INFORMATION TECHNOLOGIES
ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Check for updates

UDC 004.032.26

Original Theoretical Research

https://doi.org/10.23947/2587-8999-2024-8-4-35-42

Application of Neural Networks for Solving Nonlinear Boundary Problems for Complex-Shaped Domains

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

Abstract

Introduction. Many practically significant tasks reduce to nonlinear differential equations. In this study, one of the applications of neural networks for solving specific nonlinear boundary problems for complex-shaped domains is considered. Specifically, the focus is on solving a stationary heat conduction differential equation with a thermal conductivity coefficient dependent on temperature.

Materials and Methods. The original nonlinear boundary problem is linearized through Kirchhoff transformation. A neural network is constructed to solve the resulting linear boundary problem. In this context, derivatives of singular solutions to the Laplace equation are used as activation functions, and these singular points are distributed along closed curves encompassing the boundary of the domain. The weights of the network were tuned by minimizing the mean squared error of training.

Results. Results for the heat conduction problem are obtained for various complex-shaped domains and different forms of dependence of the thermal conductivity coefficient on temperature. The results are presented in tables that contain the exact solution and the solution obtained using the neural network.

Discussion and Conclusion. Based on the computational results, it can be concluded that the proposed method is sufficiently effective for solving the specified type of boundary problems. The use of derivatives of singular solutions to the Laplace equation as activation functions appears to be a promising approach.

Keywords: nonlinear boundary problems for complex-shaped domains, neural networks

Acknowledgments. The author expresses gratitude to the editorial staff for their remarks on the formatting, which allowed the article to be made more accessible.

For Citation. Galaburdin A.V. Application of Neural Networks for Solving Nonlinear Boundary Problems for Complex-Shaped Domains. *Computational Mathematics and Information Technologies*. 2024;8(4):35–42. https://doi.org/10.23947/2587-8999-2024-8-4-35-42

Оригинальное теоретическое исследование

Применение нейронных сетей для решения нелинейных краевых задач для областей сложной формы

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

Аннотация

Введение. Многие практически важные задачи сводятся к нелинейным дифференциальным уравнениям. В настоящей работе рассмотрен один из вариантов применения нейронных сетей к решению некоторых нелинейных краевых задач для областей сложной формы, а именно к решению стационарного дифференциального уравнения теплопроводности с коэффициентом теплопроводности, зависящим от температуры.

Материалы и методы. Исходная нелинейная краевая задача сводится к линейной с помощью преобразования Кирхгофа. Нейронная сеть строится для решения полученной линейной краевой задачи. При этом в качестве активационных функций принимаются производные от сингулярных решений уравнения Лапласа, а сингулярные точки этих решений распределены по замкнутым кривым, охватывающим границу области. Для настройки весов сети минимизировалась среднеквадратическая ошибка обучения.

Результаты исследования. Получены результаты решения задачи теплопроводности для различных областей сложной формы и различных форм зависимости коэффициента теплопроводности от температуры. Полученные результаты представлены в виде таблиц, которые содержат точное решение и решение, полученное при помощи нейронной сети.

Обсуждение и заключение. По результатам проведенных расчетов можно сделать вывод о том, что предложенный метод является достаточно эффективным для решения указанного типа краевых задач. Использование в качестве активационных функций производных от сингулярных решений уравнения представляется весьма перспективным.

Ключевые слова: нелинейные краевые задачи для областей сложной формы, нейронные сети

Благодарности. Автор выражает благодарность сотрудникам редакции за указанные замечания по оформлению, которые позволили сделать статью более удобной для восприятия.

Для цитирования. Галабурдин А.В. Применение нейронных сетей для решения нелинейных краевых задач для областей сложной формы. *Computational Mathematics and Information Technologies*. 2024;8(4):35–42. https://doi.org/10.23947/2587-8999-2024-8-4-35-42

Introduction. In constructing models of various natural phenomena, the apparatus of differential equations is often employed. The complexity of the modeled phenomena leads to complex systems of differential equations with intricate domain shapes. Currently, in solving such boundary problems, the method of neural networks is increasingly utilized.

It should be noted that the theoretical foundations of the neural network method were laid in the mid-20th century by A.N. Kolmogorov [1]. The development of the theory in [2] is applied to solving the problem of membrane deflection. In [3], a neural network structure is proposed that allows solving Laplace, Poisson, and heat conduction equations. The numerical solution of the Poisson equation in a two-dimensional domain, obtained by the Galerkin method and Ritz method with deep neural networks, is presented in [4]. In article [5], approaches to solving heat and mass transfer problems based on a perceptron-type neural network are explored.

Recently, there has been a frequent use of physically-informed neural networks to solve partial differential equations [6]. Article [7] presents solutions to classical mechanics problems through the application of physically-informed neural networks. In [8], an approach to solving direct and inverse scattering problems using radial basis function neural networks is discussed. In article [9], based on the method of trust regions, a training method for RBF networks with a customizable functional basis is developed for solving boundary problems in mathematical physics. Article [10] studies the use of physically-informed neural networks in solving unsteady nonlinear differential equations describing the motion of a one-dimensional heat-conducting gas. In works [11, 12], neural networks are applied to solve the Navier-Stokes equations. In works [13, 14], radial basis functions are used as activation functions in the neural network, and their parameters are varied during training.

This work is a development of the approach to solving partial differential equations using neural networks as presented in article [15]. The aim of this study is to develop a method for applying neural networks to solve nonlinear boundary problems for complex-shaped domains.

Materials and Methods. Consider the boundary problem for the nonlinear differential equation

$$\frac{\partial}{\partial x}\left(k\left(W\right)\frac{\partial W}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\left(W\right)\frac{\partial W}{\partial y}\right)=0 \tag{1}$$

on the planar domain G bounded by a closed curve γ.

This equation describes a stationary thermal field. In this context, W represents the temperature and $k(W)$ represents the thermal conductivity coefficient. Using the Kirchhoff transformation [16, 17], this problem is reduced to a linear form. The essence of the transformation is to introduce a function $u(W)$, such that

$$grad\left(u\left(W\right)\right)=\frac{du\left(W\right)}{dW}grad\left(W\right).$$

Then we have

$$\frac{du\left(W\right)}{dW}=k\left(W\right) \tag{2}$$

where the original differential equation takes the form of $\Delta u(x, y) = 0$.

From equation (2) we obtain

$$u(x,y)=\int_{w_o}^{W} k(W)\,dW,$$

where W_o is an arbitrary initial quantity.

If the boundary conditions are given for the values of $W = W_o$, on the boundary of the domain, then for u we obtain boundary conditions:

$$u_o=\int_{w_o}^{W_o} k(W)\,dW.$$

Expressing W, gives the solution to the original boundary problem.

Thus, the original nonlinear problem is reduced to a Dirichlet problem, which is solved using a neural network [15]. The basis of the neural network is the relationship:

$$u_i=\frac{1}{2\pi}\sum_{k=1}^{N} c_k\left[\frac{\partial u}{\partial n}\right]_k [U]_{ik}-\frac{1}{2\pi}\sum_{k=1}^{N} c_k [u]_k\left[\frac{\partial U}{\partial n}\right]_{ik}.$$

In this expression $[U]_{ik}$ and $\left[\frac{\partial U}{\partial n}\right]_{ik}$ can be viewed as activation functions, and $c_k\left[\frac{\partial u}{\partial n}\right]_k$ and $c_k[u]_k$ as weights.

Using the least squares method and requiring the specified relationship to hold at each point of the boundary for all functions of the training set, a system of equations can be obtained for determining the weights. To improve the conditioning of this system of equations, it is necessary to increase the singularity of the quantities $[U]_{ik}$ and $\left[\frac{\partial U}{\partial n}\right]_{ik}$, shifting the contour integration a certain distance from the boundary of the domain γ.

The solution to the Dirichlet problem is sought in the form:

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

where $p(s_k)$ — is the value of the unknown function u at the boundary of the domain; $U(x, \sigma_k)$ and $V(x, \sigma_k)$ are activation functions; σ_k and τ_k are points on the closed curves γ_1 and γ_2, that cover the boundary of the domain; γ, x are points in the domain G.

The closed curves γ_1 and γ_2 are similar to the contour γ and are obtained by displacing each point in the direction of the outward normal to the boundary by distances ρ_1 and ρ_2 respectively. During the training process, the weights and values ρ_1 and ρ_2 are determined. To do this, the error functional is minimized:

$$\Pi(w_k,v_k,\rho_1,\rho_2)=\sum_{j=1}^{M}\sum_{i=1}^{N}\left\{\sum_{k=1}^{N} w_k p_k^j U(x_i,\sigma_k)+v_k p_k^j V(x,\sigma_k)-p_k^j\right\}^2$$

where x_i is the coordinate of the i-th point on the boundary contour γ; p_k^j is the boundary value of the j-th function of the training set at the point x_k.

From these relationships, for $\frac{\partial \Pi}{\partial w_m}=0$ and $\frac{\partial \Pi}{\partial v_m}=0$, m = 1,2,...N a system of linear equations is obtained to determine w_m and v_m. The value ρ_1 is determined by simple enumeration, and $\rho_2 = \rho_1+1$.

To assess the accuracy of the obtained solution, the values of u on the boundary of the domain, calculated using the neural network

$$\tilde{u}(s_i)=\sum_{k=1}^{N} w_k p(s_k)U(s_i,\sigma_k)+\sum_{k=1}^{N} v_k p(s_k)V(s_i,\sigma_k)$$

are compared with the specified boundary conditions $u(s)$.

The obtained network parameters do not provide the desired accuracy of the obtained solution. The accuracy can be increased by iterative refinement of the obtained result according to the following scheme:

$$\Delta u^0(s_i) = p(s_i),\ q^0(s_i) = p(s_i),$$

$$\Delta v^{n+1}(s_i)=\sum_{=}^{N} w_k \Delta u^n(s_k)U(s_i,\sigma_k)+\sum_{=}^{N} v_k \Delta u^n(s_k)V(s_i,\tau_k)$$

$$\Delta u^{n+1}(s_i) = \Delta u^{n+1}(s_i) - \Delta v^{n+1}(s_i),\ u_t^{n+1}(s_i) = \Delta u_t^{n+1}(s_i) - \Delta u^{n+1}(s_i),$$

where $u_t^{n+1}(s_i)$ represents the values of the refined solution at the boundary of the domain.

The process of refining the solution continues until the value

$$\frac{\left\| \Delta u^{n+1}(s_i) \right\|}{\left\| u_t^{n+1}(s_i) \right\|}$$

will not be small enough (less than the set value δ_o) or until it starts to grow. The results below are obtained at $\delta_o = 0.0025$.

To determine the value of u at any point x in the domain G use the formula:

$$\tilde{u}(x) = \sum_{k=1}^{N} w_k u_t(s_k) U(x, \sigma_k) + \sum_{k=1}^{N} v_k u_t(s_k) V(s_i, \tau_k).$$

For the training set, a set of functions that are solutions to the Laplace equation was used

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

where $k = 0, 1, 2, 3, \ldots, M$. Calculations were conducted for $M = 75$.

The specified functions were defined in various coordinate systems rotated relative to each other by angles that are multiples of $2\pi/5$.

Activation functions:

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

$$V(x, y, t, s) = \frac{\beta^7 - 21\beta^5\delta^2 + 35\beta^3\delta^4 - 7\beta\delta^6}{R^{10}} n_x + \frac{\delta^7 - 21\delta^5\beta^2 + 35\delta^3\beta^4 - 7\delta\beta^6}{R^{10}} n_y$$

where $\delta = x - t$; $\beta = y - s$; $R = \sqrt{\delta^2 + \beta^2}$; n_x; n_y are the coordinates of the outward normal vector to the boundary of the domain.

Research Results. The proposed method was applied to solve equation (1) for domains whose boundary γ was defined as

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

where a, a_1, g, g_1, ω are variable parameters.

Task 1. Consider the domain $G1$, corresponding to the parameter values: $a = 1.15$; $a_1 = 1.15$; $g = 0.05$; $g_1 = -0.05$; $\omega = 7$ (Fig. 1).

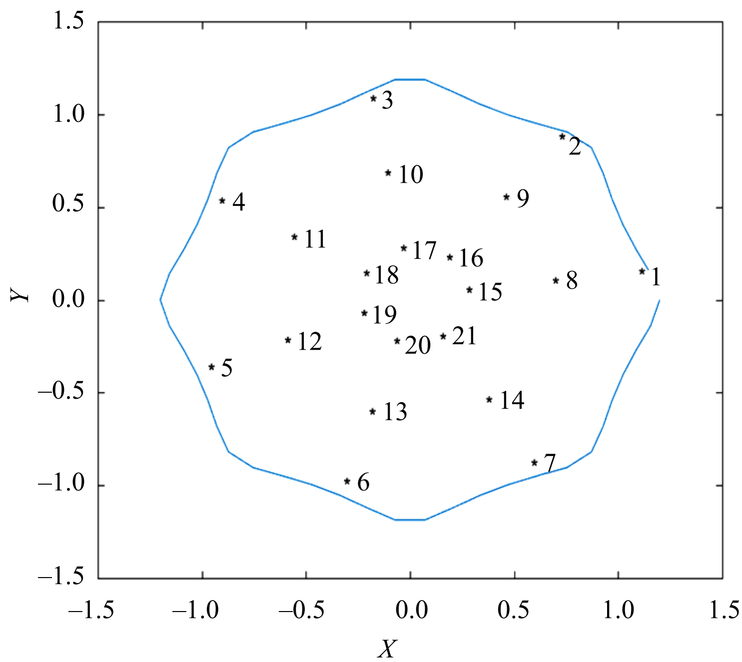

Fig. 1. Domain $G1$

Stars indicate the locations of points in domain $G1$, where the exact solution values and those obtained using the neural network for $\rho_1 = 7$, $\rho_2 = 8$ were calculated.

The equation (1) was considered for the case $k(W) = th(5W)$, which has the exact solution:

$$W_o = arch(11.25((x-1.5)^2 + (y-1.5)^2)^5.$$

The computational results are presented in Table 1.

Task 2. Let us consider domain $G2$ (Fig. 2), corresponding to parameter values $a = 1$; $a_1 = 1$; $g = 0$; $g_1 = 0.3$; $\omega = 3$; $\rho_1 = 10.51$; $\rho_2 = 11.51$.

In equation (1), $k(W) = ch(5W)$, was used, with the exact solution:

$$W_o = arsh(5e^{5x} \sin 5y) / 5.$$

The computational results for this case are presented in Table 2.

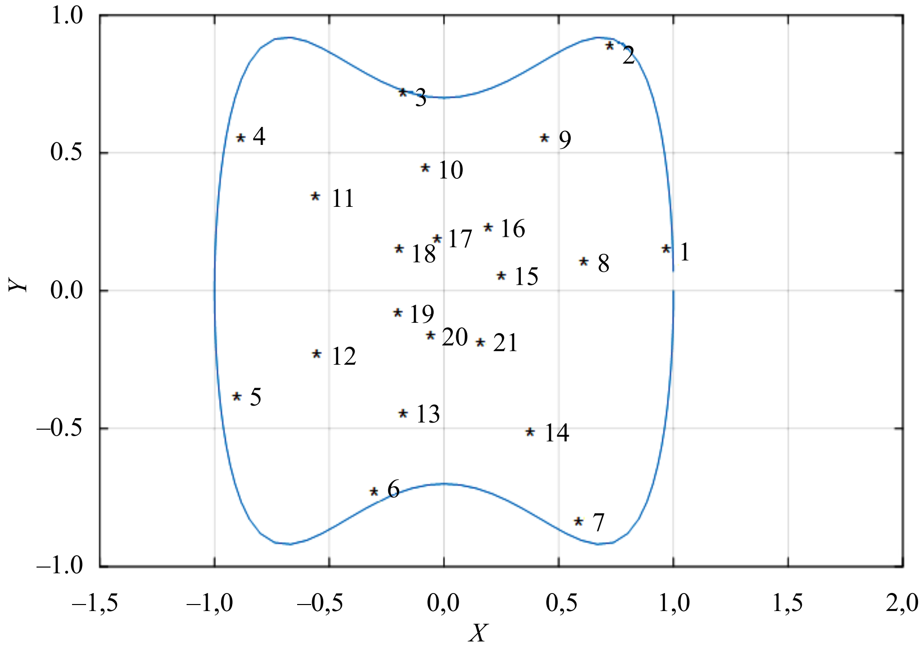

Fig. 2. Domain $G2$

Table 1

Calculation Results

Point No.	1	2	3	4	5	6	7
x	1.0534	0.6865	–0.1869	0.8848	–0.9709	–0.3193	0.5811
y	0.1275	0.8240	1.0218	0.3939	–0.3832	–0.9998	–0.9001
Exact Solution	3.0529	2.6521	2.3621	2.8550	3.3731	3.8462	4.1520
Neural Network Solution	3.0535	2.6479	2.3629	2.8491	3.3771	3.8434	4.1566
Point No.	8	9	10	11	12	13	14
x	0.6937	0.4521	0.1231	–0.5826	–0.6035	–0.1984	0.3612
y	0.0839	0.5426	0.6729	0.3252	–0.2382	–0.6215	–0.5595
Exact Solution	3.2590	3.0804	2.9504	3.1696	3.4495	3.7625	3.9935
Neural Network Solution	3.2584	3.0790	2.9487	3.1681	3.4494	3.7628	3.9947
Point No.	15	16	17	18	19	20	21
x	0.3340	0.2176	–0.0592	–0.2805	–0.2361	–0.0776	0.1413
y	0.0404	0.2612	0.3240	0.1566	–0.0932	–0.2431	–0.2189
Exact Solution	3.5104	3.4526	3.4064	3.4820	3.5887	3.7314	3.8544
Neural Network Solution	3.5100	3.4520	3.4058	3.4815	3.5885	3.7315	3.8547

Table 2

Calculation Results

Point No.	1	2	3	4	5	6	7
x	0.9085	0.6728	–0.1902	–0.8641	–0.9230	–0.3228	0.5679
y	0.1291	0.8320	0.6703	0.5138	–0.4025	–0.7484	0.8587
Exact Solution	0.3489	0.6987	0.6948	0.5361	0.3820	0.2615	0.1561
Neural Network Solution	0.3510	0.6982	0.6941	0.5346	0.3850	0.2666	0.1589
Point No.	8	9	10	11	12	13	14
x	0.5983	0.4430	–0.1253	–0.5690	–0.5737	–0.2006	0.3530
y	0.0850	0.5479	0.4414	0.3383	–0.2502	–0.4652	–0.5338
Exact Solution	0.1849	0.5233	0.5761	0.4537	0.3508	0.29214	0.1806
Neural Network Solution	0.1879	0.5231	0.5753	0.4533	0.3521	0.29578	0.1836
Point No.	15	16	17	18	19	20	21
x	0.2880	0.2133	–0.0603	0.2739	–0.2245	–0.0785	0.1381
y	0.0409	0.2638	0.2125	0.1629	–0.0979	–0.1820	–0.2088
Exact Solution	0.0585	0.2897	0.3780	0.3021	0.2534	0.2496	0.1573
Neural Network Solution	0.0607	0.2901	0.3780	0.3026	0.2544	0.2512	0.1591

Task 3. Consider equation (1) in domain $G3$ (Fig. 3).

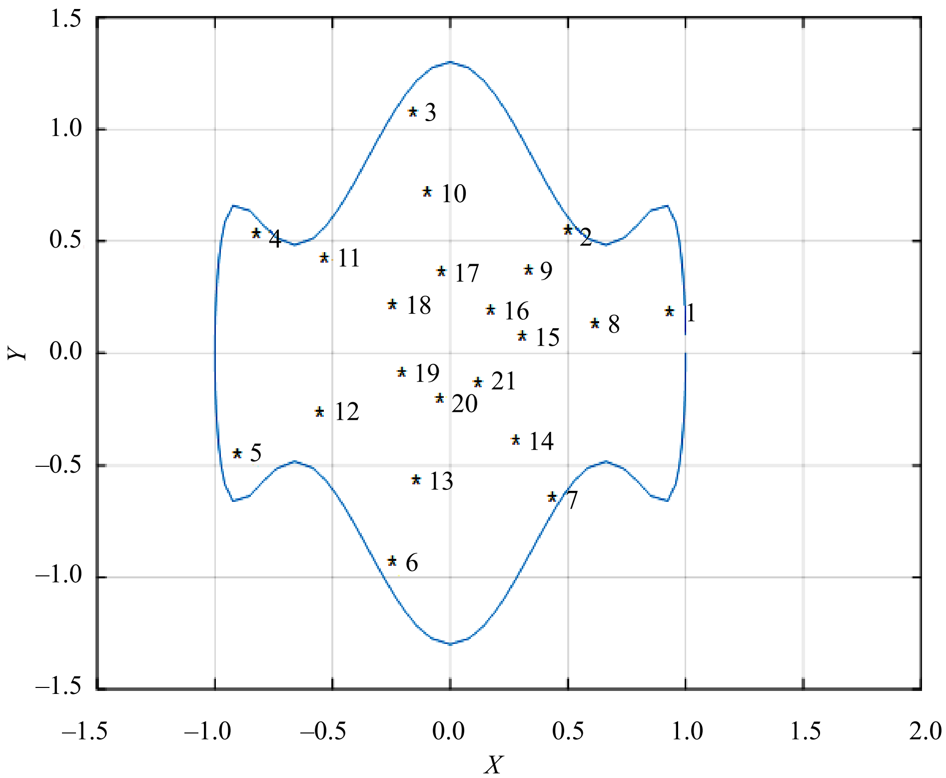

Fig. 3. Domain $G3$

For this case, the parameters are set as follows: $a = 1$; $a_1 = 1$; $g = 0$, $g_1 = 0.3$; $\omega = 5$; $\rho_1 = 11.65$; $\rho_2 = 12.65$. $K(W) = W^{1.5}$, the exact solution is given by:

$$W_o = \left\{2.5\left(\left(x^2 - y^2\right)\cos 1.5x ch1,5y + 2xy \sin 1.5x sh1.5y\right) + 25\sqrt{5}\right\}^{0.4}.$$

The results of the calculations are presented in Table 3.

Table 3

Results of Calculations

Point No.	1	2	3	4	5	6	7
x	0.9090	0.4788	–0.1752	–0.8421	–0.9238	–0.2642	0.4126
y	0.1524	0.5207	1.0491	0.5014	–0.4754	–0.9563	–0.6755
Exact Solution	4.9224	5.0814	4.9816	4.7718	4.7525	4.9738	5.0741
Neural Network Solution	5.0164	5.0845	5.0215	4.8410	4.7800	5.0122	5.1008
Point No.	8	9	10	11	12	13	14
x	0.5986	0.3153	–0.1154	–0.5546	–0.5742	–0.1642	0.2564
y	0.1004	0.3429	0.6908	0.3962	–0.2955	–0.5944	–0.4199
Exact Solution	4.9672	4.9862	4.9595	4.9035	4.8956	4.9575	4.9874
Neural Network Solution	5.0201	5.0286	5.0030	4.9507	4.9375	4.9987	5.0289
Point No.	15	16	17	18	19	20	21
x	0.2882	0.1518	–0.0555	–0.2671	–0.2246	–0.2246	0.1003
y	0.0483	0.1651	0.3326	0.1910	–0.1156	–0.2326	–0.1643
Exact Solution	4.9643	4.9616	4.9576	4.9483	4.9471	4.9575	4.9625
Neural Network Solution	5.0060	5.0049	4.9998	4.9910	4.9893	4.9988	5.0040

Fig. 4 and 5 illustrate the comparison between the exact solution of Problem 3 and the solution obtained using the neural network.

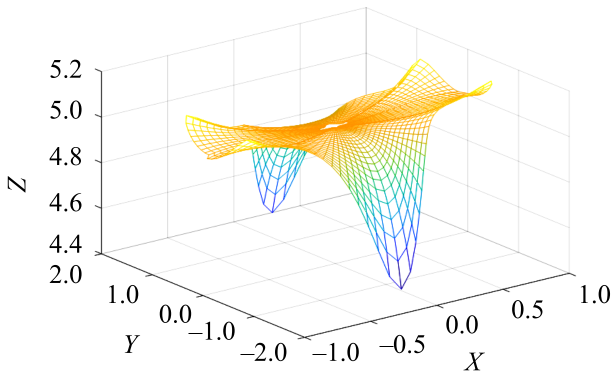

Fig. 4. Exact solution of Problem 3

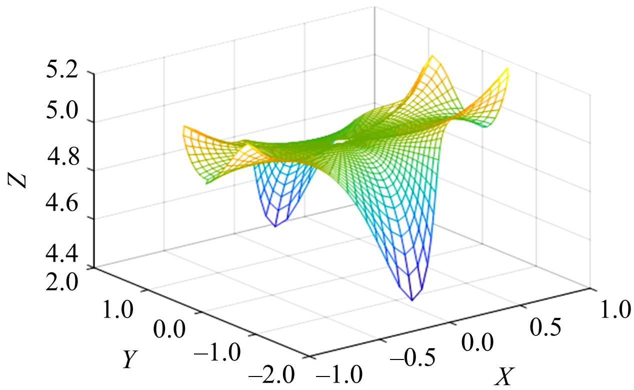

Fig. 5. Solution of Problem 3 obtained using the neural network

Discussion and Conclusion. The presented results advance the approach to solving partial differential equations using neural networks, as outlined in [15]. They convincingly demonstrate the effectiveness of the proposed method for constructing a neural network to solve boundary value problems in domains of complex shape.

This method shows significant potential, making it amenable to further development and refinement for solving a wide range of boundary value problems.

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. 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. P. 294. (In Russ.)

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

4. 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): 060801. https://doi.org/10.1115/1.4050542

5. Korsunov N.I., Lomakin A.V. Modeling processes described by a wave differential equation using cellular neural networks. *Scientific bulletins. Series History. Political science. Economy. Informatics*. 2014;15(186):103–107. (in Russ.)

6. 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.

7. 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. (In Russ.) https://doi.org/10.25559/SITITO.18.202202.310-325

8. Shevkun S.A., Samoilov N.S. Application of a neural network approach to solving direct and inverse scattering problems. *Engineering School Bulletin FEFU*. 2021;2(47):210–225. (In Russ.)

9. Gorbachenko V.I., Zhukov M.V. Solving boundary value problems of mathematical physics using radial basis networks. *Journal of Computational Mathematics and Mathematical Physics*. 2017;57(1):133–143. (In Russ.)

10. Kuznetsov K.S., Amosova E.V. Numerical solution of the system of Navier-Stokes equations in the case of a compressible medium using neural networks. *Bulletin of Tomsk State University. Management, computer technology and information science*. 2024;67:31–41. (In Russ.). https://doi.org/10.17223/19988605/67/4

11. 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

12. 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

13. Kansa E.J. Motivation for using radial basis functions to solve PDEs. URL: http://uahtitan.uah.edu/kansaweb.html (accessed: 28.11.2024).

14. 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(8):47–161.

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. (In Russ.). https://doi.org/10.23947/2587-8999-2024-8-2-68-79

16. Brebbia K., Telles J., Vroubel L. *Boundary element methods*. Moscow: Mir; 1987. 524 p. (In Russ.)

17. Аккuranov Yu.N., Mikhailov V.N. The method of boundary integral equations for solving nonlinear heat transmission problem. *USSR Comput. Maths. Math. Phys*. 1980;20:117–125.

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, 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, Galaburdin@mail.ru

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

Автор прочитал и одобрил окончательный вариант рукописи.

Received / Поступила в редакцию 12.11.2024
Reviewed / Поступила после рецензирования 03.12.2024
Accepted / Принята к публикации 07.12.2024