Workshop on Accuracy and Efficiency in Scientific Machine Learning

Hybrid Solvers: AI-Integrated Numerical Simulators for Reliable Real-Time Inference

Somdatta Goswami

Johns Hopkins University

Physics-based Models

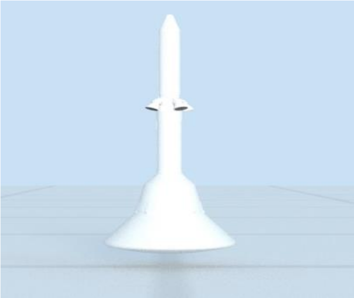
Can represent the Processes of Nature

- Physics-based models are approximated via **ODEs/PDEs**

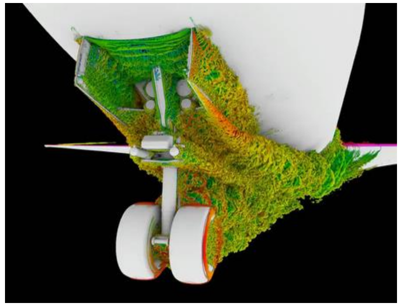
  To model earthquake: $m\dfrac{d^2u}{dt^2} + \mathrm{k}\dfrac{du}{dt} + F_0 = 0$

  To model waves: $\dfrac{\partial^2 u}{\partial t^2} - \nu^2\left(\dfrac{\partial^2 u}{\partial x^2} + \dfrac{\partial^2 u}{\partial y^2}\right) = 0$

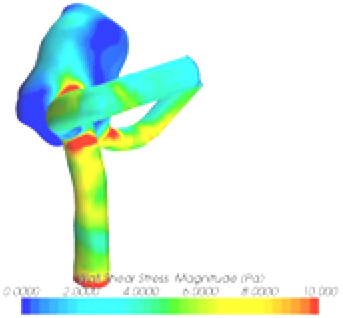
- Computational Mechanics helps us simulate these equations.



Simulation of Orion Spacecraft Launch Abort System (NASA Ames)



Detailed flow around an Aircraft's landing gear (NASA Ames)



CFD Simulation of a Patient-Specific Intracranial Aneurysm

Challenges with Numerical Methods

- Require knowledge of conservation laws, and boundary conditions
- Time consuming and strenuous simulations.
- Difficulties in mesh generation.
- Solving inverse problems or discovering missing physics can be prohibitively expensive.

- Requires very fine mesh to accurately resolve the spatial domain
- Requires small timesteps to resolve the dynamics

Employing surrogate models trained with either data and/or the governing physics of the system (for repeated runs)

Hybrid Solvers:
AI-Integrated Simulators For Spatial and Temporal Coupling

Part-1

Part-2

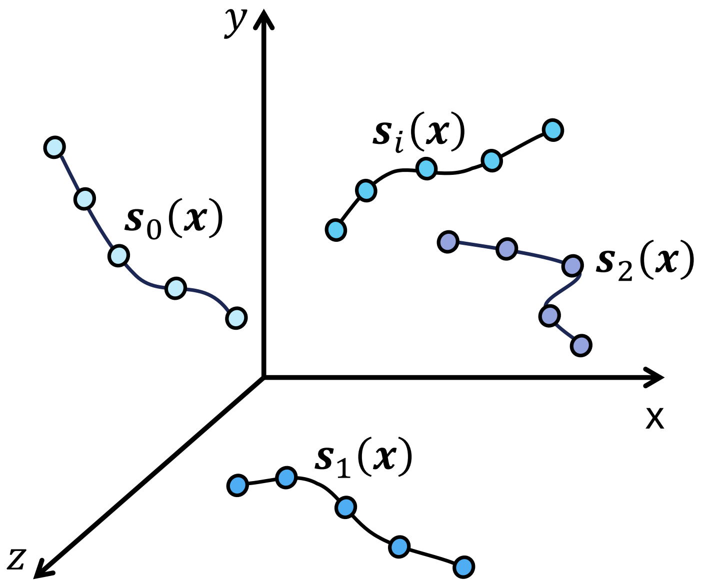
Part-3

Hybrid Solvers:
AI-Integrated Simulators For Spatial and Temporal Coupling

$\underbrace{\text{AI-Integrated Simulators For Spatial and}}_{\textit{Part-2}}$ $\underbrace{\text{Temporal Coupling}}_{\textit{Part-1}}$

$\underbrace{\text{AI-Integrated Simulators For Spatial and Temporal Coupling}}_{\textit{Part-3}}$

Operator Learning



Input space

Initial and Boundary conditions,
Material parameters, etc.

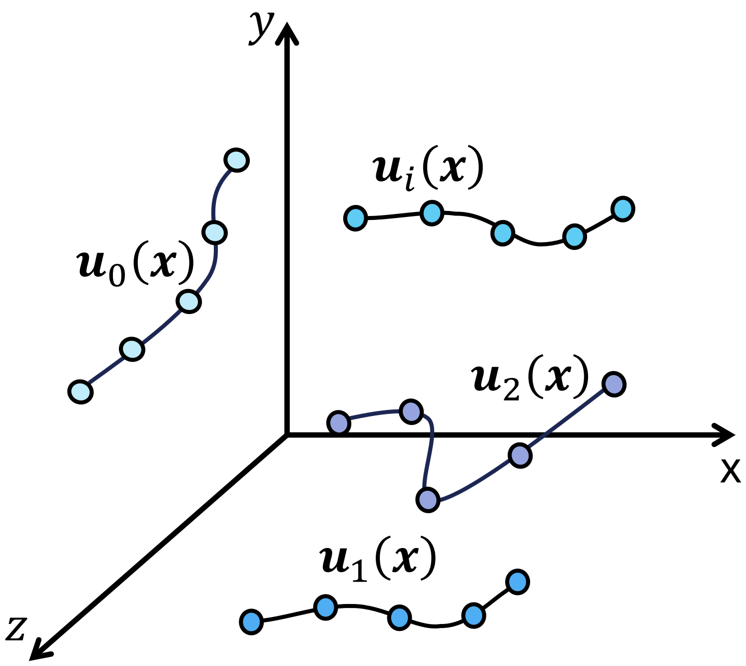
$\Phi: \mathcal{S} \rightarrow \mathcal{U}$

$\Psi: \times \Theta \rightarrow \mathcal{U}$ such that

Training

$\Psi(\cdot, \theta^*) \approx \Phi$

$\theta^* = \operatorname{argmin}_\theta l(\{\mathcal{U}_n, \Psi(\mathcal{S}_n, \theta)\})$

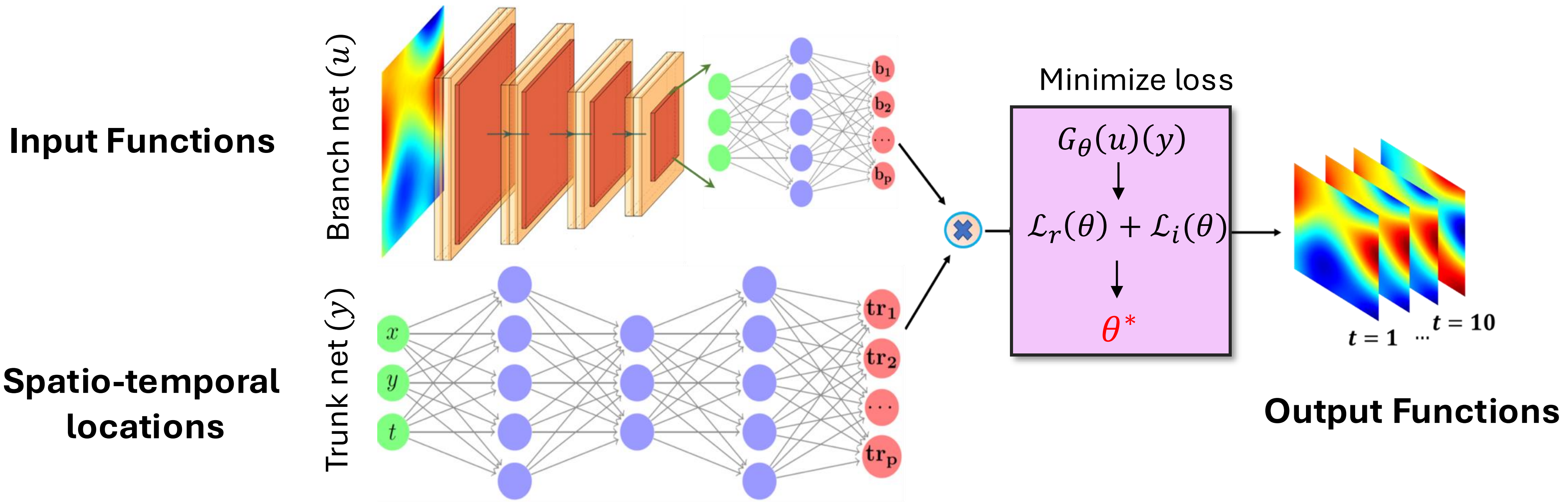


Output space

Displacement, Stress,
Pressure, etc.

Deep Operator Network (DeepONet)

Generalized Universal Approximation Theorem for Operator [Chen '95, Lu et al. '19]

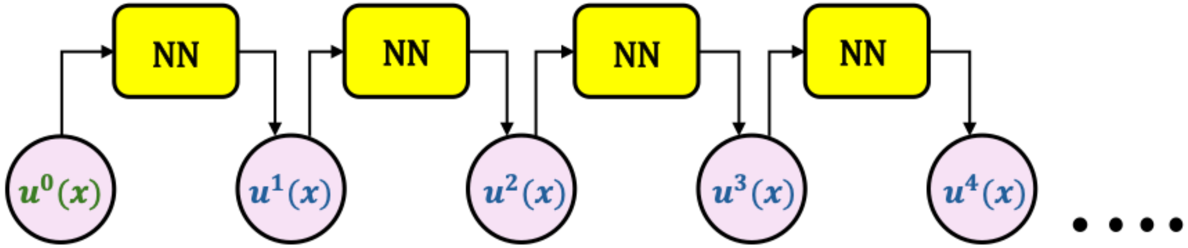


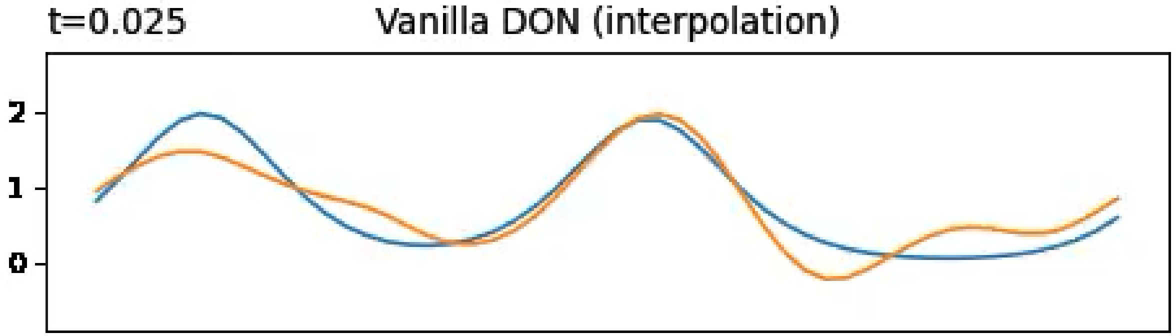
$$G_\theta(u)(y) = \sum_{i=1}^{p} \underbrace{b_i(u(x_1), u\,(x_2), \dots, u\,(x_m))}_{\text{branch net}} \cdot \underbrace{tr_i(y)}_{\text{trunk net}}$$

Challenges of Data-driven ML Models for Dynamical Systems

KdV equation: $u_t - \eta u u_x + \gamma u_{xxx} = 0$

Learning Task: $u(x, t = 0) \rightarrow u(x, t)$,

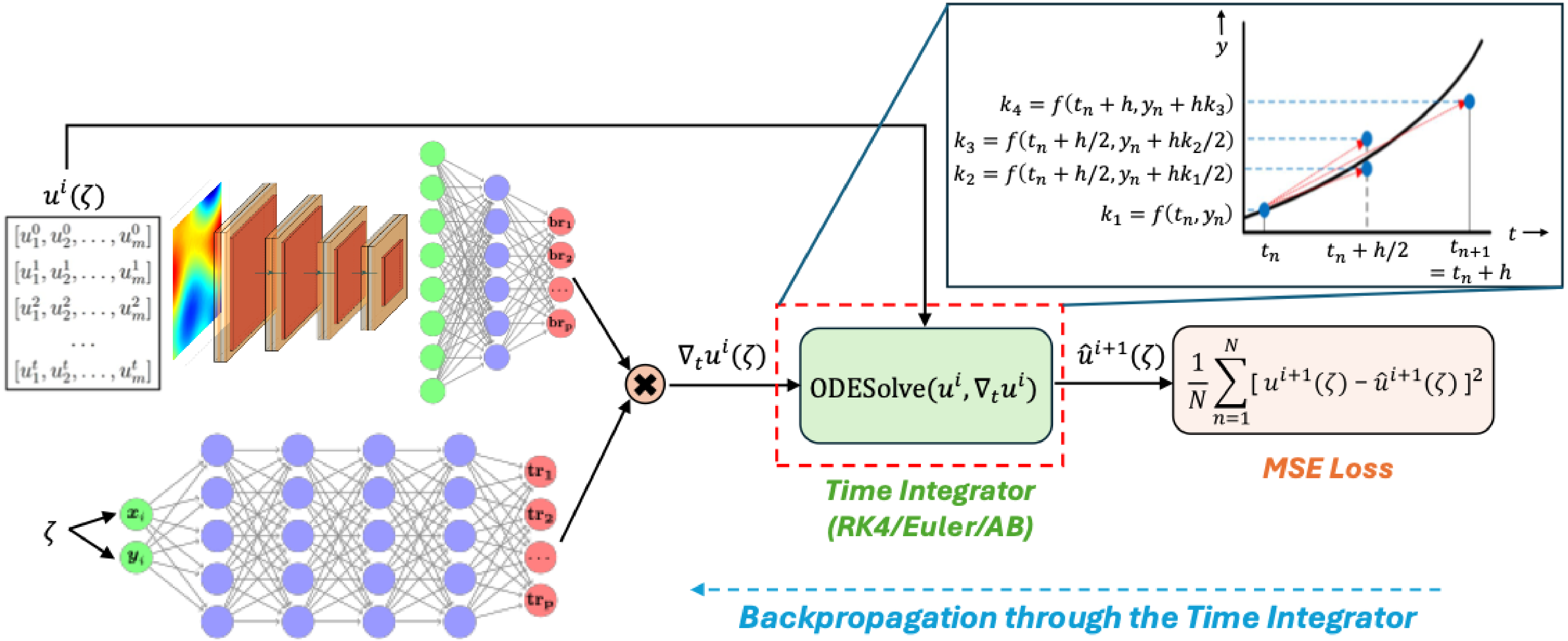




- Prediction done in a recursive manner starting from $u^0(x)$ $\Longrightarrow$ predicted $u^{i+1}(x)$ becomes $u^i(x)$ for next timestep
- Relative L_2 error between predictions and ground truth grows quickly as we progress in time. Also, evident in the contours

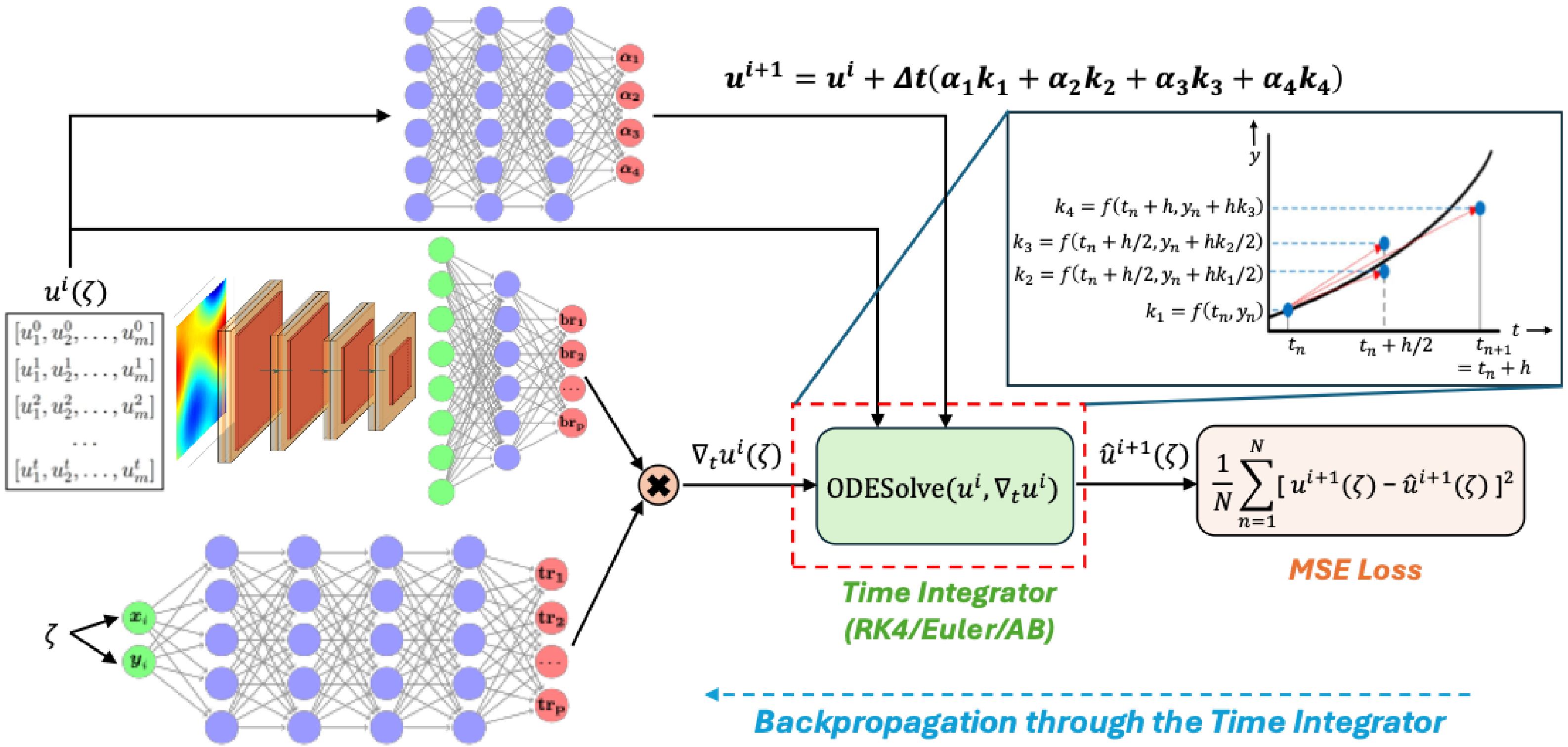
Michałowska, K., Goswami, S., Karniadakis, G. E., & Riemer-Sørensen, S. (2024, June). Neural operator learning for long-time integration in dynamical systems with recurrent neural networks. In *2024 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.

Time Integrated – DeepONet (TI-DeepONet)



While **training**, use **RK4**, however, while **inference** we could use **Adams–Bashforth or RK4**

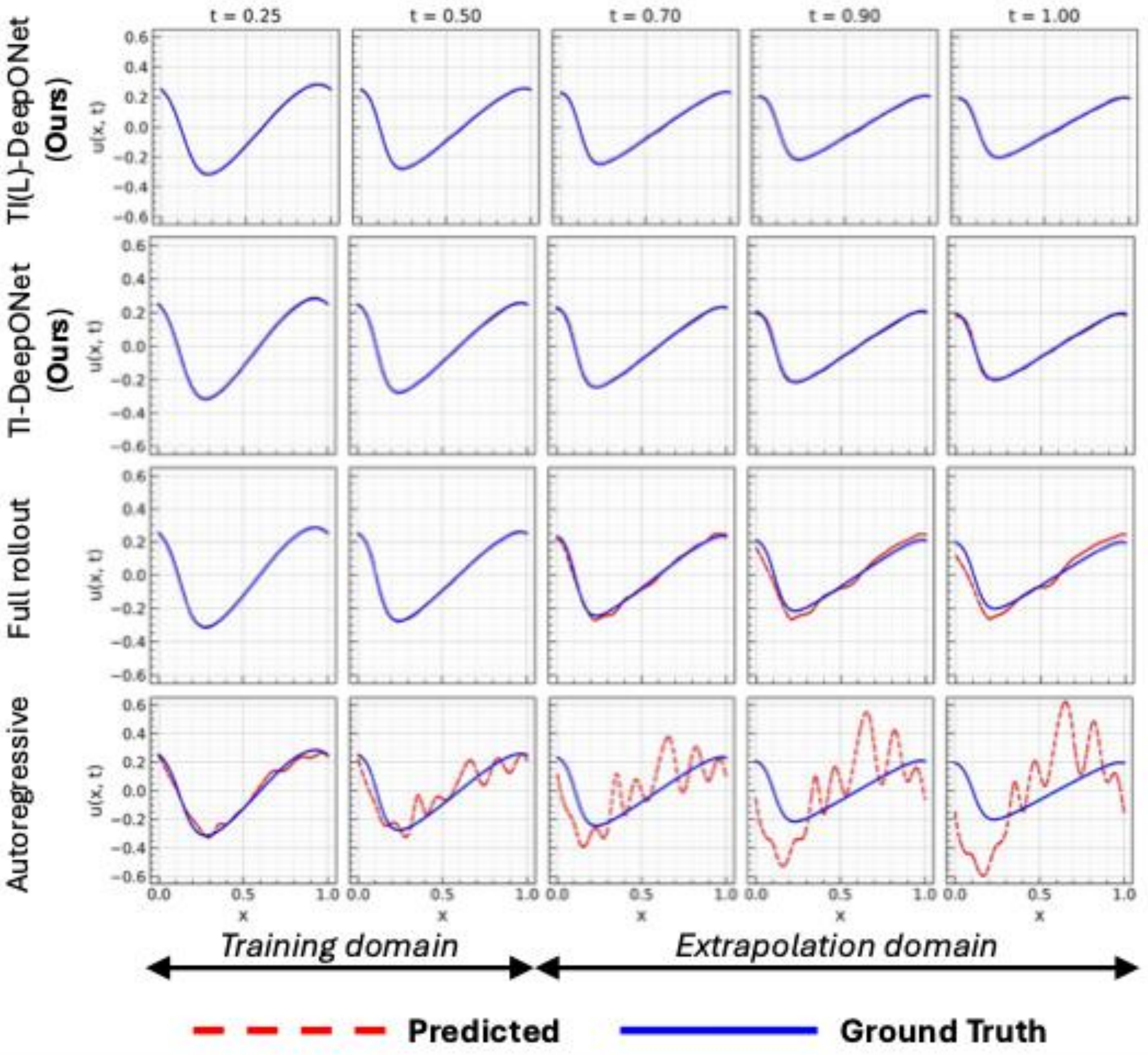
Learnable Time Integrated – DeepONet



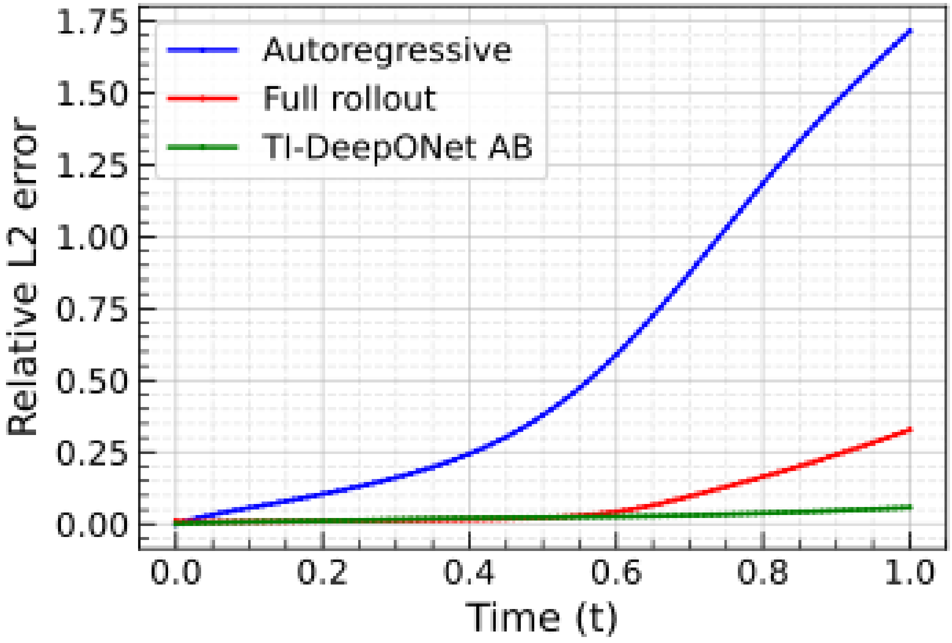
RK4 is employed in **Training** and **Inference.**

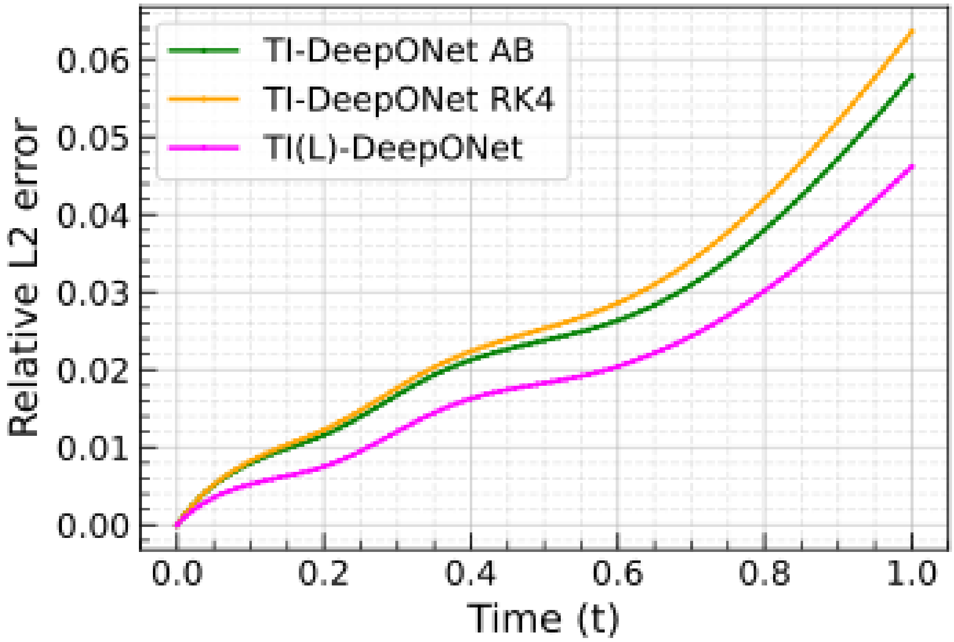
1D Burgers: Results

- Training: $t \in [0, 0.5]$, Inference: $t \in [0, 1]$
- Prediction of the solution profiles by the four different frameworks: (1) TI(L)-DeepONet, (2) TI-DeepONet, (3) Full rollout, and (4) Autoregressive
- Autoregressive accumulates errors early on and quickly deviates from the actual profile
- Full rollout performs well in training domain but starts incurring errors upon entering the extrapolation domain
- Both TI-DeepONet and TI(L)-DeepONet maintain a stable extrapolation resulting in an accurate and reliable prediction of solution state at later timesteps
- TI(L)-DeepONet slightly performs better due to its adaptivity to the local solution



1D Burgers: Results





- Autoregressive: Rapid error accumulation and resembles exponential error growth beyond $t = 0.5$.
- Full rollout: Performs well until $t = 0.5$ and then starts incurring errors.
- TI-DeepONet + AB2/AM3 inference: Stable and controlled error growth especially in extrapolation domain
- With the time integrator frameworks: TI(L)-DeepONet performs best followed by TI-DeepONet AB and then TI-DeepONet RK4

Results for 5 Independent Runs

Problem	Method	Relative L_2 error			
		t+10Δt_e	t+20Δt_e	t+40Δt_e	T^*
Burgers' (1D)	TI(L)-DeepONet	**0.019±0.003**	**0.023±0.003**	**0.036±0.004**	**0.044±0.005**
	TI-DeepONet AB	0.031±0.004	0.037±0.005	0.057±0.008	0.070±0.011
	Full Rollout	0.043±0.002	0.095±0.004	0.247±0.028	0.336±0.053
	Autoregressive	0.710±0.089	1.004±0.144	1.556±0.206	1.768±0.227
KdV (1D)	TI(L)-DeepONet	**0.054±0.019**	**0.065±0.027**	**0.075±0.031**	**0.111±0.051**
	TI-DeepONet AB	0.086±0.026	0.108±0.034	0.129±0.043	0.183±0.063
	Full Rollout	0.776±0.0004	0.716±0.0005	0.719±0.0005	0.795±0.0007
	Autoregressive	0.823±0.073	0.886±0.064	0.922±0.069	0.968±0.083
Burgers' (2D)	TI(L)-DeepONet	**0.111±0.002**	**0.121±0.003**	**0.143±0.004**	**0.155±0.004**
	TI-DeepONet AB	0.121±0.002	0.133±0.002	0.157±0.003	0.169±0.003
	Full Rollout	0.131±0.007	0.194±0.014	0.357±0.035	0.453±0.049
	Autoregressive	0.503±0.017	0.590±0.024	0.783±0.052	0.894±0.075

Nayak, D. and Goswami, S., 2025. TI-DeepONet: Learnable Time Integration for Stable Long-Term Extrapolation. *arXiv:2505.17341*.

Hybrid Solvers:
AI-Integrated Simulators For Spatial and Temporal Coupling

Part-1

Part-2

Part-3

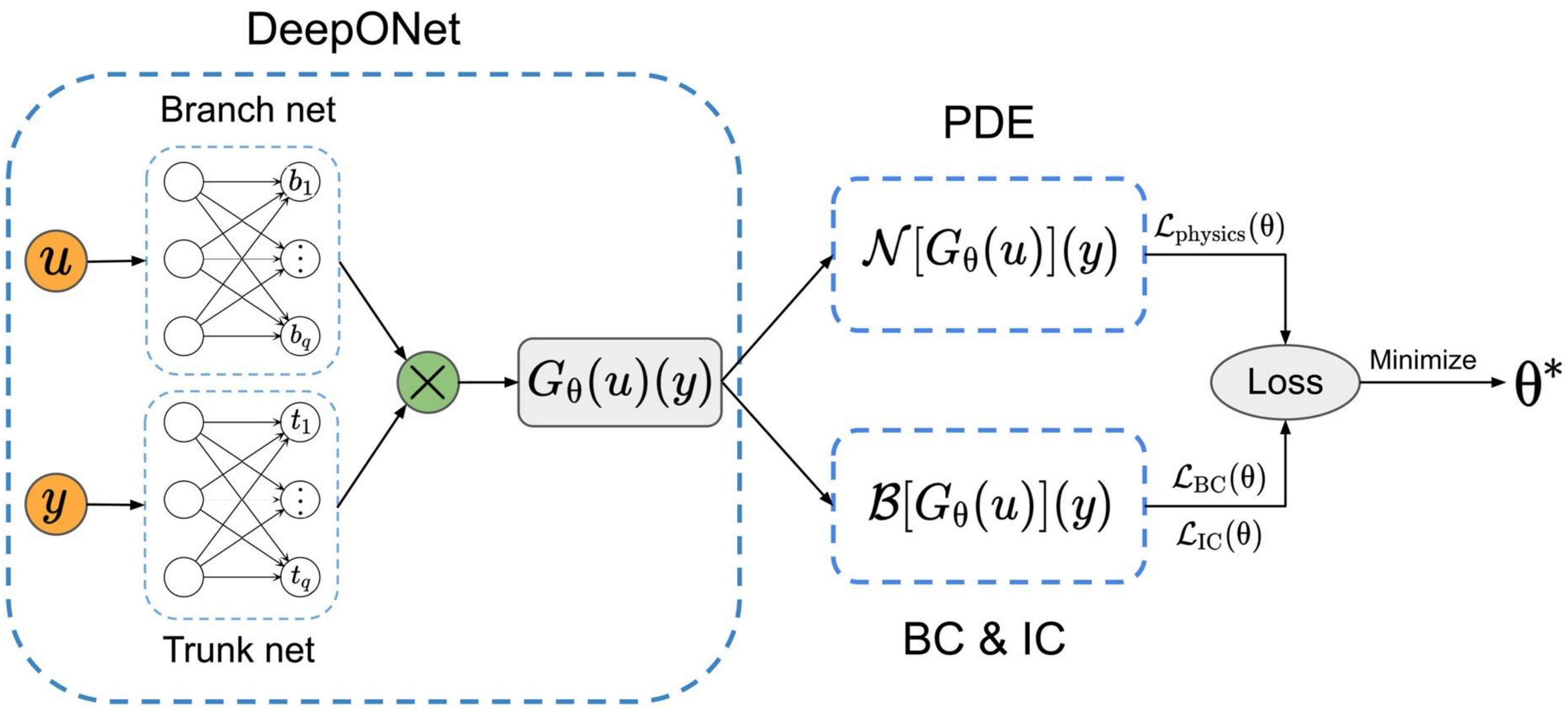
Spatial Hybrid Solver

1. Employ Domain Decomposition Framework:
   Location requiring finer discretization –approximated using **neural operators**
   Locations ‘ok’ with coarser discretization – approximated using numerical solvers.

2. The two solvers handing over an overlapping domain and are coupled using the Alternating Schwartz coupling framework.

3. To make the framework efficient, the subdomain with finer discretization is approximated using physics-informed neural operators.

Physics-Informed DeepONet



- Somdatta Goswami, Yin, M., Yu, Y., & Karniadakis, G. E. (2022). A physics-informed variational DeepONet for predicting crack path in quasi-brittle materials. Computer Methods in Applied Mechanics and Engineering, 391, 114587.
- Sifan Wang, Hanwen Wang, and Paris Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed DeepONets. Science Advances, 7(40), October 2021.

Challenges With Neural Operators

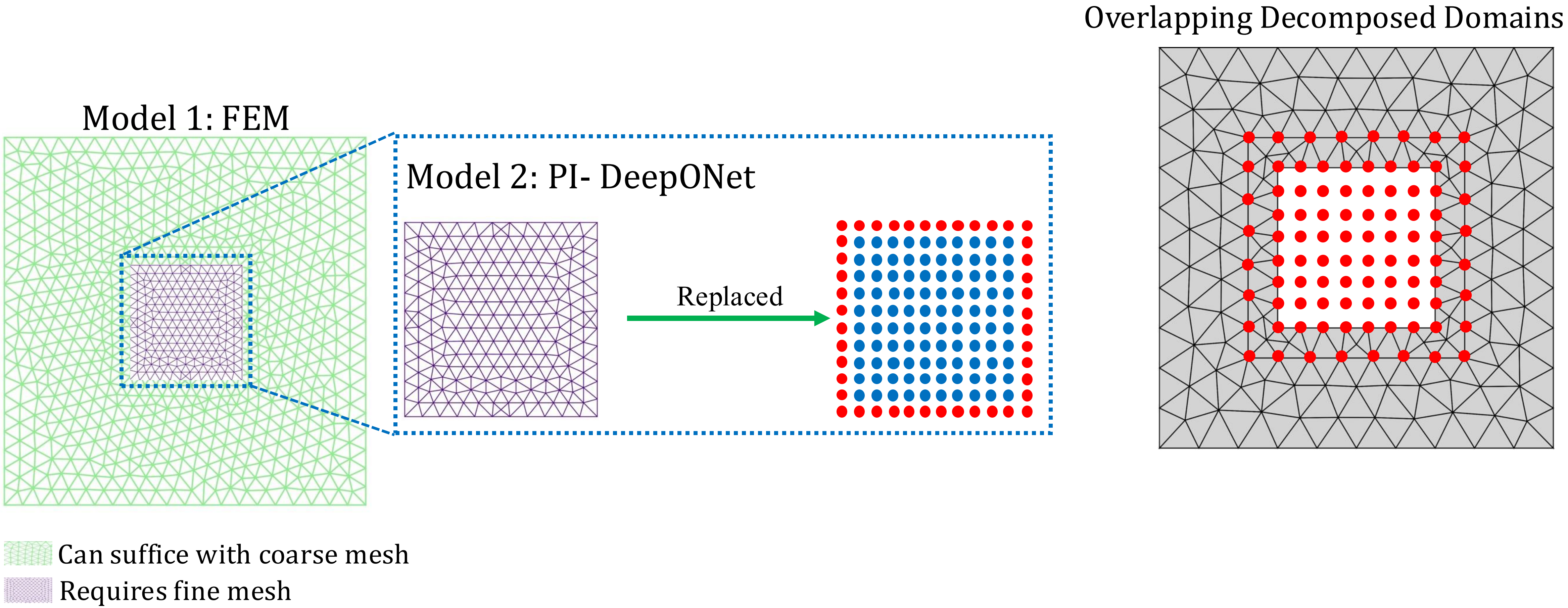
For Physics-Informed Neural Operators:

- Extremely expensive to train* due to the computation of the gradients for large number of function used to represent the function space.
- Cannot efficiently resolve Multiphysics systems**.

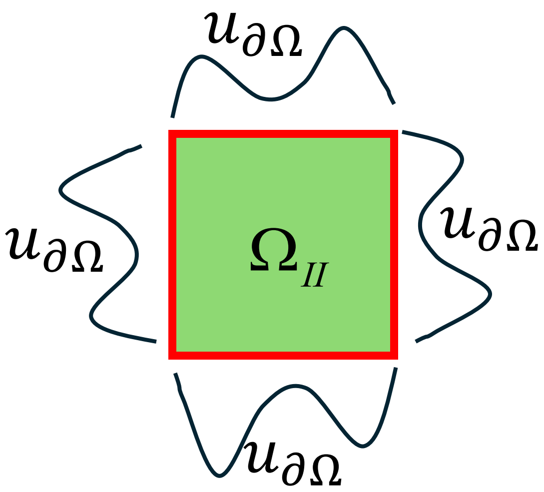
* Mandl, Luis, Somdatta Goswami, Lena Lambers, and Tim Ricken. "Separable physics-informed DeepONet: Breaking the curse of dimensionality in physics-informed machine learning." *Computer Methods in Applied Mechanics and Engineering* 434 (2025): 117586.

** Somdatta Goswami, Yin, M., Yu, Y., & Karniadakis, G. E. (2022). A physics-informed variational DeepONet for predicting crack path in quasi-brittle materials. Computer Methods in Applied Mechanics and Engineering, 391, 114587.

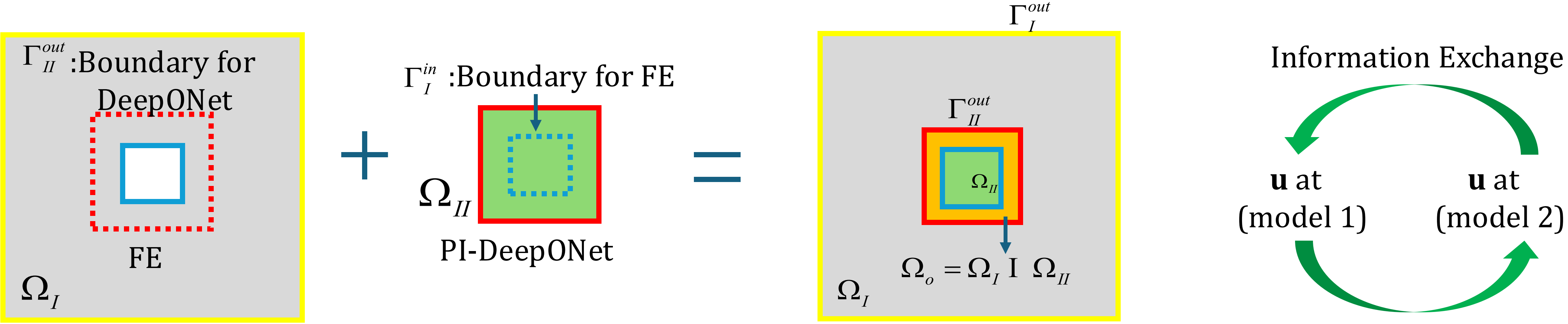
Domain Decomposition Framework



Spatial Domain Coupling



- $u_{\partial\Omega}$ generated using a Gaussian Random field.
- The PI-DeepONet is trained for Ω_{II}. Given the displacement at the boundary, the network learns the solution within the subdomain and gives back the updated displacement to the FE solver.
- The learning of the DeepONet employs only the governing physics and no labeled data.



Schwartz alternating method on Ω_0

Examples

Hybrid coupling for
1. Static loading for linear elastic material
2. Quasi-static loading for hyper elastic material
3. Dynamic Loading for linear elastic material
4. Adaptive expansion of the ML – subdomain for dynamic loading

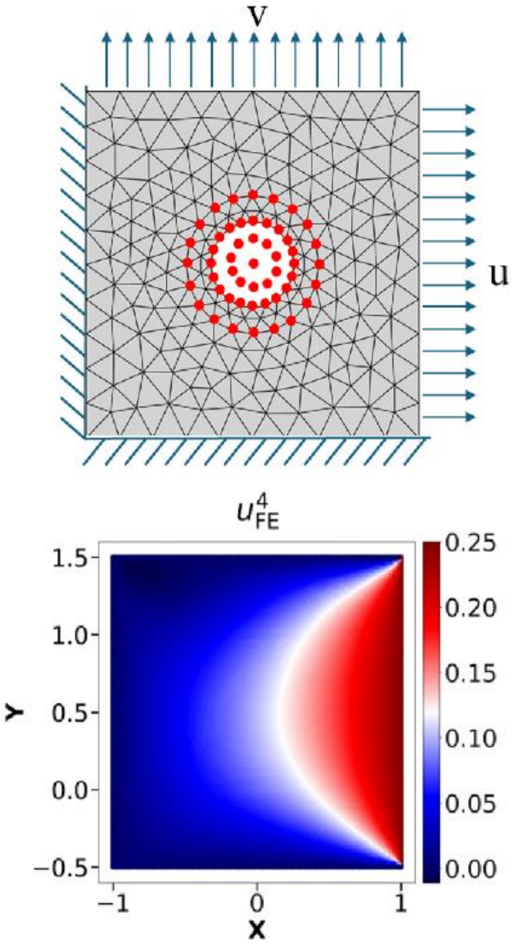
Accelerating Multiscale Modeling with Hybrid Solvers: Coupling FEM and Neural Operators with Domain Decomposition

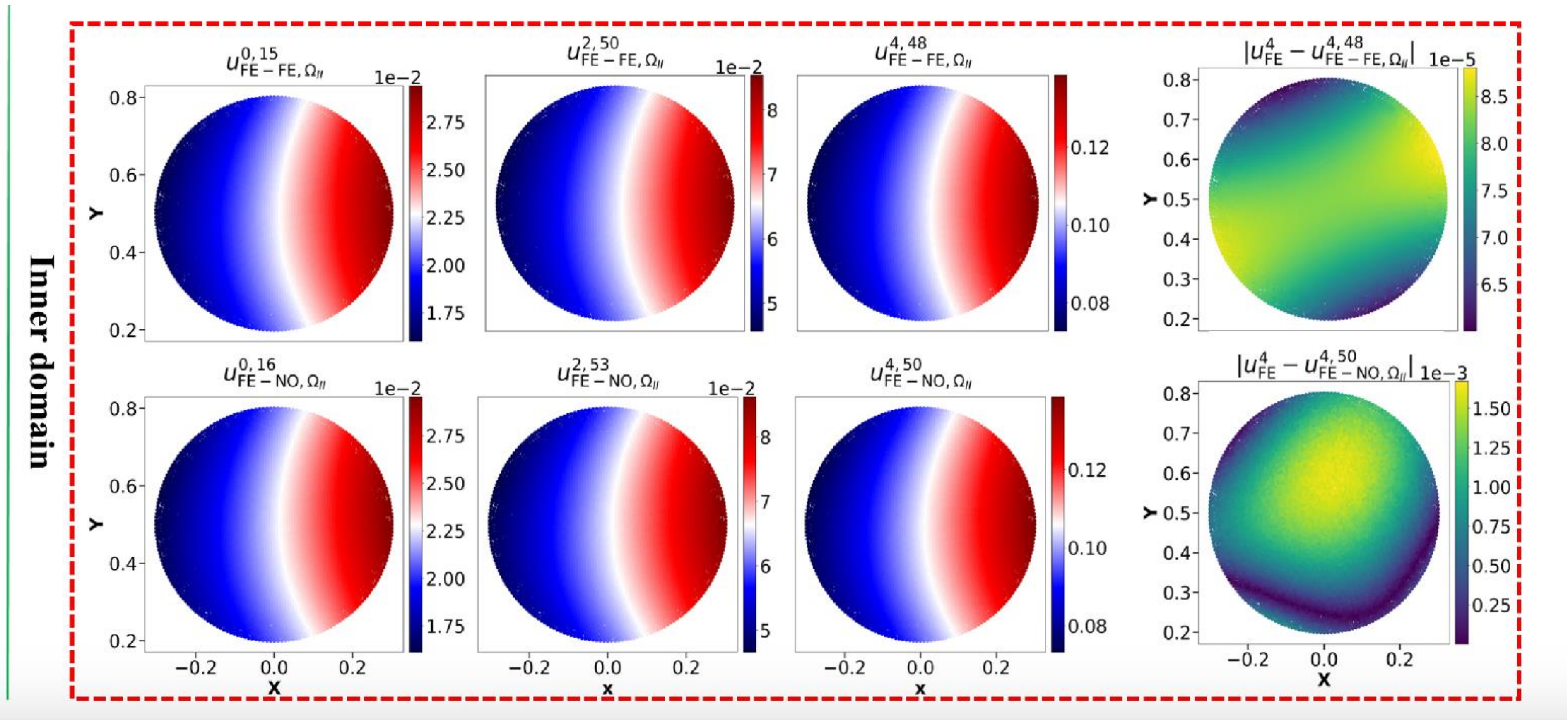
Wei Wang[a,b], Maryam Hakimzadeh[b], Haihui Ruan[a,c,*], Somdatta Goswami[b,*]

[a] *Department of Mechanical Engineering, The Hong Kong Polytechnic University*
[b] *Department of Civil and Systems Engineering, Johns Hopkins University*
[c] *PolyU-Daya Bay Technology and Innovation Research Institute*

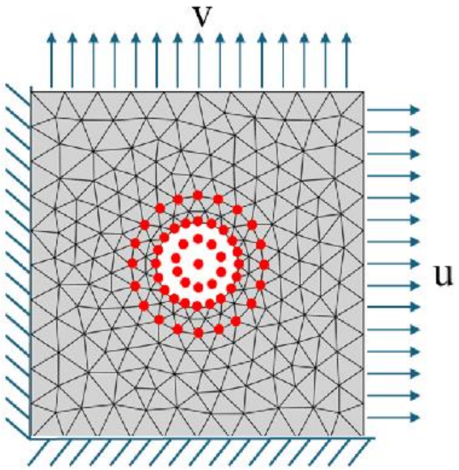
Wang, Wei, Maryam Hakimzadeh, Haihui Ruan, and Somdatta Goswami. "Accelerating Multiscale Modeling with Hybrid Solvers: Coupling FEM and Neural Operators with Domain Decomposition." *arXiv preprint arXiv:2504.11383*(2025).

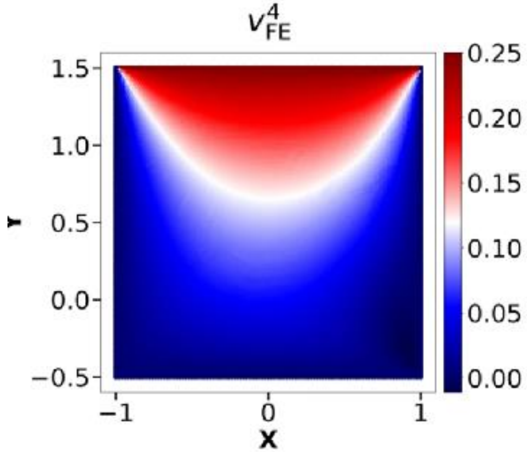
Hyper-elasticity under quasi-static loading conditions

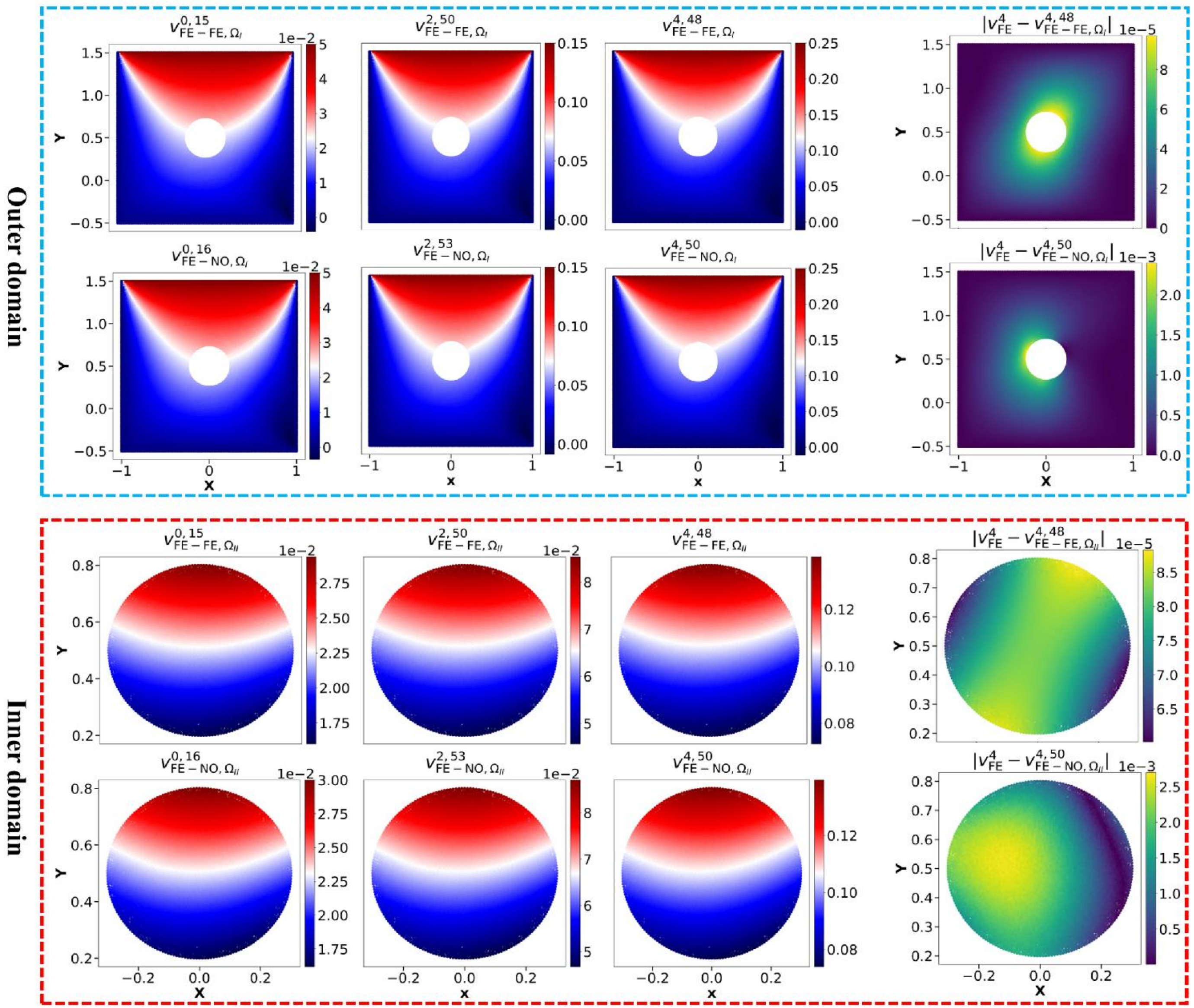




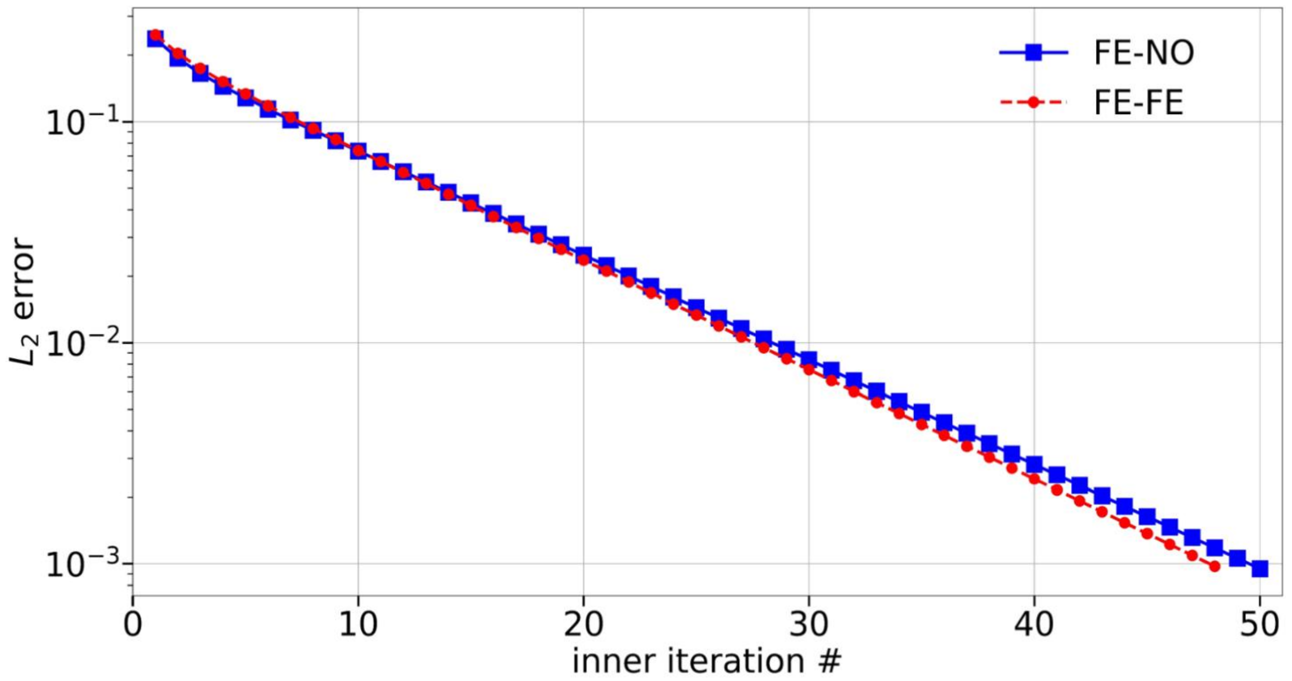
Hyper-elasticity under quasi-static loading conditions







Performance of FE-NO



At t = 4, the neural operators (NO) coupling needs more inner iterations.
No need Newton's solver for additional root-finding iterations at each inner iterations.
FE-NO coupling is 20% faster than FE-FE coupling.

Hybrid Solvers:
AI-Integrated Simulators For Spatial and Temporal Coupling

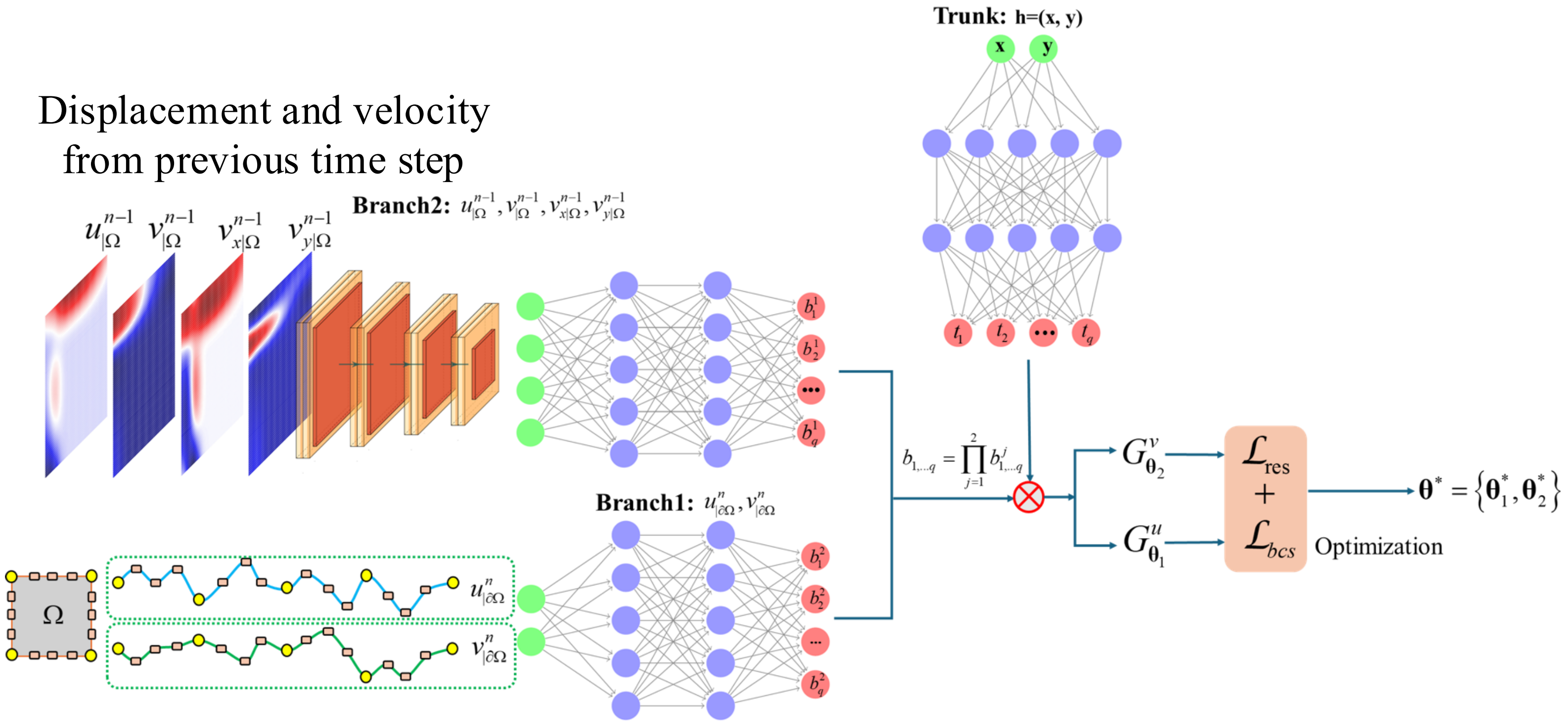
Part-1

Part-2

Part-3

Temporal Dimension Coupling

Displacement and velocity from previous time step



current boundary conditions

Newmark-beta time discretization method:

$$\dot{u}(t+dt) = \dot{u} + dt\left[(1-\gamma)\ddot{u} + \gamma\ddot{u}(t+dt)\right]$$

$$\ddot{u}(t+dt) = \frac{1}{(dt)^2\beta}\left[-u - \dot{u}\,dt + u(t+dt)\right] - \frac{(1-2\beta)}{2\beta}\ddot{u}$$

$$[M]\ddot{u}(t+dt) = [K]u(t+dt) + [F(t+dt)]$$

Examples

Hybrid coupling for
1. Static loading for linear elastic material
2. Quasi-static loading for hyper elastic material
3. Dynamic Loading for linear elastic material
4. Adaptive expansion of the ML – subdomain for dynamic loading

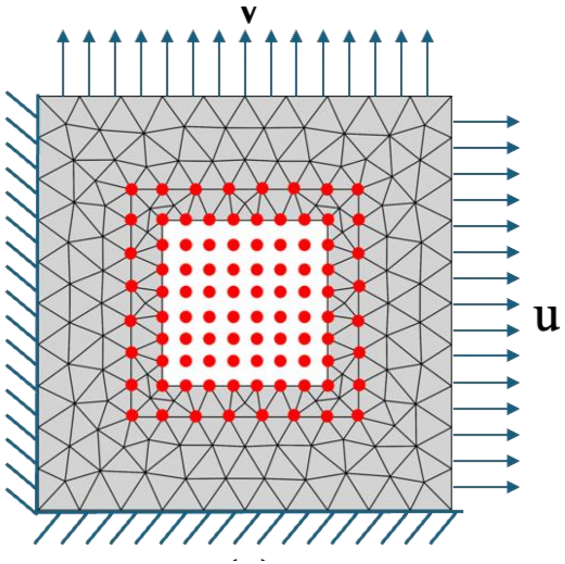
Accelerating Multiscale Modeling with Hybrid Solvers: Coupling FEM and Neural Operators with Domain Decomposition

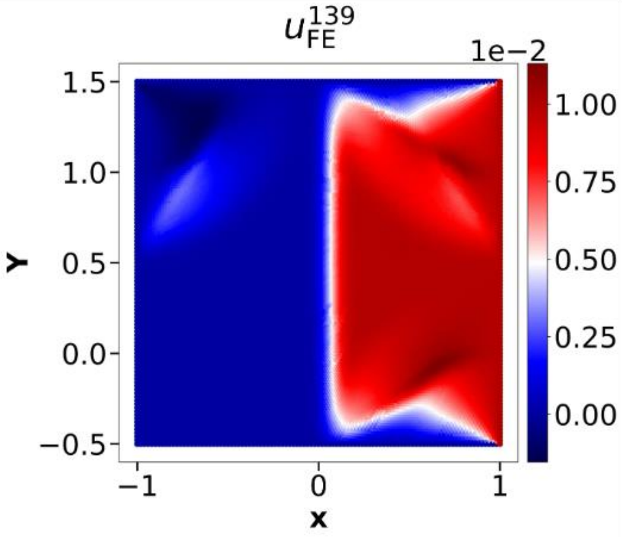
Wei Wang[a,b], Maryam Hakimzadeh[b], Haihui Ruan[a,c,*], Somdatta Goswami[b,*]

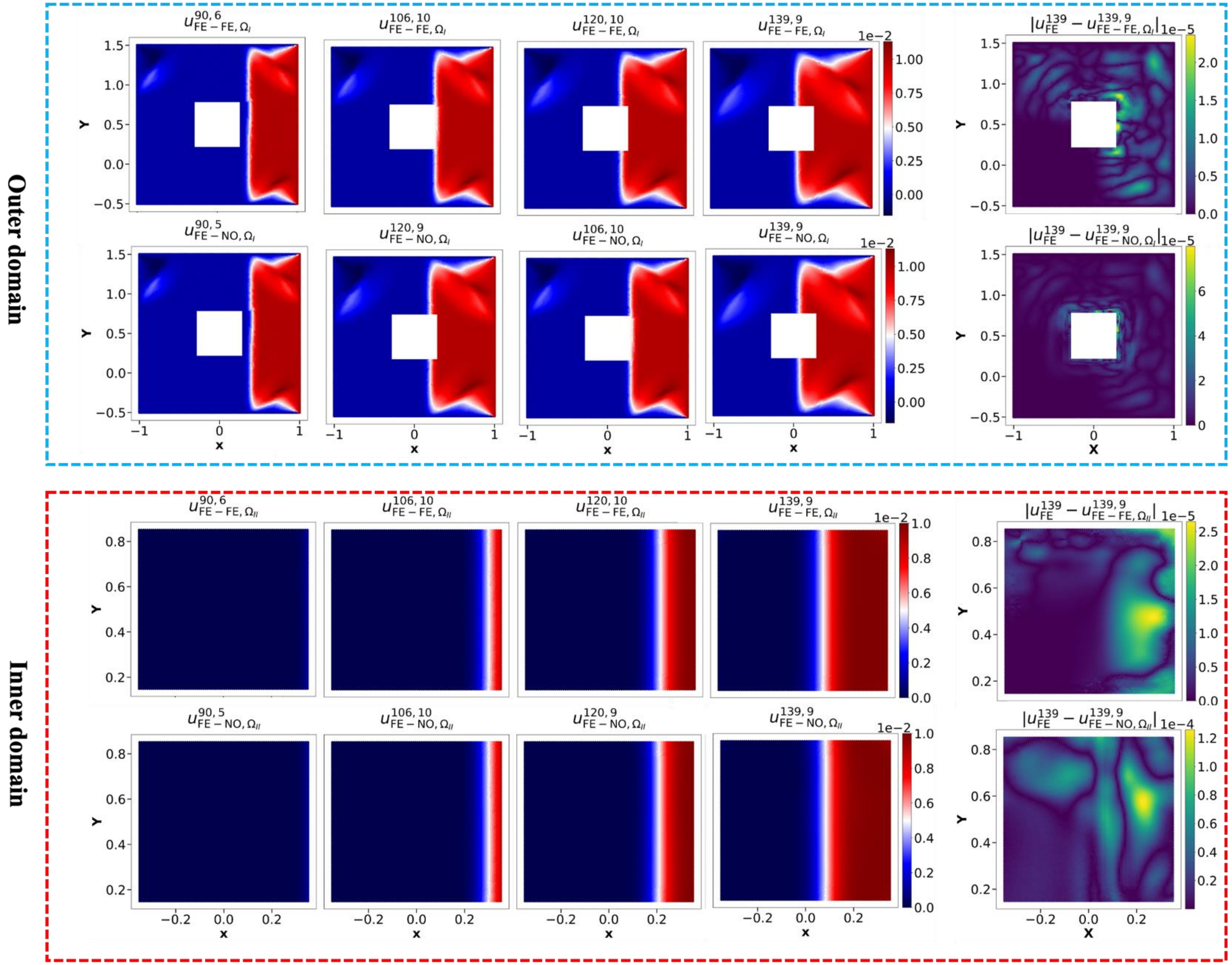
[a] Department of Mechanical Engineering, The Hong Kong Polytechnic University
[b] Department of Civil and Systems Engineering, Johns Hopkins University
[c] PolyU-Daya Bay Technology and Innovation Research Institute

Wang, Wei, Maryam Hakimzadeh, Haihui Ruan, and Somdatta Goswami. "Accelerating Multiscale Modeling with Hybrid Solvers: Coupling FEM and Neural Operators with Domain Decomposition." *arXiv preprint arXiv:2504.11383* (2025).

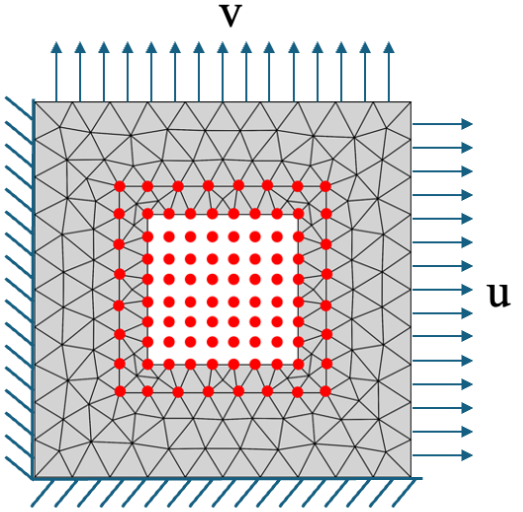
Linear Elastic Model in Dynamic Regime

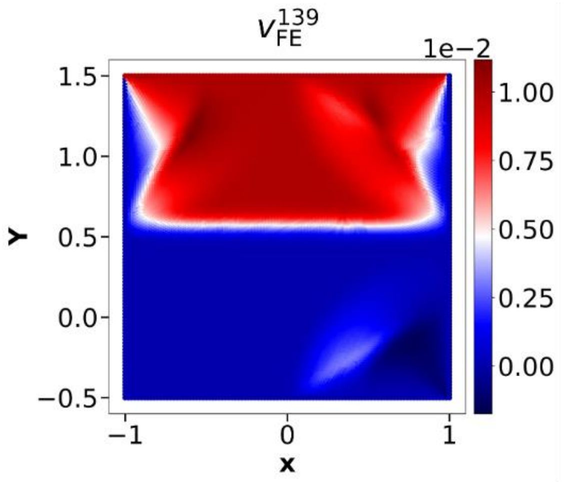


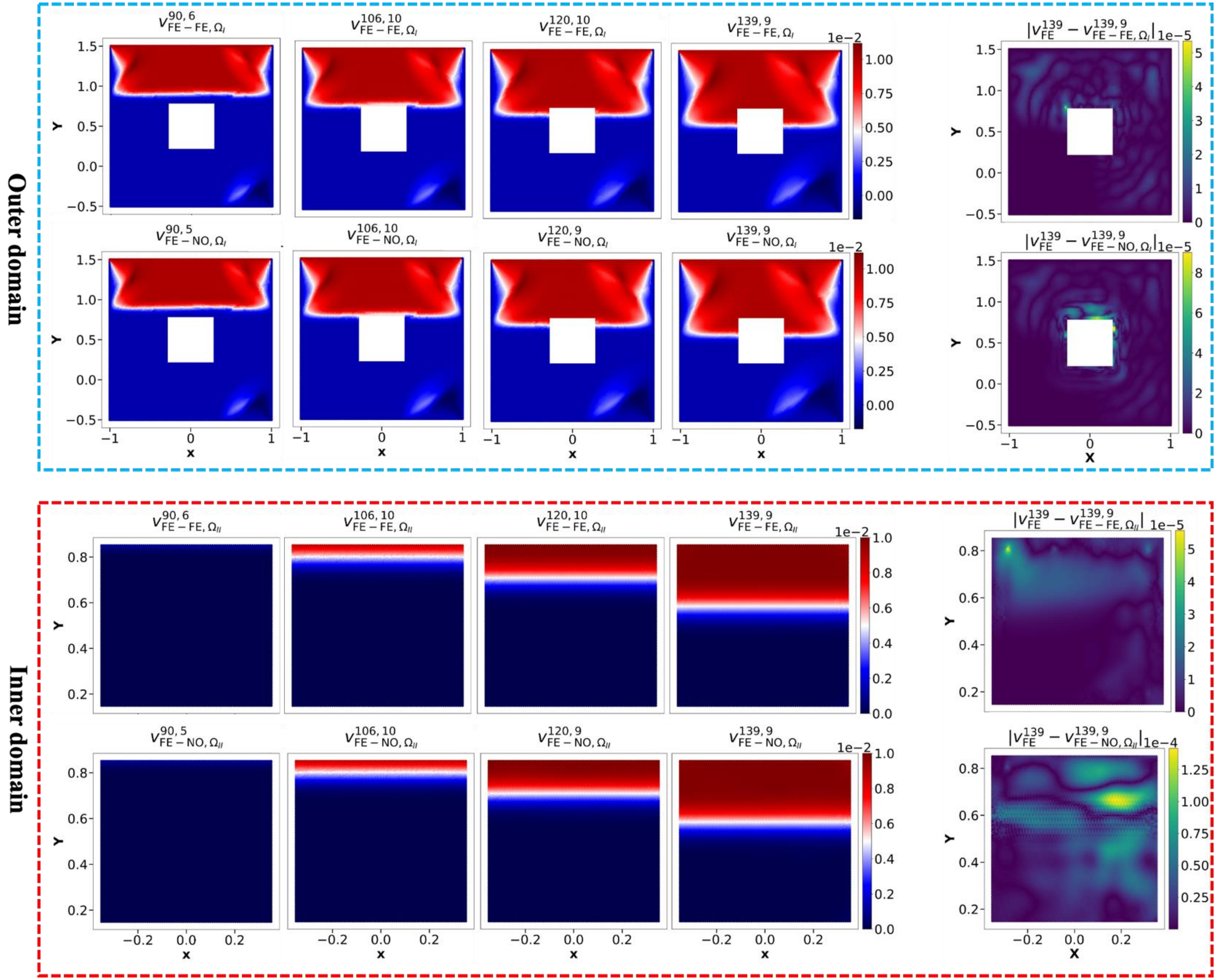




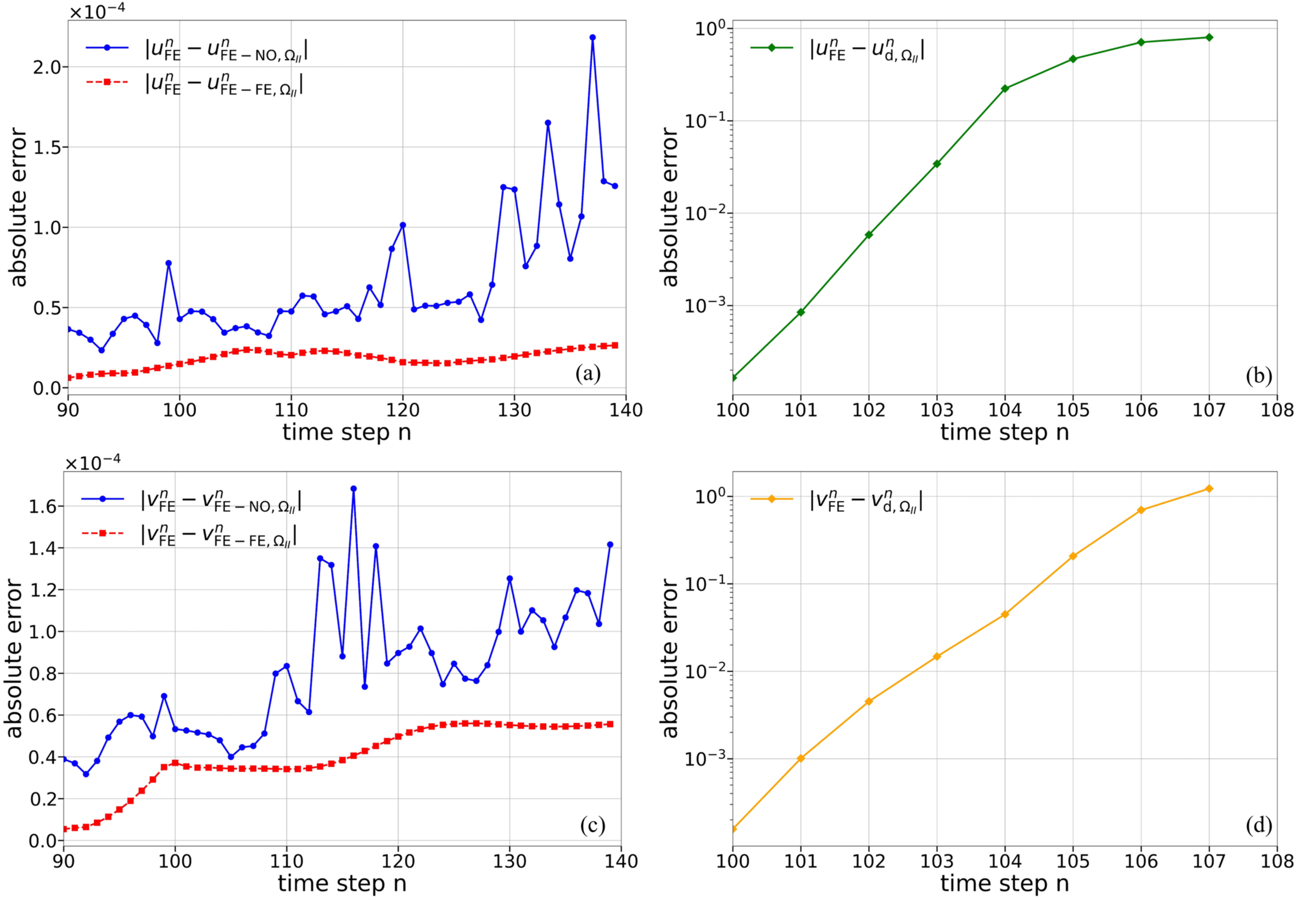
Linear Elastic Model in Dynamic Regime







Improvement in Error Accumulation



Key Points

- The FE-NO coupling reduces the compute time for each high-fidelity simulation by replacing the most finely discretized part of the domain with a pretrained physics-informed NO.

- The training of NO is carried out with only the physics of the governing equation, hence no cost for the generation of labeled data.

- We significantly reduced error accumulation for time dependent systems using Newmark-Beta time marching method. Hence the PI-DeepONet is reliable over long time horizons.

- The hybrid solvers opens up the possibility of accelerating multiscale modeling leveraging ML models.

Funding

Thank you!