# MATHEMATICAL MODELLING
# МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ





UDC 004.032.26

https://doi.org/10.23947/2587-8999-2025-9-2-44-51

*Original Empirical Research*

## Application of Neural Networks for Solving Elliptic Equations in Domains with Complex Geometries

**Alexander V. Galaburdin**
Don State Technical University, Rostov-on-Don, Russian Federation
✉ Galaburdin@mail.ru

**Abstract**

***Introduction.*** Differential equations are often used in modelling across various fields of science and engineering. Recently, neural networks have been increasingly applied to solve differential equations. This paper proposes an original method for constructing a neural network to solve elliptic differential equations. The method is used for solving boundary value problems in domains with complex geometric shapes.

***Materials and Methods.*** A method is proposed for constructing a neural network designed to solve partial differential equations of the elliptic type. By applying a transformation of the unknown function, the original problem is reduced to Laplace's equation. Thus, nonlinear differential equations were considered. In building the neural network, the activation functions are chosen as derivatives of singular solutions to Laplace's equation. The singular points of these solutions are distributed along closed curves encompassing the boundary of the domain. During the training process, the weights of the network are adjusted by minimizing the mean squared error.

***Results.*** The paper presents the results of solving the first boundary value problem for various domains with complex geometries. The results are shown in tables containing both the exact solutions and the solutions obtained using the neural network. Graphical representations of the exact and the neural network-based solutions are also provided.

***Discussion and Conclusion.*** The obtained results demonstrate the effectiveness of the proposed neural network construction method in solving various types of elliptic partial differential equations. The method can also be effectively applied to other types of partial differential equations.

**Keywords:** elliptic partial differential equations, domain with complex geometry, neural networks

**For Citation.** Galaburdin A.V. Application of Neural Networks for Solving Elliptic Equations in Domains with Complex Geometries. *Computational Mathematics and Information Technologies*. 2025;9(2):44–51. https://doi.org/10.23947/2587-8999-2025-9-2-44-51

*Оригинальное эмпирическое исследование*

## Применение нейронных сетей при решении эллиптических уравнений для областей сложной формы

**А.В. Галабурдин**
Донской государственный технический университет, г. Ростов-на-Дону, Российская Федерация
✉ Galaburdin@mail.ru

**Аннотация**

***Введение.*** При построении моделей в различных областях науки и техники часто используют дифференциальные уравнения. В настоящее время при решении дифференциальных уравнений все чаще применяются нейронные сети. В данной работе предложен оригинальный метод построения нейронной сети для решения эллиптических дифференциальных уравнений. Этот метод применяется при решении краевых задач для областей сложной геометрической формы.

***Материалы и методы.*** Предлагается метод построения нейронной сети, предназначенной для решения дифференциальных уравнений в частных производных эллиптического типа. Используя замену неизвестной функции, исходная задача сводится к уравнению Лапласа. Таким образом, рассматривались нелинейные дифференциальные уравнения. При построении нейронной сети в качестве активационных функций принимаются производные от сингулярных решений уравнения Лапласа. Сингулярные точки этих решений распределены по замкнутым кривым, охватывающим границу области. При настройке весов сети минимизировалась среднеквадратическая ошибка обучения.

***Результаты исследования.*** Представлены результаты решения первой краевой задачи для различных областей сложной геометрической формы. Результаты представлены в виде таблиц, содержащих точные решения задачи и решения, полученные с помощью нейронной сети. Дано графическое представление точного решения и решение, полученное предложенным методом.

***Обсуждение и заключение.*** Полученные результаты доказали эффективность предложенного метода построения нейронной сети при решении различных видов дифференциальных уравнений в частных производных эллиптического типа. Данный метод может эффективно применяться при решении других типов дифференциальных уравнений с частными производными.

**Ключевые слова:** дифференциальные уравнения в частных производных эллиптического типа, область сложной геометрической формы, нейронные сети

**Для цитирования.** Галабурдин А.В. Применение нейронных сетей при решении эллиптических уравнений для областей сложной формы. *Computational Mathematics and Information Technologies*. 2025;9(2):44–51. https://doi.org/10.23947/2587-8999-2025-9-2-44-51

**Introduction.** Differential equations play a crucial role in modelling processes across various fields of science and engineering. Traditional analytical and numerical methods for solving differential equations do not always yield satisfactory results. As a result, different machine learning methods are increasingly being applied to solve differential equations. In particular, artificial neural networks are often used for this purpose.

The theoretical foundations of the neural network method can be traced back to the work of A.N. Kolmogorov [1]. Today, neural networks are widely em-ployed for solving different types of differential equations. In [2], the transition from neural network architecture to ordinary differential equations and the Cau-chy problem is discussed.

Papers [3, 4] focus on the application of neural networks to solve Laplace's equation. In [5], deep learning methods are applied to solve the Poisson equation in a two-dimensional domain. Radial basis function (RBF) neural networks have become particularly widespread in solving partial differential equations [6].

In studies [7, 8], radial basis functions with tunable parameters are used as activation functions. Works [9–11] demonstrate the successful use of neural networks for solving boundary value problems related to the Navier–Stokes equations. Physics-informed neural networks (PINNs) have shown high effectiveness in solving partial differential equations, particularly in classical mechanics problems [12, 13]. In [14], a perceptron-type neural network is applied to a heat and mass transfer problem.

These studies highlight the growing popularity of neural networks for solving differential equations. The present research is devoted to the analysis of boundary value problems for partial differential equations in domains with complex geometries and builds on the approach developed in [15, 16].

**Materials and Methods.** Let us consider a boundary value problem for a differential equation:

$$U + b_1\partial_1 U + b_2\partial_2 U + cU = 0.$$

By representing the solution in the form $U=Ve^{(\lambda x+\alpha y)}$ and appropriately selecting the parameters λ and α, the problem can be reduced to a simpler equation:

$$V + aV = 0.$$

i. e., the Laplace equation.

The resulting equation was solved using a neural network with respect to the function $V$. The constructed neural networks for solving the Laplace equation can also be used to solve nonlinear elliptic equations, provided they are properly transformed.

As an example, consider the differential equation:

$$U - 2((\partial_1 U)^2 + (\partial_2 U)^2)/(3V) = 0.$$

The original differential equation is reduced to the Laplace equation by introducing a new unknown function $V = U^{1/3}$.

The neural network construction was based on the method described in [15, 16]. This method relies on a formula similar to Green's formula, in which integrals are replaced by sums:

$$V(x) = \sum_{k=1}^{N} w_k f(s_k) U(x, \sigma_k) + \sum_{k=1}^{N} v_k f(s_k) G(x, \tau_k),$$

where $f(s_k)$ is the value of the unknown function $u$ on the boundary of the domain; $U(x, \sigma_k)$ and $G(x, \tau_k)$ are activation functions; $\sigma_k$ and $\tau_k$ are points on closed curves $\gamma_1$ and $\gamma_2$, which surround the boundary $\gamma$ of the domain; $x$ is a point inside the domain $G$.

By requiring that this relation holds at every point on the boundary for all functions in the training set, and applying the least squares method, a system of equations is obtained for determining the weights $w_k$ and $v_k$.

To improve the conditioning of the matrix in the resulting system of equations, the activation functions were chosen as derivatives of the fundamental solution of the Laplace equation

$$U(x, y, t, s) = \frac{\beta^5 - 10\beta^3\delta^2 + 5\beta\delta^4 + \delta^5 - 10\delta^3\beta^2 + 5\delta\beta^4}{R^{10}},$$

$$G(x, y, t, s) = \frac{\beta^7 - 21\beta^5\delta^2 + 35\beta^3\delta^4 - 7\beta\delta^6}{R^{14}} n_x,$$

$$+ \frac{\delta^7 - 21\beta^2\delta^5 + 35\delta^3\beta^4 - 7\beta^6\delta}{R^{14}} n_y,$$

$$\delta = x - t, \beta = y - s, R = \sqrt{\delta^2 + \beta^2}.$$

This increased the singularity of the activation functions. The points $\sigma_k$ and $\tau_k$ were taken on the contours $\gamma_1$ and $\gamma_2$, which were obtained by shifting each point of the boundary contour $\gamma$ outward along the external normal to the domain boundary by distances $\rho_1$ and $\rho_2$ respectively. During the training process, the weights as well as the values of $\rho_1$ and $\rho_2$ were determined. The values $\rho_1$ and ρ2 were found using a simple brute-force search.

As a training set, a set of functions that are solutions to the Laplace equation in polar coordinates was used:

$$r^k cos(karccos(\frac{x}{r})) + r^k sin(karccos(\frac{x}{r})),\ r = \sqrt{(x^2 + y^2)},$$

where $k = 0, 1, 2, 3, \ldots, M$.

These functions were specified in different coordinate systems, each rotated relative to one another by an angle that is a multiple of $2\pi/5$.

**Results.** The proposed method was applied to solving problems in domains whose boundary $\gamma$ is defined by the equation:

$$\begin{cases} x = a\cos(t) + g\cos(\omega t), \\ y = a_1 \sin(t) + g_1 \sin(\omega t), \end{cases} \quad t \in [0, 2\pi],$$

where $t \in [0,2\pi]$; $a$, $a_1$, $g$, $g_1$, $\omega$ are adjustable parameters.

In all cases, the number of functions in the training set was taken as $M = 75$, and the number of neurons in the network was $N = 100$.

**Problem 1.** As an example, consider the following differential equation:

$$\Delta U - \partial_1 U + 5\partial_2 U + 6.5U = 0.$$

A new unknown function $V$ is introduced, which satisfies the Laplace equation:

$$U = Ve^{(0.5x - 2.5y)}.$$

The first boundary value problem was considered.

Fig. 1 shows the domain whose boundary corresponds to the following parameter values: $a = 1.15$, $g = 1.15$, $a_1 = 0.07$, $g_1 = -0.03$, $\omega = 9$.

The points in the domain where both the exact solution and the neural network solution are evaluated are marked with asterisks in the diagram.

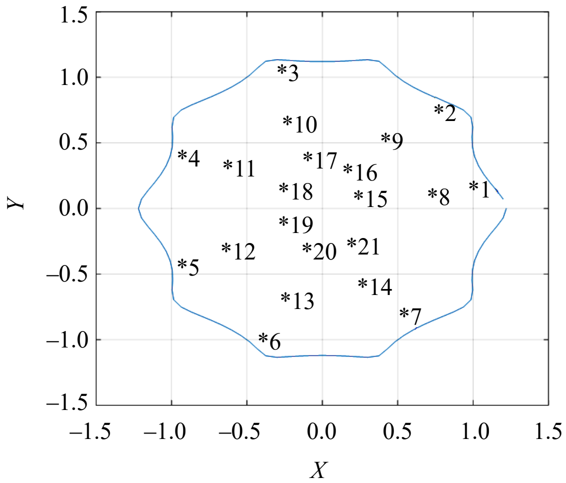


Fig. 1. Shape of the domain for Problem 1

The points in the domain where the exact solution and the neural network solution are computed are marked with asterisks on the diagram. Table 1 presents the computational results corresponding to the solution of the equation:

$$U = xye^{(0.5x-2.5y)}.$$

Table 1

Computation Results for Problem 1

| Point Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Exact Solution | 0.1158 | 0.2214 | 0.1055 | 0.0241 | –0.0142 | –0.0473 | –0.0835 |
| Neural Network Solution | 0.1148 | 0.2216 | 0.1054 | 0.0240 | –0.0142 | –0.0473 | –0.0838 |
| Point Number | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Exact Solution | 0.0385 | 0.1092 | 0.0769 | 0.0244 | –0.0157 | 0.0457 | –0.0629 |
| Neural Network Solution | 0.0382 | 0.1090 | 0.0768 | 0.0243 | –0.0158 | 0.0458 | –0.0630 |
| Point Number | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| Exact Solution | 0.0051 | 0.0213 | 0.0222 | 0.0098 | –0.0069 | –0.0175 | –0.0188 |
| Neural Network Solution | 0.0047 | 0.0210 | 0.0220 | 0.0096 | –0.0071 | –0.0177 | –0.0190 |

**Problem 2.** The following differential equation was considered:

$$\Delta U + 5\partial_1 U + 3\partial_2 U + 8.5U = 0.$$

The introduction of a new unknown function $V$:

$$U = Ve^{-(2.5x+1.5y)}$$

allows the original differential equation to be reduced to the Laplace equation with respect to the function $V$. The shape of the domain in this case was determined by the parameters в $a = 1.1$, $g = 1.1$, $a1 = 0.05$, $g1 = 0.1$, $\omega = 4$ (Fig. 2). The first boundary value problem was considered. Table 2 presents the computational results and the exact solution of the differential equation:

$$U = e^{-x} chye^{-(2.5x+1.5y)}.$$

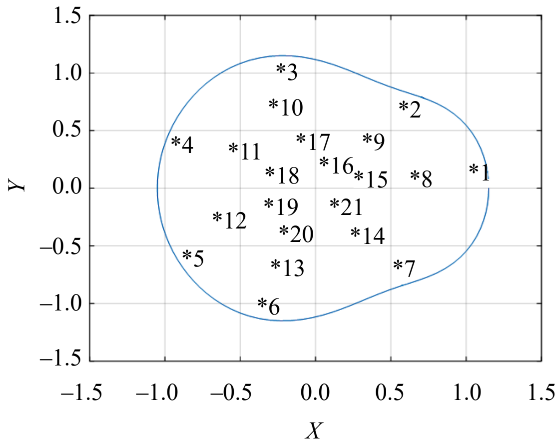


Fig. 2. Shape of the domain for Problem 2

Table 2

Computation Results for Problem 2

| Point Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Exact Solution | 0.0198 | 0.0143 | 0.0245 | 0.0622 | 0.1929 | 1.4682 | 9.0855 |
| Neural Network Solution | 0.0198 | 0.0143 | 0.0245 | 0.0622 | 0.1928 | 1.4640 | 9.0885 |
| Point Number | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Exact Solution | 0.0818 | 0.0700 | 0.1163 | 0.2289 | 0.4502 | 1.4325 | 3.9076 |
| Neural Network Solution | 0.0817 | 0.0700 | 0.1162 | 0.2288 | 0.4499 | 1.4320 | 3.9104 |
| Point Number | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| Exact Solution | 0.3377 | 0.3321 | 0.4896 | 0.6716 | 0.7856 | 1.1957 | 1.6099 |
| Neural Network Solution | 0.3377 | 0.3320 | 0.4896 | 0.6716 | 0.7857 | 1.1959 | 1.6104 |

**Problem 3.** The following differential equation was considered:

$$\Delta U - \frac{2((\partial_1 U)^2 + (\partial_1 U)^2)}{3V} = 0,$$

By introducing a new unknown function  the original equation is reduced to the Laplace equation. The first boundary value problem was considered. The shape of the domain was defined by the parameters $a = 1.1$, $g = 1.1$, $a_1 = 0.07$, $g_1 = 0.07$, $\omega = 9$ (Fig. 3).

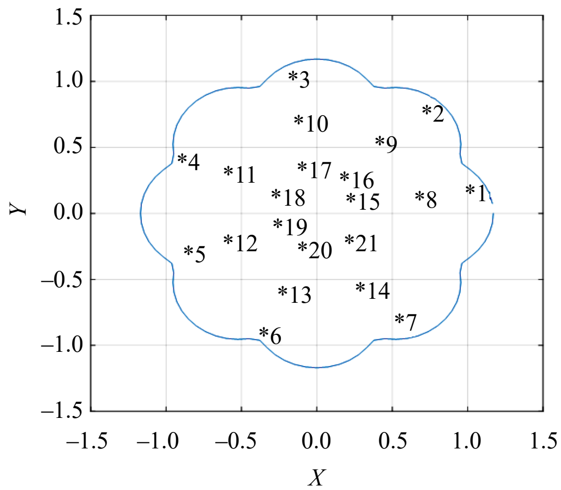


Fig. 3. Shape of the domain for Problem 3

Table 3 presents the computational results and the exact solution of the differential equation:

$$U = (xy + 2.5x + y)^3.$$

Table 3

Computation Results for Problem 3

| Point Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Exact Solution | 24.613 | 28.736 | 32.687 | 7.2807 | 0.0363 | –2.0562 | 10.894 |
| Neural Network Solution | 24.613 | 28.878 | 32.610 | 7.3419 | 0.0380 | –2.0870 | 10.834 |
| Point Number | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Exact Solution | 5.7339 | 5.9627 | 6.3055 | 1.4899 | 0.0169 | 0.2857 | –2.0582 |
| Neural Network Solution | 5.7405 | 5.9727 | 6.3164 | 1.4937 | 0.0169 | 0.2865 | –2.0625 |
| Point Number | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| Exact Solution | 0.3331 | 0.3060 | 0.2982 | 0.0754 | 0.0017 | –0.0088 | –0.0950 |
| Neural Network Solution | 0.3328 | 0.3058 | 0.2982 | 0.0753 | 0.0017 | –0.0089 | –0.0957 |

Fig. 4 and 5 show graphical solutions obtained using the neural network, as well as the exact solution of Problem 3.

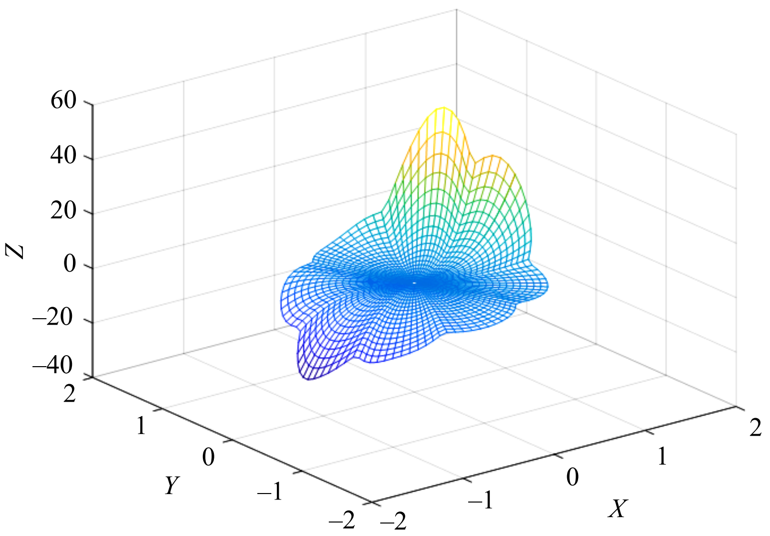


Fig. 4. Solution of Problem 3 obtained by the neural network

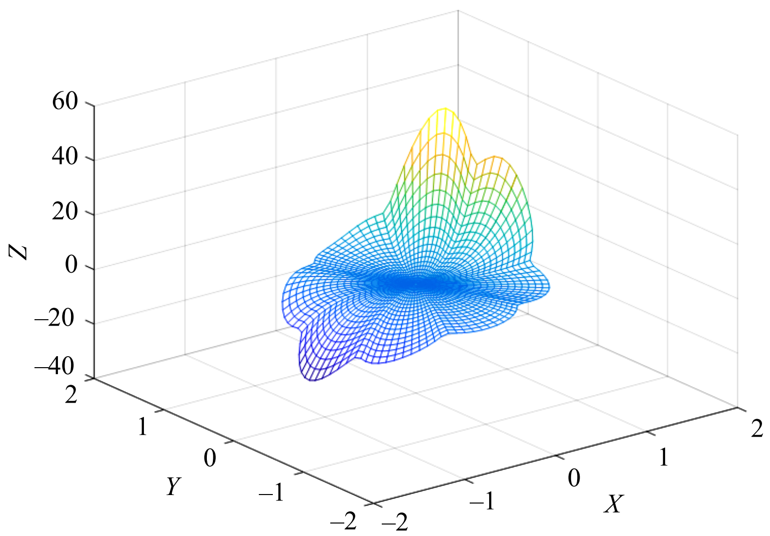


Fig. 5. Exact solution of Problem 3

**Discussion and Conclusion.** The presented results once again demonstrated the effectiveness of the neural network construction method for solving boundary value problems in domains of complex shape for various types of elliptic partial differential equations. This method can efficiently handle all types of partial differential equations. Future development of the method will focus on expanding the classes of solvable problems and improving training techniques.

**References**

1. Kolmogorov A.N. On the representation of continuous functions of several variables in the form of superpositions of continuous functions of one variable and addition. *Reports of the USSR Academy of Sciences*. 1957;114(5):953–956. (In Russ.)

2. Margasov O.A. On neural ordinary differential equations and their probabilistic extension. *Izvestiya of the Komi Science Center of the Ural Branch of the Russian Academy of Sciences. Series "Physical and Mathematical Sciences"*. 2021;6(52):14–19. https://doi.org/10.19110/1994-5655-2021-6-14-19 (In Russ.)

3. Varshavchik E.A., Galyautdinova A.R., Sedova Yu.S., Tarkhov D.A. Solution of partial differential equations for domains with constant boundaries. *Artificial intelligence in solving current social and economic problems of the 21st century: collection of articles. Art. based on materials from the Third All-Russian. scientific-practical conf.* Perm, May 14–18, 2018. Perm. state national research univ.; 2018. 294 p. (In Russ.)

4. Tyurin K.A., Bragunets V.V., Svetlov D.D. Solution of the Laplace differential equation using a modified neural network. *Young Scientist*. 2019;27(265):10–12. (In Russ.)

5. Epifanov A.A. Application of deep learning methods for solving partial differential equations. *Advances in cybernetics*. 2020;1(4):22–28. https://doi.org/10.51790/2712-9942-2020-1-4-3 (In Russ.)

6. Kovalenko A.N., Chernomorets A.A., Petina M.A. On the use of neural networks for solving partial differential equations. *Scientific bulletin. The Economics series. Computer science*. 2017;9(258):103–110. (In Russ.)

7. Kansa E.J. *Motivation for using radial basis functions to solve PDEs.* URL: http://uahtitan.uah.edu/kansaweb.html (accessed: 13.04.2025)

8. Kansa E.J. Multiquadrics. A scattered data approximation scheme with applications to computational fluid-dynamics. II. Solutions to parabolic, hyperbolic and elliptic partial differential equations. *Comput. Math. Appl*. 1990;19(89):147–161.

9. Almajid M. Abu-Alsaud M. Prediction of porous media fluid flow using physics informed neural networks. *Journal of Petroleum Science and Engineering*. 2021;208:109205. https://doi.org/10.1016/j.petrol.2021.109205

10. Eivazi H.,Tahani M., Schlatter P., Vinuesa R. Physics-informed neural networks for solving Reynolds-averaged Navier-Stokes equations. *Physics of Fluids*. 2022;34(7):075117. https://doi.org/10.1063/5.0095270

11. Chen J., Viquerat J., Hachem E. *U-net Architectures for Fast Prediction of Incompressible Laminar Flows*. URL: https://arxiv.org/pdf/1910.13532.pdf (accessed: 13.04.2025)

12. Raissia M., Perdikaris P., Karniadakis G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*. 2019;378:686–707.

13. Zrelova D.P., Ulyanov S.V. Models of physically informed aware classical Lagrangian Hamiltonian neural networks in deep learning. *Modern information technologies and IT education*. 2022;18(2):310–325. https://doi.org/10.25559/SITITO.18.202202.310-325 (In Russ.)

14. Cai S., Wang Z., Wang S., Perdikaris P., Karniadakis G.E. Physics-informed neural networks for heat transfer problems. *Journal of Heat Transfer*. 2021;143(6):60–80. https://doi.org/10.1115/1.4050542

15. Galaburdin A.V. Application of neural networks to solve the Dirichlet problem for areas of complex shape. *Computational Mathematics and Information Technologies*. 2024;8(2):68–79. https://doi.org/10.23947/2587-8999-2024-8-2-68-79 (In Russ.)

16. Galaburdin A.V. Application of neural networks to solve nonlinear boundary value problems for areas of complex shape. *Computational Mathematics and Information Technologies*. 2024;8(4):35–42. https://doi.org/10.23947/2587-8999-2024-8-4-35-42 (In Russ.)

***About the Author:***

**Alexander V. Galaburdin,** Cand. Sci. (Phys. – math.), Associate Professor at the Department Mathematics and informatics, Don State Technical University (1, Gagarin Sq., Rostovon-Don, 344003, Russian Federation), ORCID, SPIN-code, Galaburdin@mail.ru

***Conflict of Interest Statement:*** **the author declares no conflict of interest.**

***The author has read and approved the final version of manuscript.***

***Об авторе:***

**Александр Васильевич Галабурдин,** кандидат физико-математических наук, доцент кафедры математики и информатики Донского государственного технического университета (344003, Российская Федерация, г. Ростов-на-Дону, пл. Гагарина, 1), ORCID, SPIN-код, Galaburdin@mail.ru

***Конфликт интересов:*** **автор заявляет об отсутствии конфликта интересов.**

***Автор прочитал и одобрил окончательный вариант рукописи.***

**Received / Поступила в редакцию** 19.05.2025
**Reviewed / Поступила после рецензирования** 11.06.2025
**Accepted / Принята к публикации** 25.06.2025