Amortized Physics-Informed Learning via Generative Initialization of Radial Basis Functions

Edgar Torres[1]* **Mathias Niepert**[1,2]

Machine Learning and Simulation Lab, Institute for Artificial Intelligence, University of Stuttgart
[1]International Max Planck Research School for Intelligent Systems (IMPRS-IS)
[2]Stuttgart Center for Simulation Science (SimTech)

Abstract

Physics-Informed Neural Networks (PINNs) approximate solutions to partial differential equations (PDEs) in a data-free setting. This work replaces the MLP commonly used in PINNs with Radial Basis Functions (RBFs), leveraging their explicit structure and analytic derivatives to improve training efficiency and solution accuracy. Building on this RBF representation, a generative pretrained initialization model based on variational inference is introduced to further enhance adaptability. Conditioned on PDE attributes, it produces informative RBF kernel parameters that provide strong starting points for PINN training, enabling rapid adaptation to new PDE conditions with minimal fine-tuning and consistently accelerating convergence compared to standard initialization. Experiments on canonical 1D and 2D PDEs demonstrate (1) that RBF-based PINNs can outperform standard MLP-based PINNs in comparable settings and serve as adaptive models, and (2) that variational pretraining can provide effective initialization to enhance training performance. Together, these results validate inference-through-adaptation as a promising direction for scalable, data-efficient, and adaptable PINNs.

1 Introduction

Multi-query problems require repeatedly solving a PDE under varying parameters (e.g., material properties, boundary conditions, forcing terms), a setting common in design optimization, uncertainty quantification, and inverse modeling where accuracy and efficiency are essential [11]. Traditional solvers like FEM incur high cost because each parameter change demands a full re-solve [15]. PINNs mitigate this by embedding the governing equations into the loss, reducing the need for labeled data [16], but standard MLP-based PINNs converge slowly and must be retrained for every new instance [17]. Adaptive modeling approaches address this by exploiting low-dimensional solution manifolds, enabling knowledge reuse across related PDEs [9, 1].

This work introduces a framework that integrates RBFs as adaptive models with a generative hyper-network that provides task-specific initialization within a bilevel optimization setup. The use of RBFs yields explicit and localized representations that support rapid few-step adaptation, while the hyper-network supplies high-quality initial parameters that improve training efficiency across tasks. Experiments on 2D PDEs (Allen–Cahn, Burgers, Convection, Helmholtz) demonstrate that RBFs serve as effective adaptive models and that generative initialization significantly accelerates convergence, strengthening PINN performance in multi-query settings. This combination of hyper-network–driven initialization, RBF-based PINN structure, and bilevel adaptation establishes a practical paradigm for fast, efficient, and adaptive physics-informed learning.

*edgar.torres@ki.uni-stuttgart.de.
Code available at https://github.com/yoelt11/ePIL-RBF

1st Workshop on Differentiable Systems and Scientific Machine Learning @ NeurIPS 2025.

2 Related Work

PINNs incorporate PDE constraints into the loss function, allowing data-efficient learning of physical systems [16]. However, standard MLP-based PINNs often converge slowly and must be retrained for each parameter instance, which limits their scalability in multi-query settings [19]. RBFs offer explicit, localized representations that improve interpretability and stability [2, 7]. Recent studies have enhanced PINN efficiency through meta-learning, transfer learning, and generative initialization [18, 3, 5]. However, their combination with explicit adaptive models, such as RBFs, remains underexplored. Erkocc et al. [6] proposed HyperDiffusion for generative weight prediction, while Kang et al. [12] combined RBFs with PINN losses but without leveraging analytical derivatives. In contrast, our results show that properly initialized RBFs achieve fast convergence and strong accuracy without auxiliary layers. Similarly, Zhang et al. [20] demonstrated the efficiency of analytic RBF interpolation, supporting the potential of explicit representations for scalable PDE solvers.

3 Method

We consider a family of parametric PDEs indexed by a condition vector $c \in \mathcal{C}$, which may encode parameters of the PDE, as well as initial or boundary conditions. Each instance defines a differential operator $\mathcal{N}_c$ and boundary/initial operator $\mathcal{B}_c$ over a spatial domain $\Omega \subset \mathbb{R}^d$ and time interval $\mathcal{T}$:

$$\mathcal{N}_c[u](x,t) = 0 \quad \text{for } (x,t) \in \Omega \times \mathcal{T}, \qquad \mathcal{B}_c[u](x,t) = 0 \quad \text{for } (x,t) \in \partial\Omega \times \{0,T\}. \tag{1}$$

Here $\mathcal{N}_c[u](x,t)$ denotes the residual of the PDE when u is substituted into it (e.g., $\mathcal{N}_c[u] = u_t + \nabla\cdot(a_c\nabla u) - f_c$ for a diffusion problem), and $\mathcal{B}_c[u](x,t)$ enforces boundary or initial conditions such as Dirichlet or Neumann constraints. In the PINN setting, we minimize the residual PDE $\mathcal{N}c[u]$ and the violations of the initial and boundary conditions, rather than relying on labeled solutions.

Adaptive model (anisotropic RBFs). We represent u with K anisotropic Gaussian RBFs for $\xi = [x^\top, t]^\top \in \mathbb{R}^{d+1}$:

$$u(\xi;\theta) = \sum_{k=1}^{K} w_k\, \phi_k(\xi;\lambda_k), \quad \phi_k(\xi) = \exp\Big(-\tfrac{1}{2}(\xi-\mu_k)^\top \Sigma_k^{-1}(\xi-\mu_k)\Big), \tag{2}$$

with parameters $\theta = \{(w_k, \mu_k, \Sigma_k)\}_{k=1}^K$. We use $\Sigma_k = Q_k\,\mathrm{diag}(\sigma_{k,1}^2, \ldots, \sigma_{k,d+1}^2)Q_k^\top$ where Q_k is an orthogonal matrix that encodes the kernel's orientation (e.g., in 2D a rotation matrix). This explicit formulation provides analytic derivatives and exposes interpretable, localized structure, enabling targeted refinement—an advantage over black-box MLP architectures.

Physics-informed objective. We combine residuals evaluated at interior collocation points $\mathcal{X}_r$, boundary points $\mathcal{X}_b$, and initial points $\mathcal{X}_0$, plus optional data $\mathcal{D}$:

$$\begin{aligned}\mathcal{L}_{\text{PINN}}(\theta;c) = {}& \alpha_r\, \mathbb{E}_{(x,t)\in\mathcal{X}_r} \|\mathcal{N}_c[u(\cdot;\theta)]\,(x,t)\|_2^2 + \alpha_b\, \mathbb{E}_{(x,t)\in\mathcal{X}_b} \|\mathcal{B}_c[u(\cdot;\theta)]\,(x,t)\|_2^2 \\ & + \alpha_0\, \mathbb{E}_{(x,t)\in\mathcal{X}_0} \|u(x,t;\theta) - u_0(x)\|_2^2 + \alpha_y\, \mathbb{E}_{(x,t,y)\in\mathcal{D}} \|u(x,t;\theta) - y\|_2^2 .\end{aligned} \tag{3}$$

Generative initialization. While the RBF model in (2) can fit a single PDE instance, it is typically time-consuming to train a large number of instances, especially when solving many related PDEs under varying conditions c. To address this, we introduce a *generative initializer* G_ϕ that learns to predict a good starting point θ_0 for each PDE instance conditioned on c. This amortizes the search for useful kernel locations, scales, and orientations across the training distribution of PDEs. Formally, we define a latent-variable model

$$z \sim q_\phi(z|c) = \mathcal{N}(\mu_\phi(c), \mathrm{diag}\,\sigma_\phi^2(c)), \qquad \theta_0 = G_\phi(z;c), \tag{4}$$

where the encoder (μ_ϕ, σ_ϕ) maps the PDE condition c to a compact latent representation z, and the decoder G_ϕ predicts a full set of RBF parameters $\theta_0 = \{(w_k, \mu_k, \Sigma_k)\}_{k=1}^K$ with each covariance Σ_k parameterized via log-scales and orientation. We implement G_ϕ as a permutation-invariant Set Transformer [13] that maps (z, c) to an unordered set of kernel parameters. This design allows the network to allocate kernels flexibly in space–time according to the PDE structure implied by c. The resulting initialization encodes useful information about the expected solution (e.g., where discontinuities or high-frequency areas are likely to occur), thereby reducing the number of subsequent physics-informed optimization steps required.

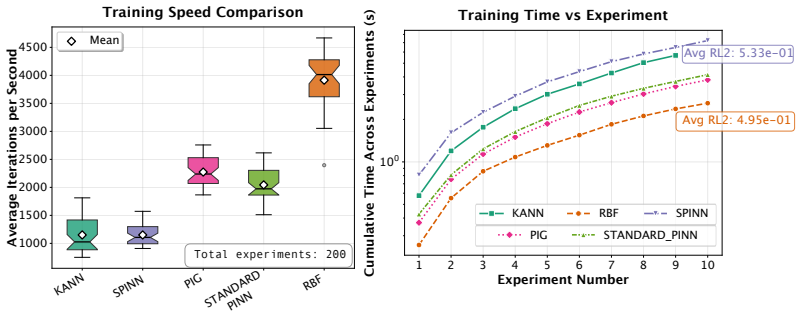

Figure 1: Training speed comparison. Left: iteration rates from 200 experiments, showing the RBF model achieves the fastest per-iteration updates. Right: total training time summed over all 200 runs on the convection equation, highlighting that the RBF model requires the least cumulative computation time.

Bilevel learning with truncated adaptation. The training is performed by solving a bilevel optimization problem. The inner problem adapts θ to minimize the physics-informed loss (3) for a given condition c, while the outer problem updates ϕ so that the initialization $\theta_0 = G_\phi(z; c)$ leads to minimal post-adaptation error. Concretely, for each sampled condition c, we draw a latent code z, compute $\theta_0 = G_\phi(z; c)$, and run T inner optimization steps:

$$\theta_{t+1} = \theta_t - \eta \, \nabla_\theta \mathcal{L}_{\text{PINN}}(\theta_t; c), \qquad t = 0, \ldots, T-1, \tag{5}$$

yielding adapted parameters $\theta^\star = \theta_T$. The outer objective encourages θ_0 to be close to $\theta^\star$, while encouraging latent regularity and diversity among kernels:

$$\mathcal{L}_{\text{outer}}(\phi) = w_{\text{adapt}} \, \|\theta^\star - \theta_0\|_2^2 + \beta \, D_{\text{KL}}\big(q_\phi(z|c) \,\|\, \mathcal{N}(0, I)\big) + \frac{\lambda_{\text{rep}}}{K(K-1)} \sum_{i \neq j} \frac{1}{1 + \|\mu_i - \mu_j\|_2^2 / \sigma_{\text{rep}}^2}. \tag{6}$$

The first term measures the magnitude of adaptation: small values indicate that the generator already predicts near-optimal kernel configurations for that PDE. The KL term regularizes the latent embedding of c, while the last term—the repulsion loss— prevents kernel collapse and encourages spatial coverage. Gradients of $\mathcal{L}_{\text{outer}}$ are computed through the truncated inner loop, similar to meta-learning approaches [8], but with physics-informed residuals as the inner objective.

During inference, the trained generator produces $\theta_0 = G_\phi(\mathbb{E}[z|c]; c)$ (or a single posterior sample), followed by a few gradient steps on (3). This yields a solution operator for the PDE with reduced adaptation cost, effectively performing *amortized physics-informed inference*.

4 Experiments

The RBF-based adaptive model was compared with an MLP-PINN, SPINN [4], PIKANN [14], and PIG [12], all reimplemented in `JAX` with matched parameters and tuned hyperparameters. Figures 1–2 show that the RBF model achieves the best accuracy–efficiency trade-off across benchmarks, with the highest iteration throughput and lowest cumulative optimization time, except for the convection equation, where convergence is slower due to its strong advective term.

4.1 Choice of Adaptive Model

To assess the RBF-based adaptive model, it was compared with four architectures: an MLP-based PINN, SPINN [4], PIKANN [14], and PIG [12]. All baselines were reimplemented in `JAX` with matched parameters and individually tuned hyperparameters. Figures 1–2 present the results. The RBF model consistently delivers the best accuracy–efficiency trade-off, with highest iteration throughput and lowest cumulative optimization time, except for the convection equation, where convergence slows due to the strong advective term.

To verify that slower convergence on the convection equation is not an RBF capacity issue, the same model was trained with a purely data-driven MSE loss. As shown in the right panel of Figure 2, the

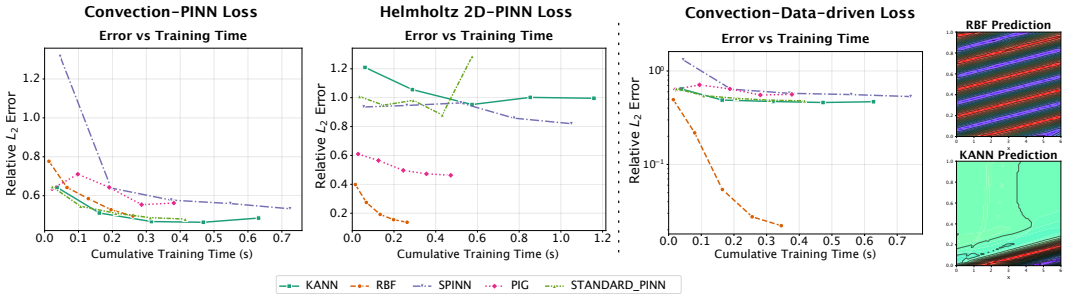

Figure 2: Model comparison across benchmarks (mean over 10 runs). Left: worst-performing case under the PINN loss. Middle: best-performing case under the PINN loss. Right: RBF retains accuracy under pure MSE, confirming model expressivity. Overall, the RBF model achieves the best accuracy–efficiency trade-off.

RBF accurately fits the solution, indicating that the bottleneck arises from the PINN loss landscape rather than model expressiveness. Future work may explore improved optimization, adaptive residual weighting, or leveraging generative-model uncertainty for selective fine-tuning to mitigate these landscape effects.

4.2 Generative Initialization

Generative initialization substantially accelerates the convergence of the convection equation, allowing the RBF to reach a high-quality solution within 50 epochs $(\approx 0.010ms)$.

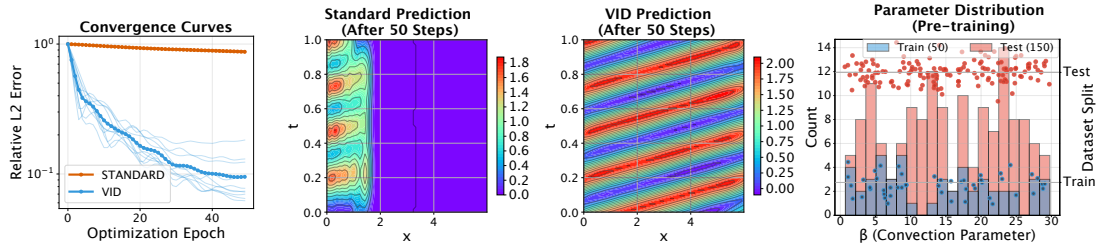

Figure 3: Comparison of standard vs. generative initialization (VID) for the convection equation. Generative initialization reduces the number of optimization steps required to reach convergence, highlighting faster training efficiency.

5 Discussion & Conclusions

This work presents preliminary evidence that combining RBFs with generative initialization yields an efficient adaptive framework for parametric PDEs. RBFs offer fast convergence, interpretable parameters, and closed-form derivatives, while learned initialization significantly reduces optimization time. Slower convergence on certain cases reflects the stiffness of the PINN loss rather than model limits. The framework also enables *inference through refinement*: when design changes are localized, only nearby kernels require updating. This RBF locality supports targeted optimization without retraining, similar to Gaussian-splat refinement [10], providing a reduced-order modeling tool for multi-query PDEs.

Limitations include the current restriction to 2D benchmarks and a narrow PDE set, leaving questions of 3D scalability and generalization to more complex dynamics open. The generative model's uncertainty remains unused but could support selective refinement or adaptive sampling. Future directions include extending the framework to higher-dimensional PDEs, broader dynamical regimes, and richer equation families, as well as exploring improved generative components (e.g., flow matching) to strengthen initialization and robustness. Overall, explicit adaptive models with task-specific initialization offer a promising route to fast and interpretable PDE solvers.

6 Acknowledgements

Funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2075 – 390740016. We acknowledge the support of the Stuttgart Center for Simulation Science (SimTech). The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Edgar Torres and Mathias Niepert. Additionally, we acknowledge the support of the German Federal Ministry of Education and Research (BMBF) as part of InnoPhase (funding code: 02NUK078).

References

[1] Peter Benner, Mario Ohlberger, Albert Cohen, and Karen Willcox. *Model reduction and approximation: theory and algorithms*. SIAM, 2017.

[2] Martin D. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, 2003.

[3] Yanlai Chen and Shawn Koohy. Gpt-pinn: Generative pre-trained physics-informed neural networks toward non-intrusive meta-learning of parametric pdes. *Finite Elements in Analysis and Design*, 228:104047, 2024.

[4] Junwoo Cho, Seungtae Nam, Hyunmo Yang, Seok-Bae Yun, Youngjoon Hong, and Eunbyung Park. Separable physics-informed neural networks. *Advances in Neural Information Processing Systems*, 36:23761–23788, 2023.

[5] Woojin Cho, Kookjin Lee, Donsub Rim, and Noseong Park. Hypernetwork-based meta-learning for low-rank physics-informed neural networks. *Advances in Neural Information Processing Systems*, 36:11219–11231, 2023.

[6] Ziya Erkoç, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Hyperdiffusion: Generating implicit neural fields with weight-space diffusion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 14300–14310, 2023.

[7] Bengt Fornberg and Natasha Flyer. *A primer on radial basis functions with applications to the geosciences*. SIAM, 2015.

[8] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning (ICML)*, pages 1568–1577, 2018.

[9] Stefania Fresca and Andrea Manzoni. Pod-dl-rom: Enhancing deep learning-based reduced order models for nonlinear parametrized pdes by proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, 388:114181, 2022.

[10] Mohamed Abdul Gafoor, Marius Preda, and Titus Zaharia. Refining gaussian splatting: A volumetric densification approach. *arXiv preprint arXiv:2508.05187*, 2025.

[11] Jan S Hesthaven, Gianluigi Rozza, and Benjamin Stamm. *Certified reduced basis methods for parametrized partial differential equations*. SpringerBriefs in mathematics. Springer International Publishing, Cham, Switzerland, 1 edition, August 2015.

[12] Namgyu Kang, Jaemin Oh, Youngjoon Hong, and Eunbyung Park. Pig: Physics-informed gaussians as adaptive parametric mesh representations. *arXiv preprint arXiv:2412.05994*, 2024.

[13] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning (ICML)*, pages 3744–3753, 2019.

[14] Subhajit Patra, Sonali Panda, Bikram Keshari Parida, Mahima Arya, Kurt Jacobs, Denys I. Bondar, and Abhijit Sen. Physics informed kolmogorov-arnold neural networks for dynamical analysis via efficent-kan and wav-kan, 2024.

[15] Alfio Quarteroni, Andrea Manzoni, and Federico Negri. *Reduced basis methods for partial differential equations: An introduction*. UNITEXT. Springer International Publishing, Cham, Switzerland, 1 edition, July 2015.

[16] Maziar Raissi, Paris Perdikaris, and George 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*, 378:686–707, 2019.

[17] Zhiyuan Ren, Shijie Zhou, Dong Liu, and Qihe Liu. Physics-informed neural networks: A review of methodological evolution, theoretical foundations, and interdisciplinary frontiers toward next-generation scientific computing. *Applied Sciences*, 15(14):8092, 2025.

[18] Edgar Torres, Jonathan Schiefer, and Mathias Niepert. Adaptive physics-informed neural networks: A survey. *arXiv preprint arXiv:2503.18181*, 2025.

[19] Sifan Wang, Xinling Yu, and Paris Perdikaris. When and why pinns fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 449:110768, 2022.

[20] Biao Zhang and Peter Wonka. efunc: An efficient function representation without neural networks. *arXiv preprint arXiv:2505.21319*, 2025.