Project Progress Report: Physics-Informed Neural Combinatorial Wavelet Neural Operator (PI-NCWNO)

An Efficient Gradient-Free Foundational Model for PDE Systems

Mehul Goyal
Indian Institute of Science Education and Research Thiruvananthapuram

Under the guidance of
Dr. Souvik Chakraborty
Assistant Professor, Indian Institute of Technology Delhi

Project Dashboard: Click here to view the live Weights & Biases Report

1 Introduction and Aim of the Project

1.1 Context and Motivation

High-fidelity simulations are the backbone of modern engineering, but they face a critical bottleneck: traditional numerical solvers are too slow for real-time applications, while deep learning surrogates require massive, expensive datasets.

This report introduces the **Physics-Informed Neural Combinatorial Wavelet Neural Operator (PI-NCWNO)**, a next-generation simulation engine designed to break this trade-off. By embedding physical laws directly into a foundational architecture, our model learns to simulate complex dynamics **without historical data**.

Key Breakthroughs:

- **6x Efficiency Gain:** A novel gradient-free training method reduces computational costs by $\approx 85\%$ compared to standard industry baselines.
- **Zero-Shot Physics:** The model learns purely from governing equations (PDEs), eliminating the "cold start" data problem.
- **Foundational Capability:** A single model architecture can learn multiple physical systems simultaneously, paving the way for adaptable Digital Twins.

1.2 Project Aim

The primary aim of this project is to develop a **Physics-Informed Neural Combinatorial Wavelet Neural Operator (PI-NCWNO)**. By embedding physical laws directly into the training process, we aim to eliminate the need for labeled data while retaining the foundational adaptability of the NCWNO. Specifically, this project seeks to:

1. **Hybridize Architectures:** Combine the foundational MoE architecture of NCWNO with physics-informed learning.
2. **Enhance Efficiency:** Replace computationally expensive Automatic Differentiation (AD, through industry standards like Pytorch) with a gradient-free stochastic projection method [2].
3. **Ensure Stability:** Utilize implicit Crank-Nicolson time-stepping residuals [3] to ensure numerical stability for long-horizon predictions.

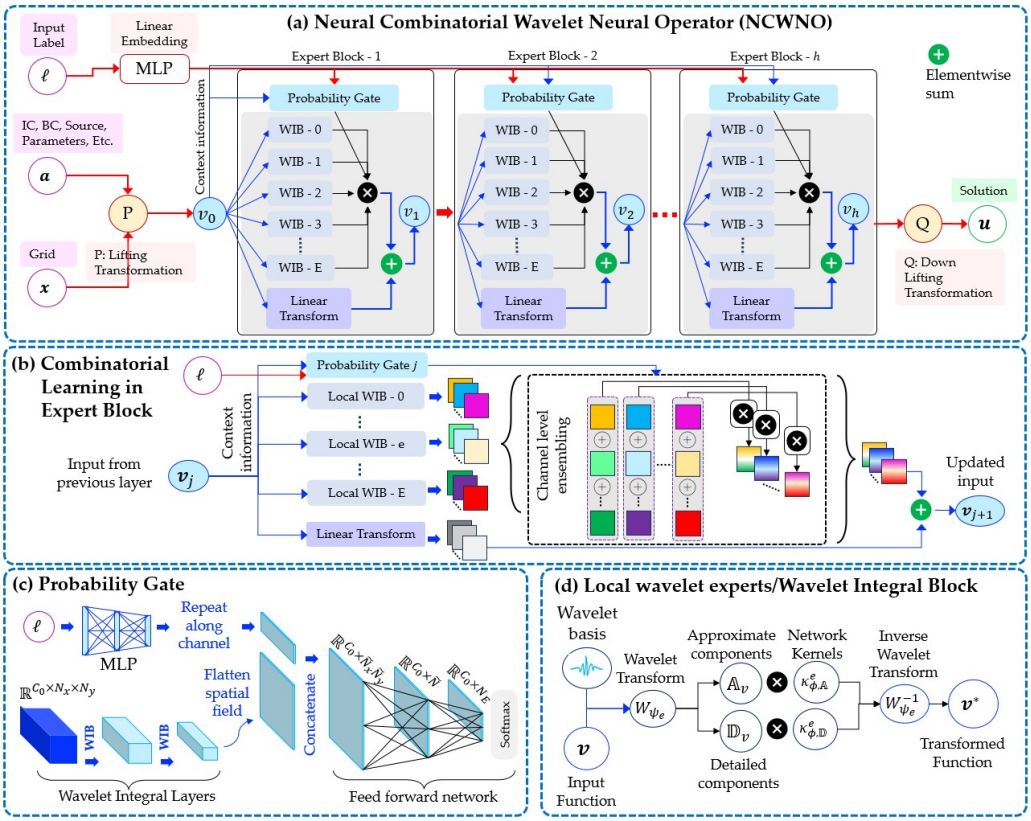

Figure 1: Architecture of NCWNO (source)

4. **Foundational Learning:** Demonstrate the model's ability to learn multiple source PDEs (Allen-Cahn, Nagumo, Burgers) simultaneously.

5. **Sequential Learning:** To continually learn the solution of other PDEs by parameter efficient fine tuning (gating params).

2 Methodology

To achieve a data-free and efficient foundational model, we integrated three distinct methodologies into a unified training pipeline.

2.1 Network Architecture: The NCWNO Backbone

I'm working on to improve the NCWNO architecture [1]. This framework utilizes a **Gating Network** to dynamically ensemble a set of **Local Wavelet Experts**.

- **Wavelet Integral Blocks:** Each expert learns distinct spectral features of the solution operator in the wavelet domain.
- **Combinatorial Transfer:** The gating network predicts mixing weights based on the input context (initial conditions, PDE coefficients), allowing the model to "switch" behaviors for different physical dynamics.

2.2 Physics-Constrained Loss: Crank-Nicolson Residuals

Instead of minimizing data mismatch (MSE), we minimize the residual of the governing equations. To handle time-dependent PDEs robustly, we implemented the implicit **Crank-Nicolson time-stepping scheme** as proposed by Geneva and Zabaras [3].

For a generic PDE $u_t = \mathcal{F}(u, u_x, u_{xx}, \dots)$, the loss function $\mathcal{L}$ for a transition from time-step n to $n+1$ is formulated as:

$$\mathcal{L} = \left\| \frac{\hat{u}^{n+1} - \hat{u}^n}{\Delta t} - \frac{1}{2}\left[\mathcal{F}(\hat{u}^{n+1}) + \mathcal{F}(\hat{u}^n)\right] \right\|^2 \tag{1}$$

where $\hat{u}$ is the model prediction. This implicit formulation creates a "trapezoidal" integration rule in time, offering significantly higher stability for larger Δt compared to explicit Euler methods.

2.3 Gradient-Free Optimization: Stochastic Projection

A major computational bottleneck in standard PINNs is the use of Automatic Differentiation (AD) to compute the spatial derivatives inside $\mathcal{F}(\cdot)$. AD builds a massive computational graph that scales poorly with network depth.

To address this, we integrated the **Stochastic Projection (SP)** method [2]. Instead of AD, spatial derivatives (e.g., u_x, u_{xx}) are estimated using a convolution-like operation with neighborhood information. The gradient G at a point x is approximated as:

$$G(x) \approx \frac{\mathbb{E}[(u(z) - u(x))(z - x)^T]}{\mathbb{E}[(z - x)(z - x)^T]} \tag{2}$$

This approach treats local field variations as stochastic projections, allowing us to compute physics residuals purely through forward passes, drastically reducing memory usage and training time.

3 Results

3.1 Configurations and Efficiency Improvements

The model was configured with **4 wavelet experts per layer** at a decomposition level of 5, operating on a 1D mesh resolution of 128. Through the integration of Stochastic Projections and vectorization of sample processing, we achieved:

- **Training Speedup:** An approximate **6x reduction in training time** compared to the baseline supervised NCWNO and standard AD-based PINN implementations.
- **Optimization:** By lowering the number of neighbor points for Stochastic Projection calculations (specifically for PDEs with sharp gradients like Burgers'), we observed a marked increase in gradient approximation accuracy.

3.2 Foundational Learning Performance

The model was trained simultaneously on the **Nagumo** and **Burgers'** equations. The foundational training successfully captured the dynamics of both systems without labeled data. The Mean Relative L^2 Errors averaged over 100 initial conditions are summarized below:

Table 1: Mean Relative L^2 Error for Foundational Tasks

Prediction Horizon	Nagumo PDE	Burgers' PDE
$T : 10 \to 20$	0.057	0.083
$T : 10 \to 40$	0.062	0.100

Observations:

- The model maintains high accuracy (Mean Sample Accuracy > 0.9) even when extrapolating to 40 time steps.
- The Burgers' equation presented higher error rates due to shock formation, which was partially mitigated by reducing the neighborhood size for stochastic gradient calculation.

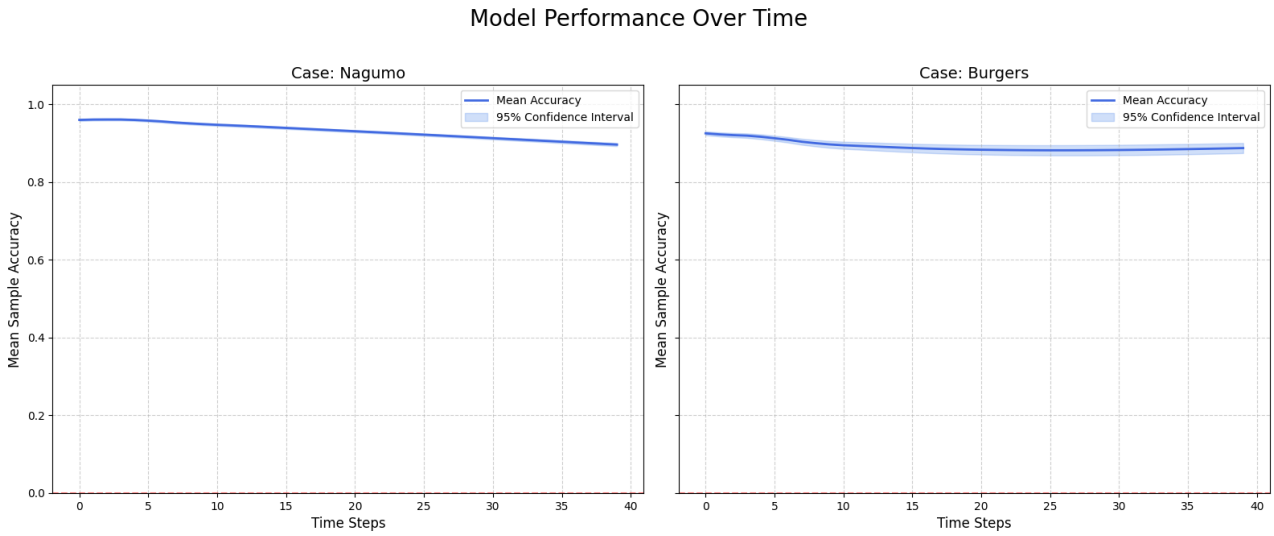

Figure 2: Foundational model performance over time

3.3 Sequential Learning (Fine-Tuning) Status

Following foundational training, we attempted combinatorial transfer learning (fine-tuning) on three unseen PDEs: **Advection**, **Allen-Cahn**, and **Heat**. Preliminary results indicate that while the model learns, the error rates remain higher than foundational tasks.

Table 2: Preliminary Sequential Learning Results (Mean Relative Error)

Target PDE	**Mean Relative Error**
Advection	≈ 0.300
Allen-Cahn	≈ 0.244
Heat Equation	≈ 0.253

While the training loss decreases and test accuracy improves over epochs, the current error levels suggest that the sequential adaptation process requires further regularization to prevent catastrophic forgetting and improve convergence on new physics.

4 Future Work and Improvements

The immediate focus of future work is to stabilize the sequential learning phase and leverage the foundational model's speed for broader applications.

References

[1] Tripura, T., & Chakraborty, S. (2023). *A Foundational Neural Operator that Continuously Learns Without Forgetting*. arXiv preprint arXiv:2310.18885.

[2] Navaneeth, N., & Chakraborty, S. (2022). *Stochastic Projection Based Approach for Gradient Free Physics Informed Learning*. arXiv preprint arXiv:2209.13724.

[3] Geneva, N., & Zabaras, N. (2020). *Modeling the dynamics of PDE systems with physics-constrained deep auto-regressive networks*. Journal of Computational Physics, 403, 109056.