Introducing!

Dhyuti Walkthroughs

To do a deep dive or build intuition in SciML algorithms and paradigms

WalkThrough 01

PINNs, inverse problems, and 1D analogs of real life examples

29th September , 2025

Dhyuti
WalkThrough
Session 1

Physics Informed Neural Networks

Inverse problems and 1D analogs of real-life examples

Rahul
Sundar

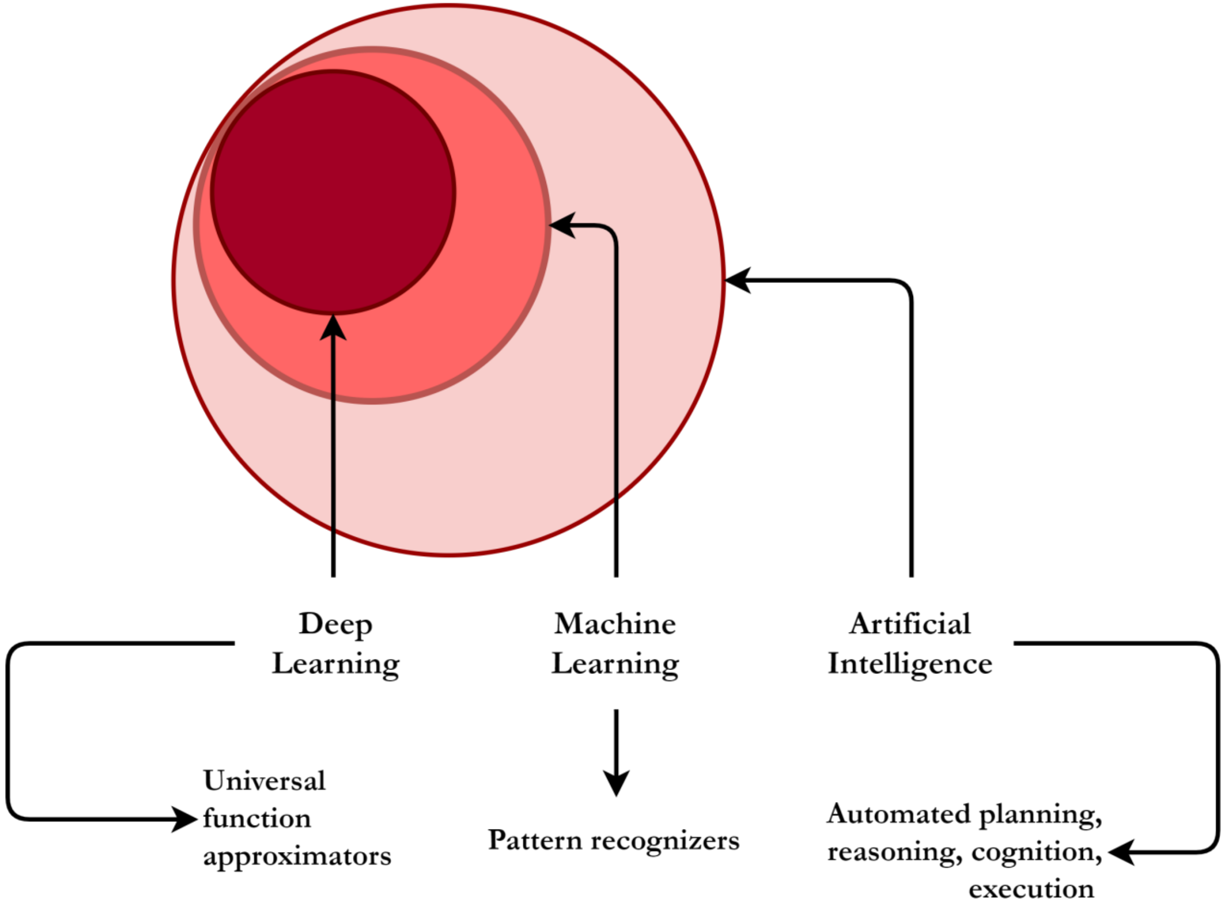
PhD Scholar / Scientist (AI/ML)
Dept. of Aerospace Engineering, IIT Madras / Verisk, India

Relevant Resources

Google Colab Notebook used in this session [*Save a copy and experiment at your own pace.* ***]***
https://colab.research.google.com/drive/1UN2B8HNfpOmohF6wM2_myq55evupZfKd#scrollTo=Izbu-AVcsyt6

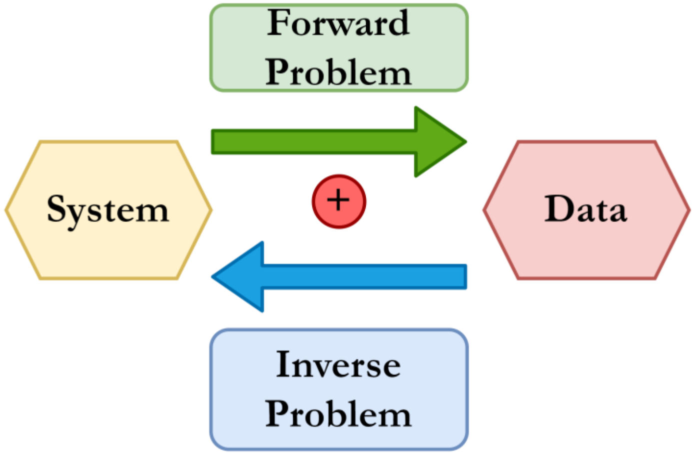
GitHub Repository*:[Look out for byte sized open source challenges in the Github issues]*
https://github.com/DhyutiLABS/DhyutiWalkthroughs

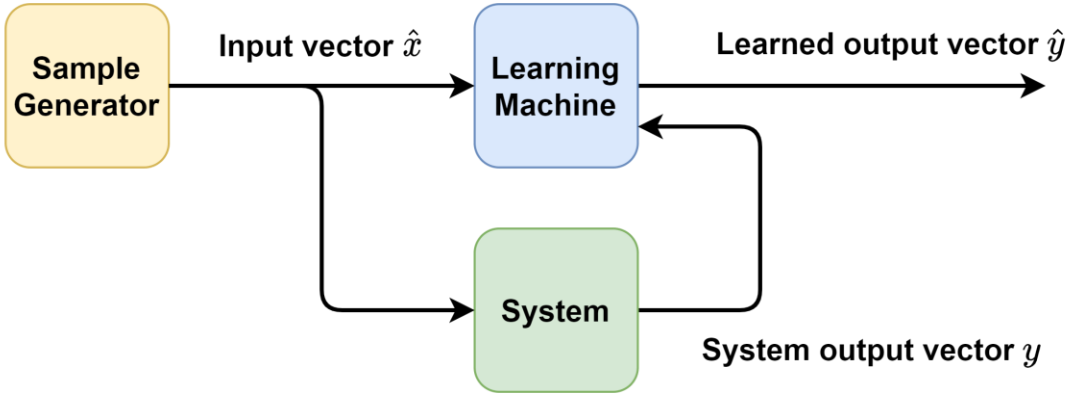
What is AI?


Deep
Learning
Machine
Learning
Artificial
Intelligence
Universal
function
approximators
Pattern recognizers
Automated planning,
reasoning, cognition,
execution

(Scientific/Science for/Science and) Machine Learning

- **SciML ↔ Observation (Simulations/Experiments) guided modelling**
  - Not a new paradigm: Existing since 1900's! [Turbulence modelling for example!]
  - **Difference then vs now:**
    - **Scale** of data, memory and compute
    - **Advancements in algorithms, hardware and software**



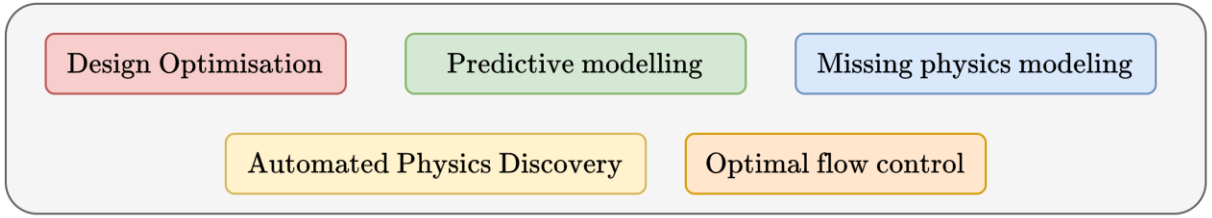


Some Review papers:

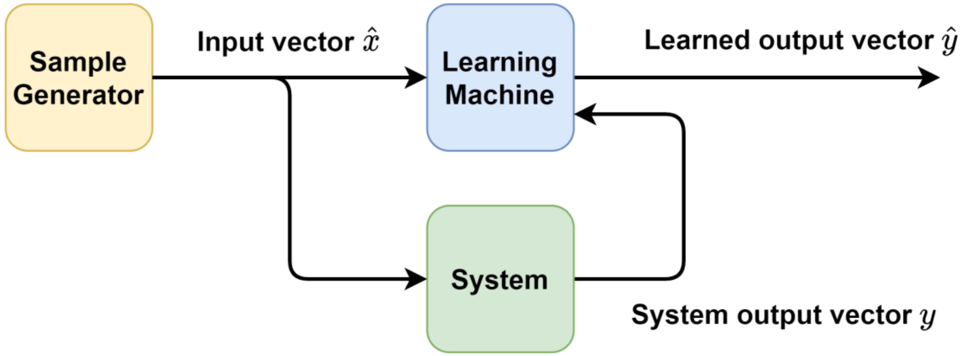
Karniadakis et al. (2021), Nature
Cuomo et al. (2022), JSC;
Kim et al. (2024), MSE

PINN relevance

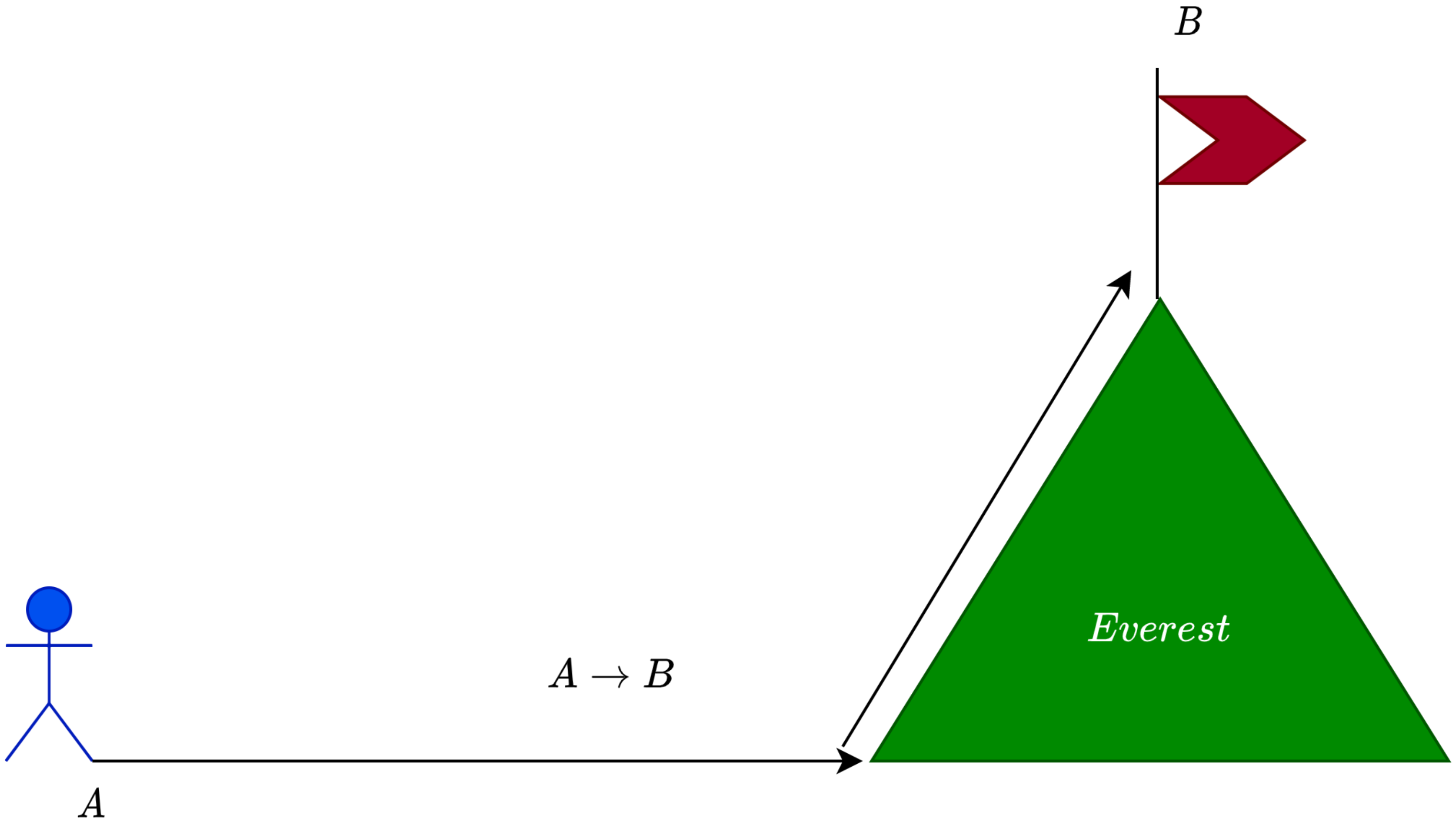
- Typical **inverse problems**:



- Typical **PINN** applications:
  - **Turbomachinery/ Energy** – Optimal control/ Digital twins (Hu et al. (2023), CJA)
  - **BioMedical Engineering** – Constitutive modelling/ Imaging (Dwivedi et al. (2024), Nature)
  - **Material science** – Automated discovery / Constitutive modelling (Cao et al. (2018), ACS Nano)
  - **Climate and weather** – Forecasting, (Bodnar et al., (2024), ArXiv)



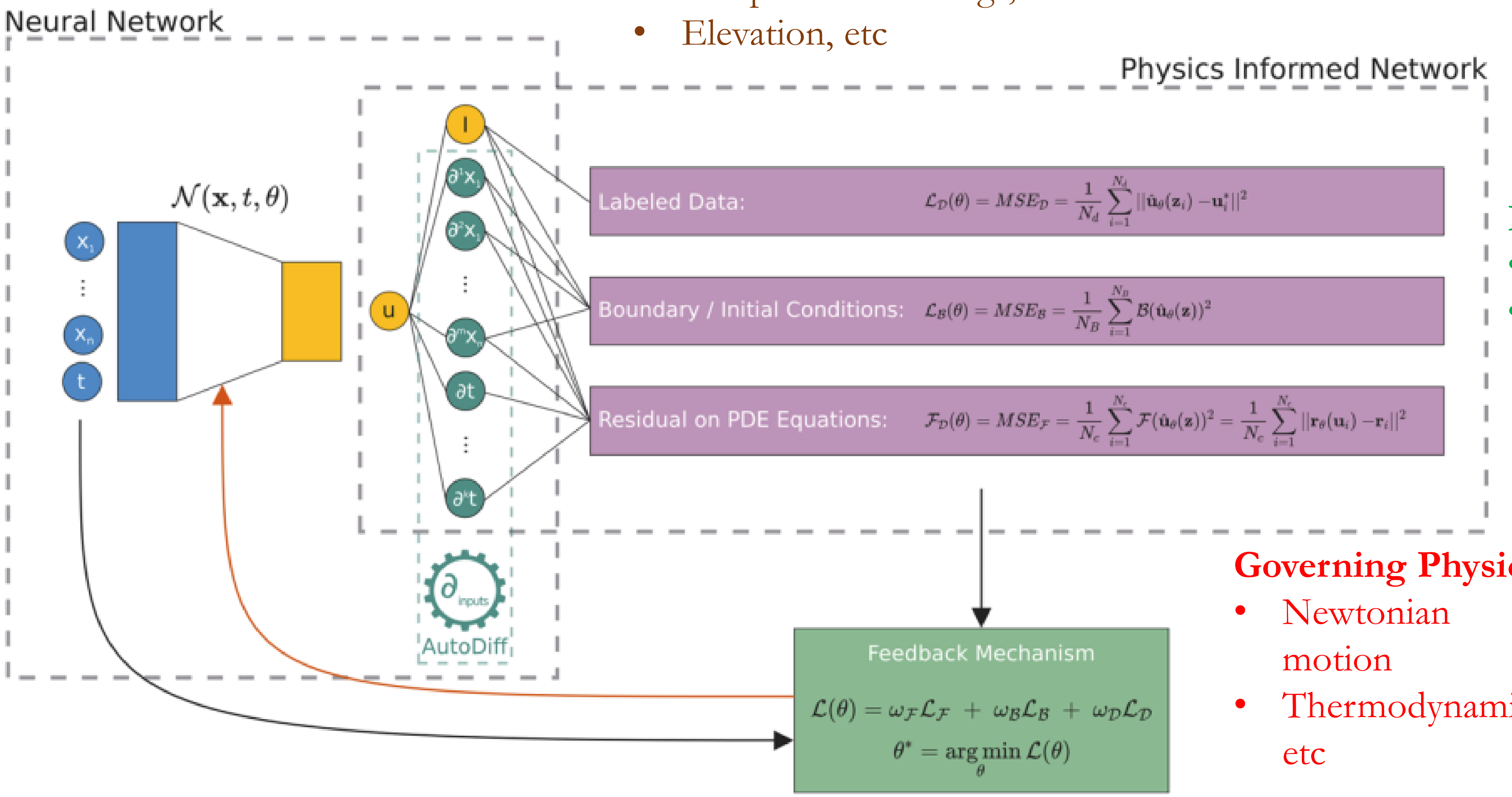
What are PINNs?



What are PINNs?

Labeled data
- Smart watch – Pace readings
- Temperature readings,
- Elevation, etc

Inputs
Your position **x** at any time **t**

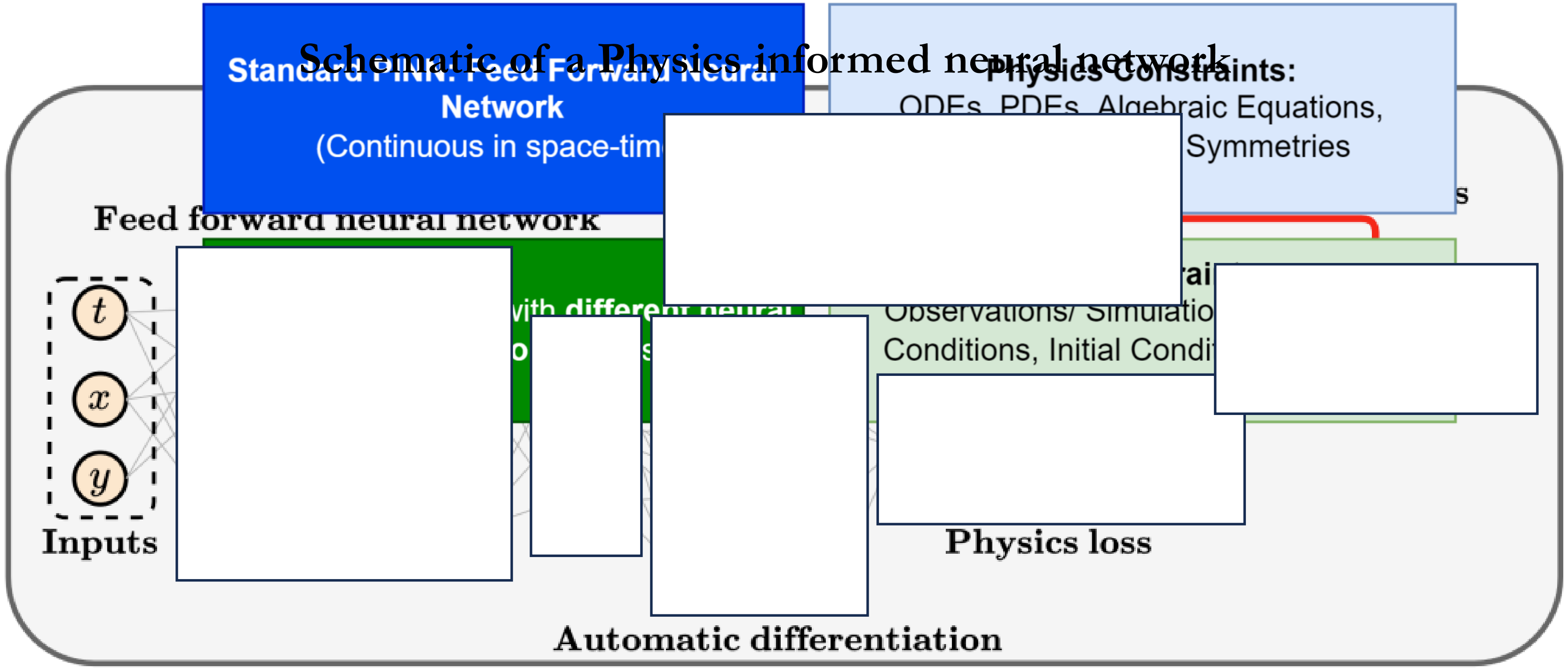


Initial Condition
- Starting point
- Elevation, etc

Governing Physics
- Newtonian motion
- Thermodynamics, etc

What are PINNs?

Physics Informed Neural Networks = A class of neural networks with Physics constraints = bridge between AI & physical reality. (Raissi et al. (2019))



Effective spatio-temporal function approximators which can work with limited / sparse data with the help of physics constraints!

Source: Sundar et al. (2024), JFS

Implementational Ease of PINNs

- PINNs are extremely easy to implement! *[Can use Cursor to vibe code!]*
- **Define:**
  - **Task**
  - **Input-coordinate system**
  - **Output variables**
  - Labelled data?
  - **Underlying governing equations:**
    - Partial Differential Equations
    - Ordinary Differential Equations
    - Nonlinear Algebraic Equations
    - Linear Equations (*Don't Even think of using PINNs for this!*

```
def u(t, x):
    u = neural_net(tf.concat([t,x],1), weights, biases)
    return u
```

Correspondingly, the *physics informed neural network* $f(t, x)$ takes the form

```
def f(t, x):
    u = u(t, x)
    u_t = tf.gradients(u, t)[0]
    u_x = tf.gradients(u, x)[0]
    u_xx = tf.gradients(u_x, x)[0]
    f = u_t + u*u_x - (0.01/tf.pi)*u_xx
    return f
```

Setup:

1. Dataset
2. Model architecture and its parameters
3. Loss function and validation metrics
4. Optimiser
5. Evaluation/ Monitoring pipeline

Rich Landscape of PINNs

	Method	Description	Representatives
Neural Solver	Loss Reweighting	Grad Norm	GradientPathologiesPINNs [54]
		NTK Reweighting	PINNsNTK [55]
		Variance Reweighting	Inverse-Dirichlet PINNs [56]
	Novel Optimization Targets	Numerical Differentiation	DGM [57], CAN-PINN [58], cvPINNs [59]
		Variantional Formulation	vPINN [60], hp-PINN [61], VarNet [62], WAN [63]
		Regularization	gPINNs [64], Sobolev Training [65]
	Novel Architectures	Adaptive Activation	LAAF-PINNs [66], [67], SReLU [68]
		Feature Preprocessing	Fourier Embedding [69], Prior Dictionary Embedding [70]
		Boundary Encoding	TFC-based [71], CENN [72], PFNN [73], HCNet [74]
		Sequential Architecture	PhyCRNet [75], PhyLSTM [76] AR-DenseED [77], HNN [78], HGN [79]
		Convolutional Architecture	PhyGeoNet [80], PhyCRNet [75], PPNN [81]
		Domain Decomposition	XPINNs [82], cPINNs [83], FBPINNs [84], Shukla et al. [85]
	Other Learning Paradigms	Transfer Learning	Desai et al. [86], MF-PIDNN [87]
		Meta-Learning	Psaros et al. [88], NRPINNs [89]

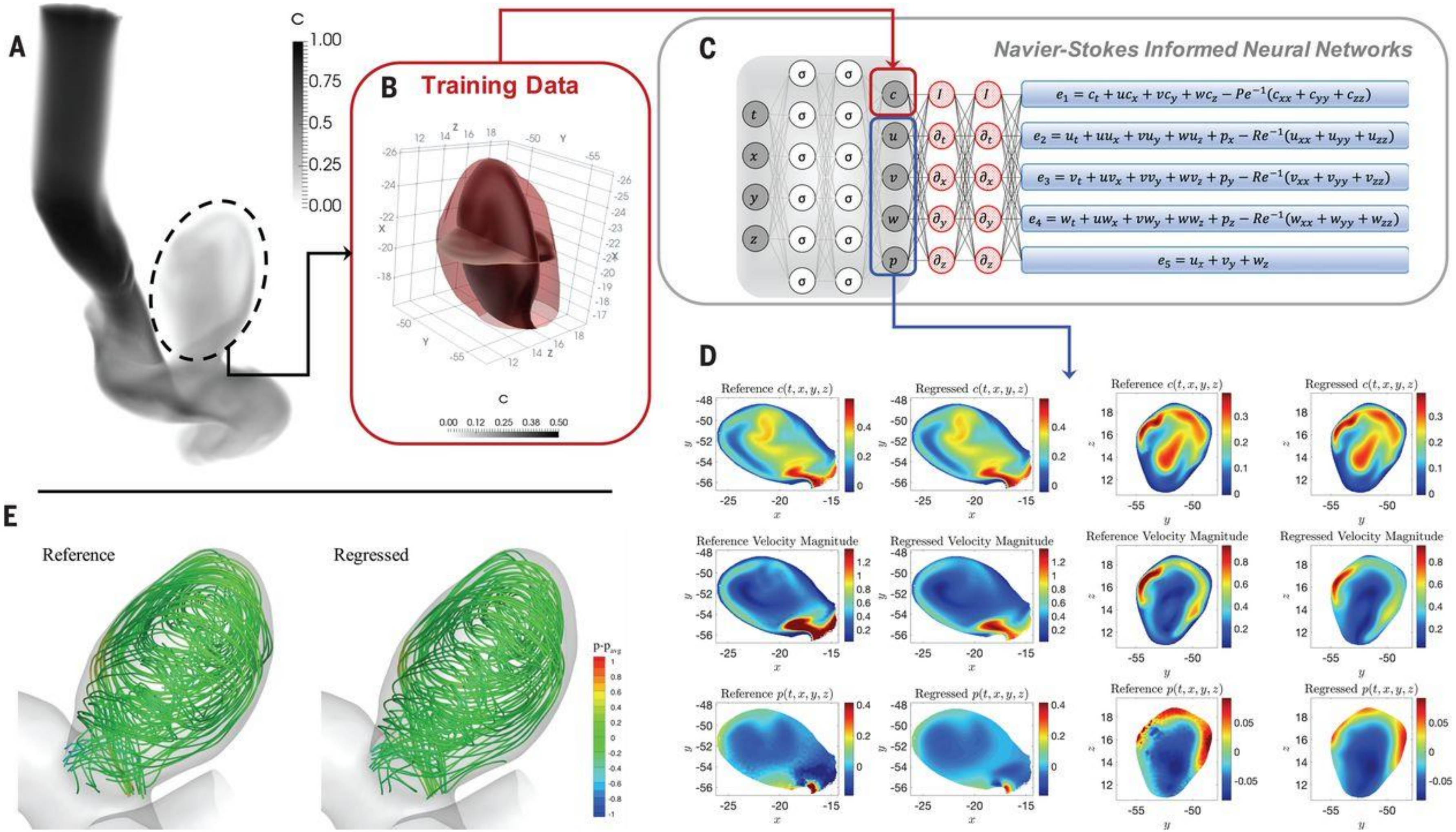
TABLE 2: An overview of variants of PINNs. Variants of PINNs include loss reweighting, novel optimization targets, novel architectures and other techniques such as meta-learning.

Source: https://arxiv.org/abs/2211.08064 *Physics-Informed Machine Learning: A Survey on Problems, Methods and Applications*

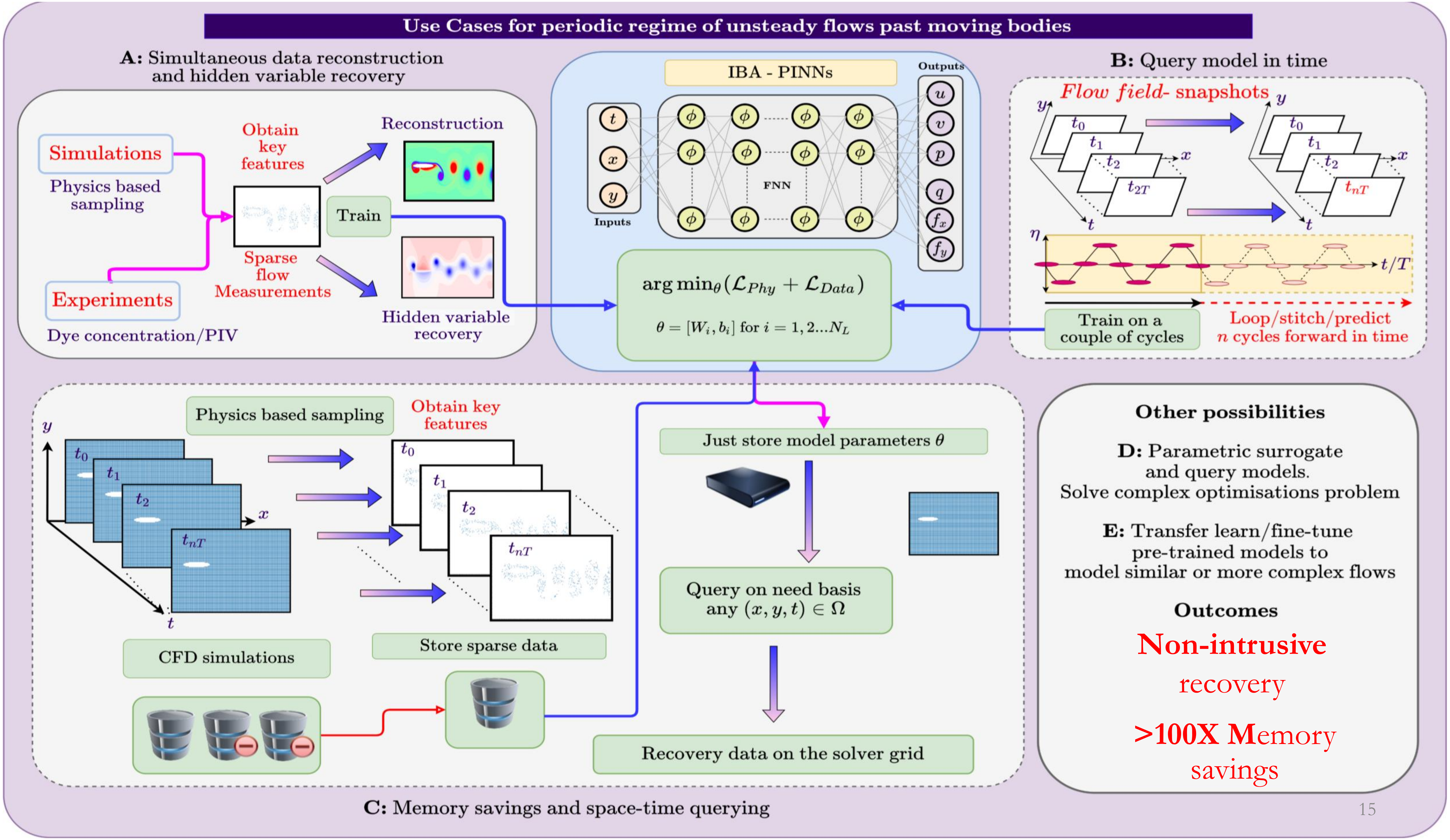
PINNs are flexible universal **physics/science** approximators. So What?

Example in healthcare

Inferring quantitative hemodynamics in 3D intracranian aneurysm



Raissi et al. (2020), Science



Use Cases for periodic regime of unsteady flows past moving bodies
A: Simultaneous data reconstruction and hidden variable recovery
Simulations
Physics based sampling
Experiments
Dye concentration/PIV
Obtain key features
Sparse flow Measurements
Train
Reconstruction
Hidden variable recovery
IBA - PINNs
Inputs
FNN
Outputs
$\arg\min_\theta(\mathcal{L}_{Phy} + \mathcal{L}_{Data})$
$\theta = [W_i, b_i]$ for $i = 1, 2 \ldots N_L$
B: Query model in time
Flow field- snapshots
Train on a couple of cycles
Loop/stitch/predict n cycles forward in time
Physics based sampling
Obtain key features
CFD simulations
Store sparse data
Just store model parameters θ
Query on need basis any $(x, y, t) \in \Omega$
Recovery data on the solver grid
C: Memory savings and space-time querying
Other possibilities
D: Parametric surrogate and query models. Solve complex optimisations problem
E: Transfer learn/fine-tune pre-trained models to model similar or more complex flows
Outcomes
Non-intrusive recovery
>100X Memory savings

PINNS across different domains

Domain	**Applications**
Aerospace & Mech.	Aerodynamics, aeroelastic design, noise reduction
Energy	Reservoir flow, geothermal, CO_2 storage, wind farm layouts
Civil/Structural	Seismic response, stress analysis, predictive maintenance
Healthcare	Blood flow, tumor growth, drug diffusion, cardiac models
Climate & Earth	Ocean currents, turbulence, weather extremes
Materials	Fracture mechanics, heat transport, crystal growth
Electronics/Photonics	Maxwell solvers, antenna design, photonic circuits
Chemistry/Bioprocess	Reaction–diffusion, enzyme kinetics, combustion
Manufacturing	Additive manufacturing defects, real-time digital twins
Environment	Groundwater, pollution dispersion, renewable systems

Maturity level of PINNS across different domains

Stage	Domains	Examples
High Maturity (Now)	Physics/Engineering, Aerospace, Energy	CFD surrogates, aero design, reservoir modeling
Medium (Emerging)	Civil, Healthcare, Climate	Seismic safety, blood flow, ocean circulation
Low (Frontier)	Materials, Electronics, Chemistry, Manufacturing, Sustainability, **Math**	Quantum transport, chip design, digital twins, pollutant modelling, **Computer aided math [DeepMind]**

PINNs can solve meaningful inverse problems and forward problems. Now what are the implications?

Business implications of PINNs

Business Theme	Examples of Domains	Implications
Cost Reduction & Efficiency	Physics, Aerospace, Energy, Civil	Replace costly FEM/CFD simulations, cut design cycles, optimize drilling & construction, reduce downtime.
Faster Time-to-Market	Aerospace, Electronics, Materials	Accelerate R&D in aircraft, chips, photonics, and new materials → competitive edge in innovation cycles.
Risk Management & Compliance	Energy, Civil, Climate, Healthcare	Safer CO_2 sequestration, seismic safety checks, climate risk analytics, regulatory approvals in medtech.
Personalization & Digital Twins	Healthcare, Manufacturing, Civil	Patient-specific modeling, real-time factory twins, predictive infrastructure maintenance.
Sustainability & ESG Value	Climate, Environmental, Chemistry	Better renewable energy optimization, pollutant dispersion modeling, green chemistry & circular economy.
New Business Models	Cross-domain	Simulation-as-a-service, AI-driven design platforms, climate analytics startups, medtech simulation services.

Limitations and future work

- Although grid agnostic, each PINN models only a **single parametric instance**. **Operator learning** methods such as **DeepONets can be explored for multi-parametric regime**.

- PINNs take **long time to train** because of being point-wise mapping models. CNNs are quicker to train but cannot work with arbitrary domains.

- **To make physics-informed training of models robust, further work** is necessary in designing diagnostics
  - To determine apriori if the models will be capable of satisfying the objective
  - To design model architectures capable of handling different spatio-temporal complexities such as multiple bodies, discontinuities, and scales.

Future: Large Science Foundation Models! Generalisable across /specialized for different physical systems!

THANK YOU

DhyutiLabs – Our open source community initiative to serve as a sandbox to test out your complex deep science ideas

Google Colab Notebook used in this session [*Save a copy and experiment at your own pace.* **]**
https://colab.research.google.com/drive/1UN2B8HNfpOmohF6wM2_myq55evupZfKd#scrollTo=Izbu-AVcsyt6

GitHub Repository:*[Look out for byte sized open source challenges in the Github issues]*
https://github.com/DhyutiLABS/DhyutiWalkthroughs

Join the Dhyuti community whatsApp Channel
https://www.whatsapp.com/channel/0029VbBVNONEawdv4268800e

Substack Newsletter
https://substack.com/@dhyutilabs?utm_source=share&utm_medium=android&r=6h5spm

References

1. Chin, Diana D., and David Lentink. "Flapping wing aerodynamics: from insects to vertebrates." *Journal of Experimental Biology* **219.7** (2016): 920-932.
2. Abdelkefi, Abdessattar. "Aeroelastic energy harvesting: A review." *International Journal of Engineering Science* **100** (2016): 112-135.
3. Bearman, P. W. "Circular cylinder wakes and vortex-induced vibrations." *Journal of Fluids and Structures* **27.5-6** (2011): 648-658.
4. I. E. Lagaris, A. Likas and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," in *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 987-1000, Sept. 1998, doi: 10.1109/72.712178.
5. Fukami, K., Hasegawa, K., Nakamura, T. *et al.* Model Order Reduction with Neural Networks: Application to Laminar and Turbulent Flows. *SN COMPUT. SCI.* **2,** 467 (2021). https://doi.org/10.1007/s42979-021-00867-3
6. Majumdar, D. ,et. al, "Capturing the dynamical transitions in the flow-field of a flapping foil using Immersed Boundary Method." *Journal of Fluids and Structures* , **95** (2020): 102999.
7. Menon, K., & Mittal, R. (2020). Dynamic mode decomposition-based analysis of flow over a sinusoidally pitching airfoil. *Journal of Fluids and Structures*, **94**, 102886.
8. Balajewicz, M., & Farhat, C. (2014). Reduction of nonlinear embedded boundary models for problems with evolving interfaces. *Journal of Computational Physics*, *274*, 489-504.
9. Hornik, K., et.al "Multilayer feedforward networks are universal approximators." *Neural networks* **2.5** (1989): 359-366.
10. Sirovich, Lawrence. "Turbulence and the dynamics of coherent structures. I. Coherent structures." *Quarterly of applied mathematics,* **45.3** (1987): 561-571.
11. Baldi, Pierre, and Kurt Hornik. "Neural networks and principal component analysis: Learning from examples without local minima." *Neural networks,* **2.1** (1989): 53-58.

References

12. Kutz, J. Nathan. "Deep learning in fluid dynamics." *Journal of Fluid Mechanics* **814** (2017): 1-4.
13. Brunton, S. L.,et. al. "Machine learning for fluid mechanics." *Annual Review of Fluid Mechanics,* **52** (2020): 477-508.
14. Karniadakis, G. E., et al. "Physics-informed machine learning." *Nature Reviews Physics* 3.6 (2021): 422-440.
15. Benner, P., et.al. "A survey of projection-based model reduction methods for parametric dynamical systems." *SIAM review* **57.4** (2015): 483-531.
16. Yu, J., et. al. "Non-intrusive reduced-order modeling for fluid problems: A brief review." *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* **233.16** (2019): 5896-5912.
17. Gupta, R., & Jaiman, R. (2022). A hybrid partitioned deep learning methodology for moving interface and fluid–structure interaction. *Computers & Fluids*, *233*, 105239.
18. Farhat, C., et al. "5 Computational bottlenecks for PROMs: precomputation and hyperreduction." *Snapshot-Based Methods and Algorithms*. De Gruyter, 2020. 181-244.
19. Lee, K., and Kevin T. C. "Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders." *Journal of Computational Physics* 404 (2020): 108973.
20. Schmid, Peter J. "Dynamic mode decomposition of numerical and experimental data." *Journal of fluid mechanics* 656 (2010): 5-28.
21. Gao, H., et. al "Non-intrusive model reduction of large-scale, nonlinear dynamical systems using deep learning." *Physica D: Nonlinear Phenomena,* **412** (2020): 132614.
22. Gao, H., & Wei, M. (2014). Global model reduction for flows with moving boundary. In *52nd aerospace sciences meeting* (p. 0222).
23. Eivazi, H., et al. "Deep neural networks for nonlinear model order reduction of unsteady flows." *Physics of Fluids* ,**32.10** (2020): 105104.
24. Calicchia, Michael A., et al. "Reconstructing the pressure field around swimming fish using a physics-informed neural network." *Journal of Experimental Biology 226.8 (2023): jeb244983.*
25. Zhu, Yongzheng, et al. "Physics-informed neural networks for incompressible flows with moving boundaries." *Physics of Fluids* 36.1 (2024).

Community guidelines

- No business / IP sensitive discussions on the forum
- Any relevant publicly available information can be discussed

Let's enable chasing hard problems in hard sciences!