and then it uses the latter to compare two algorithms which solve the same problem but observes different intermediate results. The method is tested for the problem of selecting the best strategy for combining per-frame recognition results in the task of text recognition in video stream.**

***Keywords* — Anytime algorithms, combination problems, text recognition, video stream, sequential decision making.**

I. INTRODUCTION

Anytime algorithms are algorithms which could be interrupted at any time and which yield better results between observations [1]. They serve an important role when building intelligent systems, since they allow to use the tradeoff between solution quality and spending a resource (such as computation time) to maximize overall utility of the system.

One of the desired properties of anytime algorithms is *recognizeable quality* [2] – the requiement that the system can observe the current value of the quality of the result. The knowledge of the quality of the current result, or at least how this quality changed between observations, allows to use dynamic programming to solve the problem of opti-mal stopping, or build reward functions for learning optimal behaviour when constructing higher-level decision making schemes. However, there exist a wide class of problems where the quality of the current result is not directly observable, but where the process could be considered as an anytime algorithm with mean results improving over time. The examples of such problems are text recognition in a video stream [3] or sequential tomographic reconstruction with increasing number of available projections, also called "monitored reconstruction" [4]. For such cases the "correct" solution is not known during operation (as in text recognition in video stream), or, strictly speaking, not known at all (in tomographic reconstruction). However, depending on how the problem is stated, there could still exist methods which could allow meaningful decision making relying on the general structure of the problem. For example, in the mentioned problems, if the quality of the result is determined as a distance (calculated as a value of a metric which satisfies triangle inequality) to an "ideal" result, then, for example, an optimal stopping method could be formulated, which only analyses the distances between observed values, and assumes that the algorithm is generally well-behaved (generally improves over time and yields diminishing returns over time).

In this paper we consider the problem of dynamic selection of the best anytime algorithm, given two algorithms which solve the same problem. We construct a method which would not rely on recognizeable quality, and try to approximate the potential difference between the errors of the two observed algorithms.

II. FRAMEWORK

Let Θ respresent the set of all possible solutions for a task with a given function $d : \Theta^2 \to \mathbb{R}$, which have the property $\forall a,\, b \in \Theta : a = b \Rightarrow d(a,\, b) = 0$. Let $\theta \in \Theta$ denote an ideal solution of the task. Let us define the quality of the result $r \in \Theta$ as a value inversely related to the value of the function $d(r,\, \theta)$ (i.e. the lower the value of $d(r,\, \theta)$ the "closer" the result r is to an ideal result θ). The value $d(r,\, \theta)$ will be referred to as an "error" of the result r.

Let us consider a family of anytime algorithms which imply the sequence of process stages with numbers $n = 1,\, 2,\, \ldots$. Given a certain fixed but unknown θ we observe a sequence of states $S_1,\, S_2,\, \ldots,\, S_n,\, \ldots$ where $\forall i : S_i = S_i(\theta) \in \mathbb{S}$. Anytime algorithm R on each stage n takes as an input a state S_n, and on each stage n the algorithm could be stopped with a result $R_n(S_1,\, \ldots,\, S_n) \in \Theta$. The quality of the result R_n, i.e. $d(R_n(S_1,\, \ldots,\, S_n),\, \theta)$ is unknown during the process, however we will assume that the algorithm R solves a problem of minimizing the total error on some set of input data $D \subset \mathbb{S}^n \times \Theta$:

$$\sum_{(S_1,\ldots,S_n,\theta)\in D} d(R_n(S_1,\ldots S_n),\theta) \to \min_R . \qquad (1)$$

Let us now consider two separate anytime algorithms R and Q, both solving the same problem (1). Let us assume that both algorithms during their operation receive the same sequence of input states $S_1, S_2, \dots, S_n$, however their results R_n and Q_n on n-stage could differ. We can state a problem of selecting a result $M_n \in \{R_n, Q_n\}$, $M_n = M_n(R_1, \dots, R_n, Q_1, \dots, Q_n)$ such as to minimize the total error.

Optimal selection of M_n consists of selecting a result with a minimal error, i.e.:

$$M_n = \begin{cases} R_n, & d(R_n, \theta) < d(Q_n, \theta), \\ Q_n, & d(R_n, \theta) \geq d(Q_n, \theta), \end{cases} \tag{2}$$

however in order to solve this problem one needs to estimate the error difference $d(R_n, \theta) - d(Q_n, \theta)$.

The general framework of the approach proposed in this paper consists of three approximations:

1) The approximation of the absolute value of the error difference $|d(a, \theta) - d(b, \theta)|$ for a pair of results $a, b \in \Theta$;
2) Approximation of $d(R_i, \theta)$ except for a additive constant, for a given sequence of anytime algorithm results $R_1, R_2, \dots, R_n$;
3) Approximation of the difference between the additive constants for two sequences of anytime algorithm results, resulting in a hypothesis of selecting the best result out of two.

A. Approximation of the absolute value of the error difference

For two elements $a, b \in \Theta$ instead of trying to estimate $d(a, \theta) - d(b, \theta)$ we will try to approximate the absolute value of this difference $|d(a, \theta) - d(b, \theta)|$. Let us consider a non-negative function $g : \Theta^2 \to \mathbb{R}$ complying to the property $\forall a, b \in \Theta : a = b \Rightarrow g(a, b) = 0$. Let us assume that the value of $g(a, b)$ approximates the absolute value of the difference between error levels of a and b:

$$g(a, b) \approx |d(a, \theta) - d(b, \theta)|. \tag{3}$$

As an example of such approximation, for a certain metric function d we could construct the function $g(a, b)$ as a function proportional to $d(a, b)$ with minimizing a squared approximation error for a given training dataset $D_g \subset \Theta^3$:

$$\begin{aligned} & g(a, b) = \alpha \cdot d(a, b) : \\ & \sum_{(a,b,\theta) \in D_g} \left(|d(a, \theta) - d(b, \theta)| - \alpha \cdot d(a, b) \right)^2 \to \min_{\alpha \in [0,1]} . \end{aligned} \tag{4}$$

B. Approximation of error except for a constant

Given a sequence of anytime algorithm results $R_1, R_2, \dots, R_n$ for a problem with an ideal solution θ let us construct an approximation of the error $d(R_i, \theta)$ down to an additive constant, i.e. construct $r_1, r_2, \dots, r_n$ such that:

$$\forall i \in \{1, \dots, n\} : \quad \begin{aligned} & d(R_i, \theta) \approx r_i + \gamma, \\ & \text{with } r_n = 0. \end{aligned} \tag{5}$$

Given an approximation of the absolute value of the error differences, let us construct $r_1, \dots, r_n$ by solving the following problem:

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(|r_i - r_j| - g(R_i, R_j) \right)^2 \to \min_{r_1, \dots, r_n, \text{ with } r_n = 0} . \tag{6}$$

The problem (6) is easily solvable if for each pair (r_i, r_j) the sign of the different $r_i - r_j$ is known or could be assumed. In general, when anytime algorithms are considered typically it is assumed that the results of the algorithm always improve over time (or at least, in a weaker definitions, "are well-behaved over time"). Thus, if we assume that the results of the algorithm improve over time, we can assume that $r_i - r_j > 0$ for any $i < j$, and thus easily solve the problem (6).

It is also worth to note that if for the approximation of the absolute value of error different we use a linear function of the distance (4), then with any $\alpha > 0$ the estimations $r_1, \dots, r_n$ will be the same except for a multiplicative constant, and if the estimations will be constructed in form $r_i = \alpha \cdot \hat{r}_i$, then simply the distances $d(a, b)$ can be used instead of $g(a, b)$.

C. Approximation of the error difference

Let us consider two sequences of solutions for a problem with an ideal result θ: the sequence $R_1, \dots, R_n$ of the results of the algorithm R and the sequence $Q_1, \dots, Q_n$ of the results of the algorithm Q. Let us assume that the results of both algorithms improve over time, thus by solving the problem (6) we obtain $r_1, \dots, r_n$ and $q_1, \dots, q_n$ such that:

$$\forall i \in \{1, \dots, n\} : \quad \begin{aligned} & d(R_i, \theta) \approx r_i + \gamma_R, \\ & d(Q_i, \theta) \approx q_i + \gamma_Q. \end{aligned} \tag{7}$$

Let $\gamma = \gamma_R - \gamma_Q$ and let us find γ by solving the following problem:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \left(|r_i - q_j + \gamma| - g(R_i, Q_j) \right)^2 \to \min_{\gamma} . \tag{8}$$

By enumerating all possible combinations of signs under the absolute value bars in (8) we can find all solutions for this problem in polynomial time. Given a found γ we can now make an algorithm selection decision, by selecting the result $M_n \in \{R_n, Q_n\}$ on stage n by supplying the obtained approximation to the original definition (2):

$$M_n = \begin{cases} R_n, & r_n - q_n + \gamma < 0, \\ Q_n, & r_n - q_n + \gamma \geq 0, \end{cases} \tag{9}$$

moreover, if the approximation r_i and q_i was constructed given $r_n = q_n = 0$ then it is enough to simply compare γ with zero.

It is important to note that the task (8) could have many solutions. For example, if $\forall i, j \in \{1, \dots, n\} : R_i = R_j \wedge Q_i = Q_j$ and $g(R_n, Q_n) > 0$ then $\forall i \in \{1, \dots, n\} : r_i = q_i = 0$ and the solution for the problem (8) is $\gamma = \pm g(R_n, Q_n)$, which makes the choice (9) impossible. In order to reduce ambiguity, one can set up a fixed policy to find the maximal value of γ which minimizes (8), thus favouring the algorithm Q in ambigious cases.

As in the previous subsection, it is worth to note that if in both approximations a linear function (4) is used as $g(a, b)$ with the same $\alpha > 0$ then by setting $r_i = \alpha \cdot \hat{r}_1$, $q_i = \alpha \cdot \hat{q}_i$ and $\gamma = \alpha \cdot \hat{\gamma}$ we can find $\hat{\gamma}$ with the described algorithm simply using the distances $d(a, b)$ instead of the function $g(a, b)$.

III. EXPERIMENTAL EVALUATION

For experimental evaluation of the proposed approach let us consider the task of the recognition of text string (text field of a document) in a video stream. Let Θ be a set of all possible text strings, θ – a "correct" text field recognition result. Let the algorithm R denote an algorithm of selecting a single per-frame result with maximal focus estimation score [5], and the algorithm Q represent the modification of the algorithm ROVER [6], [7] to combine per-frame recognition results.

Normalized Levenshtein distance [8] will be used as $d(a, b)$, and we will construct an approximation $g(a, b)$ as a linear function of $d(a, b)$ (4). As was mentioned before, the results of the estimations does not actually depend on α in (4), thus we will simply use the approximation $g(a, b) = d(a, b)$.

As the original dataset the documents from MIDV-500 dataset [9] were used. By recognizing the text fields using the system [10], 691 text field recognition result sequences were obtained. On 491 sequences both the results of R and Q were improving over time on all stages of the process, on 116 – only the results of R, on 72 sequences – only the results of Q, and on the remaining 84 sequences neither the results of R nor Q were strictly improving over time throughout the stages of the process.

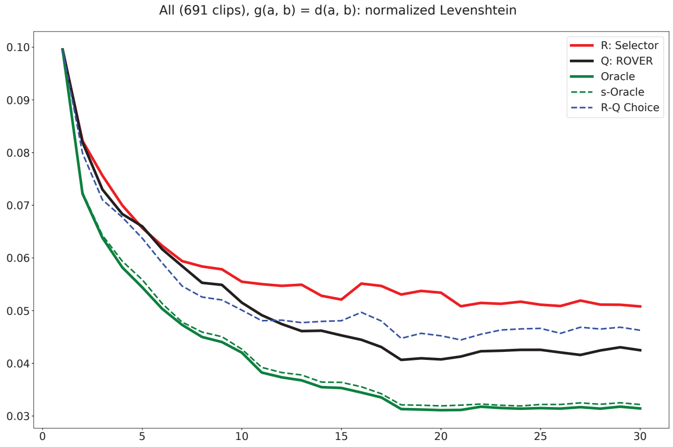


Fig. 1: Error plot for text field recognition in video stream (all sequences). Horizontal axis – frame number, vertical axis – mean error.

Figure 1 shows the dynamics of text field recognition error on all sequences of the dataset. Red and black show the results of the algorithms R and Q respectively, solid green line visualizes an "ideal selection" of the best result (2) (to which a correct result is known). Dashed green line represent the result of the proposed approach with an ideal selection of signs of the differences $r_i - r_j$ in (6). Blue dashed line show the result of the proposed method in a computable scenario (thus, having no access to ideal signs of the differences $r_i - r_j$ and the ideal result), which assumes that R and Q always improve over time.

The selection of the best result using the proposed method allows to reduce mean error (relative to the best of the two algorithms) for sufficiently low amount of frames ($n < 13$). It is worth noting that with ideal selection of signs of the differences $r_i - r_j$ in (6) the selection of the algorithm using the proposed method almost corresponds to an ideal selection.

Figure 2 shows the dynamics of the text field recognition error on the four subsets of the dataset. It can be noted that on the sequences where both R and Q strictly improve over time the proposed method of selecting the best result is almost indistinguishable from ideal choice. The highest error is observed on the subset where only Q (which has the highest overall mean quality) improves over time, as the method R not only does not improve over time on this subset, but also reaches the highest levels of error.

IV. CONCLUSION

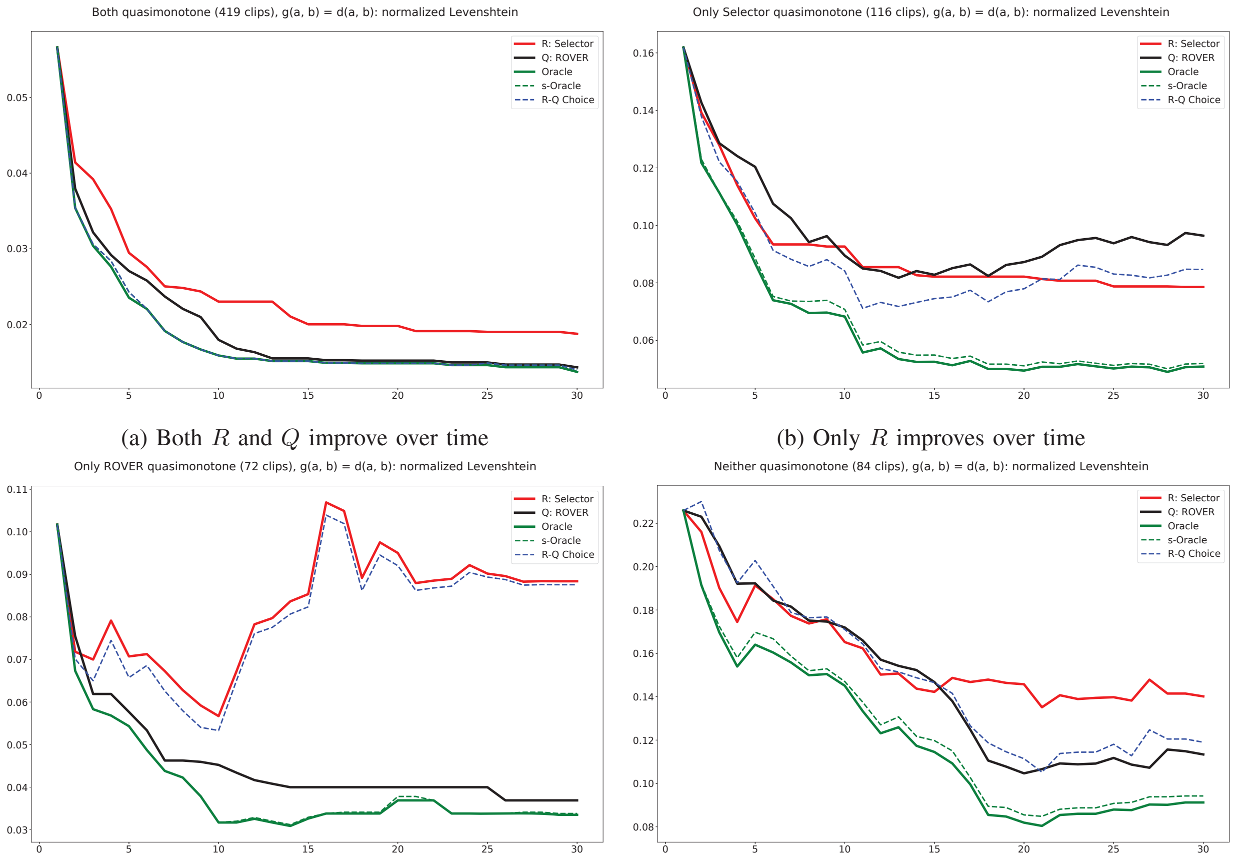
As it can be seen from the experimental evaluation, the proposed method could show promise in some practical applications, however the most crucial problem is deal with the fact that on real data the combination algorithms not always improve over time. If that were always the case, a much simpler method would be to simply select the algorithm which is most different to the known common result (which is the result on the first frame, if the target problem is text field recognition in a video stream).

However, the proposed method achieves almost optimal selection if the sign of the error difference becomes known. Thus, if there exists some other problem-specific method of determining signs of the differences $r_i - r_j$ in (6), then the proposed approach may yield good results.

As a function $g(a, b)$ which approximates the absolute value of the error difference another type of function could be used (or trained) which would take not only the two results R_i, R_j or R_i, Q_j, but all the available information $S_1, \ldots, S_n$.

Using the proposed indirect approach it is difficult to solve the problems (6) and (8) simultaneously, since the absolute value bars in (8) has to be eliminated in a different way for different solutions of (6). That is why in the provided general scheme we rely on an assumption of known signs which allow to eliminate absolute value bars in (6) (e.g. by assuming that the algorithms strictly improve over time).

The solution to the problem (6) can also potentially be used to solve the stopping problem for the anytime algorithm R [3], [4]: the optimal stopping implies the estimation of the expected difference $d(R_n, \theta) - d(R_{n+1}, \theta)$, which can try to predict by estimating all $d(R_i, \theta) - d(R_{i+1}, \theta)$ in the form $r_i - r_{i+1}$.



(a) Both R and Q improve over time

(b) Only R improves over time

(c) Only Q improves over time

(d) Neither R nor Q improve over time

Fig. 2: Error plot for text field recognition in video stream (subsets of sequences). Horizontal axis – frame number, vertical axis – mean error.

REFERENCES

1. T. Dean and M. Boddy, "An analysis of time-dependent planning", Proceedings of the Seventh AAAI National Conference on Artificial Intelligence (AAAI'88), pp. 49–54, 1988.
2. S. Zilbershtein, "Using Anytime Algorithms in Intelligent Systems", AI Magazine, vol. 17, no. 3, pp. 73–83, 1996.
3. K. Bulatov, N. Razumnyi and V.V. Arlazarov, "On optimal stopping strategies for text recognition in a video stream as an application of a monotone sequential decision model", IJDAR, vol. 22, no. 3, pp.303–314, 2019.
4. K. Bulatov, M. Chukalina, A. Buzmakov, D. Nikolaev and V.V. Arlazarov, "Monitored Reconstruction: Computed Tomography as an Anytime Algorithm", IEEE Access, vol. 8, pp. 110759–110774, 2020.
5. I. Tolstov, S. Martynov, V. Farsobina and K. Bulatov, "A modification of a stopping method for text recognition in a video stream with best frame selection", Proceedings of ICMV 2020, pp. 116051M1-116051M9, 2021.
6. J.G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)", Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 347–354, 1997.
7. K. Bulatov, "A Method to Reduce Errors of String Recognition Based on Combination of Several Recognition Results with Per-Character Alternatives", Bulletin of the South Ural State University, Series: Mathematical Modelling, Programming and Computer Software, vol. 12, no. 3, pp. 74–88, 2019.
8. L. Yujian and L. Bo, "A Normalized Levenshtein Distance Metric", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pp. 1091–1095, 2007.
9. V.V. Arlazarov, K. Bulatov, T. Chernov and V.L. Arlazarov, "MIDV-500: A Dataset for Identity Document Analysis and Recognition on Mobile Devices in Video Stream", Computer Optics, vol. 43, no. 5, pp. 818–824, 2019.
10. Y. Chernyshova, A. Sheshkus, V.V. Arlazarov, "Two-step CNN frame-work for text line recognition in camera-captured images", IEEE Access, vol. 8, pp. 32587–32600, 2020.

Artificial Neural Networks that Change their Configuration

Tatiana Kosovskaya
Sankt-Petersburg State University
Sankt-Petersburg, Russia
e-mail: kosovtm@gmail.com

https://doi.org/10.62343/csit.2024.8

***Abstract* — An analogue of a neural network, the number of layers and the number of cells in which may be changed during its retraining is suggested in the paper. The main instrument for constructing such a network is extraction of maximal common properties of pairs of objects in the training set and of that ones used for retraining. The degree of coincidence of a recognized object with the one presented in the training set may be calculated using their maximal common properties. Computational complexities of such a network construction, recognition process and the network retraining are proved. A brief description of a similar network proposed by the author earlier for complex structured objects described using predicate calculus is presented. The analysis of comparison of computational complexity of a complex structured object recognition with various methods of their description is given.**

***Keywords* — neural network; maximal common property of objects; degree of coincidence; computational complexity.**

I. INTRODUCTION

The origins of the appearance of artificial neural networks lie in the works of F. Rosenblatt, dedicated to the creation of a perceptron, a machine for solving pattern recognition problems [1]. However, the significant disadvantages of the perceptron led to the creation of other models for solving such problems. One of the most common models currently is a neural network.

An artificial neuron is only a model of a neuron in a living organism. Therefore, it will be called a cell below. At the same time, the contents of such a cell may vary depending on the way the problem is formalized.

When creating a modern artificial neural network, the researcher sets its configuration in advance: the number of network layers and the number of cells in each layer. This does not correspond to how a neural network is built in the brain of a living organism, in which new neurons are added during the learning process, some connections are broken and new ones are formed. Besides, classical artificial neural networks have the disadvantage that unpredictable ”outliers” sometimes occur while recognition new objects. Results that are poorly explained from a theoretical point of view.

In this regard, various modifications of artificial neural networks began to be developed.

Overfitting is a serious problem in such networks. For example, a dropout network was suggested in [2] to overcome such a problem. To detect regularities in datasets, polynomial neural networks were developed [3], [4]. Due to the huge volumes of processed data, fuzzy neural networks have found their application.

In this paper, it is proposed to construct an analogue of a neural network, the number of layers and the number of cells in which may be changed during its retraining.

The basis for the construction of the proposed networks is the concept of the maximum common property (MCP) of objects. This concept was introduced earlier by the author for complex structured objects described using predicate formulas [5]. The problem of finding MCP of such objects has expo-nential complexity [6].

For objects described in the terms of binary or finite-valued features, the problem of finding MCP is a polynomial one. Such a difference in computational complexity depending on the chosen description language is discussed in the Discussion section of the paper.

II. PRE-TRAINING THE NETWORK FOR OBJECTS DESCRIBED BY FINITE VALUED FEATURES

A. Setting of Problem

Let Ω be a set (potentially infinite) of objects. A set of features $p_1, \dots, p_n$ that define the description of an object ω from Ω in the form of a string of values of these features $\alpha = (\alpha_1, \dots, \alpha_n)$ is defined for objects under study. A training set (TS) $\{\omega^1, \dots, \omega^K\}$ of objects from Ω with descriptions $\alpha^k = (\alpha_1^k, \dots, \alpha_n^k)$, $k = 1, \dots, K$ is given.

It is required to construct a network that gives the answer “YES” for objects from TS. If an object is absent in TS we can not guarantee its correct recognition. Fuzzy recognition of such an object consists in calculation of maximal degree of coincidence with one of the objects from TS.

B. Important Definitions

Definition 1. *The maximal common property (MCP) of objects with descriptions* $\alpha^k = (\alpha_1^k, \dots, \alpha_n^k)$ *and* $\alpha^m = (\alpha_1^m, \dots, \alpha_n^m)$ *is a string of the form* $\alpha^{km} = (\alpha_1^{km}, \dots, \alpha_n^{km})$, *where*
if $\alpha_i^k \neq \alpha_i^m$ *then* $\alpha_i^{km} = *$,
if $\alpha_i^k = \alpha_i^m$ *then* $\alpha_i^{km} = \alpha_i^k$.

C. Initial Network Training

Network cells with zero out-degree contain descriptions of objects from TS.

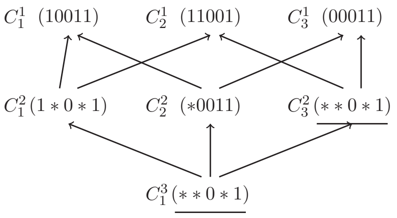


Fig. 1. The result of pairwise extractions of MCPs.

For each pair of objects from the TS $\alpha^k = (\alpha_1^k, \dots, \alpha_n^k)$ and $\alpha^m = (\alpha_1^m, \dots, \alpha_n^m)$ we find their MCP α^{km}. Cells with α^{km} and α^k and α^m are connected by oriented edges. If some pairs of objects have the same MCPs, then their corresponding cells are identified.

Repeat the process with the already extracted MCPs. At the same time, if on the lth repetition we received the same MCP that was received earlier, then the corresponding cells of the network are identified.

The process will stop as the MCP lengths decrease.

D. Computational complexity of initial network training

Let K be the number of objects in the TS.

When the MCPs for pairs of descriptions are found for the first time, $K_1 = \frac{K(K-1)}{2}$ extractions of MCP are produced.

When the MCP for pairs of descriptions obtained on the $l-1$-th layer is found for the l-th time, no more than $\frac{K_l(K_l-1)}{2}$ MCP extractions, each of which requires linear of less (at least by l) than n steps. That is, the $l-1$-th extraction requires $O(K^{2^l}n)$ steps.

At the same time, l does not exceed n. Thus, the total number of steps of initial network training will be $O(\sum_{l=1}^{n-1} K^{2^l}n) = O(K^{2^n})$.

The resulting estimate is very large and achievable.

E. Example of initial network construction

Let the objects be described by five binary features and a TS of three elements with descriptions (10011), (11001) and (00011) is given.

According to the described above algorithm the obtained network has the form presented in Figure 1.

The cells C_3^2 and C_1^3 must be glued because they contain the same MCP. The result of gluing the cells C_3^2 and C_1^3 is presented in Figure 2.

It must be noted that now we can not say what is the number of level to which the cell C_1^3 belongs.

III. RECOGNITION PROCESS

Obviously, such a network can precisely recognize only such an object whose description coincides with the descrip-tion of some object from the TS. The process of such a recognition for an object with the description (11011) is presented in Figure 3.

Let an object (which was previously absent in the TS) with the description $\alpha = (\alpha_1, \dots, \alpha_n)$ be presented for recognition.

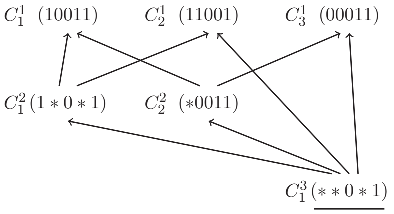


Fig. 2. The result of gluing the cells C_3^2 and C_1^3.

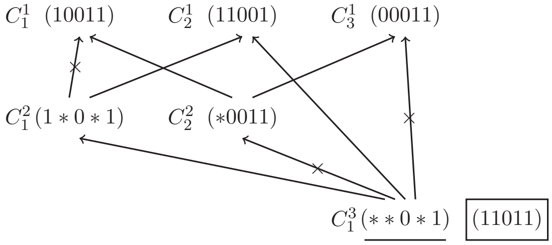


Fig. 3. Recognition of an object with description presented in TS.

At each check of whether the cell contents match its description, in case of a negative result, the degree of their coincidence deg_C is calculated in the cell C. The degree of an object description coincidence with the contents of a cell is the ratio of the number of matched values to the length of the contents. Note that the length and match of the contents is calculated without taking into account the number of occurrences of the $*$ symbol. For example, if the description of a recognized object is (10001) and the contents of a cell C is $(*0011)$ then $deg_C = \frac{3}{4}$. For a cell C' with the contents $(1*0*1)$ $deg_{C'} = 1$.

Why do we calculate the degree of coincidence in the intermediate cells? It is not necessary. But it can allow us to make a traversal of the network beginning with the cells with the largest degree of coincidence.

Let an object with the description (10001) be presented for recognition. This object was absent in the TS.

The process of recognizing an object with a description (10001) that was absent in TS is shown in Figure 4.

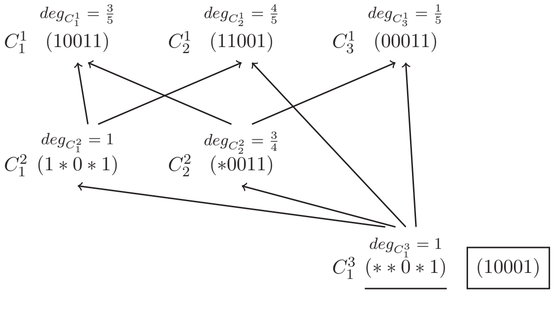


Fig. 4. Recognition of an object with description absent in TS.

A. Computational complexity of recognition process

Since the process of recognition of an object is reduced to the graph traversal from the root cells to the leafs comparing at each cells of strings with the length no more than the number of features, the estimate (very rough) for the number of steps of the recognition process will be

$$O((N+M)n),$$

where N is the number of cells in the network, M is the number of edges in the network ($M < N^2$), n is the number of features, in terms of which the description of the object is obtained.

But if the recognized object was presented in TS then its recognition will be done in $O(h \cdot n)$ number of steps, where h is the "height" of the network. Since $h < n$, this bound is $O(n^2)$.

IV. NETWORK RETRAINING

An already constructed network can be retrained on a new object with the description $\alpha = (\alpha_1, \dots, \alpha_n)$, about which it is known that the network should give an answer "YES". To do this, it is enough, firstly, to find the MCP of the presented object with the contents of the leafs. Secondly, to find the MCP of the contents of newly received cells with the contents of the old ones. And finally, to glue cells with the same contents.

As a result of this process, new cells may appear in the network, and both the number of layers and the number of cells in the network may be changed.

A. Computational complexity of network retraining

Let N be the number of cells in the network before retraining, N_0 be the number of leafs, N_1 be the number of cells having oriented edge ending in leafs, ..., N_h be the number of roots. Note, that $h < n$ and $N_0 + N_1 + \dots + N_h = N$.

The first MCP extractions from the description of a new object and the contents of leafs requires $N_0 n$ steps. These extractions can give no more than N_0 new cells.

Next extractions can require $N_1 \cdot N_0 \cdot n$, $N_2 \cdot N_1 \cdot N_0 \cdot n$, ..., $N_h \dots N_2 \cdot N_1 \cdot N_0 \cdot n$ steps. The total number of steps is $O(N_h \dots N_2 \cdot N_1 \cdot N_0 \cdot n)$.

So, we have an exponential upper bound for network retraining number of steps. For example, if $N_1 = \dots = N_h = \frac{N}{n}$ then computational complexity is $O((\frac{N}{n})^n)$.

V. NETWORK FOR COMPLEX STRUCTURED OBJECTS

The idea of fuzzy neural network that changes its configuration came to the author while investigating complex structured object recognition. It is convenient to describe such objects in the terms of predicate formulas [7], [8]. In such a case, recognition problems turn out to be NP-complete or NP-hard [6]. To decrease computational complexity of algorithms solving such problems, the author earlier has introduced notions of level description of objects, networks changing their configuration [9] and fuzzy logic-predicate networks [5].

This section will give a brief description of this approach. Then the difference in computational complexity of the two approaches will be discussed in the Discussion section.

A. Problem setting

Let an investigated object ω be a complex structured object, which is presented as a set of its elements $\omega = \{\omega_1, \dots, \omega_t\}$ and is characterized by predicates $p_1, \dots, p_n$. These predicates define some properties of its elements or relations between them. The description $S(\omega_1, \dots, \omega_t)$ of the object ω is a set of all constant literals (atomic formulas or their negations) with predicates $p_1, \dots, p_n$ which are true for ω.

The set of all objects Ω is divided into K classes $\Omega = \Omega_1 \cup \dots \cup \Omega_K$. Formula $A_k(\overline{x}_k)$ have the form of disjunction of elementary conjunctions and is true if $\omega \in \Omega_k$.

The recognition of ω consists in checking

$$S(\omega) \Rightarrow \exists \overline{x}_{k\neq} A_k(\overline{x}_k). \tag{1}$$

To denote that there exist distinct values for variables from the list of variables $\overline{x}$ the notation $\exists \overline{x}_{\neq} A_k(\overline{x})$ is used.

This is an NP-complete problem. Depending on an algorithm searching for the values of $\overline{x}_k$ in (1), computational complexity is exponential on the number of literals in $A_k(\overline{x}_k)$ or on the number of arguments in it [6].

If we need not only to check the formula (1), but to find values for $\overline{x}_k$ then it becomes NP-hard. Nevertheless, both an exhaustive search algorithm and an algorithm based on derivation in predicate calculus not only answer the question "if their exists?" but find values for $\overline{x}_k$.

B. Level description of classes

Definition. *Elementary conjunction $R(\overline{z})$ is called an MCP of two elementary conjunctions $A(\overline{x})$ and $B(\overline{y})$ if it is their maximal common up to the names of arguments sub-formula.*

Instead of the term "common up to the names of arguments sub-formula" the term "isomorphic" is used in [5].

While MCP extraction for objects described by a binary (or multi-valued) string is a polynomial-in-time procedure, MCP extraction for complex structured objects represented as a set of its elements and described by a predicate formula is NP-hard problem [6].

To decrease computational complexity of checking (1), it was suggested in [5] to extract pairwise MCPs $P_1^L(\overline{y}_1^L), \ \dots, \ P_{n_L}^L(\overline{y}_{n_L}^L)$ for disjunctive terms of $A_k(\overline{x}_k) = A_{k,1}(\overline{x}_{k,1}) \vee \dots \vee A_{k,m_k}(\overline{x}_{k,m_k})$.

Simultaneously new unary predicates $p_i^L(y_i^L)$ and new variables y_i^L for lists of initial variables $\overline{y}_i^L$ defined by equalities $p_i^L(y_i^L) \Leftrightarrow P_i^L(\overline{y}_i^L)$ are introduced.

Repeat this procedure with formulas $P_1^l(\overline{y}_1^l), \ \dots, \ P_{n_l}^l(\overline{y}_{n_l}^l)$ for $l = L, \dots, 2$ and receive $P_1^{l-1}(\overline{y}_1^{l-1}), \ \dots, \ P_{n_{l-1}}^{l-1}(\overline{y}_{n_l}^{l-1})$.

Every occurrence of $P_i^l(\overline{y}_i^l)$ in $P_{j'}^{l'}(\overline{y}_{j'}^{l'})$ ($l' > l$) and in $A_{k,j}(\overline{x}_{k,j})$ replace by $p_i^l(y_i^l)$.

Let $A_{k,j}^L(\overline{x}_{k,j}^L)$ be the results of substitutions of $p_i^l(y_i^l)$ instead of $P_i^l(\overline{y}_i^l)$ into $A_{k,j}(\overline{x}_{k,j})$. Level description of formulas $A_{k,1}(\overline{x}_{k,1}), \dots, A_{k,m_k}(\overline{x}_{k,m_k})$ has the form (2) [5].

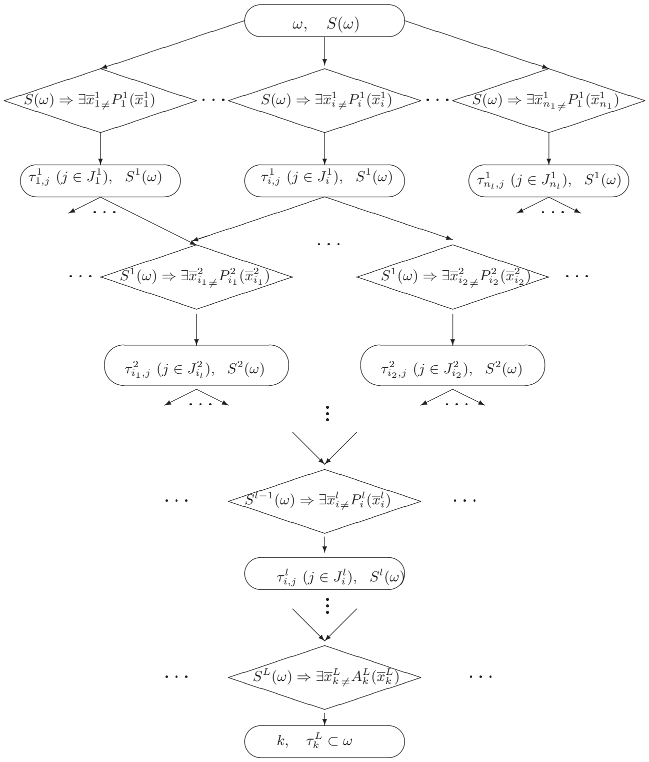

$\omega, \quad S(\omega)$
$S(\omega) \Rightarrow \exists \overline{x}^1_{1\neq} P^1_1(\overline{x}^1_1)$
$S(\omega) \Rightarrow \exists \overline{x}^1_{i\neq} P^1_i(\overline{x}^1_i)$
$S(\omega) \Rightarrow \exists \overline{x}^1_{n_1\neq} P^1_1(\overline{x}^1_{n_1})$
$\tau^1_{1,j}\ (j \in J^1_1), \quad S^1(\omega)$
$\tau^1_{i,j}\ (j \in J^1_i), \quad S^1(\omega)$
$\tau^1_{n_l,j}\ (j \in J^1_{n_l}), \quad S^1(\omega)$
$S^1(\omega) \Rightarrow \exists \overline{x}^2_{i_1\neq} P^2_{i_1}(\overline{x}^2_{i_1})$
$S^1(\omega) \Rightarrow \exists \overline{x}^2_{i_2\neq} P^2_{i_2}(\overline{x}^2_{i_2})$
$\tau^2_{i_1,j}\ (j \in J^2_{i_l}), \quad S^2(\omega)$
$\tau^2_{i_2,j}\ (j \in J^2_{i_2}), \quad S^2(\omega)$
$S^{l-1}(\omega) \Rightarrow \exists \overline{x}^l_{i\neq} P^l_i(\overline{x}^l_i)$
$\tau^l_{i,j}\ (j \in J^l_i), \quad S^l(\omega)$
$S^L(\omega) \Rightarrow \exists \overline{x}^L_{k\neq} A^L_k(\overline{x}^L_k)$
$k, \quad \tau^L_k \subset \omega$

Fig. 5. Process of level recognition.

$$\left\{\begin{array}{ccc} & A^L_{k,j}(\overline{x}^L_{k,j}) & (j = 1, \ldots, m_k) \\ p^1_1(y^1_1) & \Leftrightarrow & P^1_1(\overline{y}^1_1) \\ & \vdots & \\ p^1_{n_1}(y^1_{n_1}) & \Leftrightarrow & P^1_{n_1}(\overline{y}^1_{n_1}) \\ & \vdots & \\ p^l_i(y^l_i) & \Leftrightarrow & P^l_i(\overline{y}^l_i) \\ & \vdots & \\ p^L_{n_L}(y^L_{n_L}) & \Leftrightarrow & P^L_{n_L}(\overline{y}^L_{n_L}) \end{array}\right. . \qquad (2)$$

C. Level recognition

The process of level recognition is presented in Figure 5. (This figure was published in the author's paper [5].)

Here we can see why the author uses the term "cell" instead of the term "neuron". The contents of cells in Figure 5 are elementary conjunctions of predicate formulas. Every condition which is checked in a rhomb of this scheme has the same form as the formula (1). But its right-hand part is essentially shorter than that in (1). That is why computational complexity essentially decreases.

But, nevertheless, the problem remains NP-complete (NP-hard if we need to find values of arguments in (1)).

VI. DISCUSSION

It is well known that every information may be represented in the form of a binary string.

The main part of this paper is devoted to the recognition problem of objects described by binary strings. It was shown that computational complexity of recognition of an object with description presented in TS is $O(n^2)$, where n is the number of features (i.e., the length of description).

In section 6, it was mentioned (with reference to [5]) that computational complexity of recognizing a complex structured object is an NP-complete problem.

Is it a fraud? Or it was proved that **P**=**NP**?

Computational complexity is a function of the input data length.

Let $\omega = \{\omega_1, \ldots, \omega_t\}$ and be characterized by predicates $p_1, \ldots, p_n$. The length of a binary string which simulates an elementary conjunction of predicate formulas with t variables and containing an m-ary predicate is $O(t^m$

It corresponds to the the estimate $O(t^m \cdot |A| \cdot |S|)$ given in [7] for the number of propositional variables modelling predicate calculus formulas in a finite domain.

As polynomial of the exponent gives the exponent, we have not prove that **P**=**NP**. And it was not a fraud that the same problem with different representations of input data belongs to essentially different complexity classes.

ACKNOWLEDGMENT

The author acknowledges Saint-Petersburg State University for a research project 95438429

REFERENCES

1. F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain." Psychological Review, vol. 65, no. 6, pp. 386–408, 1958.
2. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting." J. of Machine Learning Research, no. 15, pp. 1929 1958, 2014.
3. G.G. Chrysos, S. Moschoglou, G. Bouritsas, Y. Panagakis, J. Deng and S. Zafeiriou, "P-nets: Deep Polynomial Neural Networks." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), China, pp. 7325–7335, 14 06 2020.
4. A. Zou, R. Deng, Q. Mei, et al, "Fault diagnosis of a transformer based on polynomial neural networks." Cluster Comput, vol. 22, pp. 9941–9949, 2019.
5. T. Kosovskaya, "Fuzzy logic-predicate network." In Proceedings of the 11th Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (EUSFLAT 2019), Prague, Czech, 9-13 09 2019, pp. 9 – 13.
6. T. Kosovskaya, Predicate Calculus as a Tool for AI Problems Solution: Algorithms and Their Complexity. Chapter 3 in Intelligent System. Open access peer-reviewed Edited volume. Edited by Chatchawal Wongchoosuk Kasetsart University, pp. 1 – 20, 2018. [Online]. Available: https://www.intechopen.com/chapters/58698
7. S. Russell and P. Norvig, Artificial Intelligence A Modern Approach, 4th ed.; Pearson Series in Artificial Intelligence, 2021
8. N.J. Nilsson, Problem-Solving Methods in Artificial Intelligence; McGrow-Hill Book Company, New York, 1971.
9. 9 T. Kosovskaya, "Self-modificated predicate networks." International Journal on Information Theory and Applications, vol. 22, no 3, pp. 245–257, 2015

Algorithm for Extraction Common Properties of Objects Described in the Predicate Calculus Language with Several Predicate Symbols

Kosovskaya Tatiana
St Petersburg University
St. Petersburg, Russia
e-mail: kosovtm@gmail.com

Zhou Juan
St Petersburg University
St. Petersburg, Russia
e-mail: st103098@student.spbu.ru

https://doi.org/10.62343/csit.2024.9

***Abstract* — When solving artificial intelligence problems connected with the study of complex structured objects, a convenient tool for describing such objects is the language of predicate calculus. The paper presents two algorithms for the extraction of common properties of objects described in the predicate calculus language with predicate symbols. The first of the algorithms extracts maximal common subformulas for elementary conjunctions containing 2 predicate symbols. The second algorithm extracts maximal common subformulas for elementary conjunctions with several predicate symbols. Estimates of their time complexity are given for both algorithms. Both algorithms are implemented in Python.**

***Keywords* — Predicate formulas, isomorphism of predicate formulas, complex structured object, maximal common subformula.**

I. INTRODUCTION

In artificial intelligence problems, connected with the study of complex structured objects, described by the properties of their elements and the relationships between these elements, it is convenient to use predicate calculus formulas. To describe the classes of such objects, it is necessary to highlight their common properties.

An algorithm for extraction maximal common property of two objects described by means of a single predicate is proposed in [1]. An estimate of the computational complexity is proved for this algorithm.

Presented here two algorithms aim to extract maximal common subformulas for elementary conjunctions, which contain two and several different predicate symbols, respectively.

II. WHY NECESSARY DEFINITIONS

Definition 1 [2]. *Two elementary conjunctions of atomic formulas of predicate calculus $F1(a_1,\dots,a_n)$ and $F2(b_1,\dots,b_n)$ are called isomorphic*

$$F1(a_1,\dots,a_n) \sim F2(b_1,\dots,b_n),$$

if there is such an elementary conjunction $R(x_1,\dots,x_n)$ and substitutions of arguments $a_{i_1},\dots,a_{i_n}$ and $b_{j_1},\dots,b_{j_n}$ of formulas $F1(a_1,\dots,a_n)$ and $F2(b_1,\dots,b_n)$ accordingly, instead of all occurrences of variables $x_1,\dots,x_n$ of the formula $R(x_1,\dots,x_n)$, that the results of these substitutions $R(a_{i_1},\dots,a_{i_n})$ and $R(b_{j_1},\dots,b_{j_n})$ coincide up to the order of literals with the formulas $F1(a_1,\dots,a_n)$ and $F2(b_1,\dots,b_n)$, respectively.

The resulting substitutions $\lambda 1 = \{x_1{:}\,a_{i_1},\dots,\ x_n{:}\,a_{i_n}\}$ and $\lambda 2 = \{x_1{:}\,b_{i_1},\dots,\ x_n{:}\,b_{i_n}\}$ are called unifiers of formulas $F1(a_1,\dots,a_n)$ and $F2(b_1,\dots,b_n)$ with the formula $R(x_1,\dots,x_n)$ respectively.

The formula $R(x_1,\dots,x_n)$ is below referred to as the MCF (Maximum Common sub-Formula).

Definition 2 [1]. *Two substitutions are called contradictory if two different constants a_1 and a_2 are found for the same variable x, i.e., $\{x{:}\,a_1, x{:}\,a_2\}$, or for different variables x_1 and x_2, the same constant a is found, i.e., $\{x_1{:}\,a, x_2{:}\,a\}$.*

Definition 3 [1]. *Let $R(x_1,\dots,x_n)$ and $F(b_1,\dots,b_n)$ be two elementary conjunctions of predicate formulas, with $R(x_1,\dots,x_n)$ containing only variables as arguments.*

Substitution $\{\overline{x}{:}\,\overline{b}\}$, where $\overline{x}$ is a list of some variables from $R(x_1,\dots,x_n)$, $\overline{b}$ is a list of some different constants from $F(b_1,\dots,b_n)$, is called a partial unifier of the formulas $R(x_1,\dots,x_n)$ and $F(b_1,\dots,b_n)$ if the result of applying this substitution to the formula $R(x_1,\dots,x_n)$ contains a subformula that coincides up to the order of literals with some subformula $F(b_1,\dots,b_n)$. Below all unifiers will be partial.

Definition 4 [3]. *An elementary conjunction that does not contain constants is called a common property of two objects if it is isomorphic to some subformulas of each of the descriptions of these objects.*

Definition 5 [3]. *An elementary conjunction that does not contain constants is called a maximum common property (MCP) of two objects if it is their common property with the largest number of literals.*

For further description of the algorithms some notations will be required.

Notation 1. *A number of substitutions in the unifier λ is called an unifier length and is denoted by $||\lambda||$.*

Notation 2. R_i^* is a list in ascending order of unifier lengths, containing pairs $(R_i(\overline{x}_i), \lambda 2_i)$ for all MCF $R_i(\overline{x}_i)$ of subformulas $\left(F1_i(\overline{a}_i), F2_i(\overline{b}_i)\right)$ and their unifier $\lambda 2_i$ with the corresponding subformulas of formula $F2_i(\overline{b}_i)$.

Note that when defining the formula $R_i(\overline{x}_i)$, it is always possible to organize the numbering of these variables in the list of variables $\overline{x}_i$ so that all substitutions in the unifier $\lambda 1_i$ have the form $\{x_\alpha{:}\,a_\alpha\}$ for all variables x_α included in the MCF $R_i(\overline{x}_i)$. Therefore, the unifier MCF $R_i(\overline{x}_i)$ with $F1_i(\overline{a}_i)$ will not be written out below.

Notation 3. $R_{i,j}^*$ – is a list in ascending order of unifier lengths, containing pairs $\left(R_{i,j}(\overline{x}_{i,j}), \lambda 2_{i,j}\right)$ for all MCF

$R_{i,j}(\overline{x}_{i,j})$ of subformulas $\left(F1_{i,j}(\overline{a}_{i,j}), F2_{i,j}(\overline{b}_{i,j})\right)$ and their unifier $\lambda 2_{i,j}$ with the corresponding subformulas of formula $F2_{i,j}(\overline{b}_{i,j})$.

Notation 4. R^* – is a resulting list in ascending order of unifier lengths, containing pairs $(R(\overline{x}), \lambda 2)$ for all MCF $R(\overline{x})$ of subformulas $\left(F1(\overline{a}), F2(\overline{b})\right)$ and their unifier $\lambda 2$ with the corresponding subformulas of formula $F2(\overline{b})$.

III. ALGORITHM MCF2 FOR EXTRACTING COMMON PROPERTIES OF OBJECTS DESCRIBED IN THE PREDICATE CALCULUS LANGUAGE WITH TWO PREDICATE SYMBOLS

Let a pair of elementary conjunctions of atomic predicate formulas $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$ with predicate symbols P_i, P_j and constants $a_1, \dots, a_m$ and $b_1, \dots, b_n$ as arguments be given[1]. The names of all arguments in each literal are different.

The following algorithm **MCF2** is proposed to extract the maximal elementary conjunction $R(x_1, \dots, x_s)$ $(s \leq \min(m, n))$ for which $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$ have subformulas isomorphic to $R(x_1, \dots, x_s)$.

1. Create 2 pairs of maximal subformulas from $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$, containing only a single predicate symbol: $\left(F1_i(\overline{a_i}),\ F2_i(\overline{b_i})\right)$ - with predicate P_i and $\left(F1_j(\overline{a_j}),\ F2_j(\overline{b_j})\right)$ - with predicate P_j[2]. That is, $F1(a_1, \dots, a_m) = F1_i(\overline{a_i})\ \&\ F1_j(\overline{a_j})$, $F2(b_1, \dots, b_n) = F2_i(\overline{b_i})\ \&\ F2_j(\overline{b_j})$.
2. Extract R_i^* using the algorithm **MCF1** for a pair of formulas $\left(F1_i(\overline{a_i}), F2_i(\overline{b_i})\right)$. Do the same with $\left(F1_j(\overline{a_j}), F2_j(\overline{b_j})\right)$ and obtain the list R_j^*.
3. In a nested loop over the lists R_i^* and R_j^*, check the unifiers $\lambda 2_i$ and $\lambda 2_j$ for inconsistency.

   If the unifiers are consistent, unify them and connect the current common subformulas with the sign &. The obtained formula $R_{i,j}(\overline{x}_{i,j})$ with two predicate symbols P_i and P_j, which defines the MCF of $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$ and their unifiers, is added to $R^*_{(i,j)}$.

   If the unifiers are inconsistent, then go to the next pair of pairs in the lists R_i^* and R_j^*, i.e., go to the next step of the loop.

A block diagram of the algorithm **MCF2** is shown in Fig. 1.

IV. ABOUT THE ALGORITHM MCF2 COMPLEXITY

The number of steps in items 1-2, the complexity of the algorithm **MCF1** implemented to $F1_i(\overline{a_i})$ and $F2_i(\overline{b_i})$, is $O(n_i^{2n_i})$, where n_i is maximal number of arguments in these formulas [1].

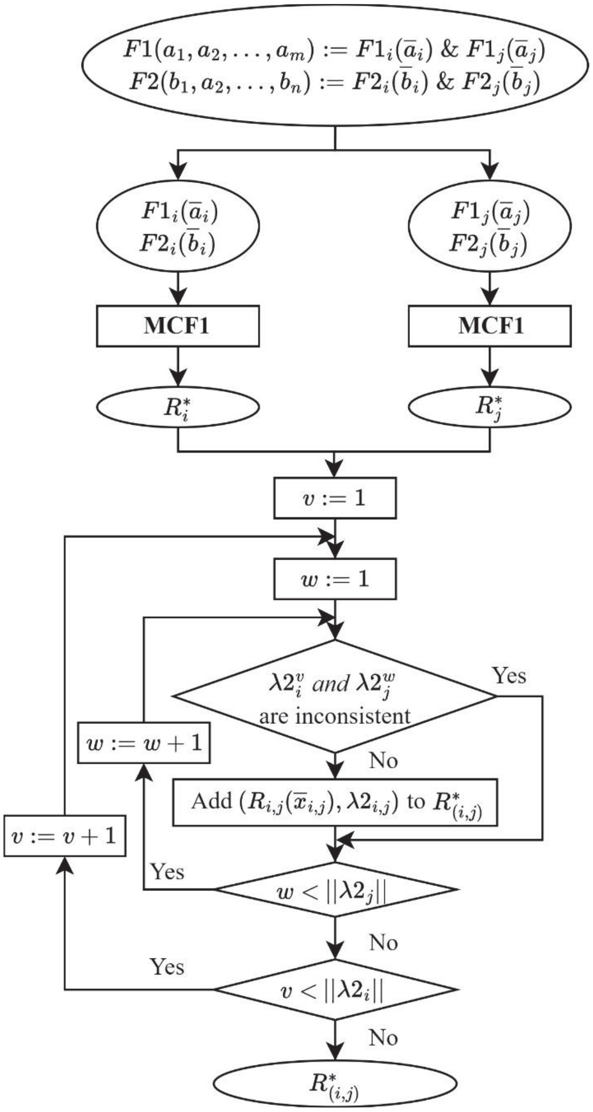

$F1(a_1, a_2, \dots, a_m) := F1_i(\overline{a}_i)\ \&\ F1_j(\overline{a}_j)$
$F2(b_1, a_2, \dots, b_n) := F2_i(\overline{b}_i)\ \&\ F2_j(\overline{b}_j)$
$F1_i(\overline{a}_i)$
$F2_i(\overline{b}_i)$
$F1_j(\overline{a}_j)$
$F2_j(\overline{b}_j)$
MCF1
MCF1
R_i^*
R_j^*
$v := 1$
$w := 1$
$\lambda 2_i^v$ and $\lambda 2_j^w$ are inconsistent
Yes
$w := w + 1$
No
Add $(R_{i,j}(\overline{x}_{i,j}), \lambda 2_{i,j})$ to $R^*_{(i,j)}$
$v := v + 1$
Yes
$w < ||\lambda 2_j||$
No
Yes
$v < ||\lambda 2_i||$
No
$R^*_{(i,j)}$

Fig. 1. Algorithm **MCF2** block diagram.

The complexity of checking for consistency of $\lambda 2_i$ with $\lambda 2_j$ (item 3) is $O(\sum_{v=1}^{||\lambda 2_i||}\sum_{w=1}^{||\lambda 2_j||}(\,||\lambda 2_i||||\lambda 2_j||)) = O(||\lambda 2_i||^2||\lambda 2_j||^2) \leq O(n_i^2 n_j^2) \leq O(n^4)$, where n_i, n_j are the numbers of arguments in the formulas F_i, F_j, respectively.

The main contribution to the **MCF2** algorithm's computational complexity estimation comes from the implementation of item 2, namely, **MCF1** implemented consistently to pairs $\left(F1_i(\overline{a_i}), F2_i(\overline{b_i})\right)$. Computational complexity of the algorithm **MCF2** is $O(n^{2n})$, where n is the maximal number of arguments in subformulas with a single predicate symbol and is not greater than the number of arguments in $F1_i(\overline{a_i})$ and $F2_i(\overline{b_i})$.

[1] The names of constants in different formulas may coincide, and constants with the same names in different formulas may be used as names of different object elements and stand in different places.

[2] Here $\overline{a}_i$ and $\overline{b}_i$ are lists of all arguments that are included in maximal subformulas with predicate P_i. Similarly for the predicate P_j.

V. ALGORITHM MCFN FOR EXTRACTING COMMON PROPERTIES OF OBJECTS DESCRIBED IN THE PREDICATE CALCULUS LANGUAGE WITH SEVERAL PREDICATE SYMBOLS

Let formulas $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$ be elementary conjunctions of predicate formulas with l predicate symbols $P_1, \dots, P_l$, and literals with the same predicate symbol are consecutive.

Consider that the numbering of literals is ordered so that if $i < j$, then the minimum number of arguments for all occurrences of the predicate P_i in $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$ does not exceed the minimum number of arguments for all occurrences of the predicate P_j in $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$.

For example, if

$$F1(a_1, \dots, a_m) = \underbrace{P_i(\cdot)\ \&\ \dots P_i(\cdot)}_{n1\ arguments}\ \&\ \underbrace{P_j(\cdot)\ \&\ \dots P_j(\cdot)}_{m1\ arguments}$$

$$F2(b_1, \dots, b_n) = \underbrace{P_i(\cdot)\ \&\ \dots P_i(\cdot)}_{n2\ arguments}\ \&\ \underbrace{P_j(\cdot)\ \&\ \dots P_j(\cdot)}_{m2\ arguments}$$

and $i < j$, then $\min\{n1, n2\} \le \min\{m1, m2\}$[3].

Algorithm **MCFn** is as follows:

1) $R^* := \emptyset$.
2) Organize the loop by $i = 1, \dots, l/2$.[4]
    a) For $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$, generate two pairs of subformulas $\left(F1_{2i-1}(\overline{a}_{2i-1}),\ F2_{2i-1}(\overline{b}_{2i-1})\right)$ and $\left(F1_{2i}(\overline{a}_{2i}),\ F2_{2i}(\overline{b}_{2i})\right)$, containing the single predicate symbol P_{2i-1} and P_{2i}, respectively.
    b) For pairs of subformulas $\left(F1_{2i-1}(\overline{a}_{2i-1}),\ F2_{2i-1}(\overline{b}_{2i-1})\right)$ and $\left(F1_{2i}(\overline{a}_{2i}),\ F2_{2i}(\overline{b}_{2i})\right)$ using the algorithm **MCF1** extract the lists R^*_{2i-1} and R^*_{2i}.
    c) For each of the obtained pairs from R^*, check $\lambda2_{2i-1}$ with $\lambda2$, then $\lambda2_{2i}$ with $\lambda2$ for consistency.
    If the unifiers are consistent, then
        I. call the algorithm **MCF2** for $(F1_{2i-1}, F2_{2i-1})$ and $(F1_{2i}, F2_{2i})$, get a list $R^*_{(2i-1,\,2i)}$ of pairs, such as $\left(R_{2i-1,\,2i}(\overline{x}_{2i-1,2i}),\ \lambda2_{2i-1,2i}\right)$[5];
        II. merge $\lambda2$ and $\lambda2_{2i-1,2i}$, attach $R(\overline{x})$ and $R_{2i-1,2i}(\overline{x}_{2i-1,2i})$. Write the resulting formula $R(\overline{x}) := R(\overline{x}) \cup R_{2i-1,2i}(\overline{x}_{2i-1,2i})$, specifying the MCF of formulas $F1(a_1, \dots, a_m)$ and $F2(b_1, \dots, b_n)$, and their unifiers in R^*.
    Otherwise, go to the next step in the cycle.

A block diagram of the algorithm **MCFn** is shown in Fig. 2.

VI. ABOUT THE ALGORITHM MCFN COMPLEXITY

The number of steps in items 1-2b is the complexity of **MCF1** implemented to pairs $\left(F1_{2i-1}(\overline{a}_{2i-1}),\ F2_{2i-1}(\overline{b}_{2i-1})\right)$ and $\left(F1_{2i}(\overline{a}_{2i}),\ F2_{2i}(\overline{b}_{2i})\right)$. It is $O(n_{2i-1}^{2n_{2i-1}} + n_{2i}^{2n_{2i}}) \le O(n^{2n})$, where n_{2i-1}, n_{2i}, n are the maximal numbers of arguments in subformulas with a single predicate symbol P_{2i-1}, P_{2i} and in the initial formulas respectively.

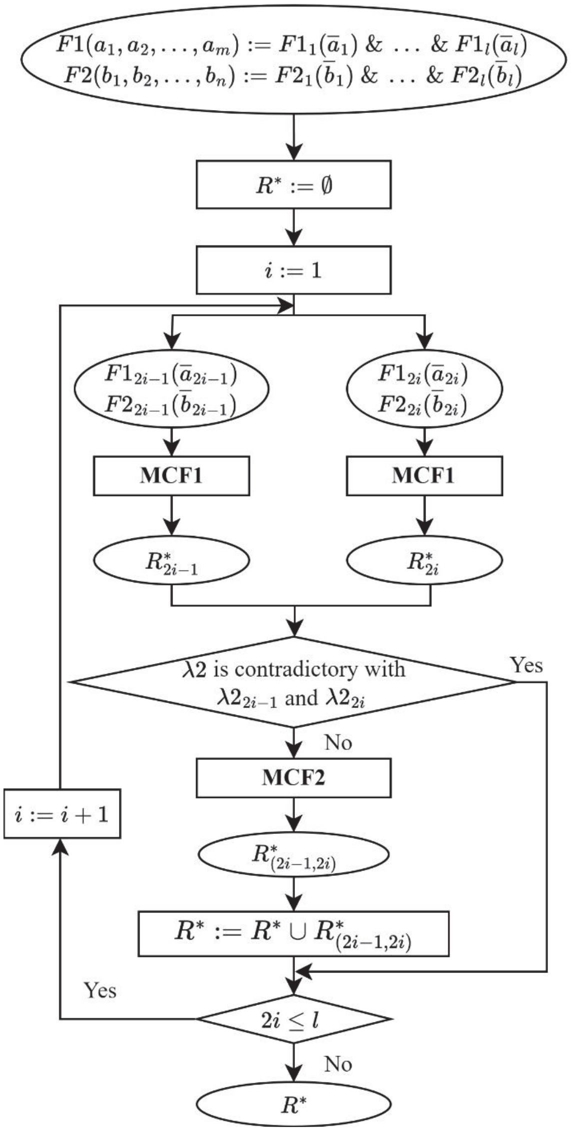

$F1(a_1, a_2, \dots, a_m) := F1_1(\overline{a}_1)\ \&\ \dots\ \&\ F1_l(\overline{a}_l)$
$F2(b_1, b_2, \dots, b_n) := F2_1(\overline{b}_1)\ \&\ \dots\ \&\ F2_l(\overline{b}_l)$
$R^* := \emptyset$
$i := 1$
$F1_{2i-1}(\overline{a}_{2i-1})$
$F2_{2i-1}(\overline{b}_{2i-1})$
$F1_{2i}(\overline{a}_{2i})$
$F2_{2i}(\overline{b}_{2i})$
MCF1
MCF1
R^*_{2i-1}
R^*_{2i}
$\lambda2$ is contradictory with $\lambda2_{2i-1}$ and $\lambda2_{2i}$
Yes
No
MCF2
$i := i + 1$
$R^*_{(2i-1,2i)}$
$R^* := R^* \cup R^*_{(2i-1,2i)}$
Yes
$2i \le l$
No
R^*

Fig. 2. Algorithm **MCFn** block diagram.

In item 2c, the complexity of checking for consistency of $\lambda2_i$ with $\lambda2_{2i-1}$, $\lambda2_i$ with $\lambda2_{2i}$, is $O(||\lambda2_{2i-1}||||\lambda2_{2i}||) \le O(n_{2i-1}n_{2i}) \le O(n^2)$.

In items 2ci-2cii, the complexity **MCF2**, is $O(n^{2n})$.

The number of executions of **MCFn** is equal to half the number of predicate symbols $l/2$. At the same time, the number of steps in Items 1-2cii is $O(n^{2n}) + O(n^2) + O(n^{2n})$.

[3] The order of predicate symbols may depend on the specifics of the initial data. For example, if in each formula there are one or two occurrences of a predicate symbol containing more than half of all variables, then by assigning the number 1 to this predicate, we can find a partial unifier for more than half of the variable values in one step (in the worst case, in 4 steps).

[4] If the initial number of predicates l is odd, l will be increased by one ($l := l + 1$), and the elementary conjunctions with the fictive predicate will be assumed to be empty ($F1_l(\overline{a}_l) := \emptyset$, $F2_l(\overline{b}_l) := \emptyset$).

[5] $\lambda2_{2i-1}$ and $\lambda2_{2i}$ only contain those partial unifiers that do not contradict the unifiers of $\lambda2$.

Summing up the obtained estimates of the number of steps, we obtain an estimate of the number of steps of the algorithm **MCFn** $O(\sum_{i=1}^{l/2}(n^{2n}+n^2+n^{2n}))=O(n^{2n})$.

Thus, the main contribution to the **MCFn** algorithm's computational complexity estimation also comes from the implementation of **MCF1**, whose computational complexity is $O(n^{2n})$.

VII. CONCLUSION

The paper presents two algorithms **MCF2** and **MCFn** for extraction maximal common subformulas (up to the precision of argument names) of two elementary conjunctions. The implementation was carried out in the Python [4] programming language.

Extraction of such subformulas is an important actual task of searching for common properties of complex structured objects (CSO) described in the predicate calculus language, when solving such problems as

- level descriptions of classes for significantly decreasing the computational complexity of CSO recognition [5,6];
- fuzzy recognition of CSO [7];
- creation of a logic ontology for CSO [8].

ACKNOWLEDGMENT

Kosovskaya T. M. acknowledges Saint-Petersburg State University for a research project 95438429.

Zhou Juan acknowledges the support of the China Scholarship Council program (Project ID: 202108620001).

REFERENCES

1. J. Zhou and T. M. Kosovskaya. “Algorithm for extraction common properties of objects described in the predicate calculus language with a single predicate symbol", *Vestnik of Saint Petersburg University. Mathematics. Mechanics. Astronomy*, vol. 11 (69), issue 4, 2024. (In print) DOI: 10.21638/spbu01.2024.409
2. T. M. Kosovskaya and D. A. Petrov. “Extraction of a maximal common sub-formula of predicate formulas for the solving of some Artificial Intelligence problems", *Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes*, vol. 13, no. 3, pp. 250–263, 2017. DOI: 10.21638/11701/spbu10.2017.303
3. T. M. Kosovskaya and J. Zhou, “Algorithms of Isomorphism of Elementary Conjunctions Checking", *Pattern Recognition and Image Analysis*, vol. 34, no. 1, pp. 102–109, 2024. DOI: 10.1134/S1054661824010103
4. E. Matthes, *Python Crash Course: A Hands-On, Project-Based Introduction to Programming*, 3rd Edition. No Starch Press, San Francisco, 2023.
5. T. Kosovskaya, *Predicate Calculus as a Tool for AI Problems Solution: Algorithms and Their Complexity*. Chapter 3 in Intelligent System. Open access peer-reviewed Edited volume. Edited by Chatchawal Wongchoosuk Kasetsart University, pp. 1 – 20, 2018. [Online]. Available: https://www.intechopen.com/chapters/58698
6. T. M. Kossovskaya, “Level descriptions of classes for decreasing step number of pattern recognition problem solving described by predicate calculus formulas", *Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes*, vol. 10, no. 2, pp. 62–70, 2008.
7. T. Kosovskaya, “Fuzzy Recognition by Logic-Predicate Network", *Advances in Science, Technology and Engineering Systems Journal*, vol. 5, no. 4, pp. 686–699, 2020. https://doi.org/10.25046/aj050482
8. T. M. Kosovskaya and N. N. Kosovskii, “Extraction of common properties of objects for creation of a logic ontology", *Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes*, vol. 18, no. 1, pp. 37–51, 2022. https://doi.org/10.21638/11701/spbu10.2022.103

Scheduling Theory Problem with an Unfixed Start Time

Ketevan Kutkhashvili The
University of Georgia
Tbilisi, Georgia
e-mail: Kkutkhashvili@yahoo.com

Neli Kilasonia
GTU Institute of Control Systems
Tbilisi, Georgia
e-mail: Kilasonianeli1@gmail.com

https://doi.org/10.62343/csit.2024.10

***Abstract* — The paper discusses the problem when the achievement of the optimal goal depends on the sequence of tasks, i.e., when it is necessary to arrange them in time. Such problems are known to belong to calendar scheduling problems and are in general of NP complexity. It is possible to build a polynomial algorithm only in the case of a certain configuration of the initial parameters. In the paper, one such configuration is considered and financial cost minimization is taken as a criterion for achieving the goal.**

***Keywords* — discret optimization, scheduling problem, algorithm.**

I. INTRODUCTION

Solving various technical, economic, organizational and social problems is often impossible without building a mathematical model. Especially in the case, when a very large amount of data needs to be processed to achieve some goal, and this data is of a discrete nature, it becomes necessary to use discrete optimization methods.

Many practical problems, for instance, transport or management and running of industry process, under conditions of fixed resources require scheduling of tasks at a time. The given system of tasks must be implemented by certain set of resources.

In terms of tasks system and the given properties of resources with certain restrictions to them we have to construct an efficient algorithm of the task implementation sequence, which gives possibility to attain efficiency by certain measure of optimum. Under measure of optimum there may be considered scheduling length in terms of time, average time of being in the tasks system or maximum cost of the system.

As it is known, schedule problems are of NP difficulty [1]-[5] and requires great deal of applications of the modern applied mathematics. Basically difficulty is caused by great volume of tasks. In such situations, new methods are created to receive best decisions and practical recommendations of planning and control [2]-[5].

Because of above-mentioned, for the certain problem it is actual to construct comparatively accurate mathematical model and create algorithms, which totally use the specific character of the problem and give possibility of the optimal decision in polynomial time.

For this problem, an algorithm of complexity P is built, which is based on a combined method consisting of the branch and bound and statistical methods.

II. PROBLEM FORMULATION

Given a set of tasks, X={ξ1,...,ξn}, which must be executed in [0, T] period by means of P={P1,...,Pm}, j=1,..., m processors. Obviously, the problem is relevant when n is a much larger number than m. Processors are partially interchangeable. They can differ both in terms of speed and functionality. Therefore, the matrix [τij]i=1,...,m, j=1,...,n , whose τij element shows the duration of execution of the task ξj on the Pi processor is given in advance.

Also is known {ω ij } i=1,...,m, j=1,...,n matrix, whose ωij element shows the price of execution of task ξj on Pi processor. In order to simplify the problem, let's assume that the tasks are mutually independent and additional resources are not taken into account.

The arrival time of each task in the system is not known exactly, but the estimated time is given in the form of an interval $t_i^0 \in [a_i; b_i]$. where a_i is the estimated minimum time for the i-th task to enter the system, and b_i is the estimated maximum time for the i-th task to enter the system.

The schedule S must be constructed in such a way that one task can be executed on one processor without interrupts, and one processor cannot execute several tasks at the same time. Our goal is to assign to each task a number t0ξj (Pi), which at the time of the start of processing of the j-th task indicates the Pi processor on which the j-th task should be executed. Among all such mappings S that satisfy the above conditions, one needs to find a mapping S* for which the following conditions are satisfied

$$\rho(S^*) = \min_S \rho(S) = \min_S \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{m} \omega_{ij} \, f_{ij}(s),$$

where $f_{ij}(s)$ is the processing time of the j-th task according to schedule S shows on the Pi processor, and ρ expresses the cost of the task system.

III. ALGORITHM DESCRIPTION

The algorithm includes the following main blocks:

- Determination of Xk system and formation of τ_i^k , $t_i^k \in [a_i; b_i]$ and $\omega_{ij}(Z)$ for the components included in it;
- Determination of the exact time of entry into the system for each task $t_i^0 \in [a_i; b_i]$ or $t_i^0 = a_i$;
- Branching rule - it defines the branching strategy and process of the search tree. Its objective is to divide the solution space into disjoint subsets, each of which is either cut or formed for consideration at a later stage, $X_1^{(k)} \subset X$;

- Selection rule - selects the tree vertex from which the next branch should start. For this, we choose $\eta = \min_{\xi_i \in X_1^k} (t_{\xi_i}^{(k)} + \tau_i)$;
- Calculation of the value of the characteristic function and expert evaluation - based on this calculated value, the branch of the tree is evaluated and those branches that do not contain the desired solution are cut. $\rho_k = \rho_k + (t_{\xi_{(k)}}^{(k)} + \tau_{i_l} - d_{i_l}) * \omega_{i_l}$;
- Forming a new step. mint $\mathrm{min}t_{P_{(j_1^*)}}^{(2)} = mint_{\xi_{(1)}^*}^{(1)} + \tau_{i_1}$, $X_1^{(k+1)} \subset X$. A transition to the second step is made if the endpoint of the interval [0, T] is not reached. Otherwise, we go to the next stage;
- Construct and compute a lower bound function - which matches each private solution to its lower bound value. $\rho_l(S_k) \leq \rho_l(S_{k+1})$;
- Checking and correction of the upper limit of the value - the upper limit of the value is initially equal to any of the full prices, which are known in advance by approximation or reasonable judgment. And if there is no such opinion in advance, then it is equal to mechanical infinity;
- Establishing a schedule for the case when $t_i^0 = b_i$;
- Selecting a common distribution from both trees and determining the best starting time for tasks whose arrival time in the system is uncertain;
- Checking the end of the algorithm operation - the operation of the algorithm is completed if all subsets of the set X_k turn out to be empty.

Due to the fact that the influence of the abundance of additional resources is neglected during scheduling, and also that the tasks are mutually independent, it is possible to construct an algorithm of O(n3) complexity. In addition, it is possible to include an expert in the process of executing the algorithm to specify the time of entry into the system of tasks.

For this problem, an algorithm of complexity P is built, which is based on a combined method consisting of the branch and bound and statistical methods.

REFERENCES

1. V. Tanaev, V. Gordon, I. Shafranski⊠ Scheduling Theory. Single-Stage Systems, Dordrecht ; Boston : Kluwer Academic Publishers,1994.
2. P. Chrლtienne, E.G. Coffman, J.Karel Lenstra Scheduling Theory and Its Applications. Wiley; 1st edition, 1995.
3. K. Kutkhashvili., Gabisonia V. The algorithm for a problem of the scheduling theory considering two criterion. Georgian Academy of Sciences A. Eliashvili Institute of Control Systems Proceedings, Tbilisi, ⊠13. 2009.
4. P.Brucker, B.Jurisch, B.Sievers A branch and bound algorithm for the job-shop scheduling problem. Discrete Applied Mathematics. Volume 49, Issues 1–3, 30 March 1994, Pages 107-127
5. F. Werner, L. Burtseva, Y. Sotskov, Algorithms for Scheduling Problems. MDPI, Basel, Switzerland. 2018.

Characterization of Sub-Gaussian Random Elements in Banach Spaces

George Giorgobiani Muskhelishvili
Institute of Computational Mathematics
of the Georgian Technical University,
Tbilisi, Georgia
e-mail: giorgobiani.g@gtu.ge

Vakhtang Kvaratskhelia Muskhelishvili
Institute of Computational Mathematics
of the Georgian Technical University,
Tbilisi, Georgia
e-mail: v.kvaratskhelia@gtu.ge

Vazha Tarieladze
Muskhelishvili Institute
of Computational Mathematics
of the Georgian Technical University,
Tbilisi, Georgia
e-mail: v.tarieladze@gtu.ge

https://doi.org/10.62343/csit.2024.11

***Abstract* — We present without proof the following result: if X is a Banach space and a weakly sub-Gaussian random element in X induces the 2-summing operator, then it is T −sub-Gaussian provided that X is a reflexive type 2 space. Using this result we obtain a characterization of weakly sub-Gaussian random elements in a Hilbert space which are T −sub-Gaussian.**

***Keywords* — Sub-Gaussian random variable, Gaussian random variable, weakly sub-Gaussian random element, T −sub-Gaussian random element, Banach space, Hilbert space.**

I. INTRODUCTION

Let $(\Omega, \mathcal{A}, \mathbf{P})$ be a probability space. Following [7] we call a real-valued measurable function $\xi : \Omega \to \mathbb{R}$ a sub-Gaussian random variable if there exists a real number $a \geq 0$ such that for every real number t the following inequality is valid

$$\mathbb{E}\, e^{t\xi} \leq e^{\frac{1}{2}a^2t^2},$$

where $\mathbb{E}$ stands for the mathematical expectation.

To each random variable ξ it corresponds a parameter $\tau(\xi) \in [0, +\infty]$ defined as follows (we agree $\inf(\emptyset) = +\infty$):

$$\tau(\xi) = \inf\left\{a \geq 0 : \quad \mathbb{E}\, e^{t\xi} \leq e^{\frac{1}{2}a^2t^2}, \quad t \in \mathbb{R}\right\}.$$

A random variable ξ is sub-Gaussian if and only if $\tau(\xi) < +\infty$ and $\mathbb{E}\xi = 0$. Moreover, if ξ is a sub-Gaussian random variable, then for every real number t

$$\mathbb{E}\, e^{t\xi} \leq e^{\frac{1}{2}\tau^2(\xi)t^2}$$

and

$$\left(\mathbb{E}\xi^2\right)^{\frac{1}{2}} \leq \tau(\xi)\,.$$

If ξ is a Gaussian random variable with $\mathbb{E}\xi = 0$, then ξ is sub-Gaussian and

$$\left(\mathbb{E}\xi^2\right)^{\frac{1}{2}} = \tau(\xi)\,.$$

Remark 1.1: [3, Example 1.2]. If ξ is a bounded random variable, i.e. if for some constant $c \in \mathbb{R}$ with $0 < c < +\infty$, we have $|\xi| \leq c$ a.s. and $\mathbb{E}\xi = 0$, then ξ is sub-Gaussian and $\tau(\xi) \leq c$.

Denote by $\mathcal{SG}(\Omega, \mathcal{A}, \mathbb{P})$, or in short, by $\mathcal{SG}(\Omega)$ the set of all sub-Gaussian random variables defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$. $\mathcal{SG}(\Omega)$ is a vector space over $\mathbb{R}$ with respect to the natural point-wise operations; moreover, the functional $\tau(\cdot)$ is a norm on $\mathcal{SG}(\Omega)$ (provided that random variables which coincide almost surely are identified) and $(\mathcal{SG}(\Omega), \tau(\cdot))$ is a Banach space [2]. For $\xi \in \mathcal{SG}(\Omega)$ instead of $\tau(\xi)$ we will write also $\|\xi\|_{\mathcal{SG}(\Omega)}$.

More information about the sub-Gaussian random variables can be found for example in [5], [6].

Let X be a Banach space over $\mathbb{R}$ with a norm $\|\cdot\|$ and X^* be its dual space. The value of the linear functional $x^* \in X^*$ at an element $x \in X$ is denoted by the symbol $\langle x^*, x\rangle$.

Following [11, p. 88] a mapping $\xi : \Omega \to X$ is called a random element (vector) in X if $\langle x^*, \xi\rangle$ is a random variable for every $x^* \in X^*$.

If $0 < p < \infty$, then a random element ξ in a Banach space X:

- has *a weak p-th order*, if $\mathbb{E}\,|\langle x^*, \xi\rangle|^p < \infty$ for every $x^* \in X^*$;
- is *centered*, if ξ has a weak first order and $\mathbb{E}\,\langle x^*, \xi\rangle = 0$ for every $x^* \in X^*$.

To each weak second-order centered random element ξ in a separable Banach space X it corresponds a mapping $R_\xi : X^* \to X$ such that

$$\langle y^*, R_\xi x^*\rangle = \mathbb{E}\,\langle y^*, \xi\rangle\langle x^*, \xi\rangle, \quad \text{for every} \quad x^*, y^* \in X^*,$$

which is called *the covariance operator of* ξ [11, Corollary 2 (p.172)].

A random element $\xi : \Omega \to X$ is called *Gaussian*, if for each functional $x^* \in X^*$ the random variable $\langle x^*, \xi\rangle$ is Gaussian.

A mapping $R : X^* \to X$ is said to be *a Gaussian covariance*, if there exists a Gaussian random element in X whose covariance operator is R.

A random element $\xi : \Omega \to X$ will be called *weakly sub-Gaussian* [10], if for each $x^* \in X^*$ the random variable $\langle x^*, \xi\rangle$ is sub-Gaussian.

A random element $\xi : \Omega \to X$ will be called $T-$*sub-Gaussian* [9] (or $\gamma-$sub-Gaussian [4]), if there exists a probability space $(\Omega', \mathcal{A}', \mathbf{P}')$ and a centered Gaussian random element $\eta : \Omega' \to X$ such that for each $x^* \in X^*$

$$\mathbb{E}\, e^{\langle x^*, \xi\rangle} \leq \mathbb{E}\, e^{\langle x^*, \eta\rangle}\,. \tag{1.1}$$

Theorem 1.2: (a) If X is finite-dimensional Banach space, then every weakly sub-Gaussian random element in X is $T-$sub-Gaussian.

(b) If X is infinite-dimensional separable Banach space, then there exist a weakly sub-Gaussian random element in X. which *is not* $T-$sub-Gaussian.

To every weakly sub-Gaussian random element $\xi : \Omega \to X$ we associate *the induced linear operator*

$$T_\xi : X^* \to \mathcal{SG}(\Omega)$$

defined by the equality:

$$T_\xi x^* = \langle x^*, \xi\rangle \qquad \text{for all} \quad x^* \in X^*.$$

Let X and Y be Banach spaces, $L(X, Y)$ be the space of all continuous linear operators acting from X to Y. An operator $T \in L(X, Y)$ is called 2-(absolutely) summing if there exists a constant $C > 0$ such that for each natural number n and for every choice $x_1, x_2, \dots, x_n$ of elements from X we have

$$\left(\sum_{k=1}^{n} ||Tx_k||^2\right)^{1/2} \le C \sup_{||x^*||_{X^*}\le 1} \left(\sum_{k=1}^{n} |\langle x^*, x_k\rangle|^2\right)^{1/2} . \quad (1.2).$$

For a 2-summing $T : X \to Y$ we denote the minimum possible constant C in (1.2) by $\pi_2(T)$.

We say that a Banach space X has type 2, if there exists a finite constant $C \ge 0$ such that for each natural number n and for every choice $x_1, x_2, \dots, x_n$ of elements from X we have

$$\left(\int_0^1 \left\|\sum_{k=1}^{n} r_k(t)x_k\right\|^2 dt\right)^{1/2} \le C \left(\sum_{k=1}^{n} \|x_k\|^2\right)^{1/2} ,$$

where $r_1(\cdot), \dots, r_n(\cdot)$ are Rademacher functions on $[0, 1]$. An example of a type 2 space is a Hilbert space as well as the spaces $l_p, L_p([0, 1]), 2 \le p < +\infty$.

II. Main results

The following theorem is a slightly corrected version of [8, Theorem 1.7].

Theorem 2.1: Let X be a separable Banach space. For a *weakly sub-Gaussian* random element $\xi : \Omega \to X$ consider the assertions:

(i) ξ is $T-$sub-Gaussian.

(ii) $T_\xi : X^* \to \mathcal{SG}(\Omega)$ is a 2-summing operator.

Then:

(a) $(i) \implies (ii)$;

(b) The implication $(ii) \implies (i)$ is true provided that X is a reflexive Banach space of type 2.

Consider now the case when $X = H$, where H denotes an infinite-dimensional separable Hilbert space with the inner product $\langle\cdot,\cdot\rangle$. As usual we identify H^* with H by means of the equality $H^* = \{\langle\cdot, y\rangle : y \in H\}$.

Theorem 2.1 implies the following result, which is related with the similar assertion contained in [1, Proposition 3.1].

Theorem 2.2: Let H be an infinite-dimensional separable Hilbert space. For a weakly sub-Gaussian random element $\xi : \Omega \to H$ the following statements are equivalent:

(i) ξ is $T-$sub-Gaussian.

(ii_m) For each orthonormal basis (φ_k) of H

$$\sum_{k=1}^{\infty} \tau^2(\langle\varphi_k, \xi\rangle) < \infty .$$

In connection with Theorem 2.2 naturally arises the following question: is it possible to replace the condition (ii_m) by the following (weaker) condition?

(ii_w) *There is an orthonormal basis* (φ_k) *of* H *such that*

$$\sum_{k=1}^{\infty} \tau^2(\langle\varphi_k, \xi\rangle) < \infty .$$

In [1, Remark 4.3] it is claimed that the answer to this question *is positive*.

At the end we pose another interesting question related to Theorem 2.2: does there exist a bounded centered random element ξ in a separable infinite-dimensional Hilbert space H such that

$$\sum_{k=1}^{\infty} \tau^2(\langle\psi_k, \xi\rangle) = \infty$$

for *every* orthonormal bases (ψ_k) of H?

ACKNOWLEDGMENT

The work was partially supported by the European Union's HORIZON EUROPE Grant Project GAIN.

REFERENCES

1. R. G. Antonini, "Subgaussian random variables in Hilbert spaces", Rend. Sem. Mat. Univ. Padova, vol. 98, pp. 89–99, 1997.
2. V. V. Buldygin and Yu. V. Kozachenko, "Sub-Gaussian random variables", Ukrainian Mathematical Journal, vol. 32, pp. 483–489, 1980.
3. V. V. Buldygin and Yu. V. Kozachenko, Metric Characterization of Random Variables and Random Processes. American Mathematical Soc., 2000.
4. R. Fukuda, "Exponential integrability of sub-Gaussian vectors", Probab. Theory Relat. Fields, vol. 85, no. 4, pp. 505–521, 1990.
5. G. Giorgobiani, V. Kvaratskhelia and M. Menteshashvili, "Unconditional Convergence of Sub-Gaussian Random Series", Pattern Recognition and Image Analysis, vol. 34, no. 1, pp. 92–101, 2024.
6. G. Giorgobiani, V. Kvaratskhelia and V. Tarieladze, "Notes on subGaussian random elements", In Applications of Mathematics and Informatics in Natural Sciences and Engineering: AMINSE 2019, Tbilisi, Georgia, pp. 197–203, Springer International Publishing, 2020.
7. J. P. Kahane, "Proprietes locales des fonctions a series de Fourier aleatoires", Studia Math., 19, pp. 1–25, 1960.
8. V. Kvaratskhelia, V. Tarieladze and N. Vakhania, "Characterization of γ-Subgaussian Random Elements in a Banach Space", Journal of Mathematical Sciences, vol. 216, no. 4, pp. 564–568, 2016.
9. M. Talagrand, "Regularity of gaussian processes", Acta Math., vol. 159, no. 1-2, pp. 99–149, 1987.
10. N. Vakhania, "On subgaussian random vectors in normed spaces", Bull. Georgian Acad. Sci., vol. 163, no. 1, pp. 8–11, 2001.
11. N. N. Vakhania, V. I. Tarieladze and S. A. Chobanyan, Probability distributions on Banach spaces. Dordrecht: Reidel, 1987.

Secure Online Voting Scheme Using Steganography

Mariam Haroutunian
Institute for Informatics and Automation
Problems of NAS RA
Yerevan, Armenia
e-mail: armar@sci.am

Arsen Margaryan
MoodMap AI
Yerevan, Armenia
e-mail: margaryanarsen83@gmail.com

Karen Mastoyan
Institute for Informatics and
Automation
Problems of NAS RA
Yerevan, Armenia
e-mail: kmastoyan@yandex.com

https://doi.org/10.62343/csit.2024.12

***Abstract* — Distrust in voting is not a rare phenomenon even in developed countries. Electronic voting, however, appeared as an alternative, but is still not practiced on a large scale. This is due to the fact that despite the huge number of articles it is not yet possible to completely ensure security, verifiability and anonymity. It is hard to create a system or a protocol fulfilling all requirements, especially unconditionally. Designing effective voting systems is challenging because these aspects often conflict with each other. There are issues that need to be resolved. Our aim is to develop a secure online voting system in Armenia suitable for a variety of choices and easily adaptable to different cases. In this paper we suggest to use steganography to create an e-voting model with added security. This research proposes a novel architecture for an internet voting system that incorporates steganography techniques to enhance the security of the system. In the proposed architecture steganography is used to hide the voter's identity and voting preferences within the data packets transmitted between the storage, that keeps all the votes, and the counting server. The use of statistical steganography ensures that the encrypted vote is undetectable, while hypothesis testing ensures the integrity of the voting process.**

***Keywords* — e-voting, security, privacy, trust, steganography**

I. INTRODUCTION

In recent years, online voting has been offered as an alternative means of conducting elections. Today there are many e-voting schemes and systems, which allows their use in many areas of public administration. Internet voting offers many advantages compared to traditional types of voting. The main benefit is to ensure better accessibility and convenience of voting (including for people with disabilities, during pandemic or citizens abroad), which can lead to higher participation in elections. Another advantage of online voting is higher speed and accuracy of results data processing. Very positive are lower voting costs for both voters and organizers.

Internet voting systems have gained popularity and have been used for government elections, referendums, municipal elections in such countries as Estonia, Switzerland, Canada, the USA, France and others. Our aim is to develop a secure online voting system in Armenia, since the electronic voting can improve the country’s image.

Because of its fundamental importance for democratic societies, Internet voting is subject to high legal standards, in particular security requirements for the voting method. However, these legal standards and resulting safety requirements partially contradict each other. As a result, many challenges arise to address mandatory security requirements.

A huge number of publications are devoted to the problems related to e-voting systems. Here we referee to the most recent surveys which in turn referee to a wide range of literature. Several remote e-voting solutions based on blockchain technology have been proposed. The survey [1] provides a comprehensive overview of the blockchain - based e-voting systems currently being implemented by various countries and companies and proposed for academic research. At the same time the challenges faced by blockchain e-voting systems are analysed here. [2] claims to apprehend the security and data management challenges in blockchain and provides an improved manifestation of the electronic voting process. In [3] the most revealing e-voting solutions based on blockchain technology are reviewed.

Indeed, in the scientifc literature, more and more research works propose e-voting applications based on blockchain. Nevertheless, only some of the proposed solutions have been implemented in real life and none of them have been tested on a large scale. Therefore, it is very difficult to conclude that blockchain is a completely secure alternative for national election at present. Although the principles on which blockchain is based are secure, e-voting applications are still vulnerable to several attacks. This makes it very challenging to guarantee the integrity of an election, which is problematic given the stakes of such an application. The National Academies of Sciences report [4] states that “blockchain technology does little to solve the fundamental security issues of elections, and indeed, blockchains introduce additional security vulnerabilities”. Blockchain technology is designed to keep information secure once it is received. It cannot defend against the multitude of threats to that information before it is entered in the blockchain, and voters cannot verify their votes are entered into the blockchain correctly without compromising ballot secrecy. Recording ballots on a blockchain also risks ballot secrecy if encryption keys are not properly protected or software errors allow decryption of individual ballots.

State-of-the-art research includes, in particular [5], where a blockchain-based voting mechanism is presented underlying 5G Network, [6] that initiated the concept of E-voting attacks in the IoT-oriented smart cities, [7] where Re-Encryption Mix-Net is suggested to provide an efficient cryptographic anonymous channel for useful applications such as e-voting and web browsing.

Information theoretic notions and tools are also considered in various problems of e-voting. In particular, for information theoretic study we referee to [8]. In [9] an information-theoretic model of a vote counting system with well-defined criteria for evaluating such a system with respect to integrity, privacy and verifiability is presented. The information-theoretic approach proved the impossibility of achieving perfect integrity, perfect verifiability and perfect privacy and the existence of the tradeoffs among these criteria.

In this paper we suggest to use steganography to create an e-voting model with added security. Steganography is the science of hiding information within other information. It is a technique that has been used for centuries to convey secret messages, and in recent years, it has found many applications in the field of computer security. In the proposed architecture, steganography is used to hide the voter's identity and voting preferences within the data packets transmitted between the storage, that keeps all the votes, and the counting server.

The previous information-theoretic research on the model of stegosystem with active adversary [10], where two-stage statistical hypothesis testing approach was studied, motivated to apply it in e-voting systems.

Our goal is to create a system that is suitable for a variety of choices and easily adaptable to different cases.

II. REQUIREMENTS

The online voting process consists of several stages. All these stages are present in traditional elections and must be carried out in exactly the same order. Online voting should allow voters to *securely* and *secretly* record their votes. Also the connection between the voter's computer and the election server must be secure, since the third party may try to change the votes while they are being transmitted to the server. Therefore, it is necessary to use *secure communication* over the Internet. It is also necessary to use different cryptographic methods, but at the same time the solution must be *user-friendly* to ensure maximum efficiency. There are quite a few cryptographic schemes which fulfill wide requirements for electronic elections. Their only disadvantage is inconvenience: they use sophisticated cryptographic tools that make them hard to implement and require expertise in various fields.

The e-voting system must be *reliable* to prevent election fraud (insider attacks) or attacks on the system from outside. When implementing a technical solution, it is necessary to take into account all types of attacks. *Interoperability* of all technical components and services used must be ensured, compatibility and correct functionality of the system must be assured, therefore suitable open electronic data interchange standards should be used.

The system must be operational under any circumstances, even if some problems arise (for example, with the network), equipment failures, hacker attacks, etc.). The reliability of the system must be tested, evaluated and improved constantly. The Internet voting system must be *uninterruptedly accessible* to voters. Its interface should be easy to use and all voters should be able to access it equally, regardless of age, education or physical condition. Ballots should be written as clearly as possible so that voters can minimize the possibility of votes being miscounted due to any misleading aspect of the ballot design.

The main security characteristics of e-voting system can be formulated in the following list:

- *Authenticity:* only eligible voters can participate in the election;
- *Eligibility*: all ballots that are to be counted are sent by eligible voters,
- *Completeness*: all ballots are counted, no ballot is ignored or erased;
- *Stability*: no ballot is subject to change after being sent to ballot storage;
- *Anonymity*: make ballots indistinguishable from one another, not to find who is the voter;
- *Privacy*: ensures that there is no link between a voter and his vote, in other words, protects any information about the voter;
- *Unreusability or uniqeness*: voter can vote only once, does not permit any voter to cast more than once;
- *Verifiability*: voter can check if his vote was counted correctly and was not changed;
- *Confidentiality or non-coercibility*: a voter should not have a record of voting choice, not to be able to prove how he has voted;
- *Accuracy*: ensure that the declared results correspond precisely to the election results, that votes are recorded correctly;
- *Integrity*: Votes cannot be altered or deleted, all valid votes encountered in the ballot box, and only those, will be included in the count;
- *Accountability*: The system has the ability to verify that votes are correctly counted.

It is hard to create a system or a protocol fulfilling all requirements, especially unconditionally. Designing effective voting systems is challenging because these aspects often conflict with each other. For example, one of such challenges is trying to keep votes anonymous while still being able to verify them. Another challenge is to ensure identification and maintain voter privacy. Hence, constant investigations and improvements are needed.

III. THE STRUCTURE OF ELECTION PROCESS

The online election process (Fig. 1) starts with so called *setting phase*, where the staff creates ballot, election parameters are formed and published. These parameters include encryption keys, the number of candidates, list of voters, etc. The election is announced with sending login credentials to users.

The next is the *ballot filling phase,* during which the voter login to user interface and prepare his/her vote. The result should be an e-ballot with voter's personal info embedded in it.

In the *ballot registration phase* the ballot with embedded info and voter's personal number are sent to public ballot storage.

The voter's info must be removed from the ballot and as a result a ballot without voter info is collected, this is called *ballot anonymity phase*. The authenticity of the ballot can be checked via private keys that are only given to election committee.

At last in the *counting phase* the committee uses a personal key to decode all the ballots and count the voices.

Designing an electronic voting system involves several key components to ensure the main security characteristics. Let us discuss the correspondence of each step of e-voting to the main security requirements. Process 1 on the Fig. 1 are the *Login* steps of the staff and voters. The secure management access must solve the *authenticity challenge*. Authentication methods may include biometric verification (e.g., fingerprint or facial recognition), government-issued ID verification, or unique login credentials.

Process 2 is the *creation of the Ballot.* The system should provide a mechanism for creating and managing electronic ballots. Ballots may contain candidates' names, party affiliations, and other relevant information. Ballot creation should be customizable to accommodate various election types (e.g., local, national, or organizational elections). At this stage the *security* of information must be ensured.

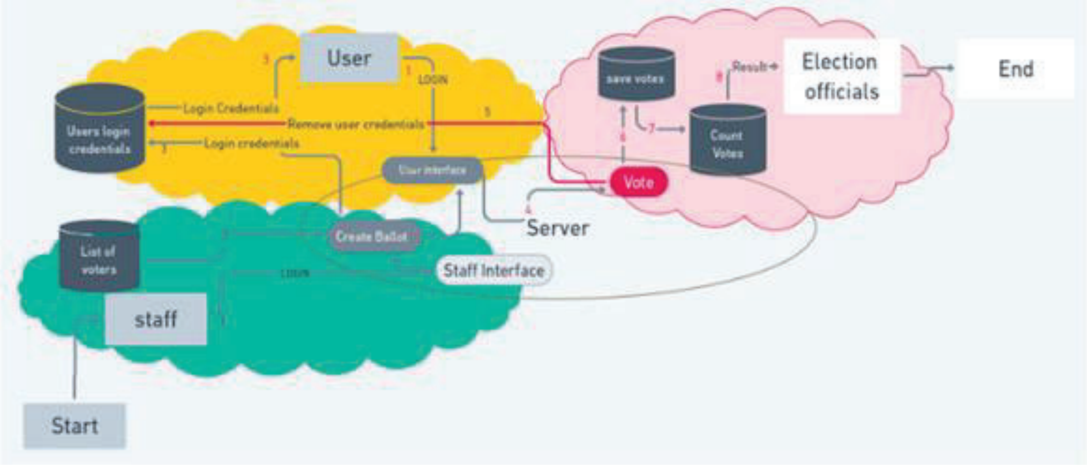


Figure 1. The structure of online election process.

Process 3 implements the *transmission of Login credentials* to voters. The challenge here is that e-voting systems must enhance the *security* of transmitting login credentials and mitigate the risks associated with unauthorized access and tampering.

Once authenticated, voters can cast their votes electronically (Fig. 1, step 4). The system should check the *eligibility* of the voter. After that the credentials must be removed from the ballot (step 5) ensuring the *unreusability* and voter *anonymity.*

In the process 6 the system records each vote maintaining *stability* and *privacy*. After the voting period ends, the system should tabulate the votes and generate results guaranteeing *completeness* and *integrity* (step 7). This process may involve aggregating individual votes, detecting any anomalies or discrepancies, and resolving any issues encountered during tabulation. The final stage 8 must record results *accurately.*

At last, to enhance trust in the e-voting system, *accountability* or independent verification mechanisms should be in place. This may involve allowing third-party organizations or experts to audit the system for vulnerabilities and verify the integrity of the voting process.

IV. OUR CONTRIBUTION

Our approach is that the use of various steganographic models can help to reduce the risks of data corruption and tampering. More specifically, the steganography models with active adversary [10] are very close to imitating attacks that can happen during elections.

Steganography, the technique of hiding secret information within non-secret information, has potential applications in e-voting systems, although its usage comes with significant considerations and challenges.

Ballot Concealment: Steganography can be used to embed encrypted ballot data within seemingly innocuous digital files, such as images or audio recordings. This could help protect the privacy of voters by concealing their choices within the noise of other data.

Steganography could be employed to embed audit trail information within e-voting records. This would enable the verification of the integrity and authenticity of the voting process without compromising voter anonymity.

Steganographic techniques could be used to embed digital signatures or cryptographic hashes within voting records. This would allow election officials and voters to verify the authenticity and integrity of the recorded votes.

Despite these potential applications, there are several challenges and considerations to using steganography in e-voting systems:

- Ensuring the security and integrity of steganographic techniques is challenging. Any vulnerabilities in the steganographic algorithms or implementation could be exploited by adversaries to manipulate or compromise the voting process.
- Detecting the presence of steganographically hidden information is difficult without access to the appropriate decoding keys or algorithms. This could hinder efforts to audit and verify the integrity of e-voting systems.
- Implementing steganography in e-voting systems without compromising usability is a significant challenge. The process of embedding and extracting hidden information must be seamless and intuitive for both election officials and voters.

Nowadays ensuring the accuracy of step 8, one of the most important issues, still remains unsolved. Secure data transfer from the storage, that keeps all the votes, to the counting server in modern systems rely on physical copies of data and are transferred manually, which is a major security issue. Thus, the usage of steganography methods is explored for that step. The main concern is that the channel between servers may be vulnerable to attacks and if an adversary can send unreliable data to the counting server, the entire electoral process will be in doubt.

In [11] most popular steganography methods were studied in terms of their viability when used for a secure electronic voting model. The statistical method was found to be most useful and practical. Unlike traditional steganography methods that rely on imperceptible changes in the cover data (like slight alterations in pixel values in images or LSB manipulation), statistical steganography operates by subtly modifying statistical properties of the cover data to embed the secret information. The goal of statistical steganography is to ensure that the statistical properties of the cover data remain similar to the original while embedding the secret message in a way that's difficult to detect statistically. This can involve techniques such as modifying the frequency distributions, correlations, or other statistical characteristics of the cover data to embed the secret message. Overall, statistical steganography aims to maintain the cover data's statistical fidelity while embedding secret information, making it a challenging task to detect without knowledge of the embedding process or access to the original, unaltered cover data.

By applying statistical steganography to the encryption of votes, the system ensures that the encrypted votes remain hidden within the data, making them undetectable to potential attackers. This helps in maintaining the confidentiality of the voting process. The verifier uses hypothesis testing to ensure the integrity of the voting process. It checks whether the votes are distributed randomly and not manipulated without using the extraction algorithm.

In [10] two-stage statistical hypothesis testing approach from the receivers point of view for the information-theoretic model of stegosystem with an active adversary is suggested. The functional dependence of reliabilities of the first and second kind of errors in both stages is constructed. The advantages of the two-stage approach are discussed.

We suggest to apply this research to the system of e-voting at the process 8. As a result of two-stage hypothesis testing the verifier will gain in time without loss in error probability. Moreover, using the functional dependence of reliabilities of the first and second kind of errors, the election officials can fix the level of important parameter and find the best possible values for others.

V. CONCLUSION

The proposed internet voting system architecture using statistical steganography and hypothesis testing provides a secure and private electronic voting system. The use of statistical steganography ensures that the encrypted vote is undetectable, while hypothesis testing ensures the integrity of the voting process. The system also ensures the privacy of the voters by using statistical steganography to encrypt and decrypt the votes. This system can be used for various types of elections, from small-scale local elections to national elections.

Future work: One of the challenges is to ensure biometric identification and maintain voter privacy. The next step of our research is implementation of privacy preserving face recognition to this system.

ACKNOWLEDGMENT

The work was supported by the Science Committee of RA, in the frames of the research 21T-1B151

REFERENCES

1. M. V. Vladucu, Z. Dong, J. Medina and R. Rojas-Cessa, "E-voting meets blockchain: A survey," in *IEEE Access*, vol. 11, pp. 23293-23308, 2023. doi: 10.1109/ACCESS.2023.3253682
2. B. Shahzad and J. Crowcroft, "Trustworthy electronic voting using adjusted blockchain technology," in *IEEE Access*, vol. 7, pp. 24477-24488, 2019. doi: 10.1109/ACCESS.2019.2895670
3. A. Benabdallah, A. Audras, L. Coudert, N. El Madhoun and M. Badra, "Analysis of blockchain solutions for E-voting: A systematic literature review," in *IEEE Access*, vol. 10, pp. 70746-70759, 2022. doi: 10.1109/ACCESS.2022.3187688
4. A Consensus Study Report of National Academies of Sciences, Engineering, and Medicine, "Securing the vote: protecting American democracy", Washington, DC: The National Academies Press, 2018. https://doi.org/10.17226/25120
5. S. Chaudhary *et al.*, "Blockchain-based secure voting mechanism underlying 5G network: A smart contract approach," in *IEEE Access*, vol. 11, pp. 76537-76550, 2023. doi: 10.1109/ACCESS.2023.3297492
6. G. Rathee, R. Iqbal, O. Waqar and A. K. Bashir, "On the design and implementation of a blockchain enabled E-voting application within IoT-oriented smart cities," in *IEEE Access*, vol. 9, pp. 34165-34176, 2021. doi: 10.1109/ACCESS.2021.3061411
7. M. Kim, "Toward round-efficient verifiable re-encryption Mix-Net," in *IEEE Access*, vol. 10, pp. 91397-91413, 2022. doi: 10.1109/ACCESS.2022.3202966
8. E. Yaakobi, M. Langberg and J. Bruck, "Information-theoretic study of voting systems," *2013 IEEE International Symposium on Information Theory*, Istanbul, Turkey, pp. 1087-1091, 2013. doi: 10.1109/ISIT.2013.6620394
9. B. Hosp and P. L. Vora, "An information-theoretic model of voting systems", Mathematical and Computer Modelling, vol. 48, pp. 1628–1645, 2008. https://doi.org/10.1016/j.mcm.2008.05.040
10. M. Haroutunian, P. Hakobyan and A. Avetisyan, "Two-stage optimal hypotheses testing for a model of stegosystem with an active adversary", JUCS - Journal of Universal Computer Science, vol. 29, no. 11, pp. 1254-1273, 2023. https://doi.org/10.3897/jucs.112913
11. A. Avetisyan, M. Haroutunian and P. Hakobyan, "Discussion on steganographic methods from the perspective of E-voting implementation", Proceedings of **14th** International Conference Computer science and information technologies CSIT 23, pp. 214-217, 2023. https://doi.org/10.51408/csit2023_51

On the Numerical Solution of the Characteristic Problem for one Quasi-Linear Equation

Giorgi Baghaturia
Muskhelishvili Institute of Computational
Mathematics of the Georgian Technical
University Tbilisi, Georgia
e-mail: gi.baghaturia@gtu.ge

Marine Menteshashvili
Muskhelishvili Institute of Computational
Mathematics of the Georgian Technical University
Sokhumi State University
Tbilisi, Georgia
e-mail: m.menteshashvili@gtu.ge

https://doi.org/10.62343/csit.2024.13

***Abstract* — For linear differential equations solving the Goursat problem means to find a solution to the equation by given values on arcs of characteristics of different families. These arcs have one common point and their tangents are different at this point. As for nonlinear equations, families of characteristic curves depend on the sought solution and therefore are unknown in advance. For that reason, it is impossible to pose the characteristic problem (analogue of Goursat problem) for a nonlinear equation in the same way we do it for a linear one. In this paper we present a numerical method to solve one class of quasi-linear equations whose one of characteristics is straight line. One of the families of characteristics is completely determined, while the other depends on the first derivatives of unknown solution and thus is not determined in advance. The type of equation is hyperbolic with possible parabolic degeneracy and this fact should be taken into account so that the problem is posed correctly. For finding a numerical solution of the problem we propose an algorithm which is based on the well-known method of characteristics.**

***Keywords* — Quasi-linear hyperbolic equation, Characteristic Problem, Grid-characteristic method.**

I. INTRODUCTION

As is known, for linear equations, Goursat problem consists in finding a solution to the equation by its values given on the arcs of characteristics of different families coming out from one point. For the numerical solution of such problems, an effective method is the grid-characteristic method. Nowadays, the grid-characteristic method is gaining popularity for linear hyperbolic systems of equations (see, for example, [1-2]). As for nonlinear equations, families of characteristic curves depend on the value of the unknown solution and therefore are not known in advance. For this reason, posing the Characteristic problem in the same way as is known for linear equations (see, for example, [3]) is impossible. We will consider the case when for a quasilinear equation it is still possible to formulate similar problem and at the same time, we will justify the numerical method.

In this paper, we consider a quasilinear equation, on the example of which the characteristic problem is correctly posed and we also introduce a numerical algorithm to solve the problem. Equations of this type were considered in papers [4–13, 15], in which initial, characteristic and initial-characteristic problems were studied.

In the plane of variables x, y, consider the following second order quasi-linear equation

$$Lu \equiv u_{xx} + (1 + u_x + u_y)u_{xy} + (u_x + u_y)u_{yy} = f(x, y, u_x, u_y). \qquad (1)$$

It should not be classified as strictly hyperbolic equation, since the corresponding characteristic form degenerates and this degeneration depends on behavior of solutions of the equation. In particular, when the sum of derivatives of solution $u_x + u_y$ equals to one, the equation has parabolic degeneracy. Therefore (1) is of mixed hyperbolic-parabolic type.

II. CHARACTERISTIC PROBLEM

In this section, we consider the equation

$$Lu = u_x + u_y - 1 \qquad (2)$$

The differential characteristic relations of equation (2) have the following form:

$$\begin{cases} dy = dx, \\ dp + (p + q)dq - (p + q - 1)dx = 0, \end{cases} \qquad (3)$$

$$\begin{cases} dy = (p + q)dx, \\ dp + dq - (p + q - 1)dx = 0, \end{cases} \qquad (4)$$

where $p = u_x,\ q = u_y$ are the Monge notations.

Having studied these systems, we come to the conclusion that one family of characteristic curves corresponding to system (3), is completely defined and is given by straight lines

$$y - x = const. \qquad (5)$$

Along this family, the characteristic invariant

$$(p + q - 1)e^{q-x} \qquad (6)$$

retains a constant value.

As for the second family of characteristics, it is defined as follows

$$(p + q - 1)e^{p+q-1-y} = const, \qquad (7)$$

along which the invariant

$$(p + q - 1)e^{-x} \qquad (8)$$

remains constant.

As the structure of invariant (8) shows, the family of characteristic curves corresponding to system (4) depends on the first derivatives of the unknown solution and thus it is not determined in advance. Since the characteristic family (6) is completely defined, the condition on this characteristic can be specified in the same way as in the case of linear equations. In particular, along the straight line $y = x + c$ the value of the solution is specified. We also have to check that this value does not cause parabolic degeneration of the equation at some point.

On the segment J_1 of the characteristic curve $F_1(x) = x$, consider the condition

$$u|_{J_1} = \varphi(x),\ x \in [0, a],\ \varphi \in C^2[0, a] \qquad (9)$$

At each point of the segment J_1 we can compute a derivative of the solution with respect to the characteristic direction. Formally we can write that

$$(u_x + u_y)|_{J_1} = u_x(x,x) + u_y(x,x) = \varphi'(x).$$

To avoid parabolic degeneracy, we require that

$$\varphi^{'}(x) \neq 1, \; x \in [0, \alpha]. \qquad (10)$$

The characteristic lines corresponding to differential relations (4) are completely determined, as they are given by equation (6). Due to the fact that the sum $p + q$ included in (6) can be calculated at each point J_1, we can draw a characteristic curve through any point of this segment that corresponds to system (4), Including the curve that passes through the end of J_1, and is defined explicitly:

$$y = (\varphi^{'}(0) - 1)(e^x - 1) + x \equiv F_2(x). \qquad (11)$$

As we see, from the given value of the solution on the segment J_1, the curve of family (4) is clearly determined. Consequently, it is possible to set the value of the solution on the arc J_2 of the characteristic curve (11):

$$u|_{J_2} = \psi(x), \; x \in [0, b], \qquad (12)$$

where $\psi \in C^2[0, b]$ is a given function. We should note, that characteristics J_1, J_2, constitute the support of the problem conditions.

Therefore, the problem can be formulated as follows:
Characteristic problem:

Find a regular solution to equation (2) along with its domain of definition, if characteristic conditions (9) and (12) are satisfied, when $\varphi(0) = \psi(0)$, $\varphi'(0) = \psi'(0)$, $\varphi''(0) = \psi''(0)$.

The following Theorem is true:

Theorem 1: If from the functional equation

$$\mathrm{y_0 = (\varphi'(x_1) - 1)(e^{x_0 - x_1} - 1) + x_0},$$

we can uniquely find the quantity $\mathrm{x_1}$ as a function of variables $\mathrm{x_0, y_0}$ there exists a unique regular solution of the Characteristic Problem (2), (9), (12), and this solution is defined in the Domain D, which is bounded by
$\mathrm{J_1,\; J_2,\; J_3\colon\; y = (\varphi'(a) - 1)(e^{x-a} - 1) + x}$,

$$\mathrm{J_4\colon\; y = (\varphi'(0) - 1)\left(e^b - 1\right) + x, \;\; x \in [b, a + b]}$$

characteristic curves.

Proof: We are able to find the values of the derivatives of the solution using the curves J_1 and J_2:

$$u_x|_{J_1} = \varphi^{'}(x) - log\frac{\psi^{'}(0) - 1}{\varphi^{'}(x) - 1} + \frac{\psi^{'}(0) - 1}{\varphi^{'}(0) - 1} - x + 1,$$

$$u_y|_{J_1} = log\frac{\psi^{'}(0) - 1}{\varphi^{'}(x) - 1} + \frac{\psi^{'}(0) - 1}{\varphi^{'}(0) - 1} + x - 1,$$

$$u_x|_{J_2} = (\varphi^{'}(0) - 1)e^x - \frac{\psi^{'}(x) - 1}{\varphi^{'}(0) - 1}e^{-x} + 2,$$

$$u_y|_{J_2} = \frac{\psi^{'}(x) - 1}{\varphi^{'}(0) - 1}e^{-x} - 1.$$

The domain of definition of the solution to the problem will be bounded by the characteristic curves emerging from the ends of the segment J_1 and the arc J_2. These curves and arcs of characteristics J_1, J_2, create the curvilinear quadrilateral D, which is completely covered by characteristic lines emanating from each point of the arc J_1 and J_2. For example, through the arbitrary point (x_1, x_1) of the segment J_1 passes a characteristic, equation of which has the following form:

$$J_5\colon\; y = (\varphi'(x_1) - 1)(e^{x - x_1} - 1) + x,$$

And through the arbitrary point (x_2, y_2) of the segment J_2 passes a characteristic curve given by the equation:

$$J_6\colon y = (\varphi'(0) - 1)(e^{x_2} - 1) + x.$$

For arbitrary chosen characteristics J_5, J_6 it's easy to determine both the point of intersection (x^*, y^*) and the value of the solution at this point. The coordinates x^*, y^* are determined as functions of x_1, x_2:

$$x^* = x_1 + log\left(1 + \frac{(\varphi'(0) - 1)(e^{x_2} - 1)}{\varphi'(x_1) - 1}\right),$$

$$y^* = x^* + (\varphi'(0) - 1)(e^{x_2} - 1)$$

Along the segment J_5 the following relation is true:

$$log(p + q - 1) - x = log(\varphi'(x_1) - 1) - x_1 \qquad (13)$$

Analogously, along the segment J_6 we have:

$$log(p + q - 1) + q - x =$$

$$=\log\left((\varphi'(0) - 1)e^{x_2}\right) + \tfrac{\psi'(x_2) - 1}{\varphi'(0) - 1}e^{-x_2} - 1 - x_2 \qquad (14)$$

Let us assume that the curves J_5 and J_6 intersect at point (x_0, y_0). In this case, from the relations (13), (14), derivatives of the solution at this point are determined as follows:

$$p(x_0, y_0) = (\varphi'(x_1) - 1)\Bigg(e^{x_0 - x_1} + 2 - log\frac{(\varphi'(0) - 1)e^{x_2}}{\varphi'(x_1) - 1} - \frac{\psi'(x_2) - 1}{\varphi'(0) - 1}e^{-x_2} - x_1 + x_2\Bigg)$$

$$q(x_0, y_0) = log\frac{(\varphi'(0) - 1)e^{x_2}}{\varphi'(x_1) - 1} + \frac{\psi'(x_2) - 1}{\varphi'(0) - 1}e^{-x_2} + x_1 - x_2 - 1.$$

So, it appears that we can find the value of the solution at this point.

All these considerations do not mean that we constructed the solution in an explicit way. In order to do so, we should have a possibility to find a value of our solution dependent on (x_0, y_0), at any point $(x_0, y_0) \in D$. Let's draw the characteristic curves of the both families at this point. Let's denote by J_7 a straight line passing through the point (x_0, y_0), corresponding to the system (3). By J_8 we denote the characteristic which corresponds to the system (4).

Let us also denote by (x_1, x_1) point of intersection of J_8 with the segment J_1. Thus, the equation of characteristic curve J_8 will be written as follows:

$$J_8\colon\; y = (\varphi'(x_1) - 1)(e^{x - x_1} - 1) + x$$

If we consider equations of J_2 and J_7 as a functional system, with respect of variables x, y, it is easy to determine the coordinates of their point of intersection:

$$x_2 = log\left(1 + \tfrac{y_0 - x_0}{\varphi'(0) - 1}\right),$$

$$y_2 = (\varphi'(0) - 1)(e^{x_2} - 1) + x_1.$$

We should find the coordinates of the point of intersection of characteristics J_8 and J_1. Since, the equation of the curve J_8 contains unknown quantity x_1, we have to act in a different way: According to our condition, J_8 goes through (x_0, y_0). So, we can write:

$$y_0 = (\varphi'(x_1) - 1)(e^{x_0 - x_1} - 1) + x_{0.}$$

And from this relation we should determine x_1 as a function of variables x_0, y_0. □

III. NUMERICAL METHOD

After these considerations, in order to solve the problem (2), (9), (12), numerically, we can formulate grid-characteristic method.

On the segment $[0, a]$ we introduce a grid ω_{h_1}:

$$\omega_{h_1} = \Big\{\widetilde{(x_0^j, y_0^j)}, \quad \widetilde{x_0^j} = h_1 j, \quad \widetilde{y_0^j} = F_1\left(\widetilde{x_0^j}\right),$$
$$\mathrm{j} = 1,2, \dots, n_1, \; h_1 = \frac{a}{n_1}, n_1 \in \mathrm{N} \Big\}.$$

On the segment $[0, b]$ we introduce a grid ω_{h_2}:

$$\omega_{h_2} = \{\widetilde{(x_\iota^0, y_\iota^0)}, \; \widetilde{x_\iota^0} = h_2 i, \; \widetilde{y_\iota^0} = F_2\big(\widetilde{x_\iota^0}\big),$$
$$\mathrm{i} = 1,2, \dots, n_2, \; h_2 = \tfrac{b}{n_2}, \; n_2 \in \mathrm{N}\}.$$

Points $\widetilde{(x_\iota^0, y_\iota^0)}$, $\mathrm{i} = 1,2, \dots, n_2$ are the first row of calculated points. The next row of points is obtained by the intersection of characteristics of the family (3) and (4) coming from the points of first row and from point $\widetilde{(x_0^j, y_0^j)}$ respectively, etc. If the j-th row of designed points $(\widetilde{x_\iota^j}, \widetilde{y_\iota^j})$, is determined, then the first approximation of the next row is determined by the formulas:

$$\frac{\tilde{y}_{i+1}^{j+1} - \tilde{y}_i^{j+1}}{\tilde{x}_{i+1}^{j+1} - \tilde{x}_i^{j+1}} = 1, \tag{15}$$

$$\frac{\tilde{y}_{i+1}^{j+1} - \tilde{y}_{i+1}^{j}}{\tilde{x}_{i+1}^{j+1} - \tilde{x}_{i+1}^{j}} = \tilde{p}_{i+1}^{j} + \tilde{q}_{i+1}^{j}, \tag{16}$$

where $\tilde{p}_{i+1}^{j}$, $\tilde{q}_{i+1}^{j}$ are the value of the functions p, q at the point $(\tilde{x}_{i+1}^{j}, \tilde{y}_{i+1}^{j})$.

After this, the values of the derivatives u_x, u_y and the solution u at the point $(\tilde{x}_{i+1}^{j+1}, \tilde{y}_{i+1}^{j+1})$, $i = 1,2, \dots, n_2 - 1$, $j = 1,2, \dots, n_1 - 1$ are calculated using the formulas:

$$\tilde{p}_{i+1}^{j+1} - \tilde{p}_i^{j+1} + \big(\tilde{p}_i^{j+1} + \tilde{q}_i^{j+1}\big)\big(\tilde{q}_{i+1}^{j+1} - \tilde{q}_i^{j+1}\big) +$$
$$+\big(1 - \tilde{p}_i^{j+1} - \tilde{q}_i^{j+1}\big)\big(\tilde{x}_{i+1}^{j+1} - \tilde{x}_i^{j+1}\big) = 0, \tag{17}$$

$$\tilde{p}_{i+1}^{j+1} - \tilde{p}_i^{j+1} + \tilde{q}_{i+1}^{j+1} - \tilde{q}_i^{j+1} -$$
$$-\big(\tilde{p}_{i+1}^{j} + \tilde{q}_{i+1}^{j} - 1\big)\big(\tilde{x}_{i+1}^{j+1} - \tilde{x}_i^{j+1}\big) = 0, \tag{18}$$

$$\tilde{u}_{i+1}^{j+1} - \frac{\tilde{u}_i^{j+1} + \tilde{u}_{i+1}^{j}}{2} =$$
$$= \tfrac{1}{2}\big(\tilde{p}_{i+1}^{j} + \tilde{p}_i^{j+1} + \tilde{q}_{i+1}^{j} + \tilde{q}_i^{j+1}\big)\big(\tilde{x}_{i+1}^{j+1} - \tilde{x}_{i+1}^{j}\big) \tag{19}$$

To clarify the calculation formulas, we use recalculation of the point $(\tilde{x}_{i+1}^{j+1}, \tilde{y}_{i+1}^{j+1})$ and of the values of $\tilde{p}_{i+1}^{j+1}, \tilde{q}_{i+1}^{j+1}, \tilde{u}_{i+1}^{j+1}$. These formulas have the following form for all $i = 1,2, \dots, n_2 - 1$, $j = 1,2, \dots, n_1 - 1$:

$$\frac{\tilde{y}_{i+1}^{j+1} - \tilde{y}_{i+1}^{j}}{\tilde{x}_{i+1}^{j+1} - \tilde{x}_{i+1}^{j}} = (\tilde{p}_{i+1}^{j+1} + \tilde{q}_{i+1}^{j+1} + \tilde{p}_{i+1}^{j} + \tilde{q}_{i+1}^{j})/2, \tag{20}$$

$$\tilde{p}_{i+1}^{j+1} - \tilde{p}_i^{j+1} + \frac{1}{2}(\tilde{p}_{i+1}^{j+1} + \tilde{q}_{i+1}^{j+1} + \tilde{p}_i^{j+1}$$
$$+ \tilde{q}_i^{j+1})\big(\tilde{q}_{i+1}^{j+1} - \tilde{q}_i^{j+1}\big) +$$
$$+\left(1 - \tfrac{1}{2}(\tilde{p}_{i+1}^{j+1} + \tilde{q}_{i+1}^{j+1} + \tilde{p}_i^{j+1} + \tilde{q}_i^{j+1})\right)\big(\tilde{x}_{i+1}^{j+1} -$$
$$\tilde{x}_i^{j+1}\big) = 0, \tag{21}$$

$$\tilde{p}_{i+1}^{j+1} - \tilde{p}_i^{j+1} + \tilde{q}_{i+1}^{j+1} - \tilde{q}_i^{j+1} - \tag{22}$$
$$-\left(\frac{\tilde{p}_{i+1}^{j+1} + \tilde{q}_{i+1}^{j+1} + \tilde{p}_{i+1}^{j} + \tilde{q}_{i+1}^{j}}{2} - 1\right)\big(\tilde{x}_{i+1}^{j+1} - \tilde{x}_i^{j+1}\big) = 0,$$

$$\tilde{u}_{i+1}^{j+1} - \frac{\tilde{u}_i^{j+1} + \tilde{u}_{i+1}^{j}}{2} =$$
$$= \tfrac{1}{2}\big(\tilde{p}_{i+1}^{j} + \tilde{p}_i^{j+1} + \tilde{q}_{i+1}^{j} + \tilde{q}_i^{j+1}\big)\big(\tilde{x}_{i+1}^{j+1} - \tilde{x}_{i+1}^{j}\big). \tag{23}$$

Theorem 2: Let $u \in C^4(D)$ and condition (10) be satisfied, then scheme (15-23) converges to the solution of the Characteristic problem (2), (9), (12) and the rate of convergence of the difference schemes is $O(h^2)$, where $h = max\,(h_1, h_2)$.

Proof. Let $u \in C^{3,3}(D)$, D be an area bounded by characteristics J_k, $k = 1, .., 4$. From the relations (15), (16), we obtain that

$$y_{i+1}^{j+1} - y_i^{j+1} - x_{i+1}^{j+1} - x_i^{j+1} = O\left(\left(h_1^{j+1}\right)^2\right) \equiv \psi_{1,i}^{j+1}, \tag{24}$$

$$y_{i+1}^{j+1} - y_{i+1}^{j} - (p_{i+1}^{j} + q_{i+1}^{j})(x_{i+1}^{j+1} - x_{i+1}^{j} = O\left(\left(h_2^{i+1}\right)^2\right) \equiv$$
$$\psi_{2,i+1}^{j}, \tag{25}$$

where

$$h_1^{j+1} = \max_i\left|x_{i+1}^{j+1} - x_i^{j+1}\right|, \; h_2^{i+1} = \max_i\left|x_{i+1}^{j+1} - x_{i+1}^{j}\right|,$$

$i = 1,2, \dots, n_2 - 1$, $j = 1,2, \dots, n_1 - 1$.

We introduce the following notations:

$$t_i^j = \tilde{y}_i^j - y_i^j, \; r_i^j = \tilde{x}_i^j - x_i^j, \; s_i^j = \tilde{p}_i^j - p_i^j,$$
$$g_i^j = \tilde{q}_i^j - q_i^j \;\; v_i^j = \tilde{u}_i^j - u_i^j.$$

Inserting the values of $\tilde{x}_i^j$, $\tilde{y}_i^j$ into (15), (16) and taking in account the equalities (24), (25), we obtain:

$$t_{i+1}^{j+1} - r_{i+1}^{j+1} = t_i^{j+1} - r_i^{j+1} - \psi_{1,i}^{j+1} = -\sum_{k=0}^{i} \psi_{1,k}^{j+1},$$

$$t_{i+1}^{j+1} - (p_{i+1}^{j} + q_{i+1}^{j})r_{i+1}^{j+1}$$
$$= t_{i+1}^{j} - (p_{i+1}^{j} + q_{i+1}^{j})r_{i+1}^{j}$$
$$+ (s_{i+1}^{j} + g_{i+1}^{j})(x_{i+1}^{j+1} - x_{i+1}^{j}) - \psi_{2,i+1}^{j} =$$
$$= -\sum_{k=0}^{j} \psi_{1,i+1}^{k+1}.$$

Consequently, if the condition (10) holds, from the last equations we conclude:

$$\left|r_{i+1}^{j+1}\right| \le c_1 h, \; \left|t_{i+1}^{j+1}\right| \le c_2 h, \; c_1, c_2 > 0$$

where

$h = \max\left(h_1^{j+1}, h_2^{i+1}\right), i = 1,2, \dots, n_2 - 1, \; j = 1,2, \dots, n_1 - 1.$

It is easy to obtain analogous estimations for s, g, v. And for the process of recalculation, we come to conclusion that if the condition (10) holds, then the following equalities are true:

$$\|r\|_C = O(h^2), \|t\|_C = O(h^2), \qquad \|s\|_C = O(h^2),$$
$$\|g\|_C = O(h^2), \; \|v\|_C = O(h^2). \; \square$$

ACKNOWLEDGMENT

The work was partially supported by the European Union’s HORIZON EUROPE Grant Project GAIN.

REFERENCES

1. Guseva, E.K., Golubev, V.I. & Petrov, I.B. Linear, Quasi-Monotonic and Hybrid Grid-Characteristic Schemes for Hyperbolic Equations. Lobachevski J Math 44, 296–312 (2023). https://doi.org/10.1134/S1995080223010146
2. Favorskaya, A.V., Petrov, I.B. (2018). Grid-Characteristic Method. In: Favorskaya, A., Petrov, I. (eds) Innovations in Wave Processes Modelling and Decision Making. Smart Innovation, Systems and Technologies, vol 90. Springer, Cham. https://doi.org/10.1007/978-3-319-76201-2_5
3. Berezin and N.P. Zhidkov, Computing Methods, part 2, Addison-Wesley; 1st edition, 1965
4. M. Menteshashvili. Study of the nonlinear characteristic problem using the Riemann invariants. Proceedings of A. Razmadze Mathematical Institute, v. 160, 2012, pp. 113-120.
5. G. Baghaturia, M. Menteshashvili. The general integral of a quasilinear equation and its application to solving a nonlinear characteristic problem. Siberian Mathematical Journal, 2019, Vol. 60, No. 6, pp. 940–951. New York, USA. Pleiades Publishing, Inc. DOI: 10.1134/S0037446619060028
6. M. Menteshashvili. The Nonlinear Cauchy Problem with Solutions Defined in Domains with Gaps. Journal of Mathematical Sciences, 2015, Volume 206, Issue 4, pp. 413-423, Springer.
7. G. Baghaturia, M. Menteshashvili. Application of General Integral of Quasi-linear Equation to Solving of Non-linear Cauchy Problem. Bulletin of TICMI, Vol. 27, No. 2, 2023, pp. 59–65, TSU Publ., Tbilisi, ISSN 1512-0082.
8. J. Gvazava, G. Baghaturia, M. Menteshashvili. Cauchy Problem for a Quasi-Linear Hyperbolic Equation with Closed Support of Data. Journal of Mathematical Sciences, 2013, Volume 193, Issue 3, pp. 364-368, Springer.
9. R. Bitsadze and M. Menteshashvili. On one nonlinear variant of the nonlocal characteristic problem. Journal of Mathematical Sciences, 2015, Volume 206, Issue 4, pp 406-412, Springer.
10. G. Baghaturia. Some non-linear versions of hyperbolic problems for one quasi-linear equation of mixed type. Journal of Mathematical Sciences (Springer), 2015, Volume 208, Issue 6, pp 621-634.
11. M. Z. Menteshashvili, On the Cauchy problem with initial data on the unit circle. (Russian) Soobsh. Akad. Nauk Gruzii 148 (1993), No. 2, 190–192.
12. J. K. Gvazava, Nonlocal and initial problems for quasi-linear, non-strictly hyperbolic equations with general solutions represented by superposition of arbitrary functions. Georgian Math. J. 10 (2003), No. 4, 687–707.
13. 6. J. K. Gvazava, The mean value property for non-strictly hyperbolic second order quasi-linear equations and the nonlocal problems. Proc. A. Razmadze Math. Inst. 135 (2004), 79–96.
14. E. Goursat, Lecons sur l'integration des equations aux derivees partielles du second ordre. Hermann, Paris 2, 1898.
15. V. Kvaratskhelia and M. Menteshashvili, Some Nonlinear Version of a Nonlocal Problem and Its Discrete Analogy, 2019 Computer Science and Information Technologies (CSIT), Yerevan, Armenia, 2019, pp. 77-78, doi: 10.1109/CSITechnol.2019.8894943.

Overview of "Sufficient" Forecasting Models

Merab Pkhovelishvili
Faculty of Informatics and Control Systems
Georgian Technical University
Tbilisi, Georgia
e-mail: m.pkhovelishvili@gtu.ge

Natela Archvadze
Faculty of Exact and Natural Sciences, Ivane Javakhishvili Tbilisi State University,
Tbilisi, Georgia
e-mail: natela.archvadze@tsu.ge

Zaza Tsiramua
Faculty of Informatics and Control Systems
Georgian Technical University
Tbilisi, Georgia
e-mail: zaza@gtu.ge

Mariam Pkhovelishvili
GTS Group - Director
Tbilisi, Georgia
e-mail: pkhovelishvilimariami@gmail.com

https://doi.org/10.62343/csit.2024.14

***Abstract* — A new hybrid forecasting model based on the use of parallel data is considered, which allows you to select a pair of forecasting models (or a combination of three, four, etc. models), the "joint" forecast probability of which gives a much better result than each model separately. We eliminated models that were not "Necessary" from consideration and used a "sufficient" number of models together, thereby further increasing the accuracy of the forecast. "Necessary" predictive models are models whose set of predictions always includes events that occur, while "sufficient" predictive models are models whose predictions always come true.Keywords—Computer science, informatics, information technology.**

***Keywords* — Forecasting models, hybrid model, decision making**

I. INTRODUCTION

When modeling any processes, when it is necessary to test hypotheses about asynchronous processes, forecasting models are often used, which makes it possible to analyze their effectiveness. Based on existing models, it is possible to create a new hybrid model [1], the purpose of which is to increase the accuracy of the forecast. This increases the relevance of the topic under discussion.

One popular forecasting method is the Bayesian method, developed by many scientists (Student, Fisher, Neumann, and Pearson, as well as the conditional Bayesian method) and is mainly used to prove asynchronous hypotheses. Our goal is to develop a new method based on the use of parallel data, thereby reducing the number of models used in the Bayesian method and, consequently, the computation time.

II. CLASSIFICATION OF FORECASTING MODELS AND HYBRID MODEL

The hybrid forecasting model we created [1, 2] using a parallel data algorithm [3] is effective in various economic and financial problems, especially in those tasks for which there are several forecasting models. Each of these models has its own advantages, but none of these models provide the ability to determine predictive value with the smallest possible error.

Concurrent data (or data sets) are different types of data that influence (or predict) the same event. For example, parallel data during a particular disaster is a collection of the following data: geological, meteorological, and hydrological parameters, historical and current data, including interpreted information from satellite, radar, and aerial photographs, as well as data obtained from hydrometeorological/geological field surveys.

The likelihood of each pattern occurring may be small, but under certain conditions, when a coincidence occurs in time and place, this is already a sign of tragedy.

We consider the intersection of sets of forecasts given by two, three, and usually n forecasting models. The superiority of selecting pairs of models over any single best forecasting model is theoretically justified. The advantage of the model obtained from the ratio of predictions of the n number of models compared to the n-1 number model [4] is shown.

The best pair of models is a pair that has no or very little overlap in predictions.

The time for recalculation of pairs of forecasting models is set - after each past event, and also to locally take into account "approximate coincidences" of forecasts when building a new model.

When constructing a hybrid model, existing forecasting models were analyzed and two types of models were identified: "necessary" forecasting models and "sufficient" forecasting models. Their characteristics are given in the article [5, 6]. "Necessary" predictive models are those whose set of predictions always includes events that occur, while "sufficient" predictive models are those whose predictions always come true.

Suppose there are $\mathrm{A}_1, A_2, \dots, A_\mathrm{n}$ models for predicting a certain event, where n is the number of models being considered. Let's assume that each model is "necessary", that is, it always predicts an event that will occur. These models do not take into account models that are "not needed" because they cannot predict the event.

In practice, there are very few "sufficient" models, and there is no single model that predicts all events, so the question is whether we can use these "sufficient" models in such a way that a combination of them can predict all events. that is, a set of "sufficient" models will simultaneously become "necessary".

III. "SUFFICIENT" PREDICTIVE MODELS

A model in which all the predictions are correct, as we have already said, is "sufficient", but there are events that were not predicted, although various combinations of these models lead to a model close to the required one. For example, consider a joint forecast of two models using a hybrid model (Fig. 1). Let the rectangle denote all events, and the circles denote 2 "sufficient" models that are built according to P_i and P_j on the antecedents and assume that they have a non-empty intersection P_{ij} (green in the figure), then combining them results in a set of event predictions. The joint forecast is outlined in the figure with a black line. In the figure, we see the regions that were not predicted by the joint prediction of these two "sufficient" models (purple rectangle in the figure).

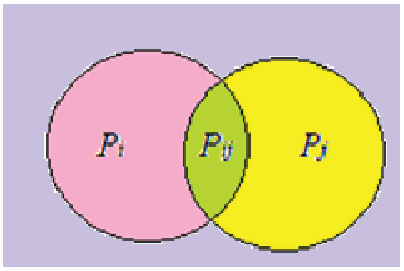


Figure 1. Joint prediction of two "sufficient" models.

Similarly, if we increase the number of models and consider three models (Fig. 2), we will see that the forecast when three models work together is larger, and closer to all combined forecasts (the purple rectangle is smaller than with two models).

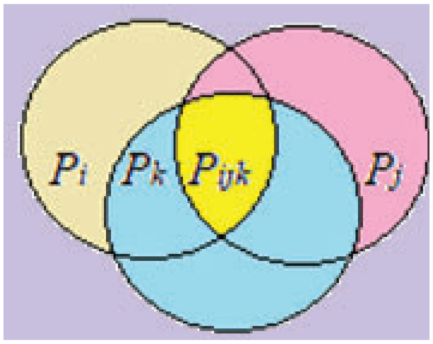


Figure 2. Joint prediction of three "sufficient" models

When using multiple "sufficient" models together, it is necessary to study which predictions were repeated in the intersection of their predictions (what happened in their intersection was repeated). This means that there are predictions that some or most models think will happen. Of the common parts, one part remains, and the other is not taken into account. Particular attention should be paid to the fact that if many models have the same predictions, then a given event will occur, but the probability of this prediction will need to be determined.

It is necessary to study the predecessor who repeats all the predictions. For example, in the case of an earthquake, several methods rely on the methods of light from a celestial body. For one model this was a prerequisite, for the second, third, fourth... This means that the brightness of the celestial body is one of the main prerequisites.

It's also important to consider how often these models make predictions. The model M_i predicted 10 times, Mj model did this 20 times, and they did everything correctly because there are "sufficient" models. But M_k model could only make a prediction once, and it was correct. We need to calculate the frequency of predictions of these models. Such a "sufficient" model, which produced only one prediction, does not need to be included in the discussion. It's good when the model made a forecast many times and it came true just as many times. Therefore, for "sufficient" models, the frequency S must be defined in % as the ratio of the number of predictions issued (n) to the number of predictions issued (m). The formula will look like:

$$S = \frac{n}{m}100\%$$

For example, if M_1 model predicted 20 times, M_2 model has 10 times and a total of 40 earthquakes, then the frequencies of these models will be:

$$S_1 = \frac{20}{40}100\% = 50\%\ ,\ S_2 = \frac{10}{40}100\% = 25\%$$

Let us consider the Bayesian method [7] for calculating a single forecast of "sufficient" models. Let's carry out calculations similar to those used in the Bayesian method - apply the same formula to the models and calculate the frequency of combining these two frequencies S_1=50% and S_2=25% of the frequency (Fig. 3):

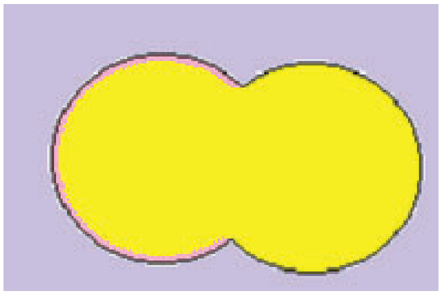


Figure 3. Calculation of the frequency of joint prediction of two "sufficient" models

where M_1 and $M_{2.}$ A combination of models is mentioned:

$$P(M_i \cup M_j) = \frac{k}{m}\ 100\ \%\ ,$$

where $k = n_i + n_J - n_p$, where n_i - this is a number of model predictions M_i , n_J —this is number of model predictions M_j, and - n_p is the number of intercept terms $M_i \cap M_j$ (we assume is 5). Let's calculate the joint prediction frequency for our example:

$$S_k = \frac{20+10-5}{40}100\% = \frac{25}{40}100\% = 62.5\%$$

We found that the prediction rates of individual models (50% and 25%) increased, which was expected since we used a hybrid forecasting model.

CONCLUSION

Based on the analysis of existing forecasting models, we divided the models into "necessary" and "sufficient", built a hybrid forecasting model with parallel data from pairs and triplets of models, excluded "unnecessary" models from consideration, and used "sufficient" models together, which further increases the accuracy of forecasting.

REFERENCES

1. Aliyev, V., Pkhovelishvili, M., Archvadze, N., Gasitashvili, Z., Tsiramua, Z. “Hybrid Prediction Method Using Experts and Models”. 2023 5th International Conference on Problems of Cybernetics and Informatics, PCI 2023, 2023.
2. Gasitashvili, Z., Pkhovelishvili, M., Archvadze, N. Prediction of events means of data parallelism. Proceedings - Mathematics and Computers in Science and Engineering, MACISE 2019, 2019, pp. 32–35, 8944725. https://ieeexplore.ieee.org/abstract/document/8944725
3. Pkhovelishvili, M.G., Archvadze, N.N., Nikoleishvili, M.M. “Analysis of earthquake prediction models to obtain the best model”. Geologiya i Geofizika Yuga Rossii, 2023, 13(1), pp. 162-172. DOI: 10.46698/VNC.2023.61.18.012
4. Prangishvili, A., Gasitashvili, Z., Pkhovelishvili, M., Archvadze, N. “Predicting Events by Analyzing the Results of the Work of Predictive Models”. Communications in Computer and Information Science, 2022, 1562 CCIS, pp. 64–78.
5. Prangishvili, A., Gasitashvili, Z., Pkhovelishvili, M., Archvadze, N. “Theory of universal approach to improve predictive models using parallel data and application examples.” IP Conference, 2023, 2757, 020004.
6. Aliyev, V., Gasitashvili, Z., Pkhovelishvili, M., Archvadze, N. Algorithm for Building a Hybrid Model of the Existing Risk model. Reliability: Theory and Applications, 2022, 17, pp. 106–110.
7. Stuart, A.; Ord, K. (1994), Kendall's Advanced Theory of Statistics: Volume I – Distribution Theory, Edward Arnold, §8.7

Development of examination software for the Faculty of Medicine

Givi Qiria
The University of Georgia
Tbilisi, Georgia
e-mail: qiriagivi@gmail.com

Sulkhan Sulkhanishvili
The University of Georgia
Tbilisi, Georgia
e-mail: s.sulkhanishvili@ug.edu.ge

Nikoloz Katsitadze
The University of Georgia
Tbilisi, Georgia
e-mail: n.katsitadze@ug.edu.ge

https://doi.org/10.62343/csit.2024.15

***Abstract* — the article discusses in detail several problematic issues that occur in the Faculty of Medicine and their examination since they need a different, specific and properly equipped environment from all other faculty students to pass the exams. Because of that, a methodology adapted to students and appropriate software is needed. The article discusses in detail the ways to eliminate the mentioned problems and talks about the solutions that we offer to the Faculty of Medicine. The article also focuses on the methods of using software and hardware, their advantages and the effect that a new and well-functioning system will have on the students of the Faculty of Medicine. The article also shows the examples of the examination systems of medical universities from foreign countries, the similar approaches that have been tested in the world and that we are going to implement To make the exams as student-oriented as possible.**

Keywords — informatics, software development, examination software, software for the Faculty of Medicine.

I. INTRODUCTION

Medicine is one of the most specific fields where making a mistake is equal to death, therefore it is necessary to provide students with the appropriate knowledge, skills, and experience. In addition, it is important to create an examination space that is properly adapted to them, which will be focused on the fact that students can demonstrate their knowledge as much as possible.

Exams for medical students differ from standard exams for students of other faculties because they use the principle of "OSCE" (Objective Structured Clinical Examination). This means that students have the opportunity to demonstrate their knowledge and skills in specially designed simulation rooms, where the environment is as close as possible to diagnostic cabinets. To put it simply, there are twelve rooms located next to each other, divided in the middle by a wall and transparent glass on one side. The first side of the room belongs to the examiner and the second to the student. In each room, students perform certain types of medical tasks within a limited time frame, after the time is up, they are required to move to another room to complete the next task. These types of exams are important in that medical situations are as close as possible to real cases and students have the opportunity to use their skills and theoretical knowledge in practice to be able to save the lives of patients. Using the „OSCE“ principle during exams is a very important step forward for medical students. However, without its specific automation, it requires a lot of human resources and time. Because of that, we had a dialogue with various universities, where we talked about the challenges and problems of the „OSCE“ principle.

At this meeting, the University of Georgia expressed its desire to participate and update the examination system of its medical school. UG (University of Georgia) also expressed its willingness to share with us all the problematic issues related to this. With joint efforts, we would be able to improve the system to the maximum and adapt it to the students.

II. ABOUT PROBLEM

The problems were discussed with the University of Georgia. Following flows were seen: the examination rooms were not equipped with the appropriate software and inventory, which subsequently created several problematic issues.

In particular, the examiner had to share the image with one laptop to twenty-four monitors via an HDMI HUB, which meant that the same information was displayed on all monitors. This created a big problem because it was not clear to the students which room they should have entered after finishing the exam in one exam room to continue the exam. Using a consumer, non-specific laptop also created a high risk of the system getting stuck, having a bug that would cause the exam to fail. In the case when the observer wanted to see some kind of information on the laptop, the students would also be able to see it on the monitors. Using a consumer, non-specific laptop also created a high risk of the system getting stuck, having a bug that would cause the subsequent test to fail. In the case when the observer wanted to see some kind of information on the laptop, the students would also see it on the monitors. One of the problems was the lack of exam software, which was reflected in the following: examiners had to prepare Excel files before each exam and save them on each exam room's computer, which would outline the assessment system and exam questions. Each Excel file should have had as many Sheets as students are taking the exam. Creating the said Excel file from scratch and filling in the information before each exam was quite tiring and uncomfortable. Moreover, if something unexpected happened and the Excel file was not saved, all the information would be deleted. Guest lecturers who observed the exam had to manually write the badge numbers of the students who entered the exam room in their Excel file, which increased the probability of making a mechanical error since the examiner could write down the student's badge number incorrectly, which would subsequently cause the student to be left without a score. The examiners entered the scores into an Excel file, which required extra work from the examiners as they had to extract the Excel file from each computer separately, view the grades, and then display it on the student portal and compile tables to see the students' average scores.

Similar technologies - the "OSCE" principle is used by 57 countries worldwide, including America, UK Britain, France and Germany[1]. The mentioned principle is similar in all countries, although the method of conducting is different.

Imam Abdulrahman Bin Faisal University uses “QuestionPro” software. This means that the traditional paper-based stations were converted into an online electronic version. Answers were filled in using smart tablets (iPads). QR codes were used for students' identification at each station to fully digitize the process and save time. To create and distribute surveys, they use “QuestionPro”, which is web-based software. It consists of tools for distributing surveys via email or website, an interface for creating survey questions, and tools for analyzing and viewing the results[2].

In UK Britain, the University of Northampton University of Northampton uses an electronic examination system for medical students, which means that a timing system and cameras[3] are installed in each simulated examination room.

III. SOLUTION OF THE PROBLEM

To make the "OSCE" principle more practical and convenient, both hardware and software are needed. A web application was created for which the following technologies and hardware were used.

Software:

- React Js[4] – web frontend;
- Spring Boot[5] – web backend;
- Android SDK and Kotlin[6] – for Android applications;
- Redis[7] – for cashing and saving sessions;
- PostgreSql[8] – for collecting data;
- Docker[9] – for virtualization.

hardware provision:

- Server PC - to run the system;
- 27-inch monitor - so that students can control the time on the monitor;
- Android box - android 8.0 + - so that the exam supervisors can display the exam time in live mode;
- Poe Tablet - android 8.0+ - so that students can register their badges. Also, the mentioned tablet allows you to control the exam time from outside without entering the exam room. PoE will be used for simple infrastructure, in particular, it will no longer be necessary to use an additional power cable;
- Poe switch lyer_3 - to supply the tablet with both electricity and network. Layer 3 will be responsible for the internal network even if the router goes down.

Through the web application, exam administrators can control who has access to the exam software and assign appropriate rights and roles. They also can compile exam questions for students and track the progress of the exam time. Exam supervisors can view students' grades, and averages in the form of charts and tables through the web application, which they can then transfer to an Excel file. Using the application, it is possible to create a reminder in the calendar, which allows us to remind relevant persons of specific information before a certain time. As soon as the student enters the exam room, the examiner will automatically see the student's code, picture, exam questions, and grading system. In the case when the examiner cannot evaluate the student on time, he is not limited in time and can write the appropriate score and comment to the student later. The application mentioned above can be used from any device that has browser support.

IV. CONCLUSION

In conclusion, we can say that it is important for the students of the Faculty of Medicine and the supervisors of the exam to have a well-organized exam system, which will be focused on the quality, efficiency and proper conduct of the exam. The advantages of our application are as follows: The architectural infrastructure is more accurate, usable and result-oriented than the old one; The time management system has become more efficient since it became possible to manage time from any device that supports a browser; It is no longer necessary to work in separate Excel files since the performance of the mentioned function (compilation of exam questions, a reflection of marks, analysis of obtained marks, entering of student badge numbers) was added to the software provision and thus we achieved time-saving, reduction of staff labor, more accuracy and in a short time as much as possible Conducting exams for more students than before. Based on all of the above, we can say that our application is a very flexible and effective tool that will simplify the work process for both the exam supervisors and the students so that they can realize their knowledge as much as possible during the exam.

REFERENCES

1. The Osce website. [Online]. Available: https://www.osce.org/participating-states?fbclid=IwAR1yB4pFF0PtMWEieYV_YbXHK3SEZU82YB7VI1tnvHAbfuggQvXvqyfmZos.
2. (2022) The Dovepress website. [Online]. Available: https://www.dovepress.com/innovative-method-to-digitize-a-web-based-osce-evaluation-system-for-m-peer-reviewed-fulltext-article-IJGM#cit0021.
3. (2017) University of Northampton – Preparing for your OSCE examination Part 2 NMC – Test of Competence. [Online]. Available: https://www.northampton.ac.uk/wp-content/uploads/2015/10/Preparing-for-your-OSCE-examination-June2017-v3.pdf.
4. The Reactjs website. [Online]. Available: https://legacy.reactjs.org/docs/getting-started.html#learn-react.
5. The Spring website. [Online]. Available: https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/.
6. The kotlinlang website. [Online]. Available: https://kotlinlang.org/docs/android-overview.html.
7. The Redis website. [Online]. Available: https://redis.io/docs/get-started/data-store/.
8. The Postgresql website. [online]. Available: https://www.postgresql.org/docs/.
9. The Docker-docs websire. [Online]. Available: https://docker-docs.uclv.cu/develop/.

Scoring the Outcomes of 1-3-3 Multistage Model of Computerized Adaptive Testing

Simon Razmadze
Georgian Computer Society
Tbilisi, Georgia
e-mail: simonrazmadze@gmail.com

https://doi.org/10.62343/csit.2024.16

***Abstract* — The paper presented considers the ordering method of outcome set for multi-stage testing (MST) of 1-3-3 model. The ordering method of outcome set is used for the estimation of results of computerized adaptive testing (CAT). This method is not tied to a specific testing procedure. Acknowledgment of this is its usage for the 1-3-3 model, which is described in the paper. To sort the set of testing outcomes the function-criteria described in the initial article are used here and a comparative analysis of obtained results is performed. The ordered outcome set is estimated by a hundred-point system according to the normal distribution.**

***Keywords* — Computerized adaptive testing, Stradaptive testing, Multistage adaptive testing, Evaluation algorithm, Ordering of a set.**

I. INTRODUCTION

Computerized adaptive testing implies the test adaptation to the level of knowledge of the test. During the testing process the system analyzes the answers and uses them to choose each following question based on the best correspondence to the level of examinee so that the questions gradually become complicated for a well-prepared examinee and simpler for a poorly prepared person. The process of test adaptation for an individual user is mentioned.

This means that the tests must be precalibrated according to their level of difficulty.

The modern Computerized Adaptive Testing (CAT) is based on Item Response Theory (IRT). IRT is a family of mathematical models that describe how people interact with test items [1]. According to this theory test items are described by their characteristics of difficulty and discrimination. Discrimination is independent of difficulty and shows how the probability of a positive response is distributed between different levels of examination. In addition, they can have a so-called "pseudo-guessing" parameter that reflects the probability that an examinee with a very low trait level will correctly answer an item solely by guessing [2].

We will try to create a test assessment system that makes it easy for the test creator to use a computer-adaptive method for creating one's own test. For this purpose, let us not discuss IRT but another traditional approach to testing - Stradaptive Testing. The term "Stradaptive" is derived from the "Stratified Adaptive", and it belongs to D.J. Weiss [3, 4].

To express the ordering method of outcomes set, a specific procedure for testing is used in Razmadze et al.'s article [5]. This procedure has an illustrative purpose for the evaluation method. The method described can be used for other similar strategies as well as for multistage testing, one of the models considered in this paper. Similar models were discussed in the articles [6] and [7].

Thus, the paper presented is devoted to the realization of an ordering method of the outcome set, in particular on the example of a three-stage 1-3-3 model.

II. ORDERING METHOD OF OUTCOME SET

The initial article Razmadze et al. [5] considers an original method of CAT result estimation for multistage testing strategy.

The method considers all possible variants of results, which is named an outcome set. The outcome set represents a non-typical unity of different dimensional elements. At Razmadze et al. [5] article comparison criteria for these elements are defined, and principles of ordering of the set are described. The article shows how to receive the final score after ordering the outcome set. The ordered criteria of outcomes set may not be singular; this is confirmed by a comparative review of two examples presented in this work.

Thus, the paper presented is devoted to the realization of an ordering method of the outcome set, in particular on the example of a three-stage 1-3-3 model.

III. THE THREE-STAGE 1-3-3 ADAPTIVE MODEL

A. The scheme of 1-3-3 model

Now let us consider the usage of the ordering of testing result scores in case of multistage adaptive testing. For this purpose, we will discuss the three-stage 1-3-3 model, which is presented in the following scheme [8]:

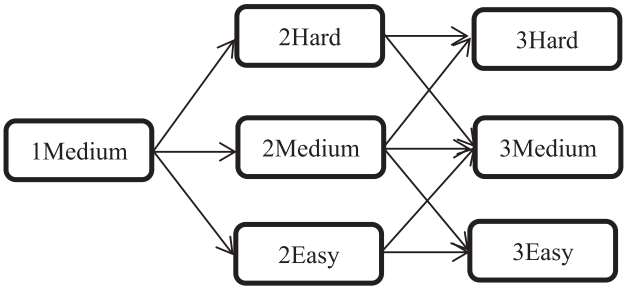


Figure 1. The 1-3-3 MST model

The number indicated in the rectangle of the module corresponds to the stage; the letters correspond to comparative difficulty (Hard; Medium; Easy). Let us number the medium difficulties of modules. Each of these numbers can be considered as the weight of corresponding module item:

Table 1. *The item weights of the three-stage 1-3-3 model*

#	1	2	3	4	5	6	7
Difficulty	3Easy	2Easy	1Medium	2Medium	3Medium	2Hard	3Hard
Weight	1	2	3	3	3	4	5

In the first row of Table 1, all modules are numbered from 1 to 7. We will be using the given numbering for defining the test outcome. Taking into account the complexity levels of the modules, the outcome is expressed as a seven-dimensional vector: $n = \{ c_1, c_2, c_3, c_4, c_5, c_6, c_7 \}$, where c_i represents the number of correct answers of i module, $i = 1, 2, \dots, 7$. Due to the fact each testee performs only one item on each stage, there can be only 3 components out of a given 7 that are different from 0 in each test outcome. In addition, each c_i component, $i = 1, 2, \dots, 7$, has a weight, predefined according to Table 1.

For practical reasons, let's assume that the number of questions to be given to the examinee in each module is equal to five. Since only three components of the vector $n = \{ c_1, c_2, c_3, c_4, c_5, c_6, c_7 \}$ can take the whole value from 0 to 5 inclusive (6 options in total), and the other four components are always zero, the total number of test outcomes will be N = 6^3 = 216.

The displayed classification can be considered as an analogy to the one used in the item response theory (IRT) (-3; 3) range, where the examinees' abilities are measured [2]. But in this case instead of (-3; 3) range we use the weights provided in Table 1. This does not distort the achievement of the initial task. By considering the weights, the scheme from Figure 1 will transform into the following:

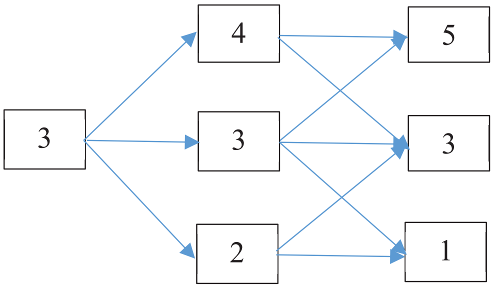


Figure 2. The 1-3-3 MST model with weights

B. *Outcome of 1-3-3 model*

In Razmadze et al. ([5], p. 1656) [article the outcome was defined as a vector drawn from the corresponding numbers of the levels of items obtained during the testing process. In this case, by definition, the outcome vector consists of the components that correspond to the number of correct answers in each module. This is more convenient for using the set ordering method for multistage adaptive tests.

C. *Outcome route*

Modules of the first and the second stage have classification cut-points that define the route of the testing outcome, in other words, choosing the second and third stage modules. Classification cut-point is the number of correct answers within the module that defines the branching – next stage module. Despite where the classification cut-points are chosen the total amount of the testing outcomes is constant and N = 216.

An example discussed in this article on the first stage of 1Medium module has two cut-points: 2 and 4. This means, that in case of less than 2 correct answers (0 or 1) an examinee will be given the easier 2Easy module of the second stage, in case 2 or 3 correct answers - 2Medium module and in case of 4 or 5 correct answers - the more difficult 2Hard module of the second stage.

The second stage modules have the following classification cut-points:

- The classification cut-point of the module 2Easy is 3. If the number of correct answers is less than 3 (0, 1 or 2), the examinee is given the easiest module 3Easy, and if the number of correct answers is 3 or more (3, 4 or 5) – the third stage 3Medium module;
- 2Medium module has two classification cut-points: 2 and 4. This means, that if the number of correct answers is less than 2 (0 or 1), the examinee will be given the easiest module of the third stage 3Easy, if the number of correct answers is 2 or 3 - 3Medium module of the third stage, and if the number of correct answers is 4 or 5 - 3Hard module of the third stage;
- 2Hard module has one cut-point: 3. If the number of correct answers is less than 3 (0, 1 or 2), the examinee is given the 3Medium module of the third stage, and if the number of correct answers is 3, 4 or 5 - the 3Hard module of the third stage.

IV. THE SET ORDERING METHOD FOR SCORING THE OUTCOMES OF 1-3-3 MODEL

A. *Ordering according to the S(n) criterion*

Let us discuss the first criteria from the initial article Razmadze et al. ([5], p. 1658, Formula (4)):

$$S(n) = \frac{R}{1+M}, \quad n \in N \quad (1),$$

where R is a weighted sum of scores of correct answers and M is a weighted sum of scores of incorrect answers.

The corresponding formulas for calculating R and M are given in the article Razmadze et al. ([5], p. 1657, Formulas (1) and (2)). Based on these formulas, in the case of the 1-3-3 MST model, we will obtain the following:

$$R = 3c_1 + 2c_2 + 3c_3 + 4c_4 + c_5 + 3c_6 + 5c_7,$$

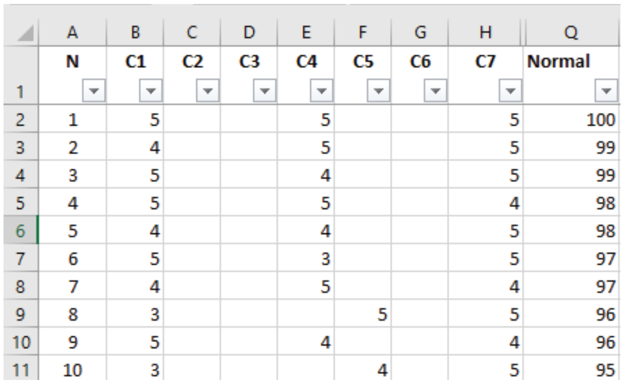
$$M = 3d_1 + 2d_2 + 3d_3 + 4d_4 + d_5 + 3d_6 + 5d_7,$$

where c_i is the number of correct answers in i module, and d_i is a number of mistakes in i module, $i = \overline{1,7}$.

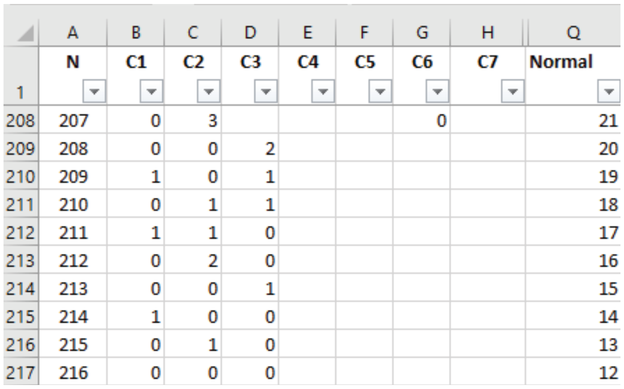
The Formula (1), which should be used for outcome estimation, now used in seven-module case. The structure of outcome set of the three-stage model discussed in this article is different from the one discussed in the initial article by Razmadze et al ([5], p. 1656). This means that the domain of a function S(n) is different. Despite this, S(n) function will provide complete ordering of set N in a given case too.

The result is provided in Table 2, where $c_1,\ c_2,\ c_3,\ c_4,\ c_5,\ c_6,\ c_7$ values are given in the columns B, C, D, E, F, G, H, respectively. The values calculated using Formula (1) are shown in column M. The data is sorted according to M column decreasing order. The table shows the first 10 (left half) and last 10 (right half) testing outcomes' estimation results.

Table 2. *1-3-3 model's outcome estimation by S(n) criterion*



	A	B	C	D	E	F	G	H	Q
1	N	C1	C2	C3	C4	C5	C6	C7	Normal
2	1	5			5			5	100
3	2	4			5			5	99
4	3	5			4			5	99
5	4	5			5			4	98
6	5	4			4			5	98
7	6	5			3			5	97
8	7	4			5			4	97
9	8	3				5		5	96
10	9	5			4			4	96
11	10	3				4		5	95



	A	B	C	D	E	F	G	H	Q
1	N	C1	C2	C3	C4	C5	C6	C7	Normal
208	207	0	3				0		21
209	208	0	0	2					20
210	209	1	0	1					19
211	210	0	1	1					18
212	211	1	1	0					17
213	212	0	2	0					16
214	213	0	0	1					15
215	214	1	0	0					14
216	215	0	1	0					13
217	216	0	0	0					12

B. *Ordering according to the $\boldsymbol{F(n)}$ Criterion*

Let us discuss the second criterion from the initial article Razmadze et al. ([5], p. 1658, Formula (9)):

$$F(n) = R * \frac{A}{\mu}\ ,\quad n \in N, \qquad (2)$$

where R is a weighted sum of scores of correct answers, A is an average complexity of incorrect answers and μ – number of mistakes.

The corresponding formulas for calculating R and A are given in the initial article by Razmadze et al. ([5], p. 1657, Formulas (1) and (3)). Based on these formulas, in the case of the 1-3-3 MST model, we will obtain the following:

$$R = 3c_1 + 2c_2 + 3c_3 + 4c_4 + c_5 + 3c_6 + 5c_7,$$

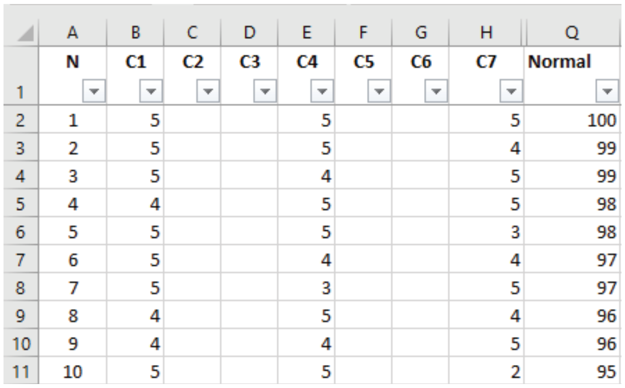
$$A = \frac{3d_1 + 2d_2 + 3d_3 + 4d_4 + d_5 + 3d_6 + 5d_7}{15 - (c_1 + c_2 + c_3 + c_4 + c_5 + c_6 + c_7)},$$

where d_i is a amount of mistakes in i module, $i = \overline{1{,}7}$.

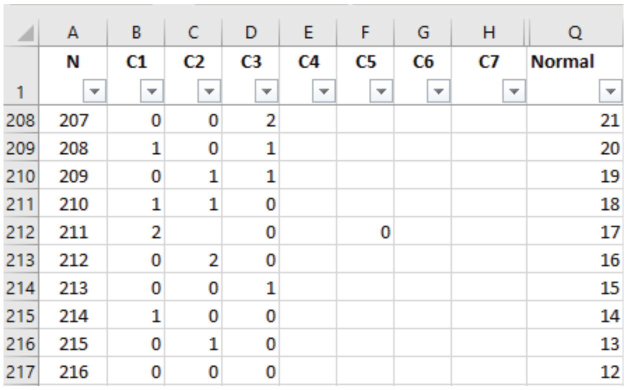
$$\mu = 15 - (c_1 + c_2 + c_3 + c_4 + c_5 + c_6 + c_7).$$

The Formula (2), which should be used for outcome estimation, is now used in the seven-module case. The structure of the outcome set of the three-stage model discussed in this article is different from the one discussed in the initial article by Razmadze et al ([5], p. 1656). This means that the domain of a function F(n) is different. Although, it is easy to check that despite this, F(n) function will provide a complete ordering of set N in the given case too.

Table 3. *1-3-3 model's outcome estimation by F(n) criterion*



	A	B	C	D	E	F	G	H	Q
1	N	C1	C2	C3	C4	C5	C6	C7	Normal
2	1	5			5			5	100
3	2	5			5			4	99
4	3	5			4			5	99
5	4	4			5			5	98
6	5	5			5			3	98
7	6	5			4			4	97
8	7	5			3			5	97
9	8	4			5			4	96
10	9	4			4			5	96
11	10	5			5			2	95



	A	B	C	D	E	F	G	H	Q
1	N	C1	C2	C3	C4	C5	C6	C7	Normal
208	207	0	0	2					21
209	208	1	0	1					20
210	209	0	1	1					19
211	210	1	1	0					18
212	211	2		0		0			17
213	212	0	2	0					16
214	213	0	0	1					15
215	214	1	0	0					14
216	215	0	1	0					13
217	216	0	0	0					12

The results obtained by using $F(n)$ criterion is shown in Table 3, where $c_1,\ c_2,\ c_3,\ c_4,\ c_5,\ c_6,\ c_7$ values are given in the columns B, C, D, E, F, G, H, respectively. The values calculated using Formula (2) are shown in column N. The data is sorted according to N Column decreasing order. Table 3 shows the first 10 (left half) and the last 10 (right half) testing outcomes' estimation results.

The whole table graphically looks as follows (Figure 3):

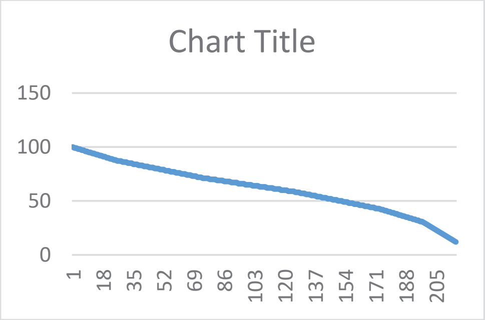


Figure 3. The graph of 1-3-3 MST model testing outcomes' score's normal distribution

V. CONCLUSION

The ordering method of the outcome set can be used in case of different testing procedures. The obvious example of this is the realization of the method for multistage adaptive testing's (MST) 1-3-3 model, which is described in the presented paper.

The author of a test has no direct contact with this method and its specific nuances because the realization of the method is a one-time procedure, carried out during the computerized adaptive testing portal formation.

The method does not require a detailed calibration of the item pool or preliminary testing of examinees to create a calibration sample. The ordering method of outcome set is oriented on the test author; it helps him avoid the problem of preliminary adaptation of test items for the examinee's knowledge level and simplifies the workload at maximum. Preliminary work for the test author might only include the division of test items into several difficulty levels based on expert assessment.

In the situation where there is a lack of information about test items and examinee's level, the method maximally uses the existing information for an examinee estimation: it takes into account all the answers to the questions provided to the examinee and the set of received answers is compared to all the possible variants and placed on corresponding level in the estimation hierarchy.

The paper presents the usage of the ordering method of outcomes set for multistage adaptive testing (MST) model as a sample. The method can be used for different modern testing models, but it is the subject of further research.

REFERENCES

1. Embretson, S. E. & Reise, S. P. (2000). Item response theory for psychologists. Multivariate Applications Books Series. Mahwah, NJ: Lawrence Erlbaum Associates Publishers;
2. Baker, F. (2001). The basics of item response theory. ERIC Clearinghouse on Assessment and Evaluation, Maryland, MD: University of Maryland, College Park. Retrieved from http://echo.edres.org:8080/irt/baker ;
3. Betz, N. & Weiss, D. (1973). An empirical study of computer-administered two-stage ability testing, https://bit.ly/2vjmqou , Technical Report. Research Report 73-4, Psychometric Methods Program, Department of Psychology, University of Minnesota, Minnesota, MN.
4. Betz, N. & Weiss, D. (1974). An empirical study of computer-administered two-stage ability testing, https://bit.ly/2ku9yht. , Technical Report Research Report 74-4, Psychometric Methods Program, Department of Psychology, University of Minnesota, Minnesota, MN.
5. Razmadze, S., Razmadze, M. & Razmadze, T. (2017). The set ordering method for scoring the outcomes of testing in computerized adaptive testing. International Journal of Scientific and Engineering Research (IJSER), Volume 8, Issue 5, May 2017. Retrieved from http://www.ijser.org/onlineResearchPaperViewer.aspx?The-Set-Ordering-Method-for-Scoring-the-Outcomes-of-Testing-in-Computerized-Adaptive-Testing.pdf ;
6. Razmadze, S. The Set Ordering Method for Scoring the Outcomes of 1-2-4 Multistage Model of Computerized Adaptive Testing, Applied Mathematics, Vol. 9 No. 1, 2019, pp. 6-12. doi: 10.5923/j.am.20190901.02. Retrieved from http://article.sapub.org/10.5923.j.am.20190901.02.html ;
7. Razmadze, S. The Set Ordering Method for Scoring the Outcomes of 1-2-3-4 Multistage Model of Computerized Adaptive Testing, LONDON JOURNAL OF RESEARCH IN SCIENCE: NATURAL AND FORMAL. Volume 23, Issue 17, pp. 1-13, „Great Britain Journals Press", London, 2023. Retrieved from https://journalspress.com/LJRS_Volume23/The-Set-Ordering-Method-for-Scoring-the-Outcomes-of-1-2-3-4-Multistage-Model-of-Computerized-Adaptive-Testing.pdf ;
8. Zheng, Y., Nozawa, Y., Gao, X. & Chang, H. (2012). Multistage adaptive testing for a large-scale classification test: Design, heuristic assembly, and comparison with other testing modes. ACT Research Report Series.

Modeling of a Queueing System with Tasks Requiring Multiprocessor Service

Vladimir Sahakyan
Institute for Informatics and Automation Problems of
NAS RA
Yerevan, Armenia
email: vladimir.sahakyan@sci.am

Artur Vardanyan
Institute for Informatics and Automation Problems of
NAS RA
Yerevan, Armenia
email: artur.vardanyan@iiap.sci.am

https://doi.org/10.62343/csit.2024.17

Abstract — This paper investigates a multiprocessor queueing system in which each incoming task requires a variable number of service nodes, reflecting the diverse resource demands found in modern parallel computing environments. By modelling the task arrival and service processes using Poisson and exponential distri-butions, the system's behaviour is analyzed through differential-difference equations that govern the steady-state probabilities. The research provides a comprehensive solution for determining the probability of different numbers of tasks being serviced, considering both the acceptance and rejection of tasks based on system capacity. The derived results offer valuable insights into optimizing system performance in terms of throughput, task completion, and resource utilization for multiprocessor systems.

Keywords — Queueing Theory, Multiprocessor System, Multi-server Queueing System, Multiprocessor Queueing System, Steady-State Probabilities.

I. INTRODUCTION

The rapid advancement of high-speed computing technologies has significantly transformed various scientific and industrial domains. Parallel computing systems, in particular, have become crucial for efficiently processing large-scale tasks that require substantial computational power. In recent years, multiprocessor systems, characterized by their ability to perform simultaneous operations across multiple processors, have gained considerable attention due to their potential to enhance performance and reduce execution times [1], [2]. In the context of queueing theory, multiprocessor systems can be modelled as multi-server service systems, where incoming tasks require the allocation of a specific number of service nodes for processing. Unlike traditional queueing models, which assume one service node per task [3], [4], this research introduces a custom number of service nodes for each task, depending on its resource requirements. This approach allows for a more realistic representation of modern computing envi-ronments, where task complexity and resource demands vary significantly.

The core objective of this study is to analyze the steady-state probabilities of the multiprocessor queueing system with tasks requiring multiple processors to service, considering exponential distributions for both task arrival and execution.

II. QUEUEING MODEL

This paper investigates a queueing system, with the follow-ing assumptions:

- A multiprocessor computing system comprising m processors (also referred to as cores or nodes, where $m \geq 1$) is considered a queueing system.
- Each task within the system is characterized by a random parameter, denoted as ν. Here, ν represents the number of computing resources required by the task for servicing, which could be processors, cores, cluster nodes, etc.
- Upon arrival in the system, tasks are subjected to either acceptance for servicing or rejection. The duration required for task servicing is partly contingent, it represents the maximum allowable time for task completion but is inherently random and may be shorter. Tasks face service denial if, upon entry into the system, it becomes evident that their specified parameters cannot be met. This occurs when the system lacks the requisite idle processors to initiate service.
- Tasks arrive at the system individually following a Poisson process characterized by a rate parameter α:
$$P(\alpha < t) = 1 - e^{-at},$$
where a is the intensity of the incoming stream($a > 0$).
- Service times are modelled by an exponential distribution with a density function described as follows:
$$P(\beta < t) = 1 - e^{-bt},$$
where β is a random value of the task execution time and b is the intensity of service($b > 0$).
- ν represents the random number of computational resources(service nodes) required for task execution. This parameter follows a probability distribution:
$$P(\nu = k) = \frac{1}{m},$$
where $k = 1, 2, ..., m$.

Upon arrival, tasks undergo either acceptance for service or rejection. Once service begins, it continues uninterrupted until completion. These assumptions provide a framework for analyzing the dynamics of task arrival and service completion within the multiprocessor queueing system.

III. MATHEMATICAL MODEL FORMULATION

To analyze the queuing system, it is essential to identify the following notation: $P_k(t)$ represents the probability that k tasks are being serviced in the system at time t.

It is a well-established principle that the flow resulting from multiple elementary flows remains elementary. Furthermore, the probability of multiple events occurring within a short interval h is negligible typically denoted as $o(h)$. Leveraging this fact, and considering all possible scenarios concerning the system's states at time t, specifically, focusing on instances where the system transitions during the time interval h to the state where k tasks are being serviced, the differential-difference equations of the system are given by:

$$\frac{dP_0(t)}{dt} = -aP_0(t) + bP_1(t), \; k = 0 \tag{1}$$

$$\begin{aligned}\frac{dP_k(t)}{dt} &= a\delta^{(1)}_{k-1}P_{k-1}(t) - [a(1-\delta^{(0)}_k) + kb]P_k(t) \\ &+ (k+1)bP_{k+1}(t), \; k \geq 1\end{aligned} \tag{2}$$

where the $\delta^{(0)}_k$ and $\delta^{(1)}_k$ probabilities are determined as follows:

$$\delta^{(0)}_k = P\left(\sum_{i=1}^{k+1}\nu_i > m \Big/ \sum_{i=1}^{k}\nu_i \leq m\right)$$

$$\delta^{(1)}_k = P\left(\sum_{i=1}^{k+1}\nu_i \leq m \Big/ \sum_{i=1}^{k}\nu_i \leq m\right)$$

It is evident that the following expressions express the conditional probability and by using the formula for calculating the conditional probability $\delta^{(0)}_k$ and $\delta^{(1)}_k$ probabilities can be calculated as follows:

$$\delta^{(0)}_k = \frac{P\left(\sum_{i=1}^{k}\nu_i \leq m < \sum_{i=1}^{k+1}\nu_i\right)}{P\left(\sum_{i=1}^{k}\nu_i \leq m\right)}$$

$$\delta^{(1)}_k = \frac{P\left(\sum_{i=1}^{k+1}\nu_i \leq m\right)}{P\left(\sum_{i=1}^{k}\nu_i \leq m\right)}$$

Using some sporting probability formulas from previous works[5] and simplifying the expressions, the following formulas are obtained:

$$\delta^{(0)}_k = \frac{k(m+1)}{m(k+1)} \tag{3}$$

$$\delta^{(1)}_k = \frac{m-k}{m(k+1)} \tag{4}$$

IV. STEADY-STATE PROBABILITIES

This section will outline the process of solving the recurrent equations derived from (1) and (2) when the queueing system goes into the steady state [6].

In steady state

$$\lim_{k\to\infty}\frac{dP_k(t)}{dt} = 0$$
$$\lim_{k\to\infty}P_k(t) = P_k$$

Therefore the equations (1) and (2) become:

$$-aP_0 + bP_1 = 0, \; k = 0$$

$$\begin{aligned}a\delta^{(1)}_{k-1}P_{k-1} - [a(1-\delta^{(0)}_k)+kb]P_k \\ + (k+1)bP_{k+1} = 0, \; k \geq 1\end{aligned}$$

Now, taking into account (3) and (4) formulas for the $\delta^{(0)}_k$ and $\delta^{(1)}_k$ probabilities and simplifying the expressions:

$$P_1 = \alpha P_0, \tag{5}$$

$$P_{k+1} = \frac{\alpha(m-k)+mk(k+1)}{m(k+1)^2}P_k - \frac{\alpha(m-k+1)}{mk(k+1)}P_{k-1}$$

where α is defined by $\alpha = a/b$.

By expanding the terms using the following definitions:

$$x_k = \frac{\alpha(m-k)+mk(k+1)}{m(k+1)^2}$$
$$y_k = \frac{\alpha(m-k+1)}{mk(k+1)}$$

The formula for P_{k+1} can be rewritten as:

$$P_{k+1} = x_kP_k - y_kP_{k-1} \tag{6}$$

The equations (5) and (6) are solved recurrently to obtain:

$$P_k = \frac{\alpha^k\prod_{i=1}^{k-1}(m-i)}{(k!)^2m^{k-1}}P_0, \; k \geq 1. \tag{7}$$

By substituting some binomial coefficient formulas and performing some straightforward mathematical transformations into equation (7), the probability P_k can be determined by the following formula:

$$P_k = \frac{\alpha^k}{k!\,m^k}\binom{m}{k}P_0, \; k \geq 1.$$

As the capacity of the system (i.e. the total number of tasks that can accommodate) is finite, say n, then using the condition of normality:

$$\sum_{k=0}^{n}P_k = 1$$

and by substituting the derived formulas for P_k into it, P_0 can be determined as the following:

$$P_0 = \frac{1}{1+\sum_{k=1}^{n}\frac{\alpha^k}{k!\,m^k}\binom{m}{k}}$$

V. CONCLUSION

In tIn this paper, a queueing model has been developed to analyze the steady-state behaviour of a multiprocessor system, where each task requires a custom number of service nodes. The inclusion of varying task resource requirements significantly enhances the model's relevance to modern high-performance computing environments. By solving the differential-difference equations and deriving closed-form expressions for the steady-state probabilities. These results provide a foundation for further optimization of multiprocessor systems, enabling better management of computational resources in environments with varying task complexities. The findings contribute to the broader field of queueing theory by extending traditional models to more accurately reflect contemporary parallel computing challenges.

REFERENCES

1. Vladimir Sahakyan, Artur Vardanyan, ”The Queue State for Multipro cessor System with Waiting Time Restriction”, Computer Science and Information Technologies 2019, Conference Proceeding, Yerevan, pp. 116–119, 2019. DOI: https://doi.org/10.1109/CSITechnol.2019.8895093.
2. Jain Anamika, Madhu Jain, Dheeraj Bhardwaj, ”Controllable multipro cessor queueing system”, Applications of Mathematical Modeling, Ma chine Learning, and Intelligent Computing for Industrial Development, pp. 61-76, 2023. ISBN: 9781003386599
3. Bocharov P.P., D'Apice C., Pechinkin A.V., Salerno S., ”Queueing Theory”, VSP, Utrecht, Boston, pp. 94-98, 2004. ISBN: 90-6764-398-X
4. Shortle J. F., Thompson J. M., Gross D., Harris C. M., Fundamentals of Queueing Theory, John Wiley and Sons, New York, pp. 35-475, 2018. ISBN: 111894352X
5. Vladimir Sahakyan, Artur Vardanyan, ”A Computational Approach for Evaluating Steady-State Probabilities and Virtual Waiting Time of a Multiprocessor Queuing System”, Programming and Computer Software, Volume 49, pp. S16–S23, 2023. DOI: https://doi.org/10.1134/S0361768823090098.
6. Kleinrock L., Queueing Systems: Vol. — Theory, John Wiley and Sons, New York, 1975, ISBN: 0-471-49110-1.

Analysis of Time Measures for the M/G/1 System in a Random Environment

Revaz Kakubava
Georgian technical university
Tbilisi, Georgia
e-mail: r.kakubava@gmail.com

Nani Salia
Georgian technical university
Tbilisi, Georgia
e-mail: nanisalia2@gmail.com

https://doi.org/10.62343/csit.2024.18

The paper discusses the M/G/1 System in a Random Environment . In the system, the server is influenced by a random environment, the latter being a simple birth-death process. An analytical stochastic model is constructed and studied in terms of operational calculus. Steady state Laplace transforms for distribution functions of time measures namely virtual waiting time, sojourn time are derived.

***Keywords*— Queueing system, random environment, time measures, Markov process, Laplace transform**

I. INTRODUCTION

The analysis and design of modern information and telecommunication systems require the development of queueing models that consider the specifics of control, management, and external actions such as preemptive and batch service. This includes taking into consideration a variety of aspects, such as parallel operations and processes, variations in parameters and characteristics of arrivals and service, failures, and environmental renewal. Analytical challenges in many practical cases are effectively addressed within the framework of queueing systems operating in a random environment [1-5]. This is particularly important in systems with situationally integrable resources, where resources can be pooled to solve specific tasks as situations arise. Situational integrability refers to the ability to combine resources of a queueing system to address specific tasks or groups of tasks as situations occur. This concept is fundamental in modern information and telecommunication networks, information processing centers including multiprocessor and multi-computing systems, switching node commutation units including special computing complexes, and multichannel data transmission channels. In such systems, the number of service facilities i.e channels for a given flow of customers changes randomly over time due to equipment failure, reallocation of server units for preemptive service, shutdowns for verification, diagnostics, maintenance, etc. Furthermore, the model can be generalized to include partial failures and other scenarios beyond those described in existing publications. This includes studying the intensity of server channels' failures and renewals, which may depend on whether the system is idle or busy, i.e., the parameters of the random environment change depending on the state of the queuing system.

The additional information about the modern situation can be found in the used literature which is given at the end of the topic [3-9]. In this paper we consider the M/G/1 queue random environment. The input flow intensity is λ. The server is influenced by a random environment i.e. a random process μ(t) with a set of states I. I is a set with elements I = {0,1,2}, and changes its state based on the state of the environment. In other words, if μ(t) = í ∈ I, í belongs to I set, then the server is in state í. State 0 is special, indicating that the server cannot begin service even if there are customers in the queue. When the system is busy, it operates in service cycles or sequences of cycles, with probabilistic characteristics independent of cycle types.

When the server is idle, the random environment μ(t) is modeled as a birth-death process with a set of states I = {0,1,2} and transition intensities $\alpha_í$ from í → í+1 where (í=0,1) and $\beta_í$ from í → í-1 (í=1,2).($\alpha_í$ sub í from state í transits in state í+1, where)When the system is empty, the birth-death process acts as a random environment, and when the system is busy, the random environment affects only the initial probabilistic characteristics of the service.

This process can describe sequences of failures and renewals in a multi-channel redundancy system, with the states of the process μ(t) defining the number of operative channels or the number of channels allotted for serving a special flow. In service cycles, the random environment μ(t) is disregarded, affecting only the initial characteristics of the queueing system. The behavior of the environment affects only the functions $H_{íj}(x)$, which is interpreted as follows: $H_{íj}(x)$ = ℙ{the service time of a customer is less than x and the state of the server or the process μ(t) at the end of the service time is j}, provided that the server was in the state í at the beginning of the service [1,2].

II. ANALYSIS OF TIME MEASURES

In this section we analyze the probability characteristics of the waiting time of the demand and the sojourn time in the system. The Laplace-Stieltjes transforms of the distribution functions of these random variables are found.

Denote by $P_i(t)$, i = 1,2, the probability that at the moment of time t the server is in the state i and there are no demands in the service system.

I want to point out u$_i$(x,t) as follows u$_i$(x,t)= $\lim_{h\to 0}$ ℙ {$x < U(t) < x + h$; after expiration U(t) (virtual waiting time) the server is in the state i}

Now I would like to formulate theorem n1

Theorem 1: The functions P_1, P_2, u_1 and u_2 satisfies the following system of differential and integro-differential equations which you can see on the slide

$$\frac{dP_1(t)}{dt} = -(\alpha+\lambda+\mu)P_1(t) + (\alpha+\beta)P_2(t) + u_1(t,0); \qquad (1)$$

$$\frac{dP_2(t)}{dt} = -(\alpha+\beta+\lambda)P_2(t) + \mu P_1(t) + u_2(t,0); \qquad (2)$$

$$\frac{\partial u_1(x,t)}{\partial \mathrm{t}} - \frac{\partial u_1(x,t)}{\partial x} =$$

$$-\lambda\, u_1(x,\mathrm{t}) + \lambda\int_0^x u_1(y,t)h_{11}(x-y)dy + \int_0^x u_2(y,t)h_{21}(x-y)dy +$$

$$+\lambda \mathrm{P}_1(t)h_{11}(x) + \lambda P_2(t)h_{21}(x) + \alpha P_1(t)\mu e^{-\mu x}; \qquad (3)$$

$\frac{\partial u_2(x,t)}{\partial t} - \frac{\partial u_2(x,t)}{\partial x} =$

$-\lambda u_2(x,t) + \lambda \int_0^x u_1(y,t) h_{12}(x-y)dy + \lambda \int_0^x u_2(y,t) h_{22}(x-y)dy$
$+\lambda P_1(t) h_{12}(x) + \lambda P_2(t) h_{22}(x).$ (4)

Now lets prove

Proof:

Let us consider an infinitesimal time interval $(t, t+h)$ and trace the behavior of the system in this interval. Conventional probabilistic reasoning leads to the following ratios: which you can see on the slide

$P_1(t+h) = P_1(t)[1-(\alpha+\lambda+\mu)h] + (\alpha+\beta)P_2(t)h + u_1(0,t) + o(h)$

$P_2(t+h) = P_2(t)[1-(\alpha+\beta+\lambda)h] + \mu P_1(t)h + u_2(0,t)h + o(h)$

$u_1(x,t+h) = u_1(x+h,t)(1-\lambda h) + \lambda h \int_0^x u_1(t,y) h_{11}(x-y)dy +$
$\lambda h \int_0^x u_2(y,t) h_{21}(x-y)dy + \lambda h P_1(t) h_{11}(x) + \lambda h P_2(t) h_{21}(x) +$
$+\alpha h P_1(t) \mu e^{-\mu x} + o(h);$

$u_2(x,t+h) = u_2(x+h,t)(1-\lambda h) + \lambda h \int_0^x u_1(y,t) h_{12}(x-y)dy +$
$+\lambda h \int_0^x u_2(y,t) h_{22}(x-y)dy + \lambda h P_1(t) h_{12}(x) +$
$+\lambda h P_2(t) h_{22}(x) + o(h),$

where $h_{ij}(x) = H'_{ij}(x)$. derivative

After simple transformations and passing to the limit at $h \to 0$, (h tends to 0) we obtain a system (1), (2), (3) and (4) expression.

We investigate the system at $t \to \infty$, (t tends to infinity) which corresponds to the stationary state.

The limits are denoted by $P_i = \lim_{t\to\infty} P_i(t)$; $u_i(x) = \lim_{t\to\infty} u_i(x,t)$; $i=1,2$, We assume these limits exist. Keep in mind that these limits are equal to 0.

$$\lim_{t\to\infty} \frac{p_1(t)}{dt} = 0 \qquad \lim_{t\to\infty} \frac{\partial u_i(x,t)}{\partial t} = 0$$

After passing to the limit at $t \to 0$ we obtain the following system:

$(\alpha + \lambda + \mu)P_1 = (\alpha + \beta)P_2 + u_1(0);$ (5)
$(\alpha + \beta + \lambda)P_2 = \mu P_1 + u_2(0);$ (6)

$\frac{du_1(x)}{dx} = -\lambda u_1(x) - \lambda \int_0^x u_1(y) h_{11}(x-y)dy -$
$\lambda \int_0^x u_2(y) h_{21}(x-y)dy - \lambda P_1(t) h_{11}(x) - \lambda P_2(t) h_{21}(x) -$
$\alpha \mu P_1(t) e^{-\mu x};$ (7)

$\frac{du_2(x)}{dx} = \lambda u_2(x) - \lambda \int_0^x u_1(y) h_{12}(x-y)dy -$
$\lambda \int_0^x u_2(y) h_{22}(x-y)dy - \lambda P_1(t) h_{12}(x) - \lambda P_2(t) h_{22}(x)$ (8)

Let us apply the Laplace transform to the last two equations. If we take into account the next expression which you see on the slide,

$\int_0^\infty e^{-sx} \frac{du_1(x)}{dx} dx = s \int_0^\infty e^{-sx} u_1(x)dx - u_1(0)$, $i=1,2$

expressing $u_i(0)$ through $\mathbb{P}_i$, from first two equations (5), (6) and solving the system after transformation with respect to $u_i^*(s)$ we obtain the next expression:

$u_i^*(s) = \frac{d_i(s)}{d(s)}$, $i=1,2$ (9)

where

$d(s) = [s-\lambda+\lambda h_{11}^*(s)][s-\lambda+\lambda h_{22}^*(s)] - \lambda^2 h_{12}^*(s) \cdot h_{21}^*(s);$
$d_i(s) = [s-\lambda+\lambda h_{jj}^*(s)]\varphi_j(s) - \lambda h_{ji}^*(s)\varphi_i(s), \quad i+j=3;$
$\varphi_1(s) = (\alpha+\lambda+\mu)P_2 - \mu P_1 - \lambda P_1 h_{12}^*(s) - \lambda P_2 h_{22}^*(s);$
$\varphi_2(s) = (\alpha+\beta+\lambda)P_1 - (\alpha+\beta)P_2 - \lambda P_1 h_{11}^*(s) - \lambda P_2 h_{21}^*(s)-$
$-\alpha\mu P_1/s+\mu.$

The expression for $u_i^*(s)$ contains two unknowns P_1 and P_2. To determine them we use the normalization condition

$P_1 + P_2 + \int_0^\infty [u_1(x) + u_2(x)]dx = 1,$

or in operational form which you can see on expression No 10.

$P_1 + P_2 + u_1^*(0) + u_2^*(0) = 1$ (10)

Using (9), we obtain the expressions for $u_i^*(0)$, $i=1,2$. After substituting these expressions into (10), we have the following equation

$[h_{12}^*(0) + h_{21}^*(0)\left(1+\frac{\alpha}{\mu}\right) - \mu(\tau_1-\tau_2)] +$
$+ [h_{12}^*(0) + h_{21}^*(0) + (\alpha+\beta)(\tau_1-\tau_2]h_2 =$
$= h_{12}^*(0)(1-\lambda\tau_2) + h_{21}^*(0)(1-\lambda\tau_1)$ (11)

Here

$\tau_1 = \int_0^\infty x\,[h_{i1}(x) + h_{i2}(x)]dx = -[h_{i1}^*(s) + h_{i2}^*(s)]'_{s=0}.$

$i=1,2$, i.e. τ_i is equal to the average service time per demand, assuming that service starts when the server is in the state i.

As it can be seen from (11), for the existence of the stationary state it is necessary to satisfy the condition which you can see on expression no 12.

$h_{12}^*(0)(1-\lambda\tau_2) + h_{21}^*(0)(1-\tau_1) > 0,$ (12)

i.e. $d'(0) < 0$.

To find the second equation with respect to P_1 and P_2 it is necessary to prove the following theorem.

Theorem 2: If condition (12) holds, the equation $d(s)=0$ has a real root $s_0>0$.

Proof: Consider a function $y=d(s)$ for real s. By direct substitution we obtain that d(0)=0. Moreover, as we pointed out above, condition (8) is equivalent to the condition d'<(0). It is also easy to see that $d(s) \to \infty$ at $s \to \infty$. Since $d(0)=0$ and $d'(0)<0$, in some neighborhood of point 0 the function y=d(s) decreases and, considering that d(∞) = ∞, it becomes obvious that there exists $s_0 > 0$ such that $d(s_0)=0$.

To illustrate this statement, let us show one of the variants for the graph of the function y=$d(s)$ in Figure. 1.

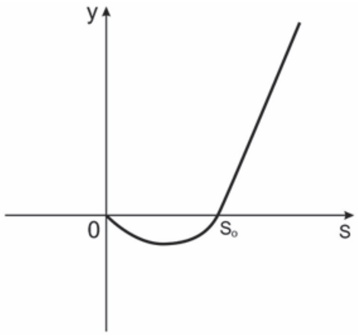


Fig.1

The functions $u_i^*(s)$ are analytic functions in the domain $Res>0$. Therefore, at the point s_0, $d_i(s_0)=0$, $i=1,2$. These equations give a single equation with respect to P_1 and P_2. It has the form $P_2 = aP_1$,

where

$$a = \frac{[s_o - \lambda + \lambda h_{22}(s_o)]\left[s_o + \alpha + \mu - \frac{\alpha\mu}{s_o + \mu}\right] + \lambda\mu h_{21}^*(s_o)}{(\alpha+\beta)[s_o - \lambda + \lambda h_{22}^*(s_o)] + \lambda(s_o + \alpha + \beta)h_{21}^*(s_o)}.$$

Finally we obtain the expressions for P_1 and P_2:

$$P_1 = \frac{h_{12}^*(0)(1-\lambda\tau_{2)} + h_{21}^*(0)(1-\lambda\tau_1)}{[h_{12}^*(0)+h_{21}^*(0)]\left(1+\alpha+\frac{\alpha}{\mu}\right)+[(\alpha+\beta)\alpha-\mu][\tau_1+\tau_{2\,]}}$$

$$P_2 = aP_1 \qquad (13)$$

Let us denote by $\Phi(s)$ Laplace transform of the virtual waiting time distribution function for $U(t)$. It is obvious that if there are no demands in the system and μ(t)=0, then $U(t)$=0 (the probability of this event is equal to $P_1 + P_2$), otherwise $U(t)$>0 (the probability density function of the virtual waiting time in this case is equal to $u_1(x)$+$u_2(x)$).

Hence

$$\Phi(s)=Me^{-s\omega(t)}= P_1 + P_2 + u_1^*(s)+u_2^*(s) \qquad (14)$$

Let us denote by $\Phi_1(s)$ the Laplace–Stieltjes transform of the distribution function for the customer sojourn time in the system. This time is the sum of waiting time and subsequent service time. The second term depends on the state in which the set of service devices was at the end of the waiting time (at the time of the beginning of subsequent service).

Considering the above, the following expression is obtained

$$\Phi_1(s)=[P_1 + u_1^*(s)]\, h_1^*(s)+[P_1+u_2^*(s)]h_2^*(s), \qquad (15)$$

where

$$h_\imath^*(s)=h_{\imath 1}^*(s)+ h_{12}^*(s)$$

I would like to underline the fact that this work was supported by Shota Rustaveli National Science Foundation of Georgia.

Combinatorial Analysis of the Reliability of Multi-core Processors

Sergo Tsiramua
University of Georgia
Tbilisi, Georgia
e-mail:s.tsiramua@ug.edu.ge

Hamlet Meladze
Muskhelishvili Institute of Computational Mathematics of Georgian Technical University
Tbilisi, Georgia
e-mail: h_meladze@hotmail.com

Tinatin Davitashvili
Ivane Javakhishvili Tbilisi State University,
Tbilisi Georgia
e-mail: tinatin.davitashvili@tsu.ge

Aleksandre Benashvili
Georgian Technical University
Tbilisi, Georgia
e-mail: sbenashvili20@gmail.com

Lazare Kvirtia
University of Georgia
Tbilisi, Georgia
e-mail: lazarekvirtia@gmail.com

Irakli Basheleishvili
Akaki Tsereteli State University
Kutaisi, Georgia
e-mail: basheleishvili.Irakli@gmail.com

https://doi.org/10.62343/csit.2024.19

***Abstract* — The demand for reliability in multi-core processors, which are ubiquitous in today's computers, smartphones, and various other smart devices, is growing. Several factors can influence the seamless operation of multi-core processors. For instance, at high frequencies, challenges such as signal distribution and maintaining low power consumption are prominent. An overheated processor can result in various issues in a computer, including freezes, unexpected shutdowns, and errors while running programs. These issues can have serious consequences, particularly during critical tasks. Processor cores belong to the class of multifunctional elements. Their multifunctionality contributes to the structural flexibility of the processor, allowing functions to be redistributed between cores and ensuring the processor can continue to operate successfully even if individual core blocks fail. This flexibility significantly enhances processor reliability, as demonstrated by quantitative assessments of the fault tolerance of multi-core processors, which is the central focus of this article.**

***Keywords* — Multicore processors, Combinatorial analysis, Multifunctional elements, Reliability, Flexibility.**

I. INTRODUCTION

Enhancing both the performance and reliability are important features of processors. Historically, the progression of computer technology has been propelled by the ability to integrate an increasing number of transistors in circuits. However, a further increase in the performance of processors through the improvement of the technological process of their manufacture is unlikely in the near future. A further significant increase in processor performance is possible only with new architectural solutions under existing technologies. A significant increase in processor performance is possible using the principle of asynchrony [1], although the realization of this idea is unlikely in the near future, as it requires fundamental changes in both the hardware and software areas of the computer.

In order to increase the performance of the processor, designers turn to architectures of parallel operation. Parallelism can be implemented at different levels. At the lowest level, it can be realized at the expense of convolution and superscalar architecture in the processor core. In this case, the processor core has several functional blocks and the parallel execution of commands is based on the parallel processing of several threads.

The next level of parallelism is achieved through multi-core architecture, where multiple processor cores are integrated into a single crystal or unit, operating in parallel. The implementation of numerous cores within multi-core processors is effective particularly when complex tasks are performed in parallel [2]. However, parallel architectures, in addition to increasing the performance of the processor, offer great opportunities in terms of increasing the reliability of the processor. The multi-core and parallel computing capabilities of the processors create a prerequisite for increasing the failure-resistanting and reliability of the processors. For instance, in the event of a functional block failure within one processor core, its functions can seamlessly transfer to a corresponding block within another core, enabling uninterrupted processor operation [3], [4]. In this case, mathematical modeling and analysis of the described process is interesting and important.

II. COMBINATORIAL ANALYSIS

It is preferable to use combinatorial analysis methods to calculate the probabilities of falsehoods associated with individual blocks in a multi-core processor. Let us consider a processor core model with four executive (functional) blocks: an arithmetic-logic device, two blocks for accessing memory, one block performing operations on floating-point numbers, which independently of each other, perform the corresponding type of operations in parallel (see Fig. 1).

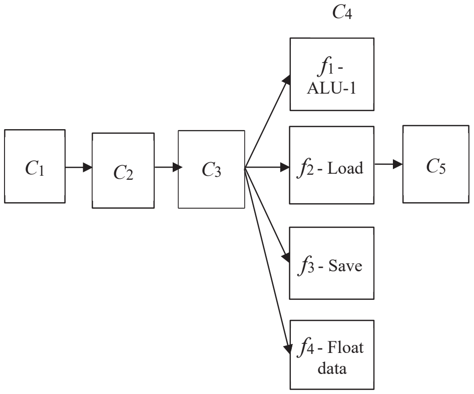


Fig. 1. Superscalar processor core with four execution blocks.

In the Fig. 1, C_1 represents command selection block, C_2 - decoding block, C_3 - operand selection block, C_4 – functional block and C_5 - the block of writing in registers.

It is supposed that the processor is in a state of marginal performance-ability if the functions of all four execution blocks are performed in its cores. Even a single block failure causes the processor to fail. With such an assumption, the shortest ways of functioning of an n-core processor can be described by the following logical function (conjunctions):

$$S_q = a_{i1}(f_1) \,\&\, a_{i2}(f_2) \,\&\, \cdots \,\&\, a_{im}(f_m) = 1,$$

where a_i denotes i-th block, $i_1,\ i_2, \cdots,\ i_m \in [1, n],\ \ q \in [1, N_S],\ N_S = 2^m$.

The condition of the processor's performance-ability is written by the disjunction of the ways of functioning:

$$F_A[a_1(f_1), a_2(f_2), \cdots, a_n(f_m)] = \bigcup_{q=1}^{N_S} S_q.$$

If we consider the example of the core given in Fig. 1, then a 2-core processor of class $n = 2, m = 4$ with 4 operational blocks can be described by the following (0,1) matrix:

$$B_A[a_i(f_j)] = \begin{Vmatrix} a_1(f_1) & a_1(f_2) & a_1(f_3) & a_1(f_4) \\ a_2(f_1) & a_2(f_2) & a_2(f_3) & a_2(f_4) \end{Vmatrix}$$

where $a_i(f_j) = 1$ if j block of a_i core performs f_j function and $a_i(f_j) = 0$, in opposite case.

If we introduce notations:

$$a_1(f_1) = x_1,\ a_1(f_2) = x_2,\ a_1(f_3) = x_3,$$
$$a_1(f_4) = x_4,\ a_2(f_1) = x_5,\ a_2(f_2) = x_6,$$
$$a_2(f_3) = x_7,\ \ a_2(f_4) = x_8,$$

then the shortest paths for the operation of a 2-core processor are written by the following conjunctions:

$$S_1 = x_1x_2x_3x_4,\ S_2 = x_1x_2x_3x_8,\ S_3 = x_1x_2x_4x_7,$$
$$S_4 = x_1x_3x_4x_6,\ S_5 = x_2x_3x_4x_5,\ S_6 = x_1x_2x_7x_8,$$
$$S_7 = x_1x_3x_6x_8,\ S_8 = x_2x_3x_5x_8,\ S_9 = x_1x_4x_6x_7,$$
$$S_{10} = x_2x_4x_5x_7,\ S_{11} = x_3x_4x_5x_6,\ S_{12} = x_1x_6x_7x_8,$$
$$S_{13} = x_2x_5x_7x_8,\ S_{14} = x_3x_5x_6x_8,\ S_{15} = x_4x_5x_6x_7,$$
$$S_{16} = x_5x_6x_7x_8.$$

The same ways of functioning with (0,1) logical variables are given in Table 1:

Table 1

	a_1				a_2			
A/F	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
X	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
1	1	1	1	1	0	0	0	0
2	1	1	1	0	0	0	0	1
3	1	1	0	1	0	0	1	0
4	1	0	1	1	0	1	0	0
5	0	1	1	1	1	0	0	0
6	1	1	0	0	0	0	1	1
7	1	0	1	0	0	1	0	1
8	0	1	1	0	1	0	0	1
9	1	0	0	1	0	1	1	0
10	0	1	0	1	1	0	1	0
11	0	0	1	1	1	1	0	0
12	1	0	0	0	0	1	1	1
13	0	1	0	0	1	0	1	1
14	0	0	1	0	1	1	0	1
15	0	0	0	1	1	1	1	0
16	0	0	0	0	1	1	1	1

Using our application based on combinatorial analysis and logical-probabilistic methods, the following results were obtained:

- When $p_i = 0.99$, where p_i - is the probability of the functionality of each functional block, the probability of the processor's faultlessness is $P(A_F) = 0.9996$;
- The number of all possible states - $N_\Omega = 2^{nm} = 2^8 = 256$;
- The number of operational states - $N_R = 81$;
- The coefficient of structural perfection of the processor - $\eta_A = N_R/N_\Omega = 0.31640625$.

The same result of η_A ($\eta_A = 0.31640625$) is obtained if we insert into the reliability polynomial the probability of operation of each functional block $p_i = 0.5$.

For the purpose of comparative analysis, let us consider a 4-core processor of class $n = m = 4$, in which each core contains 4 functional blocks. We consider the limiting case when in each core operates one block in different combinations so as to ensure the operation of all four executing blocks at the same time [5].

In such a case, when $n = m = 4$, the functional resource matrix of the processor will be $B_A(4 \times 4) = [a_i(f_j)]$, and the shortest paths of operation will be written by the following logical function [4], [6], [7]:

$$S_q = a_1(f_{j1}) \,\&\, a_2(f_{j2}) \,\&\, a_3(f_{j3}) \,\&\, a_4(f_{j4}) = 1,$$

where

$$j_1,\ j_2,\ j_3,\ j_4 \in [1, 4],\ \ j_1 \neq j_2 \neq j_3 \neq j_4,$$
$$q \in [1, N_S], \qquad N_S = 4! = 24.$$

The results of calculations performed by combinatorial analysis methods are presented in Table 2.

Table 2

γ	$N_M(n,\gamma)$	$N_L(n,\gamma)$	$g_A(\gamma)$
0	1	1	1
1	16	16	1
2	120	120	1
3	560	560	1
4	1820	1812	0.9956
5	4368	4272	0.9780
6	8008	7432	0.9281
7	11440	9312	0.8140
8	12870	8010	0.6224
9	11440	4464	0.3902
10	8008	1512	0.1888
11	4368	288	0.0659
12	1820	24	0.0132
13	560	0	0
14	120	0	0
15	16	0	0
16	1	0	0
Σ	$N_\Omega = 65536$	$N_R = 37823$	$\eta_A = 0.57713$

In Table 2:

- γ is a number of functional blocks out of order in the processor, $\gamma \in [0,16]$;
- $N_M(n,\gamma)$ is a number of all possible states with γ number of failures, $N_\Omega = 2^{16} = 65536$;
- $N_L(n,\gamma)$ is a number of operational states with γ number of failures;
- $g_A(\gamma)$ is a processor's failure-resistanting coefficient with γ number of failures.

III. CONCLUSION

Based on our evaluation of the reliability indicators of the multi-core processor, several conclusions can be drawn:

1. The expansion of the processor's core count correlates with an increase in the number of shortest operation paths N_S , thereby enhancing the flexibility of the system's structure and maneuverability of the system.
2. Augmenting the core count substantially amplifies both the total number of possible operational states N_Ω and the number of states impacting the processor's performance N_R.
3. With a rise in the number of processor cores, there is a notable escalation in the proportion of performance states relative to all possible states, indicating a higher structural perfection index η_A for the processor.

Concurrently with the core count increment, there is a marked increase in the likelihood of the processor functioning without a malfunction and a failure resistance rate.

REFERENCES

1. A. Benashvili, S. Sulkhanishvili,“Designing of the System Board on the Basic of an Asynchronous Serial Bus ”, *Georgian Engineering News*, 2015, №1(73), p. 57-60.
2. T. Davitashvili, H. Meladze, “Some Algorithms of Solving the Systems of Nonlinear Algebraic Equations on Parallel Computing Systems”, *Information and Computer Technology, Modeling and Control, Series: Computer Science, Technology and Applications,* Book Chapter 7, NOVA Science Publishers, USA, 2017, pp.69-84.
3. Arun K. Kanuparthi, R. Karri. "Reliable Integrity Checking in Multicore Processors", *ACM Trans. Architek. Code Optim*., vol. 12, no. 2, Article 10, 23 pages, 2015.
4. S.Tsiramua, H. Meladze, T. Davitashvili, I. Basheleishvili, “The systems with reconfigurable structure based on multifunctional elements”, *Proceedings of the 14th CSIT Conference 2023*, Yerevan, Armenia, September 25 – 30, pp. 198-201, https://doi.org/10.51408/csit2023_48 .
5. G. Tsiramua. *Discrete systems of variable structure.* Moscow, “Knowledge”, 1970 (Rus).
6. I. Basheleishvili, S. Tsiramua, “The Elaboration Algorithm for selection and Functions Distribution of Multifunctional Personnel”. *International Journal of Trend in Scientific Research and Development* (IJTSRD), Vol.1, Issue 5, www.ijtsrd.com, 2017, p. 828.
7. Tsiramuaa, S., Meladze, H., Davitashvili, T. “Logical-Probabilistic Modeling and Structural Analysis of Reconfigurable Systems Composed of Multifunctional Elements”, *Pattern Recognit. Image Anal*., 2024, Vol. 34, No. 1, pp. 144–157.

Robots Interaction Algorithms: The Results of Simulation Experiments

Mikhail Doryshin
HSE University
Perm, Russia
e-mail: modoryshin@edu.hse.ru

Elena Zamyatina
HSE University
Perm, Russia
e-mail: e_zamyatina@mail.ru

https://doi.org/10.62343/csit.2024.20

***Abstract* — The paper presents a study of robot interaction algorithms. The study of algorithms that control the behavior of robots is relevant due to the development of robotic means and their wide application in various fields: in manufacturing, medicine, military. The paper considers the application of nanorobot swarms in medicine. The study is performed using simulation methods. The simulation system NetLogo is used as a toolkit.**

***Keywords* — Robot, Nanorobot, Robots interaction algorithm, Simulation modeling, Agent-based modeling.**

I. INTRODUCTION

Recently, there has been a widespread use of robots in various fields: in production [1,2,3,4], in construction [5], in medicine [6], in military affairs, in regular household [10], etc.

The main advantages of automation using robots are:

- Replacing a person in the performance of tasks that may negatively affect his health;
- Improving the quality of work and productivity of enterprises;
- Reduction of staff remuneration costs;
- Elimination of the human factor from a number of business processes.

Often, when solving specific problems, one has to use several or a whole complex (“swarm”) of interacting robots. Robots act according to certain algorithms.

In this case, there is a need to apply certain methods to make sure that the interaction of robots is correct, which together try to solve the problem, to evaluate the effectiveness of robot interaction algorithms.

As such methods, it is advisable to use simulation methods that allow you to simulate the behavior in time of a complex simulated system, which is a swarm of robots (often fundamental mathematical methods cannot be applied in this case, since they require abandoning a number of details of the simulated complex system).

Each of the interacting robots can be represented by an agent acting according to a certain scenario in the agent- based simulation system. We will use NetLogo [8] as a tool for simulation experiments. Agent-based simulation system NetLogo is free software and has proven itself [9] when used in various projects. So, let's consider in more detail those tasks that are solved thanks to the use of robots..

II. RELATED WORKS

Currently, robots are used in almost all areas of human activity. Robots are used to solve a huge number of tasks, from complex and dangerous, like transporting heavy loads, to trivial, like counting goods in a store.

So the authors of the article "Robotics and automation of production: the state of the art" [1] consider the use of robots and other smart machines in industry to automate a huge range of processes from production to financial and economic. The use of robotics can not only significantly increase productivity in enterprises, but also ensure safety when performing dangerous tasks. Automation of production processes using robots makes it possible:

- to replace a person in the performance of tasks that may adversely affect his health;
- to improve the quality of work and productivity of enterprises;
- to reduce staff costs;
- to eliminate the negative impact of the human factor in a number of business processes.

The authors of the article “Task selection by autonomous mobile robots in a warehouse using deep reinforcement learning » [4] describe the solution to the problem of increasing the efficiency of a team of autonomous loading machines by improving the construction of routes for each machine, as well as preventing collisions between them.

The algorithm is based on fairly simple rules for the interaction of robots with cargo objects, as well as interaction with each other. The algorithm is designed to train the robot control system in a warehouse.

The essence of the algorithm is as follows:

- agent robots are located in the virtual warehouse,
- each of the robots must perform a specific task.

  Agent robots can be in one of two states:
- the robot is active if it is performing a task at a particular moment in time;
- the robot is inactive (idle).

Each inactive agent robot is assigned one of the pending tasks. This task can be completed in a specific location. As soon as the robot reaches the desired location, the task is marked as completed, and the robot again switches to an inactive state. To reach a specific location, robots need to avoid collisions with obstacles. The algorithm works as long as there are outstanding tasks. In the event that a collision occurs between two agent robots, the algorithm also stops working. The authors of the article claim that they managed to develop a control software system for constructing routes for loading robots in a warehouse, which allows them to perform work efficiently.

The paper [5] discusses the results of studies on the use of robots for building the walls of brick buildings. Robots are used to automate the process of laying bricks. The robot is also either active or inactive.

If the robot is active, then it performs the following sequence of operations:

- moves to the brick storage;
- picks up bricks;
- moves to the location of the bricks;
- laying bricks at the indicated position.

The conducted studies have shown that the simulation of algorithms for the interaction of robots allows avoiding collisions due to the clear formation of schedules for the stages of work performed, blocking the workspace during the laying of bricks, setting the exact duration of the stages of work (determining the speed of movement of the robot, the exact time of capturing and installing bricks), taking into account external factors. Similar studies are considered in [6,7], etc.

Let's take a closer look at the use of robotics in medicine, build a simulation model of the algorithm and perform a series of experiments that will determine, in particular, the number of nanorobots needed to destroy "bad" cells.

III. ALGORITHM OF OPERATION AND INTERACTION OF NANOROBOTS

"On the scalability of agent-based modeling for medical nanorobotics » [6] considers the use of nanorobots in the field of medicine, and more precisely, their application in the diagnosis and treatment of oncological diseases.

The task of nanorobots is to detect "bad" cells and neutralize them by introducing the appropriate drug. Each of the nanobots has two sensors with different ranges (aura-signal and aura-detector).

The nanorobot moves randomly in space. If a “bad” cell falls within the radius of action, the nanorobot detects it, immediately comes into contact with it, injects a drug and sends a signal to other nanorobots. If the nanorobot receives a signal from another robot, then it starts moving towards the robot that sent the signal and also starts injecting the drug into the “bad” cell. It should be noted that when moving in space, nanorobots may encounter obstacles on the way. They must be bypassed in order to continue the search for "bad" cells. Cells also move randomly.

The article also proposes the idea of forming an experiment roster aimed at conducting examinations of different behavioral patterns for the nanorobots, thus allowing investigation of several different robot interaction cases.

IV. NETLOGO SIMULATION SYSTEM

NetLogo discrete simulation system was created in 2002, the latest version of NetLogo 6.2.0 was released in September 2019. NetLogo can run on all underlying platforms: IOS, Windows, Linux. NetLogo provides many specialized libraries with which the user can create new models . In addition, the system contains a large number of ready-made models open for modifications. Users can also add their own models to the SIM. It should be noted that the Netlogo programming language of the same name is used to describe the models .

NetLogo implements agent-based modeling . Agents perform their assigned functions and act in parallel in time. Agents can be divided into the following types:

- turtles (represent an elementary object of any model, have properties such as number, color and coordinates, custom properties can also be added to turtles);
- links (link) (describe links between two turtles, links can be directional or non-directional);
- patches (patch) (represent elements of the modeling area that turtles can move on, patches have specific coordinates).

The development of simulation models in the NetLogo environment involves the creation of procedures that describe the interaction between agents. Currently, robots are used in almost all areas of human activity. Robots are used to solve a huge number of tasks, from complex and dangerous, like transporting heavy loads, to trivial, like counting goods in a store.

So the authors of the article "Robotics and automation of production: the state of the art" [1] consider the use of robots and other smart machines in industry to automate a huge range of processes from production to financial and economic. The use of robotics can not only significantly increase productivity in enterprises, but also ensure safety when performing dangerous tasks. Automation of production processes using robots makes it possible:

V. BUILDING A MODEL AND DESCRIPTION OF EXPERIMENTS

We will develop a simulation model to study the interaction algorithm of nanorobots designed to destroy "bad" cells.The model will realize the interaction between multiple types of NetLogo agents. First we need to identify the main elements of the algorithm that should be present in the model.

These elements are:

- Nanorobot;
- Cancer cell;
- The area of movement of the nanorobot / cell;
- Aura is the signal of the nanorobot;
- Aura is a nanorobot detector.

Now that the main elements are designated, we can designate a NetLogo agent for each of them:

- The nanorobot (turtle). The nanorobot is the main object in this model. He must constantly move around the virtual environment in search of cancer cells, so he must be presented as a turtle agent;
- The cell (turtle). Like the nanorobot, the cell is one of the main objects in the model. She also has to move around the virtual environment, so the most suitable agent for her is turtle;
- The area of movement (patch). In NetLogo models, all turtles move around patches. The movement area is an empty area in a virtual environment where cells and nanorobots can move around;
- The connection between nanorobots, received from the aura signal (link). When the nanorobot begins to interact with the cell, it sends a signal for help to other robots. Those who receive the signal begin to move in the direction of the source to take part in the elimination of the cell. Thus, a connection appears between two robots (source and receiver), which can be represented by the corresponding agent;
- The connection between nanorobots and cells (link).

In order to program the process of interaction between the robot and the cell, an element of communication between them is necessary. An appropriate agent can be used for this. Thus, when more connections appear between one cell and several nanorobots, the elimination process can be accelerated.

Let's imagine nanorobots that are constantly looking for "bad" cells using Netlogo agents (using Turtles). Since the "bad cells" also move, they should also be represented with the help of turtle agents. The area of movement of "bad" cells and nanorobots will be presented in the form of patches (Patch).

When interacting with nanorobots (a nanorobot sends a signal to other nanorobots if a "bad" cell is detected), as well as when interacting with a "bad" cell, you should use the NetLogo Link agent (communication).

The nanorobot can be in one of three states: in the state of search, in the state of interaction, in the state of destruction. Search state: the robot moves along a random trajectory, if a “bad” cell is found, it enters the interaction state. Interaction state: signaling to other nanorobots, transition to a state of destruction. Destruction state: destruction of the "bad" cell.

All actions in one state or another are described by procedures in the NetLogo language, let's list them: move-robots (describes the movement of a nanorobot); move-cells (describes cell movement); setup-space (changes the color of patches); setup-cells (used to create cells); find-cell (search for cells), etc.

Now we can move on to experimenting. For the number of cells and the number of robots, you can define a default value of 200 (200 cells and 200 robots), for the aura sensor, the default value is 10 patches, and for the aura detector, the default value is 2 patches. These values were used in the work [6]. Also, it should be noted that each cancer cell has a base health of 100 units. When interacting with a nanobot, the cell loses 5 health units per tick. A tick is a time unit of the ASIM NetLogo model. Next, we present the details and results of each of the experiments.

A. *Experiment №1*

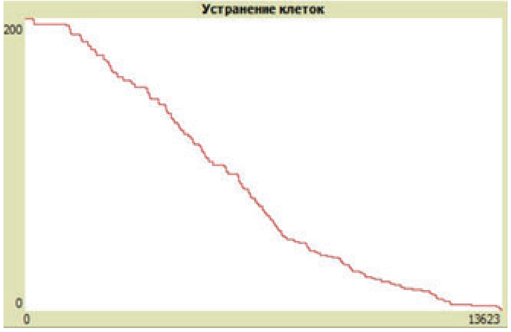


Fig. 1. Graph of the dynamics of changes in the number of cells (the number of robots is 1; the number of cells is 200)

As part of the experiment, the following input parameters are set:

- number of robots: 1 robot;
- number of cells: 200 cells;
- aura detector radius: 2 patches;
- aura signal radius: 10 patches.

The total running time of the model was 13623 ticks (see Fig 1). It should be noted that the rather long time of the model operation is obvious, since the nanorobot needs to constantly move from one cell to another.

In addition, the process of cell destruction also takes a long time, since one nanorobot is able to take only 5 cell health units per tick.

B. *Experiment 2*

Input parameters:

- number of robots: 1000 robots;
- number of cells: 200 cells;
- aura detector radius: 2 patches;
- aura signal radius: 10 patches.

The total running time of the model was 618 ticks (see Fig 2).Despite the fact that the number of nanorobots significantly outnumbers the number of "bad" cells, the process of cell destruction still took quite a long time. This is due to the fact that when receiving a signal for help, each free robot changes direction and starts moving towards the source of the signal, instead of trying to detect a "bad" cell nearby. Thus, it can be stated that the time costs are due to the movement of robots to the source.

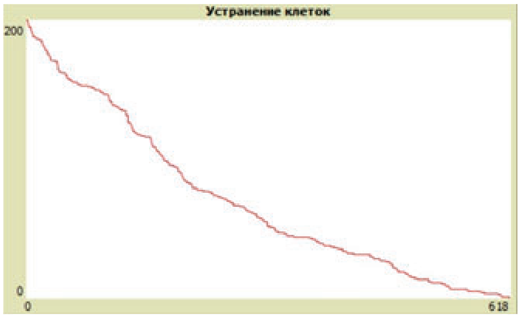


Fig. 2. Graph of the dynamics of changes in the number of cells (the number of robots is 1000; the number of cells is 200)

C. *Experiment №3*

Input parameters:

- number of robots: 200 robots;
- number of cells: 1;
- aura detector radius: 2 patches;
- aura signal radius: 10 patches.

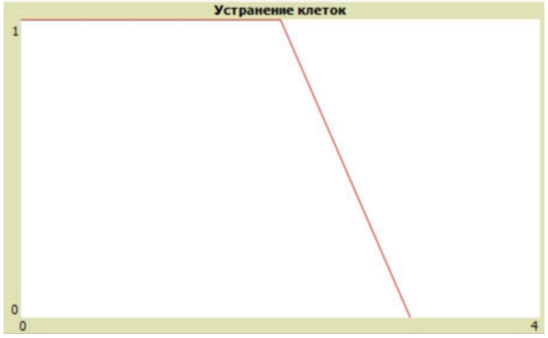


Fig. 3. Graph of the dynamics of changes in the number of cells (the number of robots is 200; the number of cells is 1)

The total running time of the model was 4 ticks (see Fig 3). Several robots managed to quickly deal with one cell. Most of the robots heading for the signal did not even have time to get to the cage.

D. *Experiment №4*

Input parameters: number of robots-200, number of cells-1000, aura-detector radius-2 patches, aura-signal radius-10 patches. The total running time of the model was 2038 ticks (see Fig 4). It should be noted that, despite the large difference between the number of cells and the number of robots, the task of destroying cells was solved quite quickly. Nanorobots were quite effectively distributed among the targets.

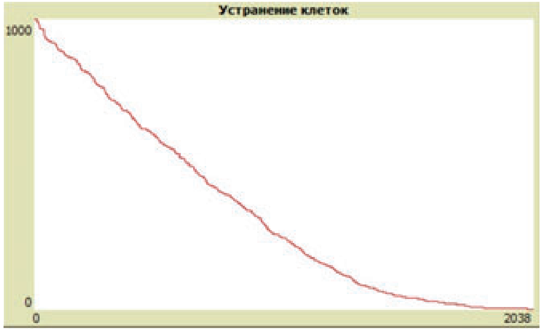


Fig. 4. Graph of the dynamics of changes in the number of cells (the number of robots is 200; the number of cells is 1000)

E. Experiment №5

Input parameters: number of robots: 200, number of cells: 300, aura detector radius: 2 patches, aura signal radius: 10 patches. The total running time of the model was 1175 ticks (see Fig. 5). Nanorobots quickly coped with the task, in fact, the number of cells exceeds the number of nanorobots by a small amount.Let's try to increase the radius of the aura-signal in order to notify as many nanobots as possible when a "bad" cell is detected.

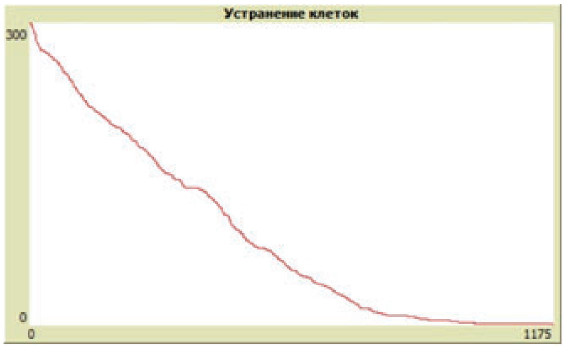


Fig. 5.Graph of the dynamics of changes in the number of cells (the number of robots is 200; the number of cells is 300, the radius of the aura-signal is 10 patches)

F. Experiment №6

Input parameters: number of robots: 200, number of cells: 200, aura detector radius: 2 patches, aura signal radius: 41 patches. The total running time of the model was 1186 ticks (see Fig.6).

Since the nanorobot that found the "bad" cell sends a "powerful" signal to all other nanorobots, they rush to the found cell, ignoring the cells that are in close proximity to them.

G. Summary

So, studies have shown that by manipulating the values of the radius of the aura-signal and the radius of the aura-detector, as well as the values of the number of nanorobots and "bad" cells, it is possible to optimize the work of nanorobots to destroy "bad" cells.

It can also be noted that the robot interaction algorithm is quite sensitive to changes in the aura-signal radius parameter.

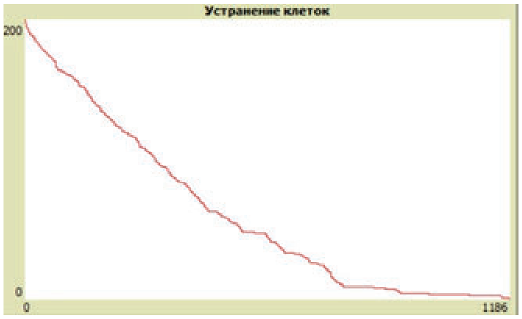


Fig. 6. Graph of the dynamics of the change in the number of cells (the number of robots is 200; the number of cells is 200, the radius of the aura-signal is 41 patches.)

VI. CONCLUSION

The paper presents studies related to the optimization of the algorithm for the interaction of nanorobots to destroy "bad" cells.

The studies were carried out using simulation methods. The system of agent-based simulation modeling NetLogo was used as a tool. Nanorobots are represented by agents - IM elements in NetLogo. The experience of using Netlogo can be extended to other examples of the use of robots.

Robots can act according to rather complex scenarios; when performing certain procedures, they can use knowledge. In order to perform simulation experiments involving intelligent robots, it is necessary to develop appropriate tools. It is in this direction that it is planned to continue research.

REFERENCES

1. G. A. Syretsky Robotics and automation of production: current state. *Computer and Information Sciences*, vol.5, no. 2- pp. 24- -29, 2017
2. D.D.Dudenhoeffer, D.J.Bruemmer and M.L.Davis. “Modeling and simulation for exploring human-robot team interaction requirements” *Proceeding of the Winter Simulation Conference.* – pp. 730—739, 2001
3. J.Viswanathan, D.M.Tilbury, W.S.Harrison, F.Gu.”Using hybrid process simulation to evaluate manufacturing system component choices: integrating a virtual robot with a physical system”. *Proceeding of the Winter Simulation Conference*- pp . 2822—2833, 2011.
4. M.P.Li, P.Sankaran, M.E.Kuhl, R.Ptucha, A.Ganguly and A. Kwasinski.”Task selection by autonomous mobile robots in a warehouse using deep reinforcement learning” *Proceeding of the Winter Simulation Conference*, pp. 680—689, 2019.
5. H. Mattern , T. Brukmann , A. Spengler, M. König.”Simulation of automated construction using wired robots“ *Proceeding of the Winter Simulation Conference Winter Simulation Conference,* pp. 3301-3133. 2016.
6. E.S. Liu. “On the scalability of agent-based modeling for medical nanorobotics” *Proceeding of the Winter Simulation Conference Winter Simulation Conference* – pp, 1427-1435, 2015
7. D.M. Turnage. “Simulation results for localization and mapping algorithms” Proceeding of the Winter Simulation Conference Winter Simulation Conference. – pp, 3040-3051, 2016.
8. NetLogo 6.2.0 User Manual: Programming Guide: [Electronic resource]. URL: NetLogo 6.2.0 User Manual: Programming Guide (northwestern.edu).
9. N.M. Ershov. *Introduction to distributed modeling by environment NetLogo.* Moscow, DMK Press , 2018. - 264 p.
10. H. Lee and A. Banerjee “Intelligent scheduling and motion control for household vacuum cleaning robot system using simulation based optimization”. *Proceedings of the Winter Simulation Conference.* – pp. 1163-1171, 2015