Multifidelity Finite Basis Physics Informed Neural Networks

Name: Damien Beecroft

Hosting Site: Pacific Northwest National Lab

Mentors: Amanda Howard and Panos Stinis

Mentor's Signature:

Abstract

Physics informed neural networks (PINNs) struggle to successfully learn solutions to differential equations that exhibit high-frequency oscillations or multi-scale behavior. Multilevel finite basis physics informed neural networks (FBPINNs) tackle this problem by recursively discretizing the solution domain and training coupled neural networks on the subdomains. High frequency structures are made less oscillatory with respect to the length scale of these smaller subdomains. This plays to the spectral bias of PINNs and enables one to learn the global solution faster. In this work we integrate multifidelity methods into multilevel FBPINNs to further improve the learning rate. This internship was virtual. I created a GitHub project to communicate what I am working on with my mentors. Anyone interested in solving differential equations could benefit from this project.

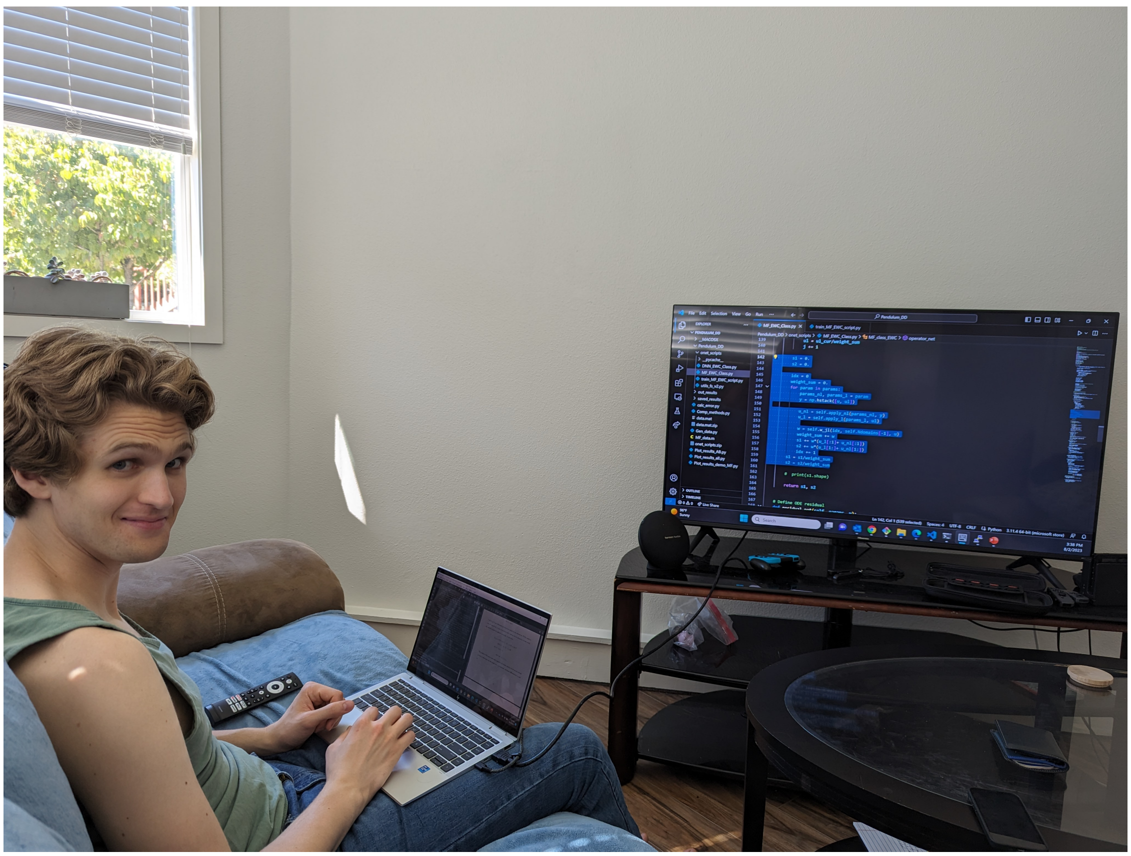

Figure 1: My internship at Pacific Northwest National Lab was virtual. My office was anywhere I lugged my computer. On some of my more adventurous days I ventured all the way down to my living room to use the television as a secondary monitor. On an even more auspicious occasion my girlfriend and soon to be doctor Caitlin Neher was kind enough to take my photo for this final report.

1 Introduction

Raissi et al. [3] invented the physics informed neural network (PINN) in 2016. This framework enables machine learning algorithms to learn solutions to differential equations with just the physical constraints. The dream is to achieve a flexible differential equation solvers that are able to find high accuracy solutions with little expert knowledge. However, this dream is still a ways off. We briefly review the concept of a PINN here. Suppose that we have the following differential equation.

$$\begin{aligned} u(t,x)_t + N[u(t,x)] &= 0 \quad \text{for} \quad x \in \Omega,\ t \in [0,T] \\ B[u(t,x)] &= 0 \quad \text{for} \quad x \in \partial\Omega \\ u(0,x) &= u_0(x) \end{aligned}$$

N and B are known differential operators that contain no time derivatives. PINNs construct a neural network $\tilde{u}(t,x)$ that, is penalized every training step in proportion to how poorly the network satisfies the physical constraints. In particular,

$$\begin{aligned} \tilde{u}(t,x)_t + N[\tilde{u}(t,x)] &= l_1 \quad \text{for} \quad x \in \Omega,\ t \in [0,T] \\ B[\tilde{u}(t,x)] &= l_2 \quad \text{for} \quad x \in \partial\Omega \\ \tilde{u}(0,x) - u_0(x) &= l_3 \end{aligned}$$

$$\text{total loss} = l_1 + l_2 + l_3$$

The gradient of the loss is taken with respect to the hidden variables of the neural network in order to determine how the weights are to be adjusted. Automatic differentiation is used to differentiate the neural network with respect to the temporal and spatial variables. PINNs struggle to learn oscillatory and multi-scale solutions, often converging to fixed point solutions of the differential equation [4]. This is often due to the fact that fixed point solutions satisfy the physical constraints of the differential equation while also maintaining small hidden weights. The boundary and initial conditions do not have enough influence to push the network away from steady state solutions. There are a variety of methods and techniques that have been introduced to improve the convergence of PINNs. My research was focused on combining two such methods. The first method is the multifidelity PINN framework introduced by Meng et al. [2]. Multifidelity PINNs learn from a high and low fidelity data source. The network learns the correlation between the two data sources to achieve an accurate prediction of the true solution. The second is the multilevel finite basis PINNs as introduced in Dolean et al. [1]. This construction takes the entire domain of the problem and decomposes it into a set of overlapping subdomains. A neural network is trained on each of these subdomains. My task this summer was to combine these two methods and test the results.

References

[1] Victorita Dolean, Alexander Heinlein, Siddhartha Mishra, and Ben Moseley. Multilevel domain decomposition-based architectures for physics-informed neural networks, 2023.

[2] Xuhui Meng and George Em Karniadakis. A composite neural network that learns from multi-fidelity data: Application to function approximation and inverse PDE problems. *Journal of Computational Physics*, 401:109020, jan 2020.

[3] Maziar Raissi, Paris Perdikaris, and George E. Karniadakis. Physics informed deep learning (part I): data-driven solutions of nonlinear partial differential equations. *CoRR*, abs/1711.10561, 2017.

[4] Franz M. Rohrhofer, Stefan Posch, Clemens Gößnitzer, and Bernhard C. Geiger. On the role of fixed points of dynamical systems in training physics-informed neural networks, 2023.