AI-Accelerated Numerical Solver: Time-advancing DeepONet-FEM coupling in Dynamic Mechanics

Wei Wang and Somdatta Goswami

Civil and Systems Engineering, Johns Hopkins University

May 29, 2025

Physics-based Models

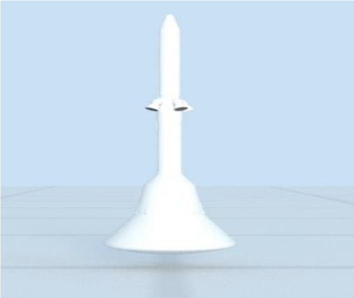
Can represent the Processes of Nature

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

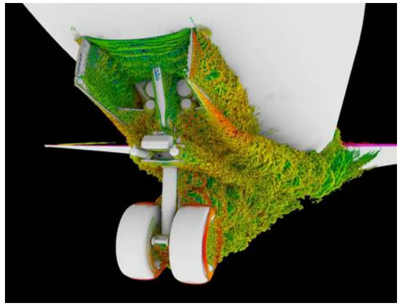
  To model earthquake: $m\frac{d^2u}{dt^2} + \mathrm{k}\frac{du}{dt} + F_0 = 0$

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

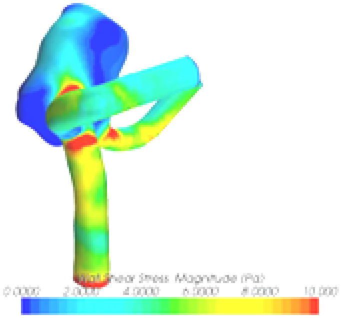
- Computational Mechanics helps us simulate these equations.



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:
Physics-Informed ML-Integrated Numerical Simulators

Part-1

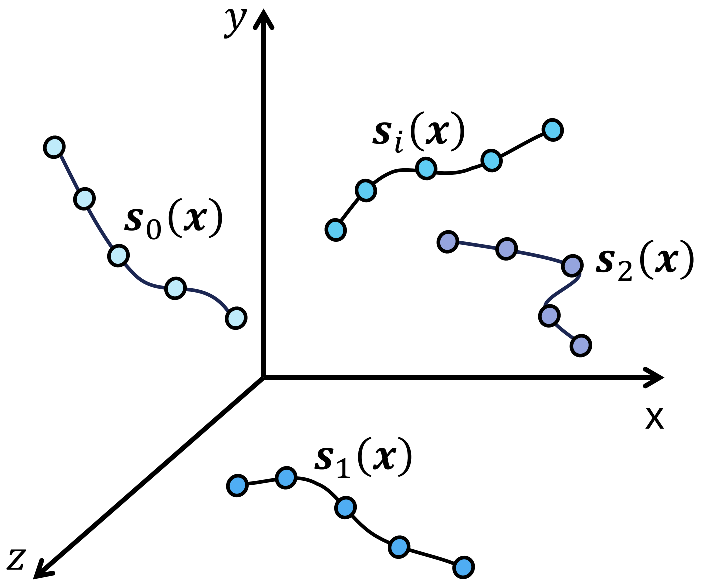
Part-2

Hybrid Solvers:
Physics-Informed ML-Integrated Numerical Simulators

Part-1

Part-2

Operator Learning



Input space

Initial and Boundary conditions,
Material parameters, etc.

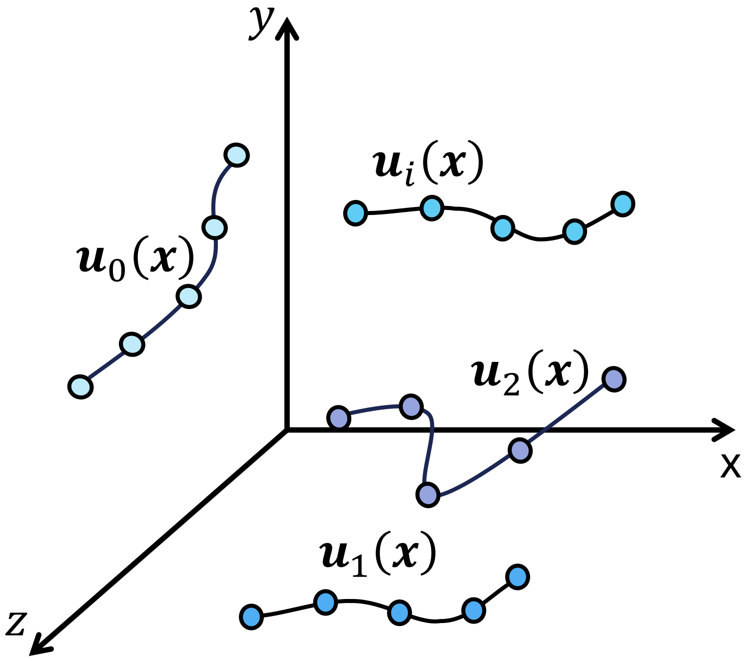
$\Phi : \mathcal{S} \to \mathcal{U}$

Training

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

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

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

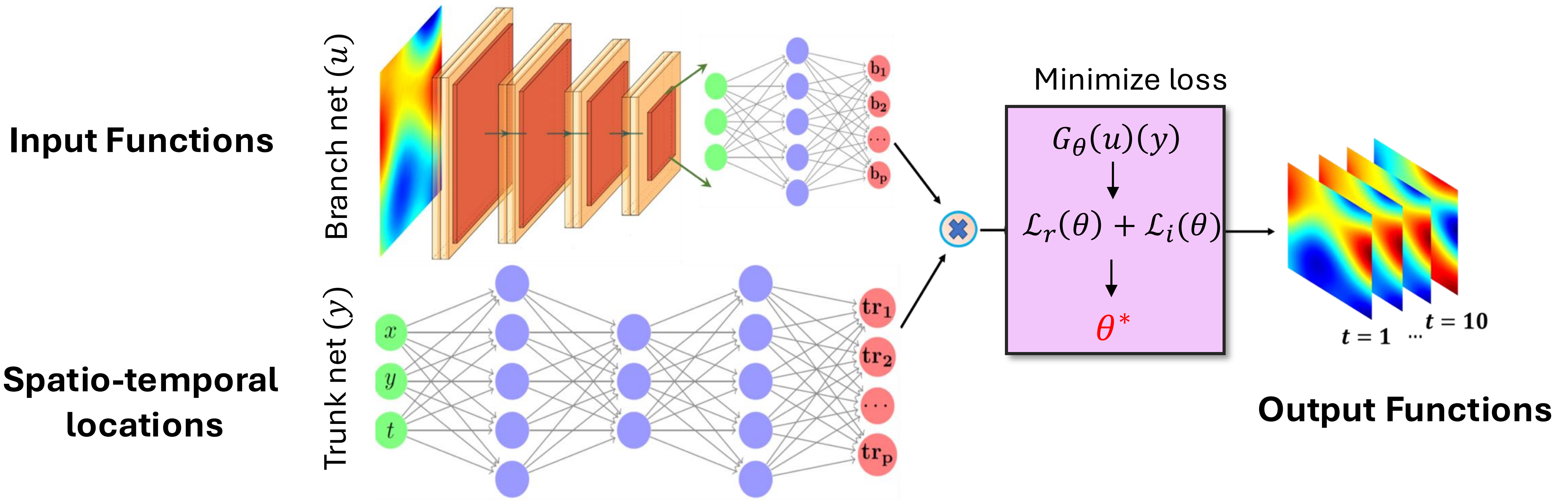


Output space

Displacement, Stress,
Pressure, etc.

Deep Operator Network (DeepONet)

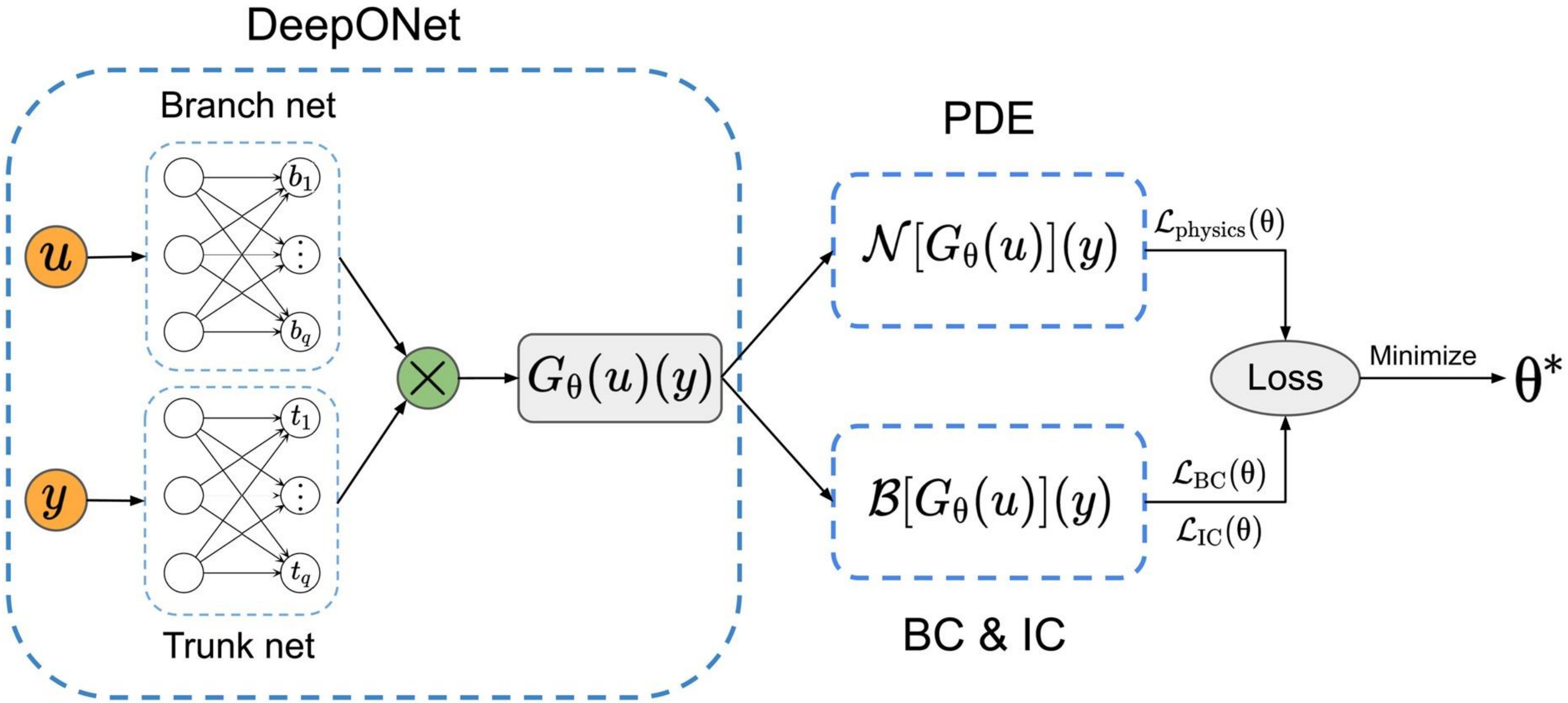
- Generalized Universal Approximation Theorem for Operator [Chen ’95, Lu et al. ’19]



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

Centrum IntelliPhysics

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 Data Driven Models: Requires voluminous amount of high-fidelity training dataset – extensive parametric sweep on the numerical solvers.

- 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.

Hybrid Solvers:
Physics-Informed ML-Integrated Numerical Simulators

$\underbrace{\text{Physics-Informed}}_{\text{Part-1}}$

$\underbrace{\text{Physics-Informed ML-Integrated Numerical Simulators}}_{\text{Part-2}}$

The Hybrid Solver

1. Employ Domain Decomposition Framework:

    Location requiring finer discretization –approximated using **physics informed neural operators**

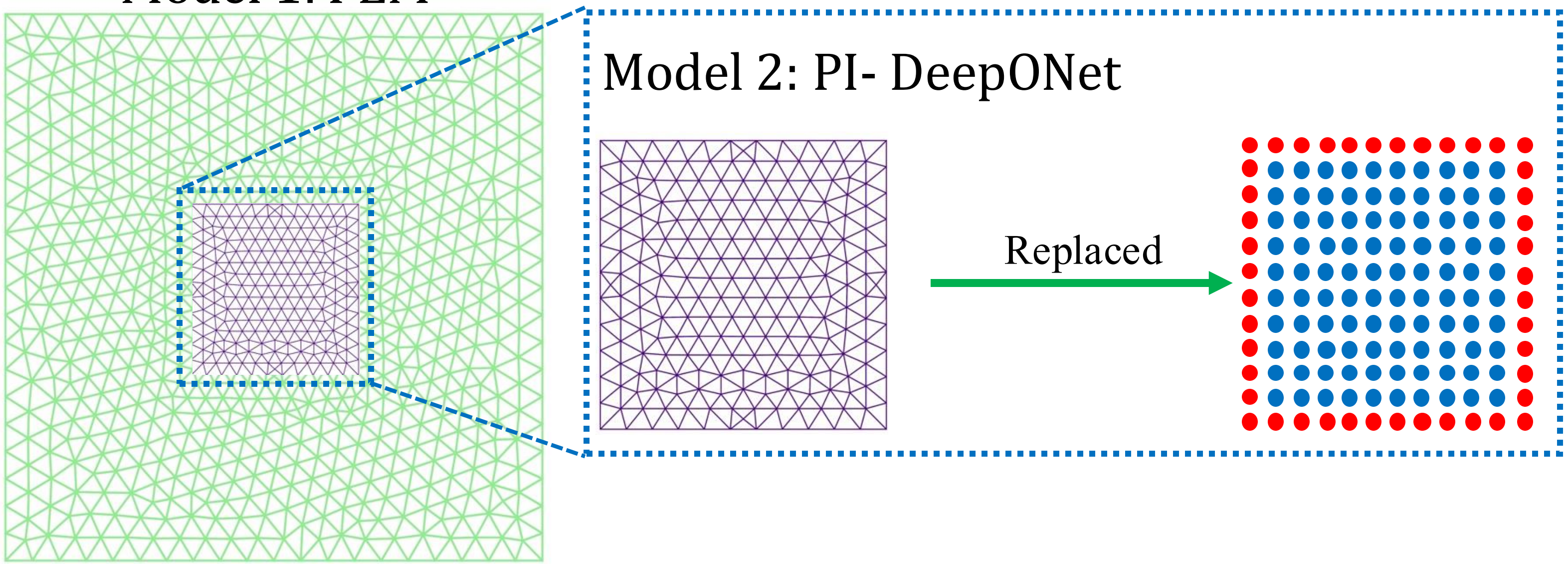
    Locations 'ok' with coarser discretization – approximated using numerical solvers.

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

3. For time dependent systems, the time marching employs a Newmark-Beta method, instead of the neural operators

Domain Decomposition Framework

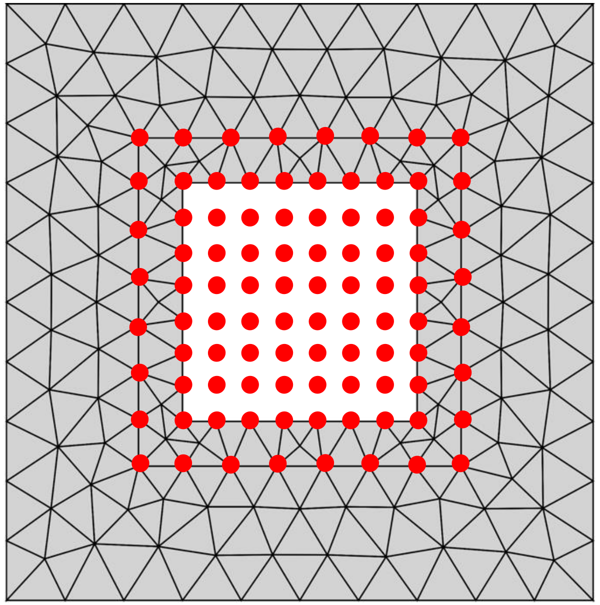
Model 1: FEM



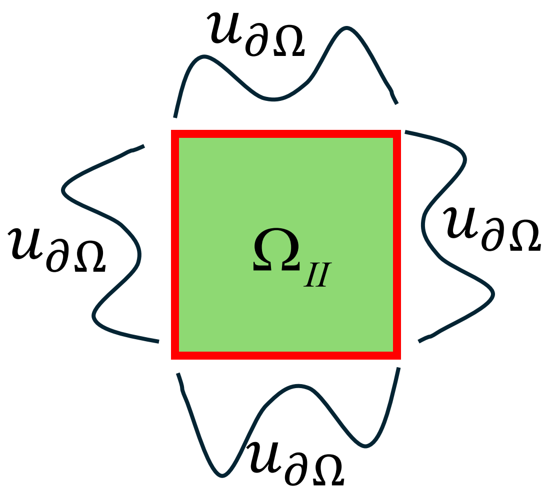
Can suffice with coarse mesh

Requires fine mesh

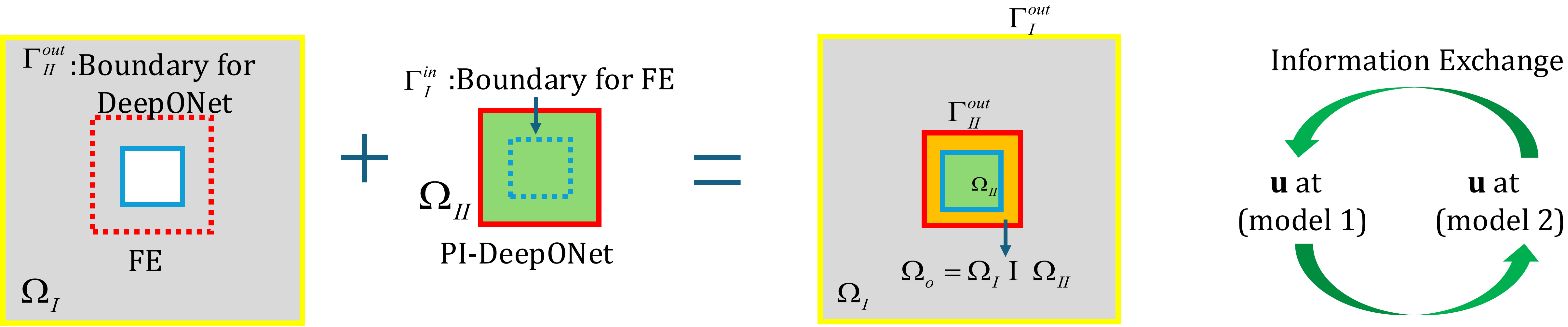
Overlapping Decomposed Domains



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

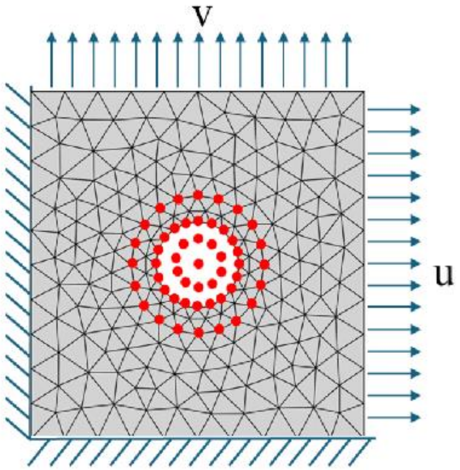
Accelerating Multiscale Modeling with Hybrid Solvers: Coupling FEM and Neural Operators with Domain Decomposition

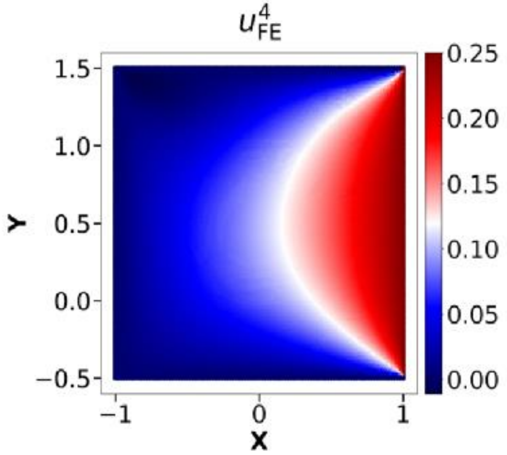
Wei Wang[a,b], Maryam Hakimzadeh[b], Haihui Ruan[a,c,*], Somdatta Goswami[b,*]

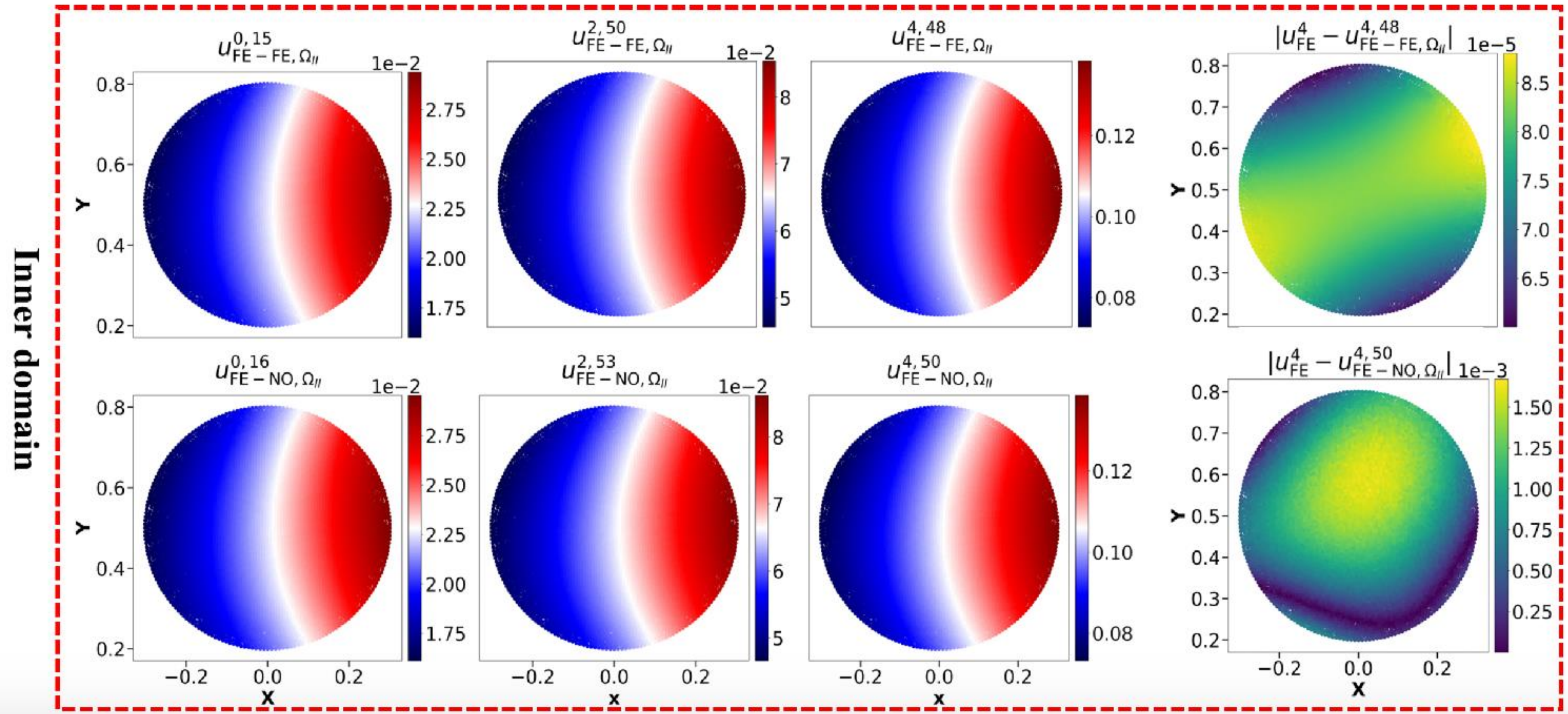
[a] *Department of Mechanical Engineering, The Hong Kong Polytechnic University*
[b] *Department of Civil and Systems Engineering, Johns Hopkins University*
[c] *PolyU-Daya Bay Technology and Innovation Research Institute*

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).

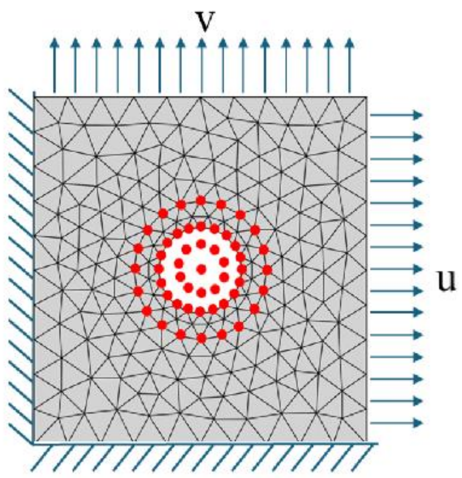
Hyper-elasticity under quasi-static loading conditions

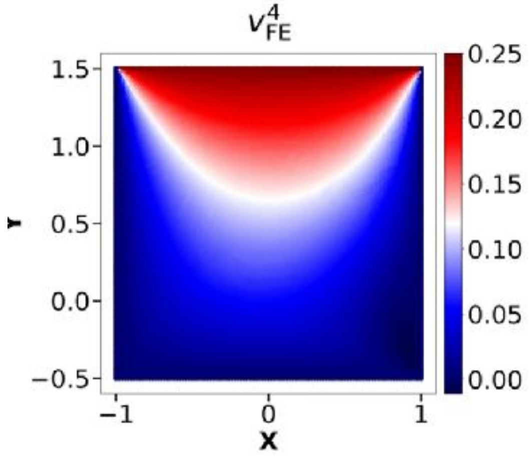


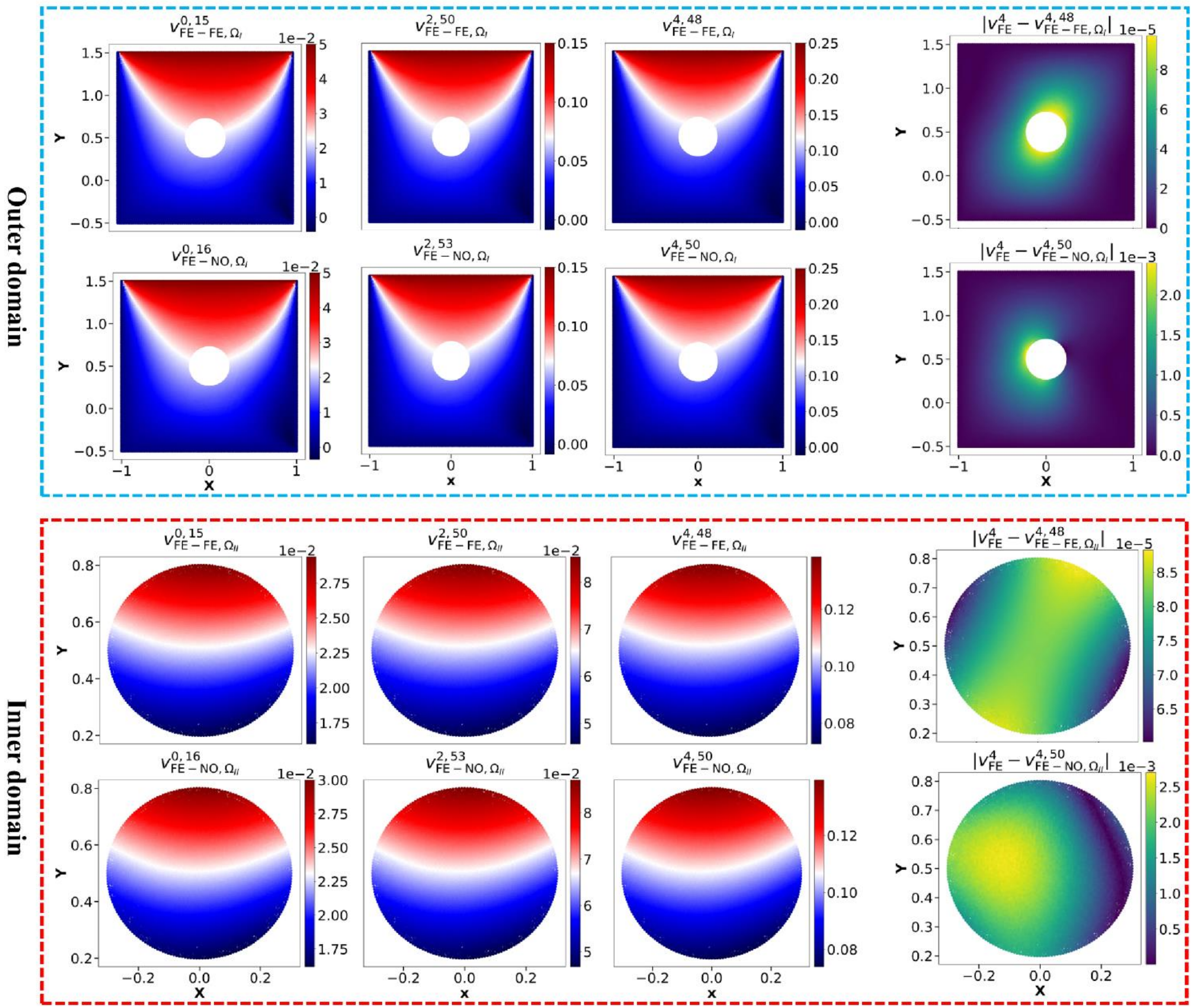




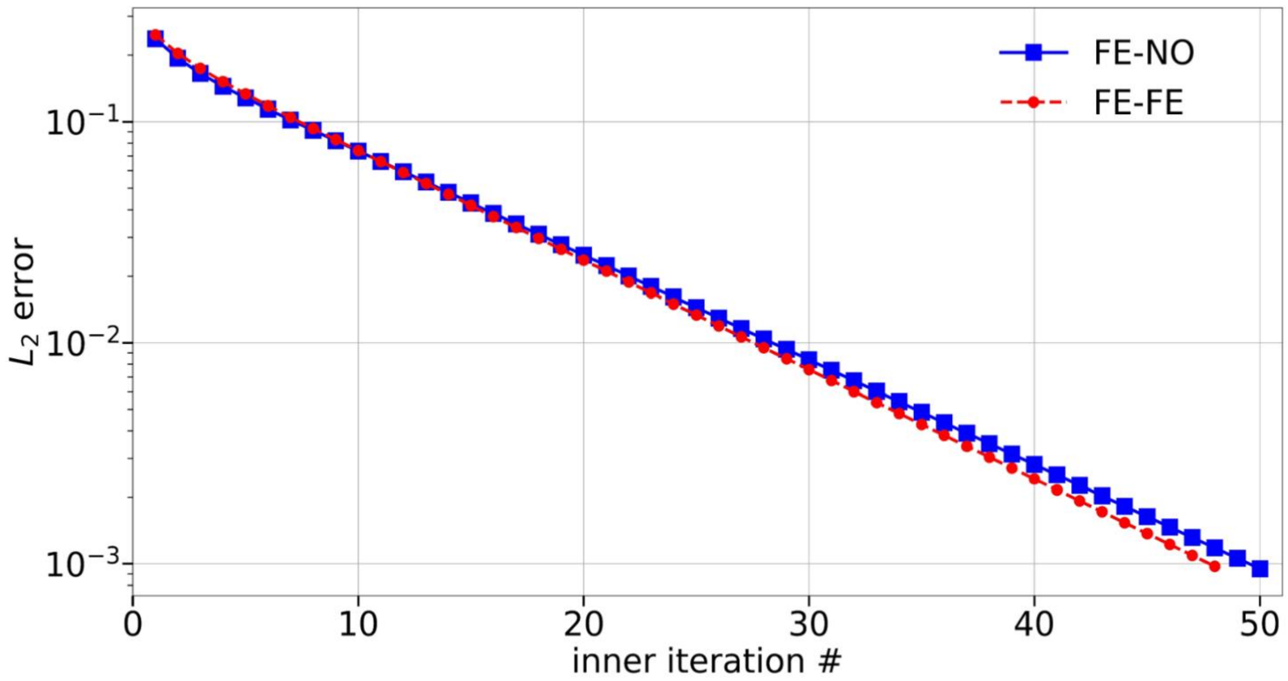
Hyper-elasticity under quasi-static loading conditions





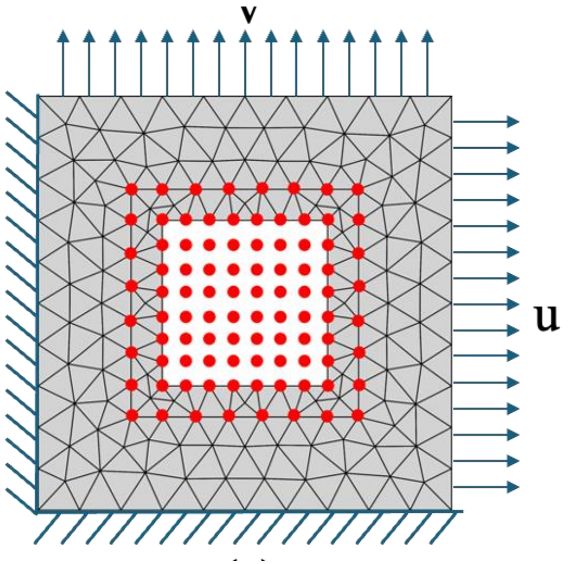


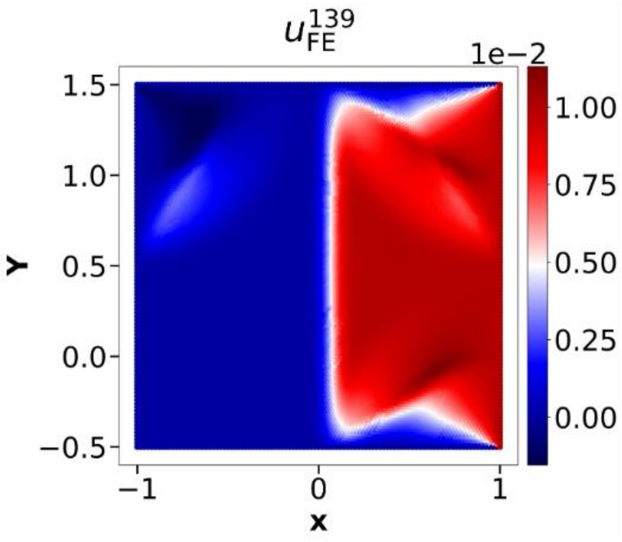
Performance of FE-NO

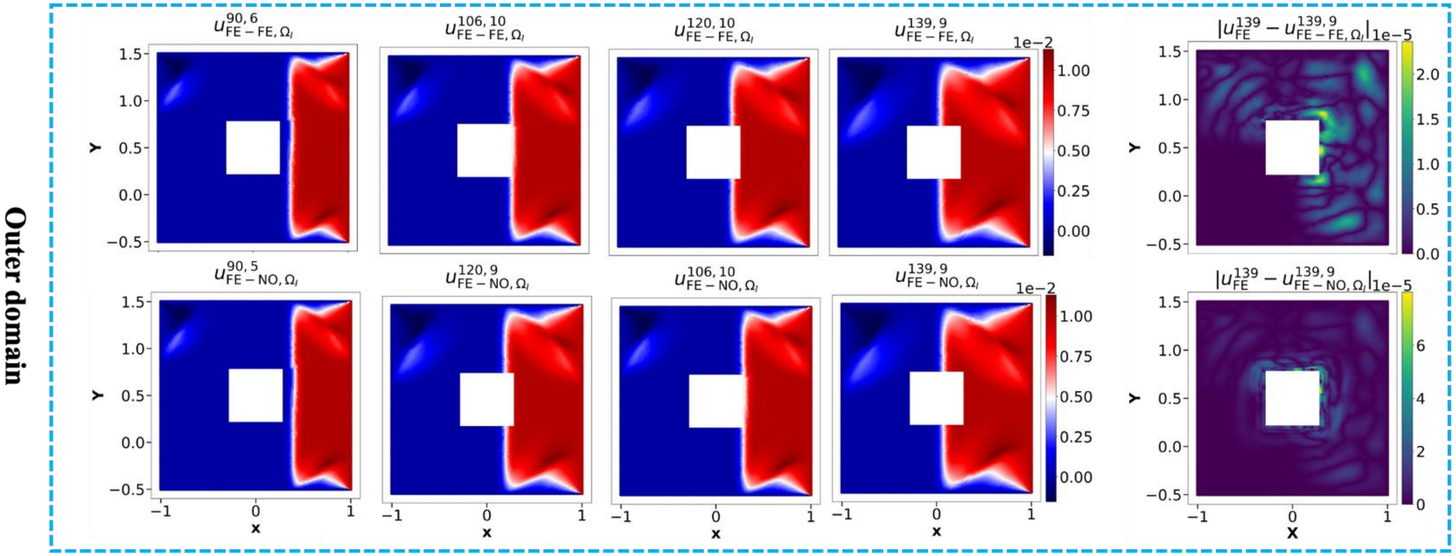


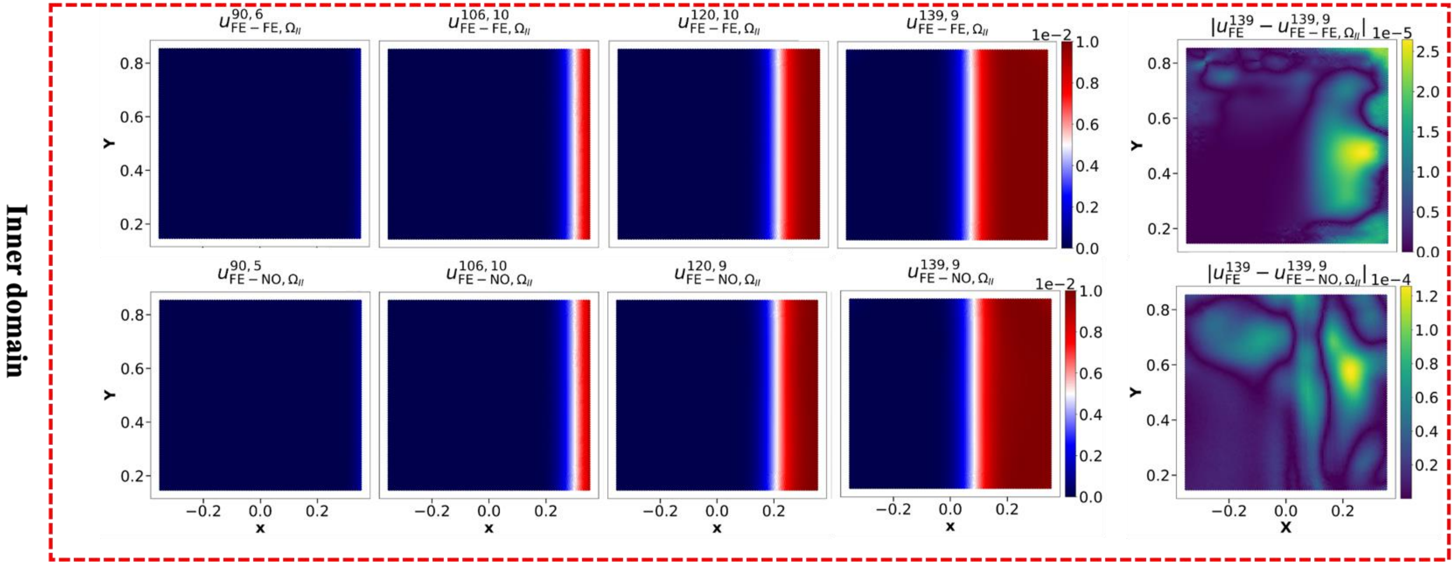
At t = 4, the neural operators (NO) coupling needs more inner iterations.
No need Newton's solver for additional root-finding iterations at each inner iterations.
FE-NO coupling is 20% faster than FE-FE coupling.

Linear Elastic Model in Dynamic Regime

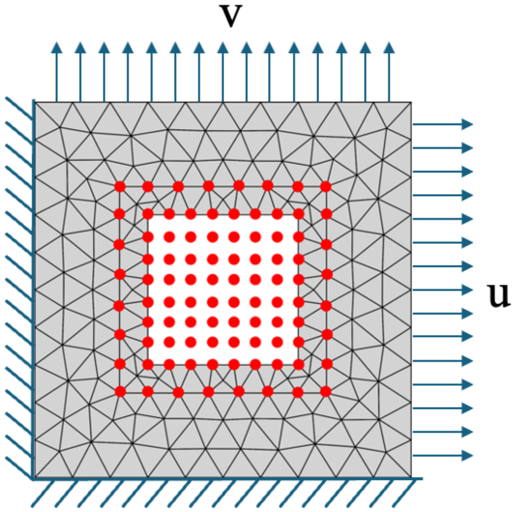


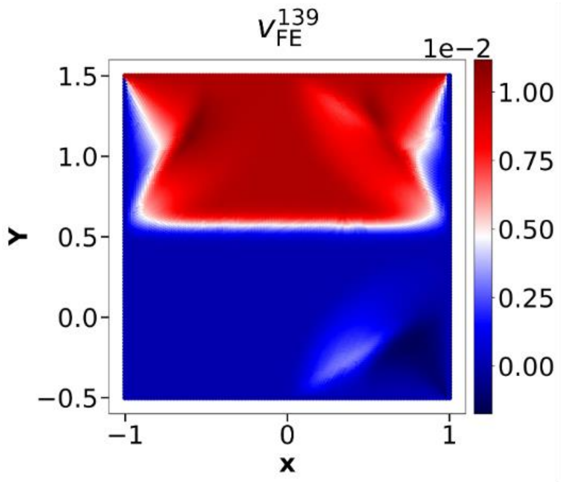


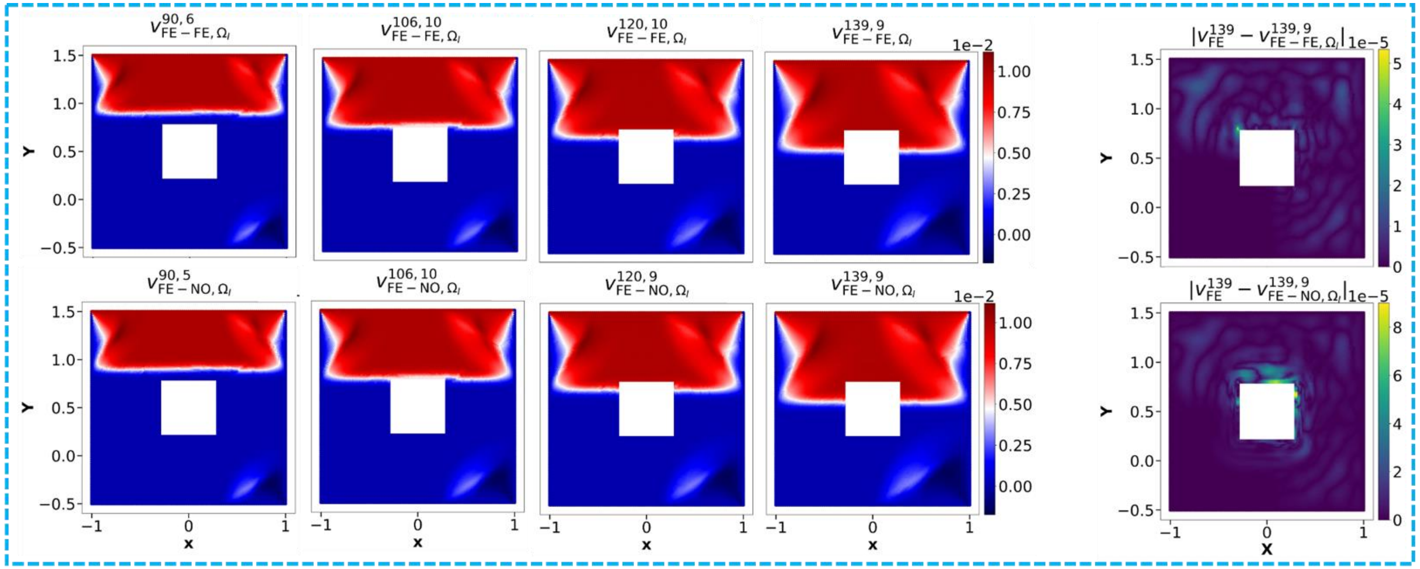


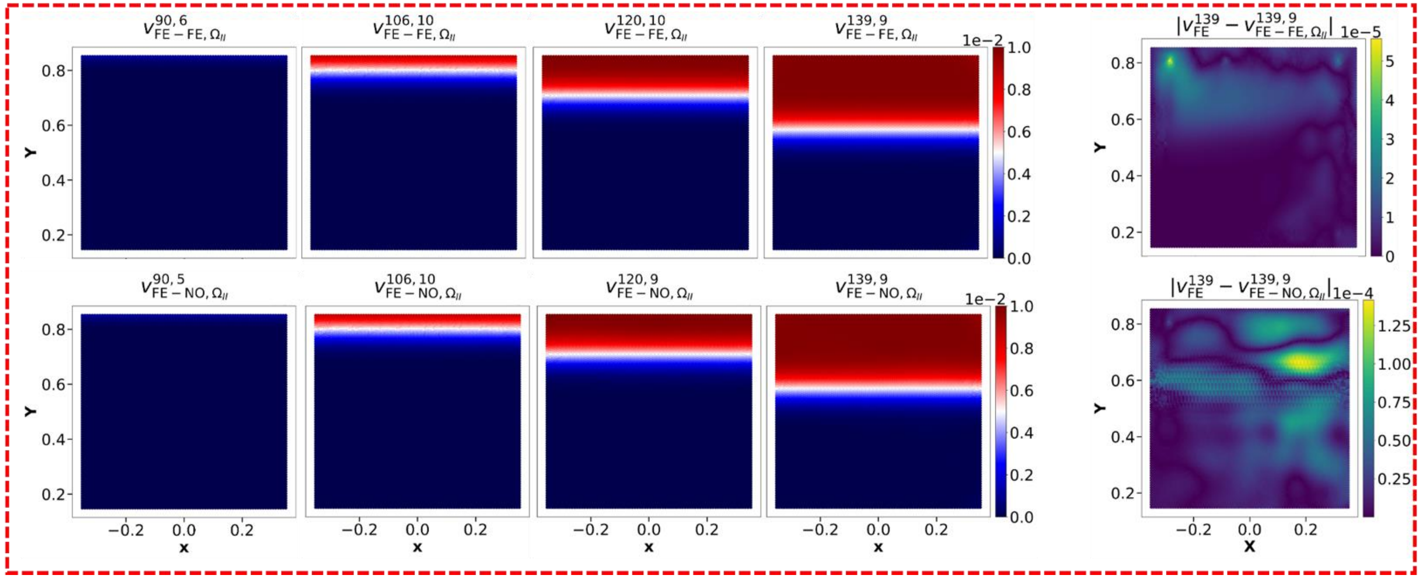


Linear Elastic Model in Dynamic Regime

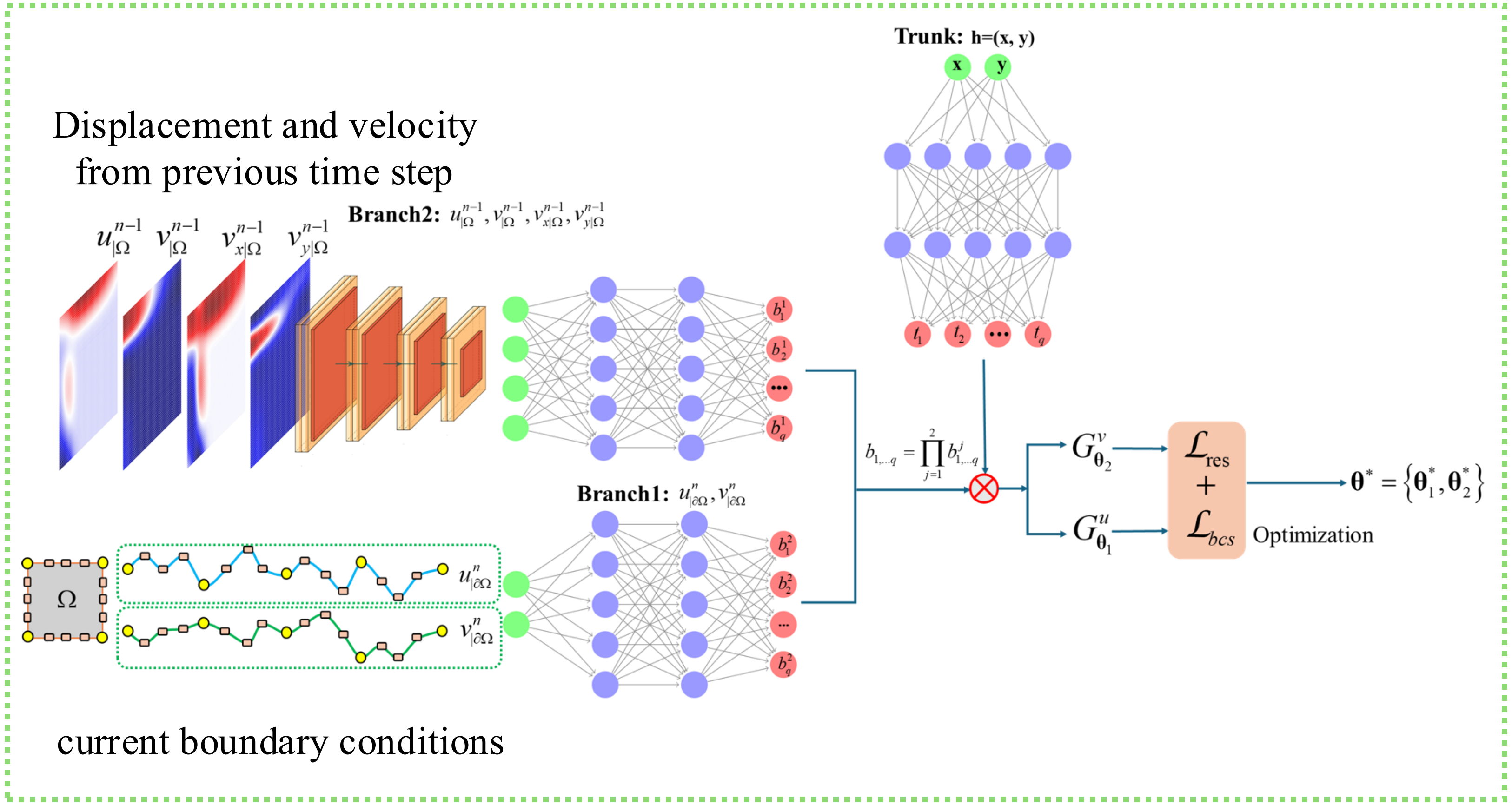








Temporal Dimension Coupling



Newmark-beta time discretization method:

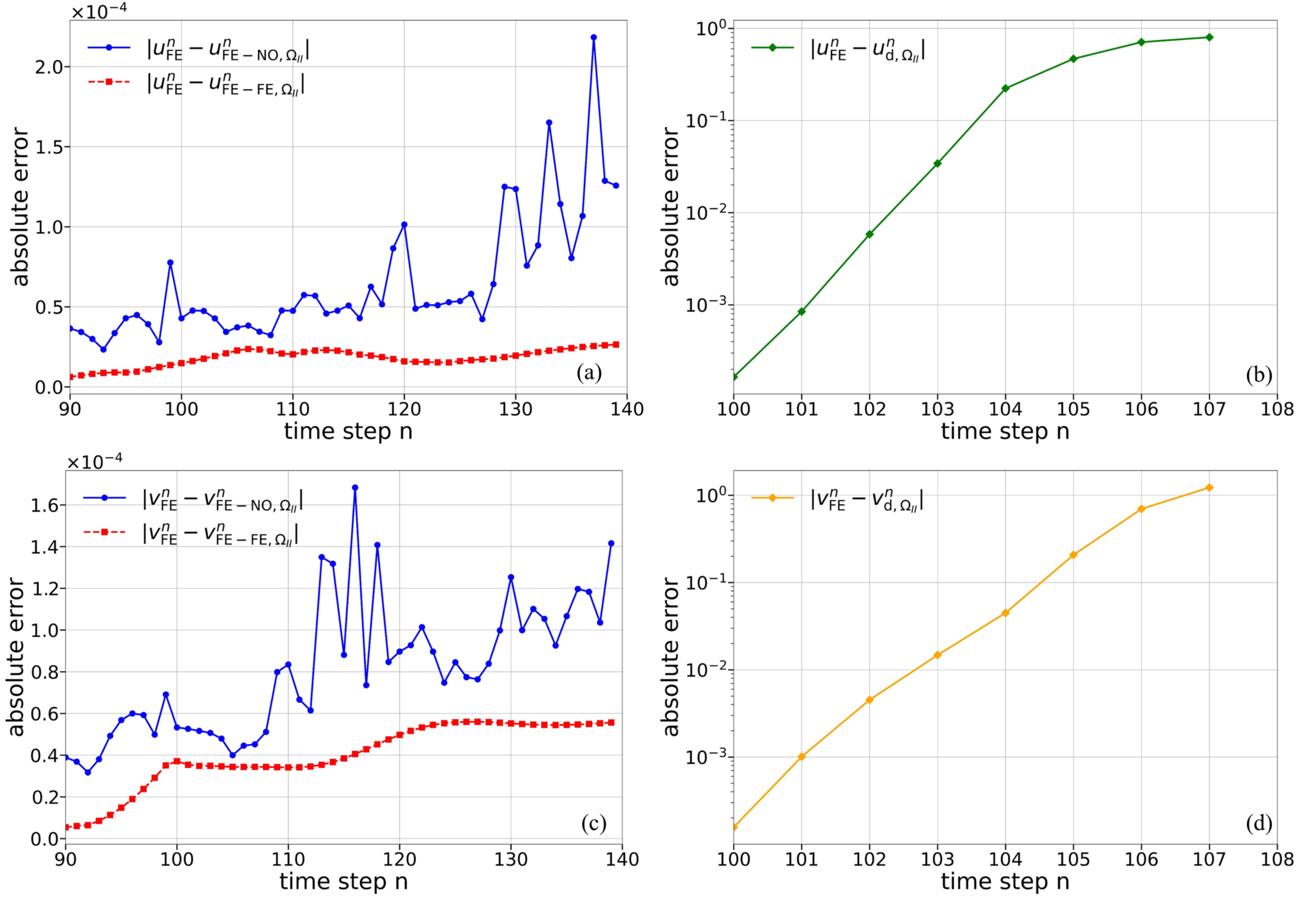
$$\dot{u}(t+dt) = \dot{u} + dt\left[(1-\gamma)\ddot{u} + \gamma\ddot{u}(t+dt)\right]$$

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

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

Centrum IntelliPhysics

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!