Computing, Telecommunications and Control

Vol. 17, No. 3
2024

Special issue
"Applied problem solving with machine learning"

COMPUTING, TELECOMMUNICATIONS AND CONTROL

ISSN 2687-0517

Информатика, телекоммуникации и управление

Том 17, № 3
2024

Тематический сборник "Решение прикладных задач методами искусственного интеллекта"

ИНФОРМАТИКА, ТЕЛЕКОММУНИКАЦИИ И УПРАВЛЕНИЕ

Contents

Applied problem solving with machine learning

Circuits and Systems for Receiving, Transmitting, and Signal Processing

Simulations of Computer, Telecommunications, Control and Social Systems

Содержание

Моделирование вычислительных, телекоммуникационных, управляющих и социально-экономических систем

Applied problem solving with machine learning
Решение прикладных задач методами искусственного интеллекта

EXO-INTELLIGENT HYBRID SUPERCOMPUTER PLATFORMS FOR SHARED-USE CENTERS

V.S. Zaborovsky, L.V. Utkin [ID] *, V.A. Muliukha* ✉ [ID]

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

✉ vladimir.muliukha@spbstu.ru

Abstract. The article discusses the possibilities of increasing the real performance of hybrid supercomputer platforms consisting of different types of processor nodes (CPU, GPU, FPGA) operating in the mode of shared-use computational resources. The conceptual difference of the proposed approach from widespread supercomputing cluster platforms can be metaphorically expressed as "Less Moore, more brain." The considered approach shifts the focus of technology development from classical methods of increasing the performance of HPC platforms by adding new hardware multi-core computing components to more complex exo-intelligent solutions that use inductive (internal) and conceptual (external) data to implement machine learning methods for the purpose of optimally distributing available hardware resources between different classes of user applications. The proposed three-level architecture of hybrid computing platforms opens up new opportunities both for efficient scaling of user program execution processes, and for reification of descriptions of new algorithms by generating corresponding texts of computer programs, as well as interpreting the results obtained based on the use of statistical information, the carrier of which is censored data characterizing the experience of executing user applications in the mode of shared use of hybrid computational resources.

ЭКЗО-ИНТЕЛЛЕКТУАЛЬНЫЕ ГИБРИДНЫЕ СУПЕРКОМПЬЮТЕРНЫЕ ПЛАТФОРМЫ ЦЕНТРОВ КОЛЛЕКТИВНОГО ПОЛЬЗОВАНИЯ

В.С. Заборовский, Л.В. Уткин ⓘ , В.А. Мулюха ✉ ⓘ

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

✉ vladimir.muliukha@spbstu.ru

Аннотация. В статье рассматриваются возможности повышения реальной производительности гибридных суперкомпьютерных платформ, состоящих из процессорных узлов различных типов (CPU, GPU, FPGA), работающих в режиме совместного использования вычислительных ресурсов. Концептуальное отличие предлагаемого подхода от широко распространенных суперкомпьютерных кластерных платформ можно метафорически выразить как "Меньше Мура, больше мозга". Рассматриваемый подход смещает фокус развития технологий с классических методов повышения производительности HPC-платформ путем добавления новых аппаратных многоядерных вычислительных компонентов на более сложные экзо-интеллектуальные решения, использующие индуктивные (внутренние) и концептуальные (внешние) данные для реализации методов машинного обучения с целью оптимального распределения доступных аппаратных ресурсов между различными классами пользовательских приложений. Предложенная трехуровневая архитектура экзо-интеллектуальных вычислительных платформ обладает новыми широкими возможностями как для эффективного масштабирования процессов выполнения пользовательских программ, так и для овеществления описаний новых алгоритмов путем генерации соответствующих текстов компьютерных программ, а также интерпретации полученных результатов на основе использования апостериорной статистической информации, носителем которой являются цензурированные данные, характеризующие опыт выполнения пользовательских приложений в режиме совместного использования гибридных вычислительных ресурсов.

Ключевые слова: высокопроизводительные гибридные вычислительные системы, машинное обучение, интеллектуальный диспетчер, функция выживаемости, объяснимый искусственный интеллект

> The source of coming-to-be for exiting things is that into which destruction, too, happens according to necessity
>
> *Anaximander of Miletus, 610—546 B.C.*

Introduction

Currently, the development and application of digital artificial intelligence systems are experiencing a period of rapid development [1—3], changing technological reality, clarifying the fundamental principles of computer science and supplementing computer programming paradigms with new mechanisms for the

interaction of natural and artificial intelligence [4]. It is well known that methods of digital modeling of physical processes and various types of intellectual activities associated with complex mathematical calculations are carried out using digital computers and are performed many times faster and with greater accuracy compared to human capabilities. At the same time, the cost of electronic memory systems is rapidly decreasing and the performance of digital microprocessors, called "graphics" accelerators, is steadily increasing, contributing to a shift in the emphasis of the development of computer technologies [4, 5] from digital modeling methods of complex physical processes towards probabilistic models, processing of graphic information and the formation of multimodal texts explaining the results of calculations [6, 7]. The ongoing changes caused an expansion of ideas about algorithms based on the ideas of computability of functions, enumerability and solvability of sets [8, 9], which stimulated the search for solutions to applied problems of a probabilistic nature, presented using semantic invariants, large linguistic models or topological data analysis methods. The processes of exponential acceleration and diversification of computer technologies were associated with the development of pre-trained transformer models (Generative Pre-trained Transformer, GPT), which have built-in attention mechanisms [10], which de facto became a bifurcation point in the evolutionary development of computer science in general. In the dialectical spiral of improving computer technologies, there has been a transition from digital platforms for the implementation of the Pythagoras's ancient formulation that "everything is number" to the reification of the meaning of the sacred phrase that "In the beginning was the Word" [John 1:1]. The bifurcation turn that occurred in the development of computer science from the concept of programming to the learning paradigm opens up new prospects for the use of digital intelligent technologies in various areas of human activity. One of the areas of application of machine learning methods that has not yet received sufficient attention is the intelligent control and "self-learning" of the computing platforms themselves, including the tasks of planning access to multimodal data stores and explaining the results of calculations.

The search for new ways to increase the real performance of computing platforms has become especially relevant in modern conditions, when the so-called "Moore's law", which for many years regulated of chip's performance and energy consumption, stopped working[1], and the time frame for the creation of quantum computers has still not been determined [11]. The article discusses the possibilities of using machine learning technologies at the "tactical level" to increase real productivity based on planning application program processing processes using resource managers of hybrid supercomputer platforms consisting of nodes of various types (CPU, GPU, FPGA) and operating in shared-use resource modes. The difference between the platforms being developed and well-known solutions based on supercomputer clusters is the presence in them of an "introverted" machine learning system for the application processes dispatcher. The hierarchical architecture of such a platform consists of three levels:

1) planning the load of computing nodes, taking into account the parameters of user requests and a labeled set of data on the results of previously performed calculations;

2) machine learning models used to assess the influence of internal and external (exo) factors on the state of the cluster's computational resources and the success of application programs;

3) explanations/interpretation of the results of performing applied tasks.

The article substantiates an attempt to materialize the ideas of digital intellectualization, for which a description of computing processes is introduced based on the "survival" function of various classes of applied tasks being performed. This processing makes it possible to quantitatively characterize the real performance of a computing platform not by counting the number of machine operations performed, but by counting the number of successfully completed applied tasks in relation to the total number of executable programs. The introduced function allows us to evaluate the measure of "usefulness" of the dispatcher's work in terms of the probability of successful execution of certain classes of applied tasks, subject to maximizing the load of all hybrid nodes of the computing cluster. To construct the survival function of applied tasks of a specific platform, it is proposed to use information about the "credit history"

---

[1] Moore's law – Wikipedia, Available: https://en.wikipedia.org/wiki/Moore%27s_law#cite_ref-30 (Accessed 09.10.2024)

of computing in relation to various types of application programs and categories of registered users. The data used to calculate this function includes the values of applied task descriptors, the number and type of computing nodes, the number of successfully or unsuccessfully completed tasks during the time interval allocated for task execution by the dispatcher, etc. The article shows how, using machine learning algorithms for models that predict the amount of time interval sufficient to successfully complete tasks and methods for explaining the results obtained (eXplainable Artificial Intelligence, XAI), it is possible not only to identify, but also to quantify various factors influencing probabilistic characteristics of successful completion of applied computing processes. The results of explanations "computed" in this way, along with the results of executing the applied tasks themselves, are available not only to users, but also to the computational resource manager, which allows increasing the level of real productivity for the entire exo-intelligent supercomputer platform. In this case, technologies for replenishing the functionality of existing computers and exo-mechano/energy platforms with additional exo- intelligent capabilities are in demand and relevant. The use of a combination of high performance computing (HPC) technologies and big data processing algorithms with intuitive principles of finding solutions and planning actions under conditions of uncertainty can be considered as a process of conceptual validating idea of "digital of intelligence". The essence of the idea is that any observable or measurable physical process must a priori have some formal description that allows for the implementation of a number of exo- intelligent operations, namely:

• copy results in the form of a symbolic code, which opens up new opportunities for storing and transferring knowledge;

• process information at high speed and constantly improve processing algorithms through machine learning processes.

However, on the way to implementing the hypothesis, both fundamental and technological problems arise: modern digital systems and software technologies are not capable of modeling processes that do not have a formal mathematical description, for example, cognitive processes. Without going into theological details, it can be noted that the need for the hypothesis of digital intelligence presupposes the existence of an a priori formal description of the observed processes and, as a consequence, the existence of an environment capable of not only generating such a description, but also implementing mechanisms for translating (reifying) this description into a real physical process.

The article examines all the questions formulated above using the example of the hybrid computing platform of the Polytechnic supercomputer, operating in the mode of a shared-use resource center. The article proposes a description of complex computing platforms using the survival function [8] of applied tasks, which characterizes the probability of successful execution of applied tasks based on the use of censored data [11] on the functioning of the computing system as a whole under the control of a resource manager.

The approach proposed in the article to describing computational processes in hybrid computing platforms is based on the principle of multiple machine learning models (an analogue of the physical principles of complementarity) used to estimate the time interval sufficient for the successful completion of various classes of applied tasks [8]. The pre-trained models are used by the dispatcher to quickly schedule the load of computing nodes in such a way as to increase the likelihood of successful completion of all applied tasks. The principal ability to solve machine learning problems in such system is based on the experimentally established fact of statistical stability (Fig. 1) of the observed processes, characterizing the proportion of applied tasks, the completion of which is marked by the system dispatcher with a success code of "0".

The data presented in Fig. 1 reflect the objectively existing features of hybrid supercomputer platforms operating in the mode of shared-use centers. These features are available for mathematical description in order to model the processes occurring and explain the results obtained. The obtained explanations of the results are local in nature and are based on an assessment of the influence coefficients of various
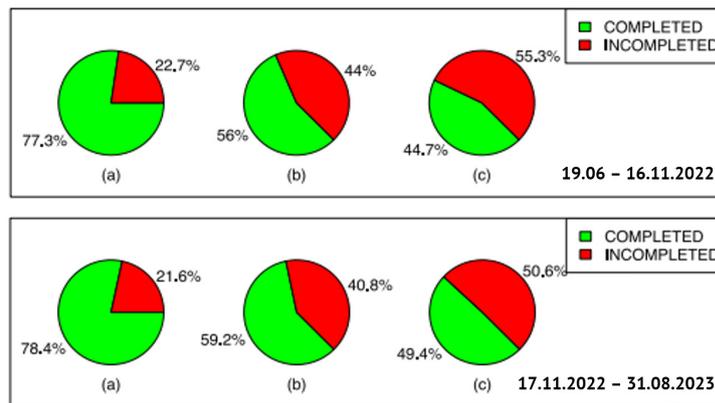
Fig. 1. (a) number of user tasks; (b) time spent; (c) CPU time

factors characterizing the operation of the supercomputer platform as a whole from the point of view of the successful execution of various classes of applied tasks. Taking into account the above, the article proposes a constructive extension of the measure of real performance of a supercomputer platform as the proportion of successfully solved problems to the total number of applied tasks completed during the observation time interval. The application of this performance indicator in practice comes down to solving the inverse problem of algorithmic modeling of "big data", taking into account the locality of the digital and linguistic variables used [12], as well as various attention mechanisms [9], including various methods of information retrieval, for example, based on the so-called inverted index algorithms. Replenishing the architecture of a computer platform with machine learning mechanisms leads to the fact that exo-intelligent systems no longer "suffer" from large volumes of processed information, but, on the contrary, acquire fundamentally new opportunities to increase their real productivity by accumulating experience in solving applied tasks in combination with "inverse" methods. » transformation of calculation results into process planning algorithms and explanations of the results obtained.

It should be noted that classical methods for solving inverse problems, developed by academician A.N. Tikhonov [13], are effective for situations, where possible relationships between solutions of computational problems and input data have an analytical representation, for example, in the form of physical laws. In the situation in which machine learning methods are proposed to used, the analyticity description of such representation is not available due to the combinatorial complexity of the process of selecting "inverse" solutions and the uncertainty of the formal description of the input data corpus. Taking into account this obstacle, the article presents a new approach to the regularization of solutions, obtained using machine learning methods and based on local interpretable model of the observable computational processes. Such interpretable model characterizes survival functions of various classes of user tasks that should be similar to those functions that were obtained experimentally on over large observation intervals [14].

Polytechnic SCC as a Machine Learning Hybrid Platform

The high-performance hybrid supercomputer platform of the "Polytechnic supercomputer center" (Polytechnic SCC) is a highly complex technical system that simultaneously performs a quadrillion (a number with 15 zeros) machine operations per second, which is used to solve various types of user tasks and algorithms that solve computational problems of various complexity classes. Supercomputer users, whose number is steadily growing every year, are interested in the successful result of their calculations, and not interested in the peak performance, which is nominally expressed in the number of floating-point arithmetic operations per second. All registered users get access to SCC resources from the dedicated server using the terminal client protocol. SCC resources are managed using the Slurm dispatcher (Fig. 2): the
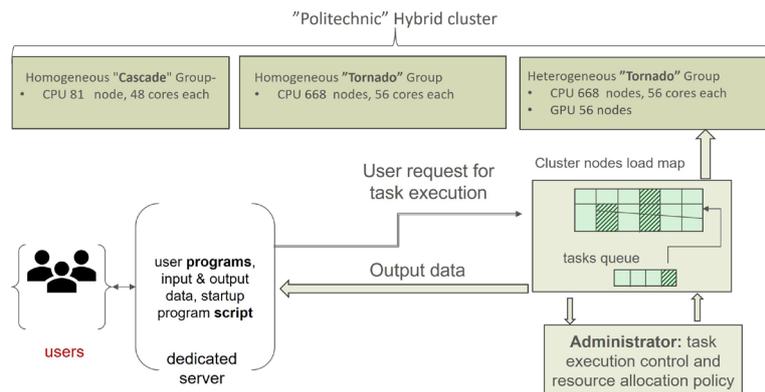
Fig. 2. Structure and data flows of the Polytechnic SCC

user requests some resource (processor cores, memory, etc.), placing his task in the queue; the system, based on the user's priorities and the current filling of the queue, selects the moment of task launch. A queue is a sequence of tasks that must be solved on a specific computational resource (a group of nodes). At the same time, each node at the current time can be occupied by only one task of a user. Thus, the node is assigned to the exclusive use of the hosted tasks, and other tasks on the busy node will not be executed.

All tasks can be divided into classes depending on the areas of expertise: Astrophysics, Bioinformatics, Biophysics, Energetics, Geophysics, IT, Mechanical engineering, Mechanics, Physics and Radiophysics. In Fig. 3 each of the classes is indicated by its own color and occupied one or several nodes of hybrid cluster: IT tasks are black, Astrophysics tasks are turquoise, Bioinformatics tasks are red, Biophysics tasks are pink, Energetics tasks are green, Geophysics tasks are yellow, Mechanical engineering tasks are blue, Mechanics tasks are gray, Physics tasks are lime, Radiophysics tasks are orange, Uncertain tasks are purple, and white indicates node idleness.

By analyzing Fig. 3, we propose two slogan that explain relationship between machine learning (ML) and HPC platform, namely "A frontier ML system is HPC", and vice versa "HPC is a frontier ML system". These slogans have not become a truism yet, but they clearly reflect the ideas of computer evolution tendency — frontier HPC systems become not only the driving force of digital transformation process, but also an active part of machine learning ecosystem. The Fig. 3 clearly shows that during runtime SCC activities some cluster nodes are idle for a long time, since there are no applied tasks in the queue that can use free computational resources upon user request. The Fig. 4 reflects present of different tasks completed status.

Fig. 4 shows the execution status of various user tasks. Although the cluster nodes are formally busy, the analogue of the metric known as efficiency in such a cluster is quite low, which clearly indicates the need to find ways to improve real productivity. Explanation of why the user task "survived" or why it was not completed within the time specified by the scheduler can speed up the SCC performance, user's understanding of the specifics of supercomputer platform and can be used to predicate duration time which is needed to successfully complete all of user tasks. Obviously, the predictive ability for the dispatcher is an exo-intelligent function that increases the likelihood of successful completion of an applied task, as well as "preparing" the cluster's computational resources for processing new tasks. There are several well-known models used in survival analysis. First, we should mention the nonparametric Kaplan−Meier approach for estimating survival functions, which describe the properties of individual components that influence the assessment of the quality of functioning and the risks of changing the behavior of complex systems. But survival functions estimator gives only the average view of tasks features and does not take into account the individual parameters $\mathbf{x}$ of each task. Therefore, based only on survival functions it is difficult to evaluate how the specific task features impact on real performance of
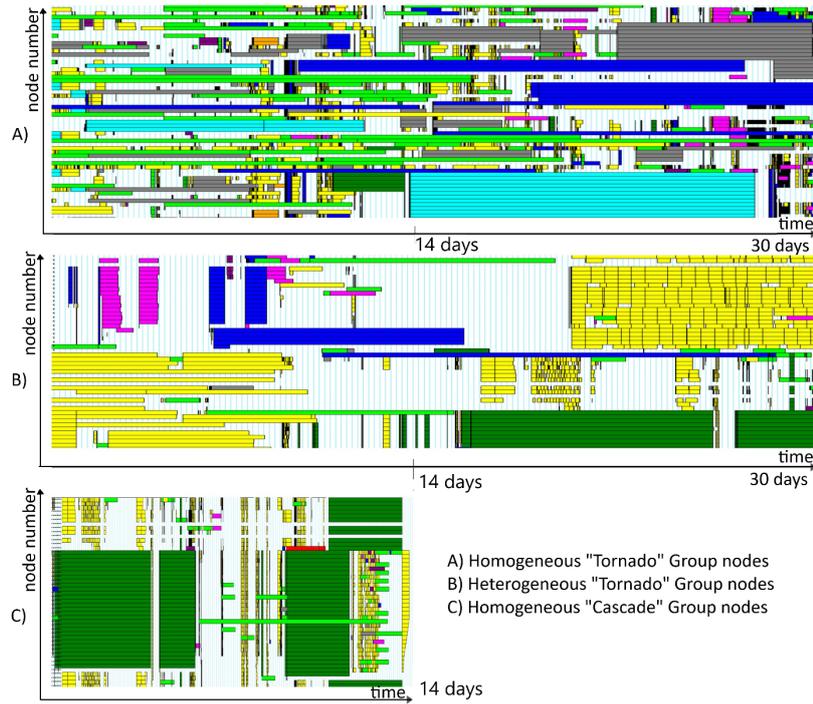
Fig. 3. Uneven loading of cluster nodes

A) Homogeneous "Tornado" Group nodes
B) Heterogeneous "Tornado" Group nodes
C) Homogeneous "Cascade" Group nodes
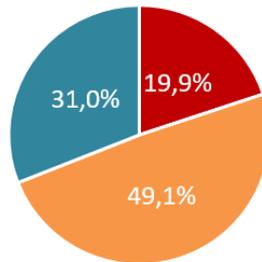
Fig. 4. Completed status of different tasks: orange sector (49,1%) − runtime errors;
blue sector (31%) − task completed successfully; red (19,9%) − tasks removed by scheduler

SCC. The approach which allows to estimate the survival and cumulative hazard feature depending the vector **x** parameter of each task is based on Cox model that is defined as

$$h\left(t|\mathbf{x}\right) = h_0\left(t\right)\exp^{\left(\mathbf{x}\mathbf{b}^T\right)},$$ (1)

where $h_0\left(t\right)$ is an arbitrary baseline hazard function and $\mathbf{b} = \left(b_1, \ldots, b_m\right)$ is an unknown vector of regression coefficients or model parameter.

As follows from Fig. 5, to solve the problem of optimizing the structure of the user tasks queue, we need to use machine learning scheduler system that is implemented to predict the execution time period, taking into account the current load of the SCC nodes and previous experience. Intelligent scheduler can identify:

- spread of the estimated execution time value,
- imbalance of tasks across the ranges of duration time to which they belong,
- insufficient amount of information in the available factors.

Uniform loading of supercomputer nodes

Homogeneous "**Cascade**" Group
• CPU 81 node, 48 cores each
Homogeneous "**Tornado**" Group
• CPU 668 nodes, 56 cores each
Heterogeneous "**Tornado**" Group
• CPU 668 nodes, 56 cores each
• GPU 56 nodes

Cluster nodes load map

User task execution request

Output data

scheduler

User' tasks queue

user **programs**, input & output data, startup program **script**

users

script after analysis

execution dataset

frontier intelligent scheduler system:
analyzing parameters, request to use accelerators, tasks classification and results interpreting

Administrator:
resource allocation policy

Fig. 5. "Less Moore, more brain": Polytechnic SSC − frontier intelligent platform

Distribution of tasks by real-time intervals

To 3 minutes

28.1%

More 3 days

3.0%

1-3 days

4.8%

Successfully
Error

3-10 minutes

25.1%

23.1%

1 hour - 1 day

15.9%

10 minutes - 1 hour
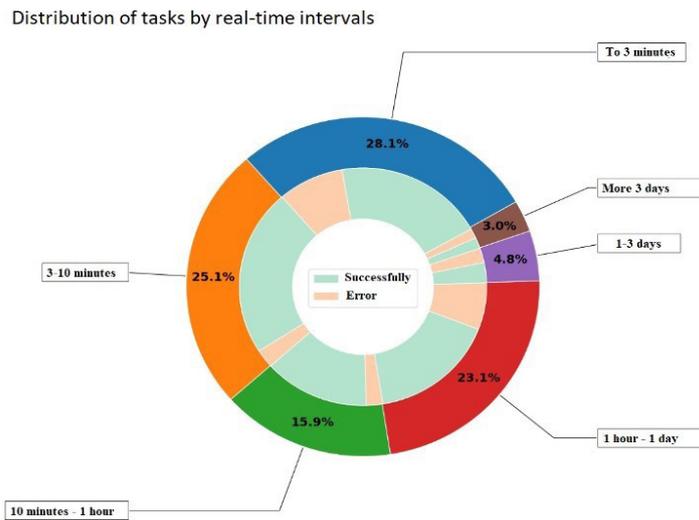
Fig. 6. The distribution of tasks in accordance with their completion time

The survival function analysis processes depicted in Fig. 6 shows that there are no real obstacles to the cluster node load being evenly distributed, because more than 89% of all tasks with actual duration between 0 and 10 minutes indicate an approximate execution time between 1 hour and 15 days. The distribution of tasks in accordance with their completion time is shown in Fig. 6. Therefore, users significantly overestimate the time it takes to complete applied tasks. User's behavior as well as tasks feature strongly affect task execution time and should be considered as a main factor of machine learning scheduler system.

It should be noted that the implementation of the intelligent component does not affect the work of the scheduler Slurm. The module for assessing the time of successful task execution is built in between the user and the scheduler and adjusts the parameters of the launched task in accordance with the forecast of the execution time. This ensures a denser queue load, reduces the average waiting time and increases the probability of successful completion of the computing task in the specified time.

Survival function in SCC real performance analysis

We represent a dataset D of survival analysis as a set of triplets of the form $\left(\mathbf{x}_i, \delta_i, T_i \right)$, where $\mathbf{x}_i = \left(x_{i1}, \ldots, x_{im} \right)$ is the feature vector which contains all available information about the i-th user task represented by m features; T_i is the i-th task completion time. In contrast to the standard regression analysis, proposed model includes additional component δ_i which is the indicator function, so that $\delta_i = 1$, if we observe a successful task completion, and $\delta_i = 0$, if the i-th task has not been successfully completed. In the first case $\left(\delta_i = 1 \right)$, time T_i corresponds to the time between the baseline time and the time of the successful task completion. This case corresponds to the uncensored observation. In the second case $\left(\delta_i = 0 \right)$, T_i is the observation time, i.e. the moment when the task is terminated due to several reasons, and we have the censored observation. The aim of survival analysis is to predict the completion time of a new task characterized by the feature vector $\mathbf{x}$ by using the training dataset consisting of n examples $\left(\mathbf{x}_i, \delta_i, T_i \right)$, $i = 1, \ldots, n$. Basic functions of survival analysis are the survival and hazard functions. The survival function, denoted as $S\left(t | \mathbf{x} \right)$, reflects probability that the task $\mathbf{x}$ is not completed up to the time t, so $S\left(t | \mathbf{x} \right) = \Pr\left\{ T > t | \mathbf{x} \right\}$. The hazard function $h\left(t | \mathbf{x} \right)$ is the rate of the task completion at time t given that no tasks completed before time t. It can be written as follows:

$$h\left(t | \mathbf{x} \right) = \lim_{\Delta t \to 0} \frac{\Pr\left\{ t \leq T \leq t + \Delta t | T \geq t | \mathbf{x} \right\}}{\Delta t} = \frac{f\left(t | \mathbf{x} \right)}{S\left(t | \mathbf{x} \right)}, \tag{2}$$

where $f\left(t | \mathbf{x} \right)$ is the density function of the task completion.

The hazard function can also be expressed though the survival function as follows:

$$h\left(t | \mathbf{x} \right) = \frac{d}{dt} \ln S\left(t | \mathbf{x} \right). \tag{3}$$

Hence, we can express the survival function through the cumulative hazard function $H\left(t \right)$ as follows:

$$S\left(t | \mathbf{x} \right) = \exp\left(-H\left(t | \mathbf{x} \right) \right). \tag{4}$$

It can be seen from above examples that the functions depend on the vector $\mathbf{x}$.

The survival function $S\left(t | \mathbf{x} \right)$ is computed as follows:

$$S\left(t | \mathbf{x} \right) = \left(S_0\left(t \right) \right) \exp\left(\mathbf{x} \mathbf{b}^T \right). \tag{5}$$

Here $S_0\left(t \right)$ is the baseline survival function. It is important to point out that $S_0\left(t \right)$ as well as $h_0\left(t \right)$ do not depend on $\mathbf{x}$ and are estimated using the Kaplan–Meier estimator. The main peculiarity of the Cox model is the linear combination of features. On the one hand, it simplifies the model and allows us to use it in the interpretation of the model predictions. On the other hand, it restricts the Cox model use because real datasets usually have a more complex structure. Various modifications have been proposed to generalize the Cox model by replacing the linear relationship with some non-linear functions, for example, with neural networks or the support vector machine. A goal of the machine learning algorithm is to predict the probabilistic characteristics of the task completion, including the survival function and the task completion time. In the case when the dataset of examples is restricted, it is possible to apply model of random survival forest as one of the efficient methods dealing with the limited number of training data. The random survival forest consists of Q survival trees define by vector $\mathbf{x}$ that consists of m features which include the most important for scheduler data: UserID (the ID of a user), GroupID (the ID of the group on behalf of which the task was queued), NumNodes (the number of nodes requested or allocated for the task), NCPUs (the total number of processors allocated to the task), NumTasks (the total

17

number of subtasks in the task), CPUs/Task (the ratio of the total number of processors to the number of subtasks), ReqB:S:C:T (the number of different hardware components requested for the task), Socks/Node (the desired ratio of the number of sockets to the number of compute nodes), NtasksPerN:B:S:C (the number of subtasks required to run on a specific number of hardware components), CoreSpec (the number of cores reserved), MinCPUsNode (the minimum ratio of the number of processors to the number of nodes), MinMemoryNode (the minimum ratio of memory in MB to the number of nodes), JobID (the task identification number), Priority (he task priority determined by the SLURM scheduler), etc. It should be noted that the completion time T_i of different tasks is changed in a large interval of time. It can be seen from this distribution that the number of "long" tasks is rather small in comparison with tasks completed in a short time. This causes a difficulty for random survival forest because survival trees are mainly trained on the "short" task and do not take into account the "long" duration. In order to overcome this problem, we propose to cluster all training dataset into K groups which are separated by using the completion time T as well as the feature vector $\mathbf{x}$. The completion time is more important factor in comparison with the feature vector because it determines the distribution of tasks. Therefore, we propose to introduce weights w_0 and w_1 of the completion time as well as the feature vector $\mathbf{x}$, respectively, so that $w_0 + w_1 = 1$, $w_0 > w_1$. In such weighted K-means clustering procedure, the distance between the centroid $\mathbf{c}$ of points from a cluster C_k and the vector $\mathbf{x}$ can be easily computed. One of the important problems to optimize the real performance is to explain why the user task is characterized by the obtained survival function or the expected time to the task completion. This problem can be referred to the XAI direction. Methods of XAI try to answer the question, which feature of an example significantly affect a prediction of a machine learning model. Most methods are explained locally, that is, they explain a prediction of a single example, and assume that the analyzed machine learning model is a black box which means that we know only its input and output data. A lot of explanation methods are based on local approximating the unknown prediction function at a point by means of the linear function of features because coefficients of the function can be regarded as quantitative impacts on the prediction.

In this case, the idea of the task description in terms of natural language requires the development of more complex and efficient structures based on the attention mechanism and transformers now attract strong attention. It should be noted that transformers proposed to solve the survival analysis problems can be solid platform for this. However, as a rule, transformers do not take the peculiarities and context of the problems, which are solved within completion time optimization problem. Therefore, productive approach is to combine the random survival forests and the transformer [12]. However, this approach cannot be directly used in survival analysis of user tasks in SSC. New approaches are needed to develop an efficient multi-modal transformer-based system based on estimation of system survives function (Fig. 7).

Perspectives

Obviously, multimodal transformers outline of the description of different applied tasks can be added to the machine learning ecosystem so that the user can conduct a dialogue with the supercomputer in terms of meaningful queries in the context of specific tasks features, using the capabilities of a pre-trained transformer (Fig. 8), which generates the source code of the task in response to the meaningful request. In such system user can analyze the accuracy of the formulated queries, conducting a meaningful interaction with a supercomputer, that inevitably improves his qualifications and task understanding. This approach has two advantages. First, it allows making the clustering procedure more flexible and enhance real cluster performance. Second, it allows supplementing the explanation procedure and developing multimodal transformer to solve the survival problems in context of scheduler efficiency. Moreover, the idea of the task description in terms of natural language requires the development of more complex and efficient SCC architecture based on machine learning mechanism. It is important to note that one of the perspective directions is to adapt the trained system to changes in the supercomputer structure, for example, to new additional computer blocks, which can be supplemented in runtime mode.
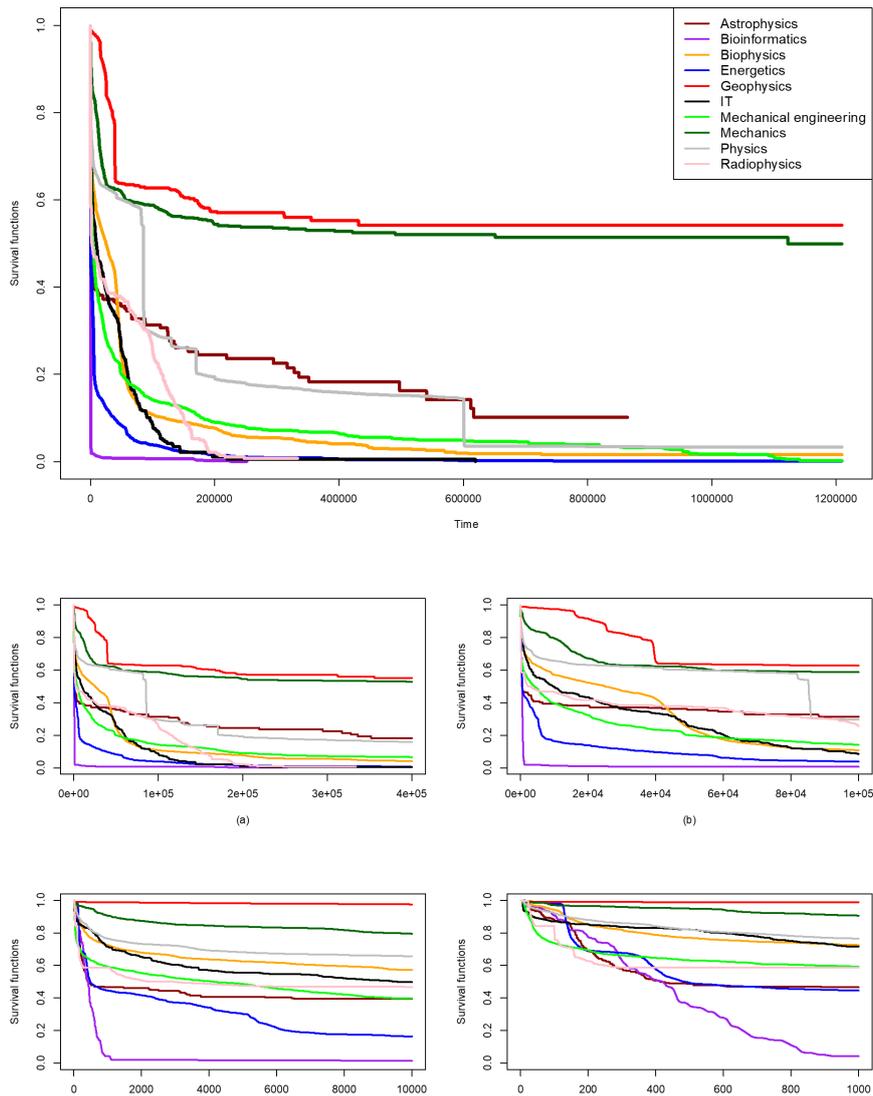
Fig. 7. Experimental survival task distribution functions: (a) the area of task execution times
not exceeding 4×10^5 sec.; (b) not exceeding 1×10^5 sec.; (c) not exceeding 1×10^4 sec.; (d) not exceeding 1×10^3 sec.

The structure of a hierarchical hybrid HPC platform with an exo-intelligent subsystem for predicting the execution time of applied tasks and a built-in multimodal transformer that generates source code for new programs based on user requests is shown in Fig. 8. The results of applying machine learning methods to a model that estimates the execution time of an applied task and is used to improve the efficiency of the dispatcher are shown in Fig. 9.

From the Fig. 9 follows that the number of successfully completed applied tasks in Hierarchical Architecture of frontier Polytechnic SCC increased compared to the result of the system without subsystem estimating the time required to successfully compute user tasks (see Fig. 4).

Conclusions

The article discusses new way to increase the real performance of hybrid HPC platforms operating in the mode of shared-use computational resources. The conceptual difference of the proposed approach can be metaphorically expressed as "Less Moore, more brain." The use of machine learning approach shifts the focus of methods of increasing the performance of HPC platforms from hardware components to more complex exo-intelligent solutions that use inductive (internal) and conceptual (external) data
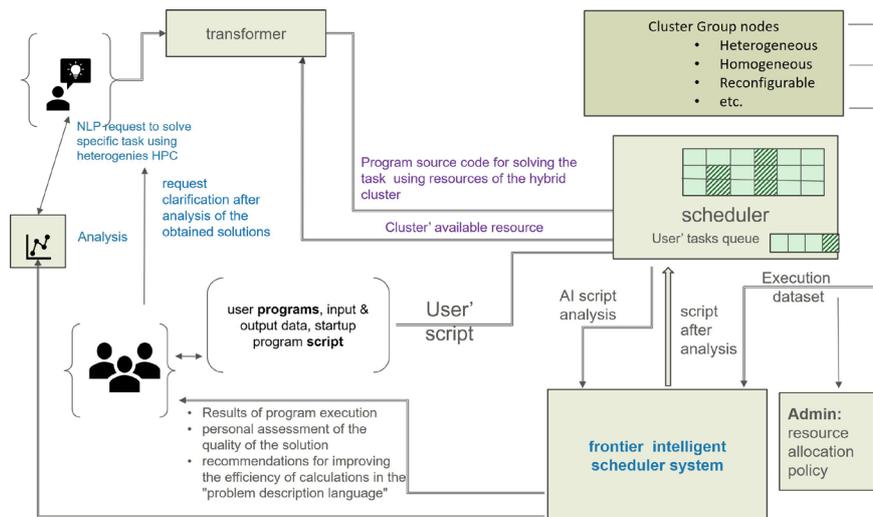
Fig. 8. Hierarchical architecture of frontier SCC "Polytechnic"
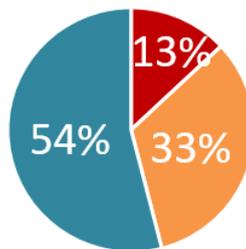
Fig. 9. Status of mechanics tasks: blue sector (54%) − task completed successfully;
orange sector (33%) − runtime errors; red (13%) − tasks removed by scheduler

to implement machine learning methods for the purpose of optimally distributing available hardware resources between different classes of user applications. The proposed three-level architecture of hybrid computing platforms opens up new opportunities both for efficient scaling of user program execution processes, and for reification of descriptions of new machine learning algorithms by setting up an appropriate survival model, as well as interpreting the results of calculation based on analysis of statistical information.

REFERENCES

1. **Alaa A., van der Schaar M.** Limits of estimating heterogeneous treatment effects: Guidelines for practical algorithm design. Proceedings of the 35th International Conference on Machine Learning, PMLR 80, 2018, pp. 129−138.

2. **Hu S., Fridgeirsson E.A., van Wingen G., Welling M.** Transformer-based deep survival analysis. Proceedings of Machine Learning research, PMLR 1, 2021, pp. 1−17.

3. **Konstantinov A.V., Utkin L.V., Lukashin A.A., Muliukha V.A.** Neural attention forests: Transformer-based forest improvement. Proceedings of the Seventh International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'23), 2023, Vol. 776, pp. 158−167. DOI: 10.1007/978-3-031-43789-2_14

4. **Lu J., Behbood V., Hao P., Zuo H., Xue S., Zhang G.** Transfer learning using computational intelligence: A survey. Knowledge-Based Systems, 2015, Vol. 80, pp. 14−23. DOI: 10.1016/j.knosys.2015.01.010

5. **Pachón-García C., Hernández-Pérez C., Delicado P., Vilaplana V.** SurvLIMEpy: A python package implementing SurvLIME. Expert Systems with Applications, 2024, Vol. 237, part C, article no. 121620. DOI: 10.1016/j.eswa.2023.121620

6. **Pölsterl, S., Navab N., Katouzian A.** An efficient training algorithm for kernel survival support vector machines, 2016. DOI: 10.48550/arXiv.1611.07054

7. **Ribeiro M.T., Singh S., Guestrin C.** "Why should I trust You?" Explaining the predictions of any classifier, 2016. DOI: 10.48550/arXiv.1602.04938

8. **Utkin L.V., Satyukov E.D., Konstantinov A.V.** SurvNAM: The machine learning survival model explanation. Neural Networks, 2022, Vol. 147, pp. 81−102. DOI: 10.1016/j.neunet.2021.12.015

9. **Wang P., Li Y., Reddy C.** Machine learning for survival analysis: A survey. ACM Computing Surveys (CSUR), 2019, Vol. 51, no. 6, pp. 1−36. DOI: 10.1145/3214306

10. **Wang Z., Sun J.** SurvTRACE: Transformers for survival analysis with competing events. Proceedings of the 13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, 2022, pp. 1−9. DOI: 10.1145/3535508.3545521

11. **Kanamori Y., Yoo S.-M.** Quantum computing: Principles and applications. Journal of International Technology and Information Management, 2020, Vol. 29, no. 2, pp. 43−71. DOI: 10.58729/1941-6679.1410

12. **Konstantinov A.V., Utkin L.V.** Interpretable machine learning with an ensemble of gradient boosting machines. Knowledge-Based Systems, 2021, Vol. 222, article no. 106993. DOI: 10.1016/j.knosys.2021.106993

13. **Tikhonov A.N.** Metody resheniya nekorrektnykh zadach [Methods for solving incorrect problems], Moscow: Nauka, 1979.

14. **Kovalev M.S., Utkin L.V., Kasimov E.M.** SurvLIME: A method for explaining machine learning survival models. Knowledge-Based Systems, 2020, Vol. 203, article no. 106164. DOI: 10.1016/j.knosys.2020.106164

15. **Cox D.R.** Regression models and life-tables. Journal of the Royal Statistical Society: Series B (Methodological). 1972, Vol. 34, no. 2, pp. 187−202. DOI: 10.1111/j.2517-6161.1972.tb00899.x

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Zaborovsky Vladimir S.
Заборовский Владимир Сергеевич
E-mail: vlad2tu@yandex.ru

Utkin Lev V.
Уткин Лев Владимирович
E-mail: lev.utkin@gmail.com
ORCID: https://orcid.org/0000-0002-5637-1420

Muliukha Vladimir A.
Мулюха Владимир Александрович
E-mail: vladimir.muliukha@spbstu.ru
ORCID: https://orcid.org/0000-0002-3583-7324

INTERPRETATION METHODS
FOR MACHINE LEARNING MODELS IN THE FRAMEWORK
OF SURVIVAL ANALYSIS WITH CENSORED DATA:
A BRIEF OVERVIEW

L.V. Utkin ✉ ⓘ , *A.V. Konstantinov* ⓘ , *D.Yu. Eremenko* ⓘ ,
V.S. Zaborovsky, V.A. Muliukha ⓘ

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

✉ lev.utkin@gmail.com

Abstract. Methods of interpretation, or explanation, of predictions are an integral part of modern black-box machine learning models. They have become widespread due to the need for the user to understand what the machine learning model is predicting. This is especially important for survival analysis models, as they are used in medicine, system reliability, safety, and also have features that make them difficult to explain and interpret. The paper discusses the main methods for interpreting survival models that deal with censored data and determine the characteristics of the time until a certain event. A feature of such models is that their predictions are presented not as a point value, but as a probabilistic function of time, for example, a survival function or a risk function. This requires the development of special interpretation methods. The most well-known methods SurvLIME, SurvLIME-KS, SurvNAM and SurvBeX, SurvSHAP(t) are considered, which are based on the use of LIME and SHAP interpretation methods, the Cox model and its modifications, as well as the Beran estimator.

МЕТОДЫ ИНТЕРПРЕТАЦИИ МОДЕЛЕЙ МАШИННОГО ОБУЧЕНИЯ В РАМКАХ АНАЛИЗА ВЫЖИВАЕМОСТИ ПРИ ЦЕНЗУРИРОВАННЫХ ДАННЫХ: КРАТКИЙ ОБЗОР

Л.В. Уткин ✉ (iD) , *А.В. Константинов* (iD) , *Д.Ю. Еременко* (iD) ,
В.С. Заборовский, В.А. Мулюха (iD)

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

✉ lev.utkin@gmail.com

Аннотация. Методы интерпретации, или объяснения, предсказаний являются неотъемлемой частью современных моделей машинного обучения типа «черный ящик». Они получили широкое распространение, что обусловлено необходимостью понимания пользователем того, что предсказывает модель машинного обучения. Это особенно относится к моделям анализа выживаемости, так как они используются в медицине, надежности, безопасности, а также имеют особенности, которые усложняют их объяснение и интерпретацию. В работе рассматриваются основные методы интерпретации моделей выживаемости, которые оперируют с цензурированными данными и определяют характеристики времени до определенного события. Особенностью таких моделей является то, что их предсказания представляются не в виде некоторого точечного значения, а в виде вероятностной функции времени, например, функции выживаемости или функции риска. Это требует необходимости разработки специальных методов интерпретации. Рассмотрены наиболее известные методы SurvLIME, SurvLIME-KS, SurvNAM и SurvBeX, SurvSHAP(t), которые основаны на использовании методов интерпретации предсказаний LIME и SHAP, модели Кокса и ее модификации, а также оценки Берана.

Ключевые слова: машинное обучение, модель выживаемости, объяснимый искусственный интеллект, цензурированные данные, модель Кокса, оценка Берана

Introduction

The increasing importance of machine learning models, particularly deep learning models, and their widespread use in various applications has led to the problem of prediction explanation and interpretation. The development and implementation of intelligent systems based on machine learning models for solving various application tasks is currently one of the most rapidly growing areas of the artificial intelligence (AI) applications, and it leads to the problem of explaining or interpreting predictions provided by the models. This problem stems from the fact that, despite the importance of the AI applications in many real tasks, there are several obstacles to further implementation of AI especially in such areas as medicine, system reliability, etc. because the corresponding machine learning models are often perceived as "black

boxes" meaning that the inner workings of these models are often completely unknown. As a result, it is difficult to understand and explain, why the models provide a particular prediction for a particular input instance. The importance of the explanation problem prompts the development of corresponding additional models aimed at explaining and interpreting the obtained predictions. The explanation of a model prediction means to find features of the explained instance that most influence the prediction. In other words, the meta-model should suggest, which features of the explained instance cause the corresponding prediction.

It should be noted that many explanation methods have been proposed recently, which are discussed in several review articles [1–4]. If we consider explanation methods in terms of simultaneously explained instance numbers, all methods can be divided into two large groups. The first group consists of *local* methods, which try to explain predictions obtained for a single test instance or for a small set of instances. The second group contains the *global* methods which explain predictions of the entire dataset on average. The first group of methods is more important, because many applications require to explain a certain instance, for example, a doctor prefers to understand a diagnosis, which is predicted by a machine learning model for a certain patient, but not for all the patients in a hospital.

Most methods of the local explanation are based on training a special meta-model, which is self-explainable, and it approximates the black-box model prediction function at a point, which corresponds to the explained instance. One of the ideas behind several explanation methods is to approximate with a linear model, because the coefficients of the linear model can be interpreted as quantitative measures of the feature importance. Following this idea, the well-known explanation model, called LIME (Local Interpretable Model-Agnostic Explanation), has been proposed by Ribeiro et al. [5]. According to LIME, a linear approximation of a non-linear prediction function of the black-box model at an instance is built. It is carried out by generating synthetic instances around the explained instance with such weights that each weight depends on a distance between the explained instance and the generated synthetic instance. Another well-known explanation method is so-called SHAP method (SHapley Additive exPlanations) proposed in [6, 7]. SHAP is based on the game-theoretic Shapley values, which can be regarded as the contributions of features into the black-box model prediction. Applications of SHAP meet two important difficulties. First, its complexity rapidly increases with the number of features. Second, it uses subsets of features as inputs for the black-box model, which must be added by some values of removed features that are not strongly defined.

To improve the linear explanation models and to overcome weakness of the linear approximations, more complex explanation models have been proposed. They are based on sums of the feature shape functions, which form the Generalized Additive Model (GAM) introduced by Hastie and Tibshirani in [8]. The GAM motivated to develop several interesting explanation models, including the Neural Additive Model (NAM) introduced by Agarwal et al. in [9], a weighted sum of separate gradient boosting machines (GBMs) presented in [10].

It is important to point out that the aforementioned models and their modifications have been developed to deal with various types of data. However, there is a class of datasets, which consider times to the events of interest.

Machine learning models, trained on data characterizing the time to occurrence of certain events of some objects depending on the structure of these objects, are becoming increasingly widespread [11]. This is due to their use in a variety of areas, for example, in the system reliability, when events of system failure are considered, in medicine, when the event is the recovery or death of a patient. One of the crucial features of many models is that events associated with some objects may not be observed, but only the last moment of observation is recorded, assuming that the event will occur in the future, but we do not know when. Such data are called censored, and they contain significantly less information about the object than uncensored data, for which the time of the event is known. However, censored data can also be used in machine learning models called survival models.

24

One of the well-known survival models is the Cox proportional hazard model [12]. According to the model, the covariates (features) of an object are linearly connected. On the one hand, this feature can be viewed as a limitation of the model since the relationship between features may be significantly nonlinear in some cases. To account for various relationships between features, a large number of survival models have been developed recently, for example, random survival forests, deep survival neural networks, modifications of the support vector machine and others [11]. At the same time, each model is a black box, that is, only inputs and the corresponding outputs (predictions) are known, but it is not known how a prediction is obtained, which features of the object influence the prediction of the model. However, an important feature of many survival models is that their predictions are presented in the form of the survival function (SF) or the cumulative hazard function (CHF). This fact significantly complicates the solution of the explanation problem and requires special approaches to its solution.

We provide a brief overview of the most important explanation methods within survival models.

Concepts of survival analysis

Let us consider the training set D consisting of n triplets (x_i, T_i, δ_i), $i = 1, ..., n$, where each triplet characterizes an object, $x_i \in \mathbf{R}^m$ is the feature vector; T_i is the event time of the i-th object; δ_i is the indicator of the event, $\delta_i = 1$, if the event is observed (uncensored observation), $\delta_i = 0$, if the event is not observed (censored observation). We aim to estimate the event time T on the basis of D for a new object having features x.

Important concepts in survival analysis are SFs and CHFs [11]. The SF $S(t|x)$ is defined as the probability of surviving the object x up to time t. Another concept is the CHF $H(t|x)$, which is expressed through the SF as $H(t|x) = -\ln(S(t|x))$.

One of the base survival models, which can be regarded as the basis for several explanation methods, is the well-known Cox proportional hazards model [12]. According to the Cox model, the conditional CHF is determined as follows [12]:

$$H(t|x,b) = H_0(t) \cdot \exp(b^T x),$$

where $H_0(t)$ is the baseline CHF, which can be estimated by using the Nelson–Aalen or Kaplan–Meier estimators; b is the vector of the model parameters.

It can be seen from the expression for the Cox CHF that the linear relationship assumption between covariates and the log-risk of an event is accepted in the model. This is a very important property of the Cox model, which allows us to approximate an arbitrary CHF produced as an output of the black-box survival model by the Cox model and, therefore, to construct methods explaining survival models.

The Cox model is popular in many real tasks. However, its linear assumption significantly restricts its application, because many real survival datasets violate this assumption. GAM incorporated into the Cox model instead of the linear expression partially relaxes this assumption. This representation will be used in one of the explanation methods. Another shortcoming of the Cox model is that it does not take into account the positional relationship of feature vectors. This difficulty can be resolved by using the Beran estimator [14]. According to the Beran estimator, a SF can be estimated as follows:

$$S_B(t|x) = \prod_{T_i \le t} \left\{ 1 - \frac{W(x, x_i)}{1 - \sum_{j=1}^{i-1} W(x, x_j)} \right\}^{\delta_i}.$$

Here the weight $W(x, x_i)$ characterizes how the vectors x, x_i are close to each other. It should be noted that the kernel function measures how similar any two feature vectors are. Therefore, the weights

are nothing else, but the normalized kernels. For example, if the kernel is Gaussian, then the weight is defined through the softmax operation: $W\left(\boldsymbol{x},\boldsymbol{x}_i\right) = \mathrm{softmax}\left(\|\boldsymbol{x}-\boldsymbol{x}_i\|^2\big/\tau\right)$, where τ is a hyperparameter. It is interesting to note that the Kaplan–Meier estimator can be viewed as a special case of the Beran estimator, when all weights $W\left(\boldsymbol{x},\boldsymbol{x}_i\right)$ are identical and equal to $1/n$, where n is the number of instances in the dataset.

Explanation formal problem statements

Formally, the explanation problem is solved by means of training a meta-model or an explanation model that approximates the explainable black-box model in the vicinity of the example being explained and belongs to a set of "simple" models that are interpretable (linear models, decision trees). The explainable black-box model implements the function $f:\mathbf{R}^m \to \mathbf{R}^d$, for example, in classification $f\left(\boldsymbol{x}\right)$ is the probability (or indicator) that the feature vector $\boldsymbol{x}$ belongs to a certain class. A meta-model is a model $g \in G$, where G is a class of interpretable models, which is a solution to the optimization problem:

$$\min_{g \in G}\left\{L\left(f,g,\theta\right)+\Omega\left(g\right)\right\},$$

where $L\left(f,g,\theta\right)$ is the measure of how poorly g approximates f; θ is the parameter vector; $\Omega\left(g\right)$ is the regularization term.

One of the most popular interpreted functions is the linear function. This is due to the fact that the coefficients of a linear function precisely characterize the influence of each feature on the value of the function. In fact, the local explanation problem comes down to approximating the function $f\left(\boldsymbol{x}\right)$ of the black-box model by a linear function $g\left(\boldsymbol{x}\right)$ at instance $\boldsymbol{x}$. The well-known LIME method [5] and its modifications are based on this idea. In LIME, to construct the approximating function $g\left(\boldsymbol{x}\right)$, N instances (vectors $\boldsymbol{z}_i \in \mathbf{R}^m$) are generated in the vicinity of the instance $\boldsymbol{x}$. Using the black-box model, the prediction $y_i = f\left(\boldsymbol{z}_i\right)$ is computed for each generated point $\boldsymbol{z}_i$ and a new dataset of N points $\left(\boldsymbol{z}_i, y_i\right)$ is formed. The resulting dataset is used to construct the linear function $g\left(\boldsymbol{x}\right)$. LIME is used to explain the classification and regression models. However, its application to survival models meets some difficulties, since, firstly, survival models deal with censored data, for which the construction of a regression model differs from standard models. Secondly, the output of the survival model, that is y, is the SF rather than the point value, which also complicates the task of explanation, since the entire SF has to be interpreted.

Another explanation method is SHAP [6, 7]. According to this method, the i-th feature average contribution is estimated by the Shapley value:

$$\phi_i\left(f\right) = \phi_i = \sum_{S \subseteq N/\{i\}} \frac{|S|!\left(m-|S|-1\right)!}{m!}\Big[f\left(S \cup \{i\}\right)-f\left(S\right)\Big],$$

where $f\left(S\right)$ is a prediction of the black-box model under condition that a subset S of features is used as the corresponding input; m is the number of all features.

A controversial question in SHAP is how to represent or to fill features from the subset $\{1,\ldots,m\}/S$ to apply the black-box model. There are several approaches to partially solve this [14]. However, every approach has disadvantages and cannot be used in all cases.

Explanation methods in survival analysis

Due to the mentioned peculiarities of the survival machine learning models, the known explanation methods like LIME and SHAP to explain the corresponding predictions cannot be directly used. Therefore, every explanation method is based on applying a trick, which allows us to adapt it to LIME or SHAP.

The main idea behind the first group of explanation methods, called SurvLIME, SurvLIME-KS, SurvLIME-Inf, is to use the Cox model to approximate the black-box model in a local area around the explained object x. This idea stems from the fact that the Cox model assumes the linear relationship $b^T x$ of the object features or covariates. An important peculiarity of the Cox model is that functions of features and the time are separated. Hence, coefficients b in the Cox model can be regarded as measures how the features impact on predictions. However, we do not approximate a point prediction, but rather functions, such as CHFs. According to the SurvLIME method [15], synthetic instances (feature vectors) z_i are randomly generated around the explainable example, and for each synthetic vector the CHF is calculated using the black-box model. Therefore, we propose to consider the distance between logarithms of the CHFs $H(t|z_i)$ predicted by the black-box model and CHFs $H^{Cox}(t|z_i)$ computed by using the Cox model. The distance is determined as follows:

$$d(z_i) = \int_0^\infty \left(\ln H(t|z_i) - \ln H^{Cox}(t|z_i) \right)^2 dt.$$

Let $t_1 \le t_2 \le \cdots \le t_n$ be the distinct event times obtained from the set of training times $\{T_1, \ldots, T_n\}$, where $t_1 = \min_{k=1,\ldots,n} T_k$, $t_n = \max_{k=1,\ldots,n} T_k$. Substituting the corresponding expressions for CHFs into the above expression and using the fact that the CHFs are stepwise functions due to the finite number of the observed event times, we obtain after simplification:

$$d(z_i) = \sum_{j=0}^n \left(\ln H_j(z_i) - \ln H_{0,j}(z_i) - b^T z_i \right)^2 \tau_j,$$

where $\tau_j = t_{j+1} - t_j$; $H(z_i)$ is the CHF $H(t|z_i)$ in the interval $[t_j; t_{j+1}]$; $H_{0,j}(z_i)$ is the baseline CHF of the Cox model in the same interval of time.

Since the CHF in the Cox model is a function of unknown coefficients b, the optimization problem for calculating the coefficients is determined by the weighted average distance between the CHFs of the black-box model and the Cox model over all generated points z_i, $i = 1, \ldots, N$, taking into account weights w_i that are determined by the distances between point x and each point z_i. This implies that the loss function is of the form:

$$L(b) = \min_b \sum_{i=0}^N w_i d(z_i).$$

It is interesting to point out that the obtained optimization problem is convex, therefore, its solution does not meet any difficulties.

The general scheme of the method is shown in Fig. 1. If the distance between risk functions is calculated based on the quadratic norm L_2, then the optimization problem is reduced to quadratic programming, which makes it possible to find a solution (vector b) quite simply. For norms L_1 and L_∞ (SurvLIME-Inf [16]) it is shown that the corresponding optimization problems are reduced to linear.

To ensure the robustness of the explanation mode, the Kolmogorov–Smirnov bounds for the SF were introduced in [17]. The proposed method, called SurvLIME-KS, is an extension of the SurvLIME method and uses the results of this method, but under the assumption that instead of a single SF, a set of SFs is used. As a result, a maximin optimization problem for calculating the optimal vector b is constructed, where the maximum is determined over all SFs within the Kolmogorov–Smirnov bounds. Despite the seeming complexity of the optimization problem, it reduces to a finite set of quadratic or linear optimization problems whose solutions do not meet any difficulties.

SurvLIME provides explanation of the survival black-box model predictions. However, it is interesting to explain why predicted results of a survival model are uncertain or to answer the question: Which
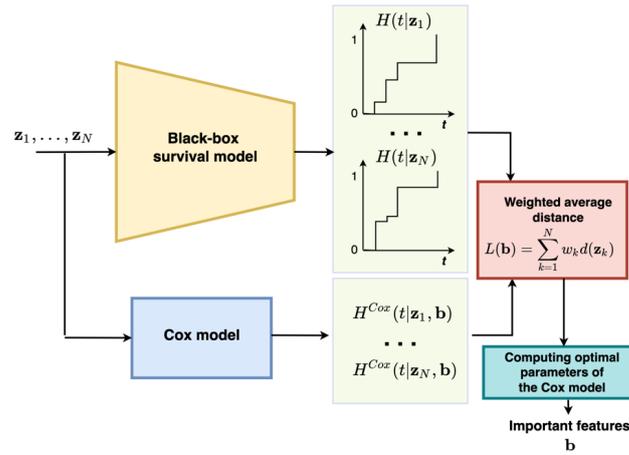
Fig. 1. SurvLIME structure [16]

features of an explained instance lead the black-box model prediction to be uncertain? Following the above question, a method for uncertainty interpretation of the black-box survival model predictions called UncSurvEx (Uncertainty Survival Explanation) has been proposed in [18]. The method like SurvLIME applies the Cox model to approximate the survival model. It is assumed that the most uncertain density function of the prediction is uniform at times $T_1, ..., T_n$. This is similar to the multi-class classification, where the prediction is the class probability distribution. The most uncertain prediction is when all probabilities are identical, and we cannot make decision about a class of an instance. According to UncSurvEx, an "certainty" measure $c(z_i)$ as the distance d between the actual SF $S(t|z_i)$ and the most uncertain "uniform" SF $S^u(t|z_i)$ is determined. In the same way, the "certainty" measure $c^{Cox}(z_i, \boldsymbol{b})$ is determined as the distance between the Cox SF $S^{Cox}(t|z_i, \boldsymbol{b})$ and the most uncertain SF $S^u(t|z_i)$. The weighted difference between $c(z_i)$ and $c^{Cox}(z_i, \boldsymbol{b})$ is minimized to get the optimal values $\boldsymbol{b}$, which show the most important features from the prediction uncertainty point of view.

Another interesting explanation method is a generalization of the NAM method [10] to the case of censored data, called SurvNAM [19]. The idea behind the method is similar to the SurvLIME method, but unlike the linear combination $\boldsymbol{b}^T\boldsymbol{x}$ of attributes adopted in SurvLIME in accordance with the Cox model, this combination in SurvNAM is replaced by a set of neural networks that calculates the functions $g_i(x_i)$ of features. The modified Cox model is of the form:

$$H(t|\boldsymbol{x}, \boldsymbol{g}) = H_0(t) \cdot \exp\left(g_1(x_1) + \cdots + g_m(x_m)\right),$$

where $\boldsymbol{g} = \left(g_1(x_1), ..., g_m(x_m)\right)$ is the vector of functions.

Neural networks implementing functions $g_i(x_i)$ are trained according to a loss function defined by the average distance between CHFs of the black-box model and the Cox model over all generated synthetic points. As a results, we obtain functions $g_i(x_i)$, called the shape functions. In order to explain a prediction by using the shape functions, the rate of change of each function has to be estimated. It characterizes the impact of the corresponding feature on the prediction. The rate of change can be simply estimated by computing the variance of $g_i(x_i)$.

The Cox model, even with the functions implemented by the neural network, requires the calculation of the baseline CHF or SF that are independent of features. A more powerful survival model is the Beran estimator. Therefore, a method called SurvBeX (Survival Beran eXplanation) [20] was developed that uses the Beran estimator instead of the Cox model. The main idea of the method is that in the Beran estimator the weight is defined as follows:
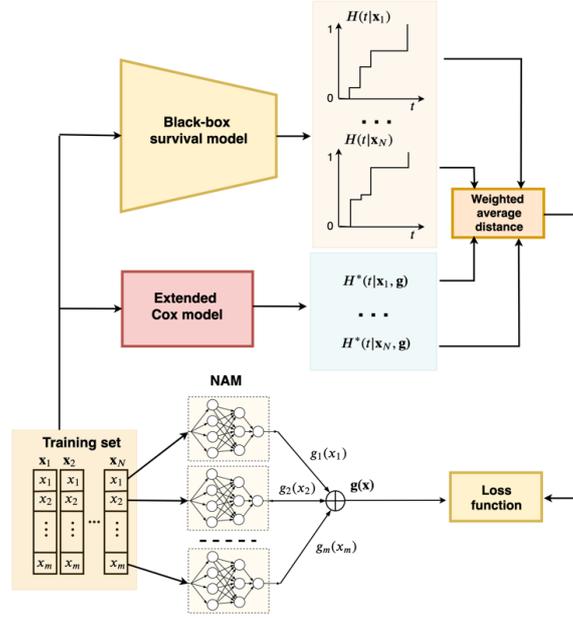
Fig. 2. SurvNAM structure [21]

$$W\left(\boldsymbol{x},\boldsymbol{x}_i,\boldsymbol{b}\right)=\mathrm{softmax}\left(\left\|b\odot\left(\boldsymbol{x}-\boldsymbol{x}_i\right)\right\|^2\right),$$

where vector $\boldsymbol{b}$ characterizes the influence of features on predictions, $\odot$ is the scalar product.

In general, SurvBeX uses the algorithm of the SurvLIME method with the generation of synthetic instances near the object being explained, but with the difference that the Beran estimator is used instead of the Cox model. In this case, we obtain a more complex optimization problem. However, numerous numerical experiments show that it can be solved quite simply by existing methods and provide significantly better interpretation results.

SurvBeX is a flexible method, which can be modified in several ways. First, we can change the kernel function. For example, if we replace the Gaussian kernel, used in the method, with the triangle kernel, then only a local area of instances around $\boldsymbol{x}$ will be included into consideration. Second, the set of training parameters can also be changed. If the number of training instances is rather large, then additional parameters may lead to a more accurate approximation of the black-box model prediction function.

So far, the explanation methods based on survival modifications of LIME and NAM have been considered. However, there is an explanation method, which extends SHAP to a case of survival analysis. The method is called SurvSHAP(t) [21], and it is based on SHAP with solid theoretical foundations and a broad adoption among machine learning practitioners. According to SurvSHAP(t), the Shapley values (contributions of the d-th feature) $\phi_d\left(\boldsymbol{x},t\right)$ are assigned for the explained instance $\boldsymbol{x}$ at a time moment t. Since the predicted SF of the black-box model is a step-wise function with changes at times $t_1\le t_1\le$ $\le\cdots\le t_n$ due to the finite number of observations, then values $\phi_d\left(\boldsymbol{x},t\right)$ are determined at all times, i.e., $\phi_d\left(\boldsymbol{x},t_1\right),\ldots,\phi_d\left(\boldsymbol{x},t_n\right)$. Values $\phi_d\left(\boldsymbol{x},t\right)$ are computed for every feature d and every time t_i in the standard way by taking points of the predicted SF $S\left(t|\boldsymbol{x}\right)$ at the point t_i as a point-valued prediction of the black-box model for the explained instance $\boldsymbol{x}$. The obtained time-dependent values $\phi_d\left(\boldsymbol{x},t\right)$ are aggregated to calculate the overall importance $\psi_d\left(\boldsymbol{x}\right)$ of the d-th feature as follows:

$$\psi_d\left(\boldsymbol{x}\right)=\int_0^{t_n}\left|\phi_d\left(\boldsymbol{x},t\right)\right|dt.$$

In contrast to SurvLIME, SurvNAM, and SurvBeX, where the meta-model is based on an assumption of the approximating model (the Cox model, the Beran estimator), an important advantage of SurvSHAP(t) is that it does not require to make any assumption about the meta-model. On the other hand, SurvSHAP(t) as a SHAP-based method inherits all problems of using the SHAP method, which include the problem of representing subsets of features as inputs and the problem of the complexity of computing the Shapley values.

Conclusion

We have briefly considered only the main explanation methods used in recent survival analysis. Due to the importance of survival analysis in many applications, including medicine, system reliability, safety, and predictive analytics, new survival explanation models are being developed and will be proposed in the near future. There are several areas of survival analysis, for which explanation methods have not been developed. For example, there are no effective explanation methods for competing risks. The development of such methods is an active area for further research. The same can be said for specific applications of survival analysis. For example, the development of explanation methods for models dealing with multi-modal data is another area for further research. New explanation methods that improve existing methods (SurvLIME, SurvNAM, SurvSHAP(t), etc.) can also be considered as areas for further research.

REFERENCES

1. **Adadi A., Berrada M.** Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). IEEE Access, 2018, Vol. 6, pp. 52138−52160. DOI: 10.1109/ACCESS.2018.2870052

2. **Burkart N., Huber M.F.** A survey on the explainability of supervised machine learning. Journal of Artificial Intelligence Research, 2021, Vol. 70, pp. 245−317. DOI: 10.48550/arXiv.2011.07876

3. **Carvalho D.V., Pereira E.M., Cardoso J.S.** Machine learning interpretability: A survey on methods and metrics. Electronics, 2019, Vol. 8, no. 8, article no. 832. DOI: 10.3390/electronics8080832

4. **Guidotti R.** Evaluating local explanation methods on ground truth. Artificial Intelligence, 2021, Vol. 291, article no. 103428. DOI: 10.1016/j.artint.2020.103428

5. **Ribeiro M.T., Singh S., Guestrin C.** "Why should I trust you?": Explaining the predictions of any classifier. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135−1144. DOI: 10.1145/2939672.2939778

6. **Lundberg S.M., Lee S.-I.** A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 2017, pp. 4765−4774. DOI: 10.48550/arXiv.1705.07874

7. **Štrumbelj E., Kononenko I.** An efficient explanation of individual classifications using game theory. Journal of Machine Learning Research, 2010, Vol. 11, pp. 1−18. DOI: 10.5555/1756006.1756007

8. **Hastie T.J., Tibshirani R.J.** Generalized additive models (Chapman & Hall/CRC Monographs on Statistics and Applied Probability), 1st ed. London: Chapman & Hall/CRC press, 1990. 352 p.

9. **Agarwal R., Melnick L., Frosst N., Zhang X., Lengerich B., Caruana R., Hinton G.** Neural additive models: Interpretable machine learning with neural nets. 35th Conference on Neural Information Processing Systems (NeurIPS 2021), 2021, Vol. 34, pp. 4699−4711. DOI: 10.48550/arXiv.2004.13912

10. **Konstantinov A.V., Utkin L.V.** Interpretable machine learning with an ensemble of gradient boosting machines. Knowledge-Based Systems. 2021, Vol. 222, article no. 106993. DOI: 10.1016/j.knosys.2021.106993

11. **Wang P., Li Y., Reddy C.K.** Machine learning for survival analysis: A survey. ACM Computing Surveys. 2019, Vol. 51, pp. 1−36. DOI: 10.48550/arXiv.1708.04649

12. **Cox D.R.** Regression models and life-tables. Journal of the Royal Statistical Society: Series B (Methodological). 1972, Vol. 34, no. 2, pp. 187−202. DOI: 10.1111/j.2517-6161.1972.tb00899.x

13. **Beran R.** Nonparametric regression with randomly censored survival data, 1981.

14. **Covert I., Lee S.-I.** Improving KernelSHAP: Practical Shapley Value Estimation via Linear Regression. International Conference on Artificial Intelligence and Statistics, 2021, pp. 3457–3465. DOI: 10.48550/arXiv.2012.01536

15. **Kovalev M.S., Utkin L.V., Kasimov E.M.** SurvLIME: A method for explaining machine learning survival models. Knowledge-Based Systems. 2020, Vol. 203, article no. 106164. DOI: 10.1016/j.knosys.2020.106164

16. **Utkin L.V., Kovalev M.S., Kasimov E.M.** An explanation method for black-box machine learning survival models using the Chebyshev distance. Artificial Intelligence and Natural Language (AINL 2020). 2020, Vol. 1292, pp. 62–74. DOI: 10.1007/978-3-030-59082-6_5

17. **Kovalev M.S., Utkin L.V.** A robust algorithm for explaining unreliable machine learning survival models using the Kolmogorov–Smirnov bounds. Neural Networks. 2020, Vol. 132, pp. 1–18. DOI: 10.1016/j.neunet.2020.08.007

18. **Utkin L.V., Zaborovsky V.S., Kovalev M.S., Konstantinov A.V., Politaeva N.A., Lukashin A.A.** Uncertainty interpretation of the machine learning survival model predictions. IEEE Access. 2021, Vol. 9, pp. 120158–120175. DOI: 10.1109/ACCESS.2021.3108341

19. **Utkin L.V., Satyukov E.D., Konstantinov A.V.** SurvNAM: The machine learning survival model explanation. Neural Networks. 2022, Vol. 147, pp. 81–102. DOI: 10.1016/j.neunet.2021.12.015

20. **Utkin L.V., Eremenko D.Y., Konstantinov A.V.** SurvBeX: An explanation method of the machine learning survival models based on the Beran estimator. 2023. DOI: 10.48550/arXiv.2308.03730

21. **Krzyziński M., Spytek M., Baniecki H., Biecek P.** SurvSHAP(t): Time-dependent explanations of machine learning survival models. Knowledge-Based Systems, 2023, Vol. 262, article no. 110234. DOI: 10.1016/j.knosys.2022.110234

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Utkin Lev V.
Уткин Лев Владимирович
E-mail: lev.utkin@gmail.com
ORCID: https://orcid.org/0000-0002-5637-1420

Konstantinov Andrei V.
Константинов Андрей Владимирович
E-mail: andrue.konst@gmail.com
ORCID: https://orcid.org/0000-0002-1542-6480

Eremenko Danila Y.
Еременко Данила Юрьевич
E-mail: eremenko.dyu@edu.spbstu.ru
ORCID: https://orcid.org/0000-0002-1115-7543

Zaborovsky Vladimir S.
Заборовский Владимир Сергеевич
E-mail: vlad2tu@yandex.ru

Muliukha Vladimir A.
Мулюха Владимир Александрович
E-mail: vladimir.muliukha@spbstu.ru
ORCID: https://orcid.org/0000-0002-3583-7324

SUPERCOMPUTER RESOURCES MANAGEMENT
USING MACHINE LEARNING METHODS UNDER CONSTRAINTS

V.A. Muliukha ✉ , *A.V. Vostrov,*
D.E. Motorin, V.V. Glazunov

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

✉ vladimir.muliukha@spbstu.ru

Abstract. Artificial intelligence and machine learning technologies are among the most promising in the field of computer science. They make it possible to obtain solutions to problems that until recently were the exclusive prerogative of humans. However, when solving practical problems, it is necessary to implement machine learning models taking into account the restrictions on available resources. Such resources can be both computational and temporary (i.e. the problem must be solved in a certain time and using certain hardware, most often it is about various mobile platforms), and informational, when it comes to small, censored, incomplete or noisy data. The paper examines machine learning methods used to solve practical problems in application areas, such as comparing the shape of three-dimensional objects and intellectualizing resource dispatching, within the framework of the concept of "Supercomputer for AI and AI for a Supercomputer". In the field of solving problems with limited data volume, a method is proposed that allows training a multilayer neural network using an ultra-small training sample to solve the problem of quantitatively assessing the proximity of the shape of arbitrary three-dimensional objects. In the field of applying machine learning models with limited resources, a method has been developed that ensures asynchronous operation of the machine learning model and the executable process, which allows for the effective use of machine learning methods under constraints.

УПРАВЛЕНИЕ ВЫЧИСЛИТЕЛЬНЫМИ РЕСУРСАМИ СУПЕРКОМПЬЮТЕРОВ МЕТОДАМИ МАШИННОГО ОБУЧЕНИЯ В УСЛОВИЯХ ОГРАНИЧЕНИЙ

В.А. Мулюха ✉ , А.В. Востров,
Д.Е. Моторин, В.В. Глазунов

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

✉ vladimir.muliukha@spbstu.ru

Аннотация. Технологии искусственного интеллекта и машинного обучения являются одними из самых перспективных в области компьютерных наук. Они позволяют получить решение задач, которые до недавнего времени были исключительно прерогативой человека. Однако при решении практических задач приходится реализовывать модели машинного обучения с учетом ограничений на доступные ресурсы, при этом, ресурсы могут быть как вычислительные и временные (т.е. задача должна быть решена за определенное время и с использованием определенного аппаратного обеспечения, чаще всего речь идет о различных мобильных платформах), так и информационные, когда речь идет о малых, цензурированных, неполных или зашумленных данных. В работе рассматриваются методы машинного обучения, используемые для решения практических задач в прикладных областях, таких как сравнение формы трехмерных объектов и интеллектуализация диспетчеризации ресурсов, в рамках концепции «Суперкомпьютер для ИИ и ИИ для суперкомпьютера». В области решения задач при наличии ограничений на объем данных предложен метод, который позволяет осуществить обучение многослойной нейронной сети с использованием сверхмалой обучающей выборки, для решения задачи количественной оценки близости формы произвольных трехмерных объектов. В области применения моделей машинного обучения при наличии ограничений на используемые ресурсы разработан метод, обеспечивающий асинхронную работу модели машинного обучения и исполняемого процесса, что позволяет эффективно использовать методы машинного обучения в условиях ограничений.

Ключевые слова: машинное обучение, диспетчеризация ресурсов, суперкомпьютер, ограничения, нейронные сети

Introduction

Machine learning is one of the fastest growing fields in modern science. The inductive approach in machine learning is a process, in which a computing device analyzes the available data and the desired output, finds correlations between them, and builds a function that relates the input data to the output of the model. Thus, machine learning models are automatically trained to solve classification and regression problems, make predictions and decisions in various fields, such as medicine, finance, manufacturing, etc.

Traditionally, one of the main advantages of machine learning is its ability to process large amounts of data that were previously unavailable for analysis. However, like any other technology, machine learning has its limitations and disadvantages. Some of these include the problem of overfitting, where the model begins to learn from spurious correlations, random noise and errors, rather than from real patterns. In addition, a problem is the availability of small data for training, when training sample sizes are not enough to identify statistically significant patterns. The development of large neural network models, in particular large language models, reveals the problem of performance and cost of running the models.

The aim of the work is to develop methods for applying machine learning models under constraints. The paper considers the results of research projects carried out by the team of the Higher School of Artificial Intelligence Technologies of Peter the Great St. Petersburg Polytechnic University. A distinctive feature of these works is the use of machine learning models under significant constraints. In the first case, the constraint was the small size of the training sample and the requirement to provide the user with the results of the comparison within 15 seconds after the corresponding request was received. In the second case, it was necessary to use predictive analytics models for task execution time on a super-computer in a mode transparent to the user, taking into account the possibility of batch task launch. To solve this problem, an approach was proposed using simple models based on ensembles of decision trees operating asynchronously in the background.

There are a number of approaches to solving the problem of using machine learning models under resource constraints. In particular, if there are restrictions on the size of the training sample, you can use regularization methods, which can reduce the influence of the noise and increase the accuracy of the model [1]. Another approach is the use of ensemble methods [2], which allow replacing one complex model that has a large number of parameters and requires a large training sample, with many simple models trained on different data or on different combinations of input data. Moreover, each of these simple models has a small number of parameters and can be trained on a relatively small sample. Errors in solving a problem made by individual simple models are leveled out using collegial decision-making methods, thereby increasing the accuracy and reliability of the model [3]. You can also increase the sample size using specialized machine learning models, for example, Siamese neural networks, which learn not from the examples themselves, but from their pairs, while the number of pairs grows in proportion to the square of the sample elements and allows training even on a small number of examples [4]. Another method of increasing the volume of initial data is data augmentation. It is a method of generating new data by combining the original data, as well as by adding various noise. In addition, the approach used in augmentation is used to increase the accuracy of the output of various generative models that generate a response based on the user's request. Adding additional contextual data to a request is called Retrieval Augmented Generation (RAG) [5] and is one of the most widely used methods for improving the performance of the large language models.

Machine learning models, especially those built on the basis of neural networks, are quite demanding in terms of the amount of computational resources required both during the training process and at the stage of model operation. There are several basic methods for reducing the computational complexity of machine learning models. One of the main approaches is to try to replace large and complex models with the simpler ones. In this case, methods, such as knowledge distillation [6] are distinguished, in which the nodes of the original neural network that make a small contribution to the result are reset and excluded from the model. In this way, it is possible to reduce the amount of calculations, while maintaining comparable accuracy of the model. The same method is used in the models of explainable artificial intelligence, when the task is to leave only a small number of nodes that make the greatest contribution to the formation of the result of the model, thus highlighting the key elements in the source data, because of which a specific decision was made.

In addition, one solution is to move to other types of machine learning models that are less computationally intensive, such as forests or deep forests. The Higher School of Artificial Intelligence Technologies

researchers proposed methods for combining forests and neural networks [7]. Decision trees carry out the basic classification of the data, spending less resources than the neural networks, since at each step a decision is made based on only one of the attributes. The use of neural networks in the leaves of such trees adds flexibility to the approach and allows us to model complex functions that are not available to classical decision trees.

Further, the article discusses practical problems solved at Peter the Great St. Petersburg Polytechnic University under resource restrictions within the framework of the concept of "Supercomputer for AI and AI for a Supercomputer".

Solving the problem of comparing objects under restrictions on initial data volume

One of the practical problems solved under restrictions on the volume of the training sample was the development of a system for comparing the shape of three-dimensional objects for Rospatent [8]. The task was calculating the semantic proximity function or determining the similarity of digital three-dimensional model using neural networks. Comparison of objects in terms of their semantic proximity in the theory of machine learning refers to a class of tasks that are united under the name of distance metric learning [9]. The main idea underlying the solution to the object comparison problem is that the distance between semantically similar objects should be less than the distance between semantically different objects.

Thus, if there are two training sample vectors $x_i \in \mathbf{R}^m$ and $x_j \in \mathbf{R}^m$, then the distance $d\left(x_i, x_j\right)$ should be minimized, if x_i and x_j are semantically close objects, and this distance should be maximized, if x_i and x_j are semantically distant. The most common and popular distance function is the quadratic Mahalanobis distance $d_M^2\left(x_i, x_j\right)$, which is defined for two vectors as:

$$d_M^2\left(x_i, x_j\right) = \left(x_i - x_j\right)^T M\left(x_i - x_j\right),$$

where $M \in \mathbf{R}^{m \times m}$ is a symmetric positive semidefinite matrix (its eigenvalues are non-negative). The Mahalanobis distance, unlike the Euclidean distance, is scale invariant and allows having correlations between the input data.

The distance matrix is symmetric and positive definite, which allows it to be represented in the form:

$$S^{-1} = W^T W, \ W \in \mathbf{R}^{p \times M}, \ p \le M.$$

Then the Mahalanobis distance can be rewritten as:

$$d_M^2\left(x_i, x_j\right) = \left\| W_{x_i} - W_{x_j} \right\|_2.$$

This equality means that calculating the Mahalanobis distance is equivalent to finding such a linear transformation W that each vector is mapped to a lower-dimensional space, in which the Euclidean distance is equal to the Mahalanobis distance in the original space. Thanks to it, a neural network can be built that functions similarly to the calculation of the Mahalanobis distance, when the activation functions of the neurons are linear. In fact, a neural network allows one to obtain a nonlinear analogue of the Mahalanobis distance by using a combination of linear and nonlinear activation functions.

However, for effective operation, the number of the training sample for neural networks must exceed several times the number of connection weights (training parameters). To solve this problem, the project used a Siamese neural network, which is trained on pairs of data. Thus, having only 600 initial 3D models allowed us to train a Siamese neural network consisting of two six-layer fully connected neural networks containing 96, 150, 120, 80, 40, 32 and 16 neurons, respectively. The second feature of the Siamese neural network is the possibility of using it in so-called one-shot learning (learning from one

example) or few-shot learning (learning using several examples). Therefore, it learns in a situation, where there is only one or a few examples in one class and the initial sample is poorly balanced. This is possible, because the Siamese neural network is trained not on individual examples, but on pairs of examples. Hence, it is not the number of objects themselves, but the number of possible pairs of objects that are the elements for learning.

The second challenge addressed by this project was ensuring the required level of performance, when comparing shapes of 3D models. The models themselves came into the system in the form of wireframe models of arbitrary size (up to 100 MB per model). At the same time, the comparison of a newly received model with all existing ones in the database should not, according to customer requirements, last more than 15 seconds. Due to restrictions on the hardware and the requirement to ensure the functioning of the system in the customer's circuit, a transition from a direct comparison of three-dimensional objects to a comparison of their shape descriptors was proposed, which ensures the identification of significant features of the appearance of three-dimensional objects.

To form a descriptor of the shape of a three-dimensional object, a three-dimensional modification of the chord method was used, which made it possible to move from a three-dimensional model consisting of an arbitrary number of points to a diagram of the lengths of segments connecting points on the surface of the model. The descriptors themselves were generated in an asynchronous mode, while the model entered the system, and during the expert's work, only the calculated descriptor of the new model was compared with the previously calculated descriptors of other models in the database. The system provides the ability to change the descriptor for comparison. If a new descriptor appears, for each incoming model, both old and new descriptors are calculated, until all models in the database are recalculated. In this case, before a complete recalculation, the models are compared using old descriptors, about which the relevant information is displayed to the user, and after the recalculation, only new descriptors are used and the old ones are no longer calculated.

The transfer of data processing procedures to the background allowed us to reduce significantly the hardware requirements of the developed system, while also ensuring the required level of performance, which is one of the key elements of the "Supercomputer for AI" approach.

Using machine learning to improve supercomputer performance

As part of the "AI for a supercomputer" approach, Peter the Great St. Petersburg Polytechnic University developed methods for increasing the operating efficiency or performance of a supercomputer, including using artificial intelligence technologies. Research into the prospects for using various methods of managing computational resources has been actively conducted in different countries over the past few years. At the same time, the developed mathematical, heuristic and metaheuristic methods for optimizing task parameters are used to reduce the energy, resource and time costs of performing computational tasks on supercomputers.

A two-criteria hybrid workflow scheduling algorithm in cloud computing is presented in [10]. The proposed Heterogeneous Gravity Search Algorithm (HGSA) is a hybrid of the popular metaheuristic Gravity Search Algorithm (GSA) and the equally popular heuristic Heterogeneous Earliest Finish Time (HEFT) for scheduling workflow applications. The results show that HGSA performs better than Hybrid Genetic Algorithm by 14%, GSA by 20%, and HEFT by 35% in terms of fitness value. The results were obtained using analysis of variance (ANOVA) statistical test. In our work, this algorithm turns out to be ineffective, since it significantly underestimates the expected value of the task execution time.

The problem of optimizing the scheduling of large applications with parallel processes in heterogeneous computing systems, such as hybrid clouds, is NP-complete and is decisive for Quality of Service (QoS). The article [11] considers the formulation of the optimization problem of a large number of homogeneous parallel batches of problems, which are a bottleneck. The planning problem is solved by a cooperative game algorithm with two-criteria optimization in terms of execution time and economic

costs and with two restrictions − on network throughput and memory size. The proposed algorithm showed better results in the execution time of computational tasks compared to such greedy approaches as G-Min-Min, G-Max-Min or G-Sufferage, and the efficiency increases with the size of the infrastructure. It should be noted that this algorithm has sufficient accuracy, but at the same time has extremely low performance, because the entire set of tasks in the queue is optimized. In conditions of receipt of tens or even hundreds of new tasks per second during batch setting, this method is not applicable to our problem.

A dynamic game-theoretic approach to solving problems of scheduling scientific computing in cloud systems is considered in [12]. A multi-objective workflow scheduling framework is proposed to reduce cloud creation times and costs, while maximizing load distribution among heterogeneous cloud virtual machines. The performance of the developed system was tested on randomly generated templates of scientific workflows and on data from third-party commercial clouds. This approach, like the previous one, is intended to optimize the distribution of a finite set of tasks and cannot be used effectively in the event of new ones arriving. Similar optimization methods are used in [13, 14].

Evolutionary and swarm algorithms are used in problems of scheduling m jobs for n resources for cloud computing and allow obtaining suboptimal solutions. In [15], the multi-objective bacteria foraging optimization algorithm (MOBFOA), which is a modification of the BFOA algorithm for solving multicriteria planning problems with Pareto optimal front. The modification consists of selecting bacterial positions from dominant and non-dominant fronts to obtain a diversity of solutions, which increases the accuracy and the speed of convergence by introducing an adaptive chemotactic step size. The performance of the proposed algorithm was compared in terms of the speed of convergence to the Pareto optimal front and the distribution of solutions in space, with NSGA-II and OMOPSO. The use of these algorithms requires the presence of software agents on all computing nodes, which significantly complicates its implementation on a functioning supercomputer. A similar approach is used in [16].

The growth in the performance of supercomputer systems is associated with an increase in electricity consumption. Balancing energy consumption and performance of computational resources is discussed in [17]. The authors proposed a new scheduling algorithm based on energy-aware duplication − novel energy-aware duplication-based scheduling (NEADS). During the duplication process, they are counted as favorite predecessor (FP), and the next FP. If the FP does not meet the duplication condition, the predecessor FP will not be checked. The second FP is then evaluated. This allows you to avoid replication of indirect tasks on the same processor. Accordingly, additional communication and computation costs can be reduced. Experimental results show that for synthetic applications, NEADS is effective in reducing energy consumption. This approach can be effectively used in systems that process a large number of small tasks. In case of the SCC Polytechnic, planning is performed at the node level as a whole and we can determine the type of the executed task indirectly only by meta-features without access to the executable code of the task.

Accurately predicting the task execution time in a complex high-performance dynamic computing environment is a complex task, subject to a large number of constraints and parameters. One single strategy is not suitable for solving all types of heterogeneous problems. To solve this problem, a joint forecasting model using multiple strategies multi-strategy collaborative prediction mode (MSCPM) was proposed in [18] to perform online tasks in runtime mode. The accuracy of forecasts was assessed using the concept "Ensuring forecast accuracy" introduced by the authors. Experimental results based on a cluster computing environment show that the predictions of the joint prediction model are consistent with the optimal predictions within the results provided by single strategies, and MSCPM provides improved accuracy in predicting the execution time of online tasks. In our work, we proposed an approach using different prediction models for different classes of computational tasks. At the same time, unlike [18], in our work, in addition to the parameters of the task itself, we also use accumulated statistical data on users and users' groups that run the analyzed task. Using additional data on users allows us to increase the accuracy of task execution time estimation by 10−12%.
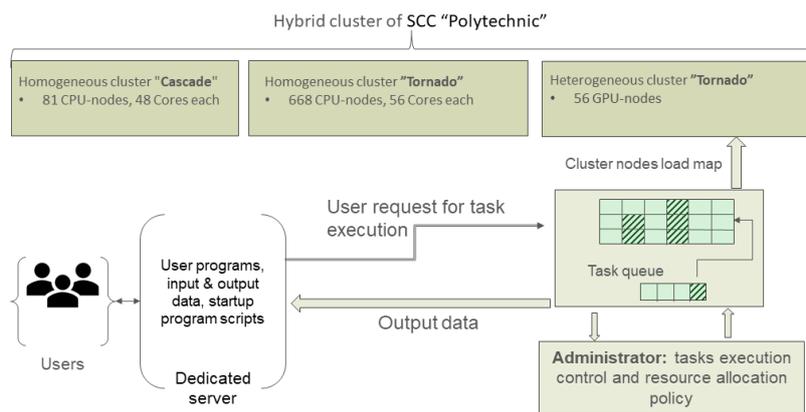
Fig. 1. Current scheme for launching a task by a user at the SCC Polytechnic

Our project involved an unbalanced sample of task execution times as part of the task of increasing dispatch efficiency at the SCC Polytechnic. SCC is a hybrid computing system consisting of several computing clusters with different architectures (homogeneous and heterogeneous), located in a single information and computing field, as well as a shared storage Luster, with a capacity of about 1 PB, allowing file exchange between different computing systems.

Registered users gain access to computational resources from the machine login1.hpc.spbstu.ru. Login is carried out using the SSH protocol (terminal client).

Cluster resources are managed using the Slurm software package [19]. The principle of its operation is as follows: the user requests a certain resource (processor cores, memory, etc.), placing his task or tasks in the queue. The system, based on the user's priorities and the current filling of the queue, selects the moment to launch the task. A queue is a sequence of tasks that must be solved on a specific computational resource (group of nodes). Moreover, each node at the current time can be occupied with only one task of one user. Thus, the node is given exclusive use to the task hosted on it, and other tasks on the busy node will not be executed.

The current scheme for launching a task at the SCC Polytechnic SCC is shown in Fig. 1.

The low efficiency of dispatching is determined, firstly, by the low quality of data received by the dispatcher and used by it in the process of its work. Most users either incorrectly indicate the expected task completion time or do not indicate it at all, leaving the default parameter in the metadata.

As part of the project, an intelligent module was developed and implemented that predicts the task execution time using data on previously performed calculations, in the context of the user running this task.

The scheme for launching tasks at the SCC Polytechnic using an intelligent module is shown in Fig. 2.

When developing the architecture of the new task launch system, the following principles were used:

• the implementation of the intelligent module should be carried out in a "transparent" mode for the user, that is, the algorithm and sequence of user actions as a result of the implementation of the intelligent module should not change in any way;

• as a result of system actions, user tasks should not be removed from execution in the event of an erroneously low calculation time prediction;

• when choosing the time to use to schedule a task, Slurm uses the minimum time between predicted and specified by the user.

It was also necessary to take into account the specifics of queuing tasks, namely batch scheduling of tasks for execution. Using machine learning models to analyze all incoming tasks in a batch in real time requires significant computational resources that can be more efficiently used to solve user problems. To implement the principle of transparency for the user, it was decided not to introduce any additional
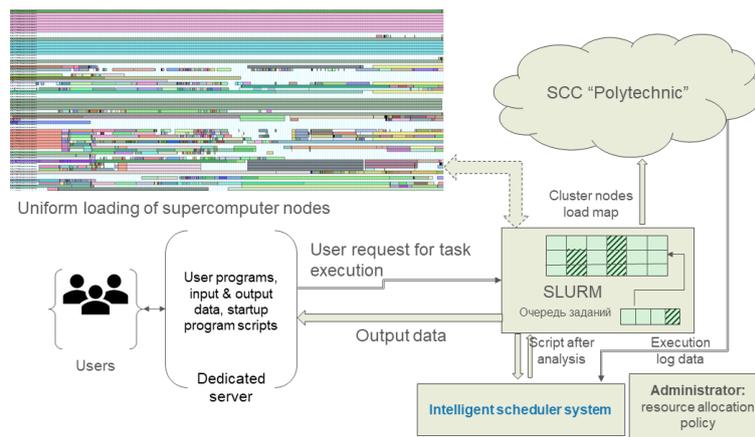
Fig. 2. Scheme for launching a task by a user at the SCC Polytechnic using an intelligent module

elements between the user and the dispatcher. Thus, the intelligent module performs asynchronous processing of tasks that are already in the Slurm database, i.e. are waiting in the execution queue. At the same time, by adjusting the execution time of the user's task, the intelligent module changes the time value of tasks already in the queue, and the dispatcher, during a scheduled review of the queue, adjusts it based on the new time value and transfers some of the tasks, filling the free slots in the schedule. In this case, the intelligent module stores the initial time specified by the user, the value of the predicted time, and monitors the actual time for completing this task. If the predicted time is approaching and the task has not completed, then the intelligent module iteratively increases the specified execution time up to the value specified by the user when starting the task.

At the same time, the machine learning model itself, which predicts the task execution time, uses ensemble learning, in which several simple models, for example, small random forests, are trained on random subsamples of the original data sample. The use of ensemble models can significantly reduce the requirements for resources required at the model training stage, as well as speed up the training procedure itself.

Conclusion

Using machine learning in small sample settings is a complex task that requires careful data analysis and selection of optimal methods. However, thanks to the development of new methods and technologies, the possibilities of using machine learning in such cases are constantly expanding.

There are a number of techniques that allow machine learning models to be used even when there are limitations on the resources used, be it the amount of data used to train the model or the resources required to train or operate it. The most typical technique is to use specialized models that support few-shot methods learning or even one-shot learning, when the training procedure takes place using an extremely small amount of data. In addition, if appropriate, you can use preprocessing of the original data or post-processing of the obtained results, often using other machine learning models.

We have proposed a method based on the use of Siamese neural networks for estimating the shape similarity of three-dimensional objects, which allows training a multilayer neural network using an ultra-small training sample of less than 700 objects.

To improve the efficiency of using machine learning models in the presence of limitations on execution time and computational resources, when estimating the duration of solving a computational problem, a method for asynchronously applying a machine learning model and the Slurm dispatcher has been developed at the SCC Polytechnic. According to the results obtained during modeling, the use of machine learning methods allows reducing the average time a task spends in the queue by 10−15% and

increasing the ratio of the number of solved problems to a given period of time. At the same time, the module does not affect the operation of the Slurm dispatcher itself.

The use of post-processing methods, in addition to increasing the efficiency of the machine learning model, can be used to implement mechanisms for interpreting (or explaining) the results of the model [20].

REFERENCES

1. **Boyd S.P., Vandenberghe L.** Convex Optimization. UK: Cambridge University Press, 2004.

2. **Domingos P.** A Unified Bias-Variance Decomposition and its Applications. International Conference on Machine Learning, 2000, Pp. 231−238.

3. **Breiman L.** Some infinity theory for predictor ensembles, 2000.

4. **Leal-Taixé L., Canton-Ferrer C., Schindler K.** Learning by tracking: Siamese CNN for robust target association. 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2016, Pp. 418−425. DOI: 10.1109/CVPRW.2016.59

5. **Lewis P., Perez E., Piktus A., Petroni F., Karpukhin V., Goyal N., Küttler H., Lewis M., Yih W.-t., Rocktäschel T., Riedel S., Kiela D.** Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, 2021, arXiv:2005.11401. DOI: 10.48550/arXiv.2005.11401

6. **Hinton G., Vinyals O., Dean J.** Distilling the Knowledge in a Neural Network, 2015, arXiv:1503.02531v1. DOI: 10.48550/arXiv.1503.02531

7. **Konstantinov A., Utkin L., Muliukha V.** LARF: Two-Level Attention-Based Random Forests with a Mixture of Contamination Models. Informatics, 2023, Vol. 10, No. 2, Article no. 40. DOI: 10.3390/informatics10020040

8. **Chernoskulova E.** 3D-sistema dlia Rospatenta [3D system for Rospatent]. Scientific Russia, Available: https://scientificrussia.ru/articles/3d-sistema-dlya-rospatenta (Accessed 01.07.2024). (In Russ.)

9. **Suárez J.L., García S., Herrera F.** A tutorial on distance metric learning: Mathematical foundations, algorithms and software, 2018, arXiv:1812.05944v3. DOI: 10.48550/arXiv.1812.05944

10. **Choudhary A., Gupta I., Singh V., Jana P.K.** A GSA based hybrid algorithm for bi-objective workflow scheduling in cloud computing. Future Generation Computer Systems, 2018, Vol. 83, Pp. 14−26. DOI: 10.1016/j.future.2018.01.005

11. **Duan R., Prodan R., Li X.** Multi-Objective Game Theoretic Scheduling of Bag-of-Tasks Workflows on Hybrid Clouds. IEEE Transactions on Cloud Computing, 2014, Vol. 2, No. 1, Pp. 29−42. DOI: 10.1109/TCC.2014.2303077

12. **Wu L., Wang Y.** Scheduling Multi-Workflows over Heterogeneous Virtual Machines with a Multi-Stage Dynamic Game-Theoretic Approach. International Journal of Web Services Research (IJWSR), 2018, Vol. 15, No. 4, Pp. 82−96. DOI: 10.4018/IJWSR.2018100105

13. **Coello C.A.C., Pulido G.T., Lechuga M.S.** Handling multiple objectives with particle swarm optimization. IEEE Transactions on Evolutionary Computation, 2004, Vol. 8, No. 3, Pp. 256−279. DOI: 10.1109/TEVC.2004.826067

14. **Wang Y., Liu H., Zheng W., Xia Y., Li Y., Chen P., Guo K., Xie H.** Multi-Objective Workflow Scheduling with Deep-Q-Network-Based Multi-Agent Reinforcement Learning. IEEE Access, 2019, Vol. 7, Pp. 39974−39982. DOI: 10.1109/ACCESS.2019.2902846

15. **Kaur M., Kadam S.** A novel multi-objective bacteria foraging optimization algorithm (MOBFOA) for multi-objective scheduling. Applied Soft Computing, 2018, Vol. 66, Pp. 183−195. DOI: 10.1016/j.asoc.2018.02.011

16. **Zhou X., Zhang G., Sun J., Zhou J., Wei T., Hu S.** Minimizing cost and makespan for workflow scheduling in cloud using fuzzy dominance sort based HEFT. Future Generation Computer Systems, 2019, Vol. 93, Pp. 278−289. DOI: 10.1016/j.future.2018.10.046

17. **Liang A., Pang Y.** A Novel, Energy-Aware Task Duplication-Based Scheduling Algorithm of Parallel Tasks on Clusters. Mathematical and Computational Applications, 2017, Vol. 22, No. 1, Article no. 2. DOI: 10.3390/mca22010002

18. **Tao M., Dong S., Zhang L.** A multi-strategy collaborative prediction model for the runtime of online tasks in computing cluster/grid. Cluster Computing, 2011, Vol. 14, Pp. 199−210. DOI: 10.1007/s10586-010-0145-4

19. **Zaborovsky V.S., Utkin L.V., Muliukha V.A., Lukashin A.A.** Improving Efficiency of Hybrid HPC Systems Using a Multi-agent Scheduler and Machine Learning Methods. Supercomputing Frontiers and Innovations, 2023, Vol. 10, No. 2, Pp. 104−126. DOI: 10.14529/jsfi230207

20. **Kovalev M.S., Utkin L.V., Kasimov E.M.** SurvLIME: A method for explaining machine learning survival models. Knowledge-Based Systems, 2020, Vol. 203, Article no. 106164. DOI: 10.1016/j.knosys.2020.106164

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Muliukha Vladimir A.
Мулюха Владимир Александрович
E-mail: vladimir.muliukha@spbstu.ru

Vostrov Alexey V.
Востров Алексей Владимирович
E-mail: vostrov_av@spbstu.ru

Motorin Dmitrii E.
Моторин Дмитрий Евгеньевич
E-mail: d.e.motorin@gmail.com

Glazunov Vadim V.
Глазунов Вадим Валерьевич
E-mail: neweagle@gmail.com

COUNT TIME SERIES ANALYSIS OF JOBS SCHEDULING IN THE HYBRID SUPERCOMPUTER CENTER

S.V. Malov ✉ ⓘD , *A.A. Lukashin*

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

✉ sergey.v.malov@gmail.com

Abstract. Increasing the efficiency of supercomputer centers is an extremely important task, especially in the context of growing demand for high-performance computing and a shortage of supercomputer resources. Statistical analysis of the results of various indicators of supercomputer performance is aimed at creating models of computing resource management and forming a basis for using artificial intelligence methods. The purpose of this research is to study the incoming flow of user requests (jobs), which largely determines the load on supercomputer resources. To analyze the incoming flow of user jobs, generalized linear models and generalized estimating equations, as well as the autoregressive conditional Poisson model, were used. It allowed taking into account the dependence of observations and the effect of overdispersion. Based on the results of supercomputer operation observations, estimates of the time trend were obtained, as well as indicators of changes in the intensity of the job flow within weekly and annual cycles with classification by areas of expertise and computing clusters. Indicators of statistical significance of changes within the weekly and annual cycles were established. As a result of an advanced statistical analysis using multiple comparison methods, statistically significant orders of the main effects of the weekly and annual factors were obtained.

АНАЛИЗ ВРЕМЕННЫХ РЯДОВ ЧАСТОТ ДЛЯ ПЛАНИРОВАНИЯ ЗАДАЧ ГИБРИДНОГО СУПЕРКОМПЬЮТЕРНОГО ЦЕНТРА

С.В. Малов ✉ (iD) , *А.А. Лукашин*

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

✉ sergey.v.malov@gmail.com

Аннотация. Повышение эффективности использования суперкомпьютерных центров является крайне важной задачей, особенно в условиях растущего спроса на высокопроизводительные вычисления и дефицит суперкомпьютерных ресурсов. Статистический анализ результатов различных показателей функционирования суперкомпьютера направлен на создание моделей управления вычислительными ресурсами и формирование базы для использования методов искусственного интеллекта. Целью данного исследования является изучение входящего потока заявок пользователей, во многом определяющего загрузку ресурсов суперкомпьютера. Для анализа входящего потока заявок пользователей используются обобщенные линейные модели и обобщенные уравнения оценивания, а также пуассоновская авторегрессионная модель, применение которых позволяет учитывать зависимость наблюдений и эффект избыточной дисперсии. По результатам наблюдений за работой суперкомпьютера получены оценки временного тренда, а также показатели изменений интенсивности потока заявок в рамках недельного и годового циклов с классификацией по областям знаний и вычислительным комплексам. Установлены показатели статистической значимости изменений в рамках недельного и годового цикла с учетом данной классификации. В результате углубленного анализа с использованием методов множественного сравнения получены статистически значимые порядки главных эффектов недельного и годового факторов.

Ключевые слова: дискретные временные ряды, обобщенные уравнения оценки, пуассоновская условно авторегрессионная модель, множественные сравнения, суперкомпьютерный кластер, планирование задач

Introduction

High-performance computing is an important element in computer-aided engineering and fundamental research. Large world-leading research centers use their supercomputers, while the smaller ones use supercomputers operating in shared-use centers. A shared-use center serves a wide variety of users conducting research in various domains including but not limited to mechanical engineering, physics, electronics, life sciences, artificial intelligence etc. This results in very different jobs running on the

same supercomputer cluster in terms of number of cores, memory, software, and time [10]. This makes job scheduling more complex and inefficient as it is difficult to set parameters suitable for all types of jobs.

Modern supercomputers, possessing significant computational resources, simultaneously perform many jobs belonging to different fields of knowledge and imposing different requirements to computational resources and software calculations. Users set jobs for execution using a job scheduler, which forms a queue and schedules their using of the supercomputer resources. The statistical analysis of user job flow is significant for understanding the specifics of using supercomputers as shared-use centers. It allows to proceed to the development of intelligent algorithms for increasing the efficiency of supercomputer system resource usage. The load on the supercomputer resources is largely determined by the incoming flow of user jobs, which is studied in this paper.

Statistical data on supercomputer operation provides new opportunities for optimizing resource utilization. Understanding the parameters of user job flow allows to significantly improve the overall performance of supercomputer systems. The work on data collection and analysis is described in [6], and works [1, 10] demonstrate statistical and machine-learning analysis of supercomputer data. We perform statistical analysis of incoming flow of user jobs that determine requirements of the supercomputer resources at any given time. Statistical analysis of the incoming flow of user jobs allows to optimize the tools of queue management for executing computational jobs and distributing them among computational clusters. For this study, a dataset containing two years of supercomputer center jobs information was collected.

The Poisson process model is applicable for an ideal homogeneous flow. The number of jobs in disjoint and identically sized time intervals are independent and identically distributed random variables having a Poisson distribution with some fixed $\lambda > 0$. The homogeneity requirement of the job flow is too restrictive in practical cases, prompting the use of advanced models for analysis. Statistical analysis of heterogeneous job flow is usually based on time series data on the number of jobs obtained in some equal time intervals (e.g. days). Classical methods in time series analysis require observations to be normally distributed, which is not applicable to count data, especially if some atoms have sufficiently high probabilities. In the particular case of counts of jobs with sufficiently high probabilities of small counts, the classical time series analysis is not applicable. The generalized log-linear regression model (see, e.g. [8]) can be used for statistical analysis of homogeneous counting time series, if the observed counts are independent and have Poisson distribution. The property of equidispersion (equality of mean and variance) of the Poisson distribution is often violated in favor of overdispersion. The same estimating equations lead to consistent estimator of the regression parameters under some mild regularity conditions, even if the independence and the Poisson distribution properties are not satisfied and the number of the observed count time series tends to infinity and the length of each time series remains fixed, which is typical for longitudinal data analysis [2]. The consistent robust variance estimator can be obtained using so-called "sandwich" method. The use of so-called "working correlation matrix" and the generalized estimating equations (GEE) [7, 11] gives more efficient estimators of the regression parameters. It should be noted that the consistency of the robust variance estimator is confirmed as the number of the observed time series increases, whereas at a fixed number of the time series of the increasing length, the consistent variance estimation requires some restrictions on the distributions and dependence structure of the observations.

The alternative framework in heterogeneous flow data analysis is the conditional Poisson model. The multivariate 1st order Poisson autoregressive model [4] assumes that the conditional distribution of count Y_{it} at time t has the Poisson distribution with the following parameter:

$$\mu_{it} = a_{it}\lambda_{it} + v_{it}Y_{i,t-1} + \gamma_{it}\sum_{j\neq i}v_{ij}Y_{j,t-1}, \tag{1}$$

where $\log \lambda_{it} = X'_{it}\beta$; $\log v_{it} = Z'_{it}\beta$ and $\log \gamma_{it} = G'_{it}\beta$, whereas X'_{it}, Z'_{it} and G'_{it} are the regressors. The Poisson autoregressive model as a natural generalization of the Poisson model with independent counts has much wider application area due to the independent counts property violation and the over-dispersion effect. In the particular case of the spatial component absence $\gamma_{it} = 0$ (see [3]),

$$\mathbf{D}Y_{it} = \frac{\mathbf{E}Y_{it}}{1-v_{it}^2} \text{ and } \mathbf{Cov}\left(Y_{it}, Y_{it+k}\right) = \frac{v_{it}^k \mathbf{E}Y_{it}}{1-v_{it}^2}.$$

The multivariate Poisson autoregressive spatial model is widely used in epidemiology. A set of statistical tools for multivariate Poisson autoregressive spatial model is implemented in package *surveillance* [5, 9] for the R programming language[1].

For stratified statistical analysis of user job flows the Poisson log-linear generalized model and the independence estimating equations with the robust "sandwich" variance estimator, implemented in the *geepack* R-package, were used, as well as the univariate 1st order Poisson autoregressive model. All observed jobs were divided into 11 groups based on user area of expertise and 5 groups based on the computing clusters, to which the jobs were submitted, and only 4 of the 5 groups were analyzed. The generalized regression models included a smooth time trend as well as weekly/annual periodic factors. The main goal of the statistical analysis was the investigation of the dynamic change of the intensities of job flow over time in the presence of periodic factors classified by user's area of expertise and computing cluster. In addition to the regression fit and the statistical significance analysis of the periodic factors, some significant partial orders of the main effects using advanced contrasts analysis were obtained.

Explanatory analysis of users' job flow

The study examined historical data on job execution in the "Polytechnic Supercomputer Center". In total, the dataset contained 1545793 records of running jobs. Each record contained a user label, the number of requested resources (processors and supercomputer nodes), and job execution parameters, including how many and what resources were issued, when and how the job was completed. Based on the user label, each job was assigned to an area of expertise, such as physics or mechanics. A total of 11 areas of expertise were identified:

- astrophysics;
- bioinformatics;
- biophysics;
- energetics;
- geophysics;
- IT;
- mechanical engineering;
- mechanics;
- physics;
- radiophysics;
- a special group called geovation.

The last group is related to geophysical software, which runs in an automated mode (the jobs are submitted to the supercomputer queue automatically). Also, these jobs are quite small, but there are a lot of them processed in parallel. This explains the significant number of such jobs, but compared to the number of consumed resources (in terms of node-hours) the figures will be different. All jobs were divided into separate queues representing computing clusters, to which they were submitted:

- "Tornado" – a homogeneous cluster based on CPU (612 node cluster with 28-core compute nodes);

---

[1] The R Project for Statistical Computing, Available: https://www.R-project.org/ (Accessed 25.09.2024)

a)

b)
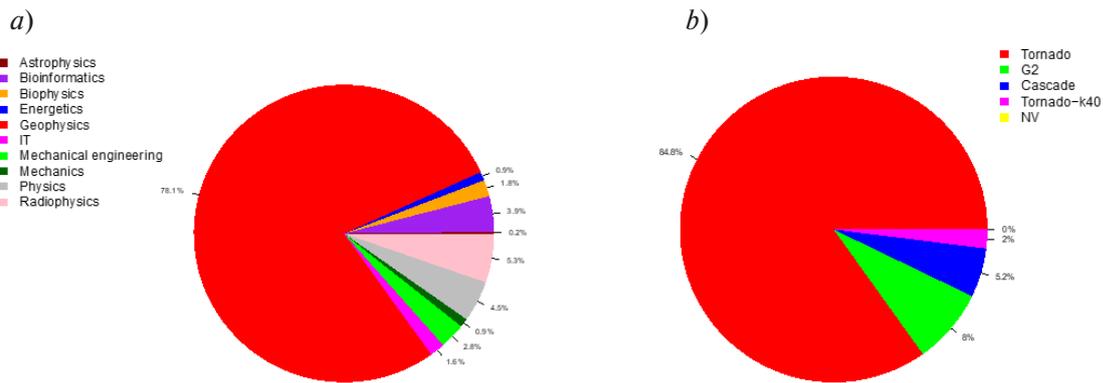
Fig. 1. Percentage of jobs: (*a*) from users in different areas of knowledge; (*b*) different computing clusters

• "G2" – a special cluster for geophysics;
• "Cascade" – a homogeneous cluster with large nodes (81 node cluster with 48-core compute node);
• "Tornado-k40" – a heterogeneous cluster with GPUs (56 node cluster with 28-core nodes with 2 GPUs);
• "NV" – a heterogeneous cluster with GPUs with large nodes (48-core nodes with 8 GPUs).

The percentage of received jobs depending on grouping factors is shown in Fig. 1.

The number of user jobs received from 01.09.2021 to 31.08.2023 is given in Table 1.

Table 1

The number of user jobs divided into groups

Area of expertise	Computing cluster					
	Tornado	**G2**	**Cascade**	**Tornado-k40**	**NV**	**Total**
Astrophysics	2812	0	0	0	0	2812
Bioinformatics	59567	0	0	66	0	59633
Biophysics	23830	2	1	3788	0	27621
Energetics	13893	12	238	145	18	14306
Geophysics	4985	8	1632	1199	2	7826
Geovation	984698	122755	77596	10464	0	1195513
IT	17734	3	421	6780	0	24938
Mechanical engineering	35476	0	51	7174	13	42714
Mechanics	14076	31	44	331	0	14482
Physics	67988	0	0	747	0	68735
Radiophysics	82047	0	0	125	0	82172
Total	1307106	122811	79983	30819	33	1540752

It should be noted, that the distribution of numbers in table 1 is highly unbalanced, with the majority of jobs (63.9%) coming from users in the geophysics area of expertise and being processed by the "Tornado" computing cluster. Moreover, the simultaneous use of two grouping factors, area of expertise and computing cluster, is unpractical due to the presence of a large number of empty cells. Since the total number of jobs received on the computing cluster "NV" the corresponding flow was not analyzed.

Fig. 2 shows the additive time trend estimators of the combined flow using the moving average method with a window size of 365 days, the smoothed moving average obtained by kernel smoothing of the
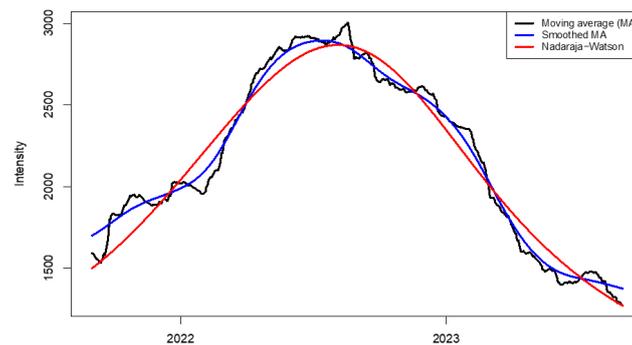
Fig. 2. The additive trend obtained by three different methods
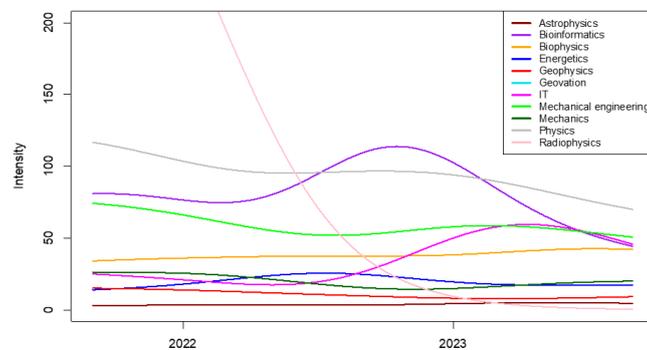
Fig. 3. The additive time trend for user job flow from different areas of expertise
(except for the geovation area)

moving average estimator with the Gaussian kernel and a sufficiently small smoothing parameter of 30, as well as the Nadaraya–Watson estimator with the Gaussian kernel and smoothing parameter of 120. The window size for the moving average method was chosen to exclude the seasonal component effect, and the smoothing parameter for the Nadaraya–Watson estimator was chosen to obtain the estimator sufficiently similar to the moving average. The presence of a time trend in the combined job flow and a sufficient increase in intensity in 2022 are evident, which explains the need to consider the time trend in the statistical analysis models. It should be noted, that the time trend of the combined job flow is determined primarily by jobs in the geovation group, since these jobs are the majority. The Nadaraya–Watson estimators of the additive time trends with the same smoothing parameter depending on the user's area of expertise (except for the geovation group) are presented in Fig. 3. An increase in the intensity of the job flow from the bioinformatics group in the second half of 2022 should be noted, while the other groups are not typical by this effect. Additionally, a significant decrease in the intensity of the job flow for users in the radiophysics area of expertise should be noted and, to a lesser extent, physics area, as well as a slight increase in the intensity of the job flow for users in the IT area of expertise.

A study of changes in the intensities of the job flows over time depending on the computing cluster, to which they were submitted (see Fig. 4.), shows that the increase in the intensity of the job flow observed in 2022 is characteristic only for "Tornado" cluster, and there is also a decrease in the intensity of the job flow for "G2" and an increase for "Cascade" clusters.

The variety of the time trends for different job flows is a strong argument in favor of using nonparametric trend estimates in regression models.
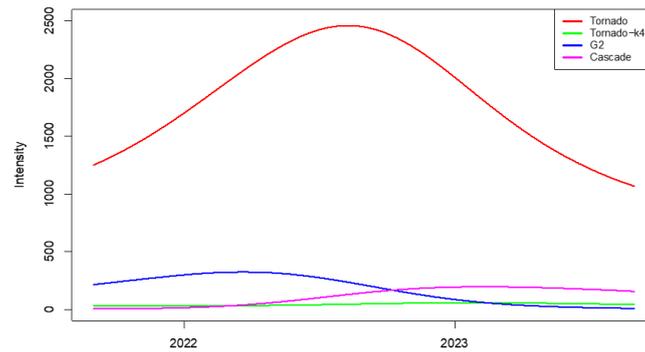
Fig. 4. The additive time trend for user job flow obtained in different computing clusters

Regression analysis of user job flows

The stratified statistical analysis of the job flows groped separately by user's area of expertise and the computing cluster was performed. All statistical inferences were adjusted for 15 flows, including the combined job flow and excluding the "NV" flow, therefore the significance level taking into account the Bonferroni correction is $1/300 \approx 0.003$.

To investigate behavior of the job flows intensities within the annual and weekly cycles, the GEE framework based on the generalized log-linear Poisson model with two additive factors Month and Day of the Week was used:

$$\log\left(\lambda_t\right) = \mu_1 + \alpha_j 1_{\{Month=j\}} + \beta_r 1_{\{Day=r\}} + \log\left(X_t\right), \tag{2}$$

where λ_t is the intensity of the job flow, X_t is the corresponding estimated time trend and t is a day of observation from the beginning of the study, and independence estimating equations. In order to fit the models the R-function *geeglm*() of package *geepack* was used.

Estimates of the multipliers for weekly and annual cycles are given in Tables 2 and 3, respectively. P-value in the last column characterizes the statistical significance of the effect of the corresponding factor on the intensity of the job flow.

The statistical analysis revealed a statistically significant effect of the annual periodic factor for each of the job flows adjusted to the total number of flows, while the effect of the weekly periodic factor was significant only for the mechanical engineering, mechanics an radiophisics flows, as well as for the combined flow and for "Tornado", "Tornado-k40", "G2" flows. Within the annual cycle, a slight decrease in the intensity of user job flows in the summer was observed, which is typical only for researchers in some areas of expertise, and a large variation in intensity throughout the year for researchers in radiophysics, information technology and bioinformatics areas of expertise. It should also be noted, that the sufficient increase of the intensity of the "Cascade" job flow at the end of the year had occurred.

The advanced statistical analysis of pairwise contrasts for the main effects of periodic factors allowed to find several partial orders with a joint reliability of 95%. Let θ_i and θ_j be the logarithmic main effects of levels i and j, respectively, of the factor under study. The pairwise contrast $\psi_{ij} = \theta_i - \theta_j$ allows to determine, whether the main effect of i-th level is smaller than, equal to or larger than the main effect of j-th level.

In order to obtain statistically significant inferences, two-sided joint confidence intervals for the parameters ψ_{ij} with all pairs of levels i and j were constructed using the Bonferroni method. If the confidence interval for the parameter ψ_{ij} lies entirely to the right of zero, the main effect of level i of the factor is less than the main effect of level j, and if it lies entirely to the left of zero, the main effect of level i is larger than the main effect of level j. All the significant inferences obtained in such a manner have the joint reliability of at least 95%.

48

Table 2

Multipliers and main effects for weekly cycle

Flow	MULT	MON	TUE	WED	THU	FRY	SAT	SUN	P-value
Astrophysics	0.87	0.98	1.02	1.1	1.05	1.13	0.89	0.88	0.7849
Bioinformatics	0.54	1.93	0.69	1.36	1.22	0.8	0.78	0.73	0.5897
Biophysics	0.91	1.16	1.14	1.11	0.94	1.07	1.09	0.62	0.3994
Energetics	0.84	1.27	1.39	1.09	1.04	0.82	0.73	0.83	$9.9*10^{-3}$
Geophysics	0.83	0.97	1.78	1.15	0.88	1.27	0.59	0.77	0.1600
Geovation	0.83	1.14	1.24	1.25	1.09	1.29	0.69	0.58	$3.9*10^{-3}$
IT	0.58	1.17	1.02	1.85	0.84	0.97	0.59	0.95	0.3356
Mechanical engineering	0.84	1.42	1.45	1.52	1.37	1.34	0.45	0.39	$1.3*10^{-29}$
Mechanics	0.91	1.36	1.32	1.25	1.06	1.17	0.62	0.58	$5.3*10^{-8}$
Physics	0.96	1.07	1.02	1.27	1.2	1.04	0.76	0.76	$1.3*10^{-2}$
Radiophysics	0.22	3.78	1.23	1.12	1.43	1.05	1.92	0.07	$2.9*10^{-18}$
Tornado	0.88	1.25	1.16	1.22	1.08	1.19	0.77	0.58	$2.4*10^{-3}$
Tornado-k40	0.83	1.28	1.4	1.43	0.94	1.17	0.55	0.64	$9.6*10^{-4}$
G2	0.76	1.48	1.41	1.34	1.42	1.4	0.39	0.46	$2.0*10^{-12}$
Cascade	0.74	1.05	1.41	1.36	0.93	1.4	0.65	0.59	$7.4*10^{-3}$
Combined	0.88	1.25	1.19	1.23	1.09	1.21	0.73	0.57	$1.3*10^{-4}$

Table 3

Multipliers and main effects for annual cycle

Flow	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	P-value
Astrophysics	1.53	2.01	1.55	1.66	1.15	1.31	0.65	0.81	1.09	0.56	0.46	0.57	$2.6*10^{-13}$
Bioinformatics	3.04	0.47	0.47	0.38	0.52	2.44	0.61	0.13	2.16	4.95	2.43	1.52	$7.6*10^{-10}$
Biophysics	0.96	0.89	0.86	1.13	1.04	1.97	1.26	0.39	0.84	0.91	1.27	1.22	$2.2*10^{-6}$
Energetics	0.32	0.64	1	1.08	1.67	1.88	1.08	1.91	0.46	1.48	1.11	0.89	$1.5*10^{-21}$
Geophysics	0.61	0.96	0.88	1.75	0.34	0.89	2.55	0.48	0.95	1.43	1.3	1.68	$9.8*10^{-6}$
Geovation	0.5	1.96	1.36	1.35	0.85	1.1	0.41	2.1	1.23	1.05	0.89	0.59	$1.29*10^{-28}$
IT	1.51	4.39	0.91	4.61	1.71	0.08	0.37	0.24	1.8	1.73	1.37	0.68	$4.4*10^{-26}$
Mechanical engineering	1.16	0.93	1.22	1.1	0.69	1.23	0.71	0.66	0.96	1.04	1.06	1.67	$2.8*10^{-13}$
Mechanics	1.28	0.79	2.11	1.17	0.85	0.84	1.11	0.88	0.97	1.03	0.79	0.72	$2.9*10^{-6}$
Physics	0.87	0.98	0.83	0.96	0.76	1.13	1.05	1.11	0.83	0.99	1.78	1	$3.2*10^{-4}$
Radiophysics	5.89	0.24	2.01	2.13	0.14	0.04	0.13	1.21	5.61	2.33	4.5	3.37	$2.3*10^{-26}$
Tornado	0.73	1.72	1.17	1.17	0.73	1.05	0.45	1.87	1.3	1.16	0.91	0.66	$3.5*10^{-23}$
Tornado-k40	1.41	1.23	0.82	0.63	2.21	0.7	0.39	0.56	1.34	1.14	1.65	1.3	$1.2*10^{-6}$
G2	0.66	1.69	1.75	1.74	0.88	1.77	0.83	2.04	0.38	0.3	0.87	1.13	$1.1*10^{-17}$
Cascade	0.28	0.43	1.33	2.48	1.29	0.72	0.41	0.85	1.97	1.76	3.7	0.61	$7.7*10^{-16}$
Combined	0.7	1.62	1.19	1.23	0.78	1.05	0.46	1.75	1.21	1.08	1	0.7	$3.5*10^{-24}$

The P-value is determined as the minimal $\alpha \le 0.05$, such that all the confidence intervals for the pairwise contrasts of joint significance level $1 - \alpha$ that were entirely in the region to the right or to the left of zero still remain in the same region. The obtained significant orders of the main effects can be visualized as a graph. The nodes of the full graph of significant orders are related to the corresponding levels of factor, and the edges are present, if the order (smaller than or larger than) is confirmed statistically at the established level of confidence adjusted to the number of flows and total number of pairwise contrasts. All levels of the factor can be ordered by the value of the estimator, in which case the edge orientation can be omitted. The edges of the reduced graph are arranged in increasing order of the effect level estimators, and the edge between every two nodes (right and left) is present only if every node to the right of the right node and each node to the left of the left node of the pair are connected by an edge at the full graph of significant orders. Nodes that are not informative for the significant orders can be removed. Although the reduced graph is not uniquely defined by the full graph, there is a subjective component in the choice of the reduced graph version, and some significant orders can be missed, the reduced graph seems more practical for interpreting the results of ordering than the full graph.

The results of the advanced analysis for the multiplicative main effects of the weekly and annual periodic factors are presented as the reduced graphs (one for each flow) in Tables 4 and 5, respectively. For example, the intensity of the combined flow in July is significantly smaller than in November, June, October, March, September, April, November and August; the intensity in July, December and January is significantly smaller than in September, April, February and August; the intensity in July, December,

Table 4

Partial orders of the intensities of the job flows within the annual cycle

Flow	Significant partial orders (reduced graph)	P-value
Astrophysics	NOV DEC JUL AUG SEP MAY JUN JAN MAR APR FEB	0.0031
Bioinformatics	APR MAR JAN OCT	0.0017
Biophysics	AUG APR DEC NOV JUN	0.0024
Energetics	JAN SEP FEB DEC MAR APR JUN NOV OCT MAY AUG	0.0026
Geophysics	MAY JUL	$5*10^{-4}$
Geovation	JUL JAN DEC MAY NOV OCT JUN SEP APR MAR FEB AUG	$4.5*10^{-4}$
IT	JUN AUG JUL DEC MAR NOV JAN MAY OCT SEP FEB APR	0.0028
Mechanical engineering	MAY JUL MAR JUN DEC	$6.2*10^{-5}$
Mechanics	DEC NOV FEB MAR	$6.8*10^{-4}$
Physics	NOV MAY	0.0011
Radiophysics	JUN JUL MAY FEB AUG MAR APR DEC NOV SEP JAN	0.0015
Tornado	JUL DEC JAN MAY NOV JUN OCT APR MAR SEP FEB AUG	0.0030
G2	OCT SEP JAN JUL NOV MAY DEC FEB APR MAR AUG	0.0017
Cascade	JAN JUL FEB DEC JUN AUG MAY MAR OCT SEP APR NOV	0.0017
Combined	JUL DEC JAN MAY NOV JUN OCT MAR SEP APR FEB AUG	0.00213
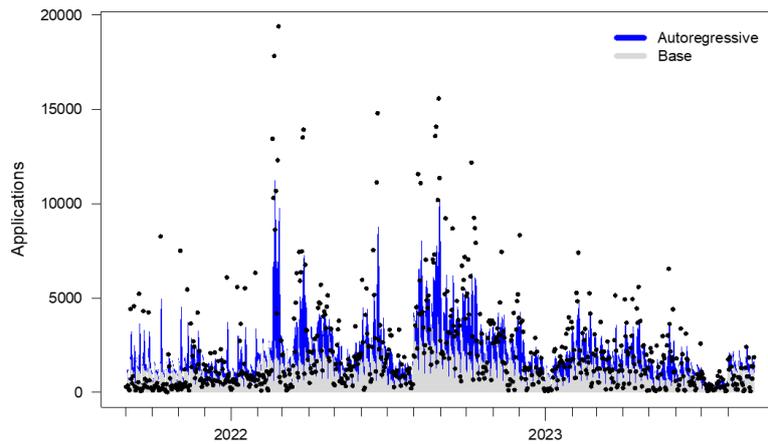
Fig. 5. Observed numbers and Poisson autoregressive predictors

January and May is significantly smaller than in April, February and August; the intensity in July, December, January, May and November is significantly smaller than in August. The intensity of the job flow for users in the bioinformatics area of expertise in April and March is smaller than in January and October (other nodes are omitted). Moreover, for example, the intensity of user job flow in "G2" cluster in December is significantly smaller than in September, but this pairwise order is not marked at the reduced graph in the table, due to the method limitations, since the order of December and September is not significant, whereas the estimator of the intensity in September is smaller than in October.

Table 5

Partial orders of the intensities of the job flows within the annual cycle

Flow	Significant partial orders (reduced graph)	P-value
Mechanical engineering	SUN SAT FRI THU MON TUE WED	$1.9*10^{-8}$
Mechanics	SUN THU FRI WED TUE	0.0032
Radiophisics	SUN TUE THU SAT MON	$1.2*10^{-4}$
G2	SAT SUN WED FRI TUE THU MON	$9.5*10^{-4}$
Combined	SUN TUE WED MON	0.0021

Next, the univariate Poisson autoregressive models (1) was fitted, where the parameterization for λ_t is determined by (2), $\log v_{it} = \alpha$ for all t, and $\alpha > 0$ is the parameter of autoregression, for each user job flow separately. In order to fit the models, the R-function *hhh4*() of package *surveillance* was used. The obtained estimators of the base and autoregressive components of the combined flow are visualized in Fig. 5.

The stratified statistical analysis of user job flows from different areas of expertise and computing clusters showed the significance of annual and weekly periodic factors for each flow adjusted to the number of flows (the maximal P-value of the likelihood ratio test $8.1*10^{-4}$ was obtained for weekly periodic factor of user job flow in radiophysics) and the regression component is formally significant for all job flows, with the exception of the user job flow with the radiophysics area of expertise. In conclusion, it should be noted, that the estimators of the base component λ_t in the Poisson autoregressive model do not determine the intensity changes due to the presence of the autoregressive component.

Discussion

All the jobs were initially classified by the user's area of expertise and by the computing cluster, to which the job was submitted. Considering the results of the explanatory analysis, a stratified approach to study the user job flow was applied. Based on the number of jobs per day for each group of jobs, a time series was generated.

Two approaches were used for stratified analysis of user job flows: the generalized linear model and generalized estimating equation (GEE) based on pseudo-likelihood function, and the Poisson autoregressive model. The GEE analysis revealed significant difference in the intensities in different month of the year for each of user job flows, but no implicit seasonal changes were found, nor did it reveal a common form of the intensity changes for all the job flows. Advanced statistical analysis allowed to reveal some significant partial orders of month by the intensity values for each of user job flow. The statistically significant difference in the intensities of job flows on different days of the week were found for only a part of the flows: mechanical engineering, mechanics and radiophysics, as well as the "G2" cluster, and the combined flow. For each of these five flows, some partial orders of days of the week in terms of intensity values were obtained.

The Poisson autoregressive analysis showed significantly lower variance of the regression and autoregression parameters estimators, which indicated greater stability of the model compared to GEE. The statistical significance of weekly and annual periodic factors of the base component were detected for each of the user job flows. The statistical significance of the autoregressive component was detected for each of the user job flows, excluding users in radiophysics area of expertise. The statistical significance of the autoregressive component can be explained both by the dependence of observations and overdispersion and indicates the inexpediency of using the Poisson generalized linear model, when the observations are independent.

REFERENCES

1. **Baranov A.V., Nikolaev D.S.** Machine learning to predict the supercomputer jobs execution time. Software & Systems, 2020, Vol. 33, pp. 218−228. DOI: 10.15827/0236-235X.130.218-228

2. **Diggle P.J., Heagerty P., Liang K.-Y., Zeger S.L.** Longitudinal data analysis. 2nd edition. Oxford: Oxford University Press Inc., 2002.

3. **Fokianos K., Rahbek A., Tjøstheim D.** Poisson autoregression. Journal of the American Statistical Association. 2009, Vol. 104, no. 488, pp. 1430−1439. DOI: 10.1198/jasa.2009.tm08270

4. **Held L., Höhle M., Hofmann M.** A statistical framework for the analysis of multivariate infectious disease surveillance counts. Statistical Modelling, 2005, Vol. 5, no 3, pp. 187−199. DOI: 10.1191/1471082X05st098o

5. **Höhle M., Meyer S., Paul M.** surveillance: Temporal and Spatio-Temporal Modeling and Monitoring of Epidemic Phenomena. 2022, Available: https://CRAN.R-project.org/package=surveillance (Accessed 25.09.2024)

6. **Klinkenberg J., Terboven C., Lankes S., Müller M.S.** Data Mining-Based Analysis of HPC Center Operations. 2017 IEEE International Conference on Cluster Computing (CLUSTER), 2017, pp. 766−773. DOI: 10.1109/CLUSTER.2017.23

7. **Liang K.-Y., Zeger S.L.** Longitudinal data analysis using generalized linear models. Biometrika, 1986, Vol. 73, no 1, pp. 13−22. DOI: 10.1093/biomet/73.1.13

8. **McCullagh P., Nelder J.A.** Generalized linear models. 2nd edition. London: Chapman & Hall, 1989.

9. **Meyer S., Held L., Höhle M.** Spatio-temporal analysis of epidemic phenomena using the R package surveillance. Journal of Statistical Software, 2017, Vol. 77, no. 11, pp. 1−55. DOI: 10.18637/jss.v077.i11

10. **Zaborovsky V.S., Utkin L.V., Muliukha V.A., Lukashin A.A.** Improving Efficiency of Hybrid HPC Systems Using a Multi-agent Scheduler and Machine Learning Methods. Supercomputing Frontiers and Innovations, 2023, Vol. 10, no. 2, pp. 104−126. DOI: 10.14529/jsfi230207

11. **Zeger S.L., Liang K.-Y., Albert P.S.** Models for longitudinal data: A generalized estimating equation approach. Biometrics, 1988, Vol. 44, no. 4, pp. 1049−1060.

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Malov Sergey V.
Малов Сергей Васильевич
E-mail: sergey.v.malov@gmail.com
ORCID: https://orcid.org/0000-0003-0093-6506

Lukashin Alexey A.
Лукашин Алексей Андреевич
E-mail: lukash.spb.ru@gmail.com

PREDICTIVE MODELS AND DYNAMICS
OF ESTIMATES OF APPLIED TASKS CHARACTERISTICS
USING MACHINE LEARNING METHODS

A.V. Konstantinov ✉ 🆔

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

✉ andrue.konst@gmail.com

Abstract. The paper considers the machine learning problem of simultaneous estimation of the conditional survival distribution and dynamic characteristics of computational tasks. The problem arises in cluster workload management and is extremely relevant for optimal scheduling. To solve the problem, a new method is proposed, based on the combination of the attention mechanism and the random survival forest. The key feature is the use of a tree structure derived from a random survival forest. The forest construction algorithm uses only the survival dataset. Each leaf uses the unconditional Kaplan-Meier estimate, which is a serious limitation of the forest, especially for rare events in some parts of the feature space. Moreover, the random survival forest does not allow estimating the dynamic parameters of the task. The proposed method solves these problems by extending the already constructed random survival forest with the attention mechanism inside each leaf of the tree. The Beran estimator is used to model survival distribution, and the Nadaraya-Watson regression with the same parameters is used to predict the dynamic characteristics of tasks. To do this, subsets of training data corresponding to the same leaf as the input vector are used. As a result, the joint model is obtained that allows us to estimate the survival function more accurately and at the same time to predict the dynamic characteristics of the task. The developed model combines the advantages of smooth models based on the attention mechanism and stepwise decision trees.

МОДЕЛИ ПРЕДСКАЗАНИЯ И ДИНАМИКА ОЦЕНОК ХАРАКТЕРИСТИК ПРИКЛАДНЫХ ЗАДАЧ МЕТОДАМИ МАШИННОГО ОБУЧЕНИЯ

А.В. Константинов ✉ ⓘ

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

✉ andrue.konst@gmail.com

Аннотация. В статье рассматривается задача машинного обучения, заключающаяся в одновременной оценке условного распределения выживаемости и динамических характеристик вычислительных задач. Проблема возникает при управлении рабочей нагрузкой кластера, и крайне актуальна для оптимального планирования. Для решения задачи предложен новый метод, основанный на комбинации механизма внимания и случайном лесе выживаемости. Ключевой особенностью является использование древовидной структуры, полученной случайным лесом выживания. Алгоритм построения леса опирается только на данные задачи выживаемости. В каждом листе используется безусловная оценка Каплана-Мейера, что является серьезным ограничением леса, особенно в случае редких событий в некоторых частях пространства признаков. Более того, случайный лес выживаемости не позволяет оценить динамические параметры задачи. Предлагаемый метод решает данные проблемы, дополняя уже построенный случайный лес выживаемости механизмом внимания внутри каждого листа дерева. Для моделирования выживаемости применяется оценка Берана, а для предсказания динамических характеристик задач — регрессия Надарая-Ватсона с теми же параметрами. Для этого используются подмножества обучающих данных, соответствующие тому же листу, что и входной вектор. В результате получена совместная модель, позволяющая более точно оценить функцию выживаемости и одновременно предсказать динамические характеристики задачи. Разработанная модель сочетает в себе преимущества гладких моделей, основанных на механизме внимания, и ступенчатых деревьев решений.

Ключевые слова: машинное обучение, анализ выживаемости, механизм внимания, случайный лес выживаемости, оценка Берана

Introduction

Computational clusters are widely used to solve problems that require significant computing power, allowing many computational tasks to be performed simultaneously. One of the key aspects of effective cluster resource allocation planning is estimating the parameters of computing tasks, such as execution time, and individual characteristics that are unknown at the time the task is launched. Each computational task is characterized by a feature vector that is a set of input parameters, including user-specified characteristics and parameters determined by the state of the system at the time the task is queued, including its execution time. After starting a task, two outcomes are possible: the task is completed within

the allotted time, or the task is terminated by the control system after the specified time has elapsed, in other words, censoring occurs. Thus, the task of estimating execution time is to determine the expected time to an event (completion of a task) under censoring conditions and is the task of survival analysis.

The i-th training observation consists of input feature vector x_i, time to event t_i, censoring indicator δ_i and target vector of computational task parameters y_i. The event in the current problem is task completion or interruption. The time to event t_i corresponds to the time between the task launch and completion or interruption, depending on δ_i. The censoring indicator $\delta_i = 1$, if the task has finished normally. Otherwise, the task execution has interrupted. The interruption can be caused by time limit violation or some program error. In this case, the observation is called censored in terms of survival analysis. Finally, the training data set D is composed of N labeled training observations:

$$D = \left\{ \left(x_i, t_i, \delta_i, y_i \right) \right\}_{i=1}^{N}.$$

Since the execution time does not depend on the input feature vector deterministically, we introduce T, a random variable corresponding to the execution time. The main goal is to estimate the parameters of an applied task $y(x)$, including, but not limited to, the expected execution time of the task, conditioned on the input feature vector $x : \mathbb{E}\left[T \middle| X = x \right]$. More generally, the survival distribution $S(t|x)$ is of interest.

Survival Models

In this paper, we consider two base models for survival function estimation. The first is the Random Survival Forest (RSF) [1]. It is a machine learning algorithm that does not impose any assumptions on the data distribution, which makes it different from classical survival analysis methods for conditional distribution estimation, for example Cox Proportional Hazards [2]. Instead, it partitions the data using feature vector x, and then estimates distribution shape based on unconditional non-parametric statistics. The second model is the Beran estimator [3]. It can be considered as the kernel-based extension of Kaplan-Meier unconditional estimator to the conditional case. Given the weight function W, the method estimates conditional survival function as:

$$\hat{S}(t|x) = \prod_{t_i \le t} \left(1 - \frac{W(x, x_i)}{1 - \sum_{j=1}^{i-1} W(x, x_j)} \right)^{\delta_i},$$

where W is normalized over dataset points, and training observations are ordered such that time t_i increases by the index i. Specifically, in the original Beran the weights are obtained by normalization of kernels:

$$W(x, x_i) = \frac{K(x, x_i)}{\sum_{j=1}^{N} K(x, x_j)}.$$

Let us describe the RSF construction and prediction algorithms. Like in classical Random Forest, the trees of the forest are built independently of each other, using different random dataset subsamples and feature subsets (random subspace method). Each tree is built by a recursive algorithm. At each step the algorithm considers a tree node and tries to make a split, resulting in two child nodes, connected to the node. The data point falls to the left child node if some selected feature value is less than the specific threshold, and otherwise it falls to the opposite, right node. When splitting a node, at the training stage, the feature and threshold values are determined by optimizing goodness of split criterion. The main goal of splitting is to divide the sample received into it in such a way that the survival distributions for the left

and right subtrees are as different as possible. For this purpose, the logrank test is used as a criterion. The training algorithm stops splitting a node, when the number of training data points falling to the node is less than some predefined number. The resulting decision trees can be used to estimate the conditional survival function for a new observed input feature vector: at each leaf of each tree, a nonparametric Kaplan-Meier estimator is constructed based on the data falling to the leaf. Note that even though such an estimation is unconditional, each leaf of the tree corresponds to a strictly defined region of space for which this estimation is valid.

Attention mechanism

The attention mechanism is the main element of the most successful method for processing sequences, including natural language, the Transformer [4]. It is implemented as a convex combination of vectors called value vectors, where the weights are obtained by kernel applied to the query and key vector pairs. This mechanism allows the model to focus on the most important parts of the input data when making predictions. In recent years, attention mechanisms have been successfully applied to problems from other domains, such as computer vision, speech recognition, and regression and classification [5]. Despite successful application of attention mechanism to many machine learning problems, attention mechanisms and their combinations with decision trees have not previously been used to estimate survival distributions, and simultaneously solve survival analysis and regression problems.

Attention mechanism can be formalized as follows. Let q be a "query" vector, $\left\{\left(k_i, v_i\right)\right\}_{i=1}^K$ be a set of "key-value" pairs, and "score" be a function mapping pair $\left(q, k_i\right)$ to relative score or relevance of the query to the key. Attention of "q" to the given set of pairs is the convex combination of values v_i:

$$A\left(q, \left\{\left(k_i, v_i\right)\right\}_{i=1}^K\right) = \sum_{i=1}^K \alpha_i v_i,$$

where coefficients α_i are defined as:

$$\alpha_i = \left(softmax\left[\left(score\left(q, k_j\right)\right)_{j=1}^K\right]\right)_i = \frac{\exp\left(score\left(q, k_i\right)\right)}{\sum_{j=1}^K \exp\left(score\left(q, k_j\right)\right)}.$$

Therefore, attention is a function of vector and a set of pairs of arbitrary size, which maps them the one vector, characterizing the set for the query linearly. By using this property, attention was successfully applied for improving Random Forest performance [6]. It should be noted that attention resembles well-known kernel regression algorithm, called the Nadaraya-Watson regression [7]. Indeed, if score is defined as:

$$score\left(q, k\right) = -\frac{1}{2}\|q - k\|^2,$$

and as pairs $\left(x_i, y_i\right)$ are considered, then attention is equivalent to the Nadaraya-Watson regression with the Gaussian kernel. However, the "score" function can be more complex, reflecting complex structure of the dataset, for example, it can be implemented as a neural network and trained in an end-to-end manner [8].

Attention-based Random Survival Forest

We propose a new approach based on incorporation of the attention mechanism into RSF for estimating the parameters of applied tasks, called Attention-based Random Survival Forest (ABRSF). The

key idea of ABRSF is to leverage the same attention weights to improve quality of survival estimation and to approximate the task parameter vector.

The algorithm consists of two steps. At the first step, a classical RSF is constructed using the survival dataset. Its leaf nodes estimate unconditional survival distribution by the Kaplan-Meier estimator. These estimators can be replaced by the Beran models, based on attention weights, instead of classical kernels. At the same time, the same attention model can also be considered as kernel regression and applied with the same kernel to solve the task parameter vector estimation problem [8]. Formally, in each tree leaf attention weights are calculated based on the train data points which fall into the same leaf as "keys", and an input vector as "query". Then, for survival function estimation Beran estimator is applied, where attention weights are as used instead of kernels. For the task parameter vector estimation simply the attention mechanism is used, where "values" are training dataset task parameter vectors. It is important, that the attention in the proposed model is applied only locally, where neighboring points are determined by the RSF structure, which was optimized for the survival problem. The obtained model is smooth inside each region defined by leaf, and has discontinuities at separating hyperplanes, defined by internal nodes of the forest trees.

The described one-step approach allows us to solve the formulated problem but has the following drawback: different random forest trees, depending on the training subsample and feature subspace, have different accuracy. In addition, some trees may be accurate in the context of a survival problem and less accurate in the context of a regression task of estimating target parameters. To eliminate this drawback, we modify the proposed approach by adding tree weights. In addition to estimating the target parameters, each leaf of each tree also estimates the input feature vector using the same mechanism, where feature vectors are used as "values", as well as "keys":

$$\hat{x} = A\left(x, \left\{ \left(x_{l_i}, x_{l_i} \right) \right\}_{i=1}^{K_l} \right),$$

where l represents indices of train dataset points, falling into the same leaf as x, and K_l is the number of such points. The negative distance or "score" between x and its reconstruction $\hat{x}$ can be used as a measure of attention weights quality. By how close the input feature vector is to its estimate, one can judge about closeness of the target parameter vector estimate to the true value. So, next, the feature vector reconstructions obtained from different trees act as keys for the global attention.

Let the $\hat{x}(j)$ be a reconstruction of the input feature vector x by the j-th tree, and the $\hat{y}(j)$ be the task parameter vector estimation by the same tree. Then the final estimation is defined by the attention:

$$\tilde{y} = A\left(x, \left\{ \left(\hat{x}(j), \hat{y}(j) \right) \right\}_{j=1}^{\tau} \right),$$

where τ is the number of trees. The "query" is the original feature vector, the "keys" are reconstructed feature vectors, and the "values" are the estimates of the task parameter vector. The same technique is applied to combine tree survival function estimations to the final one.

Finally, the ABRSF model consists of two layers: tree-level estimations and forest-level weighted combination. The scheme of the ABRSF model is shown in Fig. 1. At the first layer, each tree estimates three parameters: the survival function, encoded as a vector, the task parameter vector, and the input vector reconstruction. At the second layer, tree-level estimates are combined by using attention weights, obtained by collating the input vector reconstructions with the given input feature vector. After passing these two layers, the final estimates are locally smooth and more precise than piecewise constant RSF ones. Moreover, the task parameter vector is calculated by using the RSF structure, because only points in the same leaf are considered in each tree, which lead to more accurate results when the survival data is correlated with the estimated vector.
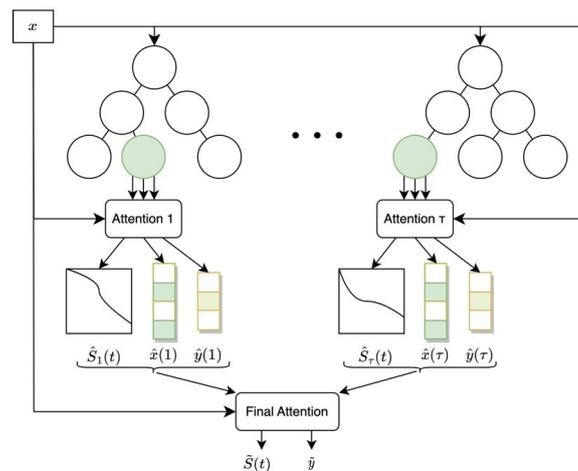
Fig. 1. ABRSF model scheme

Conclusion

The problem of joint target parameter vector and survival distribution estimation has been considered. The novel method, based on combination of Random Survival Forest and Attention mechanism, and called ABRSF, is proposed. The developed method has advantages in comparison to classical forest: it builds piecewise smooth prediction models and leverages the survival tree structure, when estimating the task parameter vector. As a further direction, this approach can be expanded by using multilayer neural networks in the attention mechanism and training the model on regression and survival analysis problems simultaneously, using the backpropagation algorithm, as well as adapting the approach for correctly processing missing features in the input data.

REFERENCES

1. **Ishwaran H., Kogalur U.B., Blackstone E.H., Lauer M.S.** Random Survival Forests. The Annals of applied statistics, 2008, Vol. 2, No. 3, Pp. 841−860. DOI: 10.1214/08-AOAS169

2. **Fox J.** Cox proportional-hazards regression for survival data. An R and S-PLUS Companion to Applied Regression, 2002, Pp. 1−18.

3. **Beran R.** Nonparametric regression with randomly censored survival data, 1981.

4. **Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I.** Attention is all you need. NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2023, pp. 1−15. DOI: 10.48550/arXiv.1706.03762

5. **Lin T., Wang Y., Liu X., Qiu X.** A survey of transformers. AI open, 2022, Vol. 3, Pp. 111−132. DOI: 10.1016/j.aiopen.2022.10.001

6. **Bierens H.J.** The Nadaraya−Watson kernel regression function estimator. Topics in Advanced Econometrics: Estimation, Testing, and Specification of Cross-Section and Time Series Models. Cambridge: Cambridge University Press; 1994.

7. **Utkin L.V., Konstantinov A.V.** Attention-based random forest and contamination model. Neural Networks, 2022, Vol. 154, Pp. 346−359. DOI: 10.48550/arXiv.2201.02880

8. **Konstantinov A.V., Utkin L.V., Lukashin A.A., Muliukha V.A.** Neural attention forests: Transformer-based forest improvement. International Conference on Intelligent Information Technologies for Industry, 2023, Pp. 158−167. DOI: 10.48550/arXiv.2304.05980

INFORMATION ABOUT AUTHOR / СВЕДЕНИЯ ОБ АВТОРЕ

Konstantinov Andrei V.
Константинов Андрей Владимирович
E-mail: andrue.konst@gmail.com
ORCID: https://orcid.org/0000-0002-1542-6480

RECONSTRUCTION OF ATTRACTORS OF SUPERCOMPUTER USER'S ACTIVITY AND IDENTIFICATION OF CRITICAL DEVIATIONS IN THEIR BEHAVIOR

I.O. Mezheneva, A.A. Lukashin ✉ *, S.K. Chatoyan*

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

✉ lukash.spb.ru@gmail.com

Abstract. The modern job scheduling system in supercomputer platforms is based on the estimates of the request for computing resources provided by users (often based on subjective considerations). However, it has been found that such estimates can be significantly inaccurate. In this regard, a practically important task arises: building a behavior model of user tasks executed in a supercomputer, identifying and evaluating critical deviations from the predicted behavior profile (based on an assessment of user confidence). Methods of nonlinear dynamics and topological data analysis are used to solve this problem. The article presents the results of experimental studies for various data sets obtained at the "Polytechnic Supercomputer Center" of Peter the Great St. Petersburg Polytechnic University. The Betti curves of the supercomputer user profile are calculated. The results of the evaluation of the comparison of several user profiles with the reference profile are presented. A desirability scale and numerical intervals for the proposed classes are proposed.

РЕКОНСТРУКЦИЯ АТТРАКТОРОВ АКТИВНОСТЕЙ ПОЛЬЗОВАТЕЛЕЙ ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ СУПЕРКОМПЬЮТЕРНЫХ ПЛАТФОРМ И ВЫЯВЛЕНИЕ КРИТИЧЕСКИХ ОТКЛОНЕНИЙ В ИХ ПОВЕДЕНИИ

И.О. Меженева, А.А. Лукашин ✉ *, С.К. Чатоян*

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

✉ lukash.spb.ru@gmail.com

Аннотация. Современная система диспетчеризации задач в суперкомпьютерных платформах основана на оценках потребности в вычислительных ресурсах, предоставленных пользователями (зачастую на основе субъективных соображений). Однако было установлено, что такие оценки могут быть существенно неточными. В связи с этим возникает важная в практическом отношении задача – построение модели поведения пользовательских заданий при их выполнении в суперкомпьютере, выявление и оценка критических отклонений от прогнозируемого профиля поведения (на основе оценки доверия к пользователю). Для решения этой задачи используются методы нелинейной динамики и топологического анализа данных. Приводятся результаты экспериментальных исследований для различных наборов данных, полученных в «Суперкомпьютерном центре "Политехнический"» Санкт-Петербургского политехнического университета Петра Великого. Посчитаны кривые Бетти профиля пользователя суперкомпьютера. Представлены результаты оценки сравнения нескольких профилей пользователей с эталонным профилем. Предложена шкала желательности и числовые интервалы для предложенных классов.

Ключевые слова: высокопроизводительные вычисления, гибридные вычислительные системы, топологический анализ данных, скалярные временные ряды, планирование задач

Introduction

Job scheduling is one of the key systems for supercomputer platforms, which significantly affects their performance [1]. The basis of this process is an assessment of the resource requirements of jobs, such as processor time and memory capacity. Based on this information, the dispatcher generates a schedule for completing tasks. However, existing dispatch systems rely on estimates provided by users, which often turn out to be inaccurate, resulting in inefficient use of valuable computing resources [2].

As the experience of operating the "Polytechnic Supercomputer Center" (SCC Polytechnic) shows, the inaccuracy of such estimates is usually due to the following reasons [3]:

• lack of experience of users of supercomputer platform resources in assessing the necessary needs for computing resources to solve a particular task;

• insufficient consideration of the specifics of the task being solved;

• complexity of predicting the behavior of complex algorithms, especially when using third-party libraries.

Moreover, since the dispatcher does not allocate resources beyond the requested amount, users tend to overestimate their estimates to ensure successful job completion.

The study is performed on the data, which was collected during the operation of the SCC Polytechnic. The dataset contains information about around 1.5 million of submitted jobs. SCC Polytechnic provides access to four different supercomputer clusters with the following parameters:

1. Cluster "Tornado" – consists of 612 nodes with 28-cores computers;
2. Cluster "Cascade" – consists of 81 nodes with 48-cores computers;
3. Cluster "Tornado-k40" – consists of 56 nodes with 28-cores computers with 2 GPUs each;
4. Cluster "NV" – consists of nodes with 48-cores computers with 8 GPUs each.

Each task in the dataset is submitted into one of these clusters and has information about its real execution and final task status.

In this regard, there is a need to develop methods for analyzing the behavior of users of supercomputer platforms – a method for reconstructing the dynamics of resource consumption, in particular, identifying deviations and evaluating them as critical. The proposed approach to computing behavior patterns is widely used for detecting anomalies in cybersecurity, retail, and other domains [4, 5]. Understanding the behavior patterns of supercomputer users allows to develop algorithms for improving the efficiency of using supercomputer resources.

The research methods include the theory of embedding time series in a reconstructed phase space, the theory of persistent homology, the theory of step functions, and decision theory methods based on the Harrington function.

The methodology of building profiles

The development of a methodology for building profiles based on complex and voluminous data, aimed at identifying deep patterns and abnormal behavior, as well as building descriptors reflecting various behavioral models, will be considered from the point of view of approaches based on topological data analysis (TDA) [6]. TDA, as a branch of data science, combines the principles of algebraic topology, differential geometry, functional analysis, mathematical statistics, and computer science.

In this approach, user behavioral profiles are built based on "data point clouds", which are disordered datasets that do not depend on a specific metric time (or similar) structure [7]. Topological spaces are mapped to these clouds of data points, to which TDA methods are then applied.

In particular, in task planning systems for supercomputer platforms, user behavior is often presented in the form of time series that cover multidimensional information about requests for computing resources: the type of resource, the amount of resources required (number of cluster nodes, processor requirements, memory, etc.) and the duration of the task (including actual operational data). Therefore, the first step in building a behavioral profile is to transform data from time series into point clouds and match them to the corresponding topological spaces. Thus, it is a process that ensures the integrity of information and the preservation of the existing "geometry" in the data, i.e. the choice of the appropriate topological space is carried out in such a way as to "cover" all the elements of the time series [8].

The main idea of TDA is to map a data set to the corresponding topological spaces, approximate them with simplicial complexes, and then apply the persistent homology technique to study the properties of these structures [6]. In the context of simplicial complexes, the theory of persistent homology relies on the mechanism of simplicial filtering, which systematically generates several nested, weakly dependent complexes, thereby revealing their evolution and stability at different levels of analysis. In this process, the key metrics are topological invariants, such as persistent homology groups and their numerical measure, the Betti numbers, which provide a deep understanding of the structural features of the data, and their geometry. The significance of each property is assessed through its "persistence" in filtration time – a concept that, although conditionally related to time, more accurately reflects the changing depth of analysis or the scale of consideration. The significance of this approach lies in the fact that the duration of the existence of such invariants directly correlates with the geometric structure of the studied simplicial complexes, which are approximating models of topological spaces corresponding

to data clouds. Thus, persistent homology acts as a means for quantifying the stability and multilevel analysis of the topological (geometric) data characteristics.

Step 1. Converting a time series to a point cloud

The development of an approach to the embedding space construction is based on the fundamental Takens theorem application, which is aimed at the attractor reconstructing of a dynamical system from a scalar observable [9]. The embedding theory establishes that to obtain a representation of the phase space of such a system, it is possible to replace true, often inaccessible, system variables with sequences of d-dimensional vectors with a delay collected from samples of the time series $x(t)$ at successive time points:

$$\vec{x}(t) = \left(x(t),\ x(t-\tau),\ \ldots,\ x\left(t-(d-1)\tau\right) \right)^T,$$

where τ is the time delay, d is the dimension of the embedding.

The main guarantee provided by the Takens theorem is that such an embedding structure preserves the key characteristics of the original time series up to continuous maps [8]. This means that when constructing a topological embedding, we can freely choose any continuous function, among which the shift introduced through delay is the simplest option among possible transformations.

We will determine the optimal parameters of the embedding dimension and the time delay using an algorithm developed based on the L-statistical methodology. This technique follows from the concept of the noise measure, first proposed by Casdagli [10, 11], and aimed at quantifying the embedding quality, based on the analysis of the disintegration of close trajectories in the reconstructed space. If the attachment is unreliable, even minor changes can significantly distort the true state of the system, increasing the influence of noise and reducing the accuracy of reconstruction.

Unlike the classical Casdagli approach, the improved L-statistics modifies, freeing itself from the need to determine a specific prediction horizon for measuring noise amplification [12]. The algorithm implementing this principle is based on the analysis of the proximity of neighbors and strives to preserve both the geometric and topological characteristics of the original and restored attractors. The goal is to maintain a correspondence between the structural features of the original time series and their projections in the embedding space, ensuring maximum informativeness without loss of significant properties. Combining the principles of redundancy and irrelevance into a single metric, L-statistics is formalized as an objective function, the optimization of which seeks to reduce both aspects simultaneously [12].

As a result, the time series is transformed into a discrete cloud of points inside a topological (usually Euclidean space) space, i.e. $\mathbb{R}^d$. Next, TDA procedures are applied sequentially to the point cloud.

Step 2. Topological data analyses

Let us start with the assumption that the point cloud is inscribed in a metric topological space (which is guaranteed for the embedding procedure) by introducing the Euclidean metric. The next step will be to triangulate this structure using the Vietoris–Rips complex. Taking into account the sensitivity of the triangulation process to the level of proximity of points (and, accordingly, the proximity parameter introduced during the construction), we use a strategy for calculating persistent homology [6], in which the Euclidean metric gradually increases and the evolution of topological features is recorded in the form of a filtered Vietoris–Rips complex [13]. Homology groups are calculated for each K^i complex:

$$H_k^i\left(K^i\right) = Z_k^i\left(K^i\right)\Big/ B_k^i\left(K^i\right),\ k = \overline{1,\ \ldots,\ n}.$$

where $Z_k^i(\cdot)$ is the cycle group, $B_k^i(\cdot)$ is a group of complex boundaries.

Persistent homology groups track the changes that occur, when the Euclidean metric (proximity parameter) changes — the appearance and disappearance of topological features — and associate the corresponding persistence with them.

As a topological descriptor — the basis of analysis, reflecting topological features in a form convenient for analysis — in this work, Betti curves were chosen, expressed through changes in Betti numbers during the process of filtration. They are step functions describing the life cycles of topological features and their resistance to changes in proximity scale. Betti numbers $\beta_k^i \left(K^i \right)$ are calculated using the formula (in the context of a vector space):

$$\beta_k^i \left(K^i \right) = \dim H_k \left(K^i \right) = \dim Z_k^i \left(K^i \right) - \dim B_k^i \left(K^i \right).$$

Betti curves were chosen because, being step functions, they allow us to calculate the average curve and to provide a simple method for estimating distances [14].

Step 3. Identification and evaluation of deviations from the basic profile

Our hypothesis is based on the idea that any deviation in the user's behavioral model entails a modification of the point cloud structure (a change in geometry in the data), which, in turn, manifests itself through noticeable shifts in topological properties. In this context, Betti curves act as a tool for visualizing the dynamics of these changes, providing a mapping of the metamorphoses of homology groups — key topological invariants.

The comparison of the obtained Betti curves with the reference profile generated according to a single methodology is performed using the Wasserstein and L_1 metrics. The evaluation process following the topological analysis requires a multidimensional approach to decision making, which involves the use of complex evaluation criteria — the construction of a generalized indicator. Within the framework of this task, we have chosen a methodology for constructing a generalized desirability indicator developed by E.K. Harrington [15].

Table 1

Desirability scale

Gradation names (linguistic meanings)	Numerical intervals
Vary bad	0—0.2
Bad	0.2—0.37
Acceptable	0.37—0.63
Good	0.63—0.8
Very good	0.8—1

The generalized Harrington desirability function provides a mechanism for converting complex topological characteristics into homogeneous numerical parameters, which greatly simplifies further interpretation and analysis. The application of this approach allows not only to more accurately assess the scale of deviations, but also to optimize the comparison process, making it more transparent and accessible to perception.

To determine the estimate, a "desirability curve" (one of Harrington's logistic functions) is used, given as follows [13]:

$$d \left(x \right) = \exp \left[-\exp \left(-y \left(x \right) \right) \right],$$

where $y \left(x \right)$ represents the encoded values of individual characteristics (scalar value), and x is a variable indicating the level or value of each characteristic. The x-axis is interpreted as a scale of individual indicators, and the d-axis is interpreted as a desirability scale, divided into five discrete ranges that determine the degrees of deviation.

The generalized desirability index is calculated as a geometric mean according to the formula [14]:

$$D = \sqrt[m]{\prod\nolimits_{i=1}^{m} d_i},$$

where m is the number of individual quality criteria; d_i is the individual scores for each criterion.

The choice of the geometric mean in constructing a generalized desirability indicator is because the geometric mean plays the role of a "smoothing" mechanism that reduces the effect of random fluctuations in estimates and provides a more stable and adequate overall picture of quality [15].

Thus, the mechanism of forming a generalized desirability index works as a highly sensitive filter that allows to identify significant deviations and evaluate them by a comprehensive desirability scale.

Experimental results

In the framework of the study, the data obtained from the Supercomputer center Polytechnic and presented as a source of information on the effectiveness of performing computational tasks were selected as the analyzed data.

The studied dataset contains 1545793 records of launched tasks and their execution results. A task may be completed successfully or it may not be completed due to a user error or lack of the requested execution time. Information about each task contains the number of requested resources (processors and supercomputer nodes), as well as the results of the task, including how many and what resources were issued, when and how the task was completed. In addition, 10 problem areas were identified: astrophysics, bioinformatics, biophysics, energetics, geophysics, IT, mechanical engineering, mechanics, physics, and radiophysics. Each task belongs to one of them.

Data analysis was made based on the following job parameters from the dataset:

1. ReqNodes — Requested minimum amount of nodes for the job/step.
2. ReqCPUS — Number of requested CPUs.
3. CPUTimeRAW — Time used (Elapsed time * CPU count) by a job or step in CPU-seconds.
4. ElapsedRaw — Job's elapsed time in seconds.
5. AllocNodes — Number of nodes allocated to the job/step. 0 if the job is pending.
6. AllocCPUS — Count of allocated CPUs.
7. TimelimitRaw — What the time limit was/is for the job. Format is in number of minutes.
8. Priority — Slurm priority.
9. Partition — Partition on which the job ran (the name of the cluster, e.g., Tornado).

This data combines user requests for computing resources with detailed performance metrics in the form of time series, including key parameters such as the requested and actual task execution time, the amount of processor time used, the degree of launch success, and other critical performance indicators.

Fig. 1 provides a visual representation of the multidimensional nature of these time series for one particular user. On the graph, each point represents a snapshot of the system state for one of the tasks — thus, a sequence of 15 such points reflects the results of the analysis for 15 sequentially completed tasks.

For each user, data is extracted and examined individually, taking into account all characteristics as part of a single multidimensional time series, using an approach where the transformation is carried out using a Takens embedding based on an algorithm for selecting parameters based on L-statistics. Due to the complexity of time series that require embedding in a space with a dimension of at least $2n + 1$, a direct visual representation of this point cloud becomes unrealizable at dimensions $d > 3$.

However, using the theory of persistent homology, we transform these point clouds into analytically controlled information. We create filtered Vietoris—Rips complexes that allow to extract persistent homology and construct average Betti curves for each user. This process forms a unique "topological portrait" of the user, which reflects his characteristics and behavior within the framework of computational tasks [16].
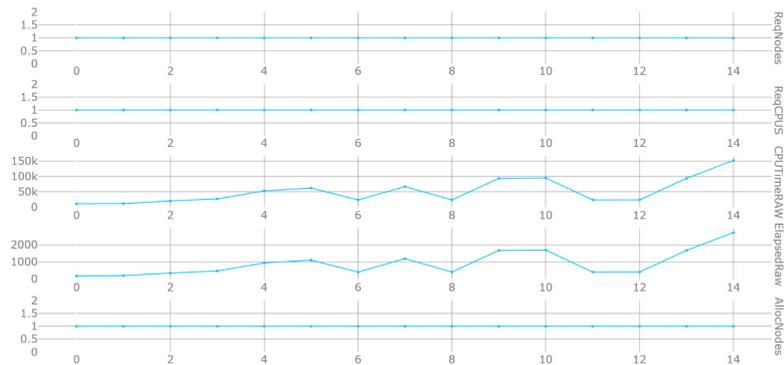
Fig. 1. Time series
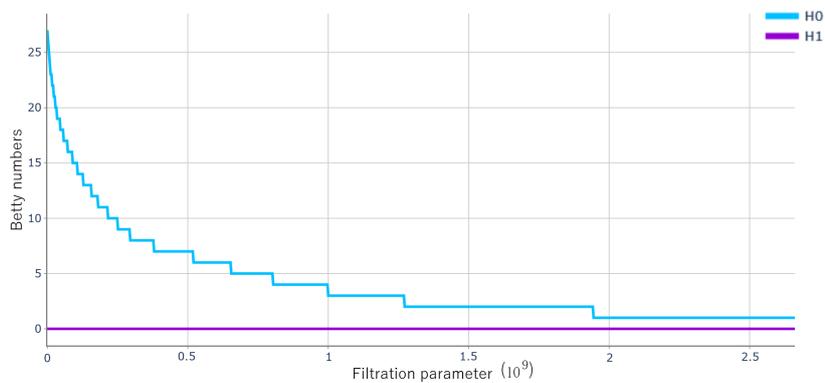
Fig. 2. Betti curves of the user profile

Fig. 2 illustrates an example of such a user profile in the form of Betti curves, where each step on the curve reflects changes in the topological structure of user data, taking into account different levels of detail and time scales.

To create a reference profile of the "ideal" user, we used a time series of the most successful users selected according to two criteria: more than 95% successful completion of tasks and a minimum deviation in the use of resources from the stated needs. The results of this analysis are presented in Fig. 3, showing Betti curves reflecting optimal topological behavior characteristics.

Table 2 shows the results of evaluating the comparison of several user profiles with a reference profile.

Thus, the proposed methodology allows to compare current user behavior with historical data as well as to provide a quantitative assessment of their behavioral effectiveness, the basis for determining the level of trust in each user.

However, it is worth noting the limitation of this approach: for new users, it is required to collect a sufficient amount of initial data to accurately build their profile and reliable assessment. A lack of initial data can make it difficult to accurately model behavior and leads to inaccuracies in the assessment.

Conclusion

Within the framework of this study, an algorithm based on topological data analysis is proposed, which builds user profiles of a supercomputer center with acceptable computational complexity using simple and effective procedures for identifying behavioral patterns. This approach demonstrates a wide
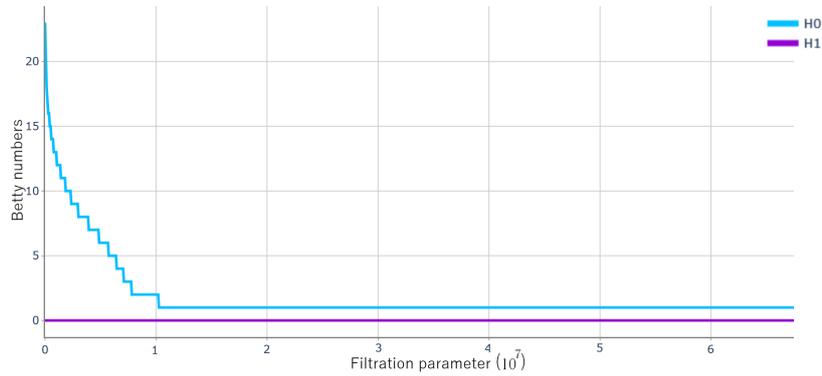
Fig. 3. Betti curves of the reference profile

Table 2

User profile ratings

User	Desirability assessment		User performance	
	Numerical evaluation	Linguistic value	Percentage of successfully completed tasks	Estimation error (average absolute logarithmic error)
0	0.115	Very bad	0.369	3.875
1	0.824	Very good	0.698	2.838
2	0.339	Bad	0.33	2.556
3	0.085	Very bad	0.001	4.261
4	0.787	Good	0.611	3.06

potential in the field of detection and evaluation of deviations. Experimental verification of the method based on real data from the SCC Polytechnic confirmed the high applicability of the methodology in the context of the development of intelligent task allocation management systems.

This approach helps to increase the efficiency of using the resources of supercomputer platforms, ensuring optimal allocation of tasks and reducing unnecessary costs, ensuring fairness and transparency of access to resources for all users, based on their behavioral characteristics. It also potentially provides tools for predicting system load and strengthens the security of platforms, allowing timely detection of abnormal activity and prevention of possible threats.

The development of adaptive dispatch systems that would take into account the dynamics of user behavior and quickly respond to changes in their requests and needs, and the integration of the developed technology with existing supercomputer infrastructure management systems, which will enhance their functionality and increase the efficiency of resource management, are promising for the development of this method.

REFERENCES

1. **Hou Z., Shen H., Zhou X. et al.** Prediction of job characteristics for intelligent resource allocation in HPC systems: a survey and future directions. Frontiers of Computer Science, 2022, Vol. 16, No. 5, Article no. 165107. DOI: 10.1007/s11704-022-0625-8

2. **Uchroński M., Bożejko W., Krajewski Z., Tykierko M., Wodecki M.** User estimates inaccuracy study in HPC scheduler. Contemporary Complex Systems and Their Dependability: Proceedings of the 13th

International Conference on Dependability and Complex Systems (DepCoS-RELCOMEX), 2019, Pp. 504−514. DOI: 10.1007/978-3-319-91446-6_47

3. **Zaborovsky V.S., Utkin L.V., Muliukha V.A., Lukashin A.A.** Improving Efficiency of Hybrid HPC Systems Using a Multi-agent Scheduler and Machine Learning Methods. Supercomputing Frontiers and Innovations, 2023, Vol. 10, No. 2, Pp. 104−126. DOI: 10.14529/jsfi230207

4. **Martín A.G., Beltrán M., Fernández-Isabel A., Martín de Diego I.** An approach to detect user behavior anomalies within identity federations. Computers & Security, 2021, Vol. 108, Article no. 102356. DOI: 10.1016/j.cose.2021.102356

5. **Martín A.G., Fernández-Isabel A., Martín de Diego I. et al.** A survey for user behavior analysis based on machine learning techniques: current models and applications. Applied Intelligence, 2021, Vol. 51, No. 8, Pp. 6029−6055. DOI: 10.1007/s10489-020-02160-x

6. **Chazal F., Michel B.** An introduction to topological data analysis: Fundamental and practical aspects for data scientists. Frontiers in Artificial Intelligence, 2021, Vol. 4, Article no. 667963. DOI: 10.3389/frai.2021.667963

7. **Yan H., Yang N., Peng Y., Ren Y.** Data mining in the construction industry: Present status, opportunities, and future trends. Automation in Construction, 2020, Vol. 119, Article no. 103331. DOI: 10.1016/j.autcon.2020.103331

8. **Seversky L.M., Davis S., Berger M.** On time-series topological data analysis: New data and opportunities. 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2016, Pp. 1014−1022. DOI: 10.1109/CVPRW.2016.131

9. **Choi K., Yi J., Park C., Yoon S.** Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines. IEEE Access, 2021, Vol. 9, Pp. 120043−120065. DOI: 10.1109/ACCESS.2021.3107975

10. **Taieb S.B., Bontempi G., Atiya A.F., Sorjamaa A.** A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. Expert Systems with Applications, 2012, Vol. 39, No. 8, Pp. 7067−7083. DOI: 10.1016/j.eswa.2012.01.039

11. **Tan E., Algar S., Corrêa D., Small M., Stemler T., Walker D.** Selecting embedding delays: An overview of embedding techniques and a new method using persistent homology Chaos, 2023, Vol. 33, No. 3, Article no. 032101. DOI: 10.1063/5.0137223

12. **Uzal L.C., Grinblat G.L., Verdes P.F.** Optimal reconstruction of dynamical systems: A noise amplification approach. Physical Review E, 2011, Vol. 84, No. 1, Article no. 016223. DOI: 10.1103/PhysRevE.84.016223

13. **Leykam D., Angelakis D.G.** Topological data analysis and machine learning. Advances in Physics: X, 2023, Vol. 8, No. 1, Article no. 2202331. DOI: 10.1080/23746149.2023.2202331

14. **Mezheneva I.O., Lukashin A.A.** Topological descriptors for data analysis in behavioral analytics systems. Matematicheskie metody v tekhnike i tekhnologiiakh [Mathematical methods in technology and engineering], 2023, Vol. 10, Pp. 104−108. DOI: 10.52348/2712-8873_MMTT_2023_10_104

15. **Thorisingam Y., Mustafa M.S.** Augmented approach to desirability function based on principal component analysis. International Journal of Academic Research in Progressive Education and Development, 2022, Vol. 11, No. 2. DOI: 10.6007/IJARPED/v11-i2/13270

16. **Nashivochnikov N.V., Pustarnakov V.F.** Topological methods of analysis in behavioral analytics systems. Voprosy kiberbezopasnosti, 2021, Vol. 2, No. 42, Pp. 26−36. DOI: 10.21681/2311-3456-2021-2-26-36

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Mezheneva Irina O.
Меженева Ирина Олеговна
E-mail: Mezheneva-I@gaz-is.ru

Lukashin Alexey A.
Лукашин Алексей Андреевич
E-mail: lukash.spb.ru@gmail.com

Chatoyan Sergey K.
Чатоян Сергей Камалович
E-mail: Chatoyan-S@gaz-is.ru

MODEL OF A SUPERCOMPUTER CLUSTER
IN THE FORM OF A QUEUEING SYSTEM WITH A RANDOM LIMIT
ON THE EXECUTION TIME OF APPLIED TASKS

O.I. Zayats, V.E. Baksheev,
V.S. Zaborovsky, V.A. Muliukha ✉ ⓘD

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

✉ vladimir.muliukha@spbstu.ru

Abstract. It is well known that the efficiency of task dispatching in any supercomputer system is determined, first of all, by the adequacy of the system model used, as well as the accuracy of the estimation of the parameters of the model itself. The article proposes a new version of the supercomputer cluster model, based on a standard model of the $M/M/\infty$ class queueing system, which is supplemented with two fundamental clarifications that reflect the features of the supercomputer operation. First, the processing time of each task is limited by the dispatcher using a random variable distributed according to the exponential law. Second, it is considered that each new task requires the allocation of a random number of service channels (processors) for its execution. The parameters of the proposed queueing model are estimated based on statistical processing of data obtained during calculations previously performed on a supercomputer. A number of examples of using the developed model are given. To calculate the parameters of the queueing system, it is proposed to use the method of generating functions.

Keywords: queueing system, task dispatching, supercomputer, random execution time, random number of service channels

МОДЕЛЬ СУПЕРКОМПЬЮТЕРНОГО КЛАСТЕРА В ВИДЕ СИСТЕМЫ МАССОВОГО ОБСЛУЖИВАНИЯ СО СЛУЧАЙНЫМ ОГРАНИЧЕНИЕМ ВРЕМЕНИ ВЫПОЛНЕНИЯ ПРИКЛАДНЫХ ЗАДАЧ

О.И. Заяц, В.Е. Бакшеев,
В.С. Заборовский, В.А. Мулюха ✉ ⓘD

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

✉ vladimir.muliukha@spbstu.ru

Аннотация. Хорошо известно, что эффективность диспетчеризации задач в любой суперкомпьютерной системе определяется, прежде всего, адекватностью используемой модели системы, а также точностью оценки параметров самой этой модели. В статье предлагается новая версия модели суперкомпьютерного кластера, основанная на типовой модели системы массового обслуживания класса M/M/∞, которая дополнена двумя принципиальными уточнениями, отражающими особенности функционирования суперкомпьютера. Во-первых, время обработки каждого задания ограничивается диспетчером с помощью некоторой случайной величины, распределенной по показательному закону. Во-вторых, считается, что каждая новая задача требует для своего выполнения выделения ей случайного числа каналов обслуживания (процессоров). Параметры предложенной модели массового обслуживания оцениваются на основе статистической обработки данных, полученных в ходе расчетов, ранее выполненных на суперкомпьютере. Приводятся ряд примеров использования разработанной модели. Для расчета параметров системы массового обслуживания предлагается использовать метод производящих функций.

Ключевые слова: система массового обслуживания, распределение задач, суперкомпьютер, случайное время выполнения, случайное число каналов обслуживания

Introduction

Analytical research of supercomputer systems is of significant theoretical and practical interest. Currently, it is generally accepted that when modeling computer systems, the use of queueing theory is an adequate apparatus [1, 2]. Supercomputer systems (computer clusters) are extremely complex technical devices. If we try to model such a device as accurately as possible with all the smallest nuances of its behavior, the resulting mathematical model turns out to be very complex and, as a rule, makes its detailed analytical study very difficult. Therefore, it seems reasonable to start with a study of a simplified model,

which so far takes into account only the most important, fundamental factors that reflect the very essence of the phenomenon under study.

The queueing model proposed in this article, designed to describe the behavior of a supercomputer, takes into account two such factors that are fundamentally important for its functioning. First, it is necessary to take into account that each newly received service request may require a random number of service channels (processors) for its execution. Second, the execution time of each service request is limited in a certain way by the task manager. We will examine in more detail each of the two above-mentioned assumptions.

Queueing systems with a branching process for executing requests, in which each request is processed by several servers at once, form a special new class [3, 4]. In the English-language scientific literature such systems are called "the fork-join queueing systems". There is no generally accepted terminological analogue in Russian-language literature yet. Some authors [3, 4] suggest using a term in Russian that is translated in English as "parallel serving systems".

In this article we will consider a parallel processing system in which each service request is split into a random number of subrequests. Such queueing systems began to be studied quite a long time ago, back in the early 1980s [5, 6]. The general idea of the functioning of a fork-join system is as follows. Its input receives a random stream of calls (requests). At the time of receipt, any request is divided into a random number of smaller related requests (subrequests), each of which can be processed by one of the system servers.

Currently, a large number of works concerning various aspects of the study of fork-join queueing systems have already been published. They can be found, for example, using a detailed review by A. Thomasian [7] or the recently published monograph by S. Sethuraman [8]. Unfortunately, all these works do not take into account such a practically significant factor as the presence of a limitation on the time for executing requests by the service channel. Meanwhile, when using these models to describe the behavior of a supercomputer, the presence of such restrictions is fundamentally important. For example, according to the "Polytechnic Supercomputer Center" of Peter the Great St. Petersburg Polytechnic University, up to 70% of tasks are removed from calculations ahead of schedule by the dispatcher program.

Restrictions on the execution time of applied tasks on a supercomputer cluster are formed in a rather complex way. The approximate task execution time specified by the user, the dispatch algorithms underlying the work of the task scheduler, and the system of priorities and push-out mechanisms used play a role here. In practice, the limitation on the time it takes to complete tasks appears as some random variable.

Description of the mathematical model of the queueing system

In this section, we will describe in more detail the queueing system under study, the functional diagram of which is presented in Fig. 1. This system works as follows. The system input receives the simplest (that is, stationary Poisson) flow of requests with λ intensity. In such a flow, all intervals τ_k between request are independent and distributed according to the same exponential law with probability density

$$a(\tau) = \lambda e^{-\lambda \tau}, \tag{1}$$

so that the average interval between requests $\overline{\tau}$ is inversely proportional to the intensity of the flow

$$\overline{\tau} = \frac{1}{\lambda}. \tag{2}$$

A fundamentally important distinctive feature of our system is that it is multi-channel, and contains an unlimited number of service channels (servers), and each request entering it requires servicing
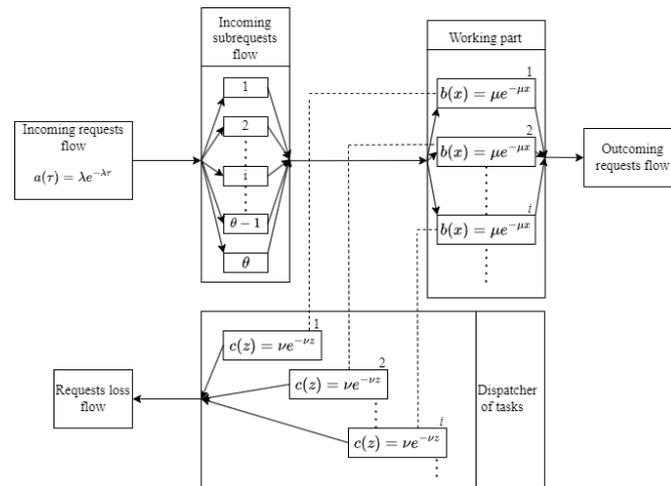
Fig. 1. Functional diagram of the queueing system under study

simultaneously by a random number of servers. The process of executing request can be represented as follows. First, any request can be divided into a number of smaller elementary requests (subrequests), each of which can be executed using only one separate server.

Let us denote the random number of subrequests into which one complete request is divided by the symbol θ. The variable θ is a discrete integer random variable, which is characterized by the following distribution series:

$$\alpha_i = \mathrm{P}\{\theta = i\}, \ \left(i = \overline{1, \infty}\right). \tag{3}$$

Without loss of generality, we assume $\alpha_0 \equiv 0$. Here it is assumed that the values θ related to different requests are statistically independent and that all of them do not depend on the intervals between the requests appearance τ_k.

All service channels (servers) are considered identical, and the execution time of any elementary task (subrequest) on each of them is distributed according to an exponential law with the parameter μ. This means that the probability density of service time X is expressed as

$$b(x) = \mu e^{-\mu x}, \tag{4}$$

and average service time $\overline{x}$ is given by the formula

$$\overline{x} = \frac{1}{\mu}. \tag{5}$$

The parameter μ has the meaning of service intensity.

Our model introduces another important complication that distinguishes it from standard models of queueing systems. The task execution time is limited by the task manager. This dispatcher (special program) issues a constraint in the form of a random variable Z, distributed according to a given law with a known probability density $c(z)$. If the inequality $Z > X$ is satisfied, then the service process is considered successfully completed. When the opposite inequality is satisfied, $Z < X$, the task is removed from execution and sent to the loss flow. In the favorable case, when all the elementary tasks that make up the complete request have been successfully completed, the request as a whole is also considered completed. Otherwise, the entire request is rejected and sent to the loss flow.

This article will examine in more detail the case when the upper bound on the service time is distributed according to the exponential law of the form

$$c(z) = \nu e^{-\nu z}, \tag{6}$$

where ν is the intensity of tasks being removed from execution by the dispatcher. In this case, the average time that dispatcher provided to each task for execution is expressed as

$$\overline{z} = \frac{1}{\nu}. \tag{7}$$

The numeric parameter $\overline{z}$ is one of the most important parameters for the task manager and can be set accordingly.

As a result of the system operating in accordance with the service process discipline described above, the incoming flow of requests is divided into two new flows: the outgoing flow of fully serviced requests, as well as a loss flow, including requests removed from processing by the dispatcher of tasks. It should be mentioned that the calculation of the loss probability, that is, the probability getting into the second stream, represents an important practical problem.

Obtaining a system of Kolmogorov equations

We will characterize the state of the system described in the previous section using the number of servers (service channels) $N(t)$ occupied at time t. The indicated process is Markov process [9]. Let us introduce the probabilities of the states of the considered process

$$P_n(t) = \mathcal{P}\{N(t) = n\}, \ \left(n = \overline{0, \infty}\right). \tag{8}$$

Process $N(t)$ is ergodic [9], therefore there are final probabilities

$$P_n = \lim_{t \to \infty} P_n(t), \ \left(n = \overline{0, \infty}\right) \tag{9}$$

that do not depend on the initial state.

The labeled state graph for the process considered has the form shown in Fig. 2. In order not to clutter the figure, it shows in detail the picture of transitions only for three states: $n = 0$, $n = 1$ and an arbitrary $n > 0$. From the transition diagram it is clear that from the state $n = 0$ you can go to the state $n > 0$, located *in the right part of the figure*, with intensity $\lambda \alpha_n$, and you can return to the state $n = 0$ only from the state $n = 1$, and with intensity μ.

A state with an arbitrary $n > 0$ can be reached from any state i located to the left of n, with intensity $\lambda \alpha_{n-i}$, since a new service request is allowed, designed to use an arbitrary number of servers. Similarly, from the state $n > 0$ you can reach any state $i > n$, located to the right of n, with intensity $\lambda \alpha_{i-n}$. In this case, the total intensity of the transition to all states lying to the right of n will obviously be equal to λ, since the distribution (3) the normalization conditions is always satisfied, which has the form

$$\sum_{i=1}^{\infty} \alpha_i = 1. \tag{10}$$

The only way to return to the state n on the right is to move back from state $(n+1)$ with intensity $(\mu + \nu)(n+1)$.

Using the labeled state graph in Fig. 2, we can write the Kolmogorov system of equilibrium equations according to the usual rules [9]:
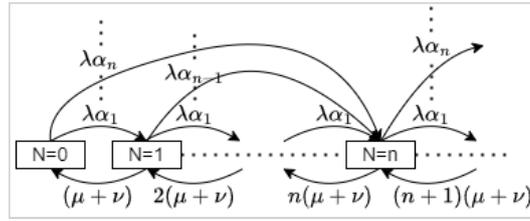
Fig. 2. Labeled state graph for the queueing system under consideration

$$
\begin{cases}
-\lambda p_0 + (\mu + \nu)\, p_1 = 0, \ \ n = 0, \\
-\left[\lambda + (\mu + \nu)n\right] p_n + (\mu + \nu)(n+1)\, p_{n+1} + \sum_{i=0}^{n-1} \lambda \alpha_{n-i} p_i = 0, \ \ n > 0.
\end{cases} \tag{11}
$$

Equations (11) express the balance of random flows that occurs when the system occupies state number n. In this case, terms with a minus sign are equal to the intensity of flows leaving state n, and terms with a plus sign correspond to all possible flows entering this state. Then physical meaning of the equilibrium equations (11) is that in a steady state, for all states of the system, the intensity of incoming flows must be equal to the intensity of outgoing flows.

Equations (11) can be rewritten in a universal form, if we introduce the concept of "empty sum". A sum is called empty, if its lower summation limit is greater than its upper summation limit. This amount is considered to be zero by definition. Then equations (11) can be rewritten as one equation valid for all $n \geq 0$:

$$
-\left[\lambda + (\mu + \nu)n\right] p_n + (\mu + \nu)(n+1)\, p_{n+1} + \sum_{i=0}^{n-1} \lambda \alpha_{n-i} p_i = 0, \ \left(n = \overline{0, \infty} \right). \tag{12}
$$

Note, that when $n = 0$, the third term in (12) is an empty sum.

Writing equations in the form (12) is more convenient, when using the method of generating functions, which will be discussed below.

At the end of this section, an important note should be made. In our system, the total intensity at which request processing on the server stops is equal to $\mu + \nu$. From the point of view of the final probabilities of the state, it does not matter at all with what intensity this request will be fully serviced, and with what intensity it will remain unserved, since it will be removed from service by the dispatcher. If the total intensity of termination of processing requests $\mu + \nu$ is given, then the form of the Kolmogorov equations and the values of the final state probabilities are also specified uniquely.

Therefore, for example, in a system without a dispatcher but with the same $\mu + \nu$ service intensity as before, the final probabilities will be the same as in our system. Of course, the loss probabilities in the two above-mentioned systems will be completely different. A special section of this article will be devoted to calculating the probability of losses.

The generating functions method

The concept of generating functions is a powerful tool for solving problems involving the analysis of numerical sequences, such as the sequence $\{p_n\}_{n=0}^{\infty}$ of state probabilities in our case. The idea of the method is to move from considering an infinite set of variables p_n depending on an integer index n, to a single function depending on a continuously changing argument.

Let us introduce the generating function $G(z)$ for probabilities p_n in the form of the following power series

$$G(z) = \sum_{n=0}^{\infty} p_n z^n, \ \left(|z| \le 1\right). \tag{13}$$

The function G is guaranteed to exist at least in region $|z| \le 1$, since the series

$$\sum_{n=0}^{\infty} p_n = 1 \tag{14}$$

is undoubtedly convergent.

To calculate the function $G(z)$, multiply both sides of equality (12) by z^n and sum over all n from zero to infinity. The resulting sums are converted as follows:

$$\sum_{n=0}^{\infty} n p_n z^n = z \sum_{n=0}^{\infty} p_n n z^{n-1} = z \frac{dG(z)}{dz}, \tag{15}$$

$$\sum_{n=0}^{\infty} (n+1) p_{n+1} z^n = \sum_{n=1}^{\infty} n p_n z^{n-1} = \frac{dG(z)}{dz}. \tag{16}$$

Double sum is calculated by changing the order of summation

$$\sum_{n=0}^{\infty} \sum_{i=0}^{n-1} \alpha_{n-i} p_i z^n = \sum_{n=1}^{\infty} \sum_{i=0}^{n-1} \alpha_{n-i} p_i z^n = \sum_{i=0}^{\infty} \sum_{n=i+1}^{\infty} \alpha_{n-i} p_i z^n =$$
$$= \sum_{i=0}^{\infty} p_i z^i \sum_{n=i+1}^{\infty} \alpha_{n-i} z^{n-i} = \sum_{i=0}^{\infty} p_i z^i \sum_{k=1}^{\infty} \alpha_k z^k = G(z) Q(z). \tag{17}$$

Here by $Q(z)$ we mean the generating function of the α_i probabilities

$$Q(z) = \sum_{k=1}^{\infty} \alpha_k z^k, \ \left(|z| \le 1\right). \tag{18}$$

After these transformations, equation (12) takes the form

$$-\lambda G - (\mu + \nu) z \frac{dG}{dz} + (\mu + \nu) \frac{dG}{dz} + \lambda G Q = 0. \tag{19}$$

Expressing the derivative $\dfrac{dG}{dz}$ from here, we obtain the following differential equation for the function $G(z)$:

$$\frac{dG}{dz} = \frac{\lambda}{\mu + \nu} \frac{1 - Q(z)}{1 - z} G. \tag{20}$$

It should be noted that the function $Q(z)$ here is known, since all probabilities α_i are specified according to the conditions of the problem.

Equation (20) must be solved under the initial condition

$$G(1) = 1, \tag{21}$$

which is a universal general property of all generating functions, resulting from the normalization condition (14) for the probabilities p_n.

Solving equation (20) under initial condition (21) we finally obtain:

$$G(z) = e^{\frac{\lambda}{\mu+\nu} \int_1^z \frac{1-Q(s)}{1-s} ds}. \tag{22}$$

Now let us look at some examples of applying the solution we just obtain.

Example 1. Each request is processed by only one server.

In this case, the probabilities α_i are given in the form

$$\alpha_i = \delta_{i,1}, \ \left(i = \overline{1,\infty}\right), \tag{23}$$

where $\delta_{i,k}$ denotes the Kronecker delta symbol, and the generating function (18) is reduced to the simplest linear function

$$Q(z) = z. \tag{24}$$

Then solution (22) is nothing more than an exponential of the form

$$G(z) = e^{\frac{\lambda}{\mu+\nu}(z-1)}, \tag{25}$$

expanding which in powers of z we get

$$p_n = e^{-\frac{\lambda}{\mu+\nu}} \left(\frac{\lambda}{\mu+\nu}\right)^n, \ \left(n = \overline{0,\infty}\right). \tag{26}$$

Thus, if each request is processed by only one server, then the total number of busy servers will be distributed according to Poisson's law with the parameter $\dfrac{\lambda}{\mu+\nu}$. This result is well known from the classical queueing theory as applied to the system of $M/M/\infty$ class [9].

Example 2. Geometric law of distribution of the number of involving servers.

Let us assume that the number θ of servers used to process a single request is distributed according to a geometric law

$$\alpha_i = \mathcal{P}\{\theta = i\} = (1-\text{æ})\text{æ}^{i-1}, \ \left(i = \overline{1,\infty}\right), \tag{27}$$

where $0 < \text{æ} < 1$ denotes the parameter of the geometric law. The geometric law is interesting because it is the only discrete law that has the property of no aftereffect [9].

It is easy to show that the generating function (18) of the following form corresponds to law (27):

$$Q(z) = \frac{(1-\text{æ})z}{1-\text{æ}z}, \tag{28}$$

wherein

$$1 - Q(z) = \frac{(1-z)}{1 - \text{æ}z}. \tag{29}$$

Substituting expression (29) into formula (22), we obtain

$$G(z) = e^{\frac{\lambda}{\mu+\nu} \int_1^z \frac{ds}{1-\text{æ}s}}. \tag{30}$$

The integral in the exponent (30) is tabular. Omitting a number of simple intermediate calculations, we get

$$G(z) = \left(\frac{1-\text{æ}}{1-\text{æ}z} \right)^{\frac{\lambda}{(\mu+\nu)\text{æ}}}. \tag{31}$$

Example 3. Average number of busy servers.

Using the general formula (20) for the generating function, one can easily find the average number of busy servers in the entire system in steady state. It is well known, that

$$\overline{n} = M[N] = \frac{dG(z)}{dz} \bigg|_{z=1}, \tag{32}$$

also it's obvious that

$$\lim_{z \to 1} \frac{1 - Q(z)}{1 - z} = Q'(1) = \overline{\theta}, \tag{33}$$

where $\overline{\theta}$ denotes the average number of servers per processed request.

Passing to the limit as $z \to 1$ in formula (20) leads us to the final expression

$$\overline{n} = \frac{\lambda}{\mu + \nu} \overline{\theta}. \tag{34}$$

Thus, the average number of busy servers in the system is directly proportional to the arrival rate and the average number of servers per request and inversely proportional to the sum of the execution and reset by the dispatcher intensities.

Calculating the probability of losing a service request

In the classical $M/M/\infty$ system, loss of requests is impossible, because such a system is an immediate queueing system with an infinite number of servers. For any newly received request for service, there is always a free server that begins to process it, and the service is always brought to its logical end [9].

In our system, processing also starts immediately after the arrival of request, but it may not be completed due to the intervention of the dispatcher (task scheduler), which dumps partially completed tasks into the loss flow. As a result, each request leaving the system can end up in one of two flows: either in the outcoming flow of fully serviced requests, or the loss flow formed by requests removed from service. In this section we will calculate the probability of being in the first of these flows.

Let us denote by A the random event that the request is eventually fully serviced. We denote the probability of event A by P_{serv}. Using the total probability formula, we can write

$$P_{serv} = \sum_{k=1}^{\infty} \mathcal{P}\{A|\theta = k\}\mathcal{P}\{\theta = k\}. \tag{35}$$

Here θ is the random number of servers that will be required to process the request in question. The first factor in each term of the sum (35) has the meaning of the conditional probability of event A, provided that exactly k servers are required, and the second factor, according to (3), is equal to α_k.

Let us calculate the conditional probabilities appearing in (35). To do this, we first find the partial probability p that one server will successfully complete the processing of the subrequest assigned to it. It's not hard to understand that

$$p = \mathcal{P}\{X < Z\}, \tag{36}$$

where X is the execution time of the subrequest, and Z is the limitation on this time on the part of the dispatcher.

As a result, we obtain a simple expression for the probability (35)

$$P_{serv} = \sum_{k=1}^{\infty} \alpha_k p^k, \tag{37}$$

in which p remains unknown for now.

To calculate the probability p, we introduce a joint law of distribution of random variables X and Z, which we denote by $f_{xz}(x,z)$. According to the assumptions made at the beginning of this article, we get

$$f_{xz}(x,z) = f_x(x)f_z(z) = b(x)c(z) = \mu\nu e^{-(\mu x + \nu z)}. \tag{38}$$

Probability (36) is represented as an integral

$$p = \int_0^{\infty}\int_x^{\infty} f_{xz}(x,z)\,dz\,dx, \tag{39}$$

which, due to the fulfilment of (38), can be easily calculated

$$p = \frac{\mu}{\mu + \nu}. \tag{40}$$

Substituting this expression into (37), we obtain

$$P_{serv} = \sum_{k=1}^{\infty} \alpha_k \left(\frac{\mu}{\mu + \nu}\right)^k = Q\left(\frac{\mu}{\mu + \nu}\right), \tag{41}$$

where $Q(z)$ is the generating function (18) for the probabilities α_k, describing the distribution of the number θ of servers involved in one complete request processing.

If we use the original formulas (5) and (7), then the expression (40) can be rewritten in the equivalent form

$$p = \frac{\overline{z}}{\overline{z} + \overline{x}}, \tag{42}$$
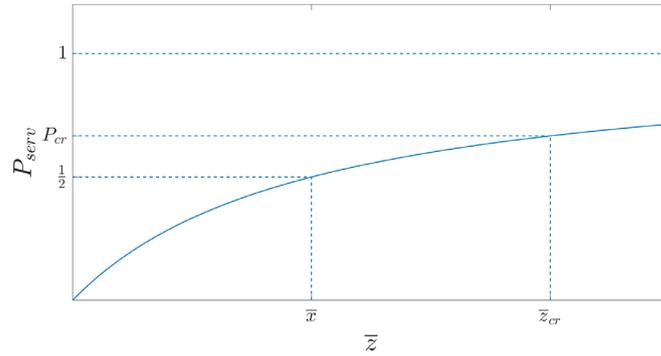
Fig. 3. Dependence of probability P_{serv} on the average time of limitation $\overline{z}$ for Example 4 for $\overline{x} = 1$

which gives an alternative expression for the desired probability

$$P_{serv} = Q\left(\frac{\overline{z}}{\overline{z} + \overline{x}}\right). \tag{43}$$

Now let us consider some examples that explain the application of the resulting formulas.

Example 4. Requests without subrequests.

In Example 1 the probabilities are given in the form (23), which leads to the generating function given (24), and then for the probability of successfully completing tasks we get the simplest expression

$$P_{serv} = \frac{\overline{z}}{\overline{z} + \overline{x}}. \tag{44}$$

The dependence of probability (44) on the average time of limitation is shown in Fig. 3. The graph shows that when inequality $\overline{z} > \overline{x}$ is satisfied, the probability of successful completion of tasks will be greater than the probability of their early reset. With the opposite sign of inequality $\overline{z} < \overline{x}$, on the contrary, the reset will, on average, occur more often than the normal standard completion of calculations.

If some critical value for probability (44) is specified in the form P_{cr} and it is required that probability P_{serv} exceed it, then $\overline{z}$ must be higher than the critical value

$$z_{cr} = \overline{x}\,\frac{P_{cr}}{1 - P_{cr}}. \tag{45}$$

Example 5. Requests that are divided into a fixed nonrandom number of subrequests.

Let us assume that the probabilities α_i are expressed in the form

$$\alpha_i = \delta_{i,k}, \tag{46}$$

where k is some integer positive, so that the generating function (18) turns out to be equal to the specified integer power of z

$$Q(z) = z^k. \tag{47}$$

Then using formula (43) we get
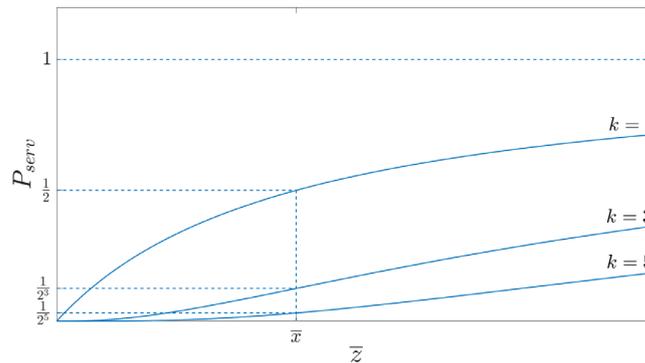
Fig. 4. Dependence of probability P_{serv} on the average time of limitation $\overline{z}$
for Example 5 for $k = 1, 3, 5$ and $\overline{x} = 1$

$$P_{serv} = \left(\frac{\overline{z}}{\overline{z} + \overline{x}} \right)^k. \tag{48}$$

A graph of dependence of P_{serv} on $\overline{z}$ for different k, relating to this case, is shown in Fig. 4.

The behavior of the curves in Fig. 4 shows that an increase in the number of servers involved leads to a decrease of the probability of successful completion of tasks.

Conclusion

In this article, we used an appropriately modified classical model of the $M/M/\infty$ queueing system to mathematically describe the behavior of a supercomputer cluster. In this model, each service request needs to use a random number of servers (processors) simultaneously. Service occurs according to an exponential law, identical for all running servers. The incoming flow is assumed to be the simplest (stationary Poisson). The execution time of each service request is limited by the dispatcher (task scheduler) by a certain random variable distributed according to an exponential law. If during this time the task has not yet been completed, then the task is discarded from service and falls into the loss flow. The paper provides an analytical solution to the described problem using the method of generating functions. The main contribution of this article is that it explicitly obtained the law of distribution of the number of busy servers, as well as the probability of successful completion of tasks in the form of a certain function of the average time limitation. The probability that any task will be completed to its logical end is one of the main indicators of the quality of a computing cluster.

REFERENCES

1. **Vishnevskii V.M.** Teoreticheskie osnovy proektirovaniia komp'iuternykh setei [Theoretical foundations of computer network design]. Moscow: Tekhnosfera, 2003, 506 p. (in Russ.)

2. **Harchol-Balter M.** Performance modeling and design of computer systems: Queueing theory in action. Cambridge: University Press, 2013, 576 p.

3. **Gorbunova A.V., Zaryadov I.S., Samouylov K.E., Sopin E.S.** A Survey on Queuing Systems with Parallel Serving of Customers. Discrete and Continuous Models and Applied Computational Science. 2017, Vol. 25, no. 4, pp. 350−362. DOI: 10.22363/2312-9735-2017-25-4-350-362

4. **Gorbunova A.V., Zaryadov I.S., Samouylov K.E.** A Survey on Queuing Systems with Parallel Servingof Customers. Part II. Discrete and Continuous Models and Applied Computational Science. 2018, Vol. 26, no. 1, pp. 13−27. DOI: 10.22363/2312-9735-2018-26-1-13-27

5. **Green L.** A queueing system in which customers require a random number of servers. Operations Research. 1980, Vol. 28, no. 6, pp. 1335−1346.

6. **Green L.** Comparing operating characteristics of queues in which customers require a random number of servers. Management science. 1981, Vol. 27, no. 1, pp. 65−74.

7. **Thomasian A.** Analysis of fork/join and related queueing systems. ACM Computing Surveys. 2014, Vol. 47, no. 2, pp. 1−71. DOI: 10.1145/2628913

8. **Sethuraman S.** Analysis of Fork-Join Systems: Network of Queues with Precedence Constraints. Boca Raton: CRC Press, 2022, 104 p. DOI: 10.1201/9781003150077

9. **Kleinrock L.** Queueing systems. NY: Wiley-Interscience, 1975, 417 p.

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Zayats Oleg I.
Заяц Олег Иванович
E-mail: zay.oleg@gmail.com

Baksheev Vitaliy E.
Бакшеев Виталий Ефимович
E-mail: yaming2110@gmail.com

Zaborovsky Vladimir S.
Заборовский Владимир Сергеевич
E-mail: vlad2tu@yandex.ru

Muliukha Vladimir A.
Мулюха Владимир Александрович
E-mail: vladimir.muliukha@spbstu.ru
ORCID: https://orcid.org/0000-0002-3583-7324

NON-INVASIVE HEART RATE MEASUREMENT SYSTEM
BASED ON VIDEO STREAM ANALYSIS

M.S. Antonenko[1] ✉ *, D.O. Budanov[1], A.Yu. Zaitceva[2]* ⓘ

[1] Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation;
[2] Institute for Analytical Instrumentation of the Russian Academy of Sciences,
St. Petersburg, Russian Federation

✉ antonenko.ms@edu.spbstu.ru

Abstract. The paper is devoted to the development and testing of a remote biomonitoring system based on the phenomenon of plethysmography. This phenomenon allows not only to measure a person's pulse rate non-invasively, but also to assess physiological state of the person. At the first stage of the system operation, it is necessary to detect regions of interest. This operation can be effectively implemented using neural networks. The task of face recognition was performed by the YOLOv7-tiny architecture, due to its speed and the ability to run on embedded systems. For the detected face, a rectangle was created, whose coordinates indicated the boundaries of the face. Next, the average brightness of the selected areas is calculated and stored in the dataset. By performing fast Fourier transform (FFT) for a given set, it is possible to obtain a signal spectrum. Using methods of digital signal processing, it is possible to filter the signal and select the part of the spectrum of interest in the region of $0.7-3$ Hz. The maximum amplitude of the harmonic will correspond to the current pulse.

СИСТЕМА НЕИНВАЗИВНОГО ИЗМЕРЕНИЯ ЧАСТОТЫ СЕРДЕЧНЫХ СОКРАЩЕНИЙ НА ОСНОВЕ АНАЛИЗА ВИДЕОПОТОКА

М.С. Антоненко[1] ✉ *, Д.О. Буданов[1], А.Ю. Зайцева[2]* 🆔

[1] Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация;

[2] Институт аналитического приборостроения РАН,
Санкт-Петербург, Российская Федерация

✉ antonenko.ms@edu.spbstu.ru

Аннотация. Статья посвящена разработке и тестированию системы дистанционного биомониторинга на основе явления плетизмографии. Данное явление позволяет не только неинвазивно измерить пульс человека, но и оценить его физиологическое состояние. На первом этапе работы системы необходимо детектировать области интереса. Данная операция может быть эффективно реализована с использованием нейронных сетей. Задачу распознавания лица выполняла архитектура YOLOv7-tiny, за счет быстродействия и возможности запуска на встраиваемых системах. Для детектированного лица создавался прямоугольник, координаты которого обозначали границы лица. Далее осуществляется вычисление средней яркости выбранных областей и сохранение в наборе данных. Выполняя Быстрое преобразование Фурье для заданного набора, можно получить спектр сигнала. Используя методы цифровой обработки сигнала можно отфильтровать сигнал и выделить интересующий нас участок спектра в районе 0.7-3 Гц. Максимальная амплитуда гармоники и будет соответствовать текущему пульсу.

Ключевые слова: дистанционный биомониторинг, телемедицина, частота сердечных сокращений, фотоплетизмография, преобразование Фурье, нейронная сеть

Introduction

Nowadays, telehealth systems capable for monitoring human physiological parameters are a promising area of research. As a rule, these systems are used for non-invasive express diagnosis of pathological conditions. Non-invasive method for measuring heart rate (HR) is based on the analysis of photoplethysmogram that is the result of recording changes that occur when small vessels are filled with blood, depending on the phase of the cardiac cycle [1]. This dependence is periodic and indicates the current pulse of a person. Moreover, it can be obtained using a conventional camera. By performing frame-by-frame image processing, it is possible to evaluate changes in skin tone and estimate current HR of a person [3]. In this case, of particular interest in the measurement are forehead and under-eye areas, where change in skin tone is most noticeable. Then, using methods of digital signal processing and computer vision, it is possible to obtain information about the current HR from the brightness curve [4].

In this paper, the algorithm for determining regions of interest (ROIs) on a person's face in a video stream is proposed. To solve this task, it is necessary to detect a face in the frame and then select the boundaries of the ROIs and outline them with geometric primitives (rectangles). At this stage, high-performance neural network models should be used. Next, the HR is calculated using classical signal analysis methods. Based on the results of comparison of existing models for face detection, a neural network
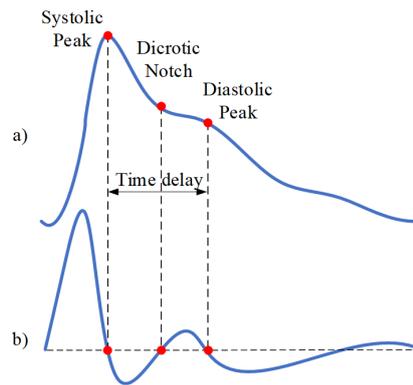
Fig. 1. Signal measurements: (a) original fingertip photoplethysmogram;
(b) first derivative wave of photoplethysmogram

with the YOLOv7-tiny architecture was chosen. The algorithm for finding ROIs on a person's face is based on coordinate regression, which is used to find the coordinates of a fixed number of points. To carry out coordinate regression it is possible to use a convolutional neural network based on the Mobile-NetV2 architecture [5].

Photoplethysmography

Photoplethysmography is the method that evaluates changes of blood volume in blood vessels each time a heart beats [1] (Fig. 1).

The arterial pulse waveform can be separated into three distinct components:

• The systolic phase, characterized by a rapid increase in pressure to a peak, followed by a rapid decline. This phase begins with the opening of the aortic valve and corresponds to the left ventricular ejection.

• The dicrotic notch, which is widely believed to represent the closure of the aortic valve.

• The diastolic phase, which represents the run-off of blood into the peripheral circulation [2].

The principle of photoplethysmography is based on determining the optical density of tissue. The ROI is illuminated from one side, after which the scattered light reflected and transmitted through the tissue area is received at the photodetector. The magnitude of its intensity is proportional to the change in blood supply to the tissue during contraction and relaxation of the heart muscle. The more blood in the vessel lead to an increase red blood cells that scatter light, the more light is reflected from them [3].

Optical methods of microcirculation analysis of biological tissues are based on the total spectral optical parameters of the medium (reflectance, scattering, absorption). For different biological environments, it depends on the functional, physiological and pathophysiological state of tissues, on their anatomy as well as on the percentage concentration of certain endogenous tissue chromophores in them, different forms of hemoglobin, connective tissue collagen, fat, water, melanin, etc. Each tissue chromophore has its own and specific spectral characteristic, which makes it possible to identify these molecular compounds by optical methods and distinguish them from other chromophores contained in the biological tissue. On this basis, methods have been developed that are now actively and quite successfully used in clinical practice. It can be single out the well-known and widespread method of pulse oximetry, which is accurate enough in assessing microcirculation and oxygen saturation. In addition, this method is fast and accessible, but has some significant drawbacks and limitations in interpreting the result [4].

The disadvantages of such methods and systems are as follows:

• The patient's movements can greatly affect the measurement result.

• Poor tissue perfusion distorts the measurement result and, as a consequence, this method depends on the pulse component.
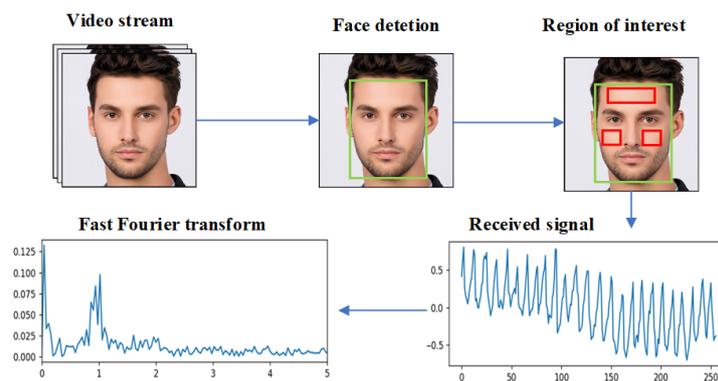
Fig. 2. Steps taken to extract the heart rate from facial video

- If there is abnormal hemoglobins in the blood, such as methemoglobin, the result may be unreliable.
- Microcirculation can be assessed only on certain skin areas: finger/nose/earlobe.
- The technique does not show the status of tissue microcirculation in the periphery, does not analyze the integral result and requires subjective interpretation of the result by medical personnel.

Changes in the parameters of backscatter radiation, in conjunction with a certain sequence of changes in the color of tissues on the face, can be recorded by wearable sensors and video camera, processed by machine learning methods to determine the presence and degree of microcirculatory pathology [5].

Heart rate determination algorithm

This section describes the basic steps taken to obtain the HR based on changes of skin tone. The entire algorithm is presented in Fig. 2.

At the first stage it is necessary to obtain an image from the camera, then detect a face and select ROIs (in this case, forehead and under-eye areas). This step is performed using the OpenCV library.

At the next stage, the average brightness of each selected area is calculated. By determining this value frame by frame for each moment of time, it is possible to obtain the dependence of changes of skin tone over time (photoplethysmogram). To obtain the numerical value of the HR, it is necessary to process the received data [7]. This can be achieved by the use of Fourier transform to convert a function from the time domain into the frequency domain. Then, the frequency with the largest amplitude in the range of 0.75−3 Hz is selected from the obtained spectrum. This value corresponds to the numerical value of the HR.

The raw heartbeat signal contains other extraneous high and low frequency components due to ambient color and motion noise induced from the data capturing environment. Therefore, to increase accuracy approximation algorithms, filters, and signal division into modes are applied [7, 8].

Using methods of NumPy statistical data processing libraries and Matplotlib data visualization libraries, the following results were obtained (Fig. 3, 4).

To process the data, first, the average value of the entire set is subtracted to eliminate the constant component of the trend. Next, appropriate filters are applied to eliminate high and low frequency components. After the Fourier transform and obtaining the spectrum, the maximum is checked against the average value to detect a static image (Fig. 5).

Results of heart rate calculation

The next step is to obtain, detect and outline a face with a primitive rectangle (bounding box). To solve this problem, a neural network for real-time object detection YOLOv7 was used. YOLOv7-tiny is the most compact and fastest model, suitable for use on devices with limited resources. Memory consumption and a recognition time were taken into account when choosing the architecture of the neural
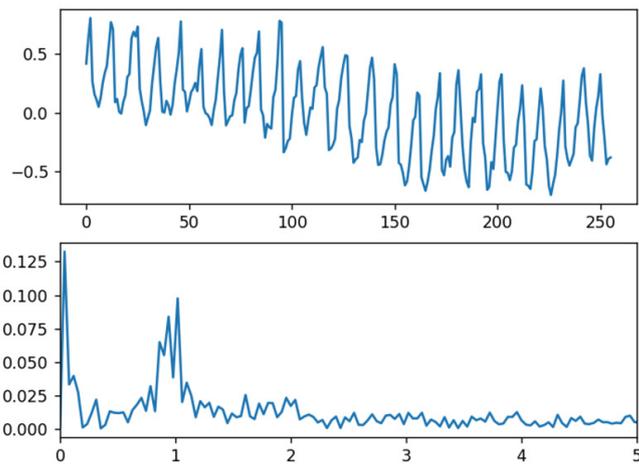
Fig. 3. Heartbeat signal in the state of calm (above) and its spectrum (below)
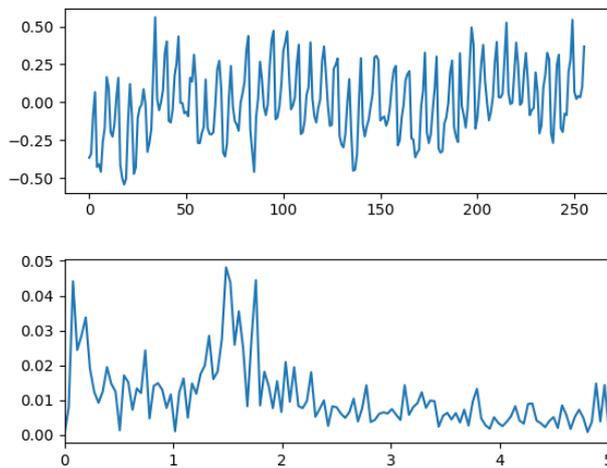
Fig. 4. Heartbeat signal (above) and its spectrum (below) under a physical activity
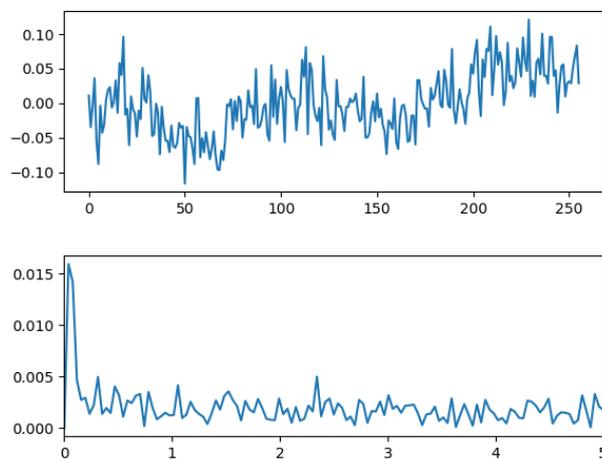
Fig. 5. Heartbeat signal of static video (above) and its spectrum (below)
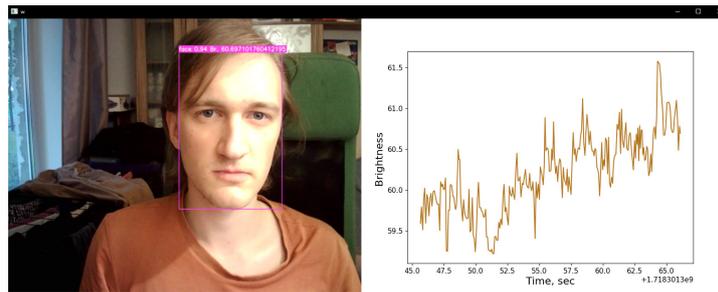
Fig. 6. Interface of an application for heart rate detection

network used in this paper. YOLOv5 and YOLOv6 offer different model sizes as well, however, on average they are more resource intensive. Developers provide a comparison of different versions of the YOLO model in [9]. In [5], for this neural network, recall, accuracy, and average accuracy were used as tracked metrics, which reached 68.7%, 62.2%, and 86.2%, respectively. All operations are performed during the single iteration unlike other detection algorithms that need to repeat the same image processing many times, such as algorithms based on region proposal networks [9]. To train the model the Wider Face dataset[1] was used, which contains 32203 images with 393703 labeled faces. Each image has the size of 640x640 pixels. During model training, the computational TPU cores were used in Google Colab with the following parameters: number of epochs − 17, batch size − 16, workers − 4.

The software part of the proposed system, containing a trained model with a HR detection algorithm and a user interface, was executed on two hardware platforms: CPU-based and GPU-based. The first platform consisted of AMD Ryzen 7 3700U CPU and 6 GB of RAM. The second platform consisted of GeForce RTX 3070 GPU with 32 GB of RAM. The GPU contained 5888 CUDA cores and 8 GB VRAM, which makes the performance of the platform several orders of magnitude better. Then a video stream from a web camera was sent to the input of the trained model and a face was detected. The resulting coordinates of bounding box were used in the HR calculation algorithm, for which the average brightness was calculated in the whole detected area. The image processing window is shown in Fig. 6. As can be seen from the left side of the figure, the camera detects a face and outlines it. The probability of detection and average brightness are shown on the top of the bounding box. The right side of the figure contains a graph of brightness (absolute value) over time (sec). This curve is polyharmonic due to the constant change in frame size and parasitic facial movements.

According to the obtained results, the influence of the camera matrix resolution prevails over the influence of the illumination level. Therefore, the influence of the illumination level will not be considered further. The frame rate of the video stream must be at least 30 fps (supported by any modern web-camera). The minimum image resolution is 640x640, since a dataset with images with this resolution was used to train the model.

Fig. 7 shows the signal spectrum before and after filtering with a high-pass filter with a cutoff frequency of 0.7 Hz. The pulse in the state of calm is 58−65 beats per minute, which corresponds to a peak around 1 Hz.

To identify the dependence of measurement accuracy from the performance of the prototype of the proposed system, measurements were carried out on two platforms: GPU (CUDA) and CPU. The results of executing the proposed application when varying the spectrum counting points are presented in Table.

Table 1 shows that calculations on the CPU have unacceptable accuracy and require more time to measure, due to the fact that video processing on the GPU occurs in real time (30 fps), while on the CPU it is 2−3 fps. As a consequence, peaks are often missed and a lot of useful information is lost. The number

---

[1] WIDER FACE: A Face Detection Benchmark, Available: http://shuoyang1213.me/WIDERFACE/ (Accessed 13.09.2024)
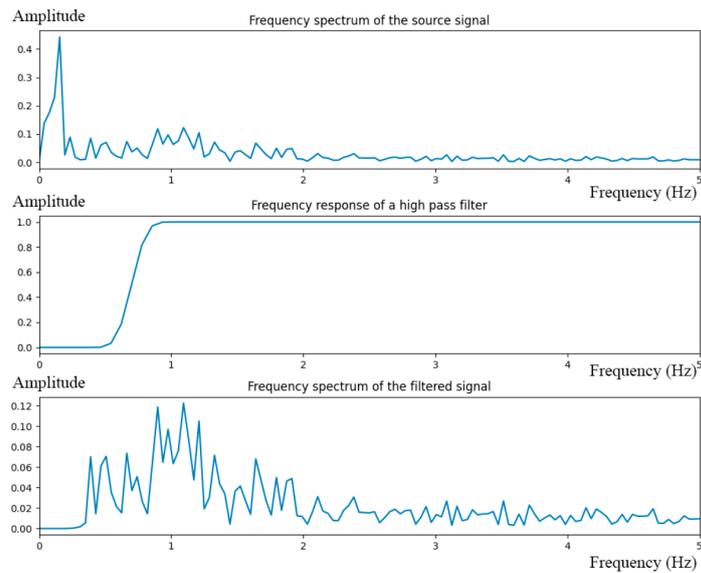
Fig. 7. Signal spectrum

of points for fast Fourier transform does not greatly affect the accuracy of measurements. However, the more points used, the more reliable results are, as there is less influence of unnecessary accidental facial movements. At the same time, the speed and complexity of calculations increase proportionally. From this point of view, it is optimal to use 128 points. For embedded systems, it is possible to use the NVIDIA Jetson platform. This single board computer has a small size and low power consumption. In addition, it contains GPU with high-performance CUDA cores suitable for machine learning tasks.

Table

Comparison of actual and received values from the platform

№	Obtained	Real	Number of points	Computing platform	Error
1	60.7	69	256	GPU	8.4
2	62	74	256	GPU	12
3	59.4	69	256	GPU	9.4
4	54.3	70	128	GPU	15.7
5	56	66	128	GPU	10
6	66	69	128	GPU	3
7	39	65	256	CPU	26
8	51	64	256	CPU	13
9	33	68	128	CPU	35
10	42	65	128	CPU	23

Comparison with other implementations

The article [7] uses a similar method for measuring HR based on changes in skin tone, as well as micromovements of the face. To remove/reduce the extraneous frequency components and trends from the signal it was decomposed using Hodrick-Prescott filter. The accuracy of the proposed method was compared with state of the art color and the motion-based methods of [11−12]. The overall error rates are less than 10% for HR estimation by counting the number of peaks for both motion and color signals.

In the description of how the application "webcam-pulse-detector" works[2], it says, that data was collected by "measuring average optical intensity in the forehead location, in the subimage's green channel alone (a better color mixing ratio may exist, but the blue channel tends to be very noisy)". Measurement accuracy is not given, however, with a good lighting and minimal noise due to motion, a stable heartbeat should be isolated in about 15 sec. In this paper, it takes 16 seconds to measure heartbeat rate using 128 points.

Several methods are considered in [10] to achieve an error of 5 beats per minute.

An illumination rectification method by using two points on the skin of the face in the same frame to extract the green spectrum of each point was proposed in [15]. After that, independent component analysis was applied to extract the photoplethysmography signal from the two green spectrum signals. Therefore, treating the effect of illumination variance as a blind source separation problem. ROI selections criteria using facial landmarks fitting was proposed by the authors of this article as well.

Conclusion

Remote biomonitoring plays an important role in modern telehealth systems. It allows to measure non-invasively the basic parameters of a person's physiological state without taking samples in a few seconds using only a camera and specialized software. In this work, a system for measuring HR based on analysis of video stream was proposed. The results of the experiment showed that for the correct and affordable work of the system it is necessary to use a GPU in a hardware part of the system. Therefore, currently proposed system has low accuracy since the entire area of the rectangle that contains face is taken into account, however this expands the measurement capabilities. In the future work, to increase accuracy, a neural network model to detect ROIs on the human face will be developed and applied.

REFERENCES

1. **Lazim M.R.M.L.M., Aminuddin A., Chellappan K., Ugusman A., Hamid A.A., Ahmad W.A.N.W., Mohamad M.S.F.** Is heart rate a confounding factor for photoplethysmography markers? A systematic review. International Journal of Environmental Research and Public Health, 2020, vol. 17, no. 7, article no. 2591. DOI: 10.3390/ijerph17072591

2. **Yartsev A.** Normal arterial line waveforms. 2015. Available: https://derangedphysiology.com/main/cicm-primary-exam/required-reading/cardiovascular-system/Chapter%20760/normal-arterial-line-waveforms (Accessed 10.05.2024)

3. **Guryleva A., Fomina D., Volkov M., Danilycheva I., Serdoteckova S.** Digital image processing for the diagnosis of the cold inducible urticaria by photoplethysmographybased methods. 2023 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF), 2023, pp. 234–245. DOI: 10.1109/WECONF57201.2023.10147961

4. **Danishevskiy N.S., Budanov D.O., Zaytseva A.Y.** Facial Landmark Detection Approaches for Remote Medical Applications. IEEE International Conference on Control in Technical Systems, 2023, pp. 177–180. DOI: 10.1109/CTS59431.2023.10288892

5. **Danishevskiy N.S., Budanov D.O., Zaytseva A.Yu.** Development of an intelligent system for remote biomonitoring of heart rate. Technical Physics Letters, 2023, vol. 49, no. 12, pp. 23–25. DOI: 10.61011/TPL.2023.12.57575.204A

6. **Huang S.-H.** Accurate pulse transit time acquisition method for multi-point photoplethysmography. International Conference on Consumer Electronics – Taiwan (ICCE–Taiwan), 2023, pp. 551–552. DOI: 10.1109/ICCE-Taiwan58799.2023.10226655

7. **Haque M.A., Nasrollahi K., Moeslund T.B.** Estimation of heartbeat peak locations and heartbeat rate from facial video. Image Analysis. SCIA 2017. Lecture Notes in Computer Science, 2017, vol. 10270, pp. 269–281. DOI: 10.1007/978-3-319-59129-2_23

---

[2] Webcam-pulse-detector, Available: https://github.com/thearn/webcam-pulse-detector (Accessed 13.09.2024)

8. **Suriani N.S., Jumain N.A., Ali A.A., Mohd N.H.** Facial video based heart rate estimation for physical exercise. 2021 IEEE Symposium on Industrial Electronics & Applications (ISIEA). 2021, pp. 1−5. DOI: 10.1109/ISIEA51897.2021.9509986

9. **Wang C.-Y., Bochkovskiy A., Liao H.-Y.M.** YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, pp. 7464−7475. DOI: 10.48550/arXiv.2207.02696

10. **Hassan M.A., Malik A.S., Fofi D., Saad N., Karasfi B., Ali Y.S., Meriaudeau F.** Heart rate estimation using facial video: A review. Biomedical Signal Processing and Control, 2017, vol. 38, pp. 346−360. DOI: 10.1016/j.bspc.2017.07.004

11. **Poh M.-Z., McDuff D.J., Picard R.W.** Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. Optics Express, 2010, vol. 18, no. 10, pp. 10762−10774. DOI: 10.1364/OE.18.010762

12. **Kwon S., Kim H., Park K.S.** Validation of heart rate extraction using video imaging on a built-in camera system of a smartphone. 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2012, pp. 2174−2177. DOI: 10.1109/EMBC.2012.6346392

13. **Balakrishnan G., Durand F., Guttag J.** Detecting Pulse from Head Motions in Video. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 3430−3437. DOI: 10.1109/CVPR.2013.440

14. **Poh M.-Z., McDuff D.J., Picard R.W.** Advancements in Noncontact, Multiparameter Physiological Measurements Using a Webcam. IEEE Transactions on Biomedical Engineering, 2011, vol. 58, no. 1, pp. 7−11. DOI: 10.1109/TBME.2010.2086456

15. **Lam A., Kuno Y.** Robust heart rate measurement from video using select random patches. Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 3640−3648.

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Antonenko Maksim S.
Антоненко Максим Сергеевич
E-mail: price.lt@yandex.ru

Budanov Dmitry O.
Буданов Дмитрий Олегович
E-mail: budanov_do@spbstu.ru

Zaitceva Anna Yu.
Зайцева Анна Юрьевна
E-mail: anna@da-24.ru
ORCID: https://orcid.org/0000-0003-0334-2770

DEVELOPMENT OF THE SYSTEM
OF AUTOMATIC GENERATION OF DATABASE MODEL
ON THE BASIS OF THE TASK TEXT IN NATURAL LANGUAGE

I.A. Lapin ✉ , *O.Yu. Sabinin*

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

✉ lapin_ia@spbstu.ru

Abstract. This paper describes an approach to the implementation of a system that would allow automatic database model generation from a natural language description given by the user. Different machine learning technique, such as transformer, named entity recognition and relation extraction are considered and applied. The implementation of the neural network model uses the capabilities of the spaCy framework to organize a generic pipeline for training. Off-the-shelf implementations of some individual components from spaCy are also used, while the rest are custom. Moreover, we describe the process of gathering and preparing raw data for training a neural network model, and generating a proper corpus from them. For this purpose, a specialized annotating tool, Doccano, is used, which satisfies all requirements and is freely available. Finally, the paper presents the model parameters used in training and the performance metrics obtained. We've been able to achieve great results for the named entity recognition component, while the performance metrics of the relation extraction component can still be improved. The paper concludes with possible directions for further work on the implementation of the described system, including the relation extraction component improvements and new features implementation.

РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЧЕСКОЙ ГЕНЕРАЦИИ МОДЕЛИ БАЗЫ ДАННЫХ НА ОСНОВЕ ТЕКСТА ЗАДАНИЯ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

И.А. Лапин ✉ , *О.Ю. Сабинин*

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

✉ lapin_ia@spbstu.ru

Аннотация. В данной статье описывается подход к реализации системы, которая позволила бы автоматически составлять модель базы данных по приведенному пользователем описанию на естественном языке. Рассматриваются и применяются различные методы машинного обучения, такие как трансформер, распознавание именованных сущностей и извлечение отношений. При реализации нейросетевой модели применяются возможности фреймворка spaCy для организации общего пайплайна для обучения. Также используются готовые реализации некоторых отдельных компонентов из spaCy, в то время как остальные являются пользовательскими. Кроме того, в статье описывается процесс сбора исходных данных для обучения нейросетевой модели, а также формирование из них надлежащего корпуса. Для этих целей используется специализированный инструмент для аннотирования — Doccano, который удовлетворяет всем функциональным требованиям, а также находится в свободном доступе. Наконец, в статье приводятся используемые при обучении параметры модели и полученные метрики производительности. В результате проведенного исследования авторам удалось достигнуть высоких показателей для компонента named entity recognition, в то время как показатели производительности для компонента relation extraction можно еще улучшить. В конце статьи приводятся возможные направления дальнейшей работы над реализацией описанной системы.

Ключевые слова: обработка естественного языка, распознавание именованных сущностей, извлечение отношений, анализ текста, классификация, реляционные базы данных, построение моделей

Для цитирования: Lapin I.A., Sabinin O.Yu. Development of the system of automatic generation of database model on the basis of the task text in natural language // Computing, Telecommunications and Control. 2024. Т. 17, № 3. С. 93—102. DOI: 10.18721/JCSTCS.17309

Introduction

In today's world, it is becoming somewhat of a *mauvais* to talk about the use of various information systems in a certain area — so global has become the digitalization of all areas of human activity. It is hard to imagine that today somewhere such systems are not used. This means that if information needs to be collected, processed and used, it also needs to be stored. Hence, there is no diminishing need for data storage tools and specialists, and approaches to this process are becoming more and more complex. Thus, in the field of databases, because it is through these tools that information storage is provided, specialists are increasingly in demand, with the growing number and complexity of information systems being developed, as well as with the need to support and expand existing ones.

It is human nature to look for simpler ways of solving the tasks we are facing, especially if these tasks become routine and take away time that should be spent on solving more complex requiring an individual approach. Therefore, we are trying to teach artificial intelligence (AI) to solve such routine tasks for humans, thereby freeing up resources for other tasks. Among other things, this is aimed at helping

people who do not have the necessary specialized knowledge in a certain field to get the opportunity to use various tools, at least their basic functionality.

This article will discuss the process of developing such a solution based on AI, which would allow one to create an initial representation of a relational database model based on the text description of the task of building a database in natural language. Such a tool should allow specialists to save time when implementing the developed database architecture, or when planning, being able to visualize different variants of possible architecture. In addition, such a solution will help to solve simple database development tasks for students, startup teams that are unwilling or unable to hire a specialist, as well as people who do not have professional knowledge in the database domain. The description of preliminary research as well as an assessment of the possibility of creating such a system, is given in [1].

Description of the chosen approach

When developing a software implementation of the system of automatic database model generation based on natural language text, we faced several major challenges:

1) to find logical entities in the text that represent the tables of the future model, as well as their attributes;

2) to relate the attributes to the logical entities, to which they refer;

3) to determine data types for the attributes;

4) to determine constraints for the tables (mainly foreign keys).

This article will discuss the solution to the first two challenges. Thus, at this stage, it is necessary to develop a software solution that would allow to input the text describing the modeling task in natural language and at the output get a certain set of logical entities with their attributes found in the text.

We decided to use the Python library spaCy[1] as a basis for building a software implementation, as this library offers a wide set of ready-made machine learning components for working with natural language text processing tasks, available for different languages, including Russian. In addition, this library offers a unified ecosystem of proprietary components, which can also include new, manually created components, which ultimately provide a unified pipeline for training and using the final model.

SpaCy uses a deep learning approach consisting of four main stages: embedding, encoding, attending, and prediction [2]. That is, first, tokenization takes place, where each representation is given a unique identifier and a table of word representations is formed, the size of which is determined by the size of the corpus dictionary. As a result, a set of vectors containing different tokens is formed.

The second stage involves the formation of a matrix containing context vectors for each token. This uses a combination of the Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) models instead of the usual recurrent neural network (RNN) implementation, since the hidden layer in RNN is constantly being rewritten and thus accumulation of information is problematic, while LSTM and GRU ensure that the results of the previous pass are saved for use in the next one. Finally, the resulting token vectors are composed of the vectors formed during the forward and backward passes.

The third stage uses the attention-mechanism to generate the final vectors. The last step, however, depends on the implementation of the particular component.

It is also worth noting that spaCy allows us to create joint models using a shared context for components, such as a vector store. We can use a transformer, or tok2vec component in the pipeline to form token vectors, and then use its results in each subsequent component in the pipeline without having to compute them again, as well as ensuring the integrity and equivalence of the store for all components. In addition, spaCy allows any transformer implementations, such as loading and inserting models with Huggingface into the pipeline.

When dealing with the first challenge — to find logical entities in the text that represent the tables of the future model, as well as their attributes — an approach to natural text processing using machine

---

[1] SpaCy, Industrial-Strength Natural Language Processing, Available: https://spacy.io (Accessed: 20.03.2024)
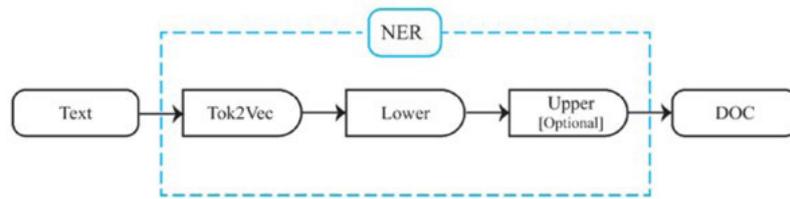
Fig. 1. NER model architecture [2]

learning technique was required, which could help to find specific designated classes of entities in the text depending on the context and the words themselves. Such a technique is called Named Entity Recognition (NER)[2], and has been used and improved for quite a long time. This technique allows solving tasks of determining named entities in text. Such tasks involve identifying certain entities that fall into different groups (classes) in unstructured text. NER technique is often used to locate people, organizations, geographic locations, etc. in text. Our task also fits well under this category of tasks, since we are solving a similar problem of locate entities belonging to a specific group in unstructured text. Thus, using NER component allows us to solve the problem of searching in text for logical entities and attributes.

To implement NER component, spaCy uses a transition-based system based on the fragmentation model proposed by Lample [3]. This is an approach based on computing different state transitions for prediction. The structure of NER component is shown in [2] (Fig. 1). The tok2vec component performs the entire process of translating tokens into vector representation. The "lower" component then creates special vectors for each property by token-property pairs, resulting in some general representation of the current state for each token. The "upper" component then uses the feed-forward network to predict weights based on the state representations.

To solve the second challenge we need to find a way to determine the relations in the text between specific attributes and the logical entities, to which they refer. The very problem of finding dependencies and relations between words in a text is not a new one and has already received different solutions many times. While earlier it was mainly pattern-matching, nowadays various machine learning technique are used, the most common and applicable of which are dependency parsing [4] and relation extraction[3].

However, pattern-matching is still often used to solve various special problems, such as one described in [5], but in a more advanced form of rule-based analysis. Nevertheless, this technique cannot be used in our case, since the text of the task description can be arbitrary, therefore, it is impossible to identify specific rules that would be used to determine the relations.

The dependency parsing technique allows us to determine hierarchical dependencies between words, mostly within a single sentence, by constructing a directed graph with nodes (words) and edges (links). Despite the examples of existence of applications of this technique in tasks related to the search for relations between NER entities [6] , dependency parsing does not allow us to obtain consistently desired result in our case, since it does not guarantee the construction of the same dependency graphs for different wording and word orders. In addition, the task of forming a dataset for training the dependency parsing component turns out to be quite extensive and redundant in this case.

Speaking of the relation extraction, it appears to be much more suitable, since it allows to search for relations between labeled entities, i.e., between entities found by NER or marked up manually in advance. In general, the algorithm of relation extraction component is shown in Fig. 2, and it can be described as follows[4]:

---

[2] What Is Named Entity Recognition? | IBM, Available: https://www.ibm.com/topics/named-entity-recognition (Accessed: 03.09.2024)

[3] Relationship Extraction | NLP-progress, Available: https://nlpprogress.com/english/relationship_extraction.html (Accessed: 20.03.2024)

[4] SPACY v3: Custom trainable relation extraction component, Available: https://youtu.be/8HL-Ap5_Axo?si=qRJOR-DbxhnZpcxj (Accessed: 13.03.2024)
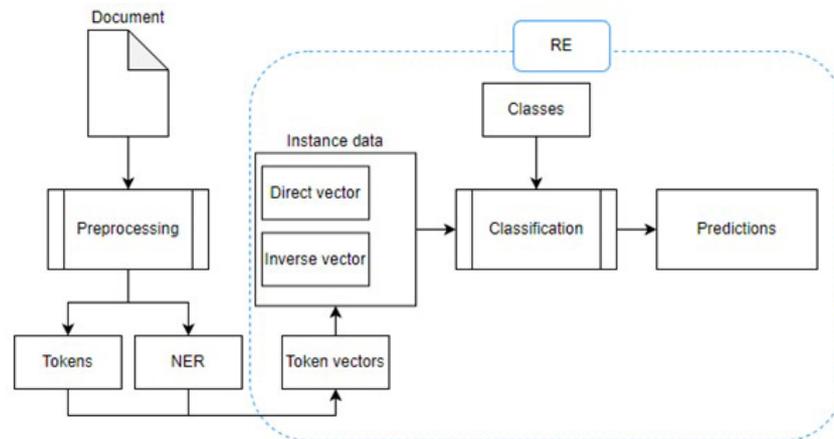
Fig. 2. Generalized RE component scheme
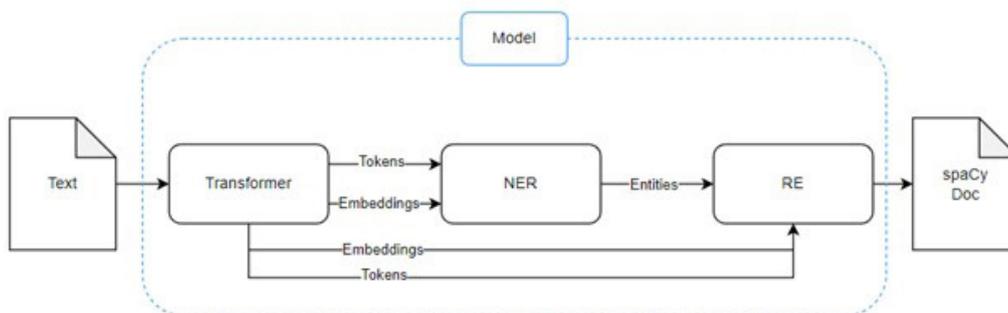
Fig. 3. Overall model scheme

1) Vector representations are analyzed and new vectors for relationships are constructed, including entity vectors and contextual information. In doing so, vectors will be created for all pairs of relations. This factor can be limited by setting the maximum window size as one of the hyperparameters. It is also worth noting that vectors will be created for both "direct" and "reverse" pairs.

2) The resulting vectors are assigned an assessment of belonging to a particular class (relation label). Any suitable classifier can be used, provided that the classification is performed with rejection, otherwise the relations will be assigned to any entities included in the window. Based on the obtained weights and taking into account the rejection threshold, the predicted labels are determined.

Since spaCy does not have a native implementation of the relation extraction component, we decided to use the standard generalized implementation provided by the spaCy developers[5]. It is worth noting that a big advantage of spaCy is the ability to easily add custom components to the standard pipeline due to the flexibility of their interfaces, as well as the capabilities of the thinc library[6], used for simplified implementation of AI components.

We decided to use one of the standard spaCy transformers, the multi-language Bert model, as a common tok2vec component for our pipeline to improve the quality of vector representations and fully use

[5] Custom spaCy Relation Extraction Component, Available: https://github.com/explosion/projects/tree/v3/tutorials/rel_component (Accessed: 20.03.2024)

[6] Thinc – A refreshing functional take on deep learning, compatible with your favorite libraries, Available: https://thinc.ai (Accessed: 20.03.2024)

```
Задача: Создать базу данных для курьерской службы, содержащую информацию о
заказах, курьерах и доставке. База данных должна содержать следующие таблицы:
    1. Таблица "Заказы" с полями: ID заказа, Дата заказа, Адрес отправителя,
Адрес получателя, Описание заказа.
    2. Таблица "Курьеры" с полями: ID курьера, ФИО курьера, Телефон курьера.
    3. Таблица "Доставка" с полями: ID доставки, ID заказа, ID курьера, Дата
доставки, Статус доставки.
```

Fig. 4. Easy level generated text

the attention-mechanism. NER and the relation extraction components rely on a shared store of vector representations obtained using the transformer. Fig. 3 shows the scheme of the resulting model.

Data mining and training set formation

To prepare a dataset for model training, we needed to collect a certain number of database modeling task texts. In the process of studying open access datasets on Kaggle and Huggingface, it became clear that, with a high degree of probability, the required dataset in ready or partially ready form does not exist, at least in the public domain. Therefore, a search on the Internet resulted in several open resources with the required tasks texts in Russian[7] [7−8].

Unfortunately, the amount of data found was insufficient to form the minimum required dataset, since, according to the spaCy developers' recommendation, the dataset for training should contain at least a couple hundred records[8]. However, using the found sources, we managed to form a set of only 58 tasks. We decided that the size of the training set for the trial implementation could be about a hundred records, and yet, it was necessary to find the missing records somewhere.

Therefore, we decided to use Large Language Models (LLMs), since at this level of development of AI technologies we can use generative AI models to generate new task texts. We decided to use domestic solutions: Sber GigaChat[9] and YandexGPT[10]. Web user interfaces in the public domain were used to work with the models. Using various queries to the models, 30 new tasks of varying levels of text complexity were generated (according to the empirical evaluation of the tasks already in the set), ranging from simple (Fig. 4) to medium (Fig. 5).

This resulted in a total dataset size of 88 records. This amount was still less than the recommended number, but was much closer to the minimum recommended one hundred records. There was also the possibility that generating more records could cause the model to "learn" the pattern of text construction offered by the generative AI and overtrained as a result. In addition, in general, the use of a large amount of surrogate data during training is considered to be an acceptable but undesirable practice, since the artificially generated data can often be very different from the real data that the model will encounter during its work. Thus, we decided to settle on a compromise − the number of records in the final dataset of about a hundred records and the ratio of generated texts to "natural" texts from open sources and their variability should allow us to obtain adequate results for real-life application.

After the initial dataset was formed, we wanted to improve the quality of the data collected. The data was examined for anomalies, problems with text formatting and overall low-quality texts that could lead to undesired interference with training were flagged.

---

[7] Tekhnologii baz dannykh i znaniy – Individualnyye zadaniya dlya samostoyatelnoy raboty [Database and knowledge technologies – Individual tasks for self-study], Available: http://bseu.by/it/tohod/indv_zadaniya.htm (Accessed: 20.03.2024) (in Russ.); FKN+ANTITOTAL – Varianty zadach – proyektirovaniye baz dannykh [Task options – Database engineering], Available: https://fkn.ktu10.com/?q=node/72 (Accessed: 20.03.2024) (in Russ.); Zadacha i teoriya po SQL, MySQL, PostgreSQL i bazam dannykh voobshche [Tasks and theory for SQL, MySQL, PostgreSQL and databases in general], Available: https://gist.github.com/codedokode/10539213 (Accessed: 20.03.2024) (in Russ.)

[8] Training Pipelines & Models spaCy Usage Documentation, Available: https://spacy.io/usage/training (Accessed: 20.03.2024)

[9] Sber GigaChat, Available: https://developers.sber.ru/gigachat (Accessed: 20.03.2024) (in Russ.)

[10] YandexGPT, Available: https://ya.ru/ai/gpt-2 (Accessed: 20.03.2024) (in Russ.)

Задание: Создать логическую модель базы данных для системы управления задачами.

Система управления задачами предназначена для организации и контроля выполнения задач в команде. Она позволяет создавать задачи, назначать ответственных, устанавливать сроки выполнения, отслеживать прогресс и оставлять комментарии.

Логическая модель базы данных должна включать следующие элементы:

1. Таблица "Задачи" с полями:
   - Идентификатор задачи (ID)
   - Название задачи
   - Описание задачи
   - Ответственный за задачу
   - Дата создания задачи
   - Дата завершения задачи
   - Статус задачи (в процессе, выполнена, отменена)
2. Таблица "Комментарии" с полями:
   - Идентификатор комментария (ID)
   - Идентификатор задачи (ID)
   - Автор комментария
   - Текст комментария
   - Дата создания комментария
3. Таблица "Статусы" с полями:
   - Идентификатор статуса (ID)
   - Название статуса
4. Таблица "Ответственные" с полями:
   - Идентификатор ответственного (ID)
   - Имя ответственного
   - Фамилия ответственного
5. Таблица "Связи" с полями:
   - Идентификатор задачи (ID)
   - Идентификатор ответственного (ID)

При создании логической модели базы данных необходимо учесть следующие требования:

1. Каждая задача должна иметь уникальное название и описание.
2. Каждая задача должна быть связана с ответственным за ее выполнение.
3. Каждая задача должна иметь дату создания и дату завершения.
4. Каждая задача может иметь несколько комментариев.
5. Каждая задача может иметь несколько ответственных.
6. Каждая задача может иметь несколько статусов.

Fig. 5. Average level generated text

Formatting and punctuation problems were corrected first:

1) Unnecessary line breaks were eliminated.

2) Insignificant task headings without punctuation and with unnecessary hyphenation were eliminated.

3) Missing punctuation marks were added.

Then the task texts were analyzed for anomalies – errors, incorrect wording, etc. As a result, most of the anomalies were corrected; if too significant changes had to be made to correct anomalies, such texts were discarded, due to the excessive labor-intensive nature of such corrections and were not included in the updated dataset.

Finally, texts that differed too dramatically from others, while providing either insufficient or poor quality information about the database model, even from a human perception point of view, were also excluded from the final set. We decided to consider such tasks as inherently incorrect and not to include them in the training set. As a result of all the improvements made, the final dataset size was reduced to 80 records.

Data markup and corpus preparation

To use the collected data set as a corpus for training the neural network model, it was necessary to mark up the data, and to convert the obtained corpus to such a format that would be easily read by spaCy when generating binary files with a special extension, which would then be used to train the model.

Data markup, or annotation, involves selecting all necessary entities and relations in texts – in our case, these are logical entities and their attributes, as well as relations of attribute dependencies on specific entities. This required finding a suitable tool that would have sufficient functionality to perform the required actions, namely to annotate entities and relations between them throughout the text.

An ideal choice for such a tool would be Explosion Prodigy[11], an advanced text annotation software with extensive functionality that allows to quickly and conveniently solve a wide range of data annotation tasks, which, moreover, is developed by the spaCy developers. However, since this tool is a paid tool and this study was conducted without additional funding, we decided to consider free analogs.

Thus, it was necessary to find an alternative tool in the public domain that offered the required functionality. As a result of studying the market of solutions offered in this area, a free open source software focused on annotating text data, Doccano[12], was selected. This tool turned out to be the only one that fully met all the requirements and offered a convenient user interface to work with.

While annotating and re-evaluating records from the dataset, we decided to extend the originally planned set of labels for entities and relations. As a result, the following set of labels was used for annotation:

1) For entities: LENTITY (logical entity), LATTRIBUTE (attribute of a logical entity), DESC (utility description for any other entities, primarily for attributes).

2) For relations: ATTIBUTE_OF (is an attribute), DESCRIBED_BY (is described).

The LENTITY and LATTRIBUTE labels were used to label logical entities and their attributes, while the DESC label was added to provide greater precision when labeling attributes in complex cases. This applies to cases when two or more attributes had the same description in the text, such as "date and time of order". In this case, we defined LATTRIBUTE labels for "date" and "time", and DESC label for "order" to subsequently form two attributes "order date" and "order time". To make such a solution work, DESCRIBED_BY relation label was introduced to show which entity with DESC label describes an entity with LATTRIBUTE label. Finally, ATTRIBUTE_OF label was used to define the relation between the attribute and the logical entity to which it refers.

Thus, the original dataset was annotated using Doccano with labels described in the previous paragraph. The result was a corpus written in JSONL file format with the following structure:

1) The text field is the task text itself in its original form.

2) The entities field is a set of JSON objects with information on annotated entities in the text (id, start of entity, end of entity, label).

3) The relations field is a set of JSON objects with information on annotated relations in the text (id, id of parent entity (from), id of child entity (to), label).

Afterwards it was necessary to edit the obtained corpus so that it could be easily converted to the internal spaCy format. It is worth noting that in the implementation of the custom relation extraction component, spaCy developers also provide a script for parsing the annotated corpus obtained using their Prodigy tool to convert into the internal binary format. Since our corpus still lacked some of the information needed, according to spaCy internal notation, we decided to write a script to supply the corpus with missing details and convert it into the required format.

Experiments

The model described in this paper was trained using the generated corpus. The following hyperparameter values were used for training, as shown in Table 1. The settings for the transformer and NER component were left at their default values, while the relation extraction component window size was set to 1000 to ensure that it could cover large intervals between potentially related entities within all or most of the text. The rejection threshold was set to 0.3, as experiments with the current version of the model have shown this threshold to be better than the more obvious threshold of 0.5. Table 2 shows the performance metrics of the trained model.

---

[11] Prodigy – Radically efficient machine teaching. An annotation tool powered by active learning, Available: https://prodi.gy (Accessed: 20.03.2024)

[12] Doccano – open-source data labeling tool for machine learning practitioners, Available: https://doccano.github.io/doccano (Accessed: 20.03.2024)

Table 1

Hyperparameter values for training

Hyperparameter	Value
Overall batch size	1000
NER hidden layer width	64
Transformer maximum batch items	4096
Transformer window	128
Transformer stride	96
RE window	1000
RE threshold	0.3

It is worth noting the excellent performance result of NER component. Despite the relatively high total error, the indicators of accuracy, recall, and F-measure are very good, showing that the model correctly performs 98% of predictions.

Table 2

Performance metrics

		Metric			
		Loss	Precision	Recall	F-score
Component	NER	4872	0.96	1.0	0.98
Value	RE	8	0.31	0.44	0.36

At the same time, the indicators for the relation extraction component are not as good, with only 36% of the conditional accuracy of predictions. It can be assumed that the problem is largely due to the fact that we consider not individual sentences, but entire texts, including those that determine the presence of relations between entities in different sentences. After all, the originally used implementation of this component was supposed to work more with individual sentences.

Conclusion

This article outlined the details of a partial implementation of a system for automatic generation of database models based on a task text in natural language. As a result, excellent results were obtained on the trained model for searching logical entities and their attributes in texts, while the component for searching relations between attributes and logical entities needs to be improved.

It is worth noting that the system is still a work in progress, since in addition to the refinement of the existing components, more extensive functionality such as defining attribute data types and searching for restrictions should be implemented. In addition, it will be necessary to post-process the results provided by the model after processing the source texts, since it is necessary to form a visual representation of the database model described in the task for the user.

REFERENCES

1. **Lapin I.A., Sabinin O.Yu.** Research and planning the development of an automated system for building relational database models based on provided task text in natural language. Theoretical & Applied Science, 2023, Vol. 11, No. 127, Pp. 311−320. DOI: 10.15863/TAS.2023.11.127.39

2. **Singh N., Kumar M., Singh B. et al.** DeepSpacy-NER: an efficient deep learning model for named entity recognition for Punjabi language. Evolving Systems, 2023, Vol. 14, Pp. 673–683. DOI: 10.1007/s12530-022-09453-1

3. **Lample G., Ballesteros M., Subramanian S., Kawakami K., Dyer C.** Neural architectures for named entity recognition. Proceedings of NAACL, 2016. DOI: 10.48550/arXiv.1603.01360

4. **Jaiswal S.** Natural Language Processing – Dependency Parsing. Medium, 2021, Available: https://towardsdatascience.com/natural-language-processing-dependency-parsing-cf094bbbe3f7 (Accessed: 20.03.2024)

5. **Kozhevnikov V.A., Sabinin O.Yu.** System of automatic verification of answers to open questions in Russian. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 3, Pp. 57–72. DOI: 10.18721/JCSTCS.11306

6. Dy Dx. Utilizing dependency trees and NER models for relation extraction task. Medium, 2021, Available: https://medium.com/@dxdy/utilizing-dependency-trees-and-ner-models-for-relation-extraction-task-effa5463cb8 (Accessed: 20.03.2024)

7. **Sidorova N.P.** Bazy dannyh: praktikum po proektirovaniyu relyacionnyh baz dannyh [Databases: A Hands-On Guide to Designing Relational Databases]. Moscow: Direkt-Media, 2020. 92 p.

8. **Potapov A.S., Astahova I.F., CHulyukov V.A., Starikov V.N.** Praktikum po informacionnym sistemam. Oracle [Oracle Information Systems Workshop]. Kiev: YUnior, 2004. 177 p.

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Lapin Igor A.
Лапин Игорь Александрович
E-mail: lapin_ia@spbstu.ru

Sabinin Oleg Yu.
Сабинин Олег Юрьевич
E-mail: sabinin_oyu@spbstu.ru

DEVELOPMENT OF AN OCT DATA
CLASSIFICATION MODEL FOR DETERMINING
THE PRESENCE AND TYPE OF OPHTHALMIC DISEASES

L.E. Aksenova[1] ✉ ⓘ , *K.D. Aksenov[1]* ⓘ ,
A.V. Prysyazhnyuk[1], V.V. Myasnikova[2,1], A.V. Krasov[3]

[1] LLC "Predict space", Novorossiysk, Russian Federation;
[2] Maykop State Technological University, Maykop, Russian Federation;
[3] Bonch-Bruevich St. Petersburg State University of Telecommunications,
St. Petersburg, Russian Federation

✉ axenovalubov@gmail.com

Abstract. Optical Coherence Tomography (OCT) is an important tool in the diagnosis of common ophthalmological diseases, such as age-related macular degeneration and diabetic retinopathy. However, the processes of analyzing and interpreting OCT data are highly complex due to the need to process a large amount of data and the time spent on research, as well as the ophthalmologist's failure to recognize minor or early signs of the disease or rare pathologies. This paper proposes a comprehensive approach to the development of an OCT image analysis system based on deep neural networks. In particular, the performance of models based on four neural network architectures − ResNet50, VGG16, InceptionV4, and ResNet101 − was evaluated. The results show that the model based on the ResNet50 architecture achieves the highest proportion of correctly classified images. Furthermore, the integration of the developed model into a chatbot significantly reduces the time needed to interpret OCT images, which can contribute to increased availability of preliminary diagnostics and improved quality of medical services.

РАЗРАБОТКА МОДЕЛИ КЛАССИФИКАЦИИ ДАННЫХ ОКТ ДЛЯ ОПРЕДЕЛЕНИЯ НАЛИЧИЯ И ТИПА ОФТАЛЬМОЛОГИЧЕСКИХ ЗАБОЛЕВАНИЙ

Л.Е. Аксенова[1] ✉ 🆔 , *К.Д. Аксенов[1]* 🆔 ,
А.В. Присяжнюк[1], В.В. Мясникова[2,1], А.В. Красов[3]

[1] ООО «Пространство интеллектуальных решений»,
Новороссийск, Российская Федерация;

[2] Майкопский государственный технологический университет,
Майкоп, Российская Федерация;

[3] Санкт-Петербургский государственный университет телекоммуникаций
им. проф. М.А. Бонч-Бруевича, Санкт-Петербург, Российская Федерация

✉ axenovalubov@gmail.com

Аннотация. Оптическая когерентная томография (ОКТ) является важным инструментом в диагностике распространенных офтальмологических заболеваний, таких как возрастная макулярная дегенерация и диабетическая ретинопатия. Тем не менее, процессы анализа и интерпретации данных ОКТ представляют высокую сложность как в виду необходимости анализа большого количества данных и затраченного на исследования времени, так и пропуска незначительных и ранних признаков заболевания или редких патологий врачом офтальмологом. В настоящей работе предложен комплексный подход к разработке системы анализа изображений ОКТ на основе глубоких нейронных сетей. В частности, была проведена оценка производительности моделей на основе четырех архитектур нейронных сетей — ResNet50, VGG16, InceptionV4 и ResNet101. Результаты показывают, что модель на основе архитектуры ResNet50 позволяет достичь наибольшей доли правильно классифицированных изображений. Кроме того, внедрение разработанной модели в чат-бот позволяет существенно сократить время интерпретации ОКТ изображений, что может способствовать увеличению доступности предварительной диагностики и улучшению качества оказания медицинских услуг.

Ключевые слова: искусственный интеллект, офтальмология, модели машинного обучения, архитектуры нейронных сетей, сверточные нейронные сети, оптическая когерентная томография, чат-бот

Introduction

According to the World Health Organization (WHO), some of the most common ophthalmic diseases are age-related macular degeneration (AMD) and diabetic retinopathy (DR). At the same time, vision-related diseases often lack obvious symptoms in the early stages and are easily overlooked by patients, leading to irreversible vision impairment by the time they visit the clinic [1]. Optical Coherence Tomography (OCT) is widely used in ophthalmology and is considered the gold standard for early diagnosis of many diseases, identification of prognostic biomarkers, monitoring disease progression, and evaluating patient

response to treatment [2]. OCT is a non-invasive imaging method that allows for high-resolution imaging of eye structures, with a resolution of up to 1−5 microns [3]. However, the analysis and interpretation of OCT data are complex due to the need to analyze large volumes of data and the time-consuming nature of the research, as well as the possibility of missing minor and early signs of disease or rare pathologies by an ophthalmologist [4].

In recent years, with the development of artificial intelligence (AI) technologies and machine learning, there has been growing interest in applying these methods to automate the process of OCT image analysis. A distinctive feature of OCT data, as well as other medical data, is the subtle structural changes that can indicate the presence of disease. For example, disease identification often requires not only detecting pathology, but also determining its location and volume [5]. Traditional image processing methods cannot effectively identify such small anomalies, which necessitates the use of more sophisticated deep learning (DL) models capable of accounting for spatial and contextual dependencies within OCT images [6]. According to the research, systems based on such AI technologies as deep neural networks (DNN) can significantly improve the accuracy and speed of diagnosing ophthalmic diseases [7, 8].

A key factor in the development of AI technologies in the field of medicine is the availability of high-quality datasets [9]. To date, numerous results have been obtained using open datasets for creating models for analyzing OCT images [10−12]. Moreover, a key task in the development of AI systems for OCT image analysis is choosing a neural network architecture that will efficiently perform classification tasks. The most common types of networks are Convolutional Neural Networks (CNNs), whose main distinguishing feature is their ability to account for spatial hierarchical dependencies in data, allowing them to effectively identify complex patterns and structures in the images [8, 9, 13]. An important aspect of choosing a neural network architecture is also the balance between model complexity and the required computational resources. One promising direction in this area is the use of transfer learning, which allows pre-trained models on large datasets to be adapted to a specific OCT image classification task. The works [14, 15] show that transfer learning can significantly improve classification results, especially in the conditions of limited training data.

The novelty of this study is the creation of a comprehensive OCT image analysis system that integrates various modules, including segmentation, classification, and quantitative assessment of biomarkers. In previous stages of the research, we developed and tested a segmentation model using clinical data for determining the type and quantitative parameters of biomarkers on OCT images [16]. The aim of this study was to compare the effectiveness of different neural network architectures for the task of OCT image classification to develop a disease classification module. The practical significance of this work is the integration of the neural network based model with the best accuracy into a chatbot to support doctors and patients by providing automatic interpretations and recommendations.

Materials and Methods

In this study, we used the publicly available OCTDL dataset, which consists of 2064 images from 821 patients [10]. The images are B-scans in .jpg format obtained using the Optovue Avanti RTVue XR optical coherence tomography scanner with a raster scan protocol. The images were grouped into the following categories:
- Age-related Macular Degeneration (AMD),
- Diabetic Macular Edema (DME),
- Epiretinal Membrane (ERM),
- Normal (NO),
- Retinal Arterial Occlusion (RAO),
- Retinal Venous Occlusion (RVO),
- Vitreomacular Interface Disorder (VID) (Fig. 1).

Table 1 provides a description of the dataset. The entire dataset was divided into training, validation, and test sets in a 65/15/20 ratio to achieve an optimal balance between the amount of data for training
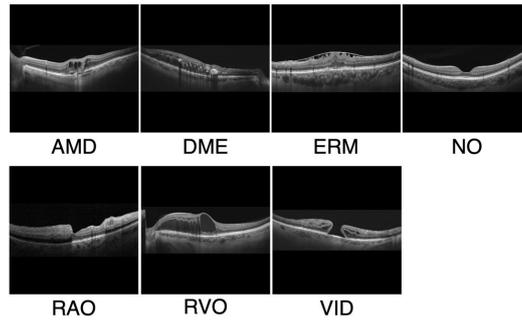
Fig. 1. Examples of the images from the publicly available OCTDL dataset

and evaluation on the independent images (Fig. 2). The distribution of images across the subsets also considered the proportion of each class in the overall dataset to minimize the impact of class imbalance.

Table 1

Distribution of image classes in the training, validation and test sets

Class	Training Set	Validation Set	Test Set	Total Images
AMD	801	192	238	1231
DME	96	21	30	147
ERM	100	20	35	155
NO	216	36	80	332
RAO	12	1	9	22
RVO	66	13	22	101
VID	49	12	15	76

For the classification of OCT images, we used four types of DNNs: ResNet50, ResNet101, InceptionV4, and VGG16. These architectures have shown efficiency in the image recognition for tasks such as skin cancer diagnosis [17], early-stage Alzheimer's disease detection [18], and retinal vessel detection in fundus images [19], as well as in quality assessment and classification of OCT images [20, 21].

The neural network architectures ResNet50, ResNet101, InceptionV4, and VGG16 are deep CNNs with varying numbers of layers, which were pretrained on large image datasets, such as ImageNet [22]. ResNet50 consists of 50 layers and employs "residual blocks." ResNet101 has a similar structure, but comprises 101 layers. InceptionV4 is a DNN with 22 layers that includes "Inception modules" designed to reduce parameters, speed up computations, and prevent overfitting [23]. The VGG16 architecture includes 13 convolutional layers, 5 pooling layers, and 3 fully connected layers, and it has a simple and deep structure [9]. Prior to training the models, we performed data preprocessing and augmentation. The preprocessing involved resizing images to 224x224 pixels, and augmentation techniques included random cropping, horizontal and vertical flipping, rotation, shifting, and Gaussian blurring. The performance of the models was evaluated using the following metrics: Accuracy (1), Precision (2), Recall (3), F1 Score (4), and AUC-ROC.

Accuracy (the proportion of correctly classified objects) measures the proportion of correct predictions among all predictions. This is a basic metric for assessing the overall effectiveness of a model.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN},$$ (1)
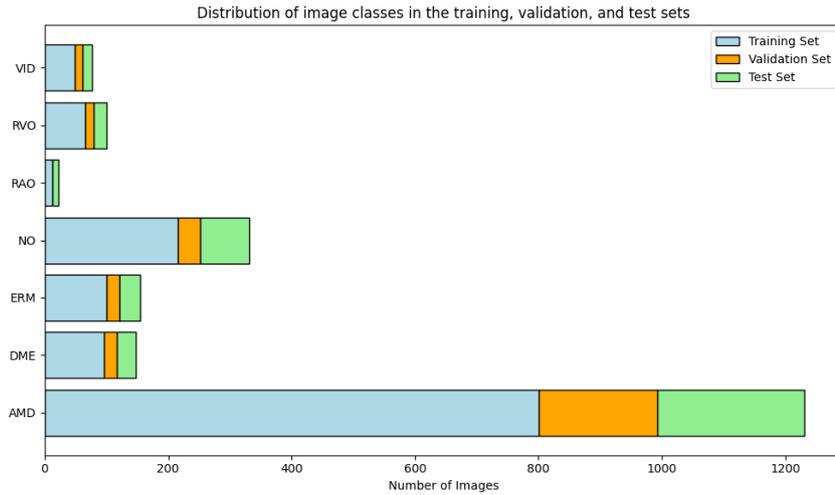
Fig. 2. Visualization of the distribution of images across datasets

where *TP* are True Positives, *TN* are True Negatives, *FP* are False Positives, and *FN* are False Negatives.

Precision (proportion of true positive results) measures the proportion of true positive results among all results classified as positive:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{2}$$

Recall measures the proportion of true positive results among all actual positive cases:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{3}$$

F1 Score is the harmonic mean of precision and recall, making it useful in tasks, where the balance between the precision and the recall is important:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{4}$$

AUC-ROC (Area under the Receiver Operating Characteristic Curve) is a measure of the model's ability to distinguish between classes. The ROC curve is a plot that shows the model's performance at all classification thresholds. AUC-ROC is calculated as the area under the ROC curve, which is constructed based on different values of sensitivity (recall) and specificity (1 − False Positive Rate). These metrics were calculated for each class using the "one vs rest" method. To evaluate the metrics for the entire dataset, the average value of the obtained metrics was calculated. As performance characteristics of machine learning models, training time, prediction time, and the number of parameters were measured.

For training the models, we used the early stopping algorithm. Early stopping was triggered, when the Accuracy metric, measured on the validation dataset, reached its maximum value.

Training and evaluation of the models were performed using the PyTorch DL framework (version 2.1.1) and cloud resources from Yandex Cloud (Yandex Cloud Documentation), which were provided as a part of the Yandex Cloud Boost program in a configuration with vCPUs on the Intel Broadwell platform and GPU NVIDIA® Tesla® V100.

To ensure user interaction with the model, an infrastructure was developed that included a chatbot[1] integrated with the DL model. For integrating the model into the chatbot, a server-side component was created in Python using the Django framework. To ensure flexibility and scalability, the system was containerized using Docker and deployed on Yandex Cloud resources.

Results and Discussion

In this study, models incorporating neural network architectures, such as VGG16, InceptionV4, ResNet50, and ResNet101 were trained on the OCTDL open dataset. To evaluate the performance of the models, metrics, such as Accuracy, F1 Score, Precision, Recall, and AUC-ROC were calculated, along with efficiency characteristics. Table 2 presents the metric values measured on the test set across all the classes for four models.

Table 2

Quantitative performance indicators of machine learning models

	ResNet50	VGG16	InceptionV4	ResNet101
Accuracy	0.93	0.92	0.93	0.91
F1 Score	0.89	0.87	0.89	0.86
Precision	0.88	0.86	0.89	0.87
Recall	0.91	0.9	0.89	0.86
AUC-ROC	0.99	0.99	0.98	0.99
Number of Epochs	100	100	100	100
Training Time (seconds)	2913	3243	1890	1032
Prediction Time (seconds)	32	38	29	20
Number of Training Parameters (millions)	25.6	138	42.7	44.5

All four models demonstrated relatively high accuracy. The model with the ResNet50 architecture achieved the highest values for Accuracy (0.93), F1 Score (0.89), Recall (0.91), and AUC-ROC (0.99). InceptionV4 also achieved the same values for Accuracy and F1 Score. However, the Precision value (0.89) for this architecture was the highest among all, while Recall and AUC-ROC were 0.89 and 0.98, respectively.

Table 3 provides the performance values for the classification of individual ophthalmic disease classes. All models show satisfactory Accuracy values (above 0.8) for all classes. The Accuracy value for the RAO class is 1, which is associated with a lack of data in the test sample.

Fig. 3 allows for a visual comparison of accuracy values across four different neural network architectures — ResNet50, VGG16, InceptionV4, and ResNet101 — relative to metrics, such as Precision, Recall, and F1 Score. The graph shows that ResNet50 and InceptionV4 architectures demonstrate the most stable and highest metric values for almost all the classes of ophthalmic diseases. A noticeable decrease in accuracy is observed for the VGG16 model for the RVO class, which may be related to the insufficient volume of data for this class.

Fig. 4 shows the results of the confusion matrix calculation relative to the classes of ophthalmic diseases for four neural network architectures. The most undesirable outcome in the clinical practice of automated algorithms is classifying a normal image, when there is a pathology present. From the table, it can be concluded that the ResNet50 and InceptionV4 architectures only misclassified in the AMD class, which was the most represented class in the test dataset. The VGG16 and ResNet101 architectures misclassified both AMD and ERM classes.

---

[1] Telegram: Contact @eye_tech_bot. Available: https://t.me/eye_tech_bot (accessed 11.10.2024)

Performance values for the classification of individual ophthalmic disease classes

ResNet50				
Class	**Precision**	**Recall**	**F1 Score**	**Total images**
AMD	0.97	0.96	0.97	238
DME	0.82	0.90	0.86	30
ERM	0.86	0.91	0.89	35
NO	0.92	0.88	0.90	80
RAO	1.00	1.00	1.00	9
RVO	0.74	0.77	0.76	22
VID	0.82	0.93	0.87	15
VGG16				
AMD	0.99	0.95	0.97	238
DME	0.76	0.93	0.84	30
ERM	0.78	0.91	0.84	35
NO	0.95	0.90	0.92	80
RAO	1.00	1.00	1.00	9
RVO	0.82	0.64	0.72	22
VID	0.74	0.93	0.82	15
InceptionV4				
AMD	0.97	0.95	0.96	238
DME	0.79	0.90	0.84	30
ERM	0.87	0.94	0.90	35
NO	0.92	0.91	0.92	80
RAO	1.00	1.00	1.00	9
RVO	0.75	0.68	0.71	22
VID	0.93	0.87	0.90	15
ResNet101				
AMD	0.97	0.96	0.97	238
DME	0.81	0.87	0.84	30
ERM	0.75	0.94	0.84	35
NO	0.92	0.88	0.90	80
RAO	1.00	1.00	1.00	9
RVO	0.70	0.64	0.67	22
VID	0.92	0.73	0.81	15

Neural network architectures, such as ResNet50, ResNet101, InceptionV4, and VGG16, have also been used by other researchers for image classification tasks. In [11], the OCTDL dataset was employed to train VGG16 and ResNet50 models. The Accuracy, F1 Score, and Recall values for the ResNet50 architecture were lower than those obtained in the current study, by 0.8, 0.2, and 0.6, respectively [10]. In [13, 23], the high efficiency of ResNet and Inception models for medical image classification tasks was demonstrated. In [8], the ResNet model achieved an accuracy of 0.97 for OCT image classification, while in [5] an accuracy of 0.95 using an Inception-based model was achieved.

Fig. 3. The results of evaluating the accuracy of machine learning models for each class

The developed model was integrated into a chatbot. The response time for producing results is less than 1 sec, ensuring prompt feedback and enabling rapid analysis even with a high data flow. An example of the response received by a user through the chatbot is shown in Fig. 5. This study did not compare these results with the speed of data assessment by clinicians. Nevertheless, according to the literature, an AI system for radiological image analysis reduced the interpretation time from 11.2 to 2.7 days, high-lighting the efficiency of automated systems in optimizing healthcare workflows and improving patient care standards [24].
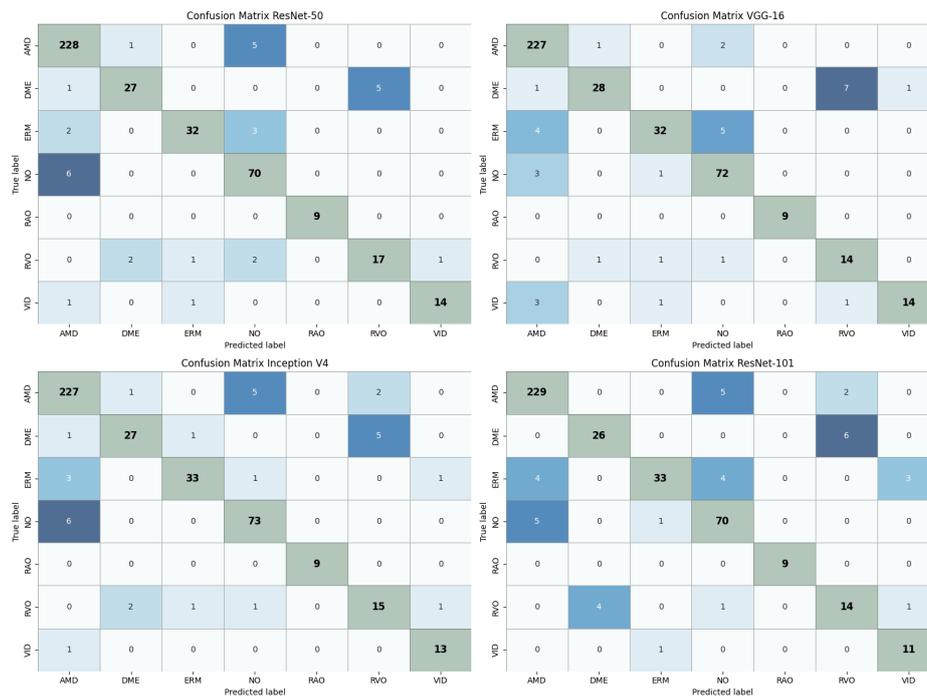
Fig. 4. Confusion matrix for four neural network architectures

Fig. 5. An example of OCT data analysis results using a chatbot integrated with a model
based on the ResNet50 architecture. The results for image 2. Classification results:
1. Age-related macular degeneration − 98.13%; 2. Norm − 0.61%; 3. Vitreomacular traction − 0.39%

Conclusion

In this study, we identified the model with the highest accuracy for evaluating seven classes of ophthalmological diseases. This model is integrated into the chatbot and provides the doctor with a preliminary result of the presence of a pathology in less than 1 second. Thus, the results of this study significantly simplify and accelerate the process of data analysis for ophthalmologists, as well as allow patients to receive an initial consultation anytime and anywhere.

REFERENCES

1. **Steinmetz J.D, Bourne R.R.A., Briant P.S. et al.** GBD 2019 Blindness and Vision Impairment collaborators. Causes of blindness and vision impairment in 2020 and trends over 30 years, and prevalence of avoidable blindness in relation to VISION 2020: the Right to Sight: an analysis for the Global Burden of Disease Study. The Lancet Global Health, 2021, Vol. 9, No. 2, Pp. e144–e160. DOI: 10.1016/S2214-109X(20)30489-7

2. **Zeppieri M., Marsili S., Enaholo E.S., Shuaibu A.O., Uwagboe N., Salati C., Spadea L., Musa M.** Optical Coherence Tomography (OCT): A Brief Look at the Uses and Technological Evolution of Ophthalmology. Medicina, 2023, Vol. 59, No. 12, Article no. 2114. DOI: 10.3390/medicina59122114

3. **Zheng S., Bai Y., Xu Z., Liu P., Ni G.** Optical Coherence Tomography for Three-Dimensional Imaging in the Biomedical Field: A Review. Frontiers in Physics, 2021, Vol. 9, Article no. 744346. DOI: 10.3389/fphy.2021.744346

4. **Hussain A., Oestreicher J.** Clinical decision-making: heuristics and cognitive biases for the ophthalmologist. Survey of Ophthalmology, 2018, Vol. 63, No. 1, Pp. 119–124. DOI: 10.1016/j.survophthal.2017.08.007

5. **De Fauw J., Ledsam J.R., Romera-Paredes B. et al.** Clinically applicable deep learning for diagnosis and referral in retinal disease. Nature Medicine, 2018, Vol. 24, Pp. 1342–1350. DOI: 10.1038/s41591-018-0107-6

6. **Wang J., Hormel T.T., Gao L., Zang P., Guo Y., Wang X., Bailey S.T., Jia Y.** Automated diagnosis and segmentation of choroidal neovascularization in OCT angiography using deep learning. Biomedical Optics Express, 2020, Vol. 11, Pp. 927–944. DOI: 10.1364/BOE.379977

7. **Grassmann F., Mengelkamp J., Brandl C., Harsch S., Zimmermann M.E., Linkohr B., Peters A., Heid I.M., Palm C., Weber B.H.F.** A Deep Learning Algorithm for Prediction of Age-Related Eye Disease Study Severity Scale for Age-Related Macular Degeneration from Color Fundus Photography. Ophthalmology, 2018, Vol. 125, No. 9, Pp. 1410–1420. DOI: 10.1016/j.ophtha.2018.02.037

8. **Kermany D.S., Goldbaum M., Cai W. et al.** Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. Cell, 2018, Vol. 172, No. 5, Pp. 1122–1131. DOI: 10.1016/j.cell.2018.02.010

9. **Simonyan K., Zisserman A.** Very Deep Convolutional Networks for Large-Scale Image Recognition. 3rd International Conference on Learning Representations (ICLR 2015), 2015, Pp. 1–14.

10. **Kulyabin M., Zhdanov A., Nikiforova A. et al.** OCTDL: Optical Coherence Tomography Dataset for Image-Based Deep Learning Methods. Scientific Data, 2024, Vol. 11, Article no. 365. DOI: 10.1038/s41597-024-03182-7

11. **Gholami P., Roy P., Parthasarathy M.K., Lakshminarayanan V.** OCTID: Optical Coherence Tomography Image Database. Computers and Electrical Engineering, 2018, Vol. 81, Article no. 106532. DOI: 10.1016/j.compeleceng.2019.106532

12. **Kermany D., Zhang K., Goldbaum M.** Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification. Mendeley Data, 2018, V2. DOI: 10.17632/rscbjbr9sj.2

13. **He K., Zhang X., Ren S., Sun J.** Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, Pp. 770–778. DOI: 10.1109/CVPR.2016.90

14. **Shin H.C., Roth H.R., Gao M., Lu L., Xu Z., Nogues I., Yao J., Mollura D., Summers R.M.** Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. IEEE Transactions on Medical Imaging, 2016, Vol. 35, No. 5, Pp. 1285–1298. DOI: 10.1109/TMI.2016.2528162

15. **Tajbakhsh N., Shin J.Y., Gurudu S.R., Hurst R.T., Kendall C.B., Gotway M.B., Liang J.** Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning? IEEE Transactions on Medical Imaging, 2016. Vol. 35, No. 5, Pp. 1299–1312. DOI: 10.1109/TMI.2016.2535302

16. **Aksenova L.E., Aksenov K.D., Kozina E.V., Myasnikova V.V.** Automated System for Analysis of OCT Retina Images Development and Testing. Doklady Mathematics, 2024, Vol. 108, Pp. 310–316. DOI: 10.1134/S1064562423701545

17. **Esteva A., Kuprel B., Novoa R. et al.** Dermatologist-level classification of skin cancer with deep neural networks. Nature, 2017, Vol. 542, Pp. 115−118. DOI: 10.1038/nature21056

18. Liu S., Liu S., Cai W., Pujol S., Kikinis R., Feng D. Early diagnosis of Alzheimer's disease with deep learning. 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI), 2014, Pp. 1015−1018. DOI: 10.1109/ISBI.2014.6868045

19. **Maji D., Santara A., Ghosh S., Sheet D., Mitra P.** Deep neural network and random forest hybrid architecture for learning to detect retinal vessels in fundus images. 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2015, Pp. 3029−3032. DOI: 10.1109/EMBC.2015.7319030

20. **Subramanian M., Shanmugavadivel K., Naren O.S., Premkumar K., Rankish K.** Classification of Retinal OCT Images Using Deep Learning. 2022 International Conference on Computer Communication and Informatics (ICCCI), 2022, Pp. 1−7. DOI: 10.1109/ICCCI54379.2022.9740985

21. **Inferrera L., Borsatti L., Miladinovic A., Marangoni D., Giglio R., Accardo A., Tognetto D.** OCT-based deep-learning models for the identification of retinal key signs. Scientific Reports, 2023, Vol. 13, Article no. 14628. DOI: 10.1038/S41598-023-41362-4

22. **Krizhevsky A., Sutskever I., Hinton G.E.** ImageNet Classification with Deep Convolutional Neural Networks. Communications of the ACM, 2017, Vol. 60, No. 6, Pp. 84−90. DOI: 10.1145/3065386

23. **Szegedy C. et al.** Going deeper with convolutions. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, Pp. 1−9. DOI: 10.1109/CVPR.2015.7298594

24. **Annarumma M., Withey S.J., Bakewell R.J., Pesce E., Goh V., Montana G.** Automated Triaging of Adult Chest Radiographs with Deep Artificial Neural Networks. Radiology, 2019, Vol. 291, No. 1, Pp. 196−202. DOI: 10.1148/radiol.2018180921

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Aksenova Lyubov E.
Аксенова Любовь Евгеньевна
E-mail: axenovalubov@gmail.com
ORCID: https://orcid.org/0000-0003-0885-1355

Aksenov Kirill D.
Аксенов Кирилл Дмитриевич
E-mail: axenov.kir@gmail.com
ORCID: https://orcid.org/0000-0001-5391-5229

Prysyazhnyuk Anton V.
Присяжнюк Антон Владимирович
E-mail: kagehitori@gmail.com

Myasnikova Viktoria V.
Мясникова Виктория Владимировна
E-mail: vivlad7@mail.ru

Krasov Andrei V.
Красов Андрей Владимирович
E-mail: krasov@inbox.ru

Circuits and Systems for Receiving, Transmitting, and Signal Processing
Устройства и системы передачи, приема и обработки сигналов

MONOLITHIC INTEGRATED CIRCUIT OF A FOUR-CHANNEL SWITCHED FILTER BANK FOR THE CENTIMETER BAND BASED ON GaAs pHEMT TECHNOLOGY

D.V. Klimenko[1], A.B. Nikitin[2] ✉ , A.A. Stroganov[2]

[1] Special Technological Center, Ltd., St. Petersburg, Russian Federation;
[2] Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

✉ nikitin@mail.spbstu.ru

Abstract. This article presents the results of the design of a four-channel switched filter bank for the centimeter band manufactured as a monolithic microwave integrated circuit based on domestic GaAs pHEMT technology. The switched filter bank includes a set of bandpass filters operating in four sub-bands of most of the C-, X- and Ku-bands, as well as broadband SP4T switches. The bandpass filters of the lower sub-bands are designed using lumped elements, and higher sub-bands filters are designed using microstrip hairpin resonators. The SP4T switch is based on SPST switches, each of which contains one series- and three parallel-connected field-effect transistors. The switched filter bank has following parameters: insertion losses of no more than 10.8 dB, stopband suppression of at least 43 dB at 30% offset or more from the passband center frequency, and a voltage standing wave ratio of no more than 1.8 in the passband.

МОНОЛИТНАЯ ИНТЕГРАЛЬНАЯ СХЕМА ЧЕТЫРЕХКАНАЛЬНОГО ПЕРЕКЛЮЧАЕМОГО БАНКА ФИЛЬТРОВ САНТИМЕТРОВОГО ДИАПАЗОНА НА ОСНОВЕ GaAs pHEMT-ТЕХНОЛОГИИ

Д.В. Клименко[1], А.Б. Никитин[1] ✉ , А.А. Строганов[2]

[1] ООО «Специальный технологический центр»,
Санкт-Петербург, Российская Федерация;
[2] Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

✉ nikitin@mail.spbstu.ru

Аннотация. В данной статье представлены результаты разработки четырехканального переключаемого банка фильтров сантиметрового диапазона частот, реализованного в виде СВЧ монолитной интегральной схемы на основе отечественной GaAs pHEMT технологии. Переключаемый банк фильтров включает в себя набор из четырех полосовых фильтров, работающих в большей части C-, X- и Ku-диапазонов, а также широкополосные SP4T коммутаторы. Полосовые фильтры низших поддиапазонов выполнены на основе сосредоточенных LC-элементов, а фильтры высокочастотной области — на микрополосковых шпилечных резонаторах. Переключатель SP4T построен на основе SPST переключателей, каждый из которых содержит один последовательно и три параллельно включенных полевых транзистора. Переключаемый банк фильтров имеет вносимые потери не более 10,8 дБ, подавление в полосе заграждения не менее 43 дБ при отстройках от центральной частоты не менее чем на ±30% и коэффициент стоячей волны по напряжению входа и выхода не более 1,8.

Ключевые слова: СВЧ монолитная интегральная схема, переключаемый банк фильтров, полосовой фильтр, SP4T-переключатель, GaAs pHEMT

Introduction

A switched filter bank is a key part of the front-end module in broadband transceivers for various purposes. The main task of the switched filter bank is to suppress unwanted interferences, which depends mostly on the frequency characteristics of the bandpass filters. For the microwave range, the choice is between filters on microstrip structures (interdigital, hairpin etc.) and on LC-elements [1].

A centimeter band switched filter bank can be designed using monolithic integrated circuit technology [2—6], or printed circuit boards [7—10].

In the switched filter bank, the frequency sub-band selection is carried out using solid-state microwave switches [11—14]. These switches can be implemented on the basis of transistor [2, 5, 7] and pin-diodes [4, 8, 9, 11]. A widely used structure of the switched N-channel filter bank is a combination

of N filters connected to two SPNT switches located at the input and output of a device [2, 4, 5, 7−10]. Along with this structure, schemes are used, in which switching and filtering functions are combined in one cascade. This solution makes it possible to reduce the size of a monolithic microwave integrated circuit (MMIC) by eliminating the input and output switches of SPNT [3, 6].

The switched filter banks often play a significant role in determining the final frequency characteristics, dimensions and cost for many radio systems. Therefore, the implementation of such a device in the form of MMIC allows to solve the problem of reducing dimensions and increasing reliability of the final product in comparison with a hybrid implementation. A significant number of works are devoted to the design of switched filter bank MMICs. But despite this, the devices considered in them, for the most part, do not fully meet the requirements for Russian radio system manufacturers: significantly different operating frequency bands, bandwidths of channel filters [2, 3, 5, 8, 9] or technology used for MMIC manufacturing [4, 6, 7, 10].

Taking this circumstance into account, as well as restrictions on the supply of imported electronic devices and components to Russia, the task of the centimeter band switched filter bank design becomes more relevant [15].

This paper considers the issues of the four-channel switched filter bank design. The studied MMIC covers C-, X- and Ku-bands. It was designed based on the domestic GaAs pHEMT technology using complex design tools for solid-state microwave devices for the 0.5 μm technological process (PDK_pHEMT05D) [16, 17].

Bandpass filters

The studied switched filter bank contains four channels with filters operating in different frequency sub-bands: No. 1 − 6 GHz ... 8 GHz, No. 2 − 7.5 GHz ... 11.4 GHz, No. 3 − 11 GHz ... 15 GHz, No. 4 − 14.5 GHz ... 18 GHz. The maximum allowed insertion loss level was determined for each bandpass filter: No.1 − 6 dB, No. 2 − 5 dB, No. 3 − 5 dB, No. 4 − 4 dB. Also, all filters must ensure VSWR in the operating frequency band no more than 1.6 and at least 45 dB stopband suppression at 30% offset or more from the passband center frequency.

A comparative analysis of various options for bandpass filters design has shown that for the considered sub-bands, in order to achieve the best frequency characteristics, it is advisable to use different types of filters: filters on lumped elements for lower sub-bands, and filters on distributed elements for higher sub-bands. The use of filters on microstrip structures for low sub-bands leads to a significant increase in the size of the circuits. The conducted studies have shown that it is easier for filters on elements with distributed parameters to ensure the required level of insertion loss in a given frequency band. It should also be noted that such filters are less sensitive to technological deviation of parameters than filters made on LC-elements.

The bandpass filters under study were designed on the basis of MMIC filters from [18]. These bandpass filters were optimized during the design of the switched filter bank. As shown in Fig. 1, CLC-resonators have simplified the design due to the absence of the need to use matching circuits. Thus, it was possible to reduce the area of filter No. 1 from 3.2 mm² to 2.7 mm² and filter No. 2 from 2.9 mm² to 2.4 mm². These changes did not affect the performance of the circuit: stopband suppression at 30% offset or more from the passband center frequency is more than 49 dB, VSWR is no more than 1.6 in the passband. Fig. 2 shows a response comparison of the bandpass filters based on LC-resonators with matching circuits and CLC-resonators without matching circuits for sub-bands No. 1 and No. 2. All results were obtained on the basis of electromagnetic analysis using the 0.5 μm GaAs pHEMT PDK.

SP4T switch

To switch between filter bank channels, a broadband single-pole four-throw switch (SP4T) was designed. For the considered SP4T switch, the following parameters were determined: the operating
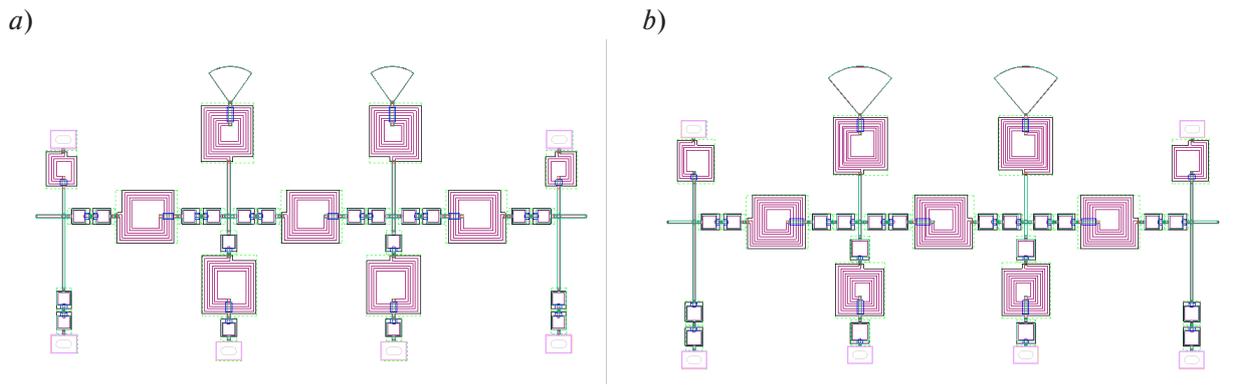
a)   *b)*

Fig. 1. MMIC layouts of optimized bandpass filters No. 1 (*a*) and No. 2 (*b*)

a)   *b)*
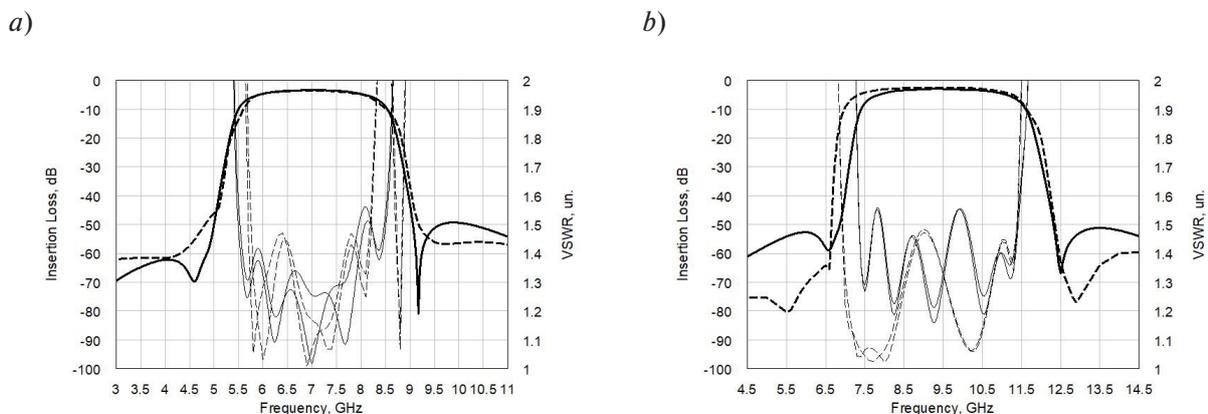
Fig. 2. Frequency responses of bandpass filter No. 1 (*a*) and No. 2 (*b*) with LC resonators
and matching circuits (solid line) and CLC-resonators without matching circuits (dashed line)

frequency band is 6 ... 18 GHz, open channel insertion losses are no more than 3 dB, insertion losses and isolation of closed channels are no less than 30 dB, input and output VSWR of open channels is no more than 1.6.

This switch is designed on the basis of three single-pole two-throw switches (SPDT), each of which contains one series-connected field-effect transistor and three parallel-connected field-effect transistors in each channel. A simplified equivalent scheme of a single SPDT channel is shown in Fig. 3.

To increase the isolation between closed channels, it was decided to add an inductor coil to each channel between parallel-connected field-effect transistors. The frequency characteristics are improved due to the parallel resonance between the inductor and the capacitances of the connected in parallel switched-off transistors. As a result, the resistance of the transistor increases in the closed state. When the channel is closed, the inductor coil is shunted by a small resistance of an open transistor connected in parallel [19].

A series-connected inductor coil has been added to the input of the SP4T switch for 50-ohm matching. Since the SP4T switch is used at the input and output of the switched filter bank, a series-connected capacitor was added to ensure DC isolation with other devices.

The results of SP4T switch electromagnetic simulation are shown in Fig. 4, a. The numbers on the graph indicate insertion losses (1) and decoupling (2) in various operation modes of the switches.

The simulation showed that in the entire frequency band, the insertion losses of open channels is no more than 2.6 dB, the insertion losses of closed channels are no less than 37 dB, and the isolation

Fig. 3. Simplified equivalent scheme of single SPDT channel
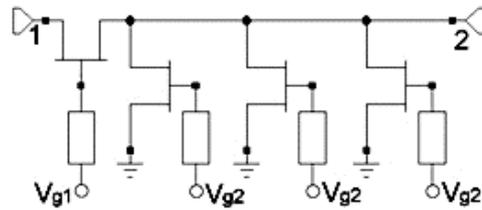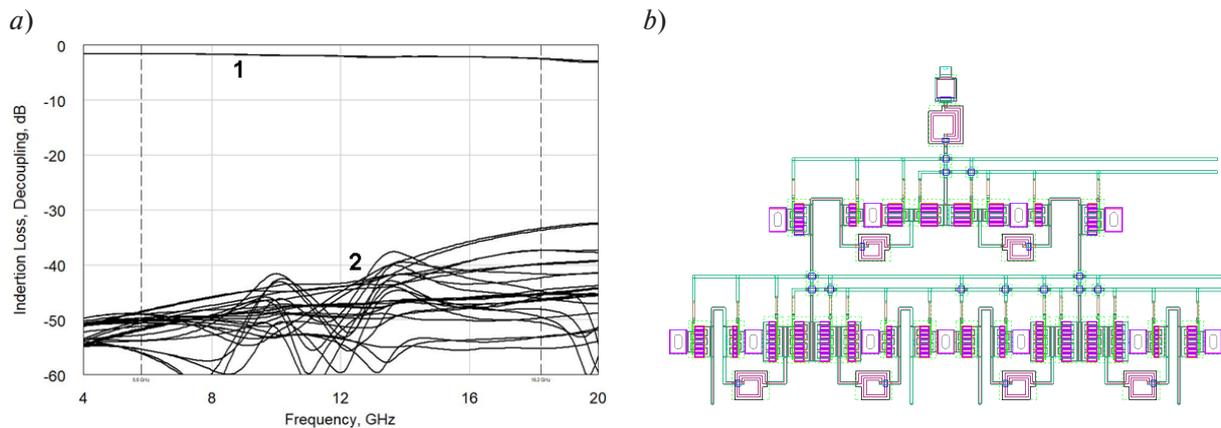
a) *b)*

Fig. 4. Frequency responses of SP4T switch (*a*), layout of SP4T switch (*b*)

between closed channels is no less than 33 dB. The input and output VSWR of open channels in all frequency sub-bands is no more than 1.5. It should be noted that the isolation between closed channels in the switched bank filter increases by the amount of bandpass filters loss. Fig. 4b shows the layout of the SP4T switch. The dimensions are 2.3x1.4 mm.

Because the SP4T switch consists of three SPDT switches and each of them requires two control signals to select an output, six control signals are needed to switch between one of the four channels. However, in this work, the switching between channels is carried out using four control signals. This is possible due to a sufficiently high level of insertion loss of the closed channel of the SPDT switch (no less than 37 dB). In each state of the switched filter bank, one channel of the output SPDT switches is open. However, because one of the channels of the input SPDT switch is closed, the necessary isolation between the outputs of the SP4T switch is provided. To simplify switching by reducing the number of control signals, it is possible to use a control driver.

A comparison of the characteristics of the designed SP4T switch and foreign analogues based on GaAs pHEMT technology is presented in Table 1.

The following parameters are used: $f_1 - f_2$ – minimum and maximum bandwidth frequencies; IL – maximum value of the insertion losses of open channels in the bandwidth; Decoupl. – the minimum value of the insertion losses of closed channels and the isolation between channels in the stopband; VSWR – the maximum value of the input and output VSWR in the bandwidth; Size – MMIC size.

A comparative analysis of the collected data showed that the designed switch surpasses foreign analogues in terms of insertion losses of the open channel and VSWR. In terms of parameters such as decoupling, insertion losses of closed channels and MMIC size, the designed switch is inferior only to MMIC [21]. Thus, the SP4T switch, designed on the basis of domestic GaAs pHEMT technology, demonstrates competitive characteristics.
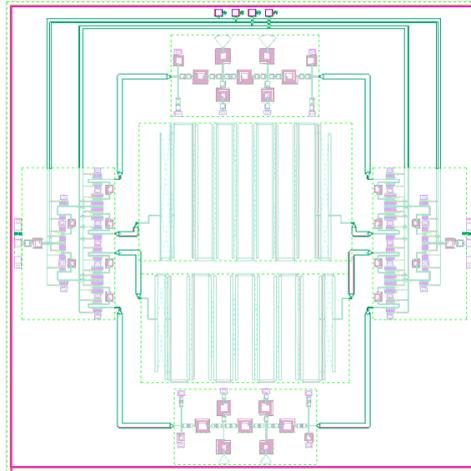
Fig. 5. Switched filter bank layout

Table 1

Characteristics of SP4T switch based on GaAs MMICs

Ref.	$f_1 - f_2$, GHz	IL, dB	Decoupl., dB	VSWR, units	Size, mm²
CMD203C4 [20]	DC-20	≤ 3.1	≥ 22	≤ 2.3	16*
HMC641ALP4E [21]	0.1-20	≤ 2.8	≥ 40	≤ 1.8	3.07
PE42542 [22]	9 kHz − 18	≤ 4	≥ 26	≤ 1.9	16*
This work	6-18	≤ 2.5	≥ 33	≤ 1.45	3.22

*packaged MMIC

Switched filter bank

The final circuit of the switched filter bank consists of two SP4T switches at the input and output and four filters. The layout of the switched filter bank is shown in Fig. 5. The dimensions of the MMIC are 7x7 mm. The filter bank is controlled by four signals supplied to the contact pads located on one side of the MMIC.

50-ohm microstrip lines were used to connect the bandpass filters to the switches. After combining the bandpass filters and switches into a single circuit, the following characteristics were simulated (Fig. 6). Insertion losses are no more than 9.2 dB for channel No. 1, 10.1 dB for channel No. 2, 10.8 dB for channel No. 3 and 10.1 dB for channel No. 4. Stopband suppression at 30% offset or more from the passband center frequency are no less than 50 dB for channel No. 1, 43 dB for channel No. 2, 49 dB for channel No. 3 and 43 dB for channel No. 4. Input and output VSWR in the operating frequency band is no more than 1.6 for channels No. 1 and No. 2, and 1.7 for channels No. 3 and No. 4.

A comparison of the characteristics of the designed switched filter bank and foreign analogues based on GaAs pHEMT technology is presented in Table 2 [23].

The following parameters are used in Table 2: f1 − f2 (FBW) − minimum and maximum bandwidth frequencies (relative bandwidth); IL − the maximum value of the insertion losses of open channels in the bandwidth; Reject. − stopband suppression at 30% offset or more from the passband center frequency; VSWR − maximum value of the input and output VSWR in the passband; Size − MMIC size.
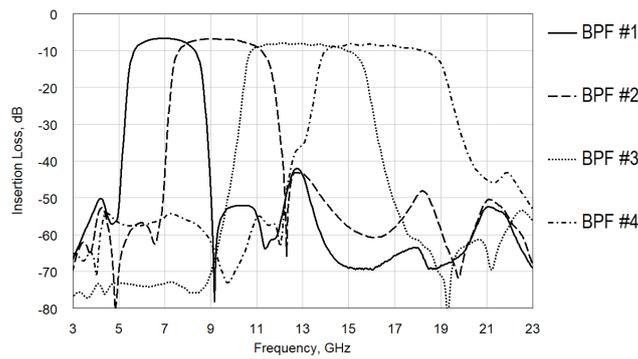
Fig. 6. Frequency responses of switched filter bank channels

Table 2

Characteristics of switched filter bank based on GaAs MMICs

Ref.	$f_1 - f_2$, GHz (FBW %)	IL, dB	Reject., dB	VSWR, units	Size, mm²
BWSBF4-5/18-7C	5−9 (57%)	≤ 11.5	≥ 40	≤ 2	13.5
	8−12 (40%)	≤ 9.5	≥ 40	≤ 2	
	11−15 (31%)	≤ 9	≥ 40	≤ 2	
	14−18 (25%)	≤ 9	≥ 40	≤ 2	
PDSBF-3/20-4	3−5 (50%)	≤ 9	≥ 20	≤ 2	12
	5−8 (46%)	≤ 8.5	≥ 20	≤ 2	
	8−13 (48%)	≤ 9	≥ 20	≤ 2	
	13−20 (44%)	≤ 9.5	≥ 20	≤ 2	
BWSBF5-R8/18-7C8	5−9 (57%)	≤ 10	≥ 40	≤ 1.8	20.25
	8−12 (40%)	≤ 10	≥ 40	≤ 1.8	
	11−15 (31%)	≤ 10	≥ 40	≤ 1.8	
	14−18 (25%)	≤ 9.5	≥ 40	≤ 1.8	
This work	6−8 (29%)	≤ 9.2	≥ 50	≤ 1.6	49
	7.5−11.4 (41%)	≤ 10.1	≥ 43	≤ 1.6	
	11−15 (31%)	≤ 10.8	≥ 49	≤ 1.7	
	14.5−18 (22%)	≤ 10.1	≥ 43	≤ 1.7	

A comparative analysis of the collected data showed that the designed switched filter bank is not inferior to foreign analogues in terms of insertion losses in the bandwidth and surpasses them in terms of stopband suppression and VSWR.

The disadvantage of the designed switched filter bank is its dimensions: the MMIC size is at least twice as large as its analogues. The dimensions of the integrated circuit can be reduced by changing the layout of the SP4T switch. Instead of using three SPDT switches, it is advisable to connect four output paths directly at one point, which will reduce the width of the MMIC. The transformation of the filter structure of sub-bands No. 3 and No. 4 with the transition from microstrip implementation to circuits based on LC-elements can also contribute to reducing the size of the layout. However, as noted above, this may lead to a decrease in the yield of the final product.

One of the options for improvement of the designed switched filter bank is its manufacture in the form of separate MMICs for each subcircuit: switches and filters. The main advantages of a single die are reliability and ease of mounting. The separation of MMIC into individual dies makes it possible to increase

the yield of the final product, since if any integrated circuit is out of order, it is enough to replace only this one, and not the entire die. The disadvantages of implementing a switched filter bank in the form of separate filters and switches include additional insertion losses and a decrease in the reliability of the final product due to the use of a greater number of wire connections. In addition, the mounting of the separated switched filter bank MMICs on a printed circuit board is more complicated.

Conclusion

In the course of the work, the switched filter bank was designed, operating in most of the C-, X- and Ku-bands. As a result of the study, the bandpass filters described in the article [18] were modified, the broadband SP4T switch was designed that covers all the necessary sub-bands and surpasses foreign analogues in most of the characteristics considered.

The designed preselector has four sub-bands in the frequency band with a relative bandwidth of 22−41%. In all four states, the switched filter bank has followings parameters: insertion losses are no more than 10.8 dB, input and output VSWR is no more than 1.8 and stopband suppression at 30% offset or more from the passband center frequency is no less than 43 dB. The switched filter bank is designed on the basis of domestic 0.5 μm GaAs pHEMT technology using integrated design tools for solid-state microwave devices.

Possible ways of improvement of the designed switched filter bank were also determined. In particular, options for reducing the size of the MMIC, simplifying the switching of the filter bank, as well as increasing the yield of the final product are proposed.

REFERENCES

1. **Hong J.-S.** Microstrip Filters for RF/Microwave Applications. 2nd ed. NJ: John Wiley & Sons, Inc. 2011. 635 p.

2. **Han Q., Luo J., Shen H., Chen Q., Qu J.** A 2−6 GHz Fully Integrated Switched Filter Bank for Multiband Wireless Communication Applications. 2019 International Conference on Microwave and Millimeter Wave Technology (ICMMT), 2019, pp. 1−3. DOI: 10.1109/ICMMT45702.2019.8992878

3. **Lu D., Liu J., Yu M.** Highly Selective Bandpass Switch Block with Applications of MMIC SPDT Switch and Switched Filter Bank. IEEE Solid-State Circuits Letters, 2022, Vol. 5, pp. 190−195. DOI: 10.1109/LSSC.2022.3194410

4. **Pang C.H, Han J.Y., Zhu M.S.** A Miniaturized 13−26.5 GHz Filter Bank Based on MMIC and LTCC. 2023 International Conference on Microwave and Millimeter Wave Technology (ICMMT), 2023, pp. 1−3. DOI: 10.1109/ICMMT58241.2023.10276511

5. **Yang X., Xing M., Wang E., Zhang L., Li N., Qian Z.** Design of S-band Monolithic Integrated Switched Filter Bank. 2017 18th International Conference on Electronic Packaging Technology (ICEPT), 2017, pp. 177−181. DOI: 10.1109/ICEPT.2017.8046433

6. **Nam S., Koohi M.Z., Peng W., Mortazawi A.** A Switchless Quad Band Filter Bank Based on Ferroelectric BST FBARs. IEEE Microwave and Wireless Components Letters, 2021, Vol. 31, no. 6, pp. 662−665. DOI: 10.1109/LMWC.2021.3069880

7. **Khan I., Rau A., Sureshkumar S.** Design of Switchable Hairpin Band Pass Filters for Low Frequency Radio Astronomy. 2019 IEEE MTT-S International Microwave and RF Conference (IMARC), 2019, pp. 1−5. DOI: 10.1109/IMaRC45935.2019.9118619

8. **Chandrashekar K, Vanitha Chavan L., Devindra M.C, Prakash S.P., Murthy K.** Design of PIN Diode Based Six Channel Switched Harmonic Filter Bank and Antenna Switch for Software Defined Radio. 2019 IEEE MTT-S International Microwave and RF Conference (IMARC), 2019, pp. 1−4. DOI: 10.1109/IMaRC45935.2019.9118667

9. **Gentili F., Urbani L., Bianchi G., Pelliccia L., Sorrentino R.** p-i-n-Diode-Based Four-Channel Switched Filter Bank With Low-Power TTL-Compatible Driver. IEEE Transactions on Microwave Theory and Techniques, 2014, Vol. 62, no. 12, pp. 3333–3340. DOI: 10.1109/TMTT.2014.2364216

10. **Al Jamal H., Joshi M., Tentzeris M.M., Kingsley N.** A Multi-Channel, Embedded, and Geometrically Optimized Filter Bank Utilizing Advanced Packaging Topologies for Miniaturized RF Modules in IoT and Wearable Systems. 2024 IEEE 74th Electronic Components and Technology Conference (ECTC), 2024, pp. 208–214. DOI: 10.1109/ECTC51529.2024.00042

11. **Raghu B., Sarkar M., Singh N., Mishra A.** Ultra Wideband Reflective SP4T Switch Using MMIC Discrete Components' Topology. 2023 IEEE Microwaves, Antennas, and Propagation Conference (MAP-CON), 2023, pp. 1–5. DOI: 10.1109/MAPCON58678.2023.10464060

12. **Berezniak A.F., Korotkov A.S.** Solid-state Microwave Switches: Circuitry, Manufacturing Technologies and Development Trends. Review (Part 1). Radioelectronics and Communications Systems, 2013, Vol. 56, pp. 159–177. DOI: 10.3103/S0735272713040018

13. **Berezniak A.F., Korotkov A.S.** Solid-state Microwave Switches: Circuitry, Manufacturing Technologies and Development Trends. Review (Part 2). Radioelectronics and Communications Systems, 2013, Vol. 56, pp. 213–226. DOI: 10.3103/S0735272713050014

14. **Kochemasov V., Rautkin Yu.** Integrated Microwave Switches. Part 1. Electronics: Science, Technology, Business, 2018, Vol. 4, pp. 122–127. DOI: 10.22184/1992-4178.2018.175.4.122.127

15. **Kochemasov V., Stroganova H.** Foreign-Made Electronic Components. Russian Exports Restriction. Electronics: Science, Technology, Business, 2013, Vol. 1, pp. 124–129.

16. **Filaretov A.G., Chaly V.P., Shukov I.V., Dudin A.L., Fazylhanov O.R., Krasovitsky D.M.** Setting up a Highly Reliable Solid State Microwave Components Production – Routines and First Experience. Nanoindustry, 2021, Vol. 14 (107), No. 7s, pp. 408–410. DOI: 10.22184/1993-8578.2021.14.7s.408.410

17. **Filaretov A., Chaly V.** Microwave Foundry of Military Quality: from Main Process Parameters to Primitives' Library. Obmen opytom v oblasti sozdaniia sverkhshirokopolosnykh radioelektronnykh sistem [Exchange of experience in the field of creation of ultra-wideband radio-electronic systems], 2020, pp. 232–235. DOI: 10.25206/978-5-8149-3074-3-232-235

18. **Klimenko D.V., Nikitin A.B., Stroganov A.A., Tsikin I.A.** Features of Centimeter-band Filter-bank Design Based on GaAs pHEMT-technology. Computing, Telecommunication and Control, 2023, Vol. 16, no. 3, pp. 7–17. DOI: 10.18721/JCSTCS.16301

19. **Klimenko D.V., Nikitin A.B., Stroganov A.A., Tsikin I.A.** Design of Monolithic Microwave Integrated Circuit X-Band Switch on GaAs pHEMT. Journal Radioengineering, 2024, Vol. 21, no. 3, pp. 92–101. DOI: 10.18127/j00338486-202403-09

20. Qorvo, Inc. Product Data Sheet. DC-20 GHz SP4T Non-reflective Switch. Available: https://www.qorvo.com/products/d/da007458 (Accessed: 09.08.2024).

21. Analog Devices, Inc. Product Data Sheet. 0.1 GHz to 20 GHz GaAs, Nonreflective, SP4T Switch. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/hmc641a.pdf (Accessed: 09.08.2024).

22. pSemi. Product Data Sheet. UltraCMOS SP4T RF Switch 9 kHz-18 GHz. Available: https://www.psemi.com/pdf/datasheets/pe42542ds.pdf (Accessed: 09.08.2024).

23. Saintly-Tech Communications Limited. Product Data Sheet. MMIC Filter Banks. Available: http://www.sainty-tech.com/en/Filter/206.html (Accessed: 09.08.2024).

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Klimenko Denis V.
Клименко Денис Валерьевич
E-mail: devklimenko@stc-spb.ru

Nikitin Aleksandr B.
Никитин Александр Борисович
E-mail: nikitin@mail.spbstu.ru

Stroganov Alexander A.
Строганов Александр Александрович
E-mail: lemyr103@gmail.com

ENHANCED FREQUENCY BAND WIDEBAND RECEIVER USING MILLER *N*-PATH BANDPASS FILTER

T.D. Tran ✉

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

✉ thanhdat140495@gmail.com

Abstract. This paper presents the design and modelling result of the wideband receiver topology with an enhanced frequency band using Miller *N*-path bandpass filter and harmonic-rejection mixing technique. The wideband receiver has the form of monolithic microwave integrated circuits (MMIC) based on domestic GaAs pHEMT technology. The receiver consists of low noise amplifier (LNA), commutated network and harmonic recombination circuit. When using inductors to compensate the parasitic capacitances, the bandwidth of the LNA significantly increases from $0-2.3$ GHz to $0-5$ GHz, thereby increasing the receiver frequency band to $0.3-3$ GHz. The receiver achieves a gain of 15 dB, a noise figure of < 4 dB, an out-of-band IIP3 of +8 dBm, and a harmonic-rejection ratio at the third- and fifth-order local oscillator harmonics of > 50 dB.

Keywords: wideband receiver, multi-band receiver, wideband LNA, harmonic-rejection mixer, Miller *N*-path bandpass filter, MMIC, GaAs pHEMT

Citation: Tran T.D. Enhanced frequency band wideband receiver using *N*-path Miller bandpass filter. Computing, Telecommunications and Control, 2024, Vol. 17, No. 3, Pp. 124–130. DOI: 10.18721/JCSTCS.17312

ШИРОКОПОЛОСНЫЙ ПРИЕМНИК С РАСШИРЕННЫМ ДИАПАЗОНОМ ЧАСТОТ ПРИ ИСПОЛЬЗОВАНИИ ПОЛОСОВОГО *N*-КАНАЛЬНОГО ФИЛЬТРА МИЛЛЕРА

Т.Д. Чан ✉

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

✉ thanhdat140495@gmail.com

Аннотация. Представлены разработка и результаты моделирования топологии широкополосного приемника с расширенным диапазоном частот при использовании полосового *N*-канального фильтра Миллера и метода подавления гармоник. Широкополосный приемник выполнен в виде СВЧ монолитных интегральных схем на основе отечественной GaAs pHEMT технологии. Приемник состоит из малошумящего усилителя (МШУ), коммутируемой цепи и цепи рекомбинации гармоник. При использовании катушки индуктивности для компенсации паразитных емкостей, полоса частот МШУ значительно увеличивается с 0−2,3 ГГц до 0−5 ГГц, тем самым увеличивая полосу частот приемника до 0,3−3 ГГц. Приемник достигает коэффициента усиления 15 дБ, коэффициента шума < 4 дБ, внеполосной IIP3 +8 дБм, коэффициента подавления гармоник на гармониках частоты гетеродина третьего и пятого порядка > 50 дБ.

Ключевые слова: широкополосный приемник, многодиапазонный приемник, широкополосный МШУ, смеситель подавления гармоник, полосовой *N*-канальный фильтр Миллера, МИС, GaAs pHEMT

Introduction

With the increase in the number of standards in the frequency band from 0.3 GHz to 6 GHz (such as LTE, 5G, ...), the number of narrowband receivers in devices designed for each of their own standards increases. Then the device becomes bulky and expensive. Therefore, it is necessary to develop a wideband (multi-band) receiver capable of receiving several standards. In a wideband receiver, such blocks as low noise amplifier (LNA) and mixer are used for several bands and the off-chip surface acoustic wave (SAW)-based bandpass filters are removed. Thus, the size and cost of the device will be saved.

Due to the lack of the off-chip SAW filters, out-of-band blockers enter the receiver without filtering and desensitize it. Therefore, firstly, the blocks located near the antenna, such as the LNA and mixers, must have extremely high linearity. Secondly, these high-power blockers must be filtered as early as possible in the receiving path in order to reduce the linearity requirements for subsequent blocks. In other words, this means that in the receiving path these out-of-band blockers must be filtered before they are amplified. In wideband receivers *N*-path filter is a popular choice for on-chip blocker filtering because of their wide tuning range and high *Q* factor [1−5].
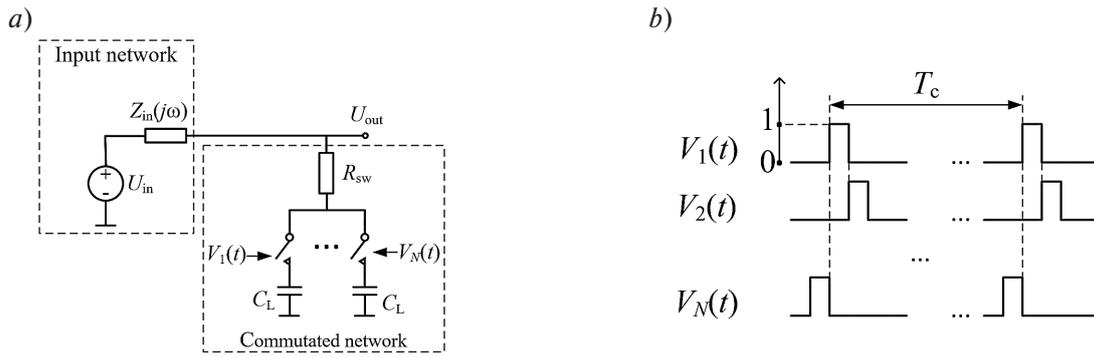
Fig. 1. Schematic of *N*-path filter (*a*), normalized control function of the *k*-th switch (*b*)

This paper describes the design of wideband receiver with enhanced frequency band using *N*-path filtering and harmonic-rejection mixing techniques with Miller effect based on the domestic GaAs pHEMT technology.

Features of N-path filtering technique

Fig. 1*a*, presents the circuit of a *N*-path filter. The input network is modeled by a voltage source $U_{in}(\omega)$ with its impedance $Z_{in}(j\omega)$. In the commutated network *N* switches are periodically commutated with the switching frequency f_c using the non-overlapping square pulses signal $V_{0k}(t)$ (Fig. 1*b*, where $V_k(t) = V_{0k}(t)/V_0$ is normalized control function of the *k*-th switch; V_0 is the maximum value of the pulses). R_{sw} is the resistor of the switch when it is ON.

Using the method used in [6], when $\omega_c/2 < \omega < 3\omega_c/2$ (where $\omega_c = 2\pi f_c$) the output voltage $U_{out}(\omega)$ of the *N*-path filter is calculated by:

$$U_{out}(\omega) = \left[R_{sw} + \frac{c(\omega) Z_{in}(j\omega)}{(1 + g(\omega))(R_{sw} + Z_{in}(j\omega))} \right] \frac{U_{in}(\omega)}{R_{sw} + Z_{in}(j\omega)}, \tag{1}$$

where:

$$c(\omega) = \frac{N \sin\left(\left(\pi - \frac{\pi}{N}\right) b(\omega)\right) \sin\left(\frac{\pi b(\omega)}{N}\right) Z_{CL}(j\omega_c)}{2\pi \sin\left(\left(\frac{\omega - \omega_c}{\omega_c}\right)\pi\right)}; \tag{2}$$

$$Z_{CL}(j\omega_c) = \frac{1}{j\omega_c C_L}; \quad b(\omega) = \frac{\omega}{\omega_c};$$

$$g(\omega) = \sum_{l=-\infty}^{+\infty} \frac{c(\omega)}{(b(\omega) + lN)^2 (R_{sw} + Z_{in}(j\omega + lNj\omega_c))}.$$

For an ideal *N*-path filter, $Z_{in}(j\omega)$ is the resistor R_S. Then from (1) we have the transfer function of the *N*-path filter in the range $\omega_c/2 < \omega < 3\omega_c/2$:

$$K(\omega) = \frac{U_{out}(\omega)}{U_{in}(\omega)} = \frac{R_{sw}}{R_{sw} + R_S} + \frac{\dfrac{c(\omega)R_S}{R_{sw} + R_S}}{R_{sw} + R_S + \displaystyle\sum_{l=-\infty}^{+\infty} \frac{c(\omega)}{(b(\omega)+lN)^2}} \approx$$

$$\approx \frac{R_{sw}}{R_{sw} + R_S} + \frac{c(\omega)}{R_S + \displaystyle\sum_{l=-\infty}^{+\infty} \frac{c(\omega)}{(b(\omega)+lN)^2}}. \tag{3}$$

When ω is close to ω_c, then $(\omega - \omega_c)/\omega_c \approx 0$ and $b(\omega) = \omega/\omega_c \approx 1$. Then from (2) we see that $|c(\omega)|$ has a much larger value than R_S. Then formula (3) becomes:

$$K(\omega) = \frac{R_{sw}}{R_{sw} + R_S} + \frac{c(\omega)}{R_S + \displaystyle\sum_{l=-\infty}^{+\infty} \frac{c(\omega)}{(b(\omega)+lN)^2}} \approx \frac{1}{\displaystyle\sum_{l=-\infty}^{+\infty} \frac{1}{(1+lN)^2}}.$$

For wideband receivers, N is usually 8. Then $K(\omega) \approx 1$. That means that when the frequency ω (ω is close to ω_c) is within the passband of the N-path filter, the maximum value of transfer function $K_{max} \approx 1$.

On the other hand, when the frequency ω is far from ω_c, $|c(\omega)|$ is very small compared to R_S because $|Z_{CL}(j\omega_c)|$ is very small compared to R_S. Then formula (3) becomes:

$$K(\omega) \approx \frac{R_{sw}}{R_{sw} + R_S} \ll 1. \tag{4}$$

This means that when the frequency ω is outside the passband of the N-path filter, the transfer function of the N-path filter $K(\omega)$ is very small compared to 1.

Miller N-path bandpass filter

Formula (4) shows that the out-of-band interferences rejection capability of the N-path filter is limited by $R_{sw}/(R_{sw} + R_S)$. Thus, to decrease $K(\omega)$, when ω is outside the passband of the N-path filter, it is necessary to decrease R_{sw} or/and increase R_S. If R_{sw} is too small, the parasitic capacitance of the switches will be very large and increase the leakage of the control signal $V_{0k}(t)$. Therefore, a more feasible way is to increase R_S. In addition, [7] also shows that the bandwidth of the N-path filter is determined by: $BW = 1/(\pi N R_S C_L)$. For standards with small bandwidths (e.g. from a few hundred kHz to a few MHz), C_L has a fairly large value and significantly increases their area on the chip. Therefore, increasing R_S is also necessary.

Article [7] shows how to increase R_S by placing the commutated network at the LNA's feedback. Then by using Thevenin equivalent model the internal resistance of the equivalent source will increase $(1 - A_0)$ times, where A_0 is the voltage gain of LNA and needs to be a negative real number. For the receiver to work in a wide frequency band, A_0 needs to keep its value constant in that frequency band. With the LNA design in [7], $|A_0|$ is often reduced at 2−3 GHz due to parasitic capacitances, thereby reducing the frequency band of the receiver.

Design of LNA

To increase bandwidth of the LNA in this paper, we add inductors in series with the load resistors of each stage of the LNA developed in [7]. These inductors will compensate the parasitic capacitances and keep $|A_0|$ from decreasing at high frequencies, thereby increasing the bandwidth of the LNA (Fig. 2a).
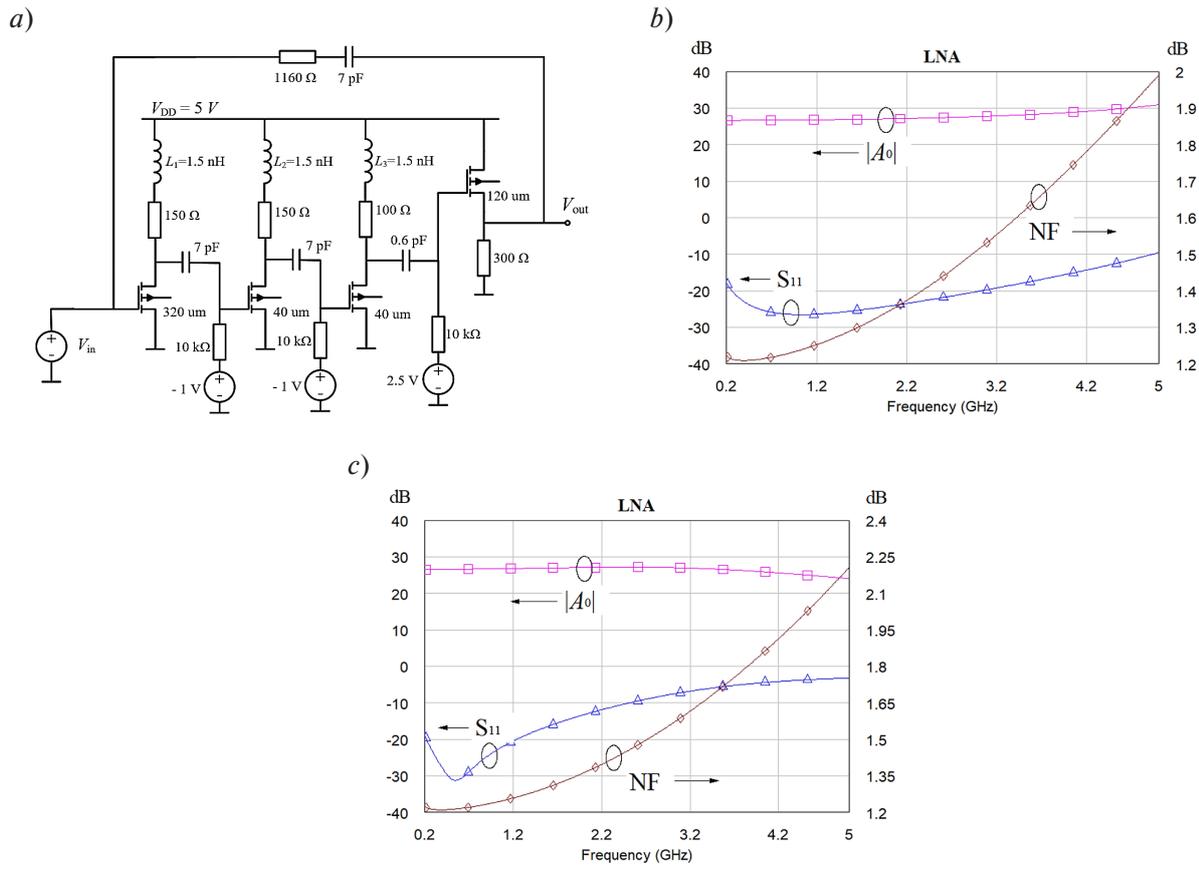
a)

b)

c)

Fig. 2. Schematic of LNA (*a*), frequency responses: S_{11}, noise figure NF and voltage gain $|A_0|$
with inductors (L_1, L_2, L_3) (*b*) and without inductors (L_1, L_2, L_3) (*c*)

The LNA is designed based on the domestic GaAs pHEMT technology using Microwave Office circuit design tool. Frequency responses (S_{11} and $|A_0|$) of the LNA design with inductors (L_1, L_2, L_3) and without inductors (L_1, L_2, L_3) are shown in Fig. 2*b* and 2*c*, respectively. The LNA design with inductors ensures flat voltage gain $|A_0| \approx 27$ dB, noise figure NF < 2 dB and $S_{11} < -10$ dB in the frequency range from 0.2 GHz to about 5 GHz. Meanwhile, for the LNA design without inductors, $S_{11} > -10$ dB when the frequency is greater than 2.5 GHz (Fig. 2*c*). In addition, the voltage gain $|A_0|$ is also reduced at frequencies around 4–5 GHz.

Design of wideband receiver

The schematic of wideband receiver is presented in Fig. 3*a*. It consists of LNA, commutated network and harmonic recombination (HR) circuit. The LNA and commutated network form Miller *N*-path bandpass filter, which acts to pass the desired signal and filter the interferences at the output of the LNA. Therefore, these interferences cannot cause nonlinear effects to the LNA. In addition, in this design, $R_S C_L \gg T_c$ (where T_c — period of $V_k(t)$), so the Miller *N*-path bandpass filter also acts as a mixer [8–9]. This mixer converts the RF frequency voltage at the output of the LNA to the baseband voltage at the capacitors C_L. In this case, the local oscillator (LO) frequency f_{LO} of the mixer is equal to the switching frequency f_c.

On the other hand, due to the lack of filtering of the bandpass filter at the input of the receiver, the mixer converts not only the desired signal at the frequency $f_{in} \approx f_{LO}$, but also the interferences at some harmonics of the LO frequency $k f_{LO}$, ($k \neq \pm 1$). To solve this problem, a HR circuit is used. The principle of operation of the HR circuit is that a "pseudo-sine-LO" signal is generated. Unlike the $V_k(t)$
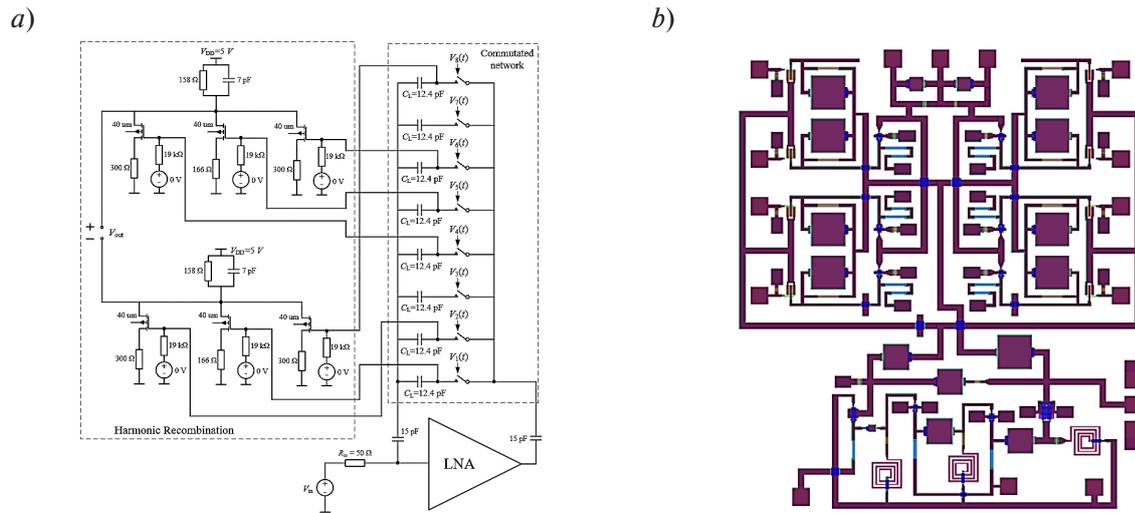
a)   *b)*

Fig. 3. The schematic (*a*), topology (*b*) of wideband receiver

signals, which have a square wave shape, the "pseudo-sine-LO" signal has a shape similar to a sine wave. Therefore, it does not contain 3rd and 5th harmonics. Consequently, interferences in these harmonics are eliminated at the output of HR circuit. In Fig. 3*a*, the baseband voltages at the capacitors C_L are converted to currents with corresponding transconductances (with a ratio of $1 : \sqrt{2} : 1$) by the common-source (CS) amplifiers. They are then summed and converted to voltages at the outputs of these CS amplifiers.

The wideband receiver is designed based on the domestic GaAs pHEMT technology using Microwave Office circuit design tool (Fig. 3*a*). MMIC topology of the receiver designed on its basis (Fig. 3*b*). Its performances are shown in Table.

Table

Performances of the wideband receiver

Parameters	Value
Frequency band of input signal, GHz	0.3−3
Gain, dB	15
Noise figure, dB	< 4
Out-of-band IIP3, dBm	8
HR3, dB	50
HR5, dB	52

where HR3 and HR5 are the ratios of the conversion gain for the desired signal to that for the signals at 3rd and 5th harmonics respectively.

The frequency band of the receiver is reduced compared to the frequency band of LNA due to the parasitic elements of the switches. Simulation results show that if the LNA does not use the inductors (L_1, L_2, L_3), the frequency band of receiver is only 0.3−1.5 GHz.

Conclusion

The design of wideband receiver with enhanced frequency band using Miller N-path bandpass filter is described. By using inductors to compensate the parasitic capacitances, the bandwidth of the LNA is significantly increased from 0−2.3 GHz to 0−5 GHz, thereby increasing the receiver frequency to 0.3−3 GHz. The MMIC topology of the receiver is designed and simulated based on the domestic GaAs

pHEMT technology using Microwave Office circuit design tool. The performances of the receiver in its frequency band: gain is 15 dB, noise figure is < 4 dB, out-of-band IIP3 is 8 dBm, HR3 and HR5 are 50 and 52 dB, respectively.

REFERENCES

1. **Lin F., Mak P.-I., Martins R.P.** Wideband Receivers: Design Challenges, Tradeoffs and State-of-the-Art. IEEE Circuits and Systems Magazine, 2015, Vol. 15, no. 1, pp. 12−24. DOI: 10.1109/MCAS.2014.2385571

2. **Shams N., Nabki F.** Blocker-Tolerant Inductor-Less Harmonic Selection Wideband Receiver Front-End for 5G Applications. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2023, Vol. 31, no. 3, pp. 369−381. DOI: 10.1109/TVLSI.2022.3223123

3. **Elmi M., Tavassoli M., Jalali A.** A wideband receiver front-end using 1st and 3rd harmonics of the N-path filter response. Analog Integrated Circuits and Signal Processing, 2018, Vol. 94, pp. 451−467. DOI: 10.1007/s10470-017-1096-y

4. **Xu Y., Zhu J., Kinget P.R.** A Blocker-Tolerant RF Front End With Harmonic-Rejecting N-Path Filter. IEEE Journal of Solid-State Circuits, 2018, Vol. 53, no. 2, pp. 327−339. DOI: 10.1109/JSSC.2017.2778273

5. **Shams N., Nabki F.** Analysis and Comparison of Low-Power 6-GHz N-Path-Filter-Based Harmonic Selection RF Receiver Front-End Architectures. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2022, Vol. 30, no. 3, pp. 253−266. DOI: 10.1109/TVLSI.2022.3142235

6. **Korotkov A.S., Chan T.D.** Analysis of a Current-Driven Passive Mixer at an Arbitrary Intermediate Frequency with Account of Input and Output Impedances. Journal of Communications Technology and Electronics, 2023, Vol. 68, pp. 77−87. DOI: 10.1134/S1064226923010072

7. **Park J.W., Razavi B.** Channel Selection at RF Using Miller Bandpass Filters. IEEE Journal of Solid-State Circuits, 2014, Vol. 49, no. 12, pp. 3063−3078. DOI: 10.1109/JSSC.2014.2362843

8. **Andrews C., Molnar A.C.** Implications of Passive Mixer Transparency for Impedance Matching and Noise Figure in Passive Mixer-First Receivers. IEEE Transactions on Circuits and Systems I: Regular Papers, 2010, Vol. 57, no. 12, pp. 3092−3103. DOI: 10.1109/TCSI.2010.2052513

9. **Soer M.C.M., Klumperink E.A.M., de Boer P.-T., van Vliet F.E., Nauta B.** Unified Frequency-Domain Analysis of Switched-Series-RC Passive Mixers and Samplers. IEEE Transactions on Circuits and Systems I: Regular Papers, 2010, Vol. 57, no. 10, pp. 2618−2631. DOI: 10.1109/TCSI.2010.2046968

INFORMATION ABOUT AUTHOR / СВЕДЕНИЯ ОБ АВТОРЕ

Tran Thanh Dat
Чан Тхань Дат
E-mail: thanhdat140495@gmail.com

INFLUENCE ANALYSIS OF COMPARATOR PARAMETERS SPREAD ON DECISION ACCURACY IN DAC SELF-CALIBRATION CIRCUIT

M.S. Yenuchenko (ID) , *N.V. Kvashina* (ID) , *I.M. Piatak* ✉

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

✉ i.m.piatak@gmail.com

Abstract. The article presents an analysis of the influence of the comparator parameter spread in the digital-to-analog converter (DAC) self-calibration circuit on the reduction of the conversion nonlinearity. DAC on sources with a switching-based self-calibration circuit is considered. The comparator response threshold value due to the spread of component parameters for 0.18 µm CMOS technology (HCMOS8D by "Mikron") is estimated. The comparator response threshold values are obtained for three sizes of comparator components. Functional modeling of the switching calibration taking into account the finite threshold of element comparison showed that the choice of the sorting algorithm affects the reduction of the conversion nonlinearity. It should be noted that for the smallest comparator option, only quick sort can provide an improvement in the integral nonlinearity for all considered conditions. The optimal size of the comparator components is determined in terms of the efficiency of nonlinearity reduction. The quick sort algorithm shows the best results both in nonlinearity reduction and in the influence of the comparator switching threshold sign.

АНАЛИЗ ВЛИЯНИЯ РАЗБРОСА ПАРАМЕТРОВ КОМПАРАТОРА НА ТОЧНОСТЬ ПРИНЯТИЯ РЕШЕНИЯ В ЦЕПИ САМОКАЛИБРОВКИ ЦАП

М.С. Енученко ⓘD , Н.В. Квашина ⓘD , И.М. Пятак ✉

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

✉ i.m.piatak@gmail.com

Аннотация. Представлен анализ влияния разброса параметров компаратора в цепи самокалибровки цифро-аналогового преобразователя (ЦАП) на снижение нелинейности преобразования. Рассмотрен цифро-аналоговый преобразователь на источниках с цепью коммутационной самокалибровки. Проведена оценка величины порога срабатывания компаратора, обусловленного разбросом параметров компонентов для технологии КМОП 0.18 μm («Микрон» HCMOS8D). Получены значения порогов срабатывания компаратора для трех размеров компонентов компаратора. Функциональное моделирование коммутационной калибровки с учетом конечного порога сравнения элементов показало, что выбор алгоритма сортировки влияет на снижение нелинейности преобразования. При этом отметим, что для наименьшего варианта компаратора только быстрая сортировка может обеспечить улучшение интегральной нелинейности для всех рассматриваемых условий. Определен оптимальный размер компонентов компаратора с точки зрения эффективности снижения нелинейности. Наилучшие результаты как по снижению нелинейности, так и по влиянию знака порога переключения компаратора показывает алгоритм быстрой сортировки.

Ключевые слова: цифро-аналоговый преобразователь, источник тока, коммутационная калибровка, рассогласование, порог компаратора, сортировка элементов

Для цитирования: Yenuchenko M.S., Kvashina N.V., Piatak I.M. Influence analysis of comparator parameters spread on decision accuracy in DAC self-calibration circuit // Computing, Telecommunications and Control. 2024. Т. 17, № 3. С. 131–139. DOI: 10.18721/JCSTCS.17313

Introduction

High-resolution digital-to-analog converters (DACs) impose strict requirements for components matching in order to provide sufficient linearity of conversion. For resolutions higher than 12 bits, the common approach to improve component matching is an electronic calibration. The calibration process includes analysis phase when component values or their ratios are assessed and the information about actual values of elements is used to minimize conversion nonlinearity. For this purpose, switching-based calibrations changes a manner of element switching [1—5].

As switching-based calibration aims to reduce part of analog blocks in favor of digital ones, a comparator is employed for analysis of element values [6—9]. The accuracy of matching assessment depends on the threshold of comparison determined by comparator properties. One of dominant factors that determine comparator threshold is a spread of its component parameters. However, there are no researches examining the effect of mismatch in the comparator on calibration results. This paper focuses
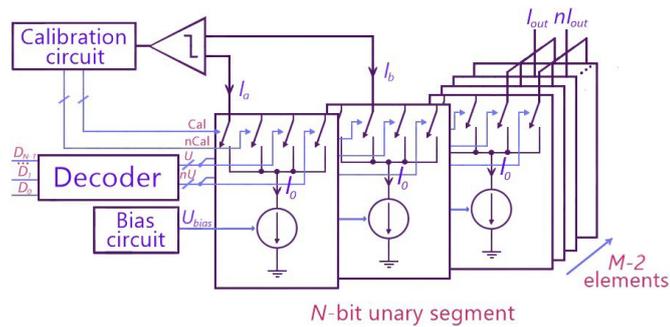
Fig. 1. Structure of a switching-based calibration circuit
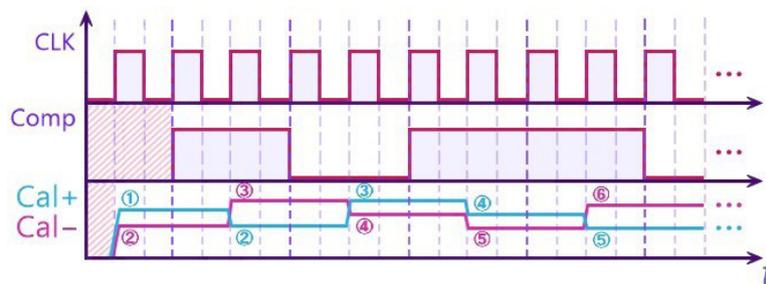
Fig. 2. Time diagrams of a sorting phase

on assessing the influence of parameters on the accuracy of decision making in terms of the self-calibration circuit.

Calibration structure

This paper focuses on a switching-based self-calibration of current-steering DACs. One of the main phases in the switching-based calibration is a sorting phase. During this phase, an ascending (or descending) order of elements is established in accordance with their actual values. After that, a sorted order of elements is used to setup a new elements switching order for minimizing integral nonlinearity (INL). Fig. 1 shows the general structure of such a calibration circuit. The key component of such a calibration performing an assessment of element values is a comparator. A sorting procedure is the following. During the calibration stage, two elements under comparison are switched off from the DAC output and connected to the comparator's input. The comparator waits for the end of switching transition in one clock cycle and compares values of the current sources in the next clock cycle. Fig. 2 illustrates the described process. An obtained decision is used by sorting algorithm for swapping (or not) of elements. Wrong decision from the comparator may break partially or entirely the sorted order of elements.

Comparator circuit

In this paper, a dynamic comparator is considered [10]. It consists of resistor pair, differential amplifier and RS-trigger circuit. Schematics of comparator blocks are depicted in Fig. 3−5. The comparator is designed with 0.18 μm "Mikron" HCMOS8D technology. The current of two elements under comparison is converted into a voltage with resistor pair (I10, Fig. 5). Then their difference is amplified (Fig. 3) and is stored in the RS-trigger (I10, Fig. 4) as a digital signal ("0" or "1"). The comparator is designed to have a nominal current value at the center of supply voltage range. The considered current sources are based on a schematic of cascode current mirror with transistor sizes of 5 μm/4 μm and 16× multiplier.

Fig. 3. Amplifier block

Fig. 4. Amplifier with digital RS-trigger

Process variations effect the properties of transistor and make them different from the nominal values. As the result, the left and right branches become different, and comparator threshold drifts from zero. The effect induced by deviations can be reduced by an increase of components area. Therefore, the optimal area of comparator's components is a matter of concern. The present work investigates this issue in terms of subsequent effects of mismatch on the correctness of sorted order.

Simulation

The following approach is considered for a simulation of the comparator threshold. There are two current sources: one with the constant current value, another changes its current around the value of the first one. A difference between input voltages at the comparator's input, when the output changes, defines the comparator threshold. A Monte-Carlo simulation with 1000 runs is used to collect statistics of threshold value. Moreover, as components' area affects their matching properties and consequently threshold, different area of transistors in the comparator is considered (area of the resistors has not been changed). Simulation results are presented in Fig. 6 and in Table 1.

The threshold has approximately zero mean value as expected and covers both negative and positive values. The standard deviation of the threshold decreases with components' area increase. However, this
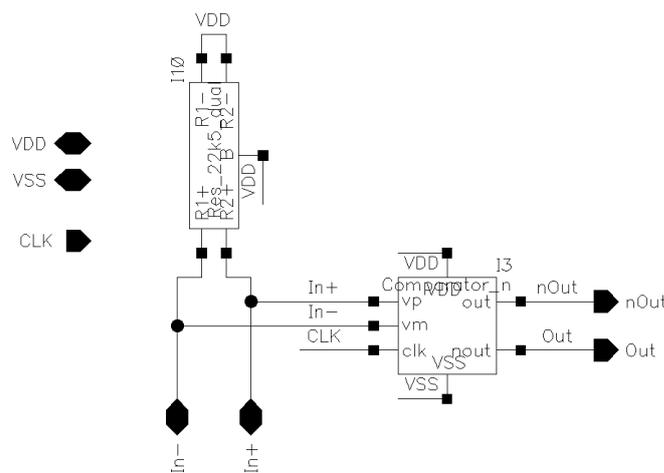
Fig. 5. Comparator block

dependency has the saturation and the threshold deviations do not noticeably (< 3%) decrease, when the transistor sizes increase by 5 times from 20 µm/10 µm to 100 µm/50 µm. Whereas, a transition from 4 µm/2.5 µm to 20 µm/10 µm (×5 area increase) decreases threshold standard deviation by 5.1 times. Therefore, there is an optimal value of components' area in terms of threshold deviation reduction.

The finite value of the threshold leads to wrong decisions when assessing the element values, in particular, elements, whose difference is equal to or less than threshold. To assess, how the threshold leads to calibration decline, a function-level simulation is used. In this type of simulation, an array of unit elements with random errors is generated. The number of generated arrays is 100. The generation of element values relies on assumption that error distribution is normal. To get dispersion of the element distribution, the mismatch of the current source is simulated by Monte-Carlo analysis. Statistics of the current source mismatch is shown in Fig. 7. The mean value is 40.35 µA, the standard deviation of random errors is 0.23%, which converts into 2.1 mV at comparator inputs. The ratio of the threshold and standard deviation of the current source is 0.74, 0.15 and 0.14 relative to the threshold values in Table 1.

Table 1

Simulation results of the comparator threshold

W, µm	L, µm	th_{mean}, mV	th_{σ}, mV
4	2.5	−0.53	1.51
20	10	−0.056	0.30
100	50	0.18	0.29

For the simulation, the following sorting algorithms are considered: bubble sort, merge sort, quick sort and selection sort. For each algorithm, the maximum error between the results of an ideal sort and a certain sorting algorithm is calculated and averaged across all generated arrays. Then, the obtained error is normalized by the threshold values. In order to assess subsequent effect on the calibration results (i.e. INL), the 1F1D algorithm [6] is considered. This algorithm is the simplest one and is used for a demonstration of sorting accuracy effect on the calibration results. The maximum values of INL after calibration are normalized to the INL before calibration and then averaged over all 100 cases for a comprehensive assessment of the calibration effect. Since the threshold also covers negative values along with positive ones, both sign cases are considered.
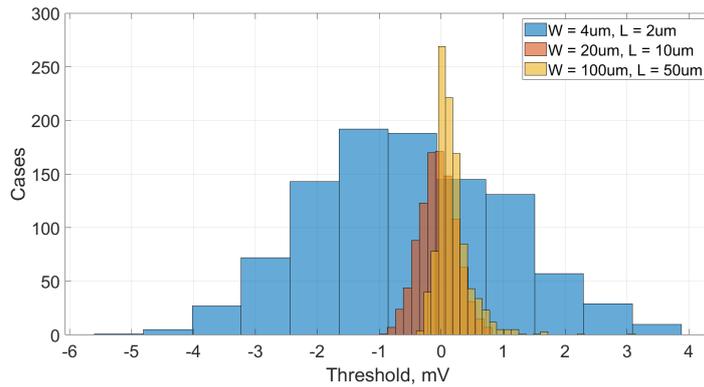
Fig. 6. Threshold statistics

The results are presented in Tables 2 and 3 for different N resolutions and positive and negative threshold values respectively. In terms of positive threshold values, the best error and INL reduction is obtained by the selection sort. The bubble and merge sorts in some conditions demonstrate worse results than before calibration (normalized INL > 1).

Table 2

Simulation results of the calibration (positive threshold)

th_σ, mV	Error/th_σ				Normalized mean max. INL			
	Bubble	**Merge**	**Quick**	**Selection**	**Bubble**	**Merge**	**Quick**	**Selection**
	$N = 6$							
1.51	3.276	2.021	1.488	0.910	1.497	0.993	0.713	0.612
0.30	2.563	1.745	1.649	0.968	0.575	0.560	0.559	0.557
0.29	2.525	1.738	1.607	0.968	0.571	0.559	0.558	0.556
	$N = 8$							
1.51	4.485	2.810	1.303	0.916	1.558	1.617	0.633	0.538
0.30	5.818	2.491	1.823	0.987	0.813	0.470	0.459	0.456
0.29	5.791	2.495	1.811	0.987	0.788	0.468	0.458	0.455
	$N = 10$							
1.51	5.468	3.550	1.261	0.927	1.462	2.975	0.898	0.519
0.30	12.580	3.301	1.681	0.990	2.876	0.438	0.384	0.384
0.29	12.567	3.275	1.711	0.991	2.838	0.436	0.385	0.383

In terms of negative threshold values, the bubble sort has the best results, while the selection sort has the worst ones. Merge and selection sorts in some condition demonstrate worse results than before calibration. The results of merge and quick sorts weakly differ at both positive and negative thresholds, which makes them resilient to the threshold sign altering. However, in case of bubble and selection sorts the threshold sign matters and must be determined.

Conclusion

Mismatch and process variations between the comparator branches leads to an appearance of a comparator threshold. This threshold defines a difference between the input signals, when the comparator switches. If difference is smaller, the relation between the two elements cannot be reliably established
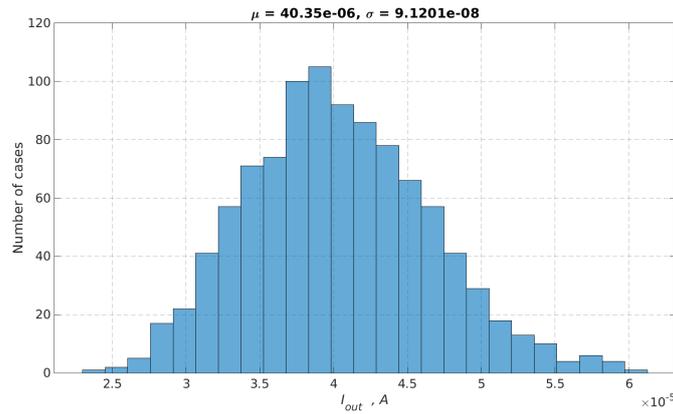
Fig. 7. Deviation of the current source value

Table 3

Simulation results of the calibration (negative threshold)

th_σ, mV	Error/th_σ				Normalized mean max. INL			
	Bubble	**Merge**	**Quick**	**Selection**	**Bubble**	**Merge**	**Quick**	**Selection**
	$N = 6$							
−1.51	−0.889	−2.079	−1.394	−3.003	0.617	1.029	0.725	1.115
−0.30	−0.960	−1.701	−1.587	−2.566	0.556	0.560	0.559	0.572
−0.29	−0.954	−1.692	−1.622	−2.528	0.555	0.559	0.558	0.570
	$N = 8$							
−1.51	−0.893	−2.800	−1.193	−4.267	0.538	1.573	0.621	1.149
−0.30	−0.972	−2.509	−1.765	−5.559	0.456	0.470	0.456	0.702
−0.29	−0.975	−2.487	−1.779	−5.468	0.456	0.469	0.457	0.692
	$N = 10$							
−1.51	−0.875	−3.536	−1.151	−5.362	0.508	3.019	0.929	1.080
−0.30	−0.964	−3.279	−1.642	−12.063	0.383	0.429	0.384	1.579
−0.29	−0.960	−3.271	−1.684	−12.017	0.383	0.424	0.384	1.615

and the further calibration process may be unsuccessful. In particular, the finite threshold leads to wrong decisions during the sorting phase of switching-based calibration. This paper investigates the influence of comparator component mismatch on the results of switching-based calibration.

The simulation results of the comparator designed using 0.18 μm "Mikron" HCMOS8D technology shows that there is an optimal comparator area in terms of threshold variation reduction. Increasing the components area is expected to reduce the standard deviation. Increasing of components by 5 times from 4 μm/2.5 μm to 20 μm/10 μm has corresponding reduction of the threshold standard deviation. However, further increasing the area cannot noticeably (< 3%) improve the comparison accuracy. Therefore, the component sizes of 20 μm/10 μm seem to be optimal values for the comparator.

The obtained comparison accuracy is verified by its influence on the results of the switching-based calibration. For this purpose, the 1F1D calibration algorithms are used as an example. In addition, the comparison accuracy has different effects on different sorting algorithms. The simulation of the calibration process shows that all sorting algorithms are able to establish almost correct order of the

elements. However, for the smallest comparator variant (4 μm/2.5 μm) only quick sort can provide INL improvement for all considered conditions. The medium comparator (20 μm/10 μm) is able to provide a significant INL improvement (up to 61%). The largest comparator (100 μm/50 μm) has almost the same threshold and provides from 0.2% to 3% INL improvement compared to the medium comparator. Therefore, the largest comparator can be discarded as an impractical solution and the optimal comparator area can be used without significant losses.

As for the choice of the sorting algorithm, the quick sort is the most effective and reliable solution. Although bubble and selection sorts can show slightly better results, they are sensitive to the influence of the threshold sign.

REFERENCES

1. **Arbet D., Nagy G., Stopjaková V., Gyepes G.** A self-calibrated binary weighted DAC in 90nm CMOS technology. 2014 29th International Conference on Microelectronics Proceedings − MIEL 2014, 2014, Pp. 383−386. DOI: 10.1109/MIEL.2014.6842170

2. **Radulov G.I., Quinn P.J., Hegt H., van Roermund A.** An on-chip self-calibration method for current mismatch in D/A converters. Proceedings of the 31st European Solid-State Circuits Conference − ESSCIRC 2005, 2005, Pp. 169−172. DOI: 10.1109/ESSCIR.2005.1541586

3. **Stefanou A., Siozios K., Hatzopoulos A.** Design of A 10-Bit, 2GS/s Current-Steering Digital-to-Analog Converter with OnLine Current Calibration. 2022 IEEE International Symposium on Circuits and Systems (ISCAS), 2022, Pp. 1319−1322. DOI: 10.1109/ISCAS48785.2022.9937511

4. **Stoops D.J., Kuo J., Hurst P.J., Levy B.C., Lewis S.H.** Digital Background Calibration of a Split Current-Steering DAC. IEEE Transactions on Circuits and Systems I: Regular Papers, 2019, Vol. 66, No. 8, Pp. 2854−2864. DOI: 10.1109/TCSI.2019.2901626

5. **Kvashina N.V., Yenuchenko M.S.** Comparative Analysis of Switching-Based Calibration Algorithms for DACs. 2022 Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), 2022, Pp. 157−161. DOI: 10.1109/ElConRus54750.2022.9755635

6. **Kvashina N.V., Yenuchenko M.S.** Elements Reordering in Switching-Based Calibration for DACs. 2022 International Conference on Electrical Engineering and Photonics (EExPolytech), 2022, Pp. 13−16. DOI: 10.1109/EExPolytech56308.2022.9950985

7. **Kvashina N.V., Yenuchenko M.S.** Switching Sequence Reordering by Foreground Self-Calibration Circuit in DACs. 2023 Seminar on Networks, Circuits and Systems (NCS), 2023, Pp. 107−110. DOI: 10.1109/NCS60404.2023.10397495

8. **Kvashina N.V., Yenuchenko M.S.** Influence Analysis of Elements Redundancy in Switching-Based DAC Calibration. 2023 International Conference on Electrical Engineering and Photonics (EExPolytech), 2023, Pp. 80−83. DOI: 10.1109/EExPolytech58658.2023.10318566

9. **Pilipko M.M., Morozov D.V., Yenuchenko M.S.** MASH 2-2 Delta-Sigma Modulator with Dynamic Element Matching in 0.18 μm CMOS Technology. Computing, Telecommunications and Control, 2023, Vol. 16, No. 3, Pp. 29−38. DOI: 10.18721/JCSTCS.16303

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Yenuchenko Mikhail S.
Енученко Михаил Сергеевич
E-mail: mixeme@outlook.com
ORCID: https://orcid.org/0000-0002-5301-3871

Kvashina Natalya V.
Квашина Наталья Владимировна
E-mail: kvashina.nv@gmail.com
ORCID: https://orcid.org/0000-0003-2637-0846

Piatak Ivan M.
Пятак Иван Михайлович
E-mail: i.m.piatak@gmail.com

Simulations of Computer, Telecommunications, Control and Social Systems
Моделирование вычислительных, телекоммуникационных, управляющих и социально-экономических систем

APPLICATION OF THE PERTURBATION METHOD TO CONSTRUCT A REFINED COMPACT MODEL OF A THERMOELECTRIC ELEMENT WITH TEMPERATURE-DEPENDENT PARAMETERS

P.P. Udalov[1] ✉ , A.V. Lukin[1], I.A. Popov[1],
V.V. Loboda[1], V.V. Varivcev[2], D.S. Butakov[2]

[1] Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation;

[2] JSC "Institute of Nuclear Materials", Zarechny, Russian Federation

✉ pp_udalov@mail.ru

Abstract. The paper presents an asymptotically substantiated compact model of the Peltier element. The problem of stationary temperature distribution in a one-dimensional thermoelectric medium with temperature-dependent physical parameters is considered. A direct asymptotic approximation is constructed under the assumption that the ratio of the temperature difference at the boundaries of the Peltier element to the mean absolute temperature of the module is a small value. Expressions for heat fluxes on the absorbing and radiating sides with a nonlinear dependence on the applied current and boundary temperatures are obtained. A method of synthesis based on the obtained solution of a compact system model of a thermoelectric module is proposed. A numerical example is used to compare the obtained model with the classical model with averaged material parameters. It is shown that the heat fluxes of the two models take different values at sufficiently large electric currents. Promising areas of using the proposed new analytical model of the Peltier element in industrial problems are discussed.

Keywords: Peltier battery, Matlab, Simscape, direct asymptotic expansion method, system-level modeling, reduced order modelling

ПРИМЕНЕНИЕ МЕТОДА ВОЗМУЩЕНИЙ ДЛЯ ПОСТРОЕНИЯ УТОЧНЕННОЙ КОМПАКТНОЙ МОДЕЛИ ТЕРМОЭЛЕКТРИЧЕСКОГО ЭЛЕМЕНТА С ТЕРМОЗАВИСИМЫМИ ПАРАМЕТРАМИ

П.П. Удалов[1] ✉ *, А.В. Лукин[1], И.А. Попов[1],*
В.В. Лобода[1], А.В. Варивцев[2], Д.С. Бутаков[2]

[1] Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация;
[2] АО «Институт реакторных материалов», Заречный, Российская Федерация

✉ pp_udalov@mail.ru

Аннотация. В работе выполняется построение асимптотически обоснованной компактной модели элемента Пельтье. Рассматривается задача о стационарном распределении температуры в одномерной термоэлектрической среде с физическими параметрами, зависящими от температуры. Построено прямое асимптотическое приближение в предположении о том, что отношение разности температур на границах элемента Пельтье к средней абсолютной температуре модуля является малой величиной. Получены выражения для тепловых потоков на поглощающей и излучающей сторонах с нелинейной зависимостью от приложенного тока и граничных температур. Предложена методика синтеза на основе полученного решения компактной системной модели термоэлектрического модуля. На численном примере проведено сравнение полученной модели с классической моделью с осредненными параметрами материала. Показано, что тепловые потоки двух моделей принимают разные значения при достаточно больших электрических токах. Обсуждаются перспективные направления использования предложенной новой аналитической модели элемента Пельтье в задачах промышленности.

Ключевые слова: батарея Пельтье, Matlab, Simscape, метод прямого асимптотического разложения, системное моделирование, модели пониженного порядка

Introduction

Thermoelectric modules based on the Peltier effect are an important part of modern thermal stabilization and cooling systems [1−11]. Such systems are widely used, for example, to create miniature refrigeration units and sensors that register temperature differences [11].

In practice, constant physical parameters of the object are used for analytical estimations of dependence of the main parameters of thermoelectric modules (cooling capacity, maximum temperature difference between module surfaces, etc.) on the boundary values of temperatures and heat fluxes. The obtained estimates under this assumption do not always coincide with experimental data [1−3, 5, 6, 12, 13]. Taking into account the dependence of material parameters on temperature can strongly change the module performance and its output parameters with respect to similar values calculated with constant material

parameters. For a more accurate description of the battery parameters or design (selection of material properties of p- and n- types) of the battery, a correct mathematical model is required.

A one-dimensional analytical model of the stationary state of a thermoelectric element is investigated in [1, 2]. Temperature distributions along the Peltier element as a function of current density under boundary conditions of the first and second kind are obtained. In [5−9], the methodology of modelling of Peltier elements based on the system approach with application of compact models is proposed. Numerical dependences of temperatures and heat fluxes of the Peltier element are obtained based on the solution of the linear boundary value problem with constant material parameters. The presented approach is useful and practical, since only manufacturers' data sheets are used to obtain the system model parameters. In [10, 11], expressions for the optimal current value, at which the maximum temperature difference occurs in the Peltier element, are given. A nonlinear equation for temperature distribution taking into account temperature-dependent material properties is obtained. The influence of temperature-dependent physical parameters on the temperature distribution along the Peltier element is discussed. It is shown that the heat conduction model with temperature-dependent properties significantly refines the temperature distribution in comparison with the model with averaged material parameters. In [14−19], mathematical models of unsteady pulse impact on a thermoelectric element leading to the so-called supercooling effect were constructed and investigated.

In this paper, we consider the problem of finding an approximate analytical solution to the problem of stationary temperature distribution for a Peltier element, when temperature-dependent physical properties of the material are taken into account. Using the mathematical apparatus of perturbation methods, modified expressions of heat fluxes at the boundaries of the Peltier element are determined. A compact system model of the Peltier element is proposed, taking into account the obtained refined thermoelectric characteristics.

Mathematical model

The stationary thermoelectric state of a one-dimensional medium is described by the equation [1−3]:

$$\kappa(T)\frac{d^2T}{dx^2} + \frac{d\kappa(T)}{dT}\left(\frac{dT}{dx}\right)^2 - j_0 T \frac{dS(T)}{dT}\frac{dT}{dx} = -\frac{j_0^2}{\sigma(T)}; \tag{1a}$$

$$T(x=0) = T_L, \ T(x=L) = T_R, \tag{1b}$$

where T is the temperature; (T), $\sigma(T)$, $S(T)$ are the temperature-dependent thermal conductivity, electrical resistivity and Seebeck coefficients, respectively; j_0 is the current density, x is the spatial coordinate.

A schematic representation of the problem under consideration is presented in Fig. 1.

An example of the dependences of experimental parameters $\kappa(T)$, $\sigma(T)$, $S(T)$ for BiSbTe n- $\left(\kappa_n^T(T), \sigma_n^T(T), S_n^T(T)\right)$ and p- $\left(\kappa_p^T(T), \sigma_p^T(T), S_p^T(T)\right)$ types in the temperature range from 300 to 500 K is given in Fig. 2, 3 [19]. The experimental data were approximated by parabolic polynomials, where the lower index n, p defines the material properties calculated for n- and p-type BiSbTe material.

Expressions for $\kappa_{n,p}^T(T)$, $\sigma_{n,p}^T(T)$, $S_{n,p}^T(T)$ are written as:

$$\kappa_{n,p}^T(T) = \kappa_0^{n,p} + \kappa_1^{n,p}T + \kappa_2^{n,p}T^2;$$

$$\sigma_{n,p}^T(T) = \sigma_0^{n,p} + \sigma_1^{n,p}T + \sigma_2^{n,p}T^2; \tag{2}$$

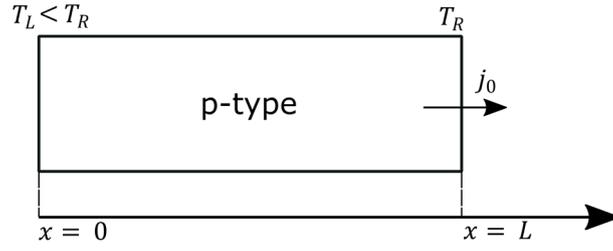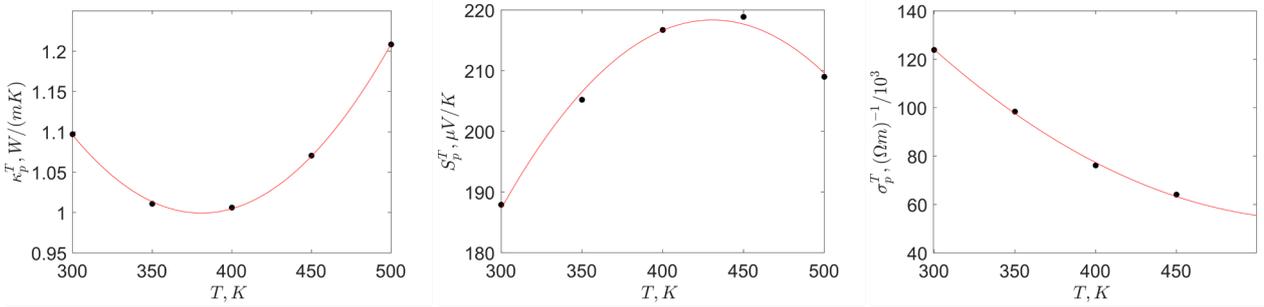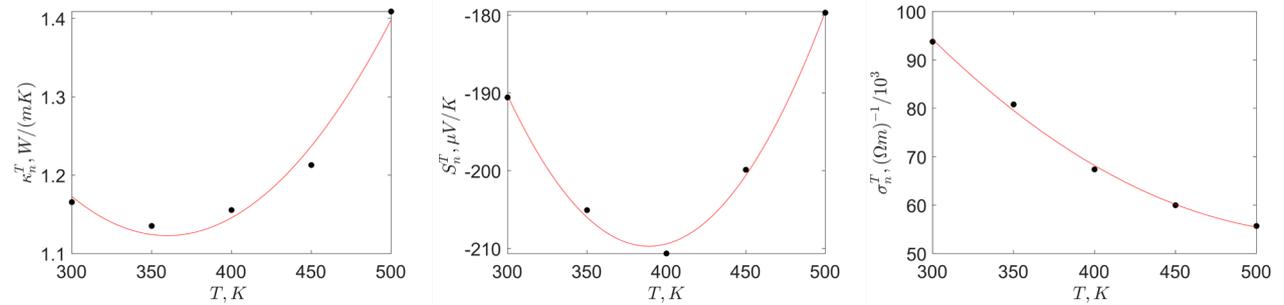$$S_{n,p}^T(T) = S_0^{n,p} + S_1^{n,p}T + S_2^{n,p}T^2.$$

Fig. 1. One-dimensional model of a p-type element (constant cross-section A)

Fig. 2. Polynomial approximation (solid line) of experimental data
for n-type $\kappa_p^T(T)$, $\sigma_p^T(T)$, $S_p^T(T)$ (points) of BiSbTe material

Fig. 3. Polynomial approximation (solid line) of experimental data
for n-type $\kappa_n^T(T)$, $\sigma_n^T(T)$, $S_n^T(T)$ (points) BiSbTe material

Based on the representation of physical parameters by expressions (2), equation (1) is characterized by a degree nonlinearity with respect to the variable T. The analytical representation of the material parameters $\kappa_{n,p}^T$, $\sigma_{n,p}^T$, $S_{n,p}^T$ (2) allows us to apply asymptotic methods [20, 21] to find an approximate solution for the temperature distribution (1a).

Let us introduce dimensionless quantities:

$$u = \frac{T - \hat{T}}{\hat{T}}, \ \xi = \frac{x}{L}, \ \hat{T} = \frac{T_L + T_R}{2}. \tag{3}$$

Then equation (1) is rewritten in the form:

143

$$\kappa(u)\frac{d^2u}{d\xi^2}+\frac{d\kappa(u)}{du}\left(\frac{du}{d\xi}\right)^2-j_0L(u+1)\frac{dS(u)}{du}\frac{du}{d\xi}=-\frac{(j_0L)^2}{\hat{T}}\frac{1}{\sigma(u)'}; \tag{4a}$$

$$u(\xi=0)=\frac{T_L-\hat{T}}{\hat{T}},\ u(\xi=1)=\frac{T_R-\hat{T}}{\hat{T}}, \tag{4b}$$

where u is the small and finite relative temperature change; $\kappa(u)$, $\sigma(u)$, $S(u)$ (lower indices n, p omitted hereafter) are defined as:

$$\kappa(u)=\hat{\kappa}_0+\hat{\kappa}_1u+\hat{\kappa}_2u^2,\ \sigma(u)=\hat{\sigma}_0+\hat{\sigma}_1u+\hat{\sigma}_2u^2,\ S(u)=\hat{S}_0+\hat{S}_1u+\hat{S}_2u^2, \tag{5}$$

and the coefficients $\hat{\kappa}_0$, $\hat{\kappa}_1$, $\hat{\kappa}_2$, $\hat{\sigma}_0$, $\hat{\sigma}_1$, $\hat{\sigma}_2$, $\hat{S}_0$, $\hat{S}_1$, $\hat{S}_2$ take the form of:

$$\hat{\kappa}_0=\kappa_0+\kappa_1\hat{T}+\kappa_2\hat{T}^2,\ \hat{\kappa}_1=\kappa_1\hat{T}+2\kappa_2\hat{T}^2,\ \hat{\kappa}_2=\kappa_2\hat{T}^2;$$

$$\hat{\sigma}_0=\sigma_0+\sigma_1\hat{T}+\sigma_2\hat{T}^2,\ \hat{\sigma}_1=\sigma_1\hat{T}+2\sigma_2\hat{T}^2,\ \hat{\sigma}_2=\sigma_2\hat{T}^2; \tag{6}$$

$$\hat{S}_0=S_0+S_1\hat{T}+S_2\hat{T}^2,\ \hat{S}_1=S_1\hat{T}+2S_2\hat{T}^2,\ \hat{S}_2=S_2\hat{T}^2.$$

In practice, u is often a small value [10, 11, 20], the temperature difference across the Peltier element is of the order of a few tens of degrees, while the average temperature is of the order of hundreds of degrees Kelvin. Problem (4), together with equations (5), forms a second order nonlinear differential equation, which has no analytical solution. To evaluate the solution of this problem, it is assumed that u is a small value, and the asymptotic method of direct expansion is applied [20−22].

Asymptotic solution of the one-dimensional thermoelectric problem

To find a uniformly suitable solution of the third order of smallness, we decompose u into the following series:

$$u(\xi,\varepsilon)=\varepsilon u_0(\xi)+\varepsilon^2u_1(\xi)+\varepsilon^2u_2(\xi)+\ \ldots, \tag{7}$$

where ε is a small parameter.

Let us consider the term in the right part of the equation (4a), for this purpose we divide equation (4) by the value κ_0, then:

$$\frac{(j_0L)^2}{\hat{T}}\frac{1}{\kappa_0\sigma}=\frac{I^2R\Theta}{\hat{T}}=\frac{dT}{\hat{T}}, \tag{8}$$

where $I=j_0A$ is an electric current through the element, $R=\frac{1}{\sigma}\frac{L}{A}$ is an electric resistance, $\Theta=\frac{1}{\kappa_0}\frac{L}{A}$ is a thermal resistance of the element, dT is a temperature difference in the element at the heat flux equal to the electric power I^2R.

From a practical point of view, it makes sense to apply a thermoelectric element only in the region, where the heat flux from the Joule−Lenz effect is significantly smaller in comparison with the basic Fourier heat flux. In this regard, we will assume that the summand in the right-hand side can be attributed to quantities of the order ε. Let us take into account the smallness of the heat fluxes caused by the

Thomson effect for the class of materials under consideration $\left(\dfrac{j_0 L \hat{S}_1}{\hat{\kappa}_0} \sim 10^{-1} \right)$. We will assume that the summand with the Seebeck coefficient of equation (4a) does not enter the generating problem, that is,

$$j_0 L \frac{dS}{du} \sim \varepsilon.$$

Substituting the equations of expansion (7) into equation (4) and equating the coefficients at the same powers of ε, we obtain:

ε

$$u_0'' = -\frac{j_0^2 L^2}{\hat{T}\hat{\kappa}_0 \hat{\sigma}_0}, \tag{9a}$$

$$u_0(\xi = 0) = \frac{T_L - \hat{T}}{\hat{T}}, \ u_0(\xi = 1) = \frac{T_R - \hat{T}}{\hat{T}}; \tag{9b}$$

ε^2

$$u_1'' = \frac{j_0 L S_1}{\hat{\kappa}_0} u_0' - \left(\frac{\hat{\kappa}_1}{\hat{\kappa}_0} + \frac{\hat{\sigma}_1}{\hat{\sigma}_0} \right) u_0 u_0'' - \frac{\hat{\kappa}_1}{\hat{\kappa}_0} u_0'^2, \tag{10a}$$

$$u_1(\xi = 0) = 0, \ u_1(\xi = 1) = 0; \tag{10b}$$

ε^3

$$
\begin{aligned}
u_2'' = {} & \frac{j_0 L S_1}{\hat{\kappa}_0} u_1' - \left(\frac{\hat{\kappa}_1}{\hat{\kappa}_0} + \frac{\hat{\sigma}_1}{\hat{\sigma}_0} \right)(u_0 u_1'' + u_1 u_0'') - \\
& - \left(\frac{\hat{\kappa}_2}{\hat{\kappa}_0} + \frac{\hat{\sigma}_2}{\hat{\sigma}_0} + \frac{\hat{\kappa}_1}{\hat{\kappa}_0}\frac{\hat{\sigma}_1}{\hat{\sigma}_0} \right) u_0^2 u_0'' - \left(\frac{2\hat{\kappa}_2}{\hat{\kappa}_0} + \frac{\hat{\kappa}_1}{\hat{\kappa}_0}\frac{\hat{\sigma}_1}{\hat{\sigma}_0} \right) u_0 u_0'^2 - \\
& - \frac{2\hat{\kappa}_1}{\hat{\kappa}_0} u_0' u_1' + \frac{j_0 L}{\hat{\kappa}_0}\left(\left(1 + \frac{\hat{\sigma}_1}{\hat{\sigma}_0} \right) S_1 + 2 S_2 \right) u_0 u_0',
\end{aligned}
\tag{11a}
$$

$$u_2(\xi = 0) = 0, \ u_2(\xi = 1) = 0. \tag{11b}$$

The first approximation problem given by equations (9) is similar to the thermoelectric problem with constant (averaged) material properties:

$$u_0 = -\frac{c_0}{2}\xi^2 + c_1 \xi + c_2, \tag{12}$$

where

$$c_0 = \frac{j_0^2}{\hat{\kappa}_0 \hat{\sigma}_0}\frac{L^2}{\hat{T}}, \ c_1 = \frac{c_0}{2} + \frac{T_R - T_L}{\hat{T}}, \ c_2 = \frac{T_L - \hat{T}}{\hat{T}}. \tag{13}$$

By substituting the solution (11) into the boundary value problem (9), it is possible to obtain the solution for u_1. Any further refinements of the solution become more complex in structure, however, their explicit form can be obtained using computer algebra methods [23] and then transferred to the simulation environment for direct calculations or optimization procedures. For typical parameter values, the constructed solutions converge to a direct numerical solution quickly: a two- or three-term approximation can satisfy the practical needs. In the numerical example (see Section 4), the maximum relative error with respect to the direct numerical solution of the nonlinear boundary value problem realized using the Matlab bvp4c built-in function [23] is 0.69% and 0.042% for the two- and three-term solution, respectively. In the following calculations, the trinomial solution will be used.

Substituting (13) into equations (10, 11), we obtain expressions for temperatures u_1 and u_2. Due to the adopted system of notations, $\varepsilon = 1$ [21]. In this case, the equation for the dimensionless temperature u is written in the form of:

$$u = d_1\xi^8 + d_2\xi^7 + d_3\xi^6 + d_4\xi^5 + d_5\xi^4 + d_6\xi^3 + d_7\xi^2 + d_8\xi + d_9, \tag{14}$$

where the constants d_i, $i = \overline{1,9}$ are determined from equations (9–11).

The heat flux profile $q = q(u,\xi)$ can be calculated as [1, 2]:

$$q(u,\xi) = -\frac{\hat{T}}{L}\kappa(u)u' + j_0\hat{T}S(u)(u+1). \tag{15}$$

Consider the values of heat flux q on the left q_L and right q_R sides of the Peltier element.

From equation (15), taking into account (3), (13), the heat fluxes q_L, q_R have the following form:

$$q_L = q\left(\frac{T_L - \hat{T}}{\hat{T}}, 0\right) = \frac{\hat{T}}{L}\kappa_L u'(0) + j_0 S_L T_L,$$

$$q_R = q\left(\frac{T_R - \hat{T}}{\hat{T}}, 1\right) = \frac{\hat{T}}{L}\kappa_R u'(1) + j_0 S_R T_R, \tag{16}$$

or:

$$q_L = -\frac{d_8\kappa_L}{2L}(T_R + T_L) + j_0 S_L T_L,$$

$$q_R = -\frac{(8d_1 + 7d_2 + 6d_3 + 5d_4 + 3d_6 + 2d_7 + d_8)\kappa_R}{2L}(T_R + T_L) + j_0 S_R T_R, \tag{17}$$

where κ_L, κ_R, S_R are κ and S calculated on the left and right sides of the Peltier element.

Equations (17) are modified expressions for heat fluxes on the radiating and absorbing sides of the Peltier element in the case of temperature-dependent material parameters. Further on the basis of expressions (15, 17) the heat fluxes on both sides of the Peltier element are estimated.

Numerical examples

The following examples show the results of calculations of temperature and heat fluxes of the thermoelectric element. The material properties taken in the calculations are given in Table 2 for p-type material. The dependence of material properties on temperature is assumed according to Fig. 2 [1, 2]. For n-type material, the constructions are similar and are not given.

Table 1

Values of thermoelectric cell parameters for BiSbTe n-type material

Variable	Value
T_L	300 K
T_R	320 K
j_0	0.83 MA/m^2
A	38.6 mm^2
L	5 mm
$\hat{\kappa}_0$	1.22 W/(m·K)
$\hat{\kappa}_1$	−0.65 W/(m·K^2)
$\hat{\kappa}_2$	1.1 W/(m·K^3)
$\hat{\sigma}_0$	1.1×10^4 (Ω·m)$^{-1}$
$\hat{\sigma}_1$	−9.93×10^4 (Ω·m)$^{-1}$/K
$\hat{\sigma}_2$	5.05×10^4 (Ω·m)$^{-1}$/K^2
$\hat{S}_0$	−1.8×10^{-4} V/K
$\hat{S}_1$	−1.6×10^{-4} V/K^2
$\hat{S}_2$	1.85×10^{-4} V/K^3
$\langle\kappa\rangle$	1.198 W/(m·K)
$\langle\sigma\rangle$	70×10^3 (Ω·m)$^{-1}$
$\langle S\rangle$	−2×10^{-4} V/K

Table 2

Values of thermoelectric cell parameters for BiSbTe p-type material

Variable	Value
$\hat{\kappa}_0$	1.17 W/(m·K)
$\hat{\kappa}_1$	−0.86 W/(m·K^2)
$\hat{\kappa}_2$	1.12 W/(m·K^3)
$\hat{\sigma}_0$	13.96×10^4 (Ω·m)$^{-1}$
$\hat{\sigma}_1$	−17.97×10^4 (Ω·m)$^{-1}$/K
$\hat{\sigma}_2$	9.4×10^4 (Ω·m)$^{-1}$/K^2
$\hat{S}_0$	1.74×10^{-4} V/K
$\hat{S}_1$	1.56×10^{-4} V/K^2
$\hat{S}_2$	−1.38×10^{-4} V/K^3
$\langle\kappa\rangle$	1.06 W/(m·K)
$\langle\sigma\rangle$	82×10^3 (Ω·m)$^{-1}$
$\langle S\rangle$	2.1×10^{-4} V/K

where $\langle\kappa\rangle$, $\langle\sigma\rangle$, $\langle S\rangle$ are averaged material parameters defined as
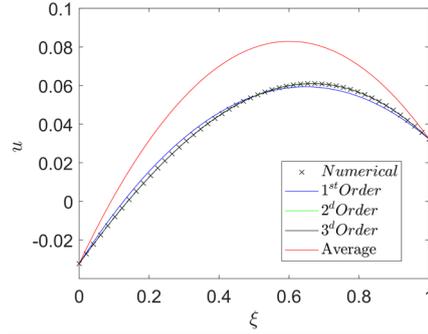
Fig. 4. Temperature profile distributions in the cases of numerical solution (crosses),
results obtained with the approximated solution (blue, green and black solid lines)
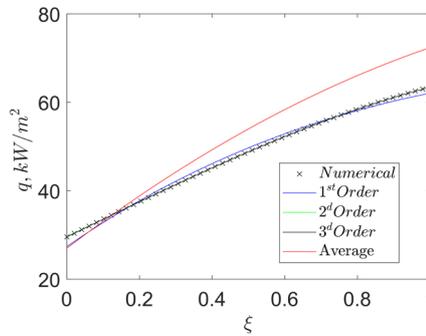and the solution with averaged material parameters (red solid line)

Fig. 5. Distributions of heat flux profiles in cases of numerical solution (crosses),
results obtained with the approximated solution (blue, green and black solid lines)
and the solution with averaged material parameters (red solid line)

$$< \kappa >= \frac{\int_{T_{\min}}^{T_{\max}} \kappa(T)\,dT}{T_{\max} - T_{\min}}, \quad < \sigma >= \frac{\int_{T_{\min}}^{T_{\max}} \sigma(T)\,dT}{T_{\max} - T_{\min}}, \quad < S >= \frac{\int_{T_{\min}}^{T_{\max}} S(T)\,dT}{T_{\max} - T_{\min}}, \tag{18}$$

$T_{\max}$ = 500 K, $T_{\min}$ = 300 K are maximal and minimal temperatures (see Fig. 2, 3).

We compare the temperature and heat flux distributions in the Peltier element in the case of p-type BiSbTe material between the numerical solution of equation (4) in Matlab (using bvp4c function), the asymptotic solution (9−14) and the solution of equation (4) with averaged material parameters (18) (see Fig. 4, 5).

Fig. 4, 5 show that in the case of temperature-dependent material parameters, the direct expansion method gives more accurate temperature and heat flux profiles compared to the case of averaged material parameters [1, 2]. The first asymptotic term gives better results than the solution with parameters averaged over the total temperature interval of the input data. This is explained by the fact that the physical parameters of the Peltier element are a series that decompose with respect to the average temperature of the Peltier element $\hat{T}$.

Compact Peltier battery model

To simulate the behavior of the Peltier element, a system-level model has been developed to simulate the behavior of the Peltier battery and predict the amount of heat it can transfer. The principle of
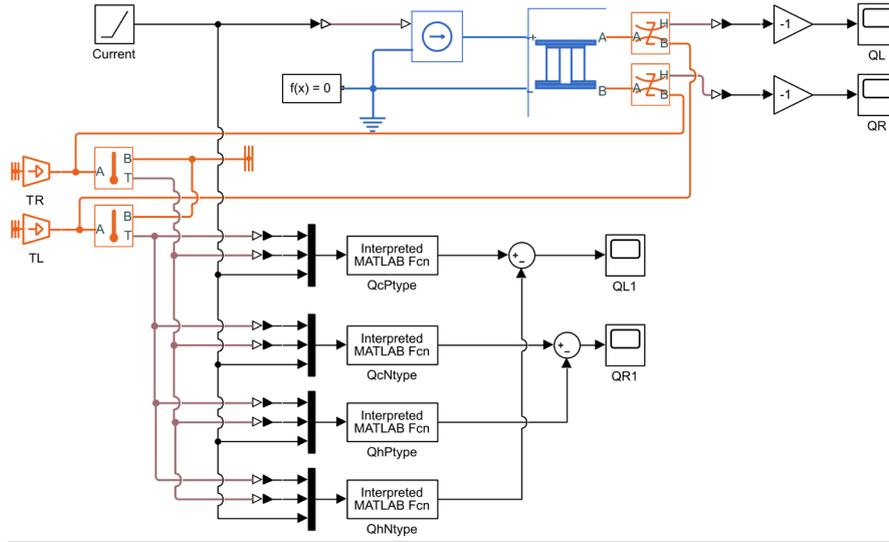
Fig. 6. Models of Peltier batteries

these models is to isolate the thermal and electrical parts of the system and to model them based on the thermal and electrical circuits with concentrated parameters [5, 6, 24]. Many works (see, for example, [5, 6]) are devoted to the study of the case of averaged material parameters, which do not depend on temperature and simplify the construction of equivalent circuits. Further, we propose to modify the scheme [5] shown in Fig. 6 by adding temperature-dependent material properties $\kappa = \kappa\left(u, \hat{T}\right)$, $S = S\left(u, \hat{T}\right)$, $\sigma = = \sigma\left(u, \hat{T}\right)$, which are obtained after solving problems (4, 5).

The heat fluxes in the case of average material parameters Q_L and Q_R on the cold and hot sides of the Peltier element can be expressed as [10, 11]:

$$Q_L = \frac{\left(T_L - T_R\right)}{\Theta} + SIT_L - \frac{1}{2}I^2 R,$$

$$Q_R = \frac{\left(T_L - T_R\right)}{\Theta} + SIT_R + \frac{1}{2}I^2 R \tag{19}$$

and the voltage drop across the thermoelectric element is written as:

$$V = S\left(T_R - T_L\right) + IR, \tag{20}$$

where R is the electrical resistance, Θ is the thermal resistance, I is the electric current.

The averaged material parameters R, Θ, S in equations (19) are written in terms of n- and p-type material properties in the form [4]:

$$R = \left(\frac{1}{<\sigma_n>} + \frac{1}{<\sigma_p>}\right)\frac{L}{A}, \quad S = \left|< S_n >\right| + \left|< S_p >\right|,$$

$$\Theta = \left(\frac{1}{<\kappa_n>} + \frac{1}{<\kappa_p>}\right)\frac{L}{A}. \tag{21}$$
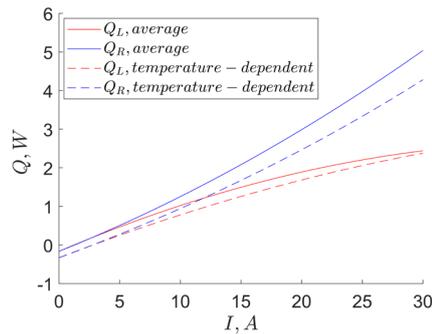
149

Fig. 7. Distribution of heat fluxes as a function of current strength
in the cases of asymptotic solution (dashed lines) and solution with averaged material parameters (solid line)

The Matlab/Simulink software package is used to build compact thermoelectric models of the Peltier element. In accordance with the above, Fig. 7 shows the diagrams of Peltier battery described by equations (16, 19) with material parameters given in Tables 1, 2. The upper part of the diagram shows the Peltier battery model with averaged material properties (21), which is specified by the "Peltier Device" block, and the lower part of the diagram shows the case of temperature-dependent material of the Peltier battery model (19). The defining equations of the thermal state of the "Peltier Device" block are similar to the expressions (19, 20). The lower blocks "QcPtype", "QcNtype", "QhPtype", "QcNtype" are a compact record of the heat flux expressions given by expressions (17).

Fig. 7 shows the dependences of heat fluxes Q_R, Q_L according to the defining equations (16, 19).

As can be seen from Fig. 7, the model of the Peltier element calculated with temperature-dependent parameters gives different values of heat fluxes in comparison with the model with averaged material parameters. This effect is because in the case of initial averaging of material parameters, the calculation of the thermal state is carried out at the temperature corresponding to the averaged values of material parameters. In the asymptotic solution, the system parameters depend on both the boundary conditions and the applied current, which leads to differences in the results.

Conclusion

In the present work, a model of reduced dimensionality for a Peltier element is constructed using the asymptotic direct expansion method. It is shown that taking into account temperature-dependent material parameters can significantly affect the boundary heat fluxes and integral characteristics of the thermoelectric device. The implementation of the proposed Peltier battery circuit in the Matlab/Simscape software package is demonstrated. The model can be useful for the analysis of quasi-stationary thermoelectric processes, when the total heat capacity of the Peltier battery is negligibly small in relation to the heat capacity of the thermostabilized object. The subject of further research may be the extension of the proposed approach to the modelling of fast non-stationary thermoelectric processes, in particular, to the modelling of the so-called "supercooling" phenomenon [11].

REFERENCES

1. **Seifert W., Ueltzen M., Strumpel C., Heiliger W., Müller E.** One-dimensional modeling of a Peltier element. Proceedings ICT2001. 20th International Conference on Thermoelectrics (Cat. No.01TH8589), 2001, Pp. 439—443. DOI: 10.1109/ICT.2001.979925

2. **Seifert W., Ueltzen M., Müller E.** One-dimensional modelling of thermoelectric cooling. Physica Status Solidi (a), 2002, Vol. 194, no. 1, pp. 277—290. DOI: 10.1002/1521-396X(200211)194:1<277::AID-PSSA277>3.0.CO;2-5

3. **Landau L.D., Lifshitz E.M.** Electrodynamics of Continuous Media. Vol. 8 (1st ed.). Oxford: Pergamon Press, 1960.

4. **Ioffe A.F.** Physics of Semiconductors. New York: Academic, 1960.

5. **Lineykin S.B., Ben-Yaakov S.** PSPICE-compatible equivalent circuit of thermoelectric cooler. 2005 IEEE 36th Power Electronics Specialists Conference, 2005, Pp. 608−612. DOI: 10.1109/PESC.2005.1581688

6. **Zavorotneva E.V., Indeitsev D.A., Lukin A.V., Popov I.A., Udalov P.P.** Technique for compact modeling of thermoelectric systems. Computing, Telecommunications and Control, 2021, Vol. 14, No. 2, Pp. 29−48. DOI: 10.18721/JCST-CS.14203

7. **Chavez J.A., Ortega J.A., Salazar J., Turo A.A., Garcia M.J.** SPICE model of thermoelectric elements including thermal effects. Proceedings of the 17th IEEE Instrumentation and Measurement Technology Conference, 2000, Pp. 1019−1023. DOI: 10.1109/IMTC.2000.848895

8. **Bagnoli P.E., Casarosa C., Ciampi M., Dallago E.** Thermal resistance analysis by induced transient (TRAIT) method for power electronic devices thermal characterization. I. Fundamentals and theory. IEEE Transactions on Power Electronics, 1998, Vol. 13, No. 6, Pp. 1208−1219. DOI: 10.1109/63.728348

9. **Moumouni Y., Baker R.J.** Improved SPICE modeling and analysis of a thermoelectric module. 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS), 2015, Pp. 1−4. DOI: 10.1109/MWSCAS.2015.7282015

10. **Kaganov M.A., Privin M.R.** Termoelektricheskie teplovye nasosy [Thermoelectric heat pumps]. Leningrad: Energiya, 1970. (In Russ.)

11. **Iordanishvili E.K., Babin V.P.** Nestatsionarnye protsessy v termoelektricheskikh i termomagnitnykh sistemakh preobrazovaniia energii [Nonsteady-state processes in thermoelectric and thermomagnetic energy conversion systems]. Moscow: Nauka, 1983. (In Russ.)

12. **Ma M., Yu J.** A numerical study on the temperature overshoot characteristic of a realistic thermoelectric module under a current pulse operation. International Journal of Heat and Mass Transfer, 2014, Vol. 72, Pp. 234−241. DOI: 10.1016/j.ijheatmasstransfer.2014.01.017

13. **Yuan C.D., Jadhav O.S., Rudnyi E.B., Hohlfeld D., Bechtold T.** Parametric model order reduction of a thermoelectric generator for electrically active implants. 2018 19th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE), 2018, Pp. 1−6. DOI: 10.1109/EuroSimE.2018.8369946

14. **Ma M., Yu J., Chen J.** An investigation on thermoelectric coolers operated with continuous current pulses. Energy Conversion and Management, 2015, Vol. 98, Pp. 275−281. DOI: 10.1016/j.enconman.2015.03.105

15. **Lv H., Wang X.-D., Wang T.-H., Meng J.-H.** Optimal pulse current shape for transient supercooling of thermoelectric cooler. Energy, 2015, Vol. 84, Pp. 788−796. DOI: 10.1016/j.energy.2015.02.092

16. **Piggott A.J., Allen J.S.** Peltier supercooling with isosceles current pulses: Cooling an object with internal heat generation. ECS Journal of Solid State Science and Technology, 2017, Vol. 80, No. 5, Pp. 3. DOI: 10.1149/2.0391712jss

17. **Moreno-Navarro P., Pérez-Aparicio J.L., Gómez-Hernández J.J.** Optimization of pulsed thermoelectric materials using simulated annealing and non-linear finite elements. Applied Thermal Engineering, 2017, Vol. 120, Pp. 603−613. DOI: 10.1016/j.applthermaleng.2017.04.036

18. **Snyder G.J., Fleurial J.P., Caillat T., Yang R., Chen G.** Supercooling of Peltier cooler using a current pulse. Journal of Applied Physics, 2002, Vol. 92, No. 3, Pp.1564−1589. DOI: 10.1063/1.1489713

19. **Lee H.S.** Thermoelectrics: Design and materials, New York: John Wiley & Sons, 2016. DOI: 10.1002/9781118848944

20. **Zino I.E., Tropp E.A.** Asimptoticheskie metody v zadachakh teorii teploprovodnosti i termouprugosti [Asymptotic Methods in Problems of the Theory of heat Conduction and Thermoelasticity], Leningrad: Izdatel'stvo LGU, 1970. (In Russ.)

21. **Nayfeh A.H.** Perturbation Methods, New York: John Wiley & Sons, 2008.

22. **Nayfeh A.H., Balakumar B.** Applied nonlinear dynamics: analytical, computational, and experimental methods, New York: John Wiley & Sons, 2008. DOI: 10.1002/9783527617548

23. **Etter D.M., Kuncicky D.C., Hull D.W.** Introduction to MATLAB 6, New Jersey: Prentice Hall Hoboken, 2002.

24. **Kubov V.I., Dymytrov Y.Y., Kubova R.M.** LTspice-model of thermoelectric Peltier-Seebeck element. 2016 IEEE 36th International Conference on Electronics and Nanotechnology (ELNANO), 2016, Pp. 47−51. DOI: 10.1109/ELNANO.2016.7493007

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Udalov Pavel P.
Удалов Павел Павлович
E-mail: pp_udalov@mail.ru

Lukin Alexei V.
Лукин Алексей Вячеславович
E-mail: lukin_av@spbstu.ru

Popov Ivan A.
Попов Иван Алексеевич
E-mail: popov_ia@spbstu.ru

Loboda Vera V.
Лобода Вера Владимировна
E-mail: vera_loboda@spbstu.ru

Varivcev Artem V.
Варивцев Артем Владимирович
E-mail: vav3@niiar.ru

Butakov Denis S.
Бутаков Денис Сергеевич
E-mail: irm@irmatom.ru