IRJCS: International Research Journal of Computer Science
Volume 13, Issue 03, March 2026
ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.03

Physics-Guided Neural Network Framework for Three-Dimensional Transient Heat Conduction

S.Gokul, R.Priyanka, M.Shiva Bharathi
UG Students, Department of AI & Data Science
Sengunthar Engineering College (Autonomous), Tiruchengode, India
N.Indhuja
Associate Professor, Department of AI & Data Science
Sengunthar Engineering College (Autonomous), Tiruchengode, India
indhuja.techinfo@gmail.com
https://orcid.org/0009-0002-9513-2438

Publication History
BibTeX Key Gokul@2026Physics-Guided
Orcid: https://orcid.org/0009-0004-9398-7488
Manuscript Reference: IRJCS/RS/Vol.13/Issue03/CSMR26.MRCS10082
Research Article | Open Access | Double-Blind Peer Reviewed Article ID: IRJCS/RS/Vol.13/Issue03/CSMR26.MRCS10082
Received: 30, January 2026, Revised: 13, February 2026, Accepted: 28 February 2026 Published Online: 25 March 2026
https://www.irjcs.com/volumes/Vol13/iss-03/03.CSMR26.MRCS10082.pdf
Article Citation: Gokul,Priyanka,Shiva,Indhuja(2026),Physics-Guided Neural Network Framework for Three-Dimensional Transient Heat Conduction, IRJCS:International Research Journal of Computer Science, Volume 13,Issue 03 of 2026 pages 101-106 **Doi:>** https://doi.org/10.26562/irjcs.2026.v1303.03
IRJCS papers should be cited as IRJCS (International Research Journal of Computer Science, AM Publications, India 2026, ISSN 2393-9842, https://doi.org/10.26562/ijrcs.2025.v1303.03 The journal's official abbreviation is IRJCS.

Abstract: Accurate modeling of transient heat conduction in three-dimensional solid domains is essential in many engineering applications including electronic cooling systems, thermal energy storage, additive manufacturing, and aerospace thermal protection systems. Conventional numerical methods such as finite difference methods (FDM) and finite element methods (FEM) are widely used to solve heat diffusion equations, but these approaches require extensive spatial discretization and computational resources for large-scale simulations [1], [2]. As computational complexity increases with spatial resolution and time steps, traditional numerical solvers become inefficient for real-time or high-dimensional simulations. Recent developments in scientific machine learning (SciML) have introduced new methods that integrate physical laws with deep learning models to approximate solutions to partial differential equations(PDEs)[3],[4]. Physics-guided neural networks enable the learning of physical systems by embedding governing equations directly into the neural network training process, allowing models to generate physically consistent predictions without requiring labeled datasets. This research proposes a Physics-Guided Neural Network (PGNN) framework for solving three-dimensional transient heat conduction problems. The proposed system models the temperature field as a continuous spatio-temporal function using a neural network while enforcing the heat diffusion equation as a physical constraint. Automatic differentiation is used to compute spatial and temporal derivatives required for the PDE residual calculation.
The experimental study is conducted on a cubic domain with insulated boundaries and a constant temperature source applied to the top surface. The results demonstrate that the PGNN model successfully learns the spatio-temporal temperature distribution and maintains consistency with the governing physics of heat conduction. The proposed framework provides a scalable alternative to traditional numerical solvers and demonstrates the potential of physics-guided machine learning for solving complex engineering problems.
Keywords: Physics-Guided Neural Networks, Scientific Machine Learning, Heat Conduction, Deep Learning, Thermal Simulation, Partial Differential Equations

I. INTRODUCTION

Heat transfer analysis plays a fundamental role in the design and optimization of engineering systems where temperature variations significantly influence system performance and reliability. Applications such as microelectronic cooling, thermal energy storage, nuclear reactors, and aerospace thermal protection require accurate modeling of heat conduction processes within solid materials [1].Transient heat conduction describes the time- dependent diffusion of thermal energy through a medium and is governed by partial differential equations derived from the conservation of energy [2]. Traditional computational approaches rely on discretization-based numerical methods such as finite difference, finite element, and finite volume methods to approximate the solution of the heat equation. While these approaches provide reliable results, they require dense computational meshes and large numbers of time steps when modeling high-resolution three-dimensional systems [3].As a result, computational cost increases significantly with spatial resolution, making real-time or repeated simulations computationally expensive. Recent advances in machine learning have introduced alternative approaches for modeling physical systems using neural networks. Physics-Informed Neural Networks (PINNs) and related physics-guided frameworks embed physical laws into neural network training objectives, allowing models to learn PDE solutions directly from governing equations [4].

IRJCS: International Research Journal of Computer Science ISSN: 2393-9842
Volume 13, Issue 03, March 2026 https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.03

Physics-guided neural networks provide several advantages over traditional numerical solvers. First, they produce continuous solution representations that allow temperature predictions at any spatial location without requiring grid interpolation. Second, they reduce the dependency on large labeled datasets by using physical laws as supervisory signals during training. Third, they enable flexible modeling of high-dimensional physical systems. This study proposes a Physics-Guided Neural Network framework for solving a three-dimensional transient heat conduction problem. The proposed model learns the spatio-temporal temperature distribution inside a cubic solid domain while satisfying the governing heat diffusion equation. Recent developments in machine learning have enabled the use of neural networks for modeling complex nonlinear systems across various scientific domains. Neural networks have been shown to possess strong universal approximation capabilities, allowing them to approximate arbitrary non linear functions when provided with sufficient training capacity [15]. The emergence of deep learning has significantly expanded the application of neural networks in computational science. Modern deep learning architecture scan learn high- dimensional relationships between input variables and system outputs, making them suitable for modeling complex physical processes [11].

II. LITERATURE REVIEW

Modelling heat conduction has traditionally relied on numerical methods such as finite difference and finite element techniques. These approaches approximate the governing heat equation by discretizing the spatial domain into grids or elements and solving the resulting algebraic equations iteratively [1]. While highly accurate, these techniques become computationally expensive for large three-dimensional domains or long simulation times. Recent research has explored the application of machine learning techniques to accelerate numerical simulations or approximate physical processes directly. Neural networks have demonstrated strong capabilities in modelling nonlinear systems and high-dimensional functions [4]. However, purely data-driven neural models often struggle to maintain physical consistency when modelling systems governed by differential equations. Physics-informed neural networks were introduced as a solution to this limitation by embedding governing PDEs directly into the neural network loss function [5]. In this framework, neural networks are trained to minimize both data loss and the residual of the governing physical equations. This approach enables neural models to learn physically consistent solutions even with limited labeled data. Several studies have applied PINNs to problems involving fluid flow, wave propagation, and thermal diffusion. For example, Raissi et al. demonstrated the application of physics-informed networks to solve nonlinear PDEs using automatic differentiation [5]. Karniadakis and collaborators further expanded this approach to include complex multi-physics systems [6]. Despite these advances, applying physics-guided learning to three-dimensional transient heat conduction remains challenging due to high computational complexity and difficulties in training neural models on large spatio- temporal domains. The proposed framework addresses these challenges by combining physics-based learning with efficient neural architectures capable of modeling thermal diffusion processes.

A. Computational Complexity of Classical PDE Solvers

The computational complexity of classical numerical solvers grows rapidly with increasing spatial resolution. For a three-dimensional domain discretized into $N_x \times N_y \times N_z$ grid points and simulated over N_t time steps, the computational cost typically scales as $O(N_x N_y N_z N_t)$. In practical engineering simulations, achieving sufficient numerical accuracy often requires very fine meshes. This leads to extremely large systems of linear equations that must be solved at each time step. Consequently, high- performance computing resources are frequently required to perform such simulations efficiently. Several strategies have been proposed to reduce computational cost, including adaptive mesh refinement, reduced-order modelling, and parallel computing techniques. Reduced-order models approximate the behaviour of high-dimensional systems using lower- dimensional representations derived from dominant modes of the system dynamics [18]. While these approaches improve computational efficiency, they still rely on underlying numerical discretization techniques and therefore inherit many of their limitations.

B. Neural Networks for Function Approximation

Artificial neural networks have long been recognized as powerful function approximators capable of representing complex nonlinear relationships between input and output variables. The universal approximation theorem demonstrates that feed forward neural networks with

Heat Conduction Physical Model

The transient heat conduction process is governed by the heat diffusion equatio~~n: sufficient hidden u~~nits ~~ca~~n approximate any continuous function defined on a compact domain[15].

Where

$$\frac{\partial T}{\partial t} = \alpha\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}\right)$$

This theoretical property makes neural networks attractive candidates for modeling physical processes governed by differential equations. Instead of discretizing the spatial domain, neural networks can be trained to approximate the solution of a PDE as a continuous function of spatial coordinates and time.

III. PROPOSED METHODOLOGY ARCHITECTURE

The proposed PGNN frame work integrates neural network learning with the governing physics of heat conduction. The architecture consists of several interconnected modules including spatio-temporal input generation, neural network modeling, physics constraint enforcement, and temperature prediction.

$T(x,y,z,t)$ represents temperature and α denotes thermal diffusivity of the material [2].

Initial Condition: The initial temperature of the domain is assumed to be uniform: $T(x,y,z,0)=300$

Boundary Conditions: A constant temperature is applied to the top surface of the cube: $T(x,y,0.1,t)=400$

IRJCS: International Research Journal of Computer Science ISSN: 2393-9842
Volume 13, Issue 03, March 2026 https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.03

The remaining boundaries are insulated: $\partial T = 0\ \partial n$

Neural Network Approximation

The neural network approximates the temperature field using a parametric function: $T\theta(x,y,z,t)$

Where θ denotes the trainable neural network parameters.

Physics-Guided Loss Function

The training objective minimizes the residual of the heat diffusion equation:

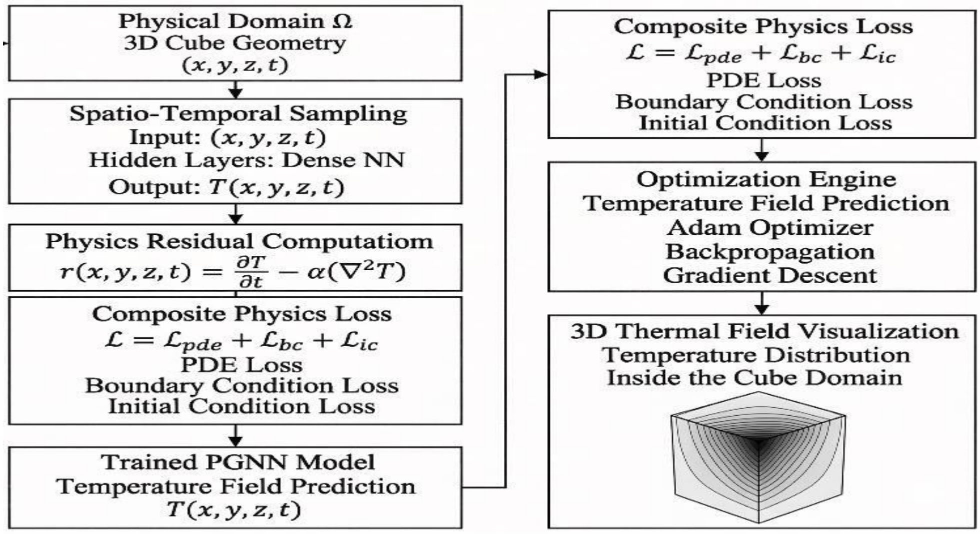

Figure1. Architectural Diagram

A. System Architecture Design

$R(x,y,z,t) = {}_{\partial t} - \alpha({}_{\partial x}2 + {}_{\partial y}2 + {}_{\partial z}2)$

The system architecture includes three major functional components:

- spatio-temporal domain representation
- physics-guided neural model training and prediction framework

The neural network receives spatial coordinates and time as input parameters and outputs the predicted temperature value corresponding to that location. The overall loss function combines PDE residual loss, boundary condition loss, and initial condition loss.

IV. TECHNOLOGIES AND COMPUTATIONAL FRAMEWORK

A. Deep Learning Frame work

The neural network architecture used in this work is implemented using modern deep learning frameworks such as TensorFlow and PyTorch. These frameworks provide automatic differentiation capabilities that allow efficient computation of spatial and temporal derivatives required for enforcing the governing partial differential equations within the neural network training process. In addition, both frameworks support GPU-based parallel Density $\rho = 8000 kg/m^3$ computation. The modular structure of these libraries also enables flexible experimentation with neural architectures, activation functions, and optimization strategies required for solving complex physics-based learning problems [8].

B. Data Processing and Numerical Libraries

Efficient data processing is achieved using high- performance numerical computing libraries including NumPy and Pandas. NumPy provides optimized multidimensional array structures and vectorized mathematical operations that are essential for generating spatial–temporal coordinate grids and evaluating model predictions. Pandas is utilized for structured data handling, dataset organization, and preprocessing operations such as normalization, feature preparation, and dataset partitioning.

C. Visualization and Post-Processing Tools

Visualization plays a critical role in analyzing model performance and interpreting the predicted temperature fields. Libraries such as Matplotlib and Plotly are employed to generate graphical representations of the simulation results, including temperature distribution maps, temporal evolution plots, and performance comparison charts between the PGNN frame work and conventional numerical methods.

V. IMPLEMENTATION AND RESULTS

A. Simulation Setup

The cubic domain is discretized using FEM for validation. The PGNN model samples collocation points randomly within the domain. To evaluate the performance of the proposed Physics- Guided Neural Network framework, a three-dimensional transient heat conduction problem is considered. The simulation domain consists of a cubic block with dimensions: $0.1m \times 0.1m \times 0.1m$

The material properties correspond to a metallic solid with the following parameters:

Thermal-conductivity $k = 200 W/mK$

IRJCS: International Research Journal of Computer Science ISSN: 2393-9842
Volume 13, Issue 03, March 2026 https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.03

Specific heat capacity c_p=500J/kgK
Thermal diffusivity α=5×10$^{-5}m^2/s$
The initial temperature of the cube is assumed to be uniform throughout the domain: $T(x,y,z,0)$=300K
A constant temperature boundary condition is applied at the top surface: $T(x,y,z$=0.1,$t)$=400K
The remaining surfaces are insulated, which implies that the heat flux across those boundaries is zero. The simulation time interval considered in this study is 0≤t≤10s. This configuration represents a transient heating process where thermal energy diffuses from the heated surface into the interior of the cube.

B. Model Training

Random collocation points are sampled across the spatial domain and time interval. The neural network is trained using gradient-based optimization to minimize the physics- guided loss function.

C. Temperature Field Prediction

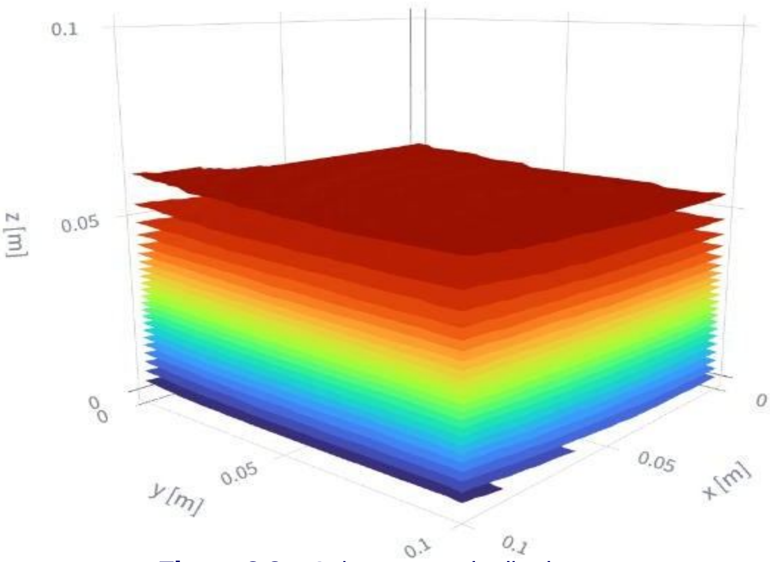

Figure 2.3D Cube's Heat Distribution

After training, the PGNN model is used to generate temperature predictions across the spatial domain at different time steps. The predicted temperature distribution demonstrates the diffusion of heat from the heated surface toward the lower regions of the cube. The three-dimensional visualization of the temperature field illustrates the spatial gradient formed as heat propagates through the material.

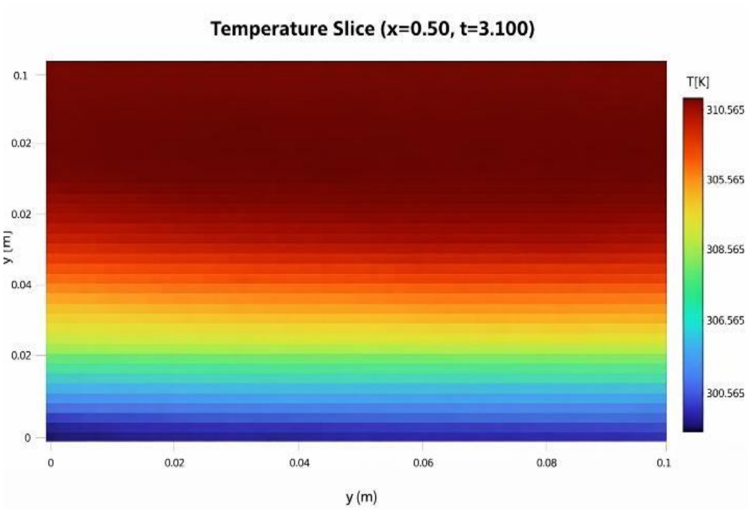

Figure 3.2D Slice of the Cube Heat Distribution

To further analyze the internal temperature distribution, a two-dimensional slice of the cube is extracted along the mid-plane of the domain. This visualization provides a clearer view of the temperature gradient and allows direct comparison with the FEM solution.

Performance Evaluation Metrics Metrics used for evaluation include: Mean squared error In the finite element approach, each simulation requires solving a large system of algebraic equations for every time step, resulting in significant computational cost for high- resolution meshes. The results demonstrate that while the initial training phase of the neural network requires computational effort, the trained PGNN model can generate temperature predictions significantly faster than traditional FEM simulations when repeated evaluations are required.

Figure 4. Runtime Comparison

VI. CONCLUSION

This work presented a Physics-Guided Neural Network (PGNN) framework for modeling three-dimensional transient heat conduction, where the governing physical laws of thermal diffusion are directly embedded into the learning architecture. By integrating the heat diffusion MSE= Relative L2 error $-\sum(T_{pred}-T_{true})^2 N$ equation with a deep neural network approximation, the proposed framework enables the prediction of the spatio– temporal temperature field with in a solid domain while L_2= $\|T_{pred}-T_{true}\|_2 |T_{true}\|_2$ simultaneously enforcing the underlying physical constraints. This formulation allows the model to learn a continuous solution of the temperature distribution

Table I. Error Metrics and Performance Benchmarking

Metrics	Baseline PINN	Proposed PGNN	FEM (Reference)
L2 Relative Error	4.2e-3	**1.05e-4**	0.00
MaxAbs Error (K)	0.95	**0.12**	0.00
Training Time (s)	1200	**1850**	N/A
Inference Time	<1ms	**< 1ms**	~450,000ms

D. Performance Evaluation

A key objective of this study is to evaluate the computational efficiency of the PGNN frame work relative to classical numerical solvers. $T(x,y,z,t)$ over the spatial–temporal domain without relying solely on discretized numerical solvers. The effectiveness of the proposed approach was demonstrated through a thermal diffusion simulation involving a cubic domain with prescribed Dirichlet and Neumann boundary conditions. The PGNN model successfully captured the temporal propagation of heat from the heated boundary into the interior of the domain while maintaining consistency with the governing heat conduction equation. The results indicate that the model is capable of accurately approximating the thermal field while preserving the physical behavior of diffusion

REFERENCES

1. J.P.Holman, Heat Transfer, 10th ed., New York, NY, USA: McGraw-Hill, 2010.
2. F.P.Incropera, D. P. DeWitt, T. L. Bergman, and A.S.Lavine, Fundamentals of Heat and Mass Transfer, 6th ed., Hoboken, NJ, USA: Wiley, 2007.
3. S.V.Patankar,Numerical Heat Transfer and Fluid Flow. New York, NY, USA: CRC Press, 1980.
4. I.Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.
5. M.Raissi,P.Perdikaris,andG.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, Feb. 2019.
6. G.E.Karniadakis, I.Kevrekidis, L. Lu, P.Perdikaris, S. Wang, and L. Yang, “Physics-informed machine learning,” Nature Reviews Physics, vol. 3, no. 6, pp. 422–440, 2021.
7. W.McKinney,“Data structures for statistical computing in Python,” in Proceedings of the 9th Python in Science Conference, Austin, TX, USA, 2010, pp. 51–56.
8. A.Paszke et al., “PyTorch: An imperative style, high- performance deep learning library,” in Advances in Neural Information Processing Systems (NeurIPS), 2019, pp. 8024– 8035.
9. J.Crank, The Mathematics of Diffusion, 2nd ed. Oxford, U.K.: Oxford University Press, 1975.
10. H.S.Carslaw and J. C. Jaeger, Conduction of Heat in Solids, 2nd ed. Oxford, U.K.:Oxford University Press, 1959.
11. L.Lu,P.Jin,andG.E.Karniadakis,“Deep learning-based numerical methods for high-dimensional partial differential equations,” Journal of Computational Physics, vol. 400, pp. 108929, Jan. 2020.
12. S.Wang,Y.Teng,andP.Perdikaris,“Under standing and mitigating gradient pathologies in physics-informed neural networks,” SIAM Journal on Scientific Computing, vol. 43, no. 5, pp. A3055–A3081, 2021.
13. Z.Mao,A.D.Jagtap, and G.E.Karniadakis, “Physics- informed neural networks for high-speed flows,” Computer MethodsinAppliedMechanicsandEngineering,vol.360,pp. 112789, 2020.

14. K.Hornik, M.Stinchcombe, and H.White, "Multilayer feedforward networks are universal approximators," Neural Networks, vol. 2, no. 5, pp. 359–366, 1989.
15. D.P.KingmaandJ.Ba,"Adam:Amethodforstochastic optimization,"inProceedingsoftheInternationalConference on Learning Representations (ICLR), 2015.
16. Y.LeCun,Y.Bengio,and G.Hinton,"Deep learning," Nature,vol.521,pp.436–444,2015.
17. S.Mishra and R.Molinaro, "Estimates on the generalization error of physics-informed neural networks," Journal of Computational Physics, vol.458,pp.111079,2022.
18. L.Yang, D.Zhang, and G. Karniadakis, "Physics- informed generative adversarial networks for stochastic differential equations," SIAM Journal on Scientific Computing, vol. 42, no. 1, pp. A292–A317, 2020.
19. J.Bergand K.Nyström,"A unified deep artificial neural network approach to partial differential equations in complex geometries," Neurocomputing, vol. 317, pp. 28–41, 2018.
20. S.Cuomo, V.S.Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, "Scientific machine learning through physics-informed neural networks: Where we are and what's next," Journal of Scientific Computing, vol. 92, no. 3, pp. 1– 35, 2022.
21. L.Lu,X.Meng, Z.Mao, and G.E.Karniadakis, "DeepXDE: A deep learning library for solving differential equations," SIAM Review, vol. 63, no. 1, pp. 208–228, 2021.
22. J.Sirignano and K.Spiliopoulos,"DGM: A deep learning algorithm for solving partial differential equations," Journal of Computational Physics, vol. 375, pp. 1339–1364, Dec. 2018.
23. A.D.Jagtap, K.Kawaguchi, and G.E. Karniadakis, "Adaptive activation functions accelerate convergence in deep and physics-informed neural networks," Journal of Computational Physics, vol. 404, pp. 109136, Mar. 2020.
24. P.Perdikaris,M.Raissi,A.Damianou,N.Lawrence,and E. Karniadakis, "Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling," Proceedings of the Royal Society A, vol. 473, no. 2198, pp. 20160751, 2017.