https://doi.org/10.31891/csit-2024-1-8
UDC 004.942:669.337.111

Kyrylo KRASNIKOV
Dniprovksyi state technical university

ACCELERATING ORE SINTERING MATHEMATICAL MODEL USING GPU

The study aims to enhance the efficiency and computational speed of the ore sintering model through the utilization of graphics processing units (GPUs). The purpose of this research is to address the growing demand for faster and more scalable simulations in the field of ore sintering, a crucial process in the production of iron and steel. Methodology involves the integration of parallel computing capabilities offered by GPUs into the existing ore sintering model. By leveraging the parallel processing power of GPUs, the computational workload is distributed across multiple cores, significantly reducing the simulation time. Results demonstrate a substantial acceleration in the ore sintering simulation process. Comparative analyses between CPU and GPU implementations reveal a remarkable reduction in computation time, thereby enabling real-time or near-real-time simulations. Comparing to the multicore CPU, the integrated GPU achieves 5,9x speedup at the lowest spatial discretization and 3,6x speedup at the largest spatial discretization. The achieved speedup not only enhances the efficiency of ore sintering modeling but also opens avenues for exploring larger and more complex scenarios. This is the successful integration of GPU parallel computing into the ore sintering model, showcasing the adaptability of advanced computational technologies to traditional industrial processes. The study contributes to the field by bridging the gap between computational power and metallurgical simulations, demonstrating the potential for GPU acceleration in other areas of metallurgical processes. Practical significance of this research is underscored by its potential to revolutionize the ore sintering industry. The implemented automated info-research system with database and user interface accelerates routine tasks and provides scalability of research process. Faster simulations facilitate quicker decision-making in process optimization, leading to improved energy efficiency and reduced environmental impact. This research sets the stage for the broader adoption of GPU acceleration in metallurgical modeling, signaling a paradigm shift towards more efficient and sustainable industrial practices.

Keywords: research software, gpu acceleration, numerical modeling, ore sintering.

Кирило КРАСНІКОВ
Дніпровський державний технічний університет

ПРИСКОРЕННЯ МАТЕМАТИЧНОЇ МОДЕЛІ СПІКАННЯ РУДИ З ВИКОРИСТАННЯМ GPU

Дослідження спрямоване на підвищення ефективності та швидкості обчислень моделі спікання руди шляхом використання графічних процесорів (GPU). Метою цього дослідження є задоволення зростаючого попиту на більш швидкі та масштабовані симуляції в галузі спікання руди, що є ключовим процесом у виробництві заліза та сталі. Методологія передбачає інтеграцію можливостей паралельних обчислень, пропонованих графічними процесорами, в існуючу модель спікання руди. Завдяки використанню паралельної обчислювальної потужності графічних процесорів, обчислювальне навантаження розподіляється між декількома ядрами, що значно скорочує час моделювання. Результати демонструють значне прискорення процесу симуляції спікання руди. Порівняльний аналіз реалізацій на CPU і GPU показує значне скорочення часу обчислень, що дозволяє проводити симуляції в реальному часі або близькому до нього. У порівнянні з багатоядерним процесором, інтегрований графічний процесор досягає 5,9-кратного прискорення при найменшій дискретизації простору і 3,6-кратного прискорення при найбільшій дискретизації. Досягнуте прискорення не тільки підвищує ефективність моделювання спікання руди, але й відкриває можливості для дослідження більших і складніших сценаріїв. Успішна інтеграція паралельних обчислень на GPU в модель спікання руди демонструє адаптивність передових обчислювальних технологій до традиційних промислових процесів. Дослідження робить внесок у галузь, долаючи розрив між обчислювальною потужністю та металургійним моделюванням, демонструючи потенціал прискорення на GPU в інших сферах металургійних процесів. Практичне значення цього дослідження підкреслюється його потенціалом революціонізувати агломераційну галузь. Впроваджена автоматизована інформаційно-дослідницька система з базою даних та інтерфейсом користувача прискорює виконання рутинних завдань та забезпечує масштабованість дослідницького процесу. Швидше моделювання сприяє швидшому прийняттю рішень щодо оптимізації процесів, що призводить до підвищення енергоефективності та зменшення впливу на навколишнє середовище. Це дослідження закладає основу для більш широкого впровадження прискорення на GPU в металургійному моделюванні, сигналізуючи про зміну парадигми в бік більш ефективних і стійких промислових практик.

Ключові слова: дослідницьке програмне забезпечення, графічне прискорення, чисельне моделювання, спікання руди.

Introduction

The ore sintering process is a fundamental stage in the production of ferrous metals, playing a pivotal role in transforming raw ore into a feed material suitable for blast furnaces. Despite its significance, the computational demands associated with simulating and optimizing ore sintering have posed formidable challenges for the metallurgical industry. Traditional Central Processing Unit (CPU)-based models often struggle to deliver timely results, limiting the industry's ability to make informed decisions in real-time.

This research seeks to address this critical bottleneck by introducing a paradigm shift in the computational approach to ore sintering modeling. Our focus is on harnessing the parallel processing power of Graphics Processing Units (GPUs) to accelerate the mathematical model governing ore sintering simulations. The primary objective is to

significantly reduce the computational time required for these simulations, thereby enabling faster and more dynamic decision-making in the realm of metallurgical operations.

Related works

The field of ore sintering modeling and simulation has witnessed significant attention from researchers and practitioners aiming to improve the efficiency and sustainability of metallurgical processes [1]. In this section, we review key contributions and relevant literature, focusing on both traditional CPU-based approaches and emerging trends in GPU acceleration.

Traditional ore sintering models have been rooted in finite difference and finite element methods, providing accurate representations of the complex thermal and chemical transformations during the sintering process [2]. While these models offer valuable insights, their computational demands often lead to extended simulation times, limiting their practical utility for real-time decision-making in industrial settings.

The application of parallel computing techniques, particularly on GPUs, has emerged as a promising avenue to address these computational challenges. The work by Li et al. [3] explored the parallelization of a sintering model using GPU architecture, reporting notable improvements in simulation speed. The study primarily focused on general aspects of the scalability potential of GPU and CPU acceleration.

In [4] authors consider sintering ore bed as continuum and divided it by four zones according to heating and cooling processes. It is presented carbon dioxide content, temperature distribution, water concentration depending on the depth.

In a similar vein, recent advancements in parallel computing have been leveraged in the broader context of materials science simulations. Zhang et al. [5] demonstrated the effectiveness of GPU acceleration in accelerating molecular dynamics simulations of material behavior, showcasing the potential for parallelization in computationally intensive tasks.

Matsuda consider particles interaction, especially stress, in sintering process [6]. It improves simulation realism in the case of sinter layer compressing due melting of its components. In [7] the authors implemented parallelization of learning algorithm for CPU and GPU using CUDA. They compared learning speed on different configurations and showed significant speedup with GPU configuration. The successful application of GPU computing was later used in paper [8], in which author presented bagging parallelization with Python and LightGBM framework. The proposed algorithm demonstrated a good parallelization potential without loss of accuracy.

There is research of computation efficiency of different GPUs [9]. The author achieved speedup (almost 2x) when switched from GeForce 1650 GPU to Tesla T4 GPU. The increment in cores count corresponds to the speedup.

The present paper builds upon these foundations by offering a comprehensive exploration of GPU acceleration in the context of ore sintering. Our methodology encompasses the development and optimization of a mathematical model tailored for efficient parallel execution on GPUs. By addressing the limitations identified in previous studies, we aim to provide a holistic and scalable solution for the metallurgical industry.

While the integration of GPU acceleration in materials science and metallurgy is an evolving field, our research stands out in its specific focus on the ore sintering process. The parallels drawn from related works underscore the significance of computational acceleration in addressing industry challenges, positioning our study within the broader landscape of advancements in parallel computing for materials science and metallurgical applications.

Methodology

The methodology revolves around the development, optimization, and implementation of a mathematical model for ore sintering, specifically designed for efficient parallelization on GPUs. The primary objective is to harness the parallel computing capabilities of GPUs to accelerate the computational aspects of ore sintering simulations, ultimately reducing the time required for accurate predictions.

The mathematical model developed for ore sintering is designed to capture the intricate thermo-chemical processes occurring within the sintering bed. The ore sintering bed is modeled in three dimensions to capture the spatial distribution of temperature, chemical reactions, and material properties. The geometry is represented as a rectangular domain, with dimensions corresponding to the length, width, and height of the sintering bed (Fig. 1). The sintering process evolves over time, allowing for the dynamic simulation of the entire bed.

The assumptions are following:

1) The sintering bed is considered homogeneous and isotropic in its initial state. This assumption facilitates the modeling process by reducing the complexity associated with spatially varying material characteristics.

2) The sinter layer is assumed to have continuous and uniform properties.

3) The air flow through the bed is constant and vertical.

4) The chemical reactions within the bed are assumed to follow well-defined kinetics.

5) Radiation effects are considered in the energy balance equations.

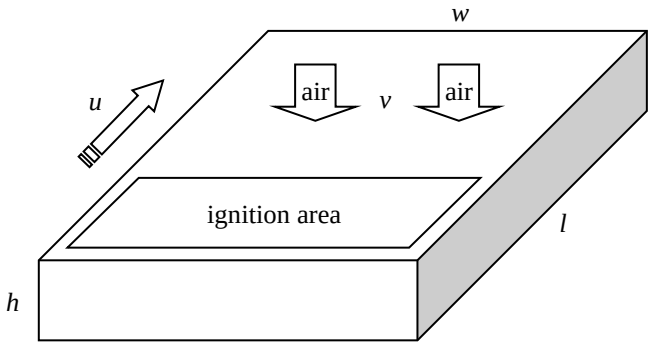

Fig.1. Considered geometry of sintering layer

The temperature distribution within the sinter bed is governed by the energy balance equation, considering heat conduction, convection, and radiation:

$$\rho c_p \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T) - \rho c_p \vec{\mathbf{u}} \cdot \nabla T + \sigma \varepsilon \left(T^4 - T_0^4 \right) + Q \text{ ,} \tag{1}$$

where ρ – the density of the sinter layer, c_p – the specific heat capacity, T – temperature, k – thermal conductivity, u – velocity vector of the layer, σ – the Stefan-Boltzmann constant, ε – emissivity of the sinter bed, T_0 – the ambient temperature, Q – heat change rate due chemical reactions. The movement of air and sinter layer is described by the mass conservation equation:

$$\nabla \cdot \vec{\mathbf{v}} = 0 \qquad \nabla \cdot \vec{\mathbf{u}} = 0 \tag{2}$$

The combustion of carbon is a fundamental process contributing to heat release in the sinter bed. This reaction involves the oxidation of carbon by oxygen. The corresponding rate equation for this reaction can be expressed as

$$\frac{\partial (\rho Y_C)}{\partial t} = -k_C \rho Y_C \left(\frac{P_{O_2}}{P_{O_2,0}} \right)^{0.5}, \tag{3}$$

where Y_C – mass fraction of carbon, k_C is the rate constant for carbon combustion, P_{O2} – partial pressure of oxygen, $P_{O2,0}$ – reference partial pressure of oxygen. The heat release associated with the combustion of carbon is added as an additional term in the energy balance equation:

$$Q_{combustion} = -\Delta H_C \frac{\partial (\rho Y_C)}{\partial t}, \tag{4}$$

where ΔH_C – heat of combustion for carbon.

The equations are supplemented by boundary conditions. There are inlet and outlet conditions, as well as convective heat transfer at the surface due to air flow:

$$u \cdot \nabla T = a \left(T - T_{ambient} \right), \tag{5}$$

where a – heat transfer coefficient.

Complementing the mathematical model, it has been developed a specialized computer program using C++ programming language, which calculates a numerical solution to discretized above equations. The method of central differences is very suitable for GPU-computing and was used to solve the equations. The single-precision floating point numbers are used whenever possible. To facilitate GPU acceleration, the mathematical model is translated into a parallelizable form. Computational tasks are decomposed to exploit the parallel architecture of GPUs, with each GPU thread handling a specific subset of calculations. This parallelized implementation enables concurrent processing, significantly reducing the overall simulation time.

Algorithm of GPU-computing:

1) Allocate GPU memory with data that corresponds to initial conditions and constants. It is used a linear memory layout for coalesced access (division by the least significant dimension) to improve data transfer rates.

2) Calculate the number of GPU-thread groups needed to distribute the workload evenly across the computational domain. The groups have access to high-speed group shared memory.

3) Start of time-stepping loop, which does calculation exclusively on the GPU, which increases its load to almost 100%. Each GPU thread processes a specific grid cell and updates the memory based on the mathematical model. Calculation using group-shared memory accelerates a frequent access to the same cells. Also there is boundary conditions' handling here.

4) After the time-stepping loop, the simulation results are copied from GPU memory to RAM. The results are saved to hard disk and optionally validated, error-checked to ensure numerical stability and accuracy of the GPU-accelerated simulation.

Also the program integrates a comprehensive database housing experiment data and parameters (Fig. 2). The database facilitates calibration of the model, enhancing its accuracy and applicability to diverse ore sintering machine configurations. This engine allows users to input specific parameters related to their ore sintering setup, initiating simulations and generating detailed results. To ensure accessibility and ease of use, the program features an intuitive user interface. Researchers can input key parameters through a user-friendly graphical interface. This enhances the program's applicability across different user backgrounds and expertise levels.

The Microsoft SQL Server and Microsoft ASP.NET technologies are used to implement the database model and user interface. For database interaction, the LINQ to SQL is used as a simple and convenient solution. C# is selected as web-server programming language, which offers both fast project realization and best-practices for code structure.

Using the program many researchers can conduct experiments and store information simultaneously. For example, one opens the user interface, adds research and factors to consider. The program helps him to generate factor levels and a full factorial experiment. The only thing to do is to import experiment results. Using the results the program is able to draw a dependency chart after a researcher specifies a variable both for the x and the y axis. Also it can group results to create multiple series on the single chart.

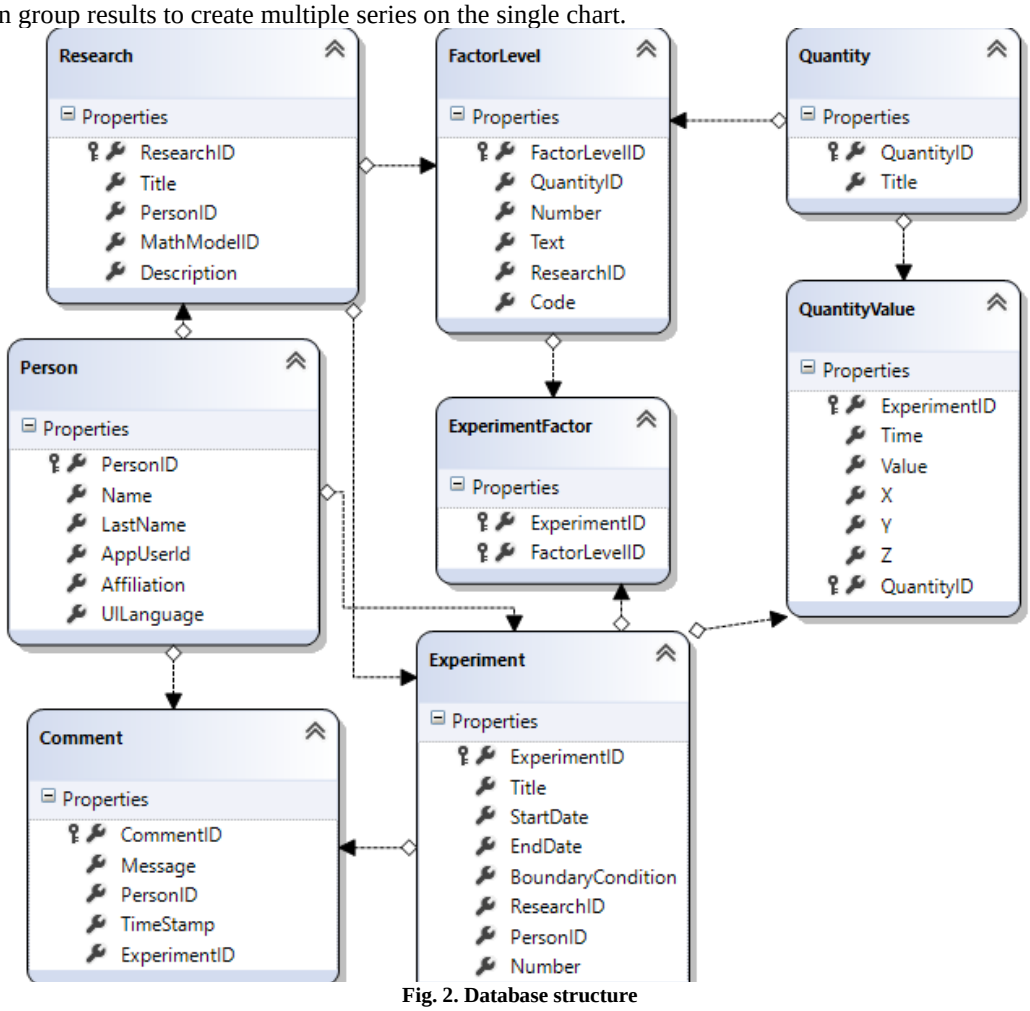

Fig. 2. Database structure

Experiments

Our investigation extends to the scalability of the GPU-accelerated model. We explore its performance across varying problem sizes (the second column in the table 1), confirming its ability to handle larger and more complex simulations efficiently. Scalability is assessed by progressively increasing the computational demands and evaluating the model's responsiveness on different GPU architectures (the third column in the table 1). A full factorial design would involve evaluating all possible combinations of all levels.

A detailed performance analysis is conducted to quantify the speedup achieved through GPU acceleration. Comparative metrics, such as computation time and resource utilization, are assessed across a range of scenarios. This evaluation provides insights into the practical benefits of utilizing GPUs for ore sintering simulations.

The result is a model time that corresponds to number of time layers the solver reaches before 60 seconds of a real time are elapsed. Thus, longer modeling time means better acceleration efficiency.

Table 1

Experimental field (24 experiments in total for full factorial experimental design)

Levels \ Factors	Number of calculated cells	Processor
2	1 382 400	Nvidia GeForce GTX 1050
1	691 200	Intel UHD 630
0	345 600	Microsoft WARP
-1	172 800	Intel CPU 1 core
-2	86 400	
-3	43 200	

Fig. 3 shows the dependency chart of the model time on the problem size. As seen, there is a strong correspondence between how fast the model time decreases and how fast the problem size increases. The yellow processor can reach the model time of the blue one, but at cost of significant (around 8 times) decrease of problem size. The yellow processor is 6-cores CPU with AVX2 enabled.

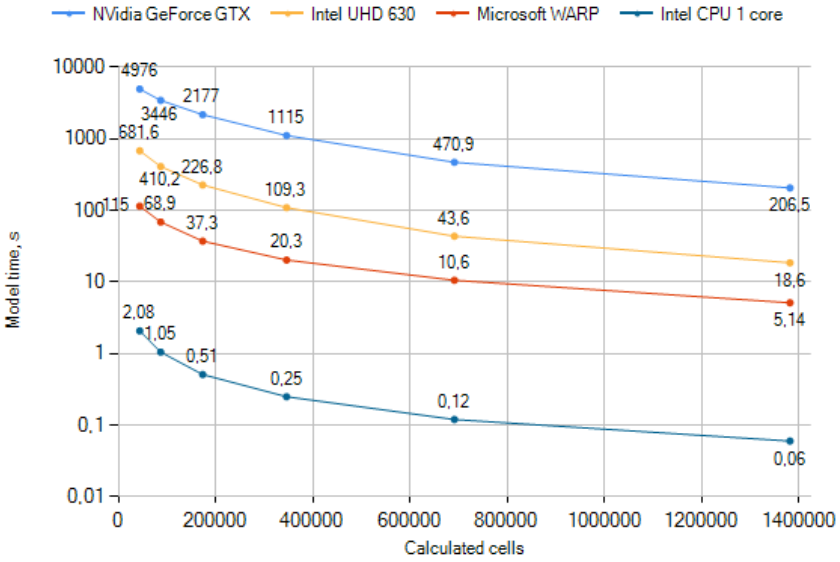

Fig. 3. Dependency of computed model time on the discretization density (higher values are better)

As grid resolution increases, it is expected the numerical solution to converge towards a more accurate temperature field (Fig. 4). The chart illustrates the average temperature stabilization as grid is refined.

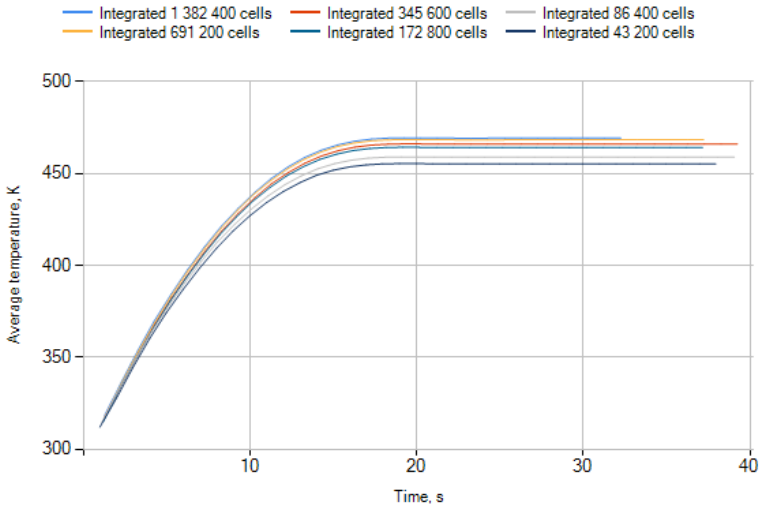

Fig. 4. Dependency of average temperature on time (taking discretization size into account)

To see a recorded visualization of temperature evolution according to the mathematical model please visits the Internet link https://www.tensorion.com/lab/metallurgy/sintering.

Conclusions

The substantial boost in calculated model time as we switch to GPU indicates a positive performance scaling. This suggests that GPUs is efficiently handling larger problem sizes, likely benefiting from parallel processing capabilities. The GPU is likely taking advantage of parallel processing to handle larger workloads more efficiently than even multicore CPU. This is consistent with the typical behavior of GPUs, which excel at massively parallelizable tasks.

The results (Fig. 3) show that comparing to the multicore CPU the integrated GPU achieves 5,9x speedup at the lowest problem size and 3,6x speedup at the largest problem size, so acceleration degrades as problem size rises and the rate of improvement diminishes. This is a common characteristic, and it's essential to consider the law of diminishing returns when scaling up computational resources. It worth noting that difference between the integrated GPU and the single core CPU is almost the same – near 310-328x speedup – on full range of problem size.

Regarding the implemented automated info-research system with database and user interface the measurements show that comparing to a manual experiment planning and analyzing the automated system increased speed of research by 47 % and decreased human error risk to zero, as well as increased scalability of the research process.

References

1. Fernández-González D., Ruiz-Bustinza I., Mochón J., González-Gasca C., Verdeja L. F. Iron Ore Sintering: Process. *Mineral Processing and Extractive Metallurgy Review*, 38(4), 215-227. 2017. doi: 10.1080/08827508.2017.1288115
2. Shrestha S., Xu J., Yu A., Zhou Z. Numerical simulation of fuel layered distribution iron ore sintering technology. *Ironmaking & Steelmaking*. 2022; 49(1):83-100. doi:10.1080/03019233.2021.1968259
3. Skorych V., Dosta M. Parallel CPU–GPU computing technique for discrete element method. *Concurrency and Computation: Practice and Experience*. 2022; 34(11):e6839. doi:10.1002/cpe.6839
4. Pahlevaninezhad M., Emami M.D., Panjepour M. Identifying major zones of an iron ore sintering bed. *Applied Mathematical Modelling*. Volume 40, Issues 19–20, 2016. doi: 10.1016/j.apm.2016.05.005.
5. Zhang M., Hibi K., Inoue J. GPU-accelerated artificial neural network potential for molecular dynamics simulation. *Computer Physics Communications*. Volume 285, 2023. doi: 10.1016/j.cpc.2022.108655.
6. Matsuda T. Finite-difference calculation on an uncertainty evaluation of the kinetic parameters from initial sintering of UO2. *Int J Ceramic Eng Sci*.1: 206–215, 2019. doi: 10.1002/ces2.10023.
7. Mochurad L., Ilkiv A. (2022). A novel method of medical classification using parallelization algorithms. *Computer Systems and Information Technologies*, (1), 23–31. doi: 10.31891/CSIT-2022-1-3.
8. Mochurad L. (2023). Cuda-based parallelization of gradient boosting and bagging algorithm for diagnosing diabetes. *Computer Systems and Information Technologies*, (2), 6–16. doi: 10.31891/csit-2023-2-1.
9. Barkovska O. (2022). Performance study of the text analysis module in the proposed model of automatic speaker's speech annotation. *Computer Systems and Information Technologies*, (4), 13–19. https://doi.org/10.31891/csit-2022-4-2

Kyrylo KRASNIKOV **Кирило КРАСНІКОВ**	PhD, Associate Professor of Systems Software Department, Dniprovskyi State Technical University, Kamianske, Ukraine, e-mail: kir_kras@ukr.net https://orcid.org/0000-0002-4241-0572 Scopus Author ID 57205319971, ResearcherID: Q-8835-2017	кандидат технічних наук, доцент кафедри програмного забезпечення систем, Дніпровський державний технічний університет, Кам'янське, Україна.