REPORT ON THE PAPER "POSEIDON: EFFICIENT FOUNDATION MODELS FOR PDES"

DEVANSH TRIPATHI[1]
ETH Zürich

ABSTRACT. In the paper [2], the author introduces a foundation model POSEIDON, for learning the solution operators of PDEs. It is based on a multiscale operator transformer, with time-conditioned layer norms that enable continuous-in-time evaluations. They propose a novel training strategy leveraging the semi-group property of time-dependent PDEs to allow for significant scaling-up of the training data. POSEIDON is a pretrained model on a diverse, large scale dataset for the governing equations of fluid dynamics. The authors show that POSEIDON exhibits excellent performance by outperforming baselines significantly, both in terms of sample efficiency and accuracy. They also the generalization ability of the model to unseen physics.

1. Introduction

Partial Differential Equations (PDEs) are referred to as the language of physics as they mathematically model a very wide variety of physical phenomena across a vast range of spatio-temporal scales. Numerical methods such as finite difference, finite element, spectral methods etc. are commonly used to approximate or simulate PDEs. However, their (prohibitive) computational cost, particularly for the so-called many-query problems, has prompted the design of various data-driven machine learning (ML) methods for simulating PDEs. Among them, operator learning algorithms have gained increasing traction in recent years.

These methods aim to learn the underlying PDE solution operator, which maps function spaces inputs (initial and boundary condition, coefficients, sources) to the PDE solution. They include algorithms which approximate a *discretization*, on a fixed grid, of the underlying solution operator. These can be based on convolutions [11], graph neural network [6, 9] or transformers [7, 1]. Other operator learning alogorithms are *neural operators* which can directly process function space input and outputs, possibly sampled on multiple grid resolutions. These include DeepONets [5], Fourier Neural Operator [3], CNO [8], among many others.

How can the number of training samples for PDE learning be significantly reduced? *Foundation models* are *generalist* models that are pretrained, at-scale, on laarge datasets drawn from a diverse set of data distributions. They leverage the intrinsic abilitiy of neural networks to learn *effective representations* from pretraining and are then successfully deployed on a variety of *downstream* tasks by *finetunning* them on a few task-specific samples. Example of such models include highly successful large language models [10].

The challenge of designing such foundation models for PDEs is formidable given the sheet variety of PDEs and data distributions. Authors concur that the feasibility of of designing PDE foundation models rests on the fundamental and unanswered science question of *why pretraining a model on a very small set of PDEs and underlying data-distributions can allow it to learn effective representations and generalize to unseen and unrelated PDEs and data-distributions via finetuning?*

The Poseidon family of PDE foundation models are based on i) scalable Operator Transformer or scOT, a *multiscalae vision transformer* with (shifted) windowed or Swin attention [4], adapted

[1] Seminar für Angewandte Mathematik, HG E 62.2, Rämistrasse 101, 8092 Zürich, Switzerland
devansh.tripathi@sam.math.ethz.ch.

for operator learning, ii) a novel all2all training strategy for efficient leveraging *trajectories* of solutions of time-dependent PDEs to scale up the volume of training data and iii) an open source large-scale pretraining dataset, containing a set of novel solution operators of the compressible Euler and incompressible Navier-Stokes equations of fluid dynamics.

2. Approach

Problem Formulation. We denote a generic time-dependent PDE as,

$$\begin{aligned} &\partial_t u(x,t) + \mathcal{L}(u, \nabla_x u_x, \nabla_x^2 u, \dots) = 0, \quad \forall x \in D \subset \mathbb{R}^d, t \in (0,T), \\ &\mathcal{B}(u) = 0, \quad \forall (x,t) \in \partial D \times (0,T), \quad u(0,x) = a(x), \quad x \in D \end{aligned} \tag{2.1}$$

Here, with a function space $\mathcal{X} \subset L^p(D;\mathbb{R}^n)$ for some $1 \leq p < \infty, u \in C([0,T];\mathcal{X})$ is the solution of ((2.1)), $a \in \mathcal{X}$ the initial datum and $\mathcal{L}, \mathcal{B}$ are the underlying differential and boundary operators, respectively. Note that (2.1) accomodates both PDEs with high-order time derivatives as well as PDEs with (time-independent) coefficients and sources by including the underlying functions within the solution vector and augmenting $\mathcal{L}$ accordingly.

Even *time-independent* PDEs can be recovered from (2.1) by taking the *long-time limit*, i.e., $\lim_{t\to\infty} u = \overline{u}$, which will be the solution of the (generic) time-independent PDE,

$$\mathcal{L}(\overline{u}(x), \nabla_x \overline{u}, \nabla_x^2 \overline{u}, \dots) = 0, \quad \forall x \in D, \quad \mathcal{B}(\overline{u}) = 0, \quad \forall x \in \partial D. \tag{2.2}$$

Solutions of the PDE (2.1) are given in terms of the underlying *solution operator* $\mathcal{S} : [0,T] \times \mathcal{X} \mapsto \mathcal{X}$ such that $u(t) = \mathcal{S}(t,a)$ is the solution of 2.1 at any time $t \in (0,T)$. Given a data distribution $\mu \in Prob(\mathcal{X})$, the *underlying operator learning task (OLT)* is,

> **OLT**: *Given any initial datum $a \sim \mu$, find an approximation $\mathcal{S}^* \approx \mathcal{S}$ to the solution operator $\mathcal{S}$ 2.1, in order to generate the entire solution trajectory $\{\mathcal{S}^*(t,a)\}$ for all $t \in [0,T]$.*

It is essential to emphasize here that the learned opeator $\mathcal{S}^*$ has to generate the *entire solution trajectory for 2.1, given only the initial datum (and boundary conditions),* as this is what the underlying solution operator $\mathcal{S}$ (and numerical approximation to it) does.

Model Architecture. The backbone for the POSEIDON foundation model is provided by scOT or *scalable Operator Transformer,*. scOT is a *hierarchical multiscale vision transformer with lead-time conditioning* that processes lead time t and function space valued initial data input a to appropriate the solution operator $\mathcal{S}(t,a)$ of the PDE 2.1.

Appendix A. Architecture of the scalable Operator Transformer (scOT)

A.1. Operator Learning with scOT.

References

[1] Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. GNOT: A general neural operator transformer for operator learning, 2023. URL `https://arxiv.org/abs/2302.14376`.

[2] Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bézenac, and Siddhartha Mishra. Poseidon: Efficient foundation models for pdes, 2024. URL `https://arxiv.org/abs/2405.19101`.

[3] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations, 2021. URL `https://arxiv.org/abs/2010.08895`.

[4] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021. URL `https://arxiv.org/abs/2103.14030`.

[5] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, March 2021. ISSN 2522-5839. doi: 10.1038/s42256-021-00302-5. URL `http://dx.doi.org/10.1038/s42256-021-00302-5`.

[6] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W. Battaglia. Learning mesh-based simulation with graph networks, 2021. URL `https://arxiv.org/abs/2010.03409`.

[7] Michael Prasthofer, Tim De Ryck, and Siddhartha Mishra. Variable-input deep operator networks, 2022. URL `https://arxiv.org/abs/2205.11404`.

[8] 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, 2023. URL `https://arxiv.org/abs/2302.01178`.

[9] Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. Learning to simulate complex physics with graph networks, 2020. URL `https://arxiv.org/abs/2002.09405`.

[10] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. URL `https://arxiv.org/abs/2302.13971`.

[11] Yinhao Zhu and Nicholas Zabaras. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, 366:415–447, August 2018. ISSN 0021-9991. doi: 10.1016/j.jcp.2018.04.018. URL `http://dx.doi.org/10.1016/j.jcp.2018.04.018`.