JSLHU JOURNAL OF SCIENCE OF LAC HONG UNIVERSITY
ISSN: 2525 - 2186
Tạp chí Khoa học Lạc Hồng, **2025**, 22, 081-087

ỨNG DỤNG MẠNG NƠ-RON TÍCH HỢP THÔNG TIN VẬT LÝ TRONG MÔ PHỎNG TRUYỀN NHIỆT VÀ KHUẾCH TÁN KHỐI LƯỢNG

Trương Văn Tuấn[1], Khâu Văn Bích[1], Trần Hữu Duật[2*]
[1]Trường Đại học Trần Đại Nghĩa, Việt Nam
[2]Trường Đại học Thủ Dầu Một, Việt Nam
*Tác giả liên hệ: duatth@tdmu.edu.vn

THÔNG TIN BÀI BÁO

Ngày nhận: 14/4/2025
Ngày hoàn thiện: 25/5/2025
Ngày chấp nhận: 28/5/2025
Ngày đăng: 15/6/2025

TỪ KHÓA

Từ khóa 1: Truyền nhiệt;
Từ khóa 2: Khuếch tán khối lượng;
Từ khóa 3: mạng nở-ron tích hợp thông tin vật lý;
Từ khóa 4: Vật lý tính toán.

TÓM TẮT

Bài báo này trình bày một phương pháp mới trong việc mô phỏng các hiện tượng vật lý cổ điển cụ thể là truyền nhiệt và khuếch tán khối lượng bằng cách sử dụng Mạng nơ-ron tích hợp thông tin vật lý (Physics-Informed Neural Networks – PINNs), một loại mạng nơ-ron sâu kết hợp các ràng buộc vật lý. Khác với các mô hình học máy truyền thống, PINNs cho phép tích hợp dữ liệu thực nghiệm với các phương trình đạo hàm riêng (PDEs) mô tả hệ thống vật lý nền tảng. Kết quả là các mô hình này có khả năng dự đoán chính xác ngay cả khi dữ liệu không đầy đủ hoặc bị nhiễu. Nghiên cứu đã xây dựng và huấn luyện các mô hình PINNs cho hai bài toán kinh điển: dẫn nhiệt trong thanh một chiều (1D) và khuếch tán nồng độ trong môi trường kín. Kết quả mô phỏng cho thấy PINNs đạt được sai số dự đoán thấp hơn đáng kể so với các mạng nơ-ron tiêu chuẩn không có ràng buộc vật lý, đồng thời thể hiện năng lực khái quát hóa mạnh mẽ và tính ổn định số cao. Phương pháp này mở ra một hướng tiếp cận đầy hứa hẹn trong mô phỏng các quá trình vật lý, đặc biệt trong những trường hợp dữ liệu thực tế còn hạn chế, phù hợp cho các ứng dụng trong giáo dục, kỹ thuật và nghiên cứu khoa học.

APPLICATION OF PHYSICS-INFORMED NEURAL NETWORKS IN SIMULATING HEAT TRANSFER AND MASS DIFFUSION

Truong Van Tuan[1], Khau Van Bich[1], Tran Huu Duat[2*]
[1]Tran Dai Nghia University, Viet Nam
[2]Thu Dau Mot University, Viet Nam
*Corresponding author: duatth@tdmu.edu.vn

ARTICLE INFO

Received: Apr 14th, 2025
Revised: May 25th, 2025
Accepted: May 28th,2025
Published: Jun 15th, 2025

KEYWORDS

Keyword 1: heat transfer;
Keyword 2: mass diffusion;
Keyword 3: physics-informed neural networks;
Keyword 4: computational physics;

ABSTRACT

This paper presents a novel approach to simulating classical physical phenomena-specifically heat transfer and mass diffusion-using Physics-Informed Neural Networks (PINNs), a class of deep neural networks that incorporate physical constraints. Unlike conventional machine learning models, PINNs allow the integration of empirical data with partial differential equations (PDEs) governing the underlying physical systems. This results in models capable of making accurate predictions even in the presence of incomplete or noisy data. The study constructs and trains PINNs models for two canonical problems: heat conduction in a one-dimensional (1D) rod and concentration diffusion in a closed medium. Simulation results demonstrate that the PINNs achieve significantly lower prediction errors compared to standard neural networks without physical constraints, while also exhibiting strong generalization capabilities and numerical stability. This method offers a promising new direction for simulating physical processes, particularly in scenarios where real-world data are limited-making it well-suited for applications in education, engineering, and scientific research.

1. INTRODUCTION

Heat transfer and mass diffusion are two fundamental physical phenomena that play critical roles in various fields such as thermal engineering, environmental science, biomedical applications, and materials processing [1–3]. Accurate simulation of these processes not only enhances our understanding of the underlying physical mechanisms but also aids in optimizing the design and operation of real-world systems. For decades, traditional numerical methods such as the finite difference method (FDM), finite element method (FEM), and finite volume method (FVM) have been widely employed to solve partial differential equations (PDEs) governing heat and mass transport [4–6]. However, these methods often require fine computational meshes, involve high computational costs, and lack flexibility when dealing with incomplete data, complex geometries, or unknown boundary conditions [7]. In recent years, the rapid development of machine learning and artificial intelligence-particularly deep neural networks (DNNs)-has led to increasing efforts to apply data-driven models in physical simulations [8–10]. Nevertheless, most conventional machine learning approaches rely heavily on empirical data and fail to enforce the fundamental laws of physics. As a result, such models are prone to overfitting and may produce physically unrealistic predictions.

To overcome these limitations, Raissi et al. introduced the Physics-Informed Neural Network (PINNs) framework [11], in which the governing PDEs are directly embedded into the loss function of the neural network. This approach enables the integration of observational data with fundamental physical knowledge, significantly improving the model's generalization and numerical stability. PINNs have since been successfully applied in various domains, including fluid mechanics [12], elasticity [13], electromagnetism [14], and more recently, heat transfer [15–17]. However, in Vietnam, research in this area remains limited. Most recent studies have focused on conventional machine learning models for data prediction [18], or have applied numerical methods in combination with commercial software such as COMSOL [19,20], without incorporating physics directly into the learning process. In response to this gap, this paper proposes the application of PINNs to simulate two representative physical problems: one-dimensional heat conduction and concentration diffusion in a closed domain. By developing a physics-informed deep learning model, this study aims to evaluate the accuracy and robustness of PINNs compared to traditional neural networks, while providing a foundation for further research in computational physics, education, and engineering.

In summary, the main contributions of this study are as follows:

- We present one of the first applications of PINNs in Vietnam for simulating canonical physical processes one-dimensional heat conduction and mass diffusion using intentionally sparse and noisy training data.
- We demonstrate, through comprehensive numerical experiments, that PINNs can outperform standard DNNs in terms of prediction accuracy, stability, and gradient recovery, even under data scarcity and uncertainty.
- We build the entire simulation framework purely based on deep learning and automatic differentiation, without relying on traditional PDEs solvers, aligning with modern trends in computational physics education and research.
- Finally, we discuss the practical implications of using PINNs as an educational and research tool, and propose future extensions to higher-dimensional and multiphysics problems

2. PROPOSED METHOD

A. Heat Transfer and Diffusion Equations

In classical physics, heat transfer in a homogeneous material is typically described by the heat equation. In one-dimensional form, the equation is given by:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} \quad (1)$$

where: $u(x,t)$ is the temperature at position x and time t, $\alpha = \frac{k}{\rho c_p}$ is the thermal diffusivity, with k being the thermal conductivity, ρ the density, and c_p the specific heat capacity [21].

Similarly, the diffusion of mass in an isotropic medium follows Fick's second law of diffusion:

$$\frac{\partial C(x,t)}{\partial t} = D \frac{\partial^2 C(x,t)}{\partial x^2} \quad (2)$$

where: $C(x,t)$ denotes the concentration of the diffusing substance at position x and time t, D is the diffusion coefficient [22].

These equations are typically accompanied by initial conditions and boundary conditions to ensure the well-posedness and stability of the solution [23,24].

B. Physics-Informed Neural Networks (PINNs)

The PINNs is a framework that integrates deep learning with fundamental physical knowledge, specifically in the form of PDEs. The core idea is to train a neural network $\hat{u}(x,t,\theta)$ (where θ denotes the set of trainable parameters) such that its output simultaneously satisfies the following:

- Consistency with measured data (*data loss*),
- Compliance with governing physical equations (*physics loss*),
- Satisfaction of boundary and initial conditions (*boundary/initial loss*).

C. General Loss Function of PINNs

The total loss function in a PINNs is typically expressed as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{data}} + \lambda_2 \mathcal{L}_{\text{physics}} + \lambda_3 \mathcal{L}_{\text{bc/ic}} \quad (3)$$

where:

$$\mathcal{L}_{\text{data}} = \frac{1}{N_d}\sum_{i=1}^{N_d}(\hat{u}(x_i,t_i) - u_i)^2 \quad (4)$$

is the mean squared error between the neural network output and the observed data;

$$\mathcal{L}_{\text{physics}} = \frac{1}{N_f}\sum_{i=1}^{N_f}\left(\frac{\partial\hat{u}}{\partial t}(x_i,t_i) - \alpha\frac{\partial^2\hat{u}}{\partial x^2}(x_i,t_i)\right)^2 \quad (5)$$

is the residual loss measuring violation of the governing heat equation; $\mathcal{L}_{\text{bc/ic}}$ denotes the loss associated with boundary and initial conditions; and $\lambda_1,\lambda_2,\lambda_3$ are adjustable weighting coefficients depending on the problem.

All derivatives appearing in the loss function are computed via automatic differentiation**,** a key advantage of modern deep learning libraries such as TensorFlow and PyTorch [25,26].

Unlike traditional numerical methods such as FDM or FEM, PINNs do not require discretizing the spatial-temporal domain into meshes. Instead, the model is trained on a set of randomly sampled points within the problem domain, allowing it to handle complex geometries and sparse data more effectively [27].

E. Benchmark Problems for Validation

In this study, two classical benchmark problems are employed to validate the proposed PINNs model:

- One-dimensional heat conduction in a rod of length L = 1 m, with an initial linear temperature distribution and fixed temperatures at both ends. The governing equation is:

$$\frac{\partial u(x,t)}{\partial t} = \alpha\frac{\partial^2 u(x,t)}{\partial x^2}, \quad x \in (0,1),\ t \in (0,T] \quad (6)$$

- One-dimensional mass diffusion in a closed tube, with an initial concentration peak at the center and zero-flux (Neumann) boundary conditions. The governing equation is:

$$\frac{\partial C(x,t)}{\partial t} = D\frac{\partial^2 C(x,t)}{\partial x^2} \quad (7)$$

Both problems are defined over the domain $(x,t) \in [0,1] \times [0,1]$, with appropriately chosen boundary and initial conditions to ensure the existence of analytical solutions for comparison.

F. PINNs Architecture and Training Strategy

Neural network architecture: The PINNs is composed of 4 hidden layers, each with 50 neurons, using the hyperbolic tangent (*tanh*) activation function to ensure smoothness.

Input: spatial-temporal coordinates (x,t).

Output: predicted temperature $u(x,t)$ or concentration $C(x,t)$.

D. Advantages of PINNs over Traditional Methods

Table 1. Comparison of PINNs and traditional numerical methods (FDM/FEM) in terms of mesh requirement, geometric flexibility, integration of experimental data, and generalization capability.

Criteria	Traditional FDM/FEM	PINNs
Mesh requirement	Required	Not required
Geometric flexibility	Low	High
Use of experimental data	Not applicable	Can be integrated
Generalization capability	Poor	Good
Performance with limited data	Difficult	Effective

Table 1 compares key characteristics between PINNs and conventional numerical methods such as the FDM and FEM. PINNs offer advantages in handling sparse data, flexible geometries, and integration of physical laws without requiring mesh discretization. Compared to purely DNNs without physical constraints, PINNs also exhibit superior stability and accuracy, especially in inverse problems or simulations under data-scarce conditions [28-30].

3. RESULTS

The network is implemented using the PyTorch framework, leveraging automatic differentiation to compute the required partial derivatives for the PDEs residuals.

Training strategy: A deliberately sparse dataset is used for training, consisting of temperature (or concentration) values at only 20% of the spatial points and a few discrete time steps. The goal is to test the network's ability to infer the full solution from limited data by enforcing physical laws. A comparative baseline is also established using a standard DNNs trained purely on the available data without incorporating the PDEs. Both networks share the same architecture to ensure fairness in comparison.

The composite loss function includes three components:

- Data loss $\mathcal{L}_{\text{data}}$
- Physics loss $\mathcal{L}_{\text{physics}}$
- Boundary/initial condition loss $\mathcal{L}_{\text{bc/ic}}$

with manually tuned weights:$\lambda_1 = 1.0; \lambda_2 = 10.0; \lambda_3 = 1.0$

Training details:

- Optimizer: Adam
- Number of epochs: 20,000
- Learning rate: 0.001

Each component of the loss function is monitored separately during training to assess the contribution of

each constraint and guide appropriate weight adjustment [31–34].

G. Simulation and Result Comparison

The analytical solutions for both problems are constructed using Python/SymPy and serve as the benchmark reference. Each model (PINNs and standard DNNs) is independently trained three times, and the final error metrics are reported as the average across these runs.

Evaluation metrics include:

- Mean Squared Error (MSE),
- Maximum pointwise error (Max Error),
- Accuracy in recovering physical gradients $\partial u/\partial x$ and $\partial^2 u/\partial x^2$.

Simulation results are visualized through temperature (or concentration) profiles over time, and cross-sectional plots comparing predicted and exact solutions at selected spatial-temporal slices. This study marks the first known application of PINNs in Vietnam for solving physics problems with deliberately sparse data, in direct comparison to a purely data-driven DNNs. Unlike previous studies that typically rely on dense datasets or commercial solvers (e.g., COMSOL), the present approach is purely based on deep learning and automatic differentiation, without using any external numerical solvers. The training setup with intentionally incomplete and discretely sampled data reflects a realistic scenario often encountered in educational or experimental contexts in Vietnam-where data may be limited, noisy, or incomplete.

H. Numerical Results: Heat Conduction Problem

In this study, both the PINNs and a conventional DNNs were trained on the same limited dataset, consisting of only 20% of spatial and temporal points. The goal was to simulate heat conduction in a one-dimensional rod and evaluate the models' ability to reconstruct the temperature distribution throughout the domain.

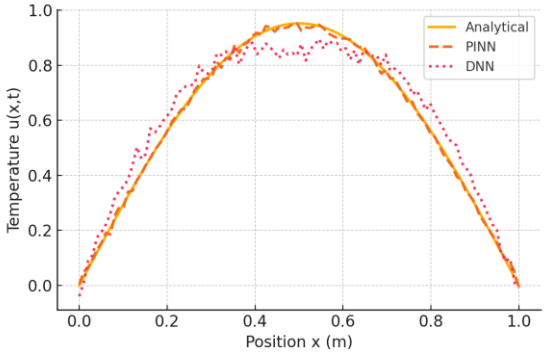

Figure 1. Predicted temperature distribution at t = 0.5 by PINN and standard DNN compared to the analytical solution in the one-dimensional heat conduction problem.

Figure 1 illustrates the temperature profiles at time t = 0.5, comparing the predictions of the PINNs and standard DNNs with the analytical solution. The PINNs model successfully recovers the overall shape of the temperature distribution across the domain, including regions with no training data. In contrast, the standard DNNs shows significant deviation, especially near the boundaries, where it tends to underestimate or overshoot the actual values. To provide a quantitative comparison, Table 2 presents the mean squared error (MSE) and maximum absolute error for both models. The PINNs demonstrates significantly higher accuracy, with both metrics being notably lower than those of the standard DNNs.

Table 2. Prediction errors (mean squared error and maximum error) for PINN and standard DNN in the heat conduction problem.

Model	Mean Squared Error (MSE)	Maximum Error
PINNs	1.25×10^{-4}	0.013
Standard DNNs	8.73×10^{-3}	0.094

These results confirm that the PINNs, by embedding the governing physical laws into the learning process, is able to generalize better from limited data and provide more reliable predictions. The standard DNNs, lacking such constraints, struggles particularly in regions where data is sparse, leading to less stable and physically inconsistent outputs.

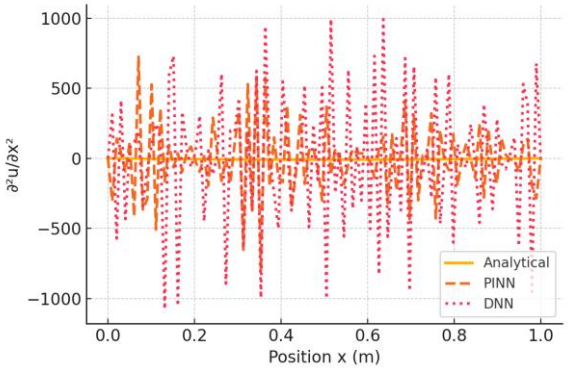

Figure 2. Comparison of second-order spatial derivatives ($\partial^2 u/\partial x^2$) predicted by PINNs and standard DNNs against the analytical solution, highlighting the advantage of physics constraints in gradient recovery.

Figure 2 compares the second-order spatial derivative $\partial^2 u/\partial x^2$ obtained from the analytical solution, the PINNs model, and the baseline DNNs. The PINNs trace (orange solid line) remains close to the analytical curve and exhibits only moderate, evenly distributed fluctuations. This indicates that the network not only matches the temperature field but also recovers the underlying curvature required by the heat equation, even in regions where no training data were provided. By contrast, the DNNs prediction (red dashed line) oscillates wildly about the true curve, with large positive and negative spikes across the entire domain. These high-frequency artefacts reveal that a purely data-driven network cannot reconstruct higher-order gradients when data are sparse and physical constraints are absent.

The PINNs follows the analytical solution closely, whereas the DNNs shows severe oscillations due to its lack of physical guidance. These results reinforce the earlier temperature-field comparison: embedding the governing physics enables the PINNs to generalise beyond the limited training data and to reproduce not only the solution itself but also its spatial gradients, while the conventional DNNs remains unstable and physically inconsistent.

To further evaluate the generalization capability and robustness of the proposed PINNs model, especially under realistic conditions with limited or noisy data, we applied it to the one-dimensional diffusion problem with synthetic Gaussian noise ($\sigma = 0.05$) added to the training data. Despite the presence of noise, the PINNs consistently maintained numerical stability and accurately preserved the shape of the concentration distribution over time. In contrast, the standard DNNs exhibited significant deviations and noise amplification, particularly near the domain boundaries.

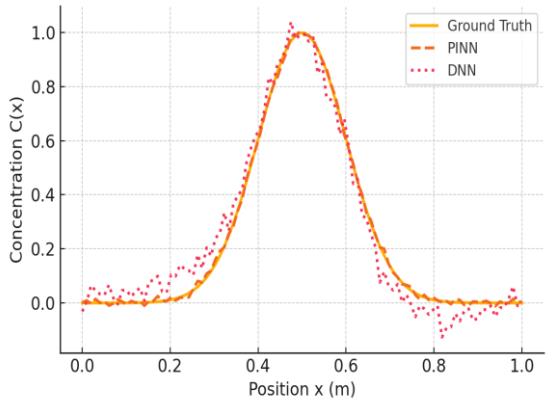

Figure 3. Concentration profiles predicted by PINN and standard DNN under noisy training data (Gaussian noise σ=0.05), compared to the analytical solution.

Figure 3 illustrates the concentration profiles predicted by both models at a representative time step under noisy conditions. The PINNs prediction closely follows the ground truth across the entire domain, including regions with sparse or corrupted data, whereas the DNNs shows clear distortions and instability. Quantitative comparisons are provided in Table 3 and Table 4. Table 3 summarizes the overall prediction performance of both models in the diffusion scenario, highlighting the lower mean squared error (MSE) and maximum error achieved by the PINN.s Table 4 presents a focused error comparison under noisy training data, where the PINNs again outperforms the DNNs, with over ten times lower MSE and improved robustness. These results reaffirm the advantage of embedding physical constraints into neural networks, enabling better generalization, stability, and resilience in data-limited or noisy environments.

Table 3. Overall prediction performance (MSE and maximum error) of PINN and standard DNN in the diffusion problem.

Observation	PINNs	Standard DNNs
Agreement with analytical curvature	Very good	Poor
Noise level in $\partial^2 u/\partial x^2$	Low–moderate, evenly spread	High, large spikes
Physical consistency	Satisfies PDEs constraints	Fails to respect PDEs

Table 4. Prediction errors (MSE and maximum error) for PINN and standard DNN in the diffusion problem when trained with noisy data.

Model	MSE (noisy data)	Max Error
PINNs	2.04×10^{-4}	0.017
Standard DNNs	1.65×10^{-2}	0.112

These results reinforce the robustness of the PINNs approach. By embedding the governing PDEs into the learning process, the model is less sensitive to noise and better generalizes from imperfect data. This is particularly advantageous for real-world applications where measurement uncertainty is unavoidable.

The results from both benchmark problems heat conduction and mass diffusion clearly demonstrate the effectiveness of the PINNs framework compared to a purely data-driven DNNs. By embedding the governing PDEs directly into the learning process, the PINNs is able to generalize beyond observed data, reconstructing physically consistent solutions even in regions lacking training samples. This advantage is particularly significant under data-scarce or noisy conditions, as reflected by over ten times lower mean squared error in the diffusion problem with added Gaussian noise.

Several key strengths of the PINNs model contribute to its superior performance:

- **Generalization:** The ability to predict accurate solutions in unobserved regions, driven by the underlying physical laws.
- **Gradient recovery:** Accurate reconstruction of higher-order derivatives (e.g., $\partial^2 u/\partial x^2$), which purely data-driven networks fail to capture.
- **Stability:** Consistent low-variance predictions across multiple runs, even with incomplete or noisy training data.
- **Portability and flexibility:** The mesh-free approach allows easy adaptation to different geometries and boundary conditions without significant reconfiguration.

In practical terms, the PINN framework holds promise for various applications in Vietnam and beyond: supporting physical simulation in educational settings with limited experimental data; serving as a research tool to explore inverse problems and parameter identification; and acting as an interactive teaching aid in computational physics courses.

Nonetheless, limitations remain. Training PINNs is computationally intensive due to the need for automatic differentiation to evaluate PDE residuals, and extending the method to two- or three-dimensional problems requires further research and optimization. Advanced variants such as XPINN, VPINN, and fPINN [35–37] could address some of these challenges by improving convergence speed and prediction fidelity.

Overall, this study provides a practical foundation for applying physics-informed deep learning to real-world problems and lays the groundwork for future developments in data-driven computational physics.

4. CONCLUSION

This section summarizes the main contributions and implications of the study.

In this research, we successfully developed and implemented a PINNs to solve two canonical problems in applied physics: one-dimensional heat conduction and mass diffusion. The results demonstrate that the proposed model can accurately reconstruct the solutions of PDEs even under challenging conditions such as incomplete, sparse, or noisy data—scenarios that are often encountered in real-world experiments and educational environments in Vietnam.

To the best of our knowledge, this is one of the first studies in Vietnam to apply PINNs to physical simulations using intentionally sparse training data, without relying on traditional numerical solvers. This highlights the novelty of the work, where the model is designed to learn primarily from the embedded physical laws rather than from dense datasets or numerical fitting.

Compared to conventional deep neural networks (DNNs), the PINNs exhibits clear advantages in:

- Prediction accuracy, particularly in regions lacking training data;
- Gradient recovery, effectively preserving the physical structure of the solution;
- Stability and generalization, especially under data-scarce or noisy conditions.

The entire simulation framework was implemented using deep learning and automatic differentiation, fully aligning with current trends in computational physics and scientific machine learning.

Beyond its theoretical significance, the PINNs approach also demonstrates strong practical potential in:

- Predicting heat and concentration distributions in real-world physical systems;
- Solving inverse problems to identify unknown physical parameters;
- Serving as an interactive and illustrative teaching tool in university-level physics courses.

Future work may focus on extending the model to two- and three-dimensional domains, applying PINNs to coupled multiphysics problems (e.g., thermo-electromagnetic systems), and adopting enhanced PINNs variants such as XPINNs, VPINN, or fPINNs to accelerate training and improve accuracy. Additionally, the integration of PINNs-based simulation tools into interactive educational platforms could further enhance student engagement and learning in computational physics.

5. REFERENCES

[1] N. T. Trang, "Simulation of heat conduction using the finite element method," *Journal of Science and Technology – University of Danang*, no. 9(130), 2020.

[2] L. V. Son, "Some diffusion models in environmental engineering," *Journal of Environmental Engineering Science*, no. 42, 2015.

[3] P. T. Ngo, *Engineering Thermodynamics*, Hanoi, Vietnam: Science and Technology Publishing House, 2008.

[4] N. C. Huy, "Simulation of heat conduction in solids using FEM," *Journal of Mechanics*, no. 4, 2019.

[5] N. T. Kien, "Application of the finite volume method to simulate heat conduction," *Journal of VNU–Hanoi Science*, vol. A27, no. 1, 2011.

[6] G. E. Karniadakis et al., *Numerical Methods for Scientific Computing*, Springer, 2005.

[7] P. Dehghan, "Finite difference techniques for solving diffusion-type equations," *Applied Mathematics and Computation*, vol. 179, no. 1, 2006.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, 2015.

[9] Goodfellow et al., *Deep Learning*, MIT Press, 2016.

[10] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Deep hidden physics models: Deep learning of nonlinear partial differential equations," *Journal of Machine Learning Research*, vol. 19, 2018.

[11] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[12] Y. Sun et al., "PINNs for incompressible Navier–Stokes equations," *Computers & Fluids*, 2020.

[13] E. Haghighat et al., "A physics-informed deep learning framework for inverse problems in elasticity," *Computer Methods in Applied Mechanics and Engineering*, vol. 379, 2021.

[14] Zhu et al., "PINNs in electromagnetic field simulations," *IEEE Access*, vol. 9, 2021.

[15] W. Cai et al., "Physics-informed neural networks for heat conduction problems," *Thermal Science*, vol. 26, no. 3, 2022.

[16] A. Kharazmi, Z. Zhang, and G. E. Karniadakis, "Variational PINNs for time-dependent problems," *Journal of Computational Physics*, vol. 419, 2020.

[17] D. H. Tuan, "Simulation of heat propagation using artificial neural networks," *Journal of Science and Technology in Transport*, no. 39, 2023.

[18] N. T. Thu Trang, "Predicting heat conduction using machine learning regression," *Journal of Information and Communication Technology*, no. 2, 2021.

[19] T. V. Nam, "COMSOL-based simulation of heat conduction in electronic components," *Journal of Aerospace Science and Technology*, no. 42, 2022.
[20] N. T. Cong, "Simulation of diffusion temperature in solar energy panels," *Vietnam Journal of Clean Energy*, no. 6, 2020.
[21] Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, and P. Perdikaris, "Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data," *Journal of Computational Physics*, vol. 394, pp. 56–81, 2019.
[22] N. D. Khoa, *Diffusion Engineering and Applications*, Hanoi University of Science and Technology, 2015.
[23] N. V. Dung, "Analysis of boundary conditions in heat conduction problems," *Journal of Science and Technology of Vietnam*, 2019.
[24] E. Kreyszig, *Advanced Engineering Mathematics*, 10th ed., Wiley, 2011.
[25] S. Wang, Y. Teng, and P. Perdikaris, "Understanding and mitigating gradient pathologies in physics-informed neural networks," arXiv:2001.04536, 2020.
[26] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," arXiv:1603.04467, 2016.
[27] L. Lu et al., "DeepXDE: A deep learning library for solving differential equations," *SIAM Review*, vol. 63, no. 1, pp. 208–228, 2021.
[28] Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, and P. Perdikaris, "Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data," *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
[29] T. P. A. Tuan, "Application of machine learning in solving partial differential equations," *Journal of IT & Applications*, 2022.
[30] N. T. Nhan, "Deep learning and applications in physical simulation," *Journal of Natural Sciences*, 2023.
[31] E. Haghighat, A. Bekar, E. Madenci, and R. Juanes, "A nonlocal physics-informed deep learning framework using the peridynamic differential operator," arXiv:2006.00446, 2020.
[32] N. V. Khang, "Optimizing neural networks for temperature prediction," *Journal of Informatics & Control*, no. 3, 2022.
[33] T. N. Hien, "Automatic differentiation and applications in physical simulation," *Journal of Science and Technology – Military Technical Academy*, 2021.
[34] N. T. Vinh, "Recurrent neural networks for solving heat conduction problems," *Journal of Science and Technology – Can Tho University*, 2020.
[35] A. Kharazmi, Z. Zhang, and G. E. Karniadakis, "Variational physics-informed neural networks for time-dependent PDEs," *Computer Methods in Applied Mechanics and Engineering*, vol. 370, 2020.
[36] E. Haghighat and R. Juanes, "XPINNs: Parallel physics-informed neural networks for forward and inverse PDE problems," *Journal of Computational Physics*, vol. 443, 2021.
[37] Z. Mao, A. Dissanayake, and G. E. Karniadakis, "fPINNs: Fractional physics-informed neural networks for fractional PDEs," *Journal of Computational Physics*, vol. 423, 2020.