

Solving the Helmholtz Equation Using Physics-Informed Graph Neural Networks

Nicolas Bosteels¹, Dennis Giannacopoulos²

¹Department of Electrical and Computer Engineering, McGill University, Montreal, Canada, nicolas.bosteels@mail.mcgill.ca

²Department of Electrical and Computer Engineering, McGill University, Montreal, Canada, dennis.giannacopoulos@mcgill.ca

Building on the graph-neural network architecture proposed by Hao Zhang et al. [1], we propose a novel physics-informed neural network (PINN) approach for efficiently solving the Helmholtz equation in two dimensions. By viewing the computational mesh as a graph, we employ an encoder-processor-decoder architecture that learns local interactions among mesh nodes while simultaneously enforcing the underlying physical laws through a PDE-informed loss term. This novelty lies in leveraging graph neural networks (GNNs) to incorporate geometric domain information for accurate and scalable numerical computation of electromagnetic fields. Numerical experiments confirm the model’s ability to replicate the known exact solution with high accuracy, suggesting that this PINN-GNN synergy has the potential for fast solutions to electromagnetic-wave-related PDEs compared to more traditional numerical methods.

Index Terms—Graph neural networks, partial differential equations, self-supervised learning, electromagnetic fields.

I. INTRODUCTION

Physics-informed neural networks have recently attracted attention as a method to integrate prior domain knowledge such as physical laws represented by partial differential equations (PDEs) within neural network architectures. Traditional approaches like finite element or finite difference methods solve PDEs discretely at grid points but can become computationally expensive for large-scale or complex domains. PINNs, however, incorporate the PDE residual into the training loss, eliminating the need for supervised learning and labeled datasets. Current PINNs [4] primarily use general-purpose deep neural networks, such as multi-layer perceptron architectures, which do not incorporate geometric domain information or leverage the mesh’s node-edge relationships.

Recent works [1] [2] [3] have demonstrated how GNNs can effectively incorporate mesh-based information. In this paper we implement GNNs as the main architecture in the PINN framework for enforcing the Helmholtz equation in order to model stationary wave phenomena. This directly addresses key issues such as handling irregular discretizations, enforcing boundary constraints, and reducing computational demands, making it particularly relevant for electromagnetic field simulations.

II. MODEL ARCHITECTURE

We follow the encoder-processor-decoder architecture, where each stage plays an integral role in transforming graph-based inputs into final solutions consistent with the PDE. In the encoder, mesh nodes and edges are assigned features including coordinates, boundary flags, as well as the forcing term. These features are passed through multilayer perceptrons (MLPs) to embed each node and edge into latent representations.

The processor stage, composed of two message-passing Graph Network (GN) blocks, updates node features by gathering information from neighboring nodes and edges. Concretely,

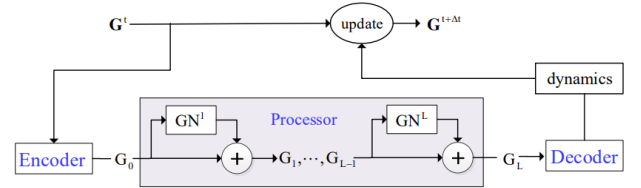


Fig. 1. Encoder-Processor-Decoder Architecture

each GN block performs an edge update, computing new edge features from the existing node and edge embeddings. Then, an aggregation step aggregates the updated edge features associated with a node, followed by a final node update that combines these aggregated edge features with the latent representation of the node. Two GN blocks were used in our architecture, achieving an adequate balance between accuracy and the over-smoothing problem that often arises in GNNs.

Finally, in the decoder, latent node features are projected back into physical solution space through another MLP, outputting an estimate

$$\hat{u}(x).$$

To enforce the Helmholtz equation:

$$\epsilon \Delta \hat{u}(x) + k^2 \hat{u}(x) = f(x), \quad x \in [0, 1]^2,$$

$$f(x, y) = 2\pi \cos(\pi y) \sin(\pi x) + 2\pi \cos(\pi x) \sin(\pi y) \\ + (x + y) \sin(\pi x) \sin(\pi y) - 2\pi^2 (x + y) \sin(\pi x) \sin(\pi y),$$

We incorporate a physics-informed neural network loss:

$$L = \frac{1}{N} \sum_{i=1}^N \underbrace{(\epsilon \Delta \hat{u}(x_i) + k^2 \hat{u}(x_i) - f(x_i))^2}_{\text{PDE residual term}} + \sum_{j=1}^{N_b} \underbrace{(\hat{u}(x_j) - g(x_j))^2}_{\text{boundary constraints}}$$

where N is the number of interior nodes, and N_b the boundary nodes.

Because we know the exact boundary values are zero on all four edges, we hard-enforce them via the following ansatz:

$$\hat{u}(x, y) = D(x, y) u_{\text{raw}}(x, y),$$

$$D(x, y) = \tanh(\pi x) \tanh(\pi(1-x)) \tanh(\pi y) \tanh(\pi(1-y)).$$

which vanishes when $x = 0$, $x = 1$, $y = 0$, or $y = 1$.

The loss then simplifies to:

$$L = \frac{1}{N} \sum_{i=1}^N \underbrace{(\epsilon \Delta \hat{u}(x_i) + k^2 \hat{u}(x_i) - f(x_i))^2}_{\text{PDE residual term}},$$

By explicitly setting the boundary values, we ensure that the network satisfies the boundary conditions, simplifying the optimization process by reducing one component of the loss function. Moreover, the Laplace operator applied here is obtained using automatic differentiation (AD). This final residual loss guides the network to produce physically consistent solutions without the need for data-driven supervision.

III. EVALUATION ON THE HELMHOLTZ EQUATION

We tested our method on the two-dimensional Helmholtz problem, using the analytical solution [3]:

$$u(x, y) = (x + y) \sin(\pi x) \sin(\pi y)$$

to evaluate its accuracy. A single unstructured triangular mesh was generated, and the node inputs, including spatial coordinates and the forcing function, were passed to the GNN. The neural network was trained by minimizing the PDE residual, converging quickly to a stable solution.

The final predicted solution can be observed in Fig. 2, which displays the voltage distribution of the field, with values ranging from 0 volts at the boundary to 1 at the center. This distribution matched closely with the known analytical field. On the left of this figure, the analytical solution can be visualized, and in the middle of the figure, the voltage distribution predicted by our model is shown. Finally, to the right of the figure is the absolute error between the predicted and exact solutions, which is always smaller than 0.07. In our experiments, we sampled 6650 collocation points or nodes which were passed as input to the neural network.

IV. CONCLUSION

In this work, we extended the graph-based PINN framework introduced by Hao Zhang et al. [1]. to solve the two-dimensional Helmholtz equation, demonstrating its capability to achieve accurate solutions on an unstructured triangular mesh. By encoding mesh data into a graph neural network architecture and using a physics-informed loss function derived from the governing PDE, the proposed method offers several advantages. These include flexible handling of irregular domain geometries, efficient boundary condition enforcement, and accurate field reconstruction. These features make the model

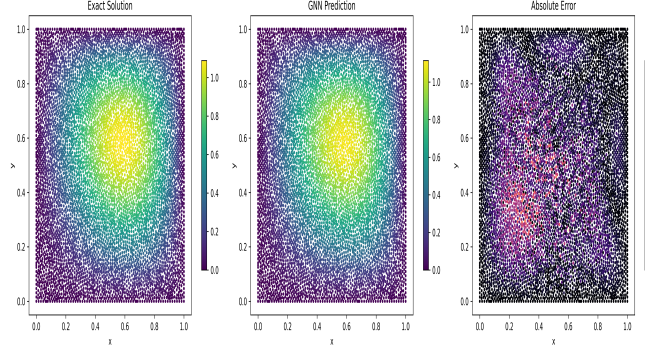


Fig. 2. Exact, Predicted, and Difference in Solutions

well suited for applications in electromagnetics, where wave-dominated phenomena often require both numerical stability and computational efficiency.

Building on these results, we are currently extending our PINN-GNN framework to tackle multi-material Helmholtz problems with several concentric inclusions. Introducing interface-jump losses alongside boundary and PDE terms requires careful balancing, for which we are exploring NTK-inspired adaptive weighting and normalized loss scaling. Our preliminary two-region experiments confirm that the graph-based approach can handle heterogeneous coefficients far more naturally than conventional PINNs, and ongoing work will further validate the method on even more complex geometries and coupled steady-state electro-thermal problems.

Looking ahead, this method has significant potential for broader applications. Beyond static wave problems, the framework can be naturally extended to tackle time-dependent PDEs and multi-physics problems, such as the dynamically coupled electro-thermal simulation problem needed for real-time radio-frequency tumor ablation (RFA) simulations. Furthermore, by incorporating more complex geometries, the approach could be applied to challenging real-world problems, such as simulations of electromagnetic scattering, and wave propagation in heterogeneous media.

REFERENCES

- [1] Zhang, H. et al. (2024) Combining physics-informed graph neural network and finite difference for solving forward and inverse spatiotemporal pdes, arXiv.org. Available at: <https://arxiv.org/abs/2405.20000v2>.
- [2] Y. Salehi and D. Giannacopoulos, "PhysGNN: A physics-driven graph neural network based model for predicting soft tissue deformation in image-guided neurosurgery", Proc. Adv. Neural Inf. Process. Syst., vol. 35, pp. 37282-37296, 2020.
- [3] J. Rylo and D. Giannacopoulos, "Mesh Error Estimation Using Graph Neural Networks," 2024 IEEE 21st Biennial Conference on Electromagnetic Field Computation (CEFC), Jeju, Korea, Republic of, 2024, pp. 1-2, doi: 10.1109/CEFC61729.2024.10586063.
- [4] Goswami, S. et al. (2022) Physics-informed Deep Neural Operator Networks, arXiv.org. Available at: <https://arxiv.org/abs/2207.05748>.
- [5] Zhang, R. et al. (2025) BO-SA-PINNs: Self-adaptive physics-informed neural networks. Available at: <https://arxiv.org/pdf/2504.09804> (Accessed: 19 April 2025).