ETH Zurich CAMLab: Training Foundation Models for PDEs with Physics-Informed Loss

Lukass Kellijs
Yale University, ETH Zurich

Write-up

1 Introduction

The goal of this project was to explore the use of physics-informed (PI) losses and their viability for the training and fine-tuning of PDE Foundation Models. Specifically, the aim was to explore fine-tuning the POSEIDON-T [Herde et al., 2024] model with various kinds of PI loss. This work built on the master thesis of Helbling [2024], which showed that using PI losses in training neural operator models may work for very simple settings but fails for even somewhat complex PDEs.

The main work performed during this project can be summarized as follows:

1. Physics-informed losses for Neural Operators: recreation of results of Helbling [2024]
   (a) Poisson Experiment
   (b) Helmholtz Experiment
2. Analysis and improvements of the work of Helbling [2024]
   (a) Exploration of non-autograd derivative approaches
   (b) Deeper exploration of the Helmholtz problem
3. Physics-informed losses for foundation models: fine-tuning of the POSEIDON-T model architecture with various physics-informed approaches:
   (a) Pure unsupervised physics fine-tuning
   (b) Separate supervised data "pre-tuning" and unsupervised physics fine-tuning
   (c) Hybrid (data and physics) fine-tuning

2 Motivation

2.1 Higher accuracy

Perhaps the key motivation behind using physics-informed losses is that they may improve the final accuracy and generalization of the models. The hope is that by setting constraints on the derivatives and boundaries of the model predictions, the actual predictions may become more physical and, as a byproduct, the accuracy of the predictions themselves could be improved and, equipped with the actual physical law information, could better generalize to more out-of-distribution (OOD) samples.

2.2 Data-efficiency

The other motivation behind using a PI loss is that it may improve data-efficiency—the amount of necessary data samples to train or fine-tune a neural operator may be decreased. This is relevant in the field of Scientific Machine Learning as generating detailed sample solutions requires computationally expensive numerical simulations.

Specifically, so-called foundation PDE models (models that are pre-trained on a large dataset of a wide variety of PDE problems) require fine-tuning on downstream tasks to produce acceptable results. If PI losses could be used in fine-tuning these models, it would greatly improve their usability.

3 Physics-informed losses for Neural Operators

To begin with, we explore the use of PI losses when training neural operators for two simple PDE equations

$$\text{Poisson Equation} \qquad \nabla^2 u = f \tag{1}$$

$$\text{Helmholtz Equation} \qquad \nabla^2 u + \omega^2 a^2 u = 0 \tag{2}$$

We follow the work of Helbling [2024] and use the CNO [Raonić et al., 2023] and FNO [Li et al., 2022] neural operator architectures.

For both problems the models were trained on the same training dataset with $N = 1024$ samples using three different approaches and corresponding loss functions:

1. **Data-Driven Approach:** trained on only the supervised data loss $\mathcal{L}_{\text{data}}$
2. **Hybrid Approach**: trained with N samples using the supervised loss $\mathcal{L}_{\text{data}}$ and then fine-tuned on the same N samples with an unsupervised loss of the form $\mathcal{L}_{\text{phys}} + \lambda \mathcal{L}_{\text{anch}}$
3. **Pure-Physics Approach:** trained only on the unsupervised physics loss $\mathcal{L}_{\text{phys}}$

3.1 Poisson Equation

The Poisson problem was formulated on a 64×64 grid, with Dirichlet boundary condition of zero $u|_{\partial\Omega} = 0$ on the boundary. In (1), $u(x, y)$ is the scalar solution field to be predicted, and $f(x, y)$ is the source term provided as input to the neural operator.

The physics-informed loss consists of two components:

$$\mathcal{L}_{\text{phys}} = \underbrace{\| -\Delta u - f \|}_{\text{PDE loss}} + \beta \underbrace{\|u|_{\partial\Omega}\|}_{\text{Boundary loss}} ,$$

where Δu is the Laplacian of the predicted solution, computed via a method of choice, and β is a weight controlling the strength of the boundary enforcement. This loss penalizes deviations from both the PDE and the prescribed boundary condition. The data-driven loss is simply the L1 norm of the residual $\mathcal{L}_{\text{data}} = \|u - u^*\|$.

In the hybrid setting an anchor loss $\mathcal{L}_{\text{anch}} = \|u_m - u^*\|$, where u_m is the prediction of the pre-trained model, acts as a regularizing term. This loss penalizes the difference in predictions between a pre-trained and fine-tuned model and thus constrains the fine-tuned model to not diverge too far away from the pre-trained model.

The results of the training are summarized in the table below. (The baseline FNO and CNO models are trained exactly as in Helbling [2024].)

Table 1: Relative test errors (%) for Poisson equation experiments across training strategies and model variants.

Training Mode	**FNO**	**FNO (Improved)**	**CNO**	**CNO (Improved)**
Data-driven	5.41	—	1.74	—
Hybrid (Pretrained + PI-loss)	6.58	**4.98**	1.83	**0.95**
Pure-physics	13.75	10.63	3.56	2.72

As can be seen, the CNO generally performs better than the FNO model in all settings. Regarding PI losses, the *Pure-Physics* approach leads to a worse fine-tuning outcome than the *Data-Driven* approach, but adding an additional unsupervised training stage in the *Hybrid* approach leads to the

lowest relative test error for both models. The *Improved* models (with exact improvements discussed in Section 4) also significantly improve model performance.

These results suggest that for the simple Poisson problem, PI losses may be effectively leveraged to train neural operators in an unsupervised setting, even with no pre-training.

3.2 Helmholtz Equation

The Helmholtz problem was formulated on a 128×128 grid, with constant Dirichlet boundary condition of $u|_{\partial\Omega} = b$ on the boundary. In (2), $u(x, y)$ is the scalar solution field to be predicted and $a(x, y)$ is a spatially varying coefficient field provided as an input channel. A fixed frequency $\omega = \frac{5\pi}{2}$ is used throughout all experiments. The neural operator receives both b and a as inputs and outputs the predicted solution u^* over the domain.

The physics-informed loss is an extension of that used for the Poisson equation:

$$\mathcal{L}_{\text{phys}} = \underbrace{\| -\Delta u - \omega^2 a u \|}_{\text{PDE loss}} + \beta \underbrace{\| u|_{\partial\Omega} - u^*|_{\partial\Omega} \|}_{\text{Boundary loss}},$$

where Δu is the Laplacian of the predicted solution and β is the boundary loss weight. As in the Poisson case, the data-driven loss is defined as $\mathcal{L}_{\text{data}} = \|u - u^*\|$, and the anchor loss $\mathcal{L}_{\text{anch}} = \|u_m - u^*\|$ is used in hybrid training as a regularizing term.

The results of the training are summarized in the table below.

Table 2: Relative test errors (%) for Helmholtz equation experiments across training strategies and model variants.

Training Mode	**FNO**	**FNO (Improved)**	**CNO**	**CNO (Improved)**
Data-driven	**16.43**	—	**14.59**	—
Hybrid (Pretrained + PI-loss)	>100	16.58	>100	14.59
Pure-physics	>100	75.55	>100	>100

As can be seen, the Helmholtz equation posed a significantly greater challenge for both neural operator architectures. Neither model was able to converge in the *Pure-Physics* setting, even when implementing improvements over previous work in Helbling [2024] (which also failed to reach convergence). Likewise, it was found that the *Hybrid* approach could not improve upon the supervised *Data-Driven* baseline and could only reach similar accuracy in the most optimal of cases (which could just as well be achieved by letting the anchor term dominate, i.e., letting $\lambda \to \infty$).

4 Improvements

The baseline *FNO* and *CNO* models used to obtain the results above were trained exactly according to Helbling [2024]. One goal of this work was to explore improvements in the training process and parameters (keeping the model architecture invariant).

4.1 Using Finite-Difference Laplacian

In [Helbling, 2024], the failure of PI losses for the *Helmholtz* problem was attributed to oscillations near the boundaries when calculating the Laplacian using Fourier-based numerical differentiation (due to the Gibbs phenomenon; see Figure 1).

To mitigate these artifacts, alternative differentiation methods were considered. While *automatic differentiation* could in principle provide accurate gradients, it is not applicable in our case, as the coordinate variables (x, y) are not part of the computational graph in the FNO, CNO, or POSEIDON architectures. Consequently, we explore a simple finite-difference (FD) method for Laplacian estimation, which consists of a 2D convolutional layer with finite-difference stencils as kernels.

We evaluated several FD stencils for discretizing the Laplace operator, including the 9-point Patra–Karttunen, 9-point Oono–Puri, and standard 13-point schemes. The errors were evaluated by

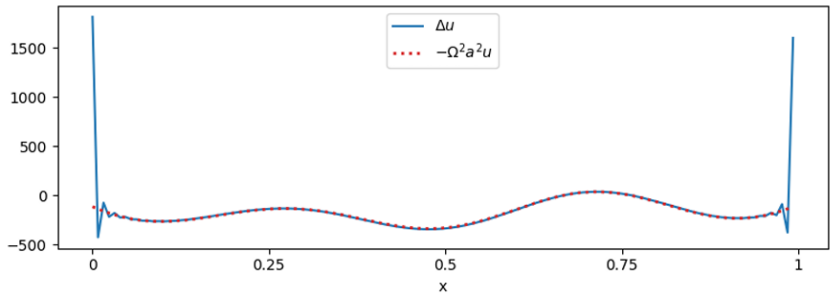


Figure 1: (From [Helbling, 2024]) Visualizing numerical errors: comparison of the Laplacian operator acting on the ground truth u, computed using numerical Fourier differentiation, against its corresponding counterpart in the Helmholtz equation (slice in the x-direction).

Table 3: Average relative error of Laplacian approximation across different differentiation methods.

Method	**Stencil Type**	**Average Relative Error (%)**
Finite Difference	9-point Patra–Karttunen	**3.00**
Finite Difference	9-point Oono–Puri	4.09
Finite Difference	13-point standard	3.82
Fourier-based	—	54.39

comparing the Laplacian of the known solution to that of the numerically simulated Laplacian in the dataset (which is simply the negative of the input function f for the Poisson case).

Furthermore, to deal with the boundary, we excluded boundary pixels from the PDE loss term and only applied the boundary loss to these boundary pixels, implicitly "cropping" the domain used for calculating the PDE loss. See Figure 2 for a visualization of the FD improvement over the Fourier approach.

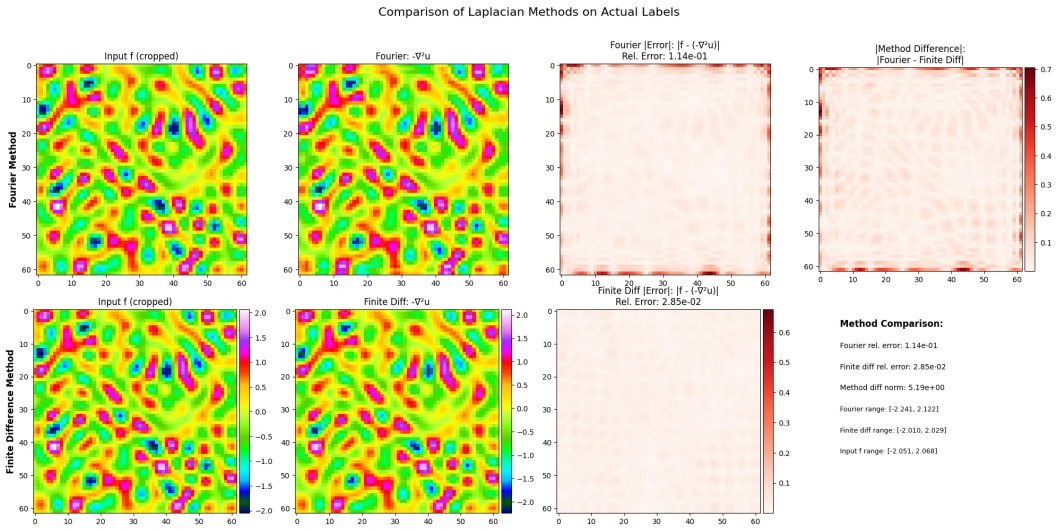


Figure 2: Comparison of FD and Fourier approaches to calculate Laplacians of known labels.

4.2 Proper weighting of boundary term

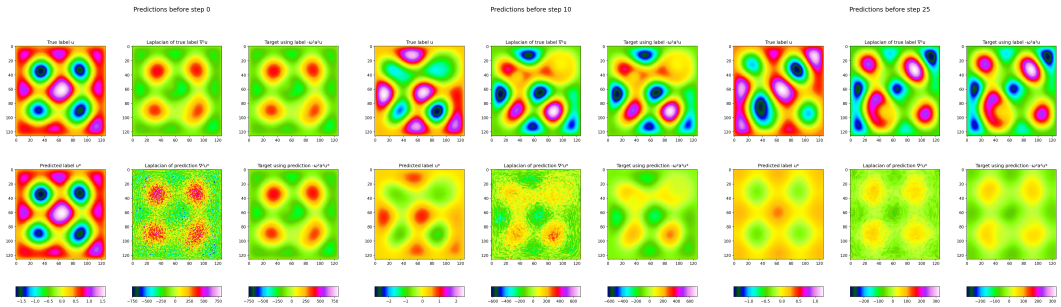
In early experiments, the boundary loss term $\|u|_{\partial\Omega}\|$ was underweighted relative to the PDE loss, often resulting in poor enforcement of the Dirichlet boundary condition. This imbalance contributed to instability and degraded generalization, especially near the domain edges.

To address this, we introduced a boundary weight hyperparameter β and performed empirical tuning. We found that increasing β such that the boundary loss magnitude was maintained at about 10% of that of the PDE loss term improved both convergence and final test accuracy.

4.3 Exploration of Helmholtz problem divergence

We explored various tweaks and improvements to attain convergence for the physics-informed setting of the Helmholtz equation (2). The main question we ask in this section is "Why does the Helmholtz problem not converge?".

Converging to non-optimal (mean) solutions The first test was to visualize the relevant predictions and their derivatives at various points in the model training. This is done in Figure 3, where the unsupervised PI fine-tuning portion of a hybrid training approach is visualized. As such, we can see how the predictions of the model change.



(a) Step 0 (b) Step 10 (c) Step 25

Figure 3: Visualizations of Helmholtz equation label/prediction u, Laplacian of label/prediction ∇u, and label/prediction "target" $\omega^2 a^2 u$ before conducting update steps on a model that has already been trained in a supervised setting.

What was discovered is that through PI fine-tuning the model can start to converge to "mean" predictions: rather than converging to the real unique solution, it finds "pseudo-solutions" which provide decent agreement between ∇u^* and its target as set by the Helmholtz equation $-\omega^2 a^2 u^*$, thus minimizing the PDE loss. There are two proposed interrelated reasons for why this may be:

1. **Moving target:** Since the "target" itself includes the prediction, which itself is noisy, this could lead the model astray from the actual physical solution.
2. **Noisy Laplacian:** Since the Laplacian essentially acts as a high-pass filter, any noise in the prediction gets amplified, and the model then uses this in the target.

Exploring Hybrid Tuning Approaches One way to deal with the aforementioned issues that was explored was to also add some weighted data-driven term to the loss $\mathcal{L}_{\text{data}}$. This, naturally, makes the fine-tuning no longer unsupervised, but it was hoped that through this method at least the model could attain better accuracy and generalization properties. Unfortunately, the PI loss term $\mathcal{L}_{\text{phys}}$ still seemed to nudge the model towards worse solutions, and no improvements in solution accuracy were observed, apart from lower errors between Laplacians of predictions and labels.

5 Physics-informed losses for foundation models: the Poisson problem

To evaluate the viability of using physics-informed losses to fine-tune POSEIDON-T, a PDE neural operator foundation model, we explored **two different applications**:

1. **Using PI losses for unsupervised fine-tuning:** exploring the benefit of using PI losses to train the model in an unsupervised setting
2. **Using PI losses as a regularizer in hybrid tuning:** exploring the benefit of using a regularizing term to improve the final accuracy in a supervised setting

For approach 1, to proactively deal with the problem of convergence we also explored **"pre-tuning"** the model with some amount of data in a supervised setting, before **"fine-tuning"** it in a fully unsupervised setting. *This was done because for some very out-of-distribution tasks the foundation model could give very incorrect predictions: it is assumed that the initial pre-tuning serves as a means of nudging the model in the right direction before using the more difficult-to-optimize unsupervised setting.*

5.1 The Poisson problem

As the main test, the POSEIDON-T model was used with the Poisson equation in (1). Unless stated otherwise, both the pre-tuning and fine-tuning were conducted with the same training parameters as in the table below.

Table 4: Fine-tuning hyperparameters and training settings.

Parameter	**Value**	**Parameter**	**Value**
Optimizer	AdamW	Weight decay	10^{-6}
Scheduler	Cosine Decay	Batch size	40
General learning rate $\tilde{\eta}_f$	$5 \cdot 10^{-5}$	Number of epochs	200
Embedding/recovery $\tilde{\eta}_\nu$	$5 \cdot 10^{-4}$	Early stopping	No
LayerNorm learning rate $\tilde{\eta}_\mathcal{N}$	$5 \cdot 10^{-4}$	Gradient clipping	5

5 models with differing numbers of trajectories (labeled samples) were pre-tuned, which **also serve as a benchmark**.

Table 5: Relative test errors (%) of pre-tuned POSEIDON-T models for the Poisson equation.

Trajectories Used for Pre-tuning	**Best L1 Relative Eval Error (%)**
$n = 0$	> 100
$n = 16$	41.91
$n = 128$	11.54
$n = 1024$	6.43
$n = 8192$	1.06

A visualization of the $n = 8192$ model's predictions is given below:

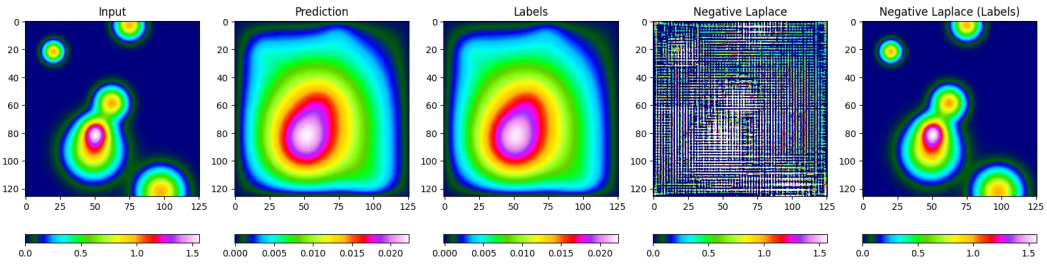


Figure 4: Predictions of $n = 8192$ pre-tuned model.

Although visually the model's predictions seem quite accurate, the Laplacian of the predictions (4th image from the left) is very noisy in a grid-like structure (compare with the same finite-difference Laplacian of the labels in the right-most figure). This is likely tied to the architecture of the POSEIDON-T model itself, which uses a U-net–like vision transformer (the grid-like patches correspond to the patches formed by de-embedding the tokens).

5.2 No pre-tuning

The $n = 0$ model corresponds to no pre-tuning. In this setting we only fine-tune the foundation model with unsupervised physics-informed loss.

After trying out different amounts n_f of unlabeled fine-tuning samples, there was an observable trend of convergence for the $n_f = 8192$ case. Finally, after increasing the initial general learning rate ($\tilde{\eta}_f \to 5 \cdot 10^{-4}$) and the initial recovery/embedding and LayerNorm rates ($\tilde{\eta}_\nu \to 1 \cdot 10^{-3}$ and $\tilde{\nu}_\mathcal{N} \to 1 \cdot 10^{-3}$), we get the model to converge with best relative evaluation error of 1.064%, comparable to the lower-learning-rate and data-driven pre-tuned model of $n = 8192$ in Table 5.

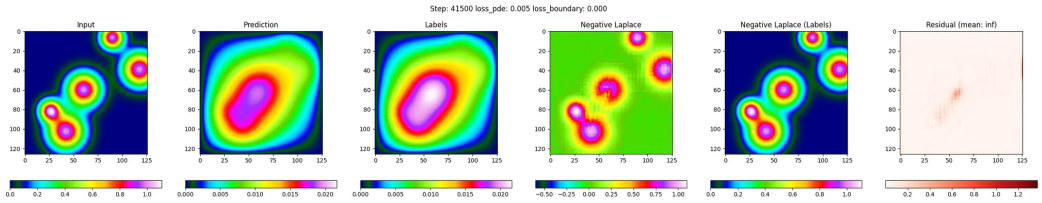


Figure 5: Predictions of $n = 0$ pre-tuned model after fine-tuning with $n_f = 8192$ unlabeled samples at higher learning rate.

As can be seen, while the accuracy is comparable, the visual results of the direct predictions seem less accurate, while the visual results of the Laplacian are much better (*note the different color scales*).

5.3 Lack of scaling law for the $n = 128$ model

The $n = 128$ pre-tuned model in particular was fine-tuned on different amounts of unlabeled trajectories. The following accuracies were obtained:

Table 6: Relative test errors (%) for different numbers of fine-tuning trajectories for the $n = 128$ model.

Trajectories Used for Fine-tuning	**Best L1 Relative Eval Error (%)**
$n = 1024$	7.02
$n = 2048$	7.09
$n = 4096$	7.06
$n = 8192$	7.35
$n = 16384$	7.25

Thus, while the PI loss was able to lead to convergence in a pure-physics setting, for the $n = 128$ case the accuracy did not increase with increasing amounts of unsupervised data. (These scaling laws merit further research, for example, running further tests with different pre-tuned models.)

5.4 Hybrid training

In evaluating the second application, a "hybrid" fine-tuning approach was used with the loss defined as $\mathcal{L}_{\text{phys}} + \lambda \mathcal{L}_{\text{data}}$ where an optimal λ was empirically found to be $\lambda = 10$. In addition, the higher learning rate as in Section 5.2 was used. The predictions are visualized in the figure below:

The resulting evaluation error of 1.516% for the hybrid model is lower than that of a benchmark model trained with $n = 1024$ purely data-driven samples at the same higher learning rate, which achieved an evaluation error of 2.264%. Also, visually comparing the Laplacian with predictions of other data-driven models (see Figure 4 for example), the derivatives of the predictions are also much improved. Similar results were observed for other settings, meaning that PI losses could be used in conjunction with labeled samples to improve the accuracy and quality of derivatives of a model's predictions in the fine-tuning phase.

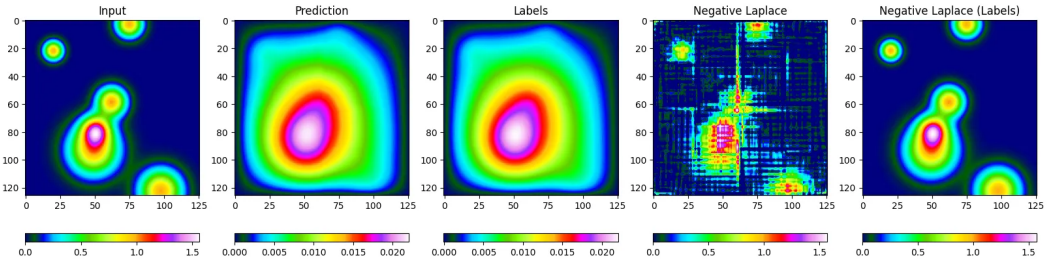


Figure 6: Predictions of a fine-tuned model with $n = 1024$, hybrid loss, and higher learning rate.

6 Summary of main findings

- **Limited Applicability:** Physics-informed losses have so far been shown to work in fully unsupervised settings only for very simple problems such as the Poisson equation. For more complex equations like Helmholtz, the loss landscape becomes difficult to optimize and models typically fail to converge to the true solution.
- **Spikes in Training:** Training neural operators with physics-informed losses often results in an initial spike in the PDE loss, indicating instability at the beginning of training. When using PDE loss terms in fine-tuning, the term should be weighted carefully, and it should be validated that the first few update epochs do not destabilize the model and lead to a spike in evaluation error (using gradient clipping or lower learning rates could help in this evaluation). In addition, the causes and ways to deal with this spiking behavior should be further explored.
- **Non-physicality of Fine-tuned NOs:** Purely data-driven fine-tuning using small datasets can lead to noisy predictions, especially in the derivatives, due to insufficient regularization or data diversity.
- **Relevance of Differentiation Approaches:** The effectiveness of physics-informed losses may be fundamentally limited by the accuracy of the numerical differentiation method used to compute the residuals. Thus PI losses may be most effectively used in models where the independent variables of the PDE are part of the computational graph, enabling the use of automatic differentiation for the computation of the physics-informed loss.
- **Higher Learning Rates for OOD Tasks:** In the POSEIDON paper, out-of-distribution downstream tasks such as Poisson and Helmholtz were found to benefit from higher learning rates during fine-tuning.
- **Potential in Hybrid Tuning:** Physics-informed losses may serve as weak regularizers (when assigned a small weight), improving the smoothness and quality of the model's predicted derivatives—even if they do not directly improve solution accuracy. Thus, such hybrid fine-tuning losses should be further explored for different datasets, as they could provide more physical solutions.

References

Arno Helbling. *Preconditioning and Fine-Tuning Physics-Informed Neural Operators*. PhD thesis, ETH Zurich, March 2024.

Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bézenac, and Siddhartha Mishra. Poseidon: Efficient Foundation Models for PDEs, November 2024. URL http://arxiv.org/abs/2405.19101. arXiv:2405.19101 [cs].

Zongyi Li, Miguel Liu-Schiaffini, Nikola Kovachki, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Learning Dissipative Dynamics in Chaotic Systems, September 2022. URL http://arxiv.org/abs/2106.06898. arXiv:2106.06898 [cs].

Bogdan Raonić, Roberto Molinaro, Tim De Ryck, Tobias Rohner, Francesca Bartolucci, Rima Alaifari, Siddhartha Mishra, and Emmanuel de Bézenac. Convolutional Neural Operators for robust and accurate learning of PDEs, December 2023. URL http://arxiv.org/abs/2302.01178. arXiv:2302.01178 [cs].