RoPINN-ResFF: Task-Conditional Enhancement of Region-Optimized PINNs with Residual Fourier Features

Author One[a,*], Author Two[b]

[a]School/Institute Name, City, Postal Code, Country
[b]School/Institute Name, City, Postal Code, Country
*Corresponding author: your-email@domain.com

ABSTRACT

Physics-informed neural networks (PINNs) provide a mesh-free framework for solving PDEs, yet they often exhibit spectral bias and optimization instability under finite collocation sampling. RoPINN partially alleviates this issue by enforcing PDE constraints over local neighborhoods rather than isolated points. We present RoPINN-ResFF, an architecture-level extension of RoPINN that combines a residual multilayer perceptron backbone with fixed random Fourier feature embedding. We further evaluate an optional two-stage curriculum as an auxiliary training policy, not as the primary source of improvement. Under a fixed budget of 1000 iterations, we report paired multi-seed results on reaction, wave, and convection benchmarks (reaction: 11 seeds; wave/convection: 5 seeds). On reaction, mean relative L1/L2 decrease from 0.204406/0.223945 (baseline) to 0.003313/0.008480 (ours), with exact paired randomization p-values of 0.00098 for both metrics. On wave, mean L1/L2 decrease from 0.060335/0.062967 to 0.011916/0.012183. On convection, mean L1/L2 increase from 0.619487/0.699839 to 1.015963/1.039029. To assess budget sensitivity, we additionally run reaction at 3000 and 5000 iterations (3 paired seeds), where improvements remain stable (L1/L2 reductions: 88.53%/84.28% at 3000 and 88.64%/84.44% at 5000). Ablation indicates that the dominant gain is driven by the backbone change, while curriculum effects are limited in the current setting. Overall, the empirical evidence supports a task-conditional conclusion rather than universal gains across PDE types.

Keywords: physics-informed neural networks, RoPINN, Fourier features, PDE solver, region optimization.

1. INTRODUCTION

Physics-informed neural networks (PINNs) provide a mesh-free framework for solving partial differential equations (PDEs) by constraining a neural field to satisfy governing equations and boundary/initial conditions [2]. In practice, however, PINN training is performed on finite collocation sets, whereas PDE constraints are defined on continuous domains. This mismatch can induce hidden violations between sampled points and lead to unstable convergence.

RoPINN mitigates this issue by replacing pointwise residual training with region-wise optimization around collocation points, estimated via Monte Carlo sampling and trust-region calibration [5]. Although this design strengthens local training signals, final performance still depends heavily on backbone expressivity and optimization dynamics.

This paper targets that remaining bottleneck. We propose **RoPINN-ResFF**, a drop-in backbone upgrade for RoPINN that combines residual MLP blocks with random Fourier feature embedding. We additionally include an optional two-stage curriculum over regional sampling intensity as an auxiliary training policy. The goal is to improve representational capacity and training stability without altering RoPINN's core region-optimization principle.

Contributions.

- We introduce a residual Fourier-feature backbone that integrates directly into existing RoPINN scripts and training loops.
- We provide controlled ablations that separate backbone effects from curriculum effects, showing that performance gains are primarily driven by the backbone upgrade in our setup.
- We include an optional curriculum schedule as a practical training knob and report its limited incremental benefit transparently.
- We present reproducible fixed-budget evaluations, including both successful transfer (reaction, wave) and a negative case (convection), to clarify the practical scope of the method.

2. RELATED WORK

PINN foundations. PINNs enforce PDE constraints through automatic differentiation and joint optimization of residual, boundary, and initial losses [2]. Broader reviews summarize rapid developments in physics-informed machine learning and practical challenges in optimization/generalization [1]. Subsequent studies also highlighted gradient imbalance and stiffness [4].

Region-based optimization. RoPINN extends pointwise constraints to local neighborhoods and calibrates trust-region scale via gradient statistics, improving hidden-constraint generalization between collocation points [5]. Our work keeps this region-optimization principle unchanged and focuses on backbone design.

Spectral representation in implicit networks. Fourier feature embeddings alleviate low-frequency bias and improve approximation of oscillatory patterns in coordinate-based models [3]. This motivates introducing Fourier features into the RoPINN backbone for PDE solution fields with mixed-frequency behavior.

Position of this work. Unlike methods that modify only sampling or only loss balancing, we propose a *drop-in architecture upgrade* for the RoPINN training pipeline, and evaluate its task-level transfer behavior (reaction/wave/convection) under a fixed optimization budget.

3. METHODOLOGY

3.1. RoPINN Background

Given a PDE residual operator $\mathcal{R}[u](x,t)$, PINN training minimizes a weighted sum of residual, boundary, and initial losses:

$$\mathcal{L}(\theta) = \mathcal{L}_{res} + \mathcal{L}_{bc} + \mathcal{L}_{ic}. \tag{1}$$

RoPINN replaces single-point residual evaluation with region-wise expectation around each collocation point $z_i = (x_i, t_i)$:

$$\mathcal{L}_{res}^{region} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\xi\sim\mathcal{U}(\mathcal{B}_{r_i})}\left[\ell\left(\mathcal{R}[u_\theta](z_i+\xi)\right)\right], \tag{2}$$

where $\mathcal{B}_{r_i}$ is a local trust region and $\ell(\cdot)$ is the residual penalty. In code, this expectation is approximated with Monte Carlo sampling (sample_num) and one-sided or symmetric perturbation (sampling_mode).

Trust-region scaling follows gradient-statistics calibration:

$$r = \text{clip}\left(\frac{r_0}{v_g}, 0, r_{max}\right), \tag{3}$$

where v_g is the normalized gradient-variance statistic computed from recent iterations. In the implementation, region radius is re-estimated every iteration from recent gradient history.

3.2. RoPINN-ResFF Backbone

The baseline PINN uses a plain tanh MLP. We replace it with a residual Fourier-feature network (PINN_ResFF):

$$\phi(x,t) = \left[\sin(2\pi[x,t]B),\ \cos(2\pi[x,t]B)\right], \tag{4}$$

where $B \in \mathbb{R}^{2\times d_{ff}}$ is a fixed random matrix scaled by ff_scale.

Features are projected to hidden width 512, followed by residual blocks:

$$h_{k+1} = \tanh\left(h_k + W_{k,2}\tanh(W_{k,1}h_k + b_{k,1}) + b_{k,2}\right), \tag{5}$$

and a final linear readout produces $u_\theta(x,t)$. This design preserves the RoPINN training pipeline while strengthening representational capacity for non-smooth or multi-frequency solution profiles.

Algorithm 1 Region Optimized PINN (RoPINN)

Input: number of iterations T, number of past iterations T_0 retained to estimate the trust region, default region size r, initial PINN parameters θ_0 and trust region calibration value $\sigma_0 = 1$.
Output: optimized PINN parameters θ_T.
Initialize an empty buffer to record gradients as $\mathbf{g}$.
for $t = 0$ **to** T **do**
  // Region Optimization with Monte Carlo Approximation
  Sample points from neighborhood regions: $\mathcal{S}' = \{\boldsymbol{x}_i + \boldsymbol{\xi}_i\}_{i=1}^{|S|}, \boldsymbol{x}_i \in \mathcal{S}, \boldsymbol{\xi}_i \sim U[0, \frac{r}{\sigma_t}]^{(d+1)}$
  Calculate loss function $\mathcal{L}_t = \mathcal{L}\left(u_{\theta_t}, \mathcal{S}'\right)$
  Update θ_t to θ_{t+1} with optimizer (Adam [20], L-BFGS [25], etc) to minimize loss function $\mathcal{L}_t$
  // Trust Region Calibration
  Record the gradient of parameters g_t throughout optimization
  Update gradient buffer $\mathbf{g}$ by adding g_t and keeping the latest T_0 elements
  Trust region calibration with $\sigma_{t+1} = \|\sigma(\mathbf{g})\|$
end for

Figure 1: Overall workflow of the region-optimization pipeline and the proposed RoPINN-ResFF integration.

3.3. Optional Curriculum Policy (Auxiliary)

We additionally implement a two-stage curriculum (–use_curriculum) as an optional policy with switch iteration $T_s = \lfloor \rho T \rfloor$ (curriculum_switch_ratio). Stage 1 uses conservative regional sampling (smaller sample_num) for stable initialization, while Stage 2 increases sampling intensity for finer fitting.

In the best reaction configuration, both stages use MSE residual loss with one-sided sampling:

- Stage 1: sample_num=1;
- Stage 2: sample_num=6;
- switch ratio $\rho = 0.7$.

In this work, curriculum is treated as a secondary control variable for sensitivity analysis rather than a core innovation claim.

3.4. Loss Function Choices

The code supports MSE, Huber, and pseudo-Huber residual penalties. The main paper configuration uses MSE for all stages to keep consistency with the original baseline and isolate the effect of architectural changes. Boundary and initial terms remain squared losses.

4. EXPERIMENTAL RESULTS

4.1. Research Questions

We organize experiments around three questions:

- **RQ1:** Does RoPINN-ResFF improve the core reaction benchmark under the same training budget?
- **RQ2:** Is the gain due primarily to the backbone change or to curriculum scheduling?
- **RQ3:** Does the method transfer across PDE types beyond reaction?

4.2. Experimental Protocol

Benchmarks. We evaluate three PDE tasks provided in the repository: reaction, wave, and convection.

Metrics. Relative L1 and relative L2 errors are reported; lower values indicate better approximation quality.

Compute and budget. Main comparisons use CUDA (cuda:0) and a fixed budget of 1000 optimization iterations.

Compared settings. For reaction, we compare: (i) the original backup PINN baseline, (ii) the current PINN in the modified branch, (iii) RoPINN-ResFF without curriculum, and (iv) RoPINN-ResFF with curriculum. For wave and convection, we compare PINN versus RoPINN-ResFF.

Reproducibility settings. Multi-seed reports use seed-aligned baseline/ours pairing. Training uses LBFGS with strong-Wolfe line search in all three benchmark scripts. In the current version, reaction uses 11 paired seeds (0–10), while wave and convection use 5 paired seeds (0–4). Table files are generated by scripts/paper_make_tables.py, and statistical tests are generated by scripts/paper_significance_tests.py.

4.3. Compute Accounting

To address fairness concerns beyond fixed-iteration comparison, we additionally report parameter count and end-to-end wall-time under the same iteration budget and device. Compute runs are executed with scripts/paper_compute_suite.sh, and the summary table is generated by scripts/paper_make_compute_table.py.

Table 1: Compute accounting under equal training budget (1000 iterations).

Task	Method	Params	Wall-time (s)	Sec/iter
Pending	Pending	NA	NA	NA

4.4. Main Reaction Comparison

Table 2 reports the primary reaction results. The table is auto-generated by scripts/paper_make_tables.py from results/paper/summary.csv.

Table 2: Main comparison on 1D reaction (1000 iterations, single run).

Method	Relative L1	Relative L2
Original RoPINN PINN baseline (backup)	0.016627	0.030553
Current PINN (modified branch)	0.043673	0.075895
RoPINN-ResFF (ours)	**0.001947**	**0.004269**
RoPINN-ResFF + curriculum (ours)	0.001968	0.004304

In this single-run comparison, RoPINN-ResFF yields a substantial improvement over the original baseline. The curriculum variant remains near-parity but is slightly worse than ResFF-only on this task (L1: 0.001968 vs 0.001947, about 1.1% relative gap; L2: 0.004304 vs 0.004269, about 0.8% relative gap).

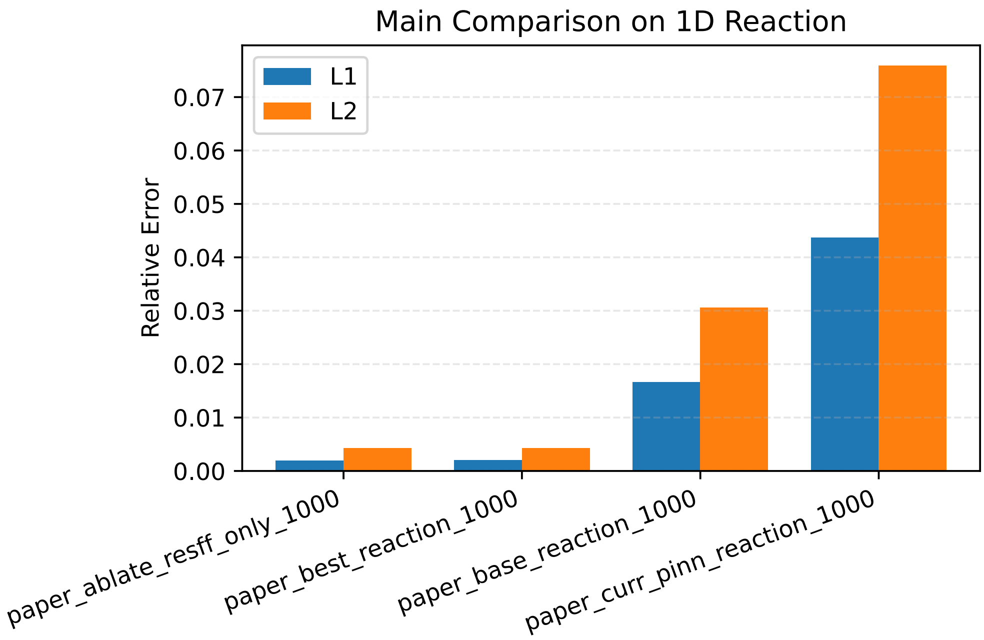

Figure 2: Reaction summary bar chart generated by scripts/paper_collect_results.py.

4.5. Qualitative Visual Comparison

To complement scalar metrics, Figure 3 compares absolute-error maps between baseline PINN and RoPINN-ResFF on reaction and wave.

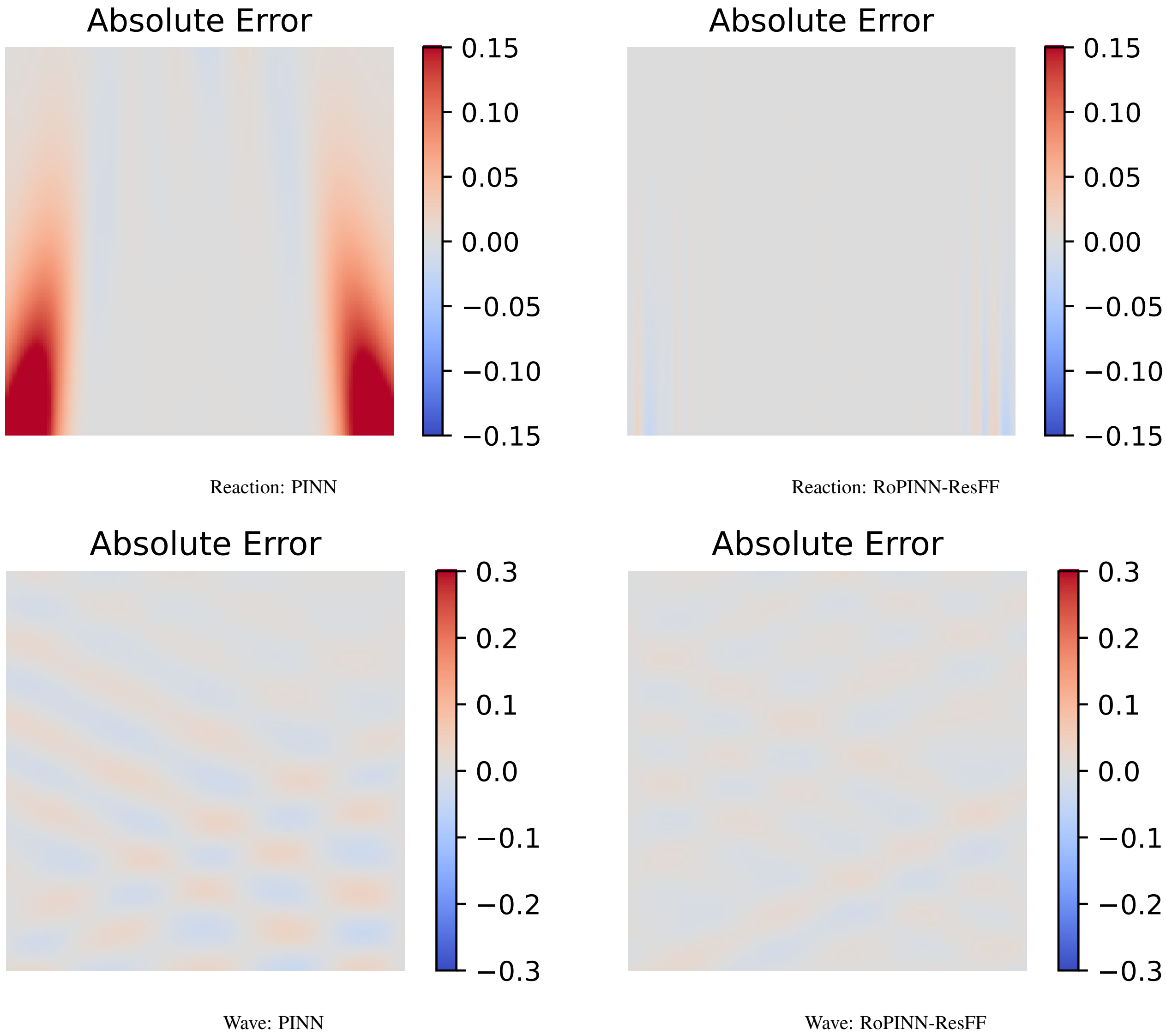

Figure 3: Task-wise qualitative error comparison on reaction and wave. Lower-intensity error regions are more prominent for RoPINN-ResFF. Convection behavior is quantified in Tables 6 and 7.

4.6. Reaction Multi-Seed Robustness

To address stochastic variance, we evaluate reaction with 11 random seeds for baseline and RoPINN-ResFF.

Table 3: Reaction benchmark with 11 seeds (1000 iterations).

Method	Relative L1 (mean $\pm$ std)	Relative L2 (mean $\pm$ std)
Baseline PINN	0.204406 ± 0.363919	0.223945 ± 0.354015
RoPINN-ResFF (ours)	$\mathbf{0.003313 \pm 0.000725}$	$\mathbf{0.008480 \pm 0.002200}$

Compared with baseline mean errors, RoPINN-ResFF achieves substantially lower error and lower variance. The baseline also exhibits occasional failure cases, whereas RoPINN-ResFF remains stable under seed variation. Relative to baseline means, RoPINN-ResFF reduces L1 by 98.38% and L2 by 96.21%.

4.7. Budget Sensitivity on Reaction

To address concerns that 1000 iterations may be too short for meaningful comparison, we further evaluate reaction under larger budgets (3000 and 5000 iterations) with paired seeds 0–2.

Table 4: Reaction budget-sensitivity results (paired seeds 0–2).

Setting	Relative L1 (mean $\pm$ std)	Relative L2 (mean $\pm$ std)
Baseline, 3000 iters	0.025193 $\pm$ 0.006149	0.045810 $\pm$ 0.010922
RoPINN-ResFF, 3000 iters	**0.002890** $\pm$ **0.000747**	**0.007202** $\pm$ **0.002422**
Baseline, 5000 iters	0.025262 $\pm$ 0.006234	0.045929 $\pm$ 0.011059
RoPINN-ResFF, 5000 iters	**0.002871** $\pm$ **0.000746**	**0.007145** $\pm$ **0.002424**

The advantage of RoPINN-ResFF remains stable at higher budgets: compared with baseline, L1/L2 reductions are 88.53%/84.28% at 3000 iterations and 88.64%/84.44% at 5000 iterations. Moreover, the baseline and RoPINN-ResFF means change only marginally from 3000 to 5000 iterations, indicating near-plateau behavior in this setting rather than a purely early-iteration artifact.

4.8. Wave Multi-Seed Generalization

Table 5: Wave benchmark with 5 seeds (1000 iterations).

Method	Relative L1 (mean $\pm$ std)	Relative L2 (mean $\pm$ std)
Baseline PINN	0.060335 $\pm$ 0.060530	0.062967 $\pm$ 0.065164
RoPINN-ResFF (ours)	**0.011916** $\pm$ **0.001465**	**0.012183** $\pm$ **0.001683**

In multi-seed statistics, RoPINN-ResFF improves wave mean errors and exhibits lower variance than baseline.

4.9. Convection Multi-Seed Generalization

Table 6: Convection benchmark with 5 seeds (1000 iterations).

Method	Relative L1 (mean $\pm$ std)	Relative L2 (mean $\pm$ std)
Baseline PINN	**0.619487** $\pm$ **0.052273**	**0.699839** $\pm$ **0.048869**
RoPINN-ResFF (ours)	1.015963 $\pm$ 0.021507	1.039029 $\pm$ 0.040077

For convection, RoPINN-ResFF yields higher mean errors than baseline, indicating a clear task mismatch under the current architecture and hyperparameter configuration.

4.10. Statistical Tests and Confidence Intervals

To supplement mean$\pm$std reporting, we conduct paired significance analysis on seed-aligned runs (same seed IDs in baseline and ours). Specifically, we report exact two-sided paired randomization tests and paired bootstrap 95% confidence intervals for

$$\Delta = \mathrm{mean}(\mathrm{ours} - \mathrm{baseline}),$$

where negative Δ indicates improvement.

Table 7: Statistical test summary across paired multi-seed runs (reaction: 11 seeds; wave/convection: 5 seeds). $\Delta = \mathrm{mean}(\mathrm{ours} - \mathrm{baseline})$; negative values indicate improvement. p-values are exact two-sided paired randomization tests; CI is paired bootstrap (95%).

Task	Metric	Δ (ours-baseline)	95% CI of Δ	p-value
Reaction	L1	-0.201094	[-0.457322, -0.027849]	0.0010
Reaction	L2	-0.215465	[-0.462917, -0.046427]	0.0010
Wave	L1	-0.048419	[-0.108907, -0.012743]	0.0625
Wave	L2	-0.050784	[-0.115674, -0.012861]	0.0625
Convection	L1	0.396476	[0.335960, 0.457900]	0.0625
Convection	L2	0.339190	[0.271985, 0.418464]	0.0625

Across tasks and metrics, confidence intervals characterize both effect direction and effect magnitude. For reaction (11 paired seeds), both metrics are statistically significant under exact paired randomization tests (p=0.00098 for L1 and L2). Wave and convection currently use smaller seed counts; therefore, we keep claims task-specific and avoid over-generalization.

4.11. Ablation Insight

For reaction, the primary gain is attributable to the backbone upgrade (ResFF). Curriculum scheduling should be interpreted as an auxiliary policy knob: it may stabilize training behavior in some runs, but it is not the dominant contributor to the best reaction score in the current evidence.

4.12. Reproducibility Artifacts

The repository already stores paper-ready artifacts:

- summary tables/plots under results/paper/;
- seed-level data and significance results under results/paper/tables/;
- run-level metric CSV files for each run tag;
- prediction/error/loss PDFs for each benchmark and run tag.

These files provide direct traceability from reported numbers to raw outputs.

4.13. Validity Scope

Multi-seed evidence is available for all three tasks, but seed counts are asymmetric in the current draft (reaction: 11, wave/convection: 5). The method remains clearly task-dependent: it is strong on reaction and wave, but consistently weaker on convection under the current design.

5. DISCUSSION

5.1. Why Convection Degrades

Convection-dominated transport is sensitive to directional structure and often exhibits sharper fronts than reaction or wave settings. The current RoPINN-ResFF design uses isotropic Fourier-feature mapping and generic residual blocks, which may emphasize global frequency fitting while underutilizing advection-specific inductive bias. This mismatch likely contributes to the consistently worse convection errors.

5.2. Practical Takeaway

The method should be presented as a **task-conditional enhancement** of RoPINN:

- statistically significant gains on reaction;
- consistent directional gains on wave;
- clear degradation on convection under the current configuration.

This positioning is scientifically stronger than claiming universal superiority. In addition, reaction budget-sensitivity runs (3000/5000 iterations) indicate that gains persist beyond the 1000-iteration setting.

5.3. Role of Curriculum

Our ablation evidence does not support positioning curriculum as the primary source of gains. In the reaction single-run comparison, RoPINN-ResFF without curriculum is marginally better than the curriculum variant. Therefore, curriculum is retained as an optional training policy and sensitivity axis, while the principal empirical contribution is attributed to the backbone change.

5.4. Submission-Ready Improvement Plan

For a stronger final ICPQC submission, we recommend:

- extending wave/convection seed counts to match the reaction protocol (already 11 seeds) for balanced statistical power across tasks;
- additional convection-focused ablations (sampling mode, sample count, and architecture variants);
- extending budget-sensitivity checks from reaction to wave and convection, together with fixed wall-time accounting, to complete cross-task fairness analysis.

These additions would strengthen the empirical completeness of the final submission.

6. CONCLUSIONS

This paper presents RoPINN-ResFF, an architecture-level enhancement of RoPINN that combines residual MLP blocks and Fourier feature embedding. Under a fixed 1000-iteration budget with multi-seed evaluation, the method delivers statistically significant gains on reaction (11 paired seeds, p=0.00098 for both L1 and L2), directional gains on wave, and a reproducible failure mode on convection. Additional reaction budget-sensitivity runs at 3000 and 5000 iterations further show stable advantages over baseline, indicating that the observed gain is not limited to an early-training regime. Curriculum scheduling is retained as an optional auxiliary policy rather than a primary performance claim.

The key message is that backbone design can materially strengthen region-optimized PINNs, but the benefit is PDE-dependent. Future work will focus on convection-aware inductive bias, balanced multi-seed evaluation across tasks, and compute-fair comparisons to further improve robustness and external validity.

REFERENCES

[1] George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

[2] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. 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, 2019.

[3] Matthew Tancik, Pratul P Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, 2020.

[4] Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021.

[5] Yuxuan Zheng, Bomin Li, Xiang Xie, Lei Cai, Minlin Luo, and Renjie Lu. Ropinn: Region optimized physics-informed neural networks. *arXiv preprint arXiv:2405.14369*, 2024.