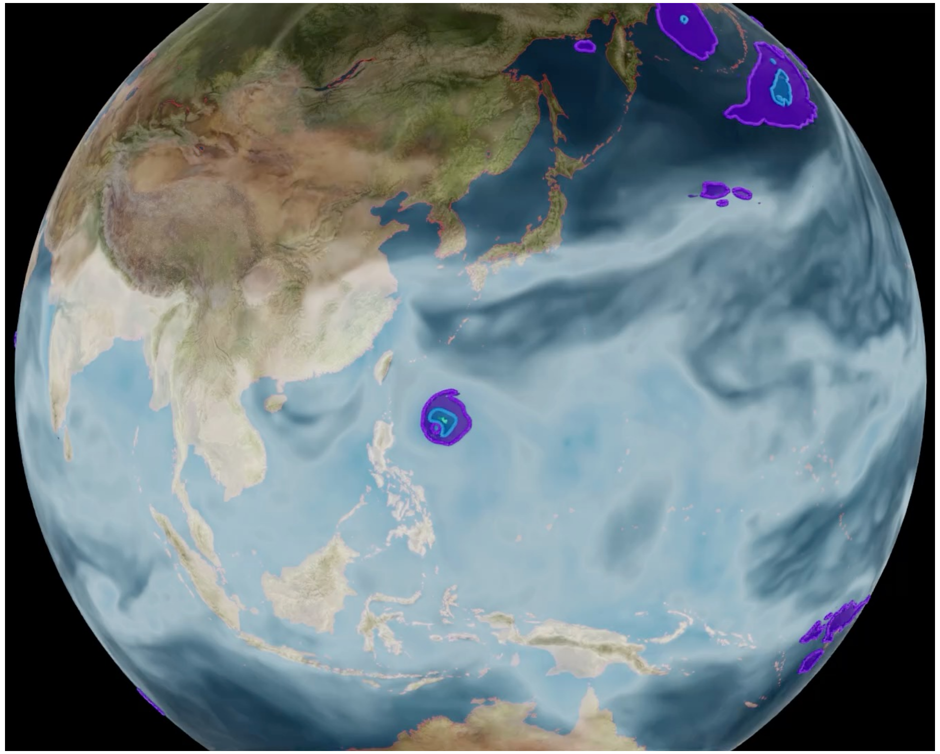
A gentle introduction to neural operator

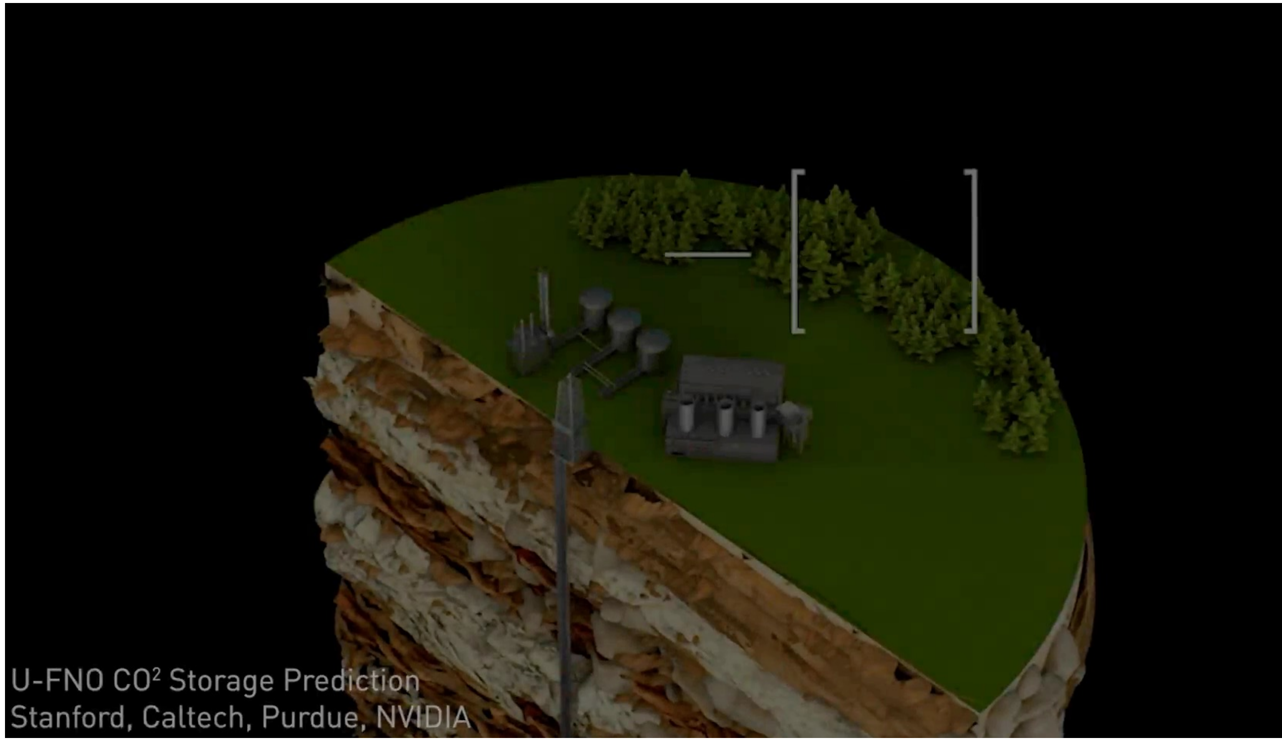
Zongyi Li

Aug 2023

SZ4D Virtual Workshop



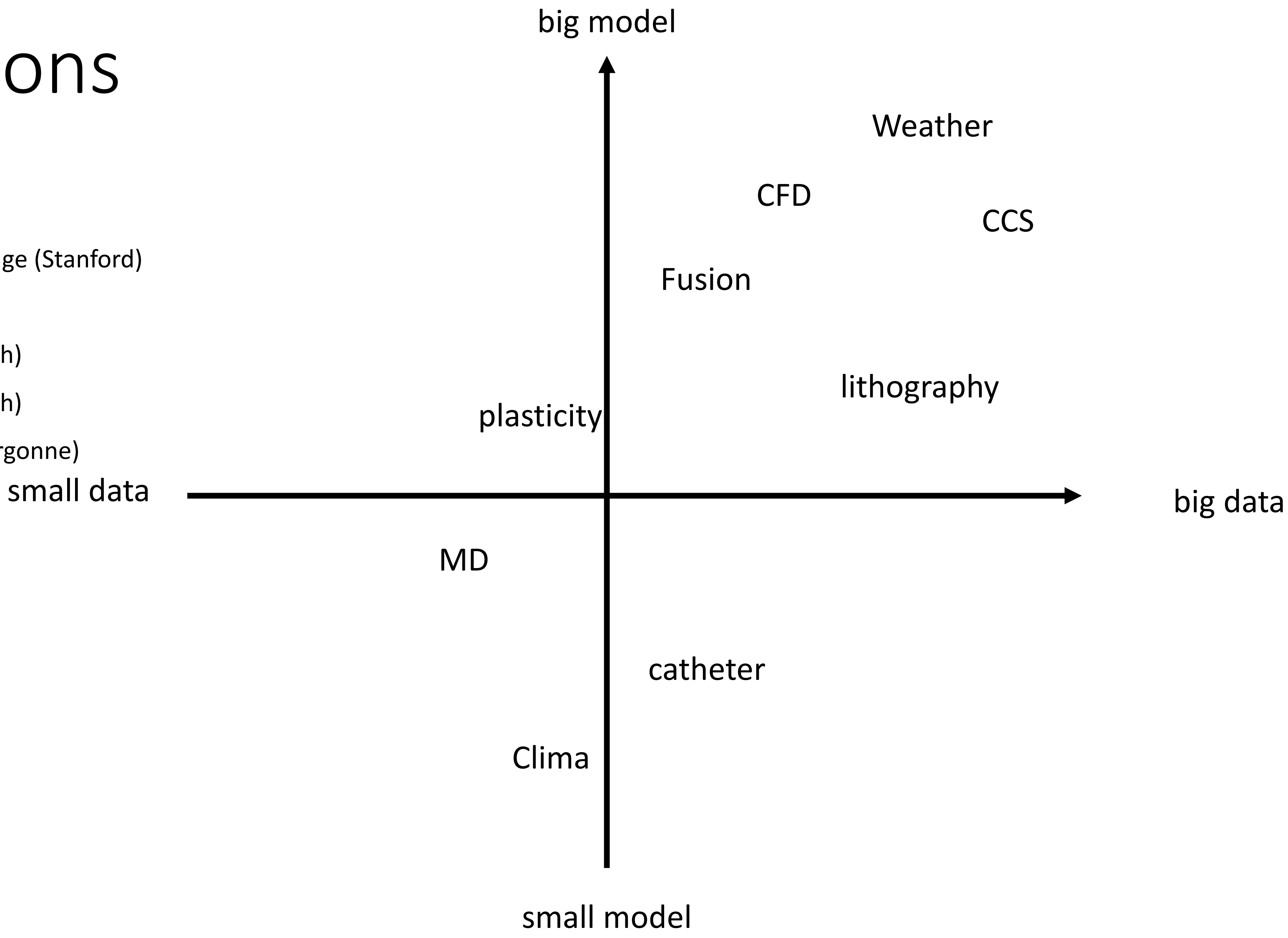
Weather prediction: Unprecedented Resolution for AI, “**100,000x**” faster than traditional methods
[1] PSHRCMKHLAHKA



CO2 storage: Multiphase flow for CO2-water interactions, “**60,000x**” faster (2d), **“700,000x”** faster (3d).
[2] WLLAAB

Applications

- Weather (Nvidia)
- Carbon Capture & Storage (Stanford)
- Lithography (Nvidia)
- Fluid Mechanics (Caltech)
- Solid Mechanics (Caltech)
- Molecular Dynamics (Argonne)
- Fusion (UK atomic)
- Catheter (Caltech)
- Clima Cloud (Caltech)



Operator learning

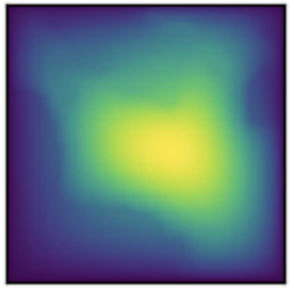
Operators are map between function space.

Given a dataset of input-output pairs, find the map (operator)

F: a->u

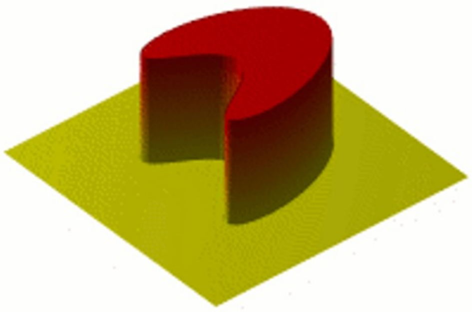
Input: coefficient

→



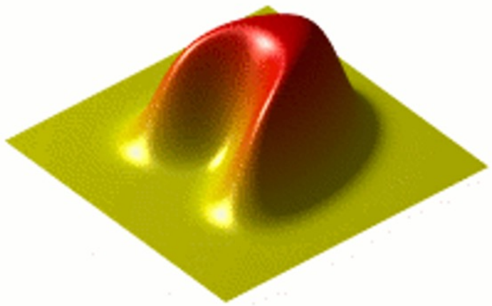
Output: solution

F: u0->u1



Input: initial

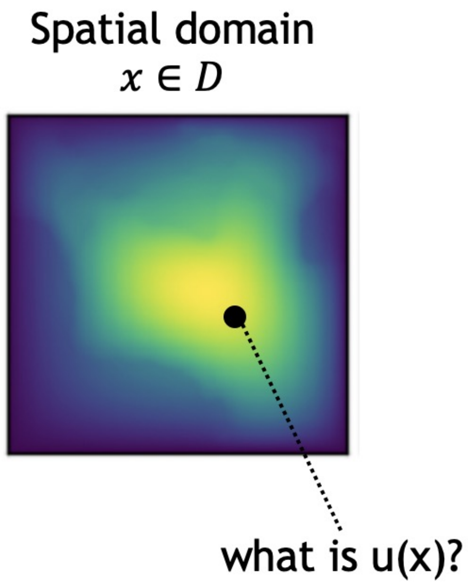
→



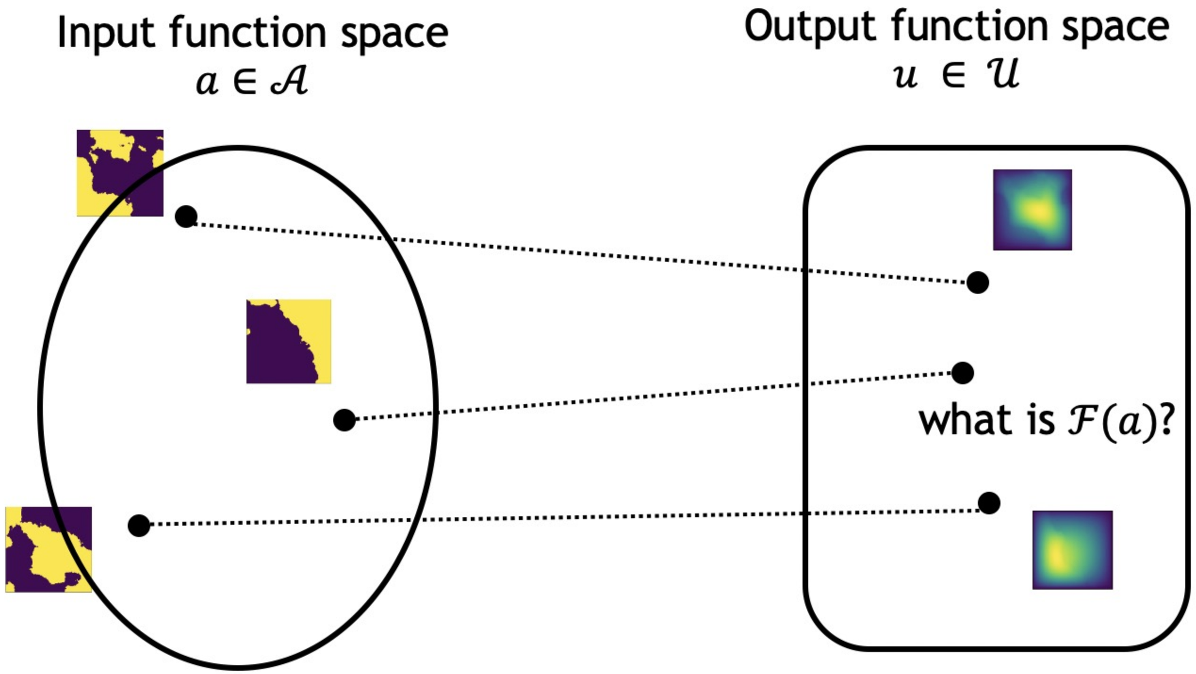
Output: solution

Solve vs learn

Solving for a PDE instance u
approximate $u(x)$ in the spatial space.



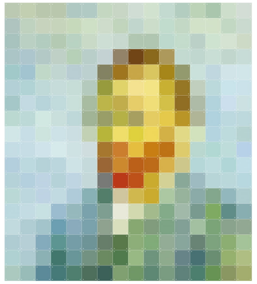
learn the solution operator $\mathcal{F}$
interpolate u in the function space.

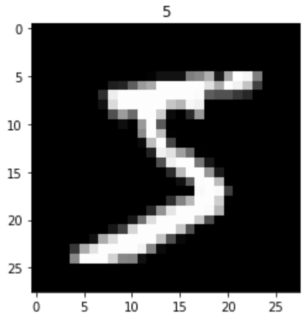


$$\mathcal{F}: \mathcal{A} \rightarrow \mathcal{U}$$

Neural networks vs Neural operators

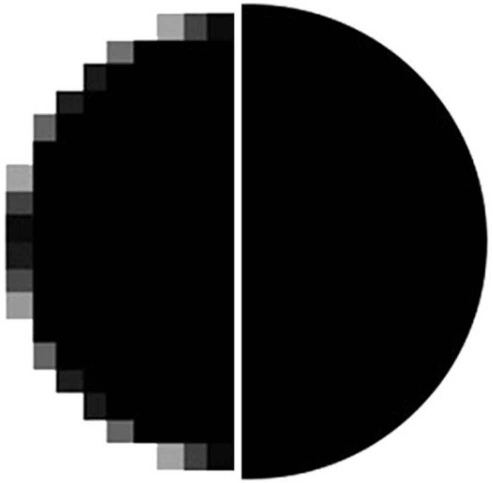
We want to define the model in the function spaces of the PDE.

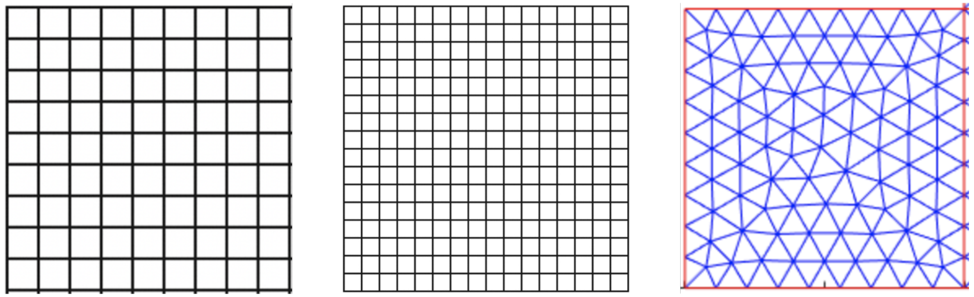


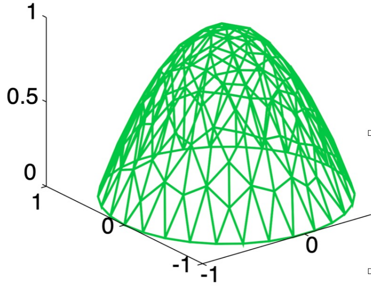


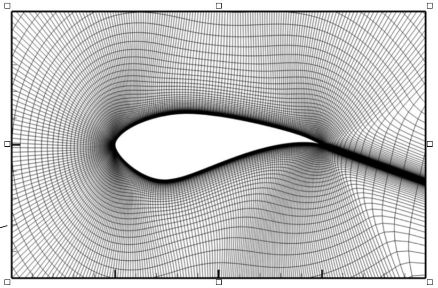
Vocabulary:
Man, woman, boy, girl, prince, princess, queen, king, monarch

	1	2	3	4	5	6	7	8	9
man	1	0	0	0	0	0	0	0	0
woman	0	1	0	0	0	0	0	0	0
boy	0	0	1	0	0	0	0	0	0
girl	0	0	0	1	0	0	0	0	0
prince	0	0	0	0	1	0	0	0	0
princess	0	0	0	0	0	1	0	0	0
queen	0	0	0	0	0	0	1	0	0
king	0	0	0	0	0	0	0	1	0
monarch	0	0	0	0	0	0	0	0	1







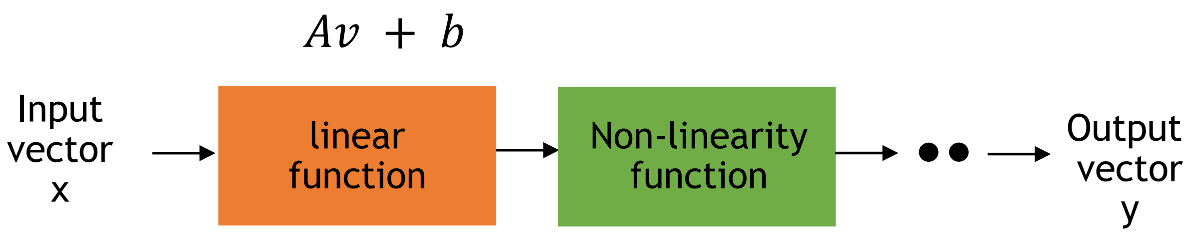


Discretized vector

Continuous function

Neural networks

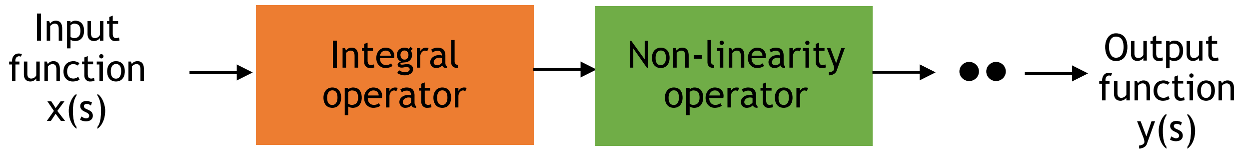
$$f: \ x \rightarrow y$$



Neural operators

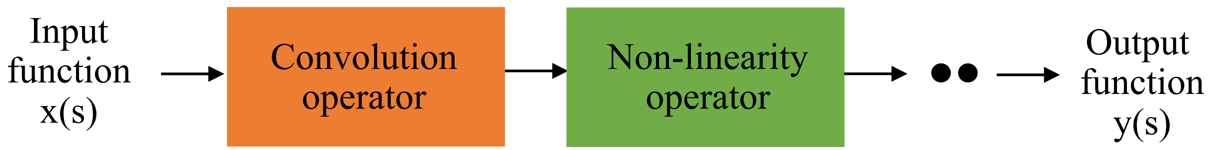
$$F: \; x(s) \rightarrow y(s)$$

$$\int \kappa(x, y)\, v(y)\, dy$$



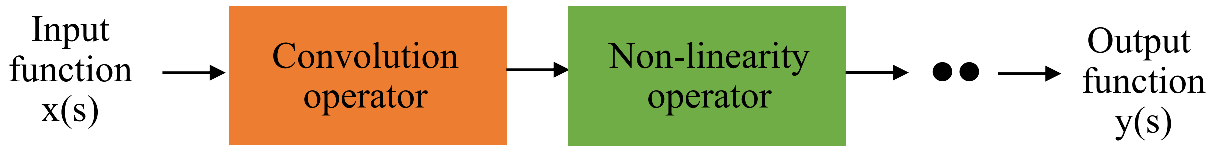
Fourier neural operators

$$F: \; x(s) \rightarrow y(s)$$



Fourier neural operators

F: x(s) → y(s)



Architecture

$$v_{l+1}(s) = \sigma\left(W_l v_l(s) + \int_D \kappa_l(s,z) v_l(z)\, dz + b_l(s) \right)$$

Fourier space

- Assume $\kappa_l(s,z) = \kappa_l(s-z)$ and parametrize its Fourier components θ_l.
  - Convolution theorem: $\int_D \kappa_l(s-z) v_l(z)\, dz = \mathcal{F}^{-1}(\theta_l \cdot \mathcal{F}(v_l))(s)$ $\quad \mathcal{O}(n \log n)$

[4] Fourier neural operator. LKALBSA (2021)

Approximation theory

- Neural operator can approximate any continuous operator.

[5] Nikola Kovachki et. al. 2021

- FNO can mimic pseudospectral solvers to get a bound on the number parameters.

[6] Nikola Kovachki et. al. 2021

- FNO (non-linear decoder) can be more efficient than DeepONet

[7] Samuel Lanthaler et. al. 2023

- FNO (non-linear decoder) can approximate any continuous operator with 1 Fourier mode (but higher channel dimensions).

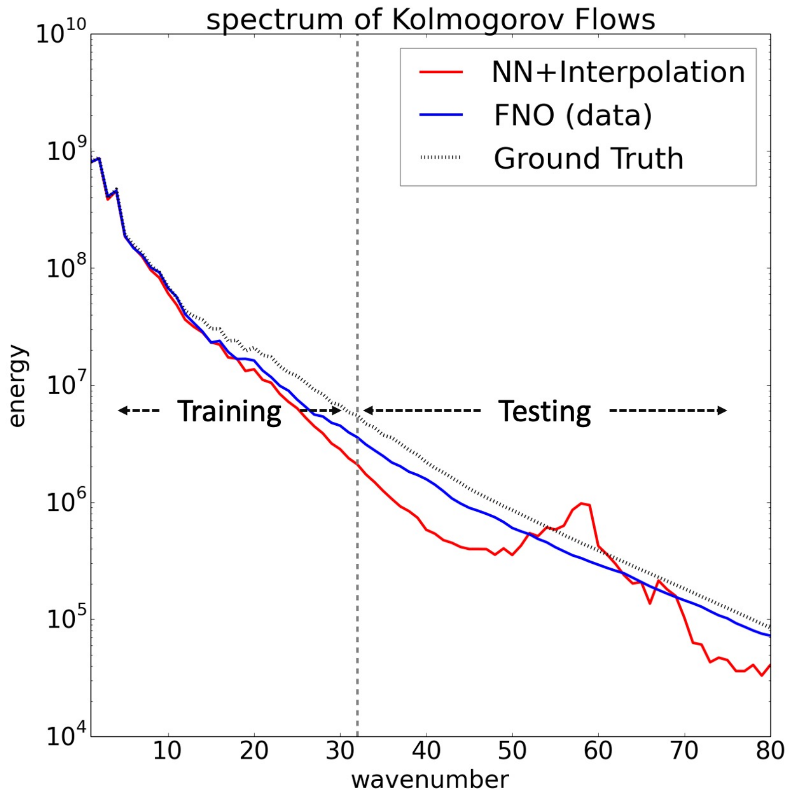
[8] Samuel Lanthaler et. al. 2023

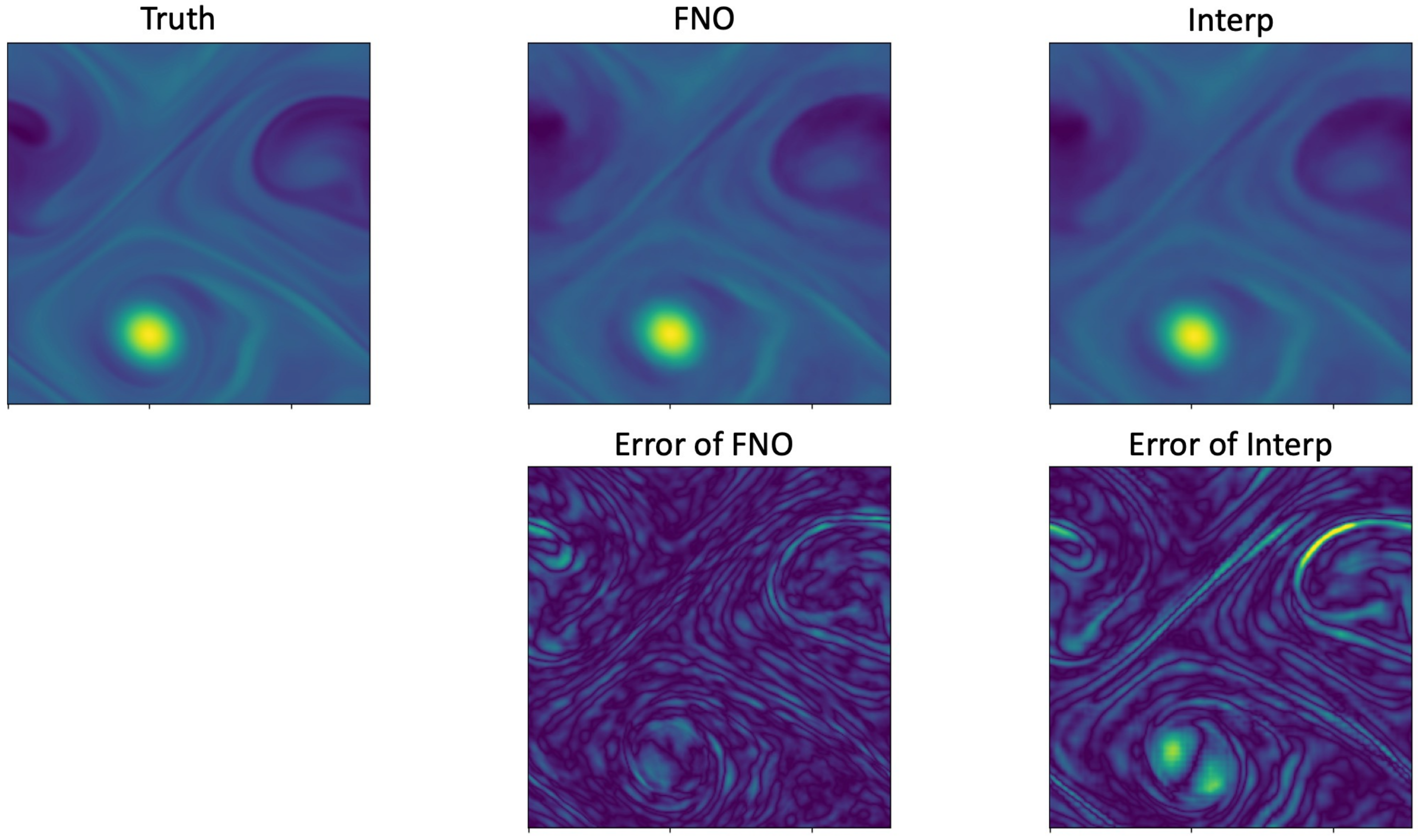
Discretization convergent (wrt refinement)

Property \ Model	CNNs	DeepONets	CNNs+Interpolation	Neural Operators
Discretization Invariance	✗	✗	✓	✓
Query at any point	✗	✓	✓	✓
Input at any point	✗	✗	✓	✓
Universal Approximation	✗	✓	✗	✓

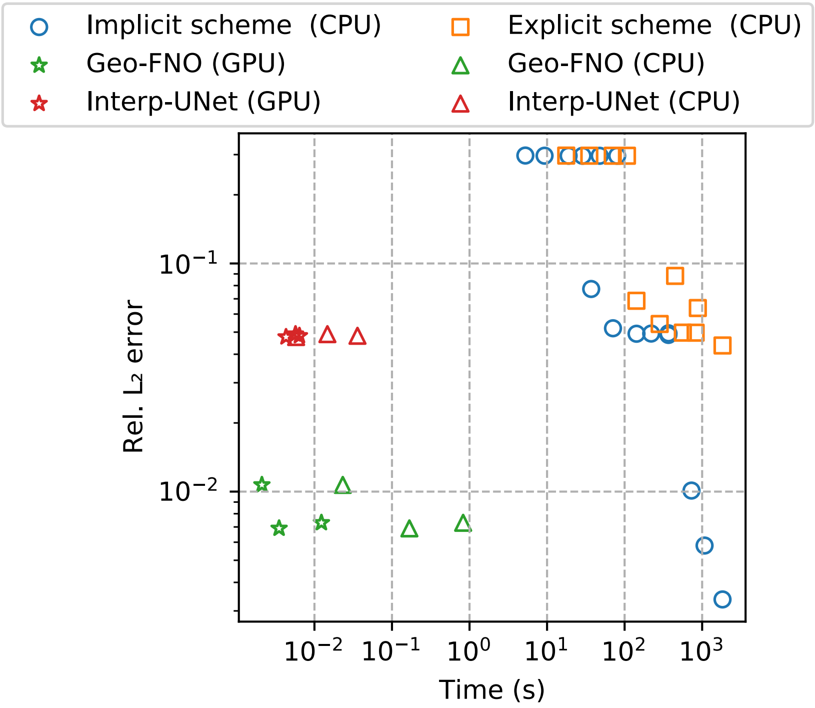
Discretization-invariance is a design philosophy. We define the problem in infinite space and then discretize the model.
In practice, many recently-developed NNs are indeed discretization-invariant (e.g. GraphCast, Transformer)

Extrapolation to unseen frequencies

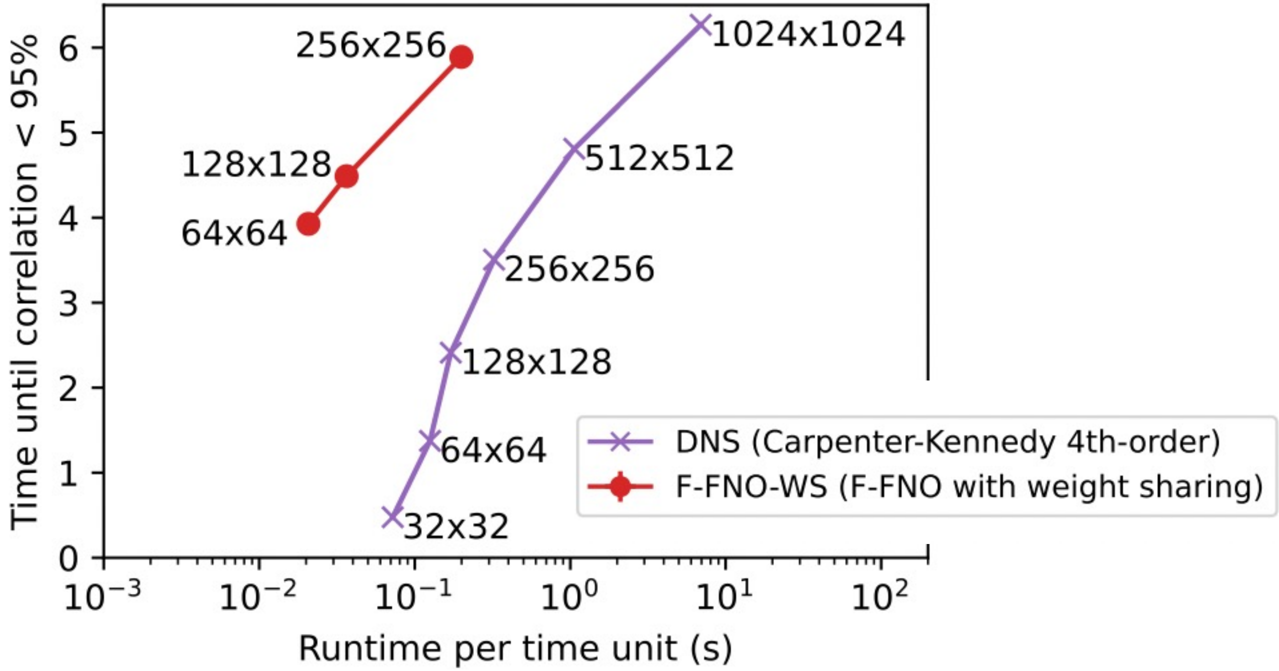




FNO compared to numerical solvers



2d Euler



2d NS

[13] F-FNO. Alasdair Tran et. al.

For larger scale, higher dimension, time-dependent problems, the neural operators seems to converge faster.

PINO: Physics-informed neural operator

Data loss: compare the prediction and ground-truth solution

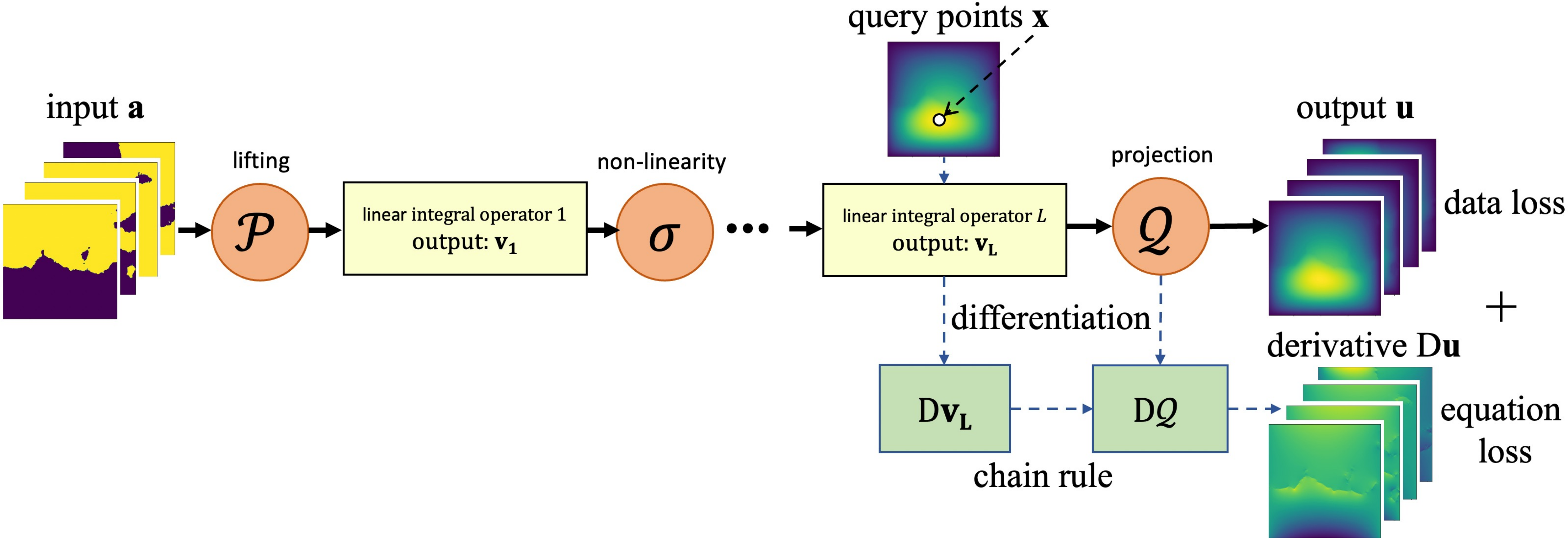
Equation loss: plug the prediction into PDE and compute the residual

$$\begin{aligned} -\nabla \cdot (a(x)\nabla u(x)) &= f(x), \quad && x \in D \\ u(x) &= 0, && x \in \partial D \end{aligned}$$

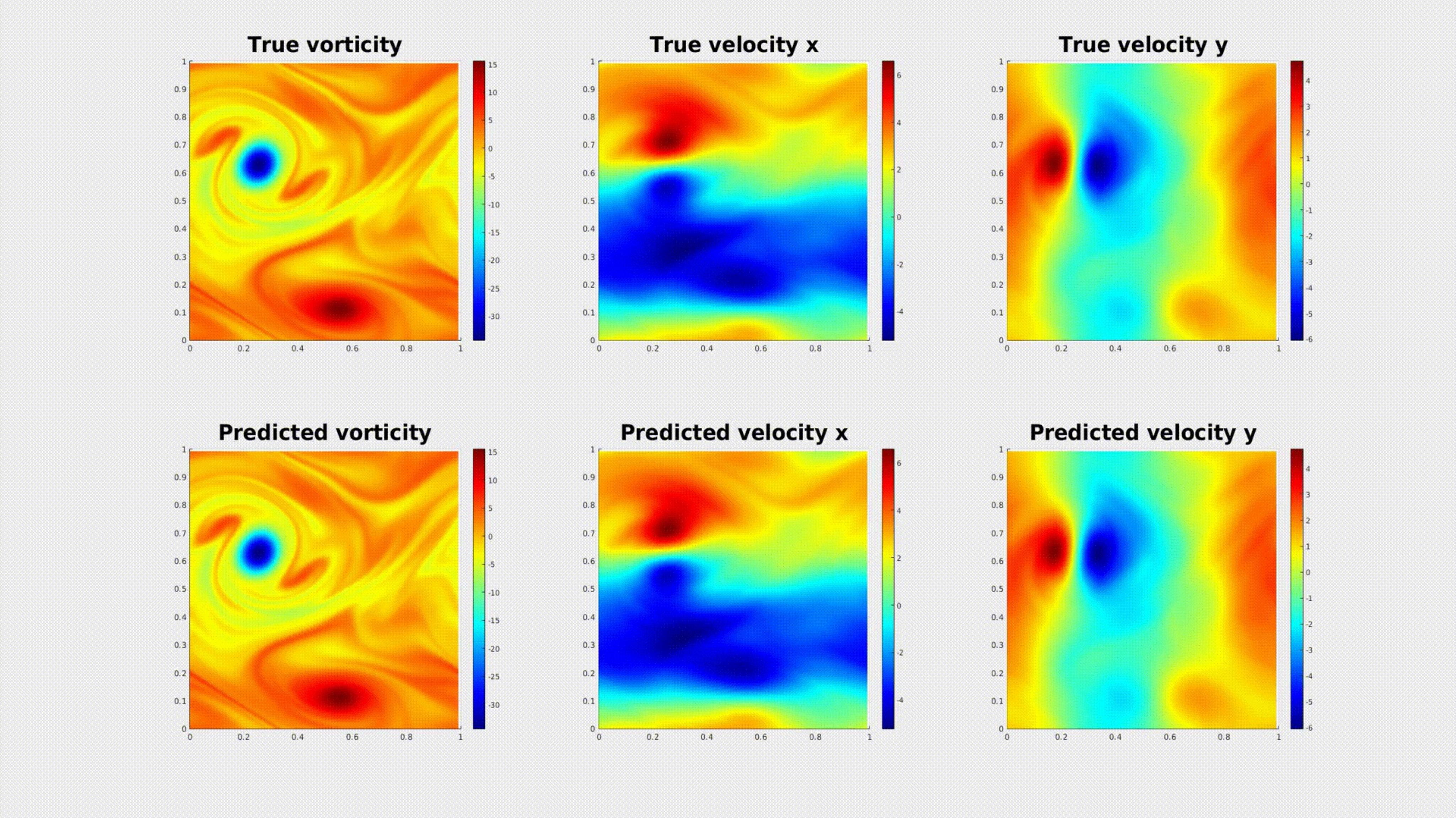
[14] PINO: LZKJCLAA

[15] Fourier-continuation PINO: MLWLBHA

PINO: Physics-informed neural operator

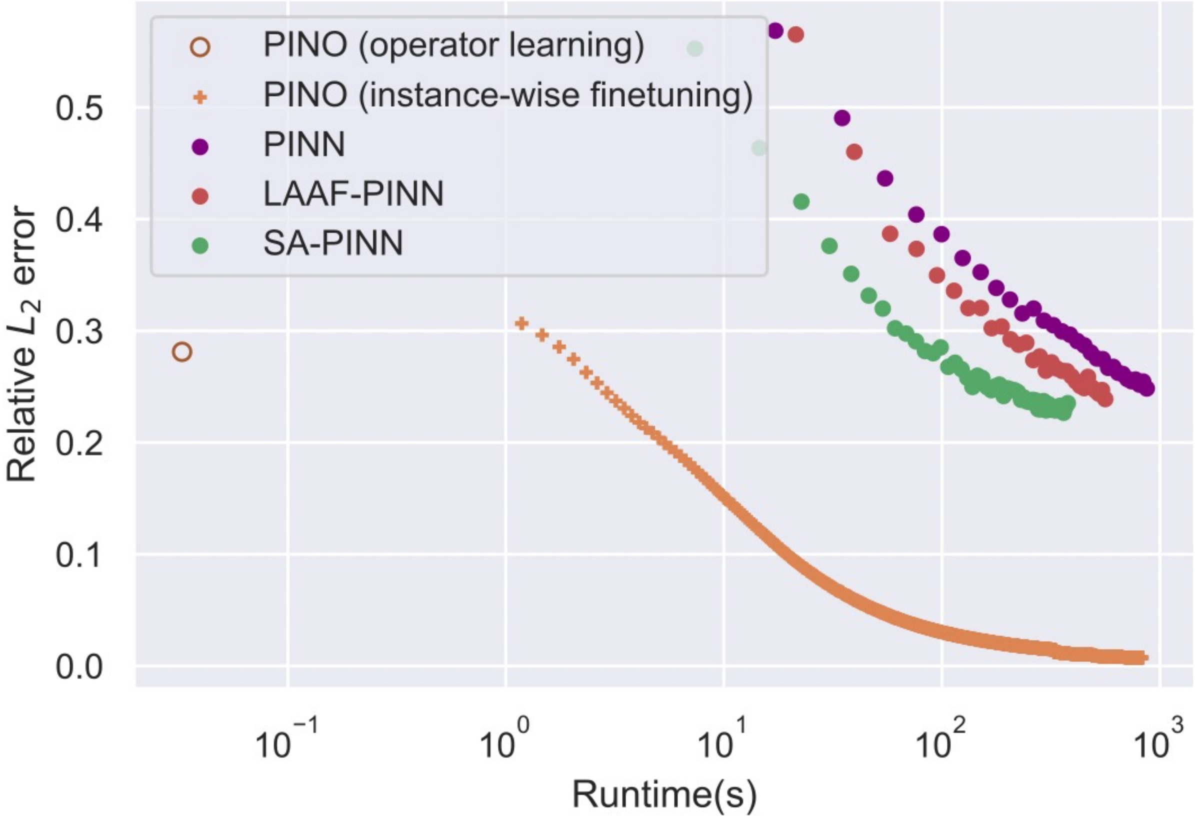


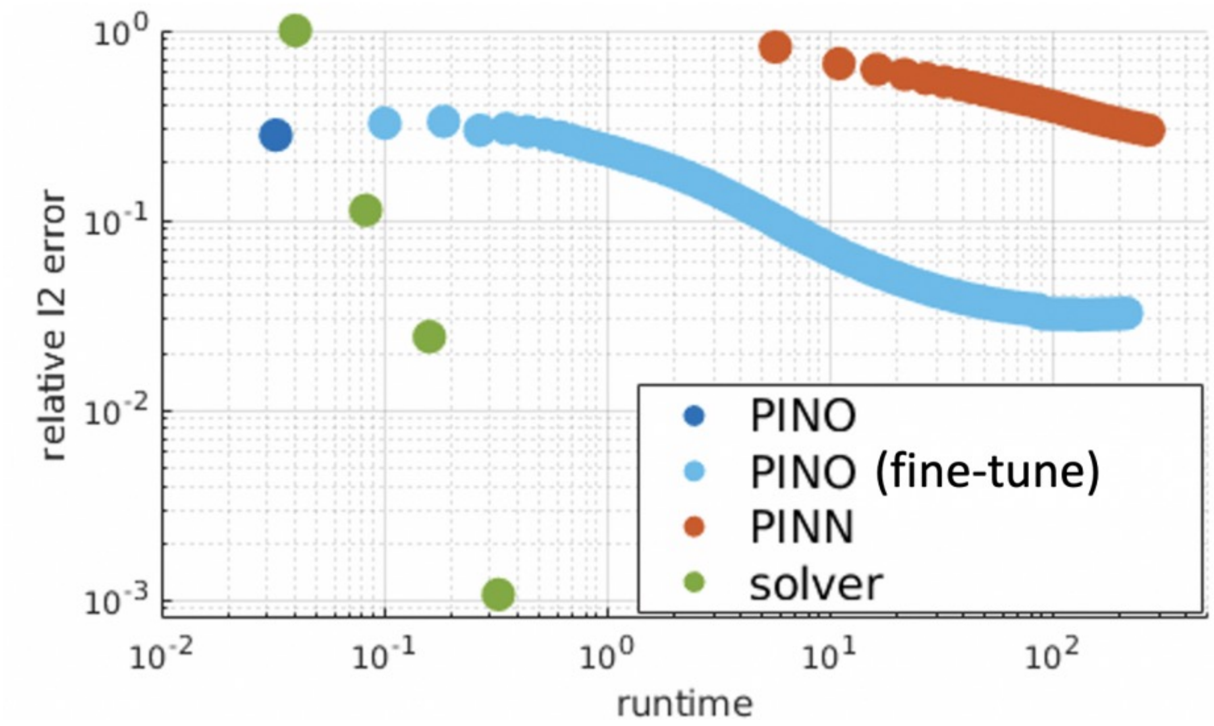
- PINO gets 2% error on Re500 [2π x 2π x 1s]
- Easily generalize from one Re to another



Relative error: PINO: 0.9%, PINN:18.7%

Convergence

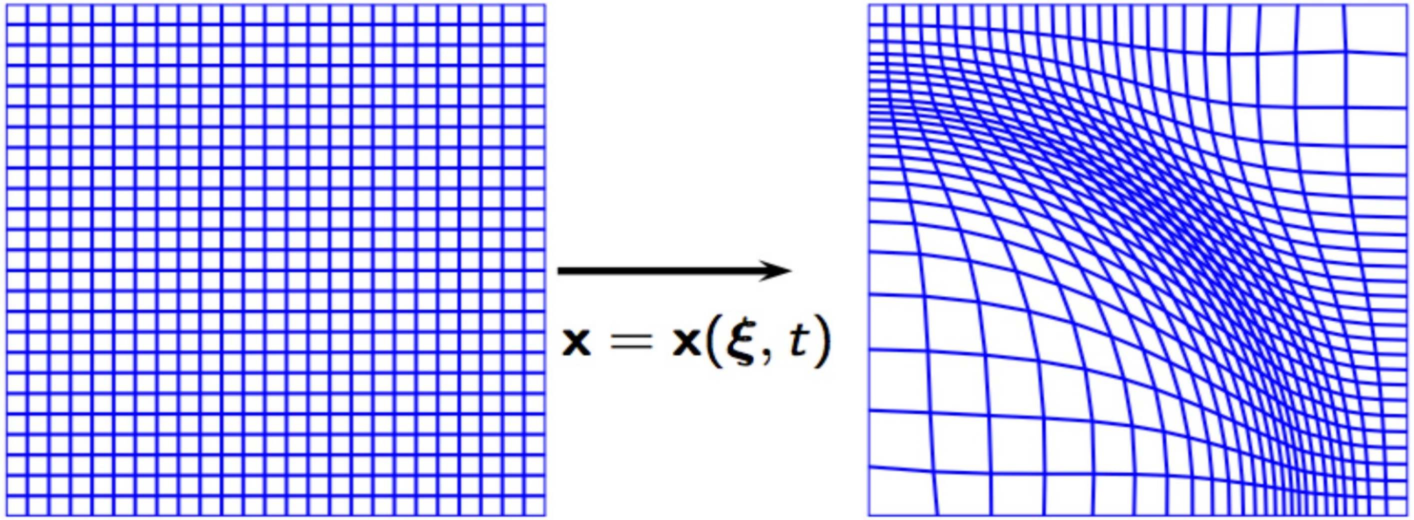




PINO converges faster than PINNs but slower than solvers

Geo-FNO: geometric-aware neural operator

Use deformation to construct adaptive meshes for complex geometries and multiscale structures.

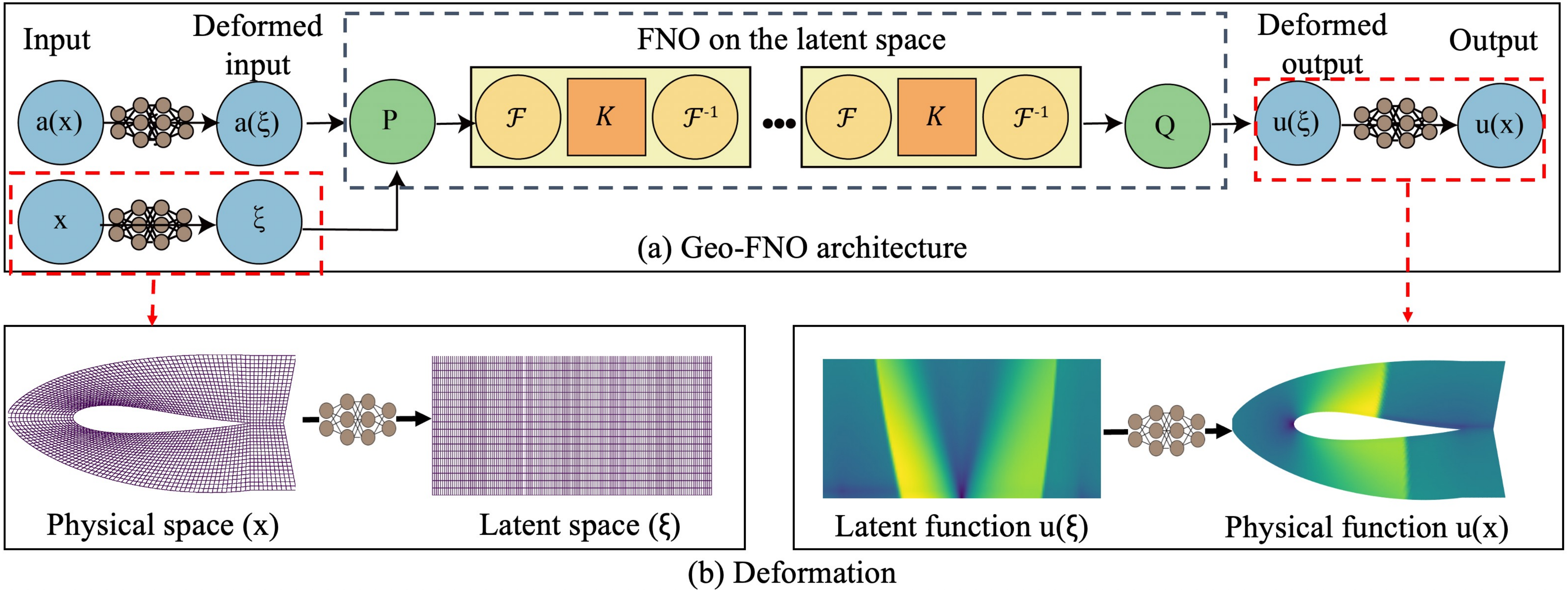


FFT computation mesh

Adaptive mesh

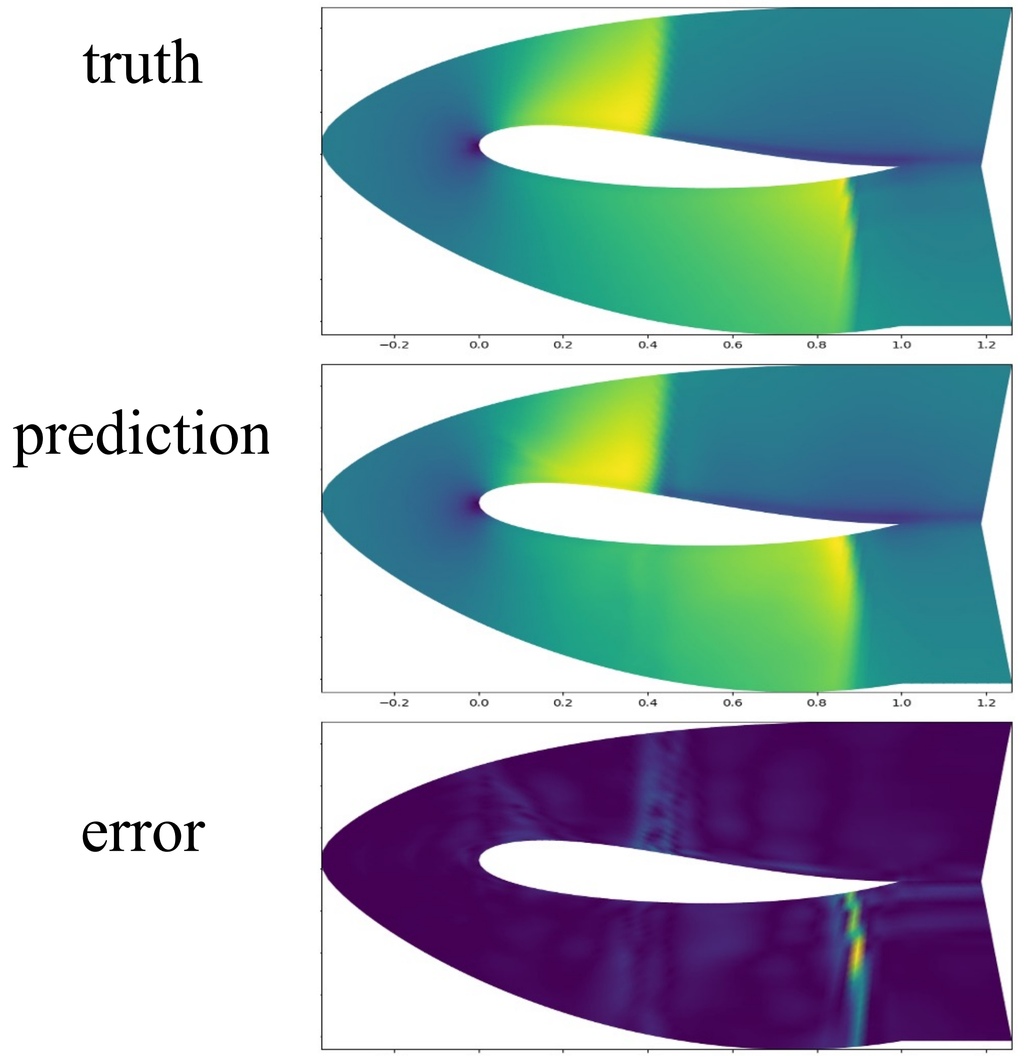
Geo-FNO

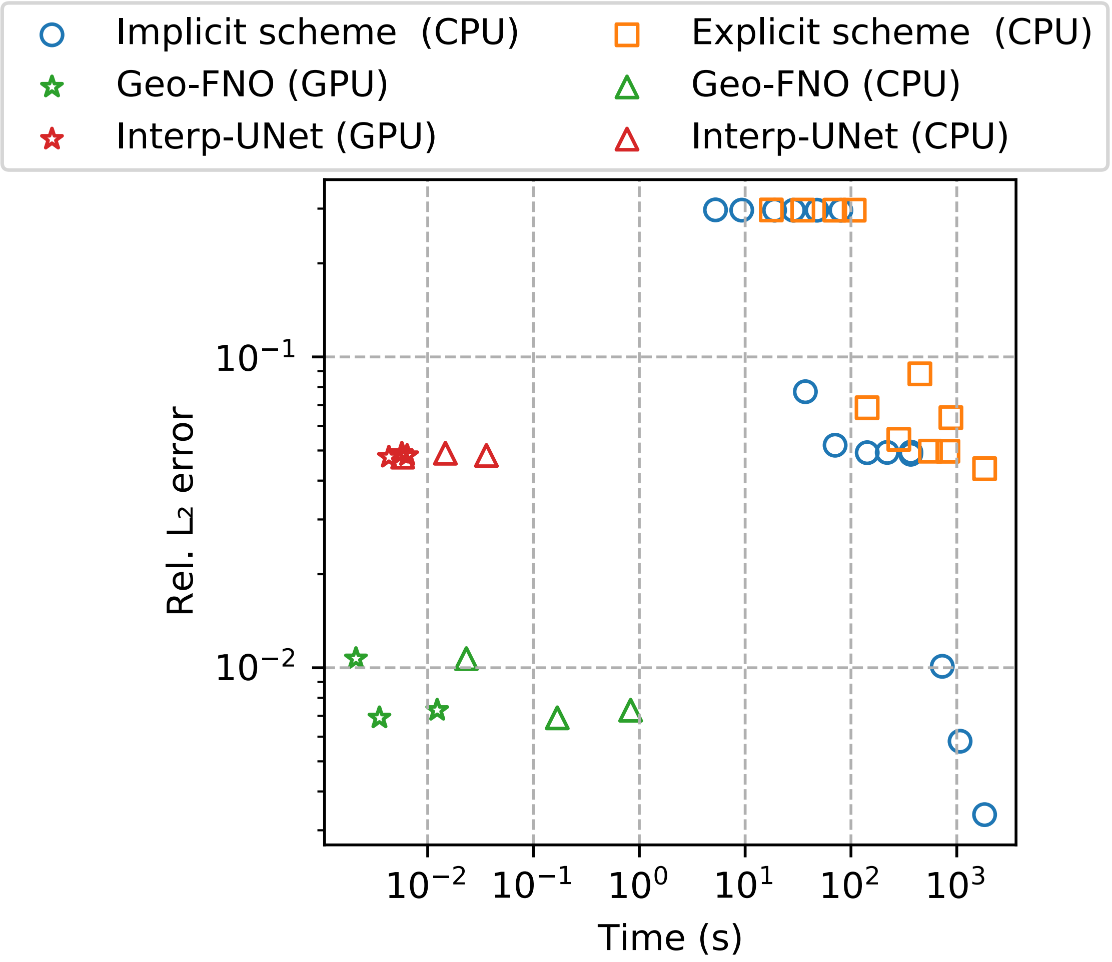
Idea: deform the irregular physical space to a uniform latent space, so the FFT can be applied in the latent space.



[19] Geo-FNO: LZLA

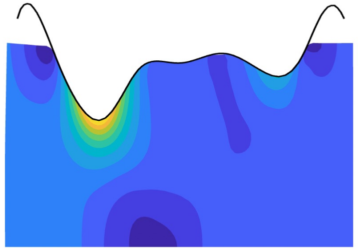
Examples: Airfoils

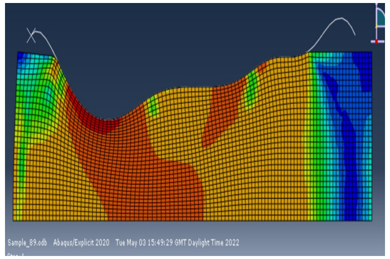




Error = 1.3%
10^5 speedup

Examples: Plastic equation

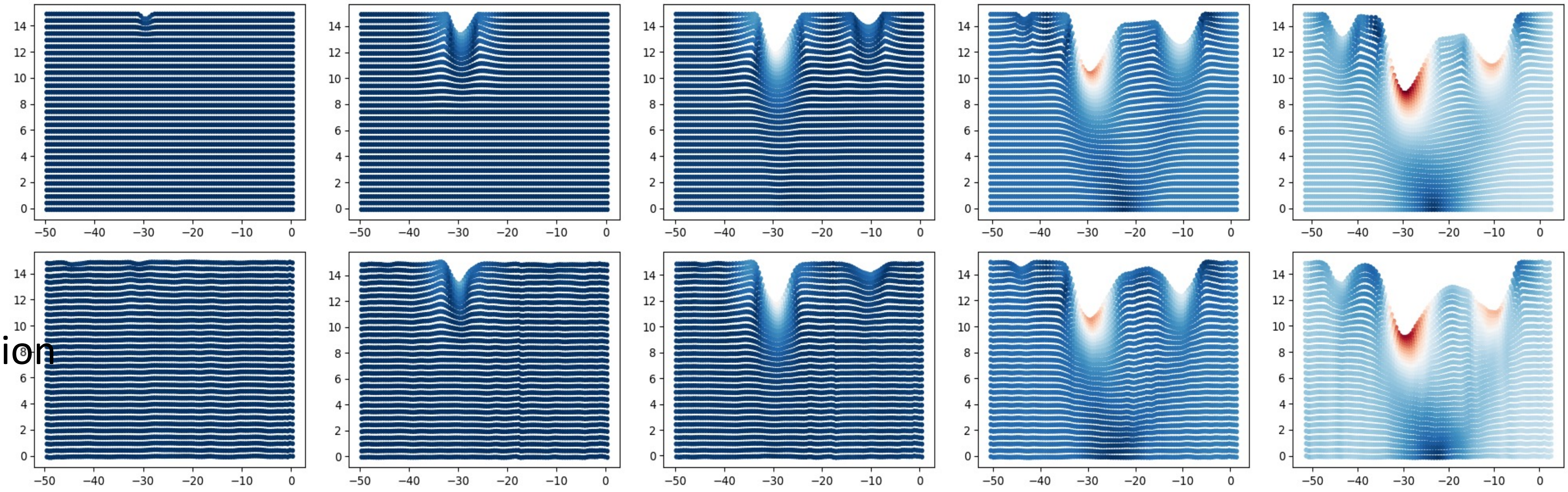




Plastic forging problem: a block of material is impacted by a frictionless, rigid die from top.

Truth

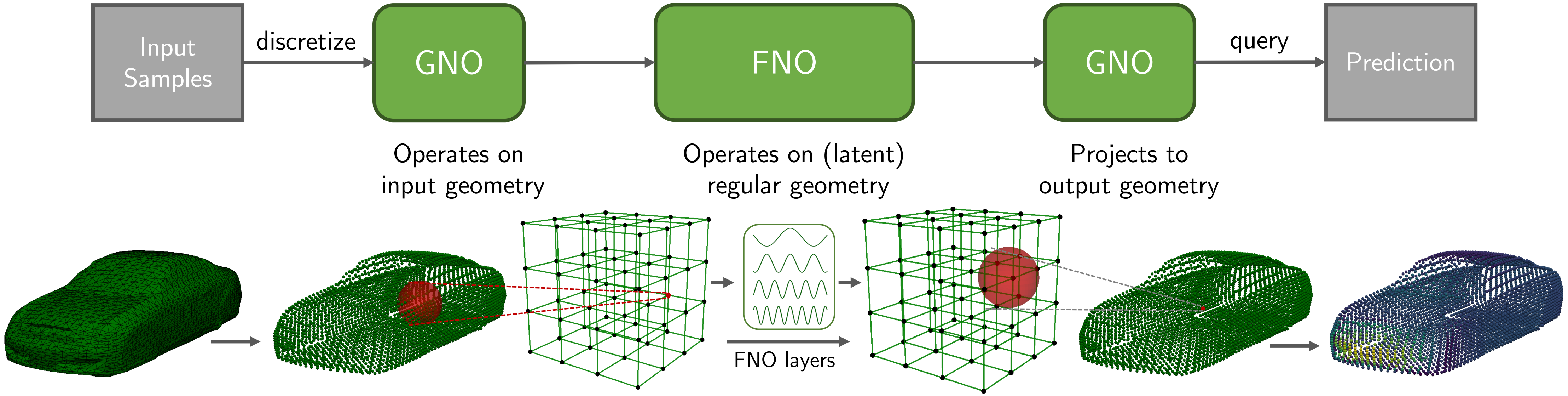
Prediction



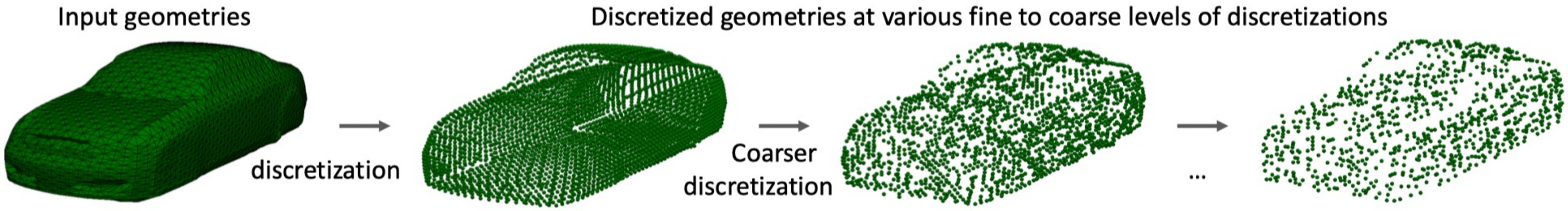
t=4 t=8 t=12 t=16 t=20

Error = 0.7%

Fourier + Graph representation

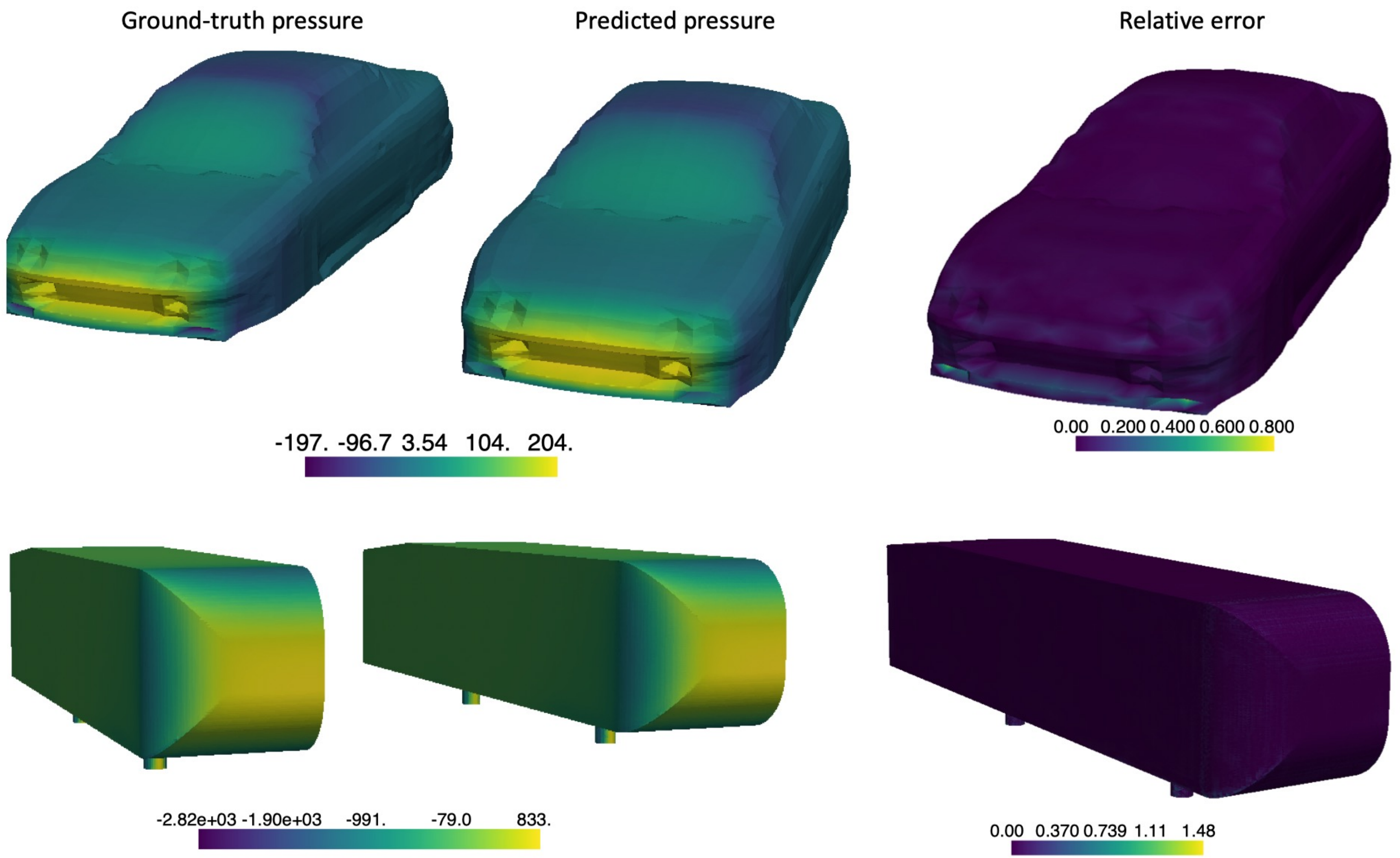


Irregular grid + Discretization consistent

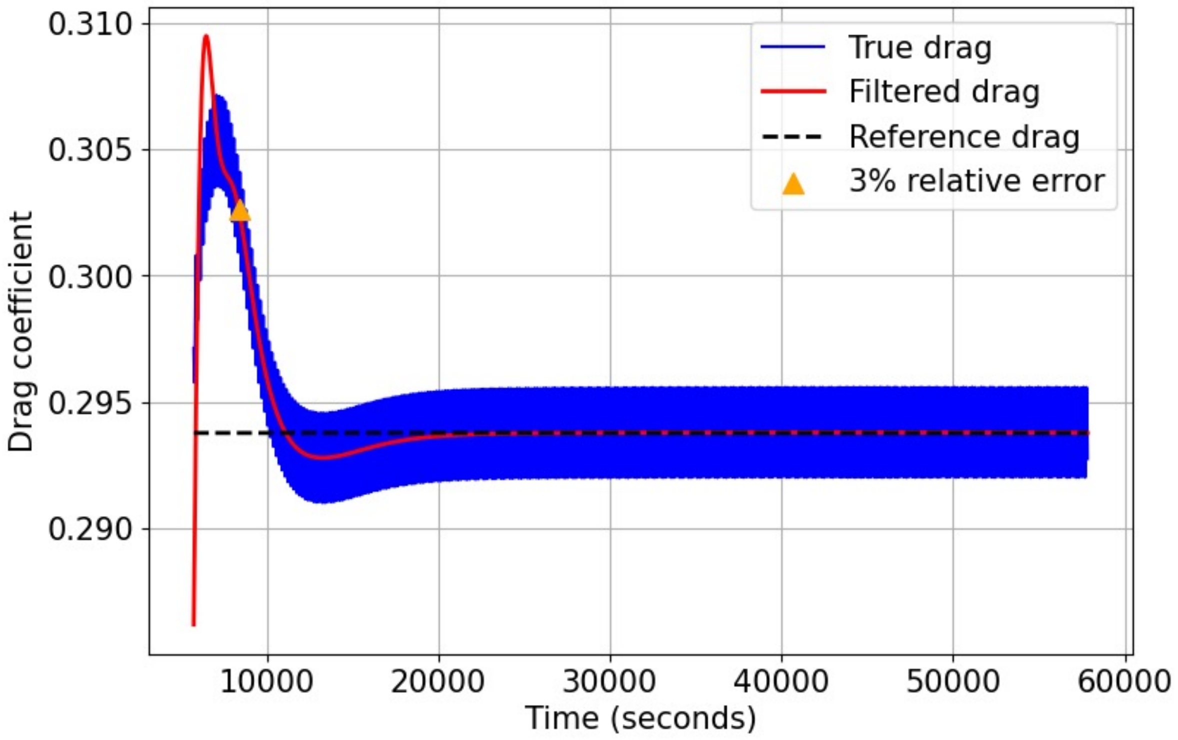


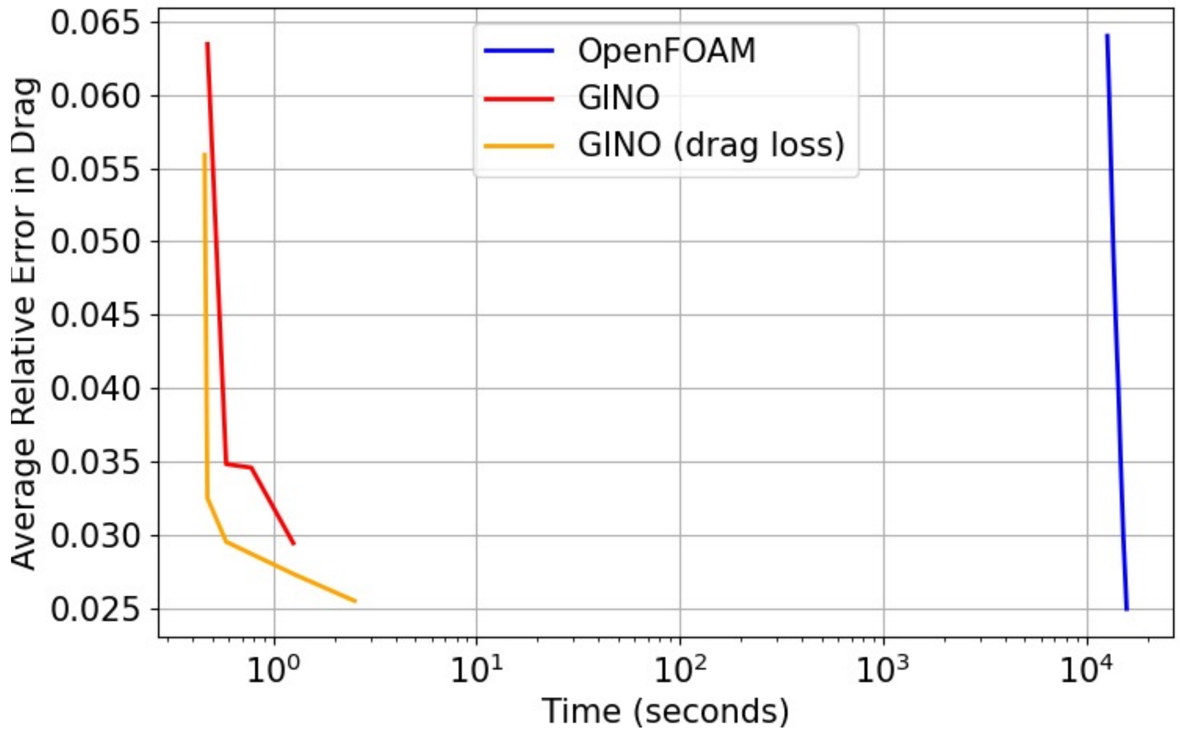
Model	Range	Complexity	Irregular grid	Discretization invariant
GNN	local	$O(N\text{degree})$	✔	✗
CNN	local	$O(N)$	✗	✗
UNet	global	$O(N)$	✗	✗
Transformer	global	$O(N^2)$	✔	✔
GNO (kernel)	radius r	$O(N\text{degree})$	✔	✔
FNO (FFT)	global	$O(N \log N)$	✗	✔
GINO [**Ours**]	global	$O(N \log N + N\text{degree})$	✔	✔

High accuracy



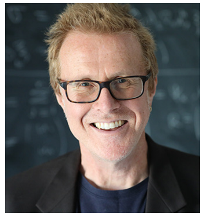
Drag coefficient



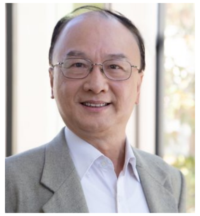


Caltech


Anima Anandkumar


Andrew Stuart


Kaushik Bhattacharya


Thomas Y. Hou

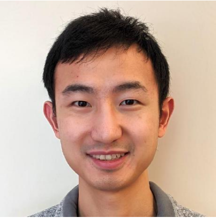

Oscar P. Bruno

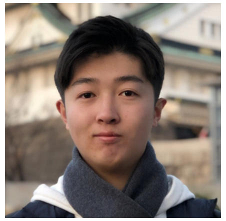

Morteza Gharib


Chiara Daraio


Kamyar Azizzadenesheli


Nikola B. Kovachki


Daniel Zhengyu Huang

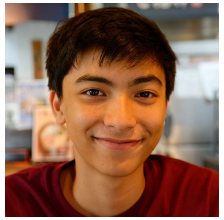

Jiawei Zhao


Hongkai Zheng


Sahin Lale


Yixuan Wang

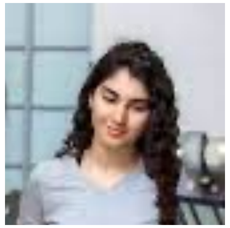

Daniel Leibovici


Miguel Liu-Schiaffini


David Jin


Derek Qin


Haydn Maust


Kimia Hasabi

Vansh Tibrewal

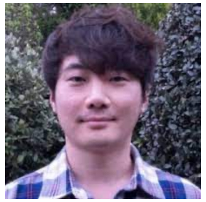
NVIDIA


Kamyar Azizzadenesheli


Nikola B. Kovachki

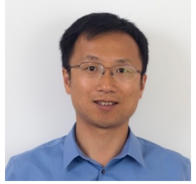

Jean Kossaifi


Boyi Li

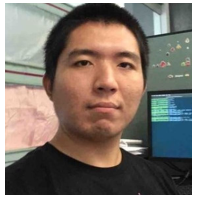

Chris Choy


Jaideep Pathak


Karthik Kashinath


Haoxing Ren


Morteza Mardani


Haoyu Yang


Thorsten Kurth


Mingjie Liu

Argonne NATIONAL LABORATORY


Arvind Ramanathan


Austin Clyde

Berkeley UNIVERSITY OF CALIFORNIA


Sanjeev Raja

UNIVERSITY OF CAMBRIDGE

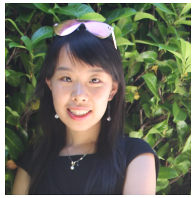

Burigede Liu

Stanford


Sally M. Benson


Gege Wen

UC San Diego


Yuanyuan Shi


Miroslav Krstic

BERKELEY LAB


Shashank Subramanian


Peter Harrington

References

1. Pathak, Jaideep, et al. "Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators." *arXiv preprint arXiv:2202.11214* (2022).
2. Wen, Gege, et al. "Real-time high-resolution CO 2 geological storage prediction using nested Fourier neural operators." *Energy & Environmental Science* (2023).
3. Li, Zongyi, et al. "Neural operator: Graph kernel network for partial differential equations." *arXiv preprint arXiv:2003.03485* (2020).
4. Li, Zongyi, et al. "Fourier neural operator for parametric partial differential equations." *arXiv preprint arXiv:2010.08895* (2020).
5. Kovachki, Nikola, et al. "Neural operator: Learning maps between function spaces." *arXiv preprint arXiv:2108.08481* (2021).
6. Kovachki, Nikola, Samuel Lanthaler, and Siddhartha Mishra. "On universal approximation and error bounds for fourier neural operators." *The Journal of Machine Learning Research* 22.1 (2021): 13237-13312.
7. Lanthaler, Samuel, et al. "Nonlinear reconstruction for operator learning of pdes with discontinuities." *arXiv preprint arXiv:2210.01074* (2022).
8. De Hoop, Maarten, et al. "The cost-accuracy trade-off in operator learning with neural networks." *arXiv preprint arXiv:2203.13181* (2022).
9. Takamoto, Makoto, et al. "PDEBench: An extensive benchmark for scientific machine learning." *Advances in Neural Information Processing Systems* 35 (2022): 1596-1611.
10. Gupta, Jayesh K., and Johannes Brandstetter. "Towards Multi-spatiotemporal-scale Generalized PDE Modeling." *arXiv preprint arXiv:2209.15616* (2022).
11. Bi, Kaifeng, et al. "Pangu-Weather: A 3D High-Resolution Model for Fast and Accurate Global Weather Forecast." *arXiv preprint arXiv:2211.02556* (2022).
12. Lam, Remi, et al. "GraphCast: Learning skillful medium-range global weather forecasting." *arXiv preprint arXiv:2212.12794* (2022).
13. Tran, Alasdair, et al. "Factorized fourier neural operators." *arXiv preprint arXiv:2111.13802* (2021).
14. Li, Zongyi, et al. "Physics-informed neural operator for learning partial differential equations." *arXiv preprint arXiv:2111.03794* (2021).
15. Maust, Haydn, et al. "Fourier Continuation for Exact Derivative Computation in Physics-Informed Neural Operators." *arXiv preprint arXiv:2211.15960* (2022).
16. Liu, Burigede, et al. "A learning-based multiscale method and its application to inelastic impact problems." *Journal of the Mechanics and Physics of Solids* 158 (2022): 104668.
17. Williams, Francis, et al. "Neural fields as learnable kernels for 3d reconstruction." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.
18. Liu, Mingjie, et al. "An Adversarial Active Sampling-based Data Augmentation Framework for Manufacturable Chip Design." *arXiv preprint arXiv:2210.15765* (2022).
19. Li, Zongyi, et al. "Fourier neural operator with learned deformations for pdes on general geometries." *arXiv preprint arXiv:2207.05209* (2022).
20. Li, Zongyi, et al. "Multipole graph neural operator for parametric partial differential equations." *Advances in Neural Information Processing Systems* 33 (2020): 6755-6766.
21. Bruno, Oscar P., Jan S. Hesthaven, and Daniel V. Leibovici. "FC-based shock-dynamics solver with neural-network localized artificial-viscosity assignment." *Journal of Computational Physics: X* 15 (2022): 100110.