Deep Learning-Based PDE Solver: PINN Versus Classical Method for the 1D Heat Equation

Salah Uddin*, Ehsan Ellahi Ashraf†, Mahathir Mohamad‡, Muhammad Ahmad Hafeez§, Imran Ihsan**

Abstract

Partial Differential Equations (PDEs) play a vital role in modeling heat transfer and diffusion phenomena in science and engineering, ensuring the development of accurate and efficient numerical solvers for consistent simulations. Conventional discretization-based procedures often involve mesh production for regular geometries and frequently encounter complications for irregular cases and sparse data. Currently, Machine Learning approaches, specifically Physics-Informed Neural Networks (PINNs), have emerged as capable mesh-free alternatives that integrate governing physical rules directly into the learning method. In this paper, PINNs have been applied to a One-Dimensional (1D) heat equation. The PINN model has been formulated by using DeepXDE and compared with the Finite Difference Method (FDM). The governing equation (GE), along with initial and boundary conditions, is rooted in the loss function, which monitors the training process, allowing the PINN to achieve physical stability by learning the result over the entire spatio-temporal field. The application influences programmed differentiation for the resulting derivatives and reduces the residual error, rather than relying on explicit discretization. Results suggest that the neural networks successfully approximate the solution to the heat equation with competing error rates. Moreover, it is flexible for noisy data and complex domains. Comparative convergence behavior and visualization results are presented to demonstrate the effectiveness of the PINN framework.

Keywords: Physics-Informed Neural Networks, Finite Difference Method, DeepXDE, Partial Differential Equations.

Introduction

Partial Differential Equations (PDEs) are used as a basic mathematical instrument in scientific computations and serve as a

---

*Corresponding Author: Department of Humanities & Sciences, College of Aeronautical Engineering, National University of Sciences & Technology, Risalpur 23200, Pakistan, sudin@cae.nust.edu.pk

†Department of Humanities & Sciences, College of Aeronautical Engineering, National University of Sciences & Technology, Risalpur 23200, Pakistan, ehsansag@cae.nust.edu.pk

‡Department of Mathematics & Statistics, Universiti Tun Hussein Onn Malaysia, Pagoh Educational Hub, Panchor 84600, Johor, Malaysia, mahathir@uthm.edu.my

§Department of Humanities & Sciences, College of Aeronautical Engineering, National University of Sciences & Technology, Risalpur 23200, Pakistan, ahmadhafeez.edu@gmail.com

** Department of Creative Technologies, Air University, Islamabad 44000, Pakistan, imranihsan@au.edu.pk

backbone to model physical systems that frequently change over space and time in engineering. In addition to the Artificial Intelligence (AI) in mathematics, Physics-informed neural networks (PINNs) have been developed and have gained a practical and transformative computational approach with Deep Learning and a robust background for solving PDEs, along with other artificial neural networks.

Traditional numerical methods for explaining time-dependent diffusion equations normally depend on mesh schemes like Finite Difference Method (FDM), Finite Element Method (FEM), and Finite Volume Method (FVM). These methods convert continuous differential operators into algebraic structures or unstructured discretization. Even though such methods are numerically well-known and recognized in the engineering domain, their presentation is frequently faced with mesh resolution, stability criteria, and time-stepping restrictions. Moreover, mesh generation and improvement could develop costly computations for the complicated geometry or complex boundary conditions, thus inspiring the exploration for alternate grid-free computational models.

When solving PDEs analytically, complex equations with a nonlinear nature typically lack a general solution, while numerical methods are faced with computational issues, precision constraints, and potential variability (Blechschmidt & Ernst, 2021). Although exact solutions are guaranteed, analytical methods are frequently reasonable for a narrow range of problems; however, numerical methods deliver approximate results that are applied for complex, real-life problems, but present space for errors and involve major computational power in addition to high accuracy (Alshanti et al., 2023; Raman, 2024; El-Metwaly & Kamal, 2024; Gürbüz & Fernandez, 2024; Lima et al., 2024; Wei, 2025).

PINNs resolve these complications by inserting the governing differential equation right into the training phase of a neural network. Despite depending on a discrete mesh, the result is estimated by a continuous function operated by the neural networks, and DeepXDE is employed for automatic spatial and temporal differentiation. This approach imposes the residual of the differential equation, initial and boundary conditions instantaneously, leading the model to train on the physically reliable solution over the complete domain. Subsequently, the method associates data-driven approximations with basic conservation laws, linking the breach between Machine Learning and conventional scientific computing.

Regardless of the rapid advancement of PINNs, many studies are conducted mainly on complex multi-dimensional standards or highly dedicated applications, which may obscure the major performance of the method. Organized estimation on canonical problems with established features remains necessary to recognize convergence properties, computational proficiency, and numerical strength. The One-Dimensional (1D) heat equation offers a perfect model for this situation because it holds well-defined physical understanding, well-known reference solutions, and undeviating numerical standards for comparison.

Even though PINNs are extensively applied in PDEs (Alianakh et al., 2025; Almusallam et al., 2025; Chang et al., 2024; Guo et al., 2020; Hu et al., 2022; Jiang et al., 2023; Katsikis et al., 2022; Luo et al., 2025; Maczuga et al., 2025; Tanyu et al., 2023; Oh & Jo, 2025; Zhang, 2025), their enactment on simple scale problems is quiet and obviously not understood. In fact, one-dimensional heat equations have been usually studied, but related investigations remain scattered. Existing studies mostly emphasize innovative simulations despite fundamental clarity and reproducibility. Conventional numerical methods, like the FDM, are well-structured; however, relative studies with deep neural networks are limited. Consequently, the comparative study will fill a significant research gap.

To overcome these problems, this study performs a computational assessment between a PINN solution and a classical FDM solver under identical numerical conditions. The key findings of this computational study include: (a) formation of a PINN coding tailored to the 1D heat equation; (b) application of a conventional FDM baseline for reasonable benchmarking; (c) numerical assessment using error norms, convergence performance, and computational rate; (d) investigation of compensations and confines of both methods for applied scientific computing; (e) demonstration of the viability of Machine Learning based solvers for low-dimensional heat models.

The rest of this study is organized as follows. The first section describes the mathematical model of the 1D heat equation and the initial and boundary conditions. The second section introduces the architecture and training model of the proposed PINN. The third section describes the FDM scheme. The fourth section explains experimental results with graphical illustrations and comparative analysis, followed by concluding remarks and future research directions in the final section.

Mathematical Model

Governing Equation (GE)

In Equation 1, the 1D PDE represents the heat conduction model, mathematically denoted as,

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}. \tag{1}$$

Here, $u(x, t)$ represents the temperature at $x \in [0, 1]$ and time $t \in [0, 1]$, with thermal diffusivity α.

Initial and Boundary Conditions (ICs and BCs)

To ensure a well-posed result of the 1D heat equation, suitable ICs and BCs are described. The IC in Equation 2 states the temperature distribution along the rod at time t = 0, whereas the BCs in Equation 3 define the temperature distribution at the spatial boundaries. In the present work, Dirichlet BCs are enforced at split ends of the field, allowing both PINN and FDM models to study and fulfil the physical constraints of the 1D heat model.

$$u(x, 0) = \sin(\pi x). \quad (2)$$

$$u(0, t) = 0, u(1, t) = 0. \quad (3)$$

Analytical Solution (AS)

For authentication, the analytical solution of the 1D heat equation, as given in Equation 4, is achieved by applying the separation of variables method. This method produces the spatial and temporal measures of the solution, causing a series form equation that fulfils the heat equation and the given ICs and BCs. The solution functions as a standard to evaluate the correctness and convergence of the suggested PINN and the traditional FDM.

$$u(x, t) = e^{-\pi^2 t} \sin(\pi x). \quad (4)$$

Physics-Informed Neural Networks (PINNs)

Before defining the PINNs architecture, physics-based data set, which will be trained on time-dependent PDE, geometry and time domain, heat equation, initial and boundary conditions, along with the collocation points, are provided.

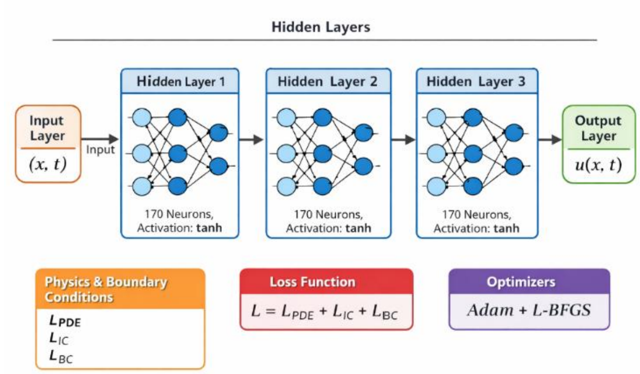

Figure 1: PINN Architecture for the 1D heat equation.

A fully connected feed-forward neural networks, has been model the heat equation solution. The architecture composed of five layers with two inputs layer, three further hidden layers of 170 neurons each and finally the output layer. Moreover, the network is trained with an activation function tanh, using Adam (lr = 0.001) initially for 2000 epochs and then developed with L-BFGS. Furthermore, predictions are then computed on 4000 test points sampled from the geometry-time domain.

Loss Function (LF)

The loss function demonstrates how sound the PINN learn the governing equations along with the ICs and BCs. It is usually framed as a weighted sum of the PDE residual loss and the constraint losses. Decreasing this loss imposes the physical laws while training the NN.

$$L = L_{PDE} + L_{IC} + L_{BCL}. \quad (5)$$

Automatic Differentiation (AD)

Automatic differentiation performs the spatial and temporal differentiation by taking derivatives of the NN output with respect to inputs. Which are obligatory to assess the heat equation residual without numerical approximation. This enables the exact computation of gradients during training. DeepXDE computes u_t, u_x, u_{xx} using TensorFlow autograd.

Classical Numerical Method (Finite Difference)

The FDM addresses the corresponding 1D heat equation by discretizing time and space using points and approximating derivatives with difference formulas. As a result, a system of algebraic equations is obtained which allow us to solve PDE iteratively. Describe FDM scheme as:

$$u_i^{n+1} = u_i^n + r(u_{i+1}^n - 2u_i^n + u_{i-1}^n),\ \ r = \alpha \frac{\Delta t}{\Delta x^2}. \quad (6)$$

This equation is implemented during the coding through the NumPy library in Python.

Results and Discussion

For the one-dimensional heat equation, both of the optimizers nearly overlap and quickly converged. Moreover, the loss drop indicates that PDE along with the initial and boundary conditions are satisfied by the PINN with high accuracy. In the PINN training phase, boundary conditions are integrated directly through the LF rather than being prescribed explicitly as in traditional numerical methods like FDM. In fact, Adam, the first optimizer, trains the model very close to the solution, and leaves slight refinements for the second L-BFGS optimizer.

Loss Curve

The loss curve in Figure 2 shows the progress of the total LF throughout the training session of the PINN. It illustrates how well the NN reduces the PDE residual, ICs, and BCs errors over time. A gradually reducing and convergent behavior shows steady training and enhanced model accuracy.

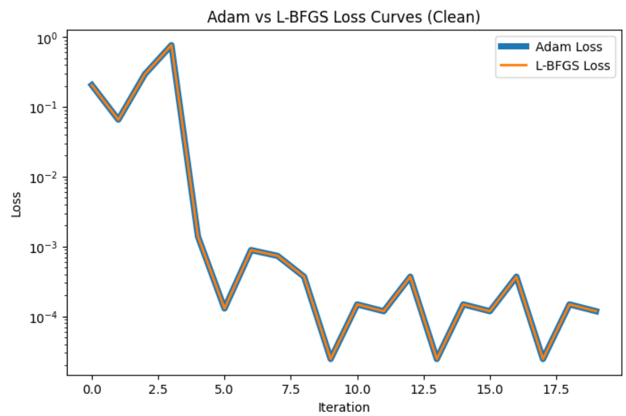

Figure 2: Adam versus L-BFGS iterative losses.

Predicted Versus Analytical Solution

The scatter plot of the true solution in Figure 3, in space and time indicates the magnitude from $t = 0\ to\ t = 1$. Dissipative dynamics far from the thermodynamic equilibrium have been observed, which is larger near $t = 0$, than gradually decreasing to zero near $t = 0.5$. Moreover, rapid decay has been observed near $t \approx 0.3 - 0.5$, causing PINN training difficulties due to stiffness.

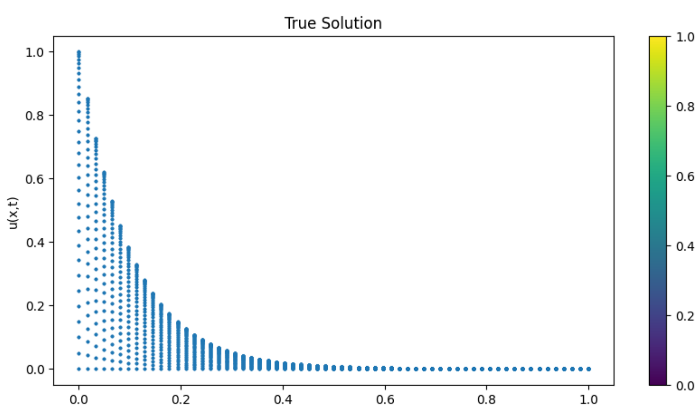

Figure 3: True solution versus time t.

PINN-predicted solution in Figure 4, captures the same decay pattern previously observed in the true solution. However, the reproduced plot shows the qualitative time decay pattern with small differences.

The absolute error plot in Figure 5, visualizes the differences between the solutions. A very small error across the heat equation domain indicates accurate PINN learning.

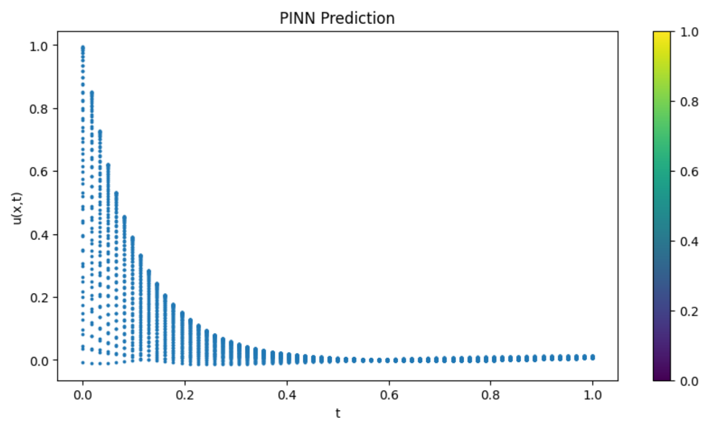

Figure 4: PINN prediction versus time t.

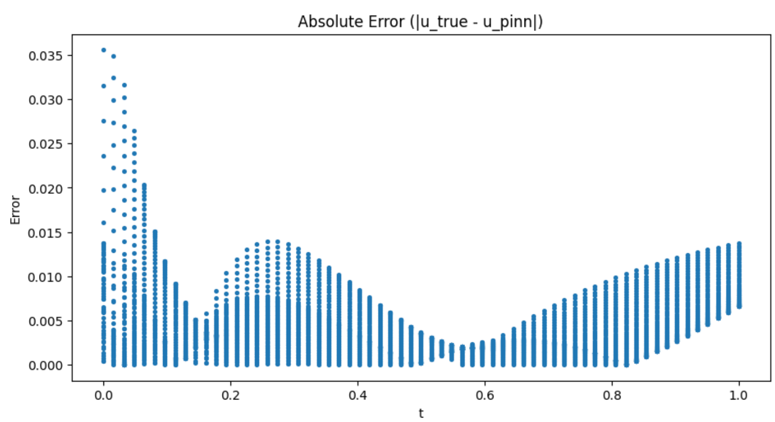

Figure 5: Absolute Error.

In Figure 6, the PINN solution is compared with the True and FDM solutions. PINN learn to reproduce the exact overall shape of the true solution; however, it miscalculates the true values frequently showing the gradual bias. Moreover, misreading indicates that PINN learned the qualitative aspect but lacks quantitative behavior.

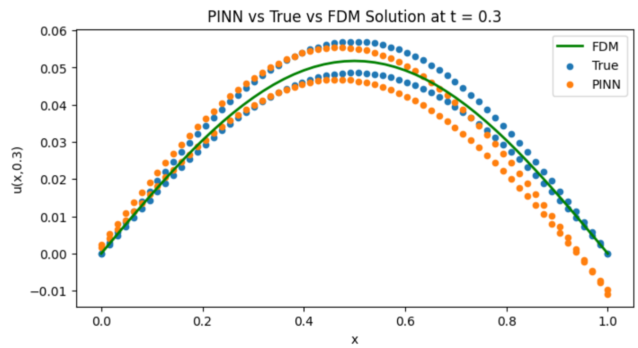

Figure 6: PINN versus FDM versus FDM solution at t =0.3.

FDM is computationally capable for conventional forward problems defined on regular grids. Its scaling is linear with the number of spatial and temporal grid points, performed through classical matrix operations. Consequently, for large-scale well-posed problems with specific meshes, it generally leaves PINNs behind in terms of time and memory consumption. On the other hand, PINNs need training on intense neural networks by decreasing a composite loss function that comprises PDE residual, boundary conditions, and initial conditions. This exercise phase contains frequent estimations of DeepXDE and iterative optimizers, making it computationally more expensive, as the number of collocation points increases. Consequently, for larger problem sizes, the computational cost of PINNs grows significantly compared to FDM.

Disparate traditional mesh-based NM, such as the FDM, which involve spatial and temporal discretization, PINNs learn the solution right from the GE in a mesh-free mode. As compared to the foundation work on PINN by Raissi et al. (2019), the current effort incorporates AD to precisely impose the heat equation residual, though satisfying the ICs and BCs. In association with the previous work that mainly established PINNs or standard solvers individually, this work offers an organized comparison between PINN and FDM. Related works have studied PINN-based learning for mechanics problems (Dalton et al., 2023), while the projected method precisely marks the heat equation and highlights procedural authentication. These differences provide the strength and practical use of the suggested work.

Conclusion

This study examined the efficiency of Machine Learning based solver for PDEs over a systematic comparison between a PINNs framework and a classical FDM scheme for the 1D heat equation. Both methods were applied under similar conditions to assess their mathematical behavior, approximation ability, and computational efficiency. PINNs are the best alternative to classical methods. Highly accurate PDE approximations can be made possible by using PINN. It has been demonstrated that the neural networks resolution carefully shadows the analytical and numerical readings through the complete spatio-temporal domain, attaining small residual and prediction slips while preserving smooth effect profiles. Moreover, the convergence features show that surrounding the leading physical constraints inside the loss function permits steady learning instead of explicit mesh discretization. Due to its flexible nature, it is more suitable for irregular or complex geometries. Future work can be extended to the 2-dimensional PDEs, improving training through adaptive loss weighting, and exploring enhanced or extended PINN frameworks with better accuracy and stability.

References

Alianakh, D., Dyyak, I., & Makar, I. (2025). Development of physics-informed neural networks for solving partial differential equations. *In Proceedings of the International Workshop on Computer Modeling and Intelligent Systems*.

Almusallam, N., Al Nuaimi, B. T., Alkattan, H., Abotaleb, M., & Dhoska, K. (2025). Physics-informed neural networks for solving the heat equation in thermal engineering. *Int. J. Tech. Phys. Probl. Eng*, *17*(1), 375-382.

Alshanti, W. G., Batiha, I. M., Hammad, M. M. A., & Khalil, R. (2023). A novel analytical approach for solving partial differential equations via a tensor product theory of Banach spaces. *Partial Differential Equations in Applied Mathematics*, *8*, 100531.

Blechschmidt, J., & Ernst, O. G. (2021). Three ways to solve partial differential equations with neural networks — A review. *GAMM Mitteilungen*, *44*(2).

Chang, Y., Yang, D., & He, X. (2024). A deep learning operator-based numerical scheme method for solving 1D wave equations. *Journal of Geophysics and Engineering*, *21*(4), 1194–1215.

Dalton, D., Husmeier, D., & Gao, H. (2023). Physics-informed graph neural network emulation of soft-tissue mechanics. *Computer Methods in Applied Mechanics and Engineering*, 417, 116351.

El-Metwaly, A. R., & Kamal, M. A. (2024). A brief review of numerical methods for solving the boundary value problems of PDE. *Journal of Physics: Conference Series*, *2847*(1).

Guo, Y., Cao, X., Liu, B., & Gao, M. (2020). Solving partial differential equations using deep learning and physical constraints. *Applied Sciences (Switzerland)*, *10*(17).

Gürbüz, B., & Fernandez, A. (2024). Numerical and Analytical Methods for Differential Equations and Systems. In *Fractal and Fractional* (Vol. 8, Issue 1). Multidisciplinary Digital Publishing Institute (MDPI).

Hu, W. F., Lin, T. S., & Lai, M. C. (2022). A discontinuity capturing shallow neural network for elliptic interface problems. *Journal of Computational Physics*, *469*.

Jiang, Z., Jiang, J., Yao, Q., & Yang, G. (2023). A neural network-based PDE solving algorithm with high precision. *Scientific Reports*, *13*(1).

Katsikis, D., Muradova, A. D., & Stavroulakis, G. E. (2022). A Gentle Introduction to Physics-Informed Neural Networks, with Applications in Static Rod and Beam Problems. *Journal of Advances in Applied & Computational Mathematics*, *9*, 103–128.

Lima, N. J., Matos, J. M. A., & Vasconcelos, P. B. (2024). Solving Partial Differential Problems with Tau Toolbox. *Mathematics in Computer Science*, *18*(2).

Luo, K., Zhao, J., Wang, Y., Li, J., Wen, J., Liang, J., Soekmadji, H., & Liao, S. (2025). Physics-informed neural networks for PDE problems: a comprehensive review. *Artificial Intelligence Review*, *58*(10).

Maczuga, P., Sikora, M., Służalec, T., Szubert, M., Sztangret, Ł., Szeliga, D., Łoś, M., Dzwinel, W., Pingali, K., & Paszyński, M. (2025). Physics Informed Neural Network Code for 2D Transient Problems (PINN-2DT) Compatible with Google Colab. *Lecture Notes in Computer Science*, *15904 LNCS*, 177–191.

Tanyu, D. N., Ning, J., Freudenberg, T., Heilenkötter, N., Rademacher, A., Iben, U., & Maass, P. (2023). Deep learning methods for partial differential equations and related parameter identification problems. *Inverse Problems*, *39*(10), 103001.

Oh, H., & Jo, G. (2025). Physics-informed neural network for the heat equation under imperfect contact conditions and its error analysis. *AIMS Mathematics*, *10*(4), 7920–7940.

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

Raman, D. (2024). Numerical and analytical approaches to solving partial differential equations. *International Journal of Engineering and Applied Sciences*, 11(3), 9-12.

Wei, M. (2025, June). Numerical Methods for Solving Partial Differential Equations. In *Proceedings of the 2025 5th International Conference on Automation Control, Algorithm and Intelligent Bionics* (pp. 274-277).

Zhang, Z. (2025). Comparative Analysis of Finite Difference Method and Neural Network Models for Solving the One-Dimensional Heat Equal. *Journal of Theory and Practice in Engineering and Technology*, *2*(5), 19–29.