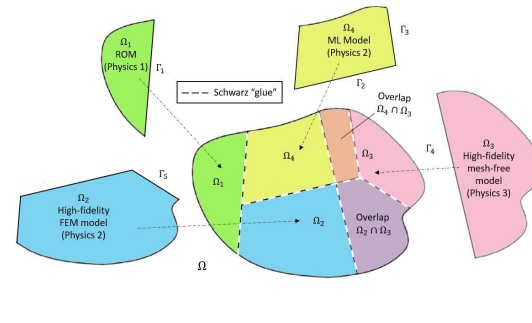
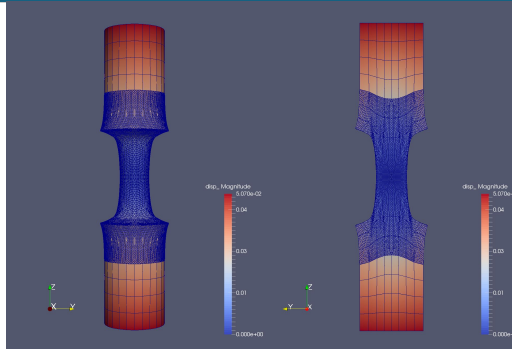
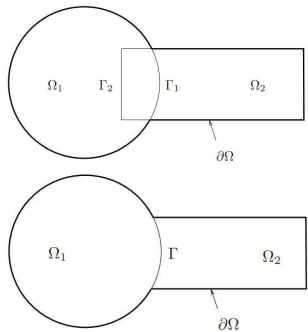


# Flexible domain decomposition-based couplings of conventional and data-driven models via the Schwarz alternating method



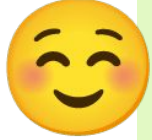
Irina Tezaur<sup>1</sup>, Chris Wentland<sup>1</sup>, Francesco Rizzi<sup>2</sup>, Joshua Barnett<sup>3</sup>,  
Alejandro Mota<sup>1</sup>

<sup>1</sup>Sandia National Laboratories, <sup>2</sup>NexGen Analytics, <sup>3</sup>Cadence Design Systems

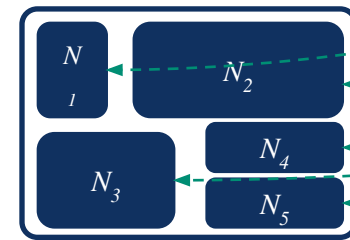
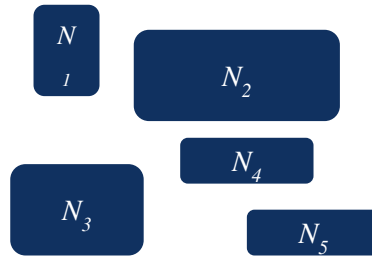
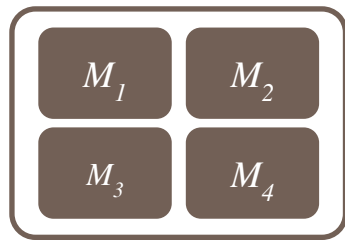
2<sup>nd</sup> AMS-UMI International Joint Meeting  
Palermo, Italy. July 23-26, 2024

SAND2024-08453C

# Motivation: multi-scale & multi-physics coupling



There exist established **rigorous mathematical theories** for **coupling** multi-scale and multi-physics components based on **traditional discretization methods** (“Full Order Models” or FOMs).



## Complex System Model

- PDEs, ODEs
- Nonlocal integral
- Classical DFT
- Atomistic, ...

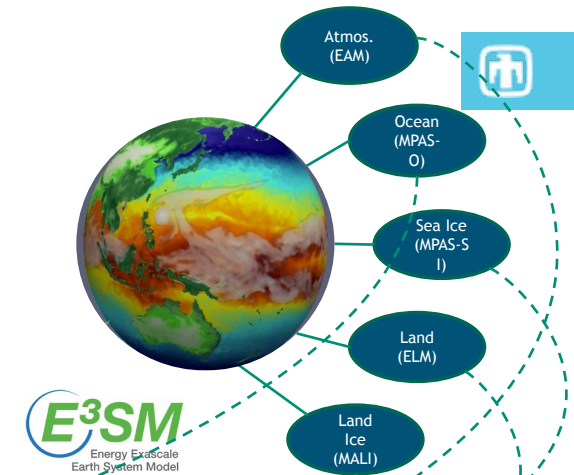
## Traditional Methods

- Mesh-based (FE, FV, FD)
- Meshless (SPH, MLS)
- Implicit, explicit
- Eulerian, Lagrangian...

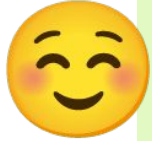
## Coupled Numerical Model

- Monolithic (Lagrange multipliers)
- Partitioned (loose) coupling
- Iterative (Schwarz, optimization)

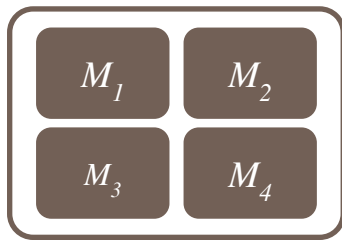
**E<sup>3</sup>SM**  
Energy Exascale  
Earth System Model



# Motivation: multi-scale & multi-physics coupling

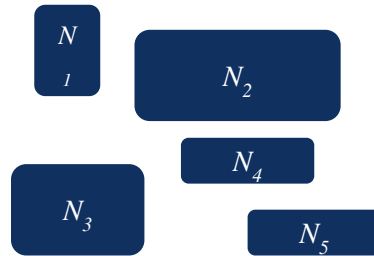


There exist established **rigorous mathematical theories** for **coupling** multi-scale and multi-physics components based on **traditional discretization methods** (“Full Order Models” or FOMs).



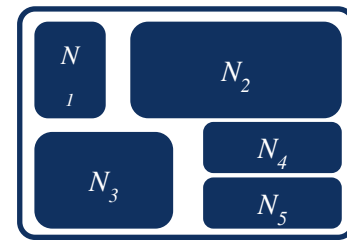
## Complex System Model

- PDEs, ODEs
- Nonlocal integral
- Classical DFT
- Atomistic, ...



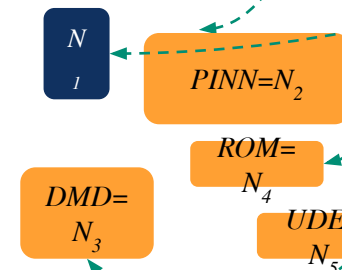
## Traditional Methods

- Mesh-based (FE, FV, FD)
- Meshless (SPH, MLS)
- Implicit, explicit
- Eulerian, Lagrangian, ...



## Coupled Numerical Model

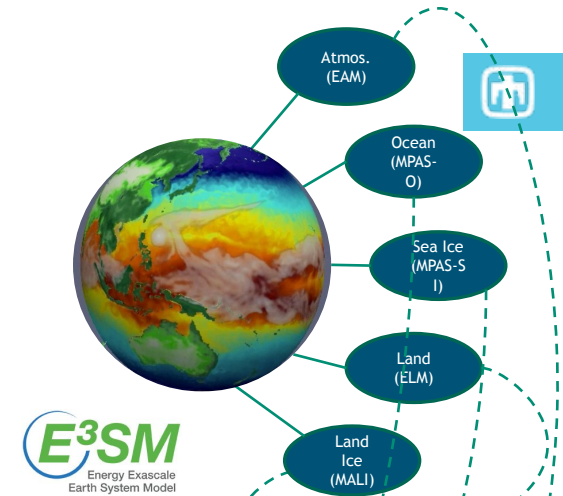
- Monolithic (Lagrange multipliers)
- Partitioned (loose) coupling
- Iterative (Schwarz, optimization)



## Traditional + Data-Driven Methods

- PINNs
- Neural ODEs
- Projection-based ROMs, ...

**E<sup>3</sup>SM**  
Energy Exascale  
Earth System Model



Unfortunately, existing algorithmic and software infrastructures are **ill-equipped** to handle plug-and-play integration of **non-traditional, data-driven models!**



## Principal research objective:

- Discover mathematical principles guiding the assembly of standard and data-driven numerical models in stable, accurate and physically consistent ways.

## Principal research goals:

- “Mix-and-match” standard and data-driven models from three-classes

□ Class A: projection-based reduced order models (ROMs) *This talk.*

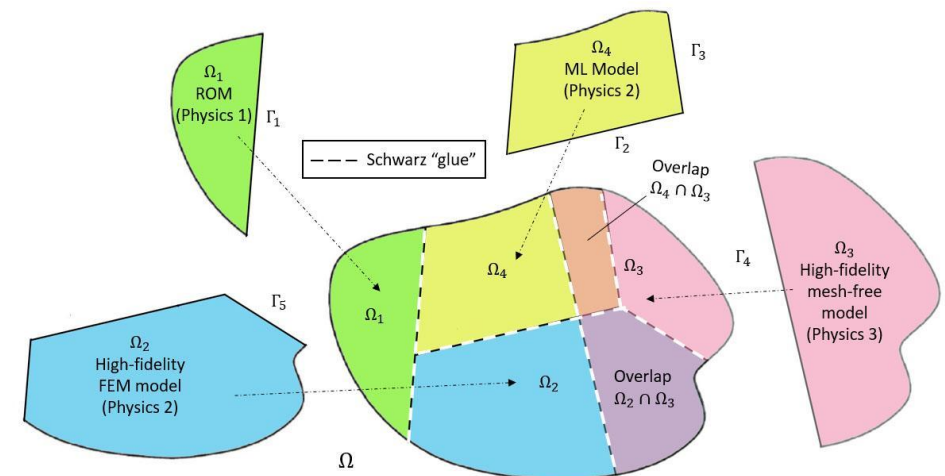
□ Class B: machine-learned models, i.e., Physics-Informed Neural Networks (PINNs)

□ Class C: flow map approximation models, i.e., dynamic model decomposition (DMD) models

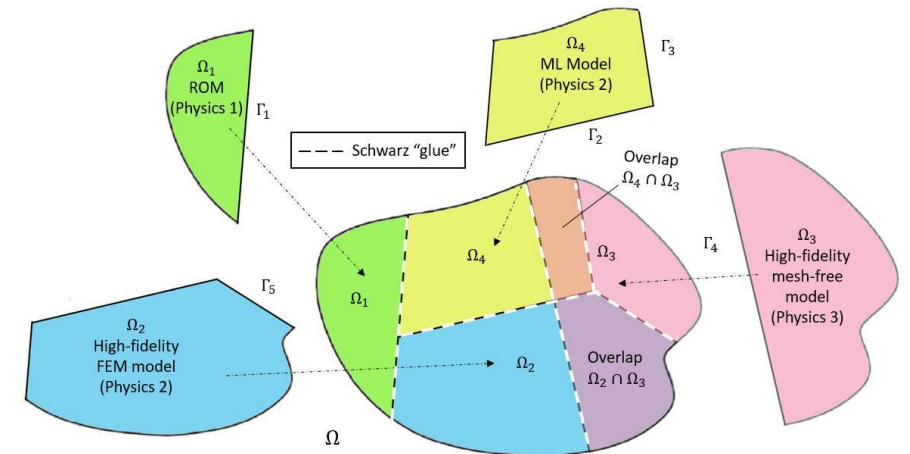
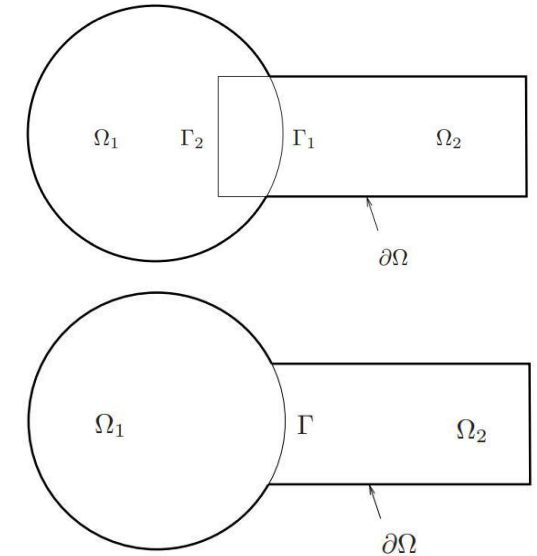
- Ensure well-posedness & physical consistency of resulting heterogeneous models.
- Solve such heterogeneous models efficiently.

## Three coupling methods:

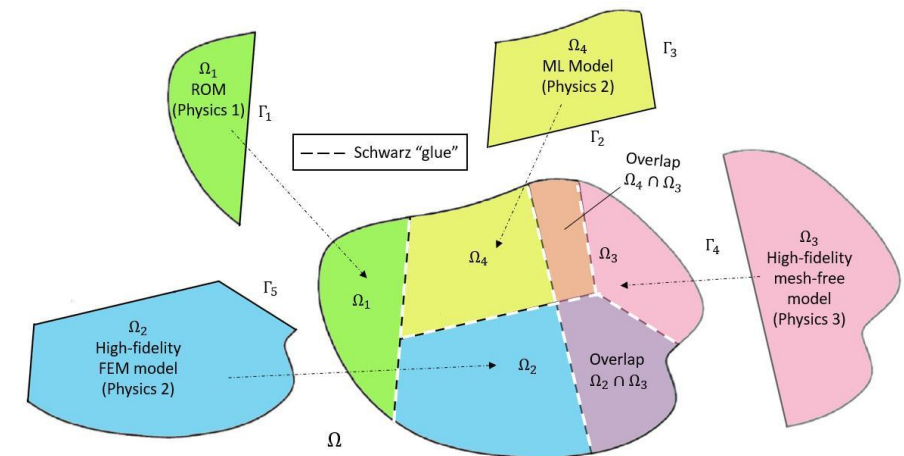
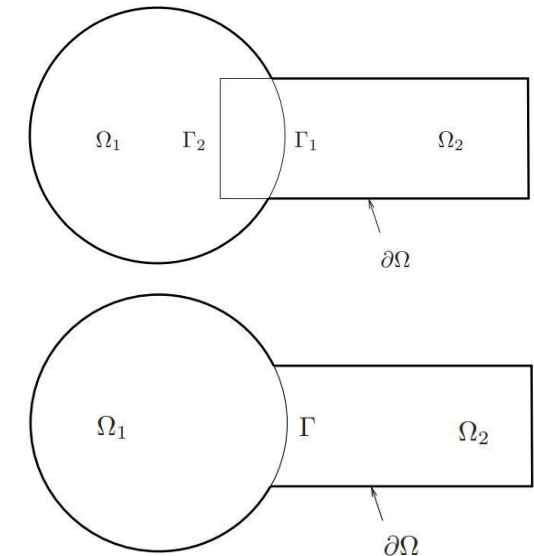
- Alternating Schwarz-based coupling *This talk.*
- Optimization-based coupling
- Coupling via generalized mortar methods



- The Schwarz Alternating Method for Domain Decomposition-Based Coupling
- Extension to FOM\*-ROM<sup>#</sup> and ROM-ROM Coupling
- Numerical Examples
  - 2D Burgers Equation
  - 2D Shallow Water Equations
  - Teaser: 2D Euler Equations Riemann Problem
- Summary & Future Work



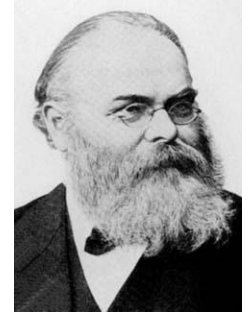
- The Schwarz Alternating Method for Domain Decomposition-Based Coupling
- Extension to FOM\*-ROM<sup>#</sup> and ROM-ROM Coupling
- Numerical Examples
  - 2D Burgers Equation
  - 2D Shallow Water Equations
  - Teaser: 2D Euler Equations Riemann Problem
- Summary & Future Work



# Schwarz Alternating Method for Domain Decomposition

- Proposed in 1870 by H. Schwarz for solving Laplace PDE on irregular domains.

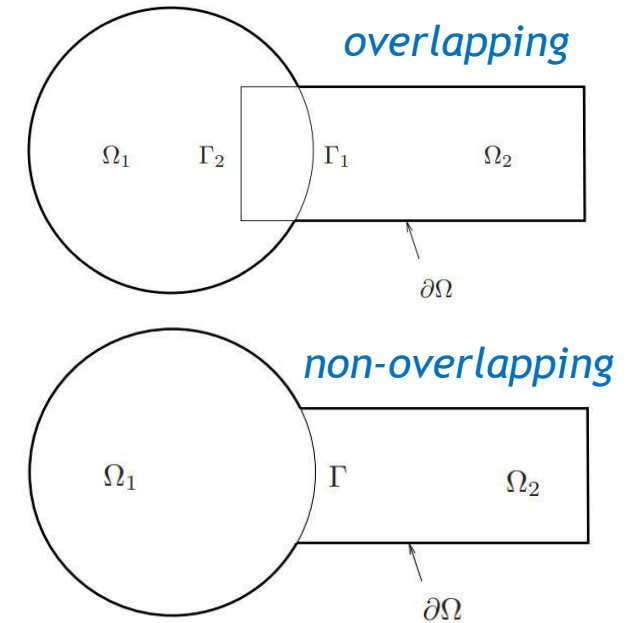
**Crux of Method:** if the solution is known in regularly shaped domains, use those as pieces to iteratively build a solution for the more complex domain.



H. Schwarz (1843-1921)



## Basic Schwarz Algorithm

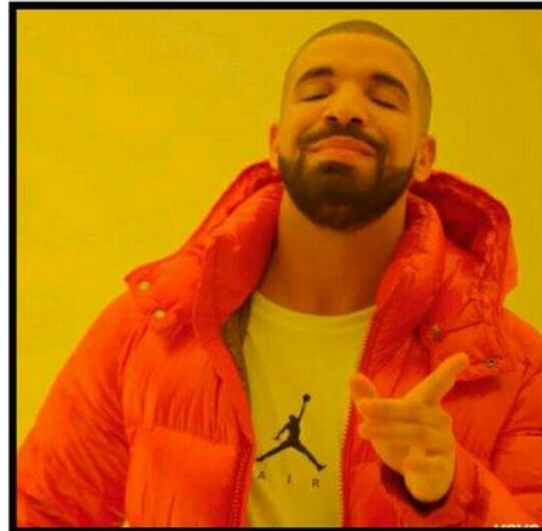


- Schwarz alternating method most commonly used as a ***preconditioner*** for Krylov iterative methods to solve linear algebraic equations.

**Idea behind this work:** using the Schwarz alternating method as a ***discretization method*** for solving multi-scale or multi-physics partial differential equations (PDEs).



AS A *PRECONDITIONER*  
FOR THE LINEARIZED  
SYSTEM



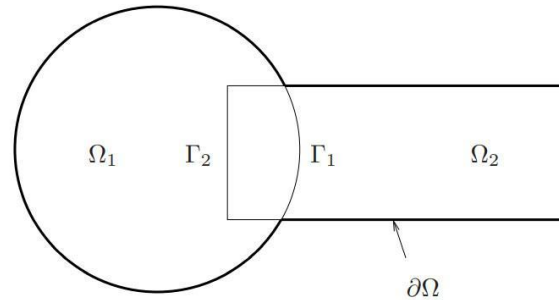
AS A *SOLVER* FOR THE  
COUPLED  
FULLY NONLINEAR  
PROBLEM



## Overlapping Domain Decomposition

$$\begin{cases} N(\mathbf{u}_1^{(n+1)}) = f, & \text{in } \Omega_1 \\ \mathbf{u}_1^{(n+1)} = \mathbf{g}, & \text{on } \partial\Omega_1 \setminus \Gamma_1 \\ \mathbf{u}_1^{(n+1)} = \mathbf{u}_2^{(n)} & \text{on } \Gamma_1 \end{cases}$$

$$\begin{cases} N(\mathbf{u}_2^{(n+1)}) = f, & \text{in } \Omega_2 \\ \mathbf{u}_2^{(n+1)} = \mathbf{g}, & \text{on } \partial\Omega_2 \setminus \Gamma_2 \\ \mathbf{u}_2^{(n+1)} = \mathbf{u}_1^{(n+1)} & \text{on } \Gamma_2 \end{cases}$$

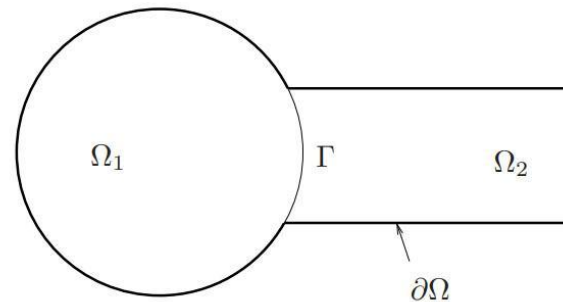


*Model PDE:*

- Dirichlet-Dirichlet transmission BCs [Schwarz 1870; Lions 1988; Mota *et al.* 2017; Mota *et al.* 2022]

## Non-overlapping Domain Decomposition

$$\begin{cases} N(\mathbf{u}_2^{(n+1)}) = f, & \text{in } \Omega_2 \\ \mathbf{u}_2^{(n+1)} = \mathbf{g}, & \text{on } \partial\Omega_2 \setminus \Gamma \\ \nabla \mathbf{u}_2^{(n+1)} \cdot \mathbf{n} = \nabla \mathbf{u}_1^{(n+1)} \cdot \mathbf{n}, & \text{on } \Gamma \end{cases}$$



### Multiplicative Overlapping Schwarz

$$\begin{cases} N(\mathbf{u}_1^{(n+1)}) = f, & \text{in } \Omega_1 \\ \mathbf{u}_1^{(n+1)} = \mathbf{g}, & \text{on } \partial\Omega_1 \setminus \Gamma_1 \\ \mathbf{u}_1^{(n+1)} = \mathbf{u}_2^{(n)} & \text{on } \Gamma_1 \end{cases}$$

$$\begin{cases} N(\mathbf{u}_2^{(n+1)}) = f, & \text{in } \Omega_2 \\ \mathbf{u}_2^{(n+1)} = \mathbf{g}, & \text{on } \partial\Omega_2 \setminus \Gamma_2 \\ \mathbf{u}_2^{(n+1)} = \mathbf{u}_1^{(n+1)} & \text{on } \Gamma_2 \end{cases}$$

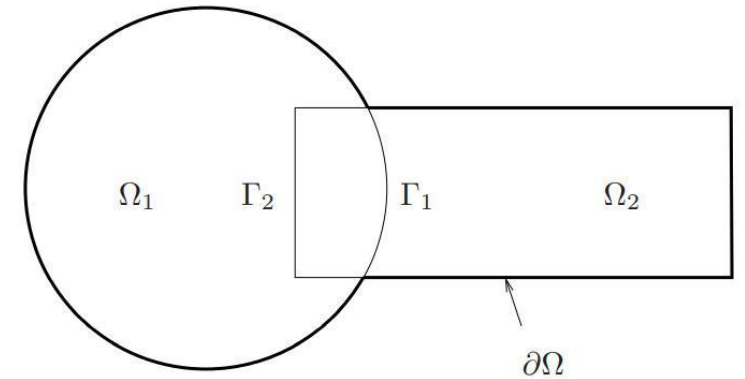
### Additive Overlapping Schwarz

$$\begin{cases} N(\mathbf{u}_1^{(n+1)}) = f, & \text{in } \Omega_1 \\ \mathbf{u}_1^{(n+1)} = \mathbf{g}, & \text{on } \partial\Omega_1 \setminus \Gamma_1 \\ \mathbf{u}_1^{(n+1)} = \mathbf{u}_2^{(n)} & \text{on } \Gamma_1 \end{cases}$$

$$\begin{cases} N(\mathbf{u}_2^{(n+1)}) = f, & \text{in } \Omega_2 \\ \mathbf{u}_2^{(n+1)} = \mathbf{g}, & \text{on } \partial\Omega_2 \setminus \Gamma_2 \\ \mathbf{u}_2^{(n+1)} = \mathbf{u}_1^{(n)} & \text{on } \Gamma_2 \end{cases}$$

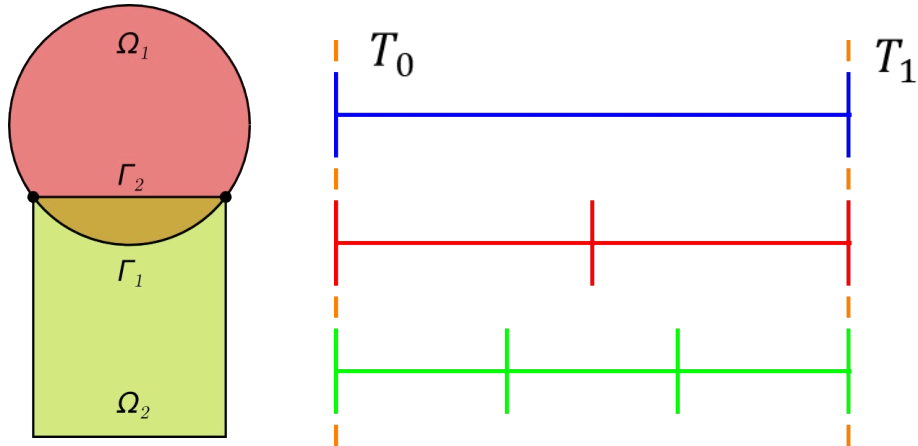
*Model PDE:*

$$\begin{cases} N(\mathbf{u}) = f, & \text{in } \Omega \\ \mathbf{u} = \mathbf{g}, & \text{on } \partial\Omega \end{cases}$$



- **Multiplicative Schwarz:** solves subdomain problems **sequentially** (in serial)
- **Additive Schwarz:** advance subdomains in **parallel**, communicate boundary condition data later
  - Typically requires a few more **Schwarz iterations**, but does not degrade **accuracy**
  - **Parallelism** helps balance additional **cost** due to Schwarz iterations
  - Applicable to both **overlapping** and **non-overlapping** Schwarz

# Time-Advancement Within the Schwarz Framework



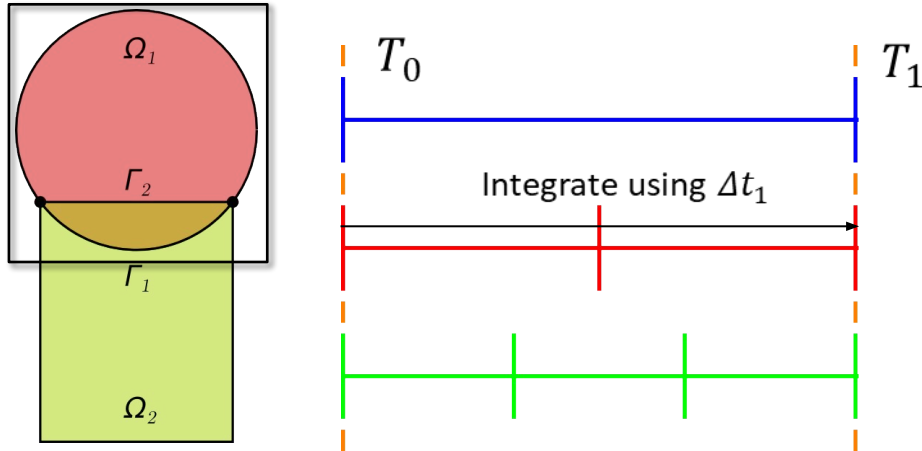
Controller time stepper

Time integrator for  $\Omega_1$

Time integrator for  $\Omega_2$

**Step 0:** Initialize  $i = 0$  (controller time index).

$$\text{Model PDE: } \begin{cases} \dot{\mathbf{u}} + N(\mathbf{u}) = \mathbf{f}, & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{g}(t), & \text{on } \partial\Omega \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, & \text{in } \Omega \end{cases}$$



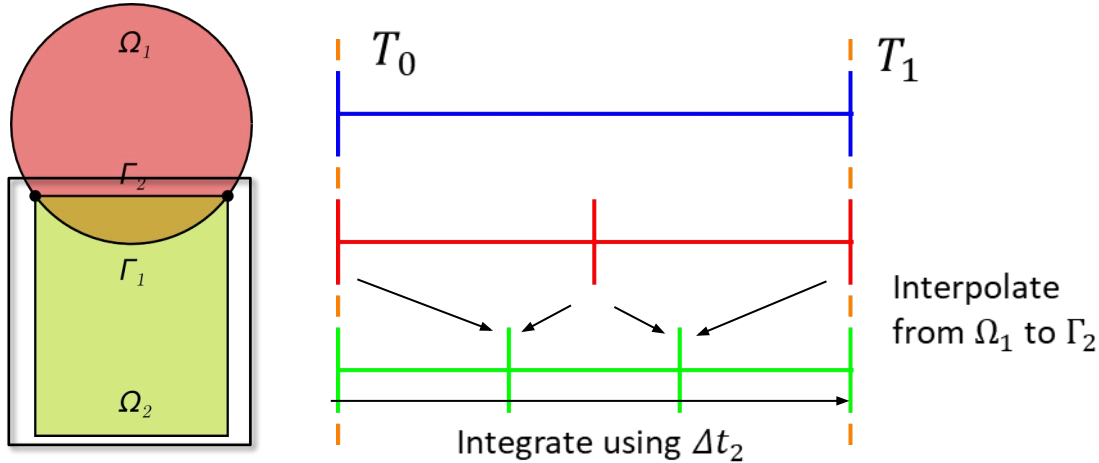
Controller time stepper

Time integrator for  $\Omega_1$ Time integrator for  $\Omega_2$ 

**Step 0:** Initialize  $i = 0$  (controller time index).

**Step 1:** Advance  $\Omega_1$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_1$  with time-step  $\Delta t_1$ , using solution in  $\Omega_2$  interpolated to  $\Gamma_1$  at times  $T_i + n\Delta t_1$ .

$$\text{Model PDE: } \begin{cases} \dot{\mathbf{u}} + N(\mathbf{u}) = \mathbf{f}, & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{g}(t), & \text{on } \partial\Omega \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, & \text{in } \Omega \end{cases}$$



Controller time stepper

Time integrator for  $\Omega_1$

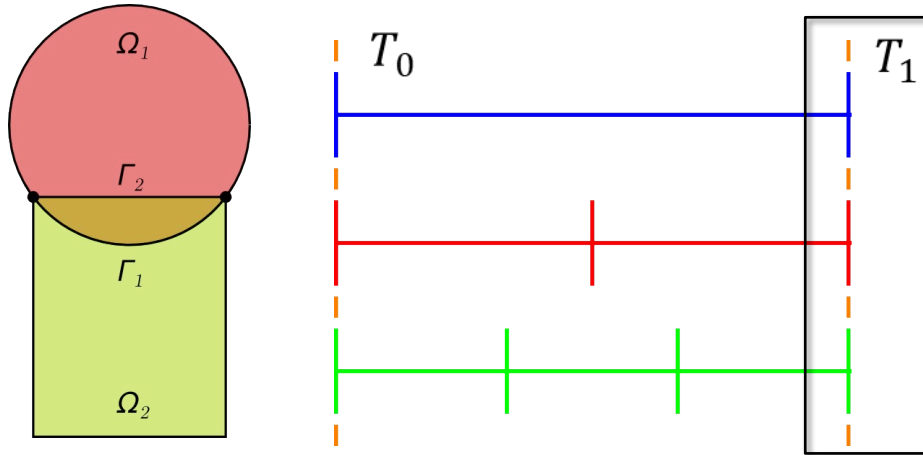
Time integrator for  $\Omega_2$

**Step 0:** Initialize  $i = 0$  (controller time index).

**Step 1:** Advance  $\Omega_1$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_1$  with time-step  $\Delta t_1$ , using solution in  $\Omega_2$  interpolated to  $\Gamma_1$  at times  $T_i + n\Delta t_1$ .

**Step 2:** Advance  $\Omega_2$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_2$  with time-step  $\Delta t_2$ , using solution in  $\Omega_1$  interpolated to  $\Gamma_2$  at times  $T_i + n\Delta t_2$ .

$$\text{Model PDE: } \begin{cases} \dot{\mathbf{u}} + N(\mathbf{u}) = \mathbf{f}, & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{g}(t), & \text{on } \partial\Omega \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, & \text{in } \Omega \end{cases}$$



Controller time stepper

Time integrator for  $\Omega_1$ Time integrator for  $\Omega_2$ 

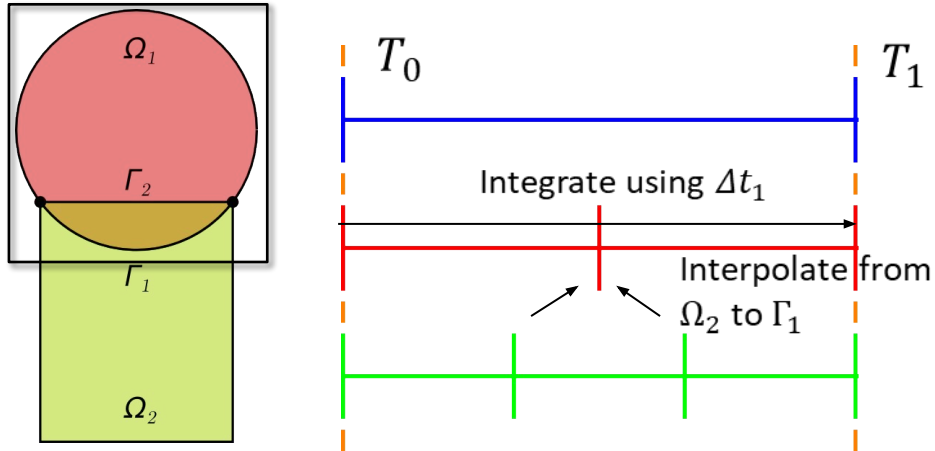
**Step 0:** Initialize  $i = 0$  (controller time index).

**Step 1:** Advance  $\Omega_1$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_1$  with time-step  $\Delta t_1$ , using solution in  $\Omega_2$  interpolated to  $\Gamma_1$  at times  $T_i + n\Delta t_1$ .

**Step 2:** Advance  $\Omega_2$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_2$  with time-step  $\Delta t_2$ , using solution in  $\Omega_1$  interpolated to  $\Gamma_2$  at times  $T_i + n\Delta t_2$ .

**Step 3:** Check for convergence at time  $T_{i+1}$ .

$$\text{Model PDE: } \begin{cases} \dot{\mathbf{u}} + N(\mathbf{u}) = \mathbf{f}, & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{g}(t), & \text{on } \partial\Omega \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, & \text{in } \Omega \end{cases}$$



Controller time stepper

Time integrator for  $\Omega_1$

Time integrator for  $\Omega_2$

**Step 0:** Initialize  $i = 0$  (controller time index).

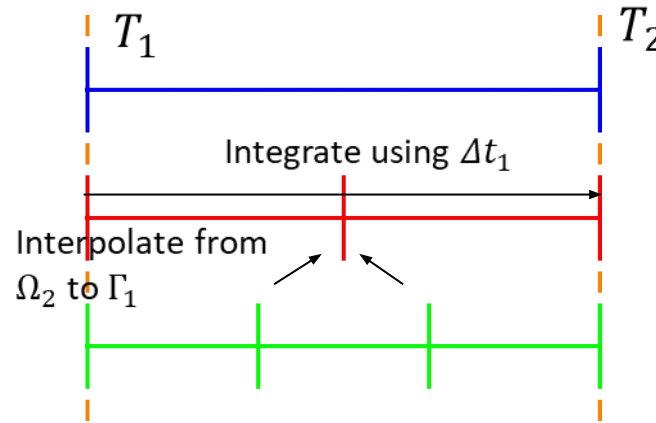
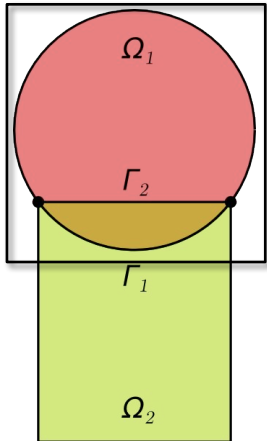
**Step 1:** Advance  $\Omega_1$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_1$  with time-step  $\Delta t_1$ , using solution in  $\Omega_2$  interpolated to  $\Gamma_1$  at times  $T_i + n\Delta t_1$ .

**Step 2:** Advance  $\Omega_2$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_2$  with time-step  $\Delta t_2$ , using solution in  $\Omega_1$  interpolated to  $\Gamma_2$  at times  $T_i + n\Delta t_2$ .

**Step 3:** Check for convergence at time  $T_{i+1}$ .

➤ If unconverged, return to Step 1.

$$\text{Model PDE: } \begin{cases} \dot{\mathbf{u}} + N(\mathbf{u}) = \mathbf{f}, & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{g}(t), & \text{on } \partial\Omega \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, & \text{in } \Omega \end{cases}$$



Controller time stepper

Time integrator for  $\Omega_1$

Time integrator for  $\Omega_2$

Can use *different integrators* with *different time steps* within each domain!

**Step 0:** Initialize  $i = 0$  (controller time index).

**Step 1:** Advance  $\Omega_1$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_1$  with time-step  $\Delta t_1$ , using solution in  $\Omega_2$  interpolated to  $\Gamma_1$  at times  $T_i + n\Delta t_1$ .

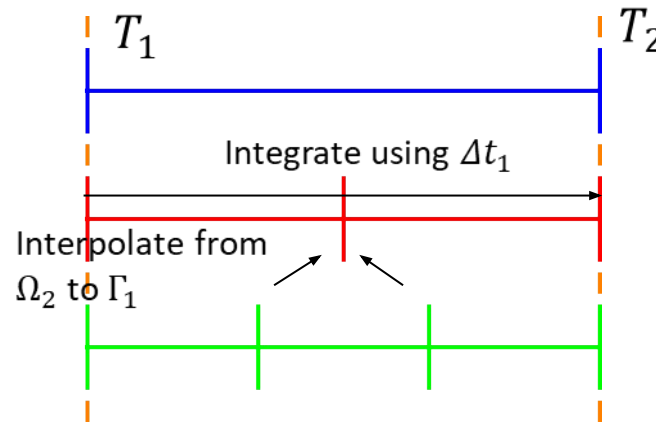
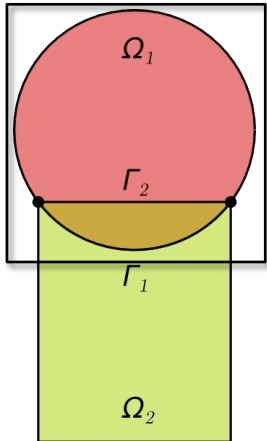
**Step 2:** Advance  $\Omega_2$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_2$  with time-step  $\Delta t_2$ , using solution in  $\Omega_1$  interpolated to  $\Gamma_2$  at times  $T_i + n\Delta t_2$ .

**Step 3:** Check for convergence at time  $T_{i+1}$ .

- If unconverged, return to Step 1.
- If converged, set  $i = i + 1$  and return to Step 1.

$$\text{Model PDE: } \begin{cases} \dot{\mathbf{u}} + N(\mathbf{u}) = \mathbf{f}, & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{g}(t), & \text{on } \partial\Omega \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, & \text{in } \Omega \end{cases}$$





Controller time stepper

Time integrator for  $\Omega_1$ Time integrator for  $\Omega_2$ 

Time-stepping procedure is **equivalent** to doing Schwarz on **space-time domain** [Mota *et al.* 2022].

**Step 0:** Initialize  $i = 0$  (controller time index).

**Step 1:** Advance  $\Omega_1$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_1$  with time-step  $\Delta t_1$ , using solution in  $\Omega_2$  interpolated to  $\Gamma_1$  at times  $T_i + n\Delta t_1$ .

**Step 2:** Advance  $\Omega_2$  solution from time  $T_i$  to time  $T_{i+1}$  using time-stepper in  $\Omega_2$  with time-step  $\Delta t_2$ , using solution in  $\Omega_1$  interpolated to  $\Gamma_2$  at times  $T_i + n\Delta t_2$ .

**Step 3:** Check for convergence at time  $T_{i+1}$ .

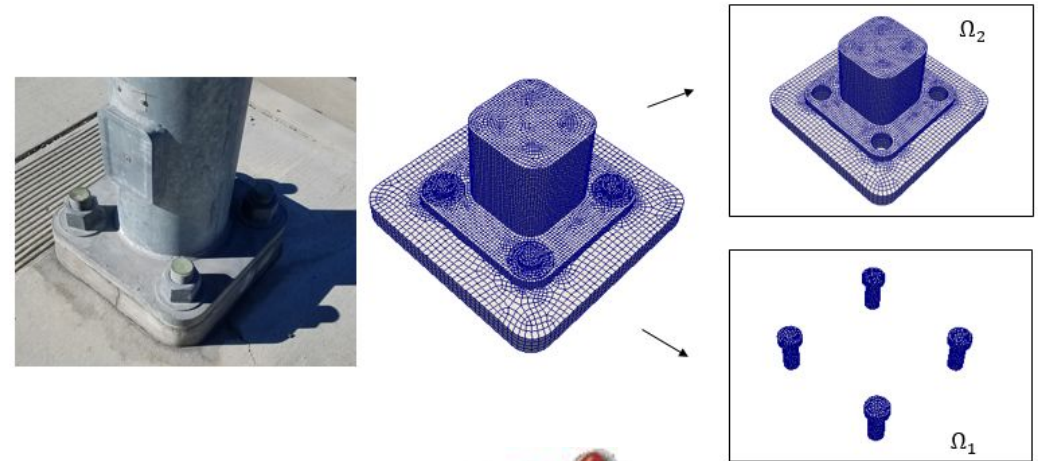
- If unconverged, return to Step 1.
- If converged, set  $i = i + 1$  and return to Step 1.

$$\text{Model PDE: } \begin{cases} \dot{\mathbf{u}} + N(\mathbf{u}) = \mathbf{f}, & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{g}(t), & \text{on } \partial\Omega \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, & \text{in } \Omega \end{cases}$$

*Model Solid Mechanics PDEs:*

$$\text{Quasistatic: } \text{Div } \mathbf{P} + \rho_0 \mathbf{B} = \mathbf{0} \quad \text{in } \Omega$$

$$\text{Dynamic: } \text{Div } \mathbf{P} + \rho_0 \mathbf{B} = \rho_0 \ddot{\boldsymbol{\varphi}} \quad \text{in } \Omega \times I$$



- “Plug-and-play” framework:

- Ability to couple regions with *different non-conformal meshes*, *different element types* and *different levels of refinement* to simplify task of *meshing complex geometries*.
- Ability to use *different solvers/time-integrators* in different regions.

<sup>1</sup> Mota *et al.* 2017; Mota *et al.* 2022. <sup>2</sup> <https://github.com/sandialabs/LCM>.

# Schwarz for Multiscale FOM-FOM Coupling in Solid Mechanics<sup>1</sup>

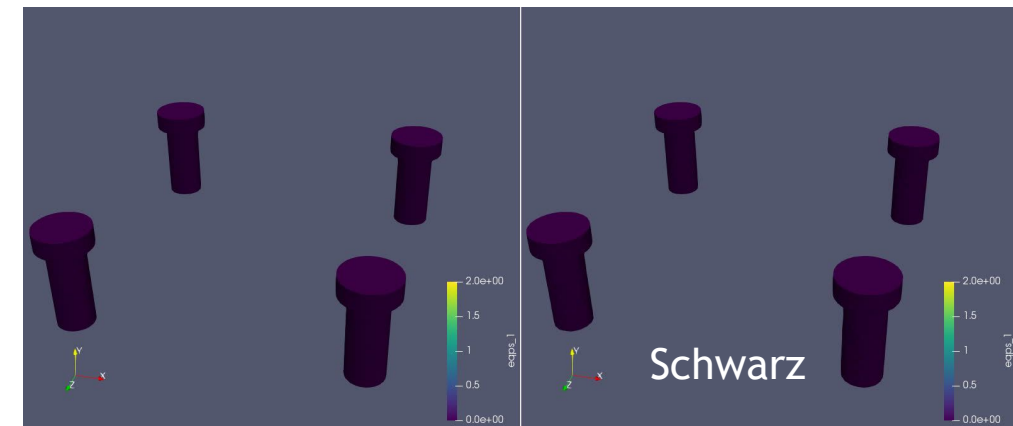
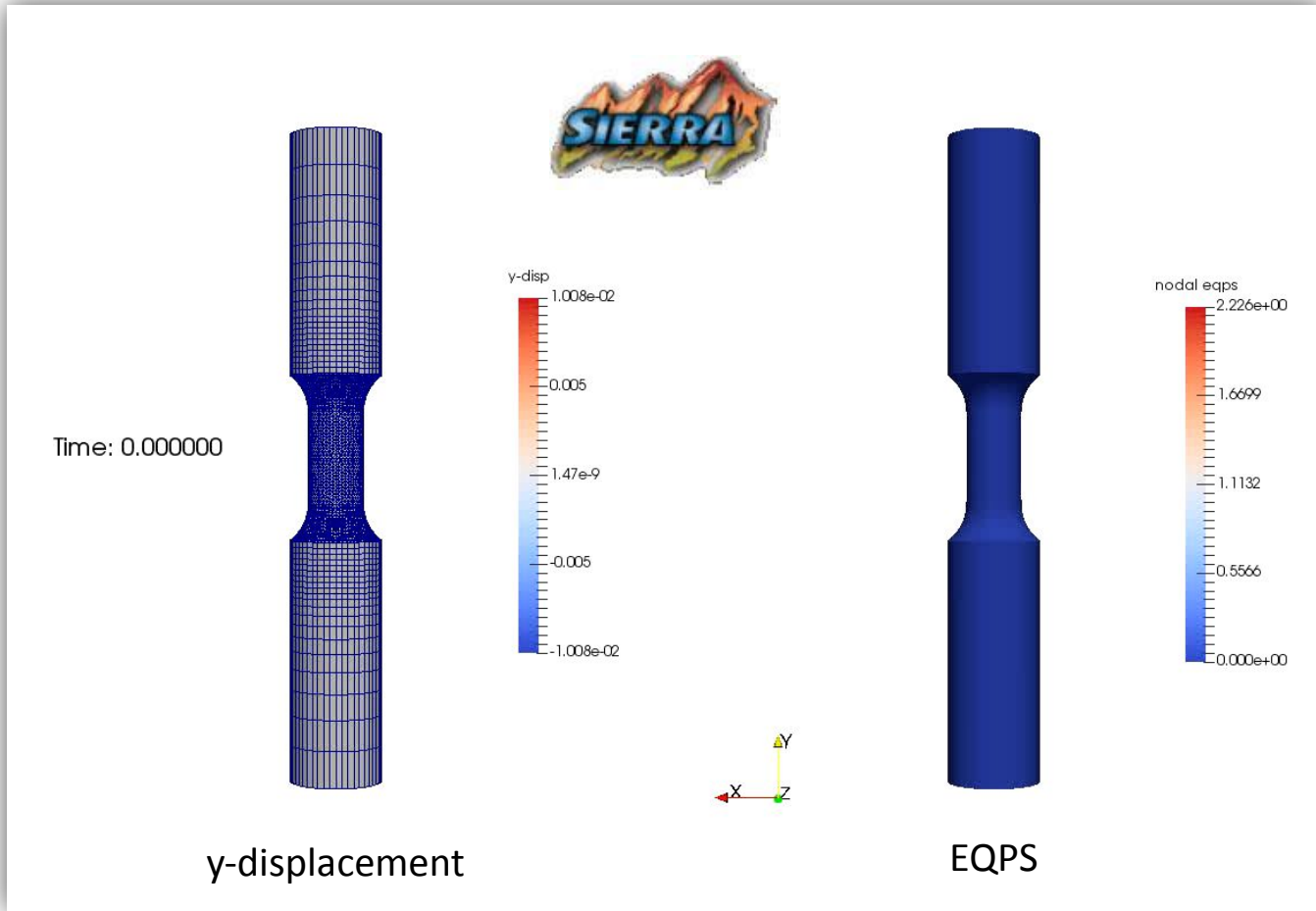
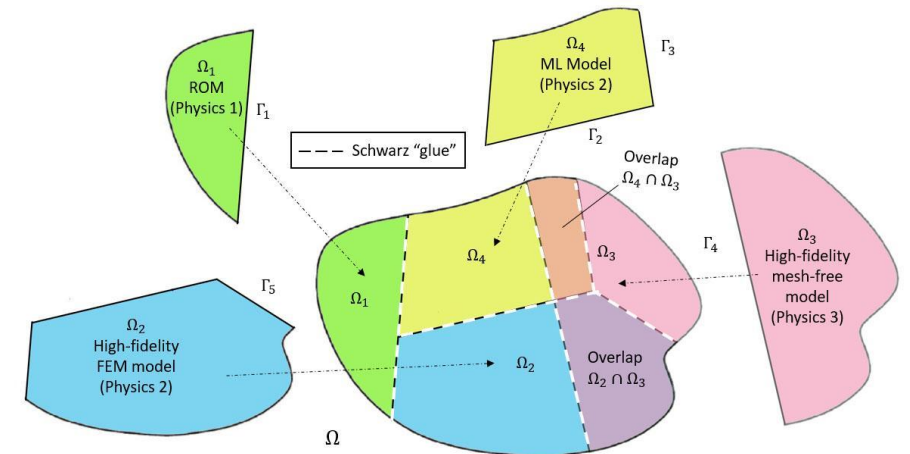
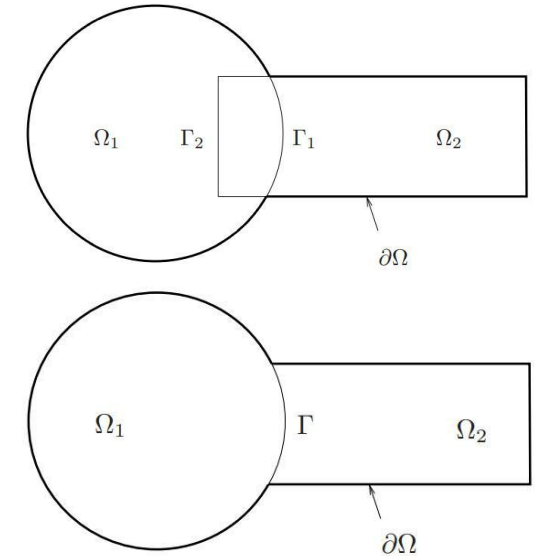


Figure above: tension specimen simulation coupling composite TET10 elements with HEX elements in Sierra/SM.

Figures right: bolted joint simulation coupling composite TET10 elements with HEX elements in Sierra/SM.

<sup>1</sup> Mota *et al.* 2017; Mota *et al.* 2022.

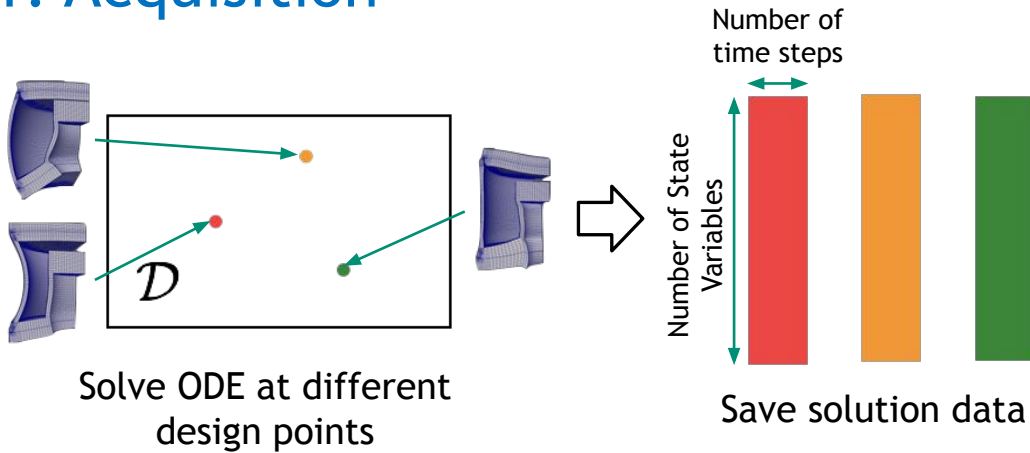
- The Schwarz Alternating Method for Domain Decomposition-Based Coupling
- **Extension to FOM\*-ROM<sup>#</sup> and ROM-ROM Coupling**
- Numerical Examples
  - 2D Burgers Equation
  - 2D Shallow Water Equations
  - Teaser: 2D Euler Equations Riemann Problem
- Summary & Future Work



Full Order Model (FOM):  $\frac{du}{dt} = f(\mathbf{u}; t, \boldsymbol{\mu})$

\* Least-Squares Petrov-Galerkin

## 1. Acquisition



## 2. Learning

Proper Orthogonal Decomposition (POD):

$$\mathbf{X} = \begin{bmatrix} \text{red} & \text{orange} & \text{green} \end{bmatrix} = \boldsymbol{\Phi} \mathbf{U} \quad \Sigma \quad \mathbf{V}^T$$

ROM = projection-based Reduced Order Model

## 3. Projection-Based Reduction

Choose ODE temporal discretization

$$\frac{du}{dt} = f(\mathbf{u}; t, \boldsymbol{\mu})$$

$$\Downarrow$$

$$\mathbf{r}^n(\mathbf{u}^n; \boldsymbol{\mu}) = \mathbf{0}, \quad n = 1, \dots, T$$

Reduce the number of unknowns

$$\mathbf{u}(t) \approx \tilde{\mathbf{u}}(t) = \boldsymbol{\Phi} \hat{\mathbf{u}}(t)$$

Minimize residual

$$\text{minimize}_{\hat{\mathbf{v}}} \left\| \begin{bmatrix} \mathbf{A} \\ \mathbf{r}^n(\boldsymbol{\Phi} \hat{\mathbf{v}}; \boldsymbol{\mu}) \end{bmatrix} \right\|_2$$

Hyper-reduction/sample mesh

HROM = Hyper-reduced ROM



## *Choice of domain decomposition*

- **Overlapping vs. non-overlapping** domain decomposition?
  - Non-overlapping more flexible but typically requires more Schwarz iterations
- **FOM vs. ROM** subdomain assignment?
  - Do not assign ROM to subdomains where they have no hope of approximating solution

## *Snapshot collection and reduced basis construction*

- Are subdomains **simulated independently** in each subdomains or together?

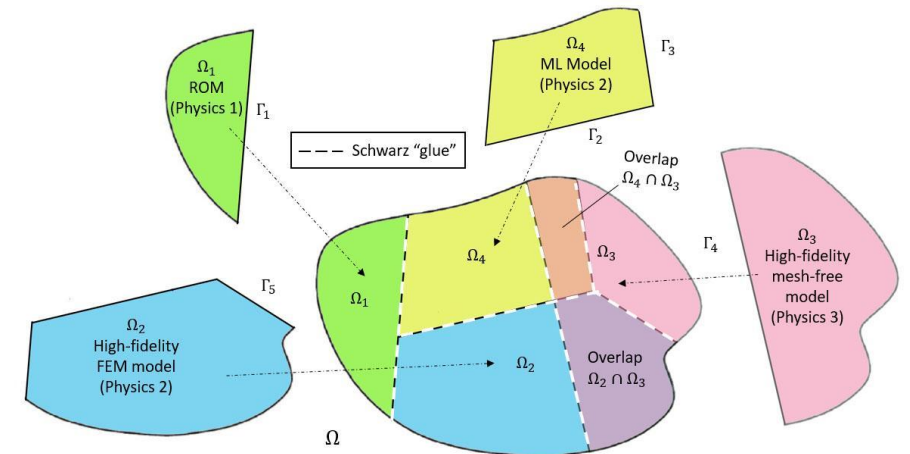
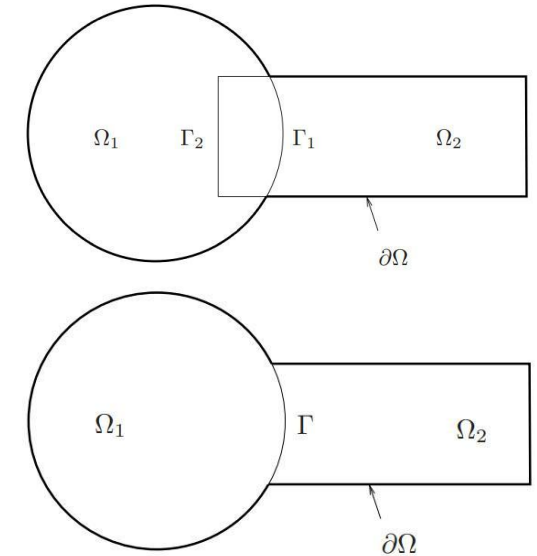
## *Enforcement of boundary conditions (BCs) in ROM at Schwarz boundaries*

- **Strong vs. weak** BC enforcement?
  - Strong BC enforcement difficult for some models (e.g., cell-centered finite volume, PINNs)
- **Optimizing parameters** in Schwarz BCs for non-overlapping Schwarz?

## *Choice of hyper-reduction*

- What **hyper-reduction** method to use?
  - Application may require particular method (e.g., ECSW for solid mechanics problems)
- How to **sample Schwarz boundaries** in applying hyper-reduction?
  - Need to have enough sample mesh points at Schwarz boundaries to apply Schwarz

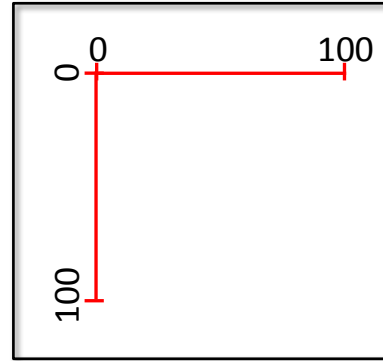
- The Schwarz Alternating Method for Domain Decomposition-Based Coupling
- Extension to FOM\*-ROM<sup>#</sup> and ROM-ROM Coupling
- Numerical Examples
  - 2D Burgers Equation
  - 2D Shallow Water Equations
  - Teaser: 2D Euler Equations Riemann Problem
- Summary & Future Work



# 2D Inviscid Burgers Equation

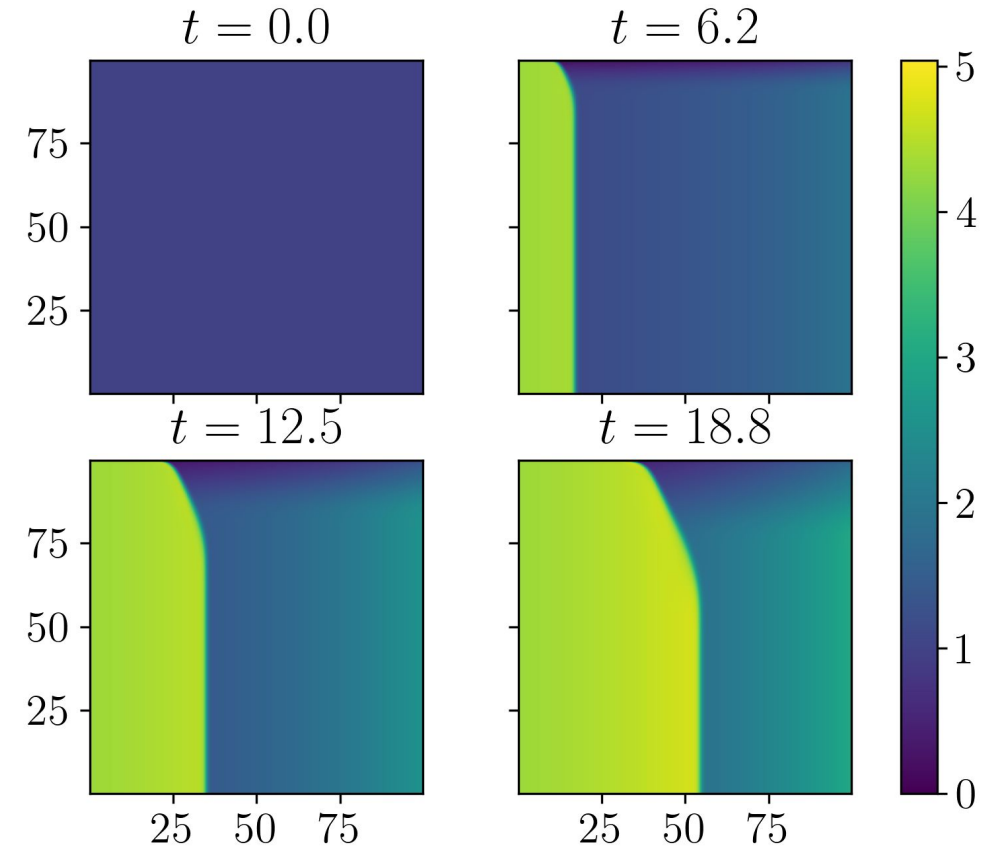


Popular analog for fluid problems where **shocks** are possible, and particularly **difficult** for conventional projection-based ROMs



## Problem setup:

- $\Omega = (0, 100)^2$ ,  $t \in [0, 25]$
- Two parameters  $\mu = (\mu_1, \mu_2)$ . **Training:** uniform sampling of  $\mu_1 \times \mu_2 = [4.25, 5.50] \times [0.015, 0.03]$  by a  $3 \times 3$  grid. **Testing:** query unsampled point  $\mu = [4.75, 0.02]$





# Schwarz Coupling Details

## Choice of domain decomposition

- Overlapping DD of  $\Omega$  into 4 subdomains coupled via multiplicative Schwarz
- Solution in  $\Omega_1$  is most difficult to capture by ROM

## Snapshot collection and reduced basis construction

- Single-domain FOM on  $\Omega$  used to generate snapshots/POD modes

## Enforcement of boundary conditions (BCs) in ROM at Schwarz boundaries

- BCs imposed strongly via Method 1 of [Gunzburger *et al.*, 2007] at indices  $i_{\text{Dir}}$

$$\mathbf{q}(t) \approx \bar{\mathbf{q}} + \Phi \hat{\mathbf{q}}(t)$$

- POD modes made to satisfy homogeneous DBCs:  $\Phi(\mathbf{i}_{\text{Dir}}, :) = \mathbf{0}$
- BCs imposed by modifying  $\bar{\mathbf{q}}$ :  $\bar{\mathbf{q}}(\mathbf{i}_{\text{Dir}}) \leftarrow \chi_q$

## Choice of hyper-reduction

- Energy Conserving Sampling & Weighting (ECSW) method for hyper-reduction
- All points on Schwarz boundaries are included in the sample mesh

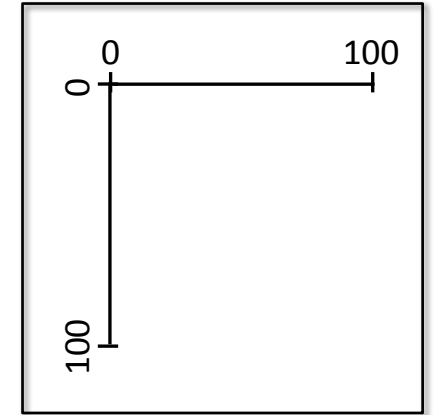


Figure above: 4 subdomain overlapping DD

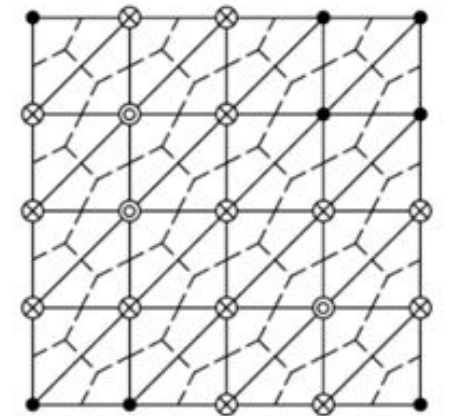
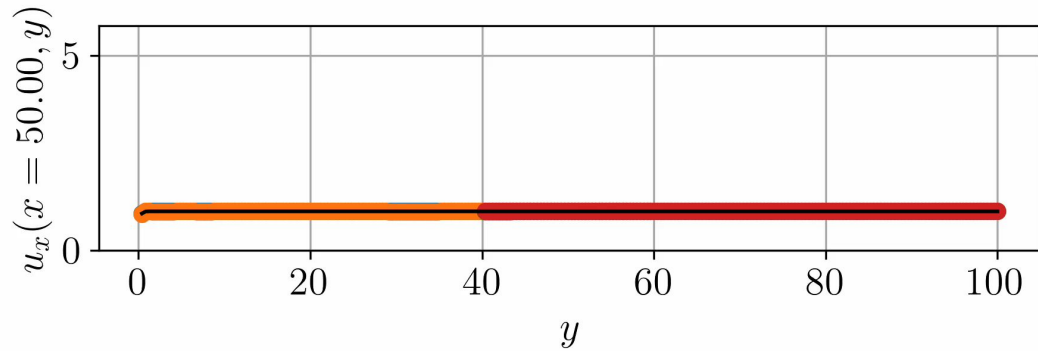
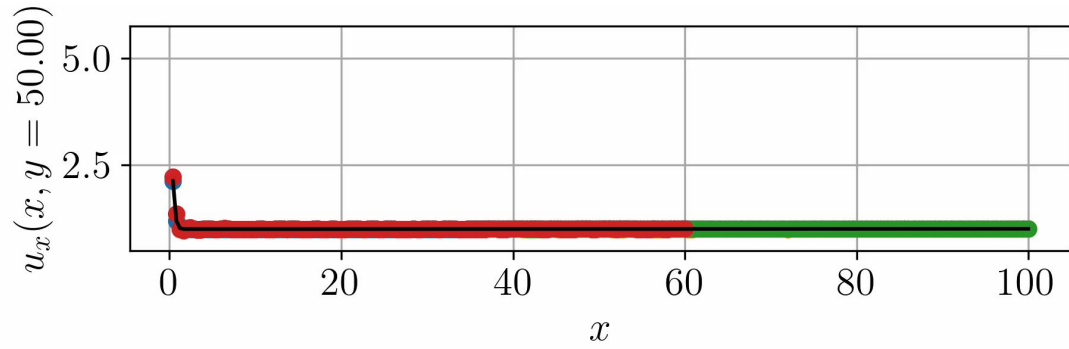
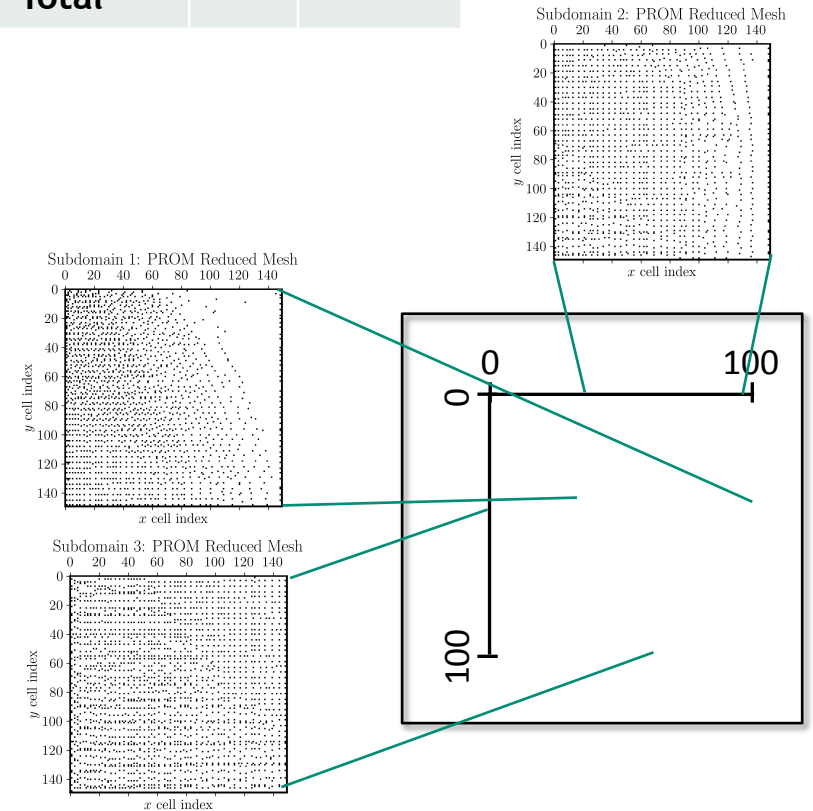


Figure above: ECSW augmented reduced mesh

# FOM-HROM-HROM-HROM Coupling

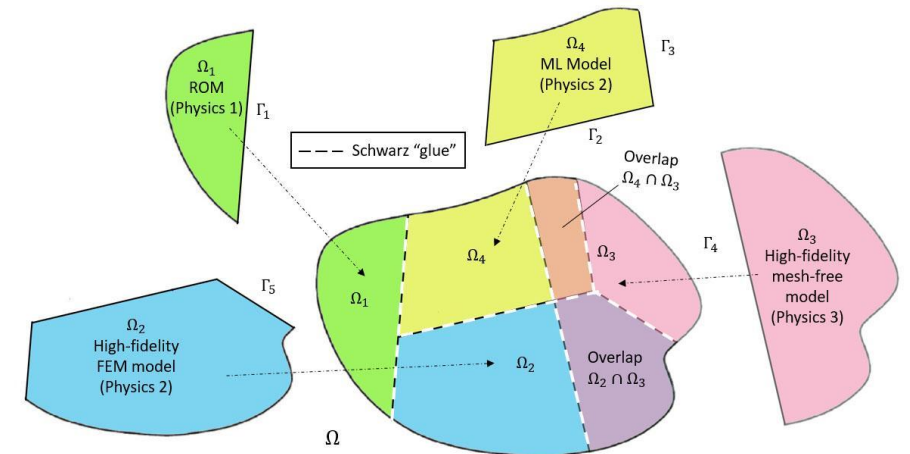
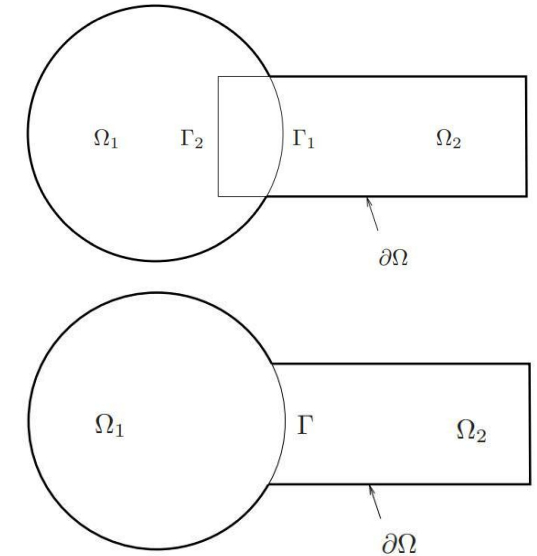


Subdomains	99% SV Energy	
	MSE (%)	CPU time (s)
		95
120	0.26	26
60	0.43	17
66	0.34	21
<b>Total</b>		<b>159</b>



Further speedups possible via code optimizations, additive Schwarz and reduction of # sample mesh points.

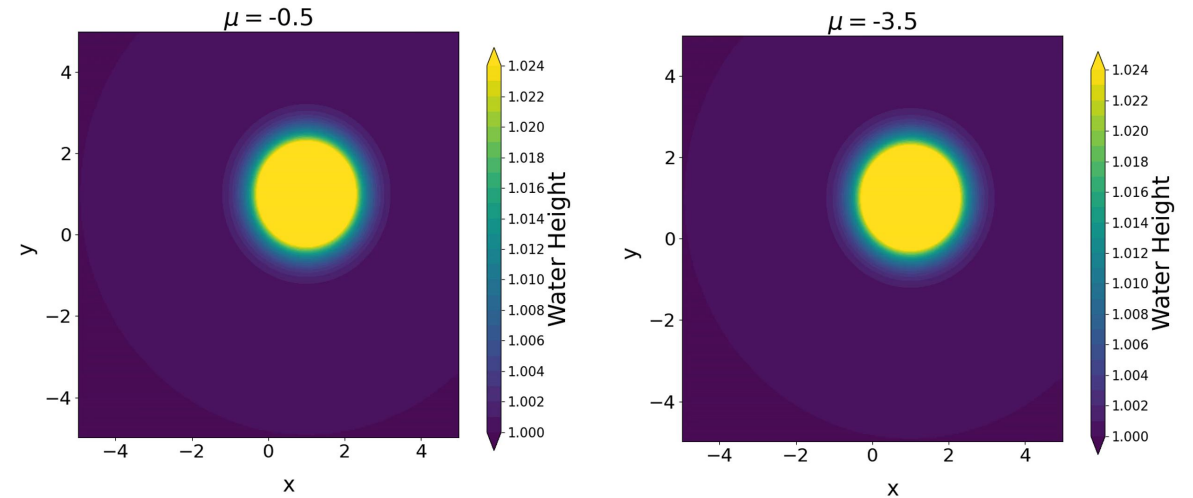
- The Schwarz Alternating Method for Domain Decomposition-Based Coupling
- Extension to FOM\*-ROM<sup>#</sup> and ROM-ROM Coupling
- Numerical Examples
  - 2D Burgers Equation
  - 2D Shallow Water Equations
  - Teaser: 2D Euler Equations Riemann Problem
- Summary & Future Work



# 2D Shallow Water Equations (SWE)



Hyperbolic PDEs modeling **wave propagation** below a pressure surface in a fluid (e.g., atmosphere, ocean).



# Schwarz Coupling Details

**Green:** different from Burgers' problem

## Choice of domain decomposition

- **Non-overlapping** DD of  $\Omega$  into 4 subdomains coupled via **additive Schwarz**
  - **OpenMP parallelism** with 1 thread/subdomain
- **All-ROM** or **All-HROM** coupling via Pressio\*

## Snapshot collection and reduced basis construction

- **Single-domain FOM** on  $\Omega$  used to generate snapshots/POD modes

## Enforcement of boundary conditions (BCs) in ROM at Schwarz boundaries

- BCs are imposed **approximately** by fictitious ghost cell states
  - Implementing Neumann and Robin BCs is **challenging**
- **Ghost cells** introduce some overlap even with non-overlapping DD
  - $\Rightarrow$  **Dirichlet-Dirichlet non-overlapping Schwarz is stable/convergent!**

## Choice of hyper-reduction

- **Collocation** for hyper-reduction: min residual at small subset DOFs
- Assume **fixed budget of sample mesh points** at Schwarz boundaries

\*<https://github.com/Pressio/pressio-demoapps>

Figure right:  
non-overlapping DD w/  
ghost cells creating  
overlap

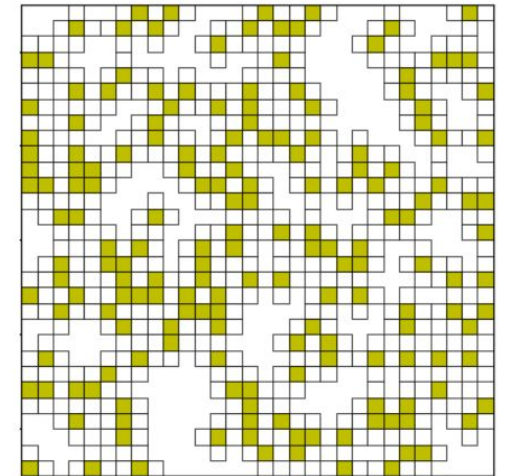
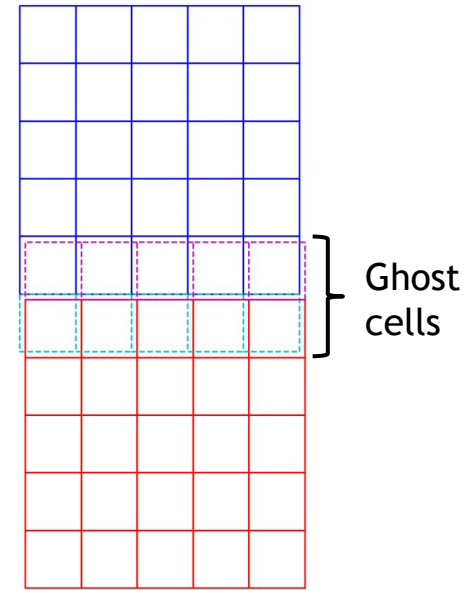
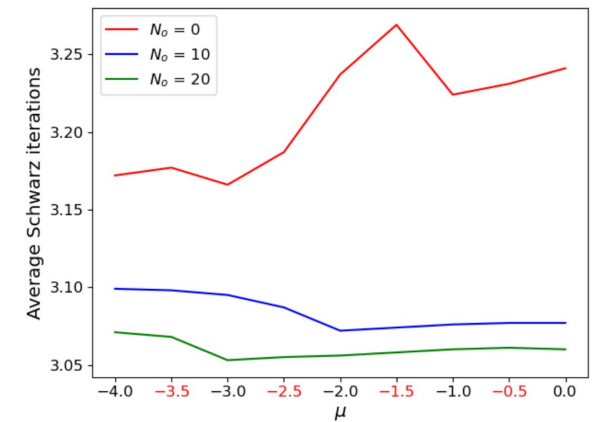
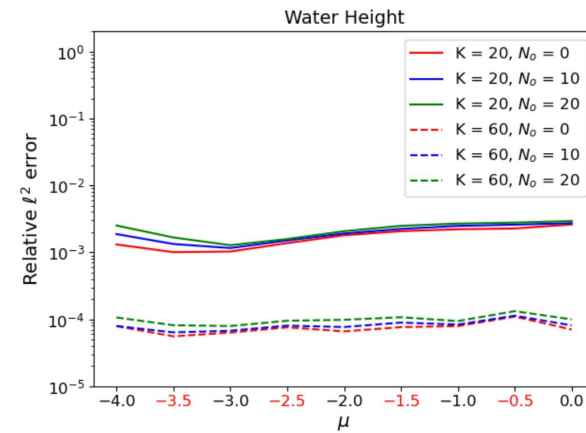
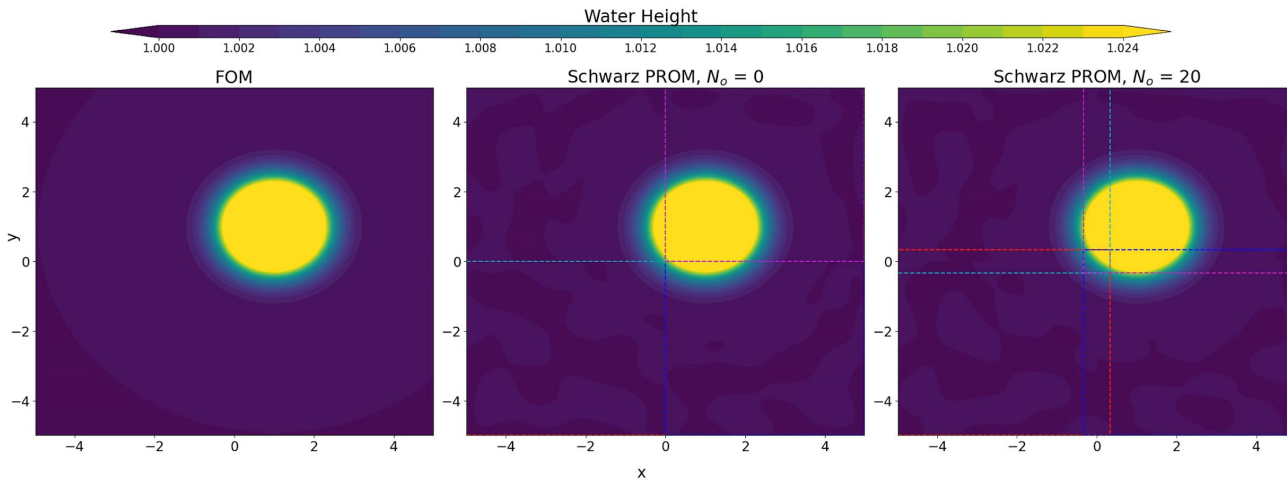


Figure above: sample mesh (yellow) and stencil (white) cells

# Schwarz All-ROM Domain Overlap Study

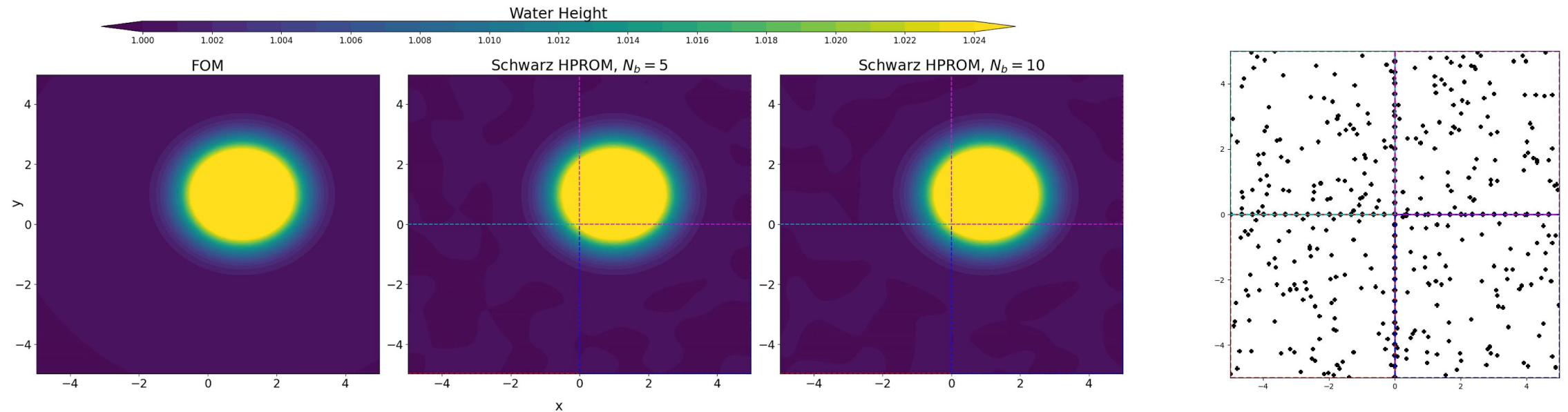


# Schwarz Boundary Sampling for All-HROM Coupling



**Key question:** how many Schwarz boundary points need to be included in **sample mesh** when performing HROM coupling?

- Naïve/sparse-sampled Schwarz boundary results in **failure** to transmit coupling information during Schwarz



Movie above: FOM (left), all HROM with  $N_b = 5\%$  (middle) and all HROM with  $N_b = 10\%$  (left). ROMs have  $K = 100$  modes and  $N_s = 0.5\%N$  sample mesh points.

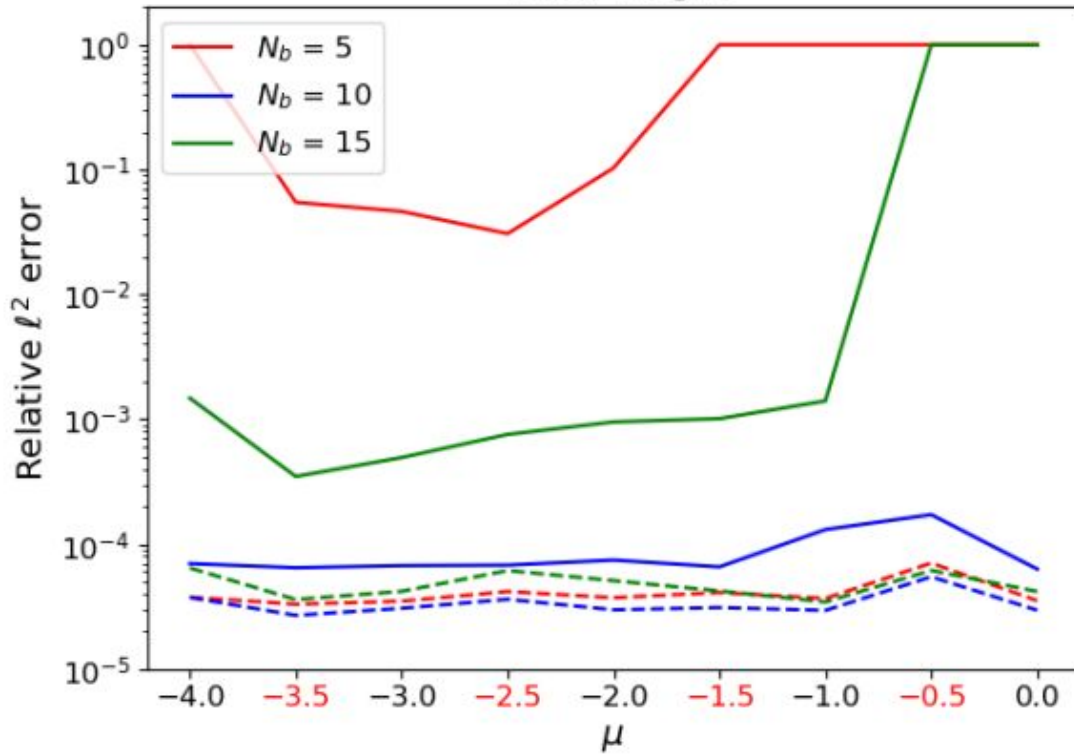
Figure above: example sample mesh with sampling rate  $N_b = 10\%$

- Including too many Schwarz boundary points ( $N_b$ ) in sample mesh given fixed budget of  $N_s$  sample mesh points may lead to too few sample mesh points in interior
- For SWE problem, we can get away with  $\sim 10\%$  boundary sampling (movie above, right-most frame)

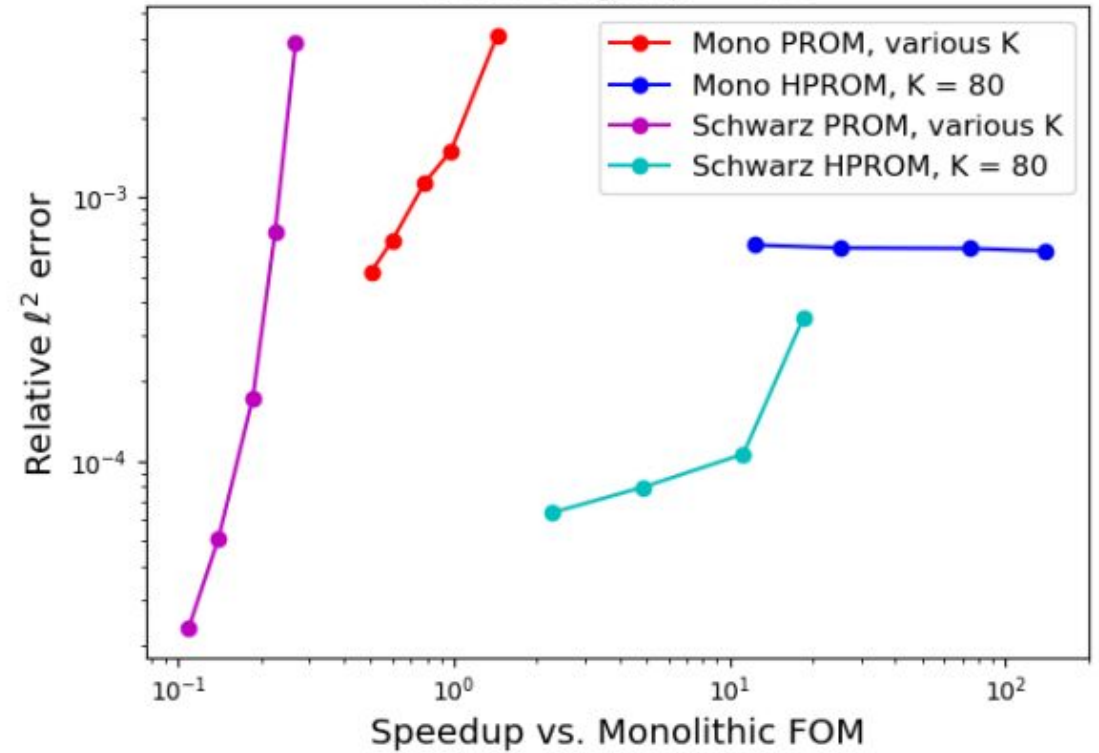
# Coupled HRROM Performance



Water Height

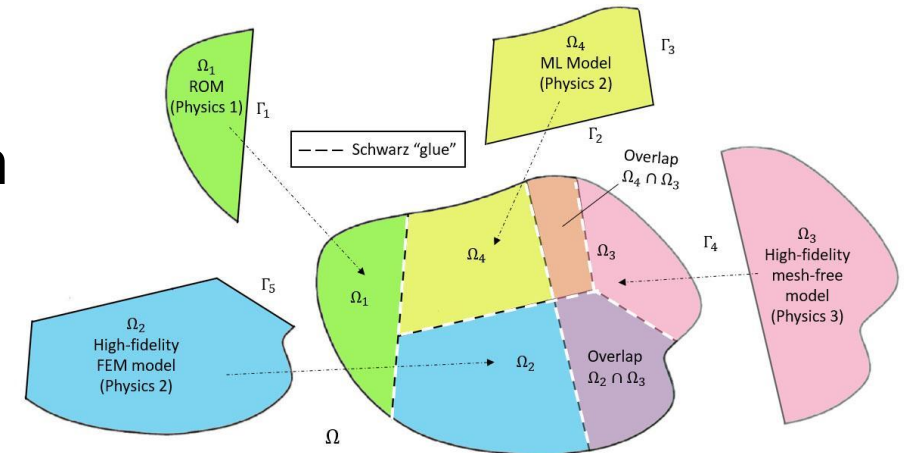
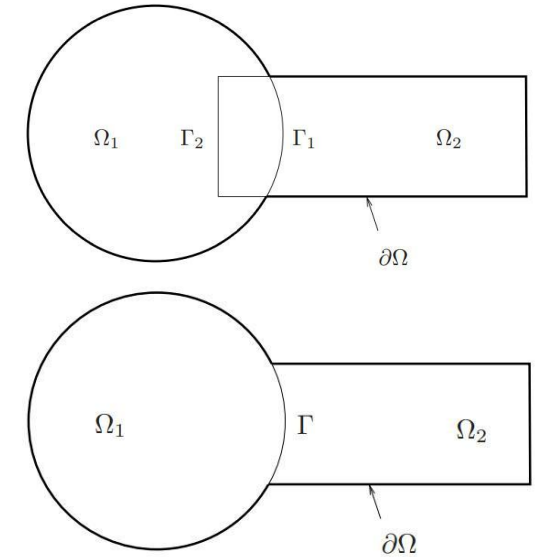


Water Height,  $\mu = -0.5$

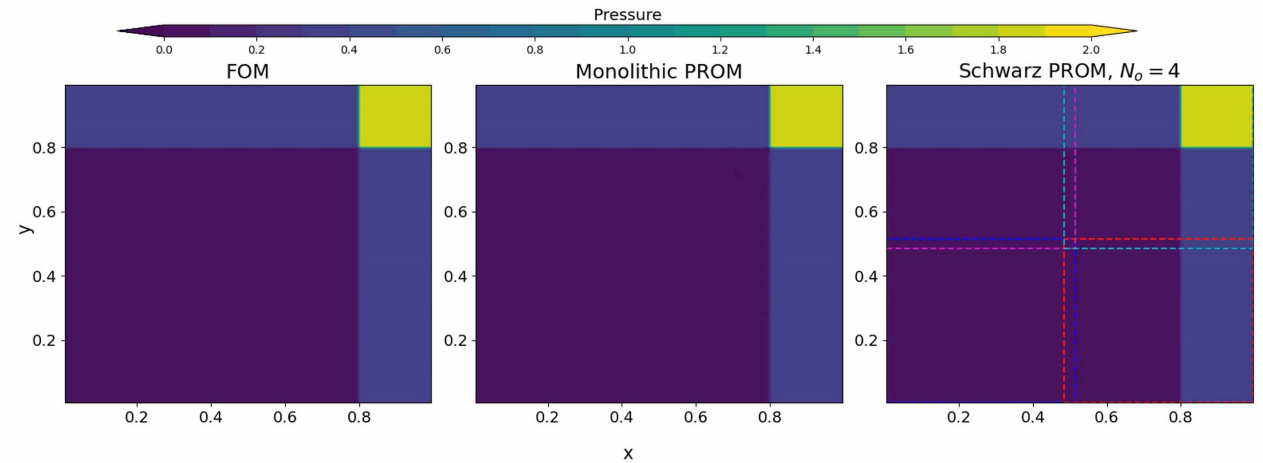




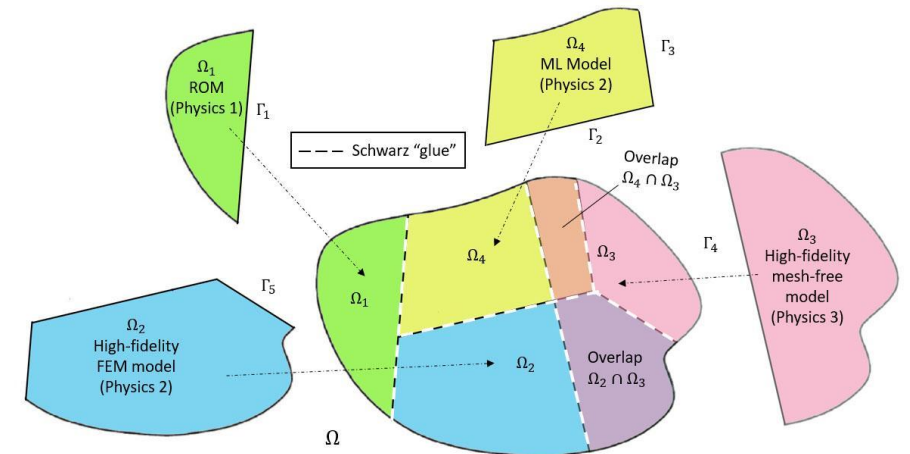
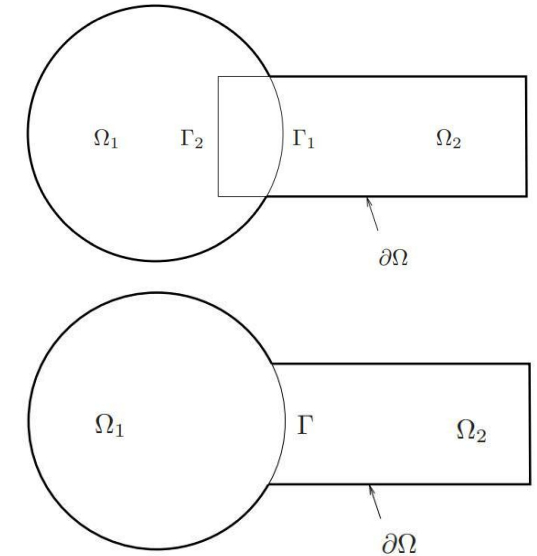
- The Schwarz Alternating Method for Domain Decomposition-Based Coupling
- Extension to FOM\*-ROM<sup>#</sup> and ROM-ROM Coupling
- Numerical Examples
  - 2D Burgers Equation
  - 2D Shallow Water Equations
  - Teaser: 2D Euler Equations Riemann Problem
- Summary & Future Work



# Teaser: 2D Euler Equations Riemann Problem



- The Schwarz Alternating Method for Domain Decomposition-Based Coupling
- Extension to FOM\*-ROM<sup>#</sup> and ROM-ROM Coupling
- Numerical Examples
  - 2D Burgers Equation
  - 2D Shallow Water Equations
  - Teaser: 2D Euler Equations Riemann Problem
- Summary & Future Work

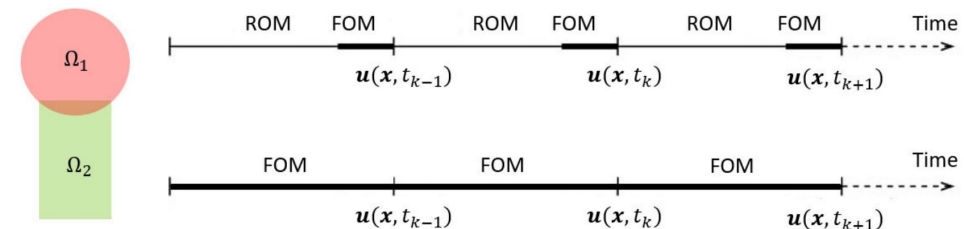
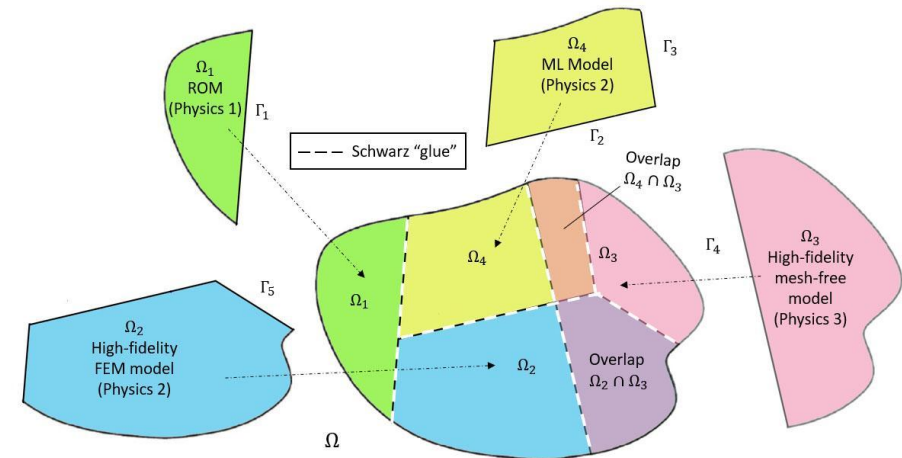


## Summary:

- Schwarz has been **demonstrated** for **coupling** of FOMs and (H)ROMs
- **Computational gains** can be achieved by coupling HROMs and using the additive Schwarz variant

## Ongoing & future work:

- Extension to **other applications** (fasteners, laser welds)
- **Rigorous analysis** of why Dirichlet-Dirichlet BC “work” when employing non-overlapping Schwarz with discretizations that employ ghost cells
- **Learning** of “optimal” transmission conditions to ensure **structure preservation**
- Extension of Schwarz to enabling coupling of **non-intrusive ROMs** (e.g., DMD, OpInf, Neural Networks)
- Development of **automated criteria** to determine appropriate use of less refined or reduced-order models without sacrificing accuracy, enabling **real-time transitions** between different model fidelities → **New project: AHead LDRD**





Irina Tezaur



Chris Wentland



Francesco Rizzi



Joshua Barnett



Alejandro Mota



Will Snyder  
*Former Intern from  
Virginia Tech  
[Schwarz + PINNs]*



Ian Moore  
*Intern from  
Virginia Tech  
[Schwarz + OpInf]*



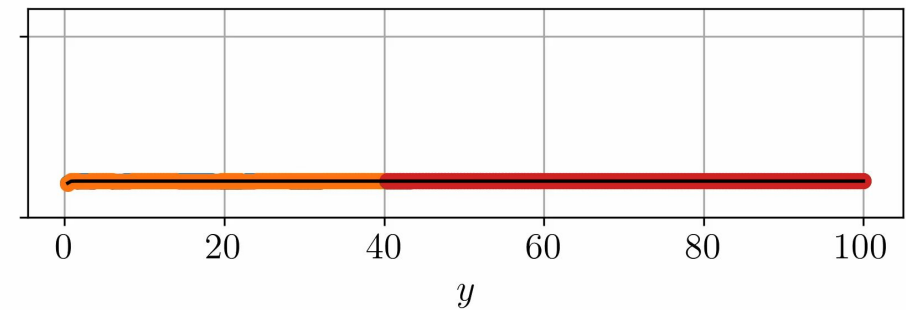
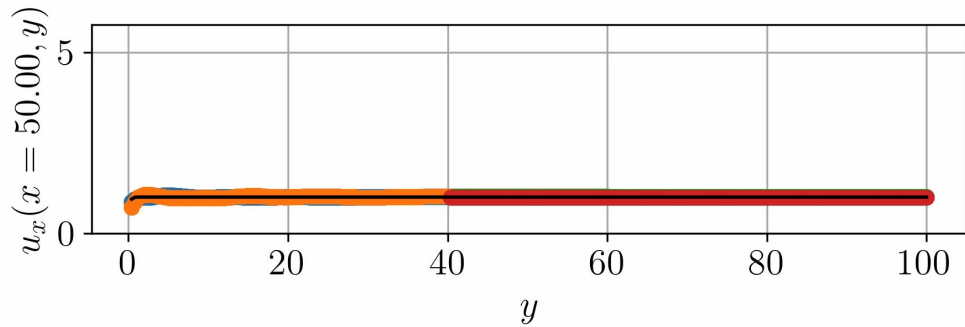
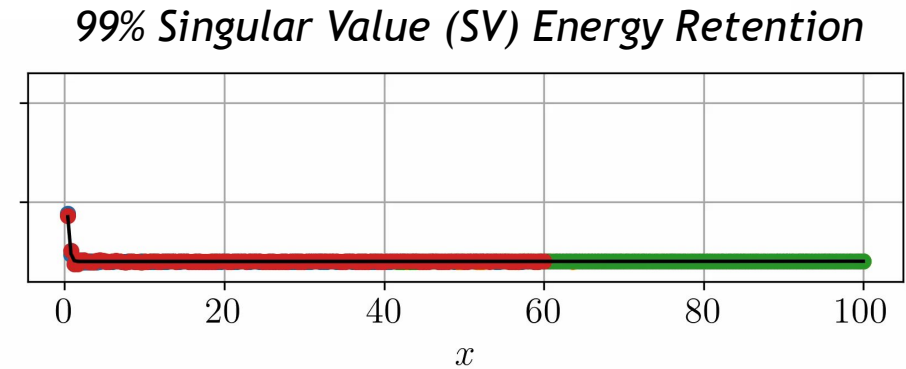
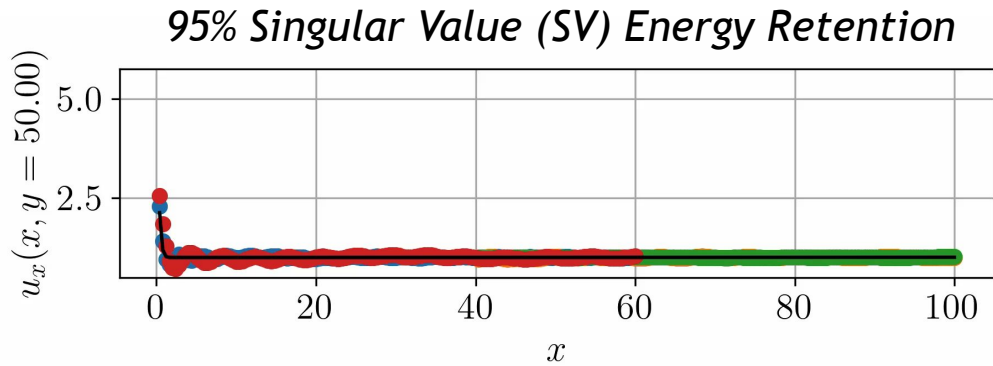
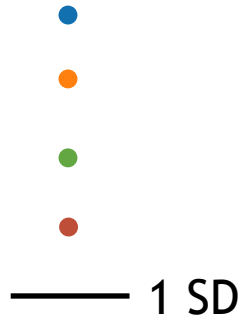


- [1] A. Mota, I. Tezaur, C. Alleman. “The Schwarz Alternating Method in Solid Mechanics”, *Comput. Meth. Appl. Mech. Engng.* 319 (2017), 19-51.
- [2] A. Mota, I. Tezaur, G. Phlipot. “The Schwarz Alternating Method for Dynamic Solid Mechanics”, *Comput. Meth. Appl. Mech. Engng.* 121 (21) (2022) 5036-5071.
- [3] J. Barnett, I. Tezaur, A. Mota. “The Schwarz alternating method for the seamless coupling of nonlinear reduced order models and full order models”, ArXiv pre-print, 2022.  
<https://arxiv.org/abs/2210.12551>
- [4] W. Snyder, I. Tezaur, C. Wentland. “Domain decomposition-based coupling of physics-informed neural networks via the Schwarz alternating method”, ArXiv pre-print, 2023.  
<https://arxiv.org/abs/2311.00224>
- [5] A. Mota, D. Koliesnikova, I. Tezaur. “A Fundamentally New Coupled Approach to Contact Mechanics via the Dirichlet-Neumann Schwarz Alternating Method”, ArXiv pre-print, 2023.  
<https://arxiv.org/abs/2311.05643>

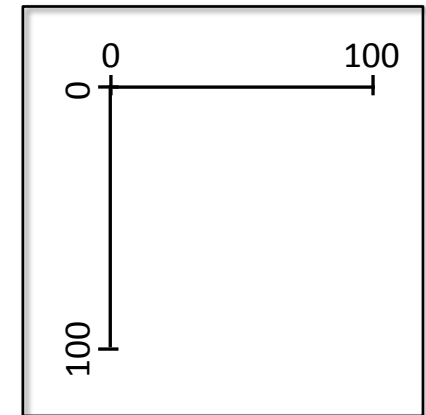
**Email:** [ikalash@sandia.gov](mailto:ikalash@sandia.gov)  
**URL:** [www.sandia.gov/~ikalash](http://www.sandia.gov/~ikalash)

## Start of Backup Slides

# All-ROM Coupling



Subdomains	95% SV Energy		99% SV Energy		
	MSE (%)	CPU time (s)	MSE (%)	CPU time (s)	
57	1.1	85	146	0.18	295
44	1.2	56	120	0.18	216
24	1.4	43	60	0.16	89
32	1.9	61	66	0.25	100
<b>Total</b>		<b>245</b>			<b>700</b>





The Schwarz alternating method has been developed for concurrent multi-scale coupling of conventional and data-driven models.

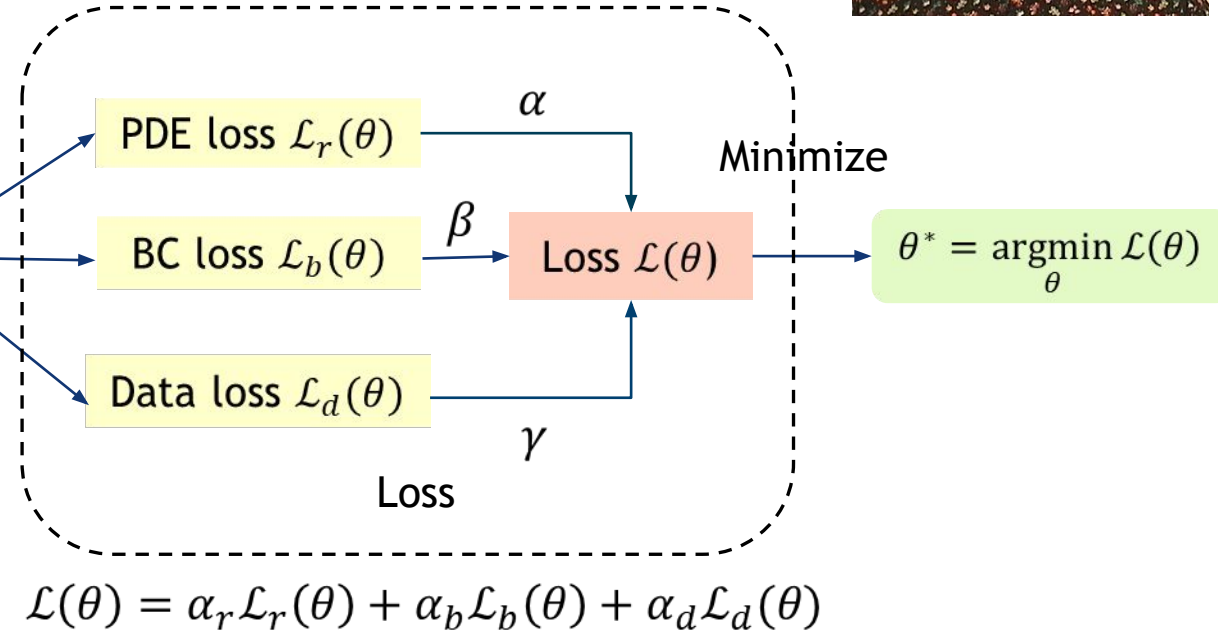
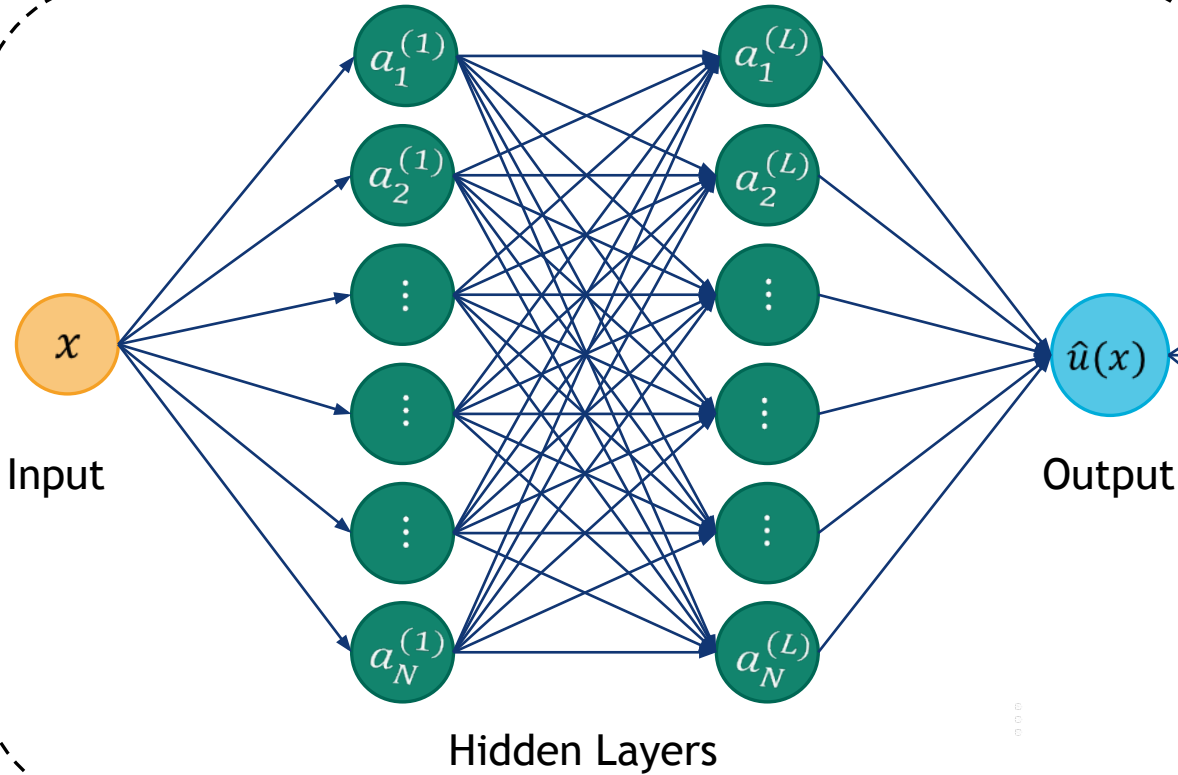
- 😊 Coupling is *concurrent* (two-way).
- 😊 *Ease of implementation* into existing massively-parallel HPC codes.
- 😊 “*Plug-and-play*” framework: simplifies task of meshing complex geometries!
  - 😊 Ability to couple regions with *different non-conformal meshes, different element types* and *different levels of refinement*.
  - 😊 Ability to use *different solvers (including ROM/FOM)* and *time-integrators* in different regions.
- 😊 *Scalable, fast, robust* on *real* engineering problems
- 😊 Coupling does not introduce *nonphysical artifacts*.
- 😊 *Theoretical* convergence properties/guarantees.

# Bonus: PINN-PINN and PINN-FOM coupling

Will Snyder  
Summer Intern  
Virginia Tech



Neural Network



*Focus thus far*

**Goal:** investigate the use of the Schwarz alternating method as a means to couple **Physics-Informed Neural Networks (PINNs)**

**Scenario 1:** use Schwarz to train subdomain PINNs (offline)

**Scenario 2:** use Schwarz to couple pre-trained subdomain PINNs/NNs (online)

## Bonus: PINN-PINN coupling

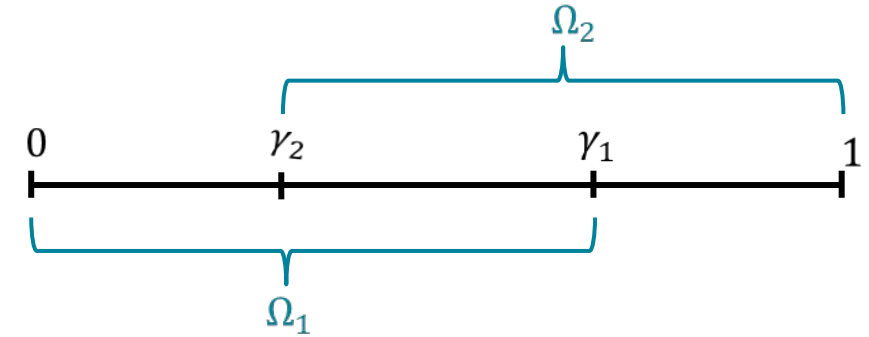


1D steady advection-diffusion equation on  $\Omega = [0,1]$ :

$$u_x - v u_{xx} = 1, \quad u(0) = u(1) = 0$$

PINNs are notoriously difficult to train for higher Peclet numbers!

Can Schwarz help?



Overlapping DD:  $\Omega = \Omega_1 \cup \Omega_2$  with boundary  $\partial\Omega = \{0,1\}$

$$\mathcal{L}_{r,i}(\theta) = \text{MSE}(-v \nabla_x^2 NN_{\Omega_i}(x, \theta) + \nabla_x NN_{\Omega_i}(x, \theta) - 1)$$

$$\mathcal{L}_{b,i}(\theta) = \text{MSE}(NN_{\Omega_i}(\partial\Omega, \theta)) + \text{MSE}(NN_{\Omega_i}(\gamma_i, \theta) - NN_{\Omega_j}(\gamma_i, \theta))$$

### Schwarz PINN training algorithm:

Loop over subdomains  $\Omega_i$  until convergence of Schwarz method

Train PINN in  $\Omega_i$  with loss  $\mathcal{L}_i(\theta) = \alpha \mathcal{L}_{r,i}(\theta) + \beta \mathcal{L}_{b,i}(\theta) + \gamma \mathcal{L}_{d,i}(\theta)$

Communicate Dirichlet data between neighboring subdomains

Update boundary data on  $\gamma_i$  from neighboring subdomains

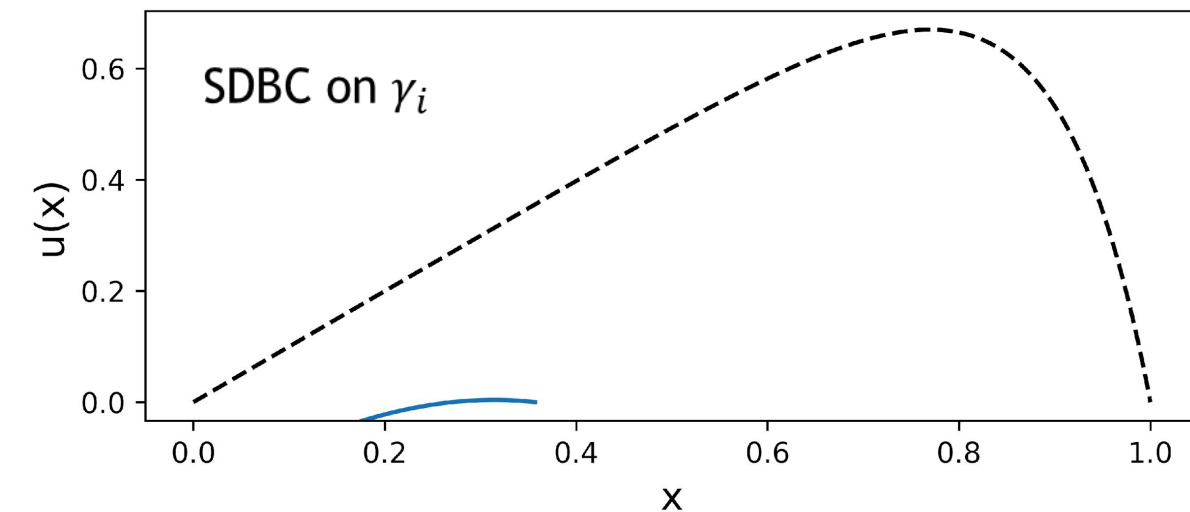
If strong enforcement of Dirichlet BC (SDBC), set  $\hat{u}_{\Omega_i}(x, \theta) = NN_{\Omega_i}(x, \theta)$

If weak enforcement of Dirichlet BC (WDBC), set  $\beta = 0$  and  $\hat{u}_{\Omega_i}(x, \theta) = v(x) NN_{\Omega_i}(x, \theta) + \psi(x) \hat{u}_{\Omega_j}(\gamma_j, \theta)$

where  $v(x)$  is chosen s.t.  $v(0) = v(\gamma_i) = v(1) = 0$  and  $\psi(x)$  is chosen s.t.  $\psi(\gamma_i) = 1$

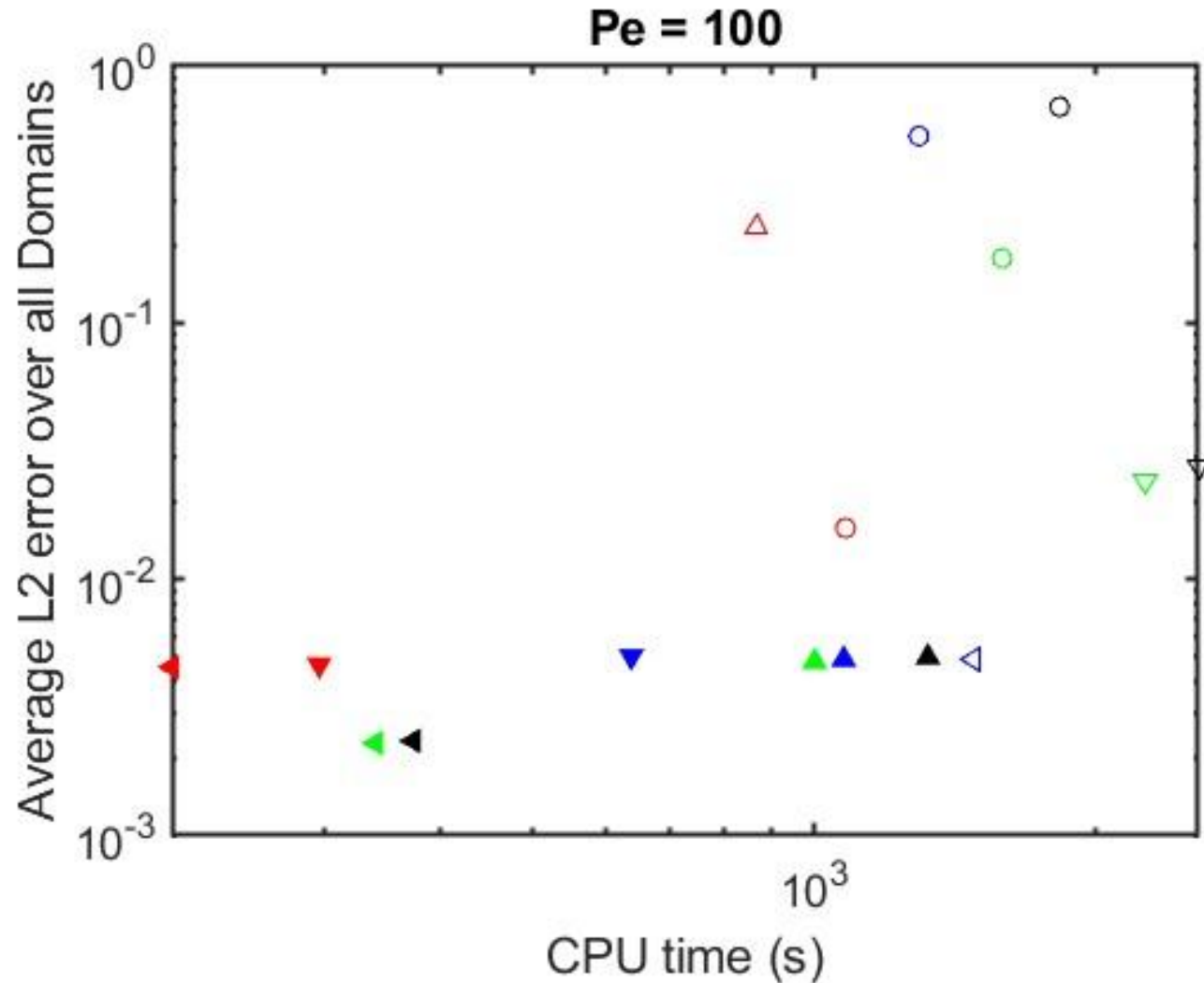


Schwarz iteration 1;  $Pe = 10$



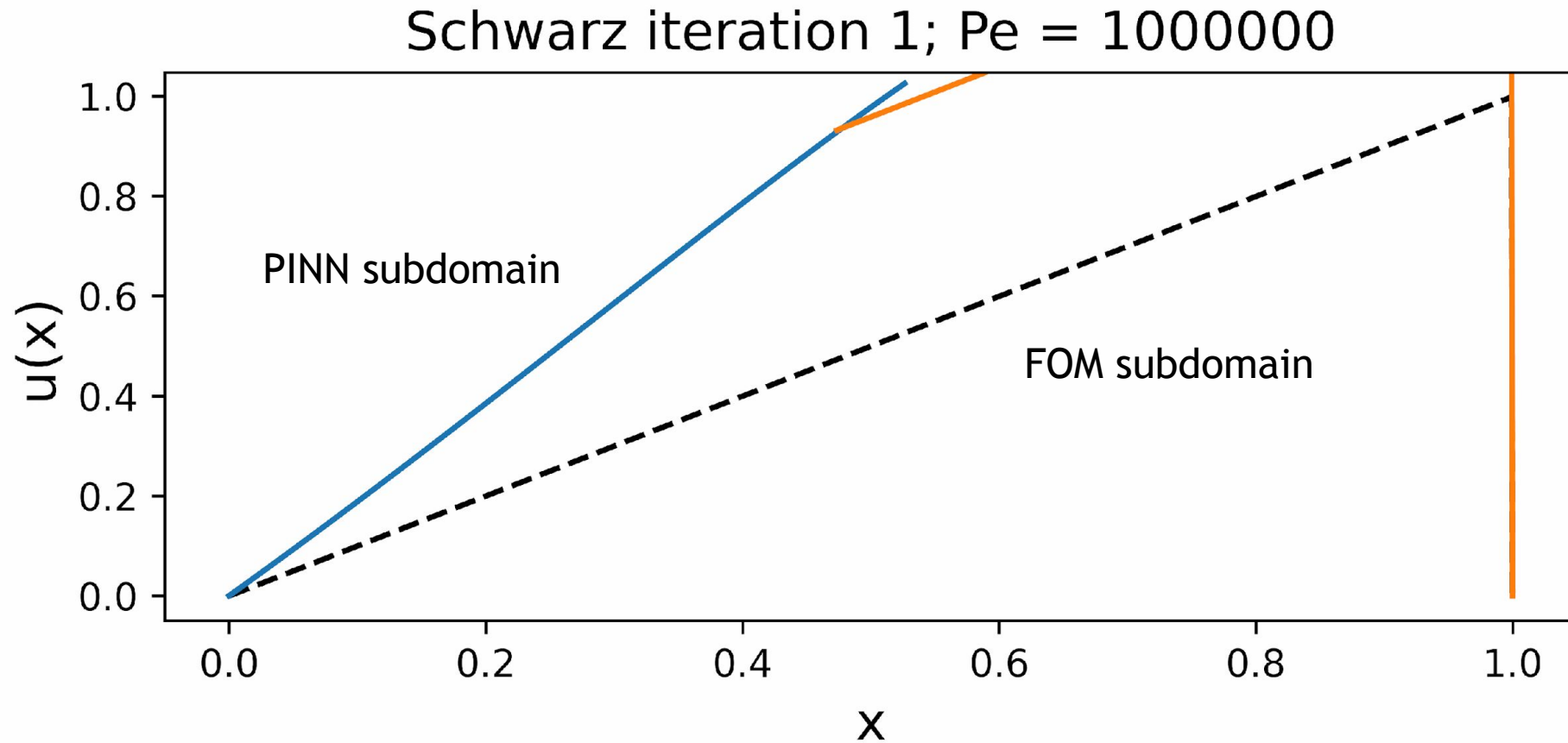
- How **Dirichlet boundary conditions** are handled has a large impact on PINN convergence
- Convergence not improved in general with **increasing overlap**
- Increasing **# subdomains** in general will increase CPU time

# Bonus: PINN-PINN coupling



- 2  $\Omega$ , no snapshots, WDBC (unconverged)
- ▼ 2  $\Omega$ , no snapshots, SDBC
- △ 2  $\Omega$ , snapshots, WDBC (unconverged)
- ▲ 2  $\Omega$ , snapshots, SDBC
- 3  $\Omega$ , no snapshots, WDBC (unconverged)
- ▼ 3  $\Omega$ , no snapshots, SDBC
- ▲ 3  $\Omega$ , snapshots, WDBC
- ◁ 3  $\Omega$ , snapshots SDBC (unconverged)
- 4  $\Omega$ , no snapshots, WDBC (unconverged)
- ▽ 4  $\Omega$ , no snapshots, SDBC (unconverged)
- ▲ 4  $\Omega$ , snapshots, WDBC
- ◀ 4  $\Omega$ , snapshots SDBC
- 5  $\Omega$ , no snapshots, WDBC (unconverged)
- ▽ 5  $\Omega$ , no snapshots, SDBC (unconverged)
- ▲ 5  $\Omega$ , snapshots, WDBC
- ◀ 5  $\Omega$ , snapshots, SDBC

- Using **SDBC**s and **data loss** helps with PINN/NN convergence and accuracy



- PINN-FOM coupling gives rapid PINN convergence for arbitrarily high Peclet numbers
- PINN-FOM couplings works with both WDBC and SDBC configurations

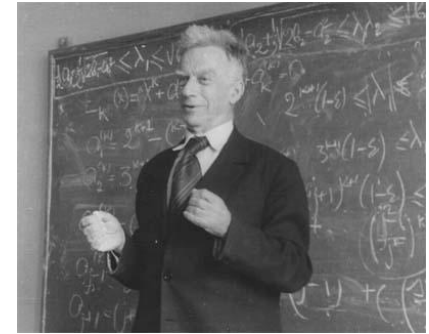
# Theoretical Foundation

Using the Schwarz alternating as a **discretization method** for PDEs is natural idea with a sound **theoretical foundation**.

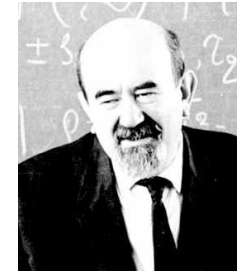
- **S.L. Sobolev (1936)**: posed Schwarz method for **linear elasticity** in variational form and **proved method's convergence** by proposing a convergent sequence of energy functionals.
- **S.G. Mikhlin (1951)**: **proved convergence** of Schwarz method for general linear elliptic PDEs.
- **P.-L. Lions (1988)**: studied convergence of Schwarz for **nonlinear monotone elliptic problems** using max principle.
- **A. Mota, I. Tezaur, C. Alleman (2017)**: proved **convergence** of the alternating Schwarz method for **finite deformation quasi-static nonlinear PDEs** (with energy functional  $\Phi[\varphi]$ ) with a **geometric convergence rate**.

$$\Phi[\varphi] = \int_B A(\mathbf{F}, \mathbf{Z}) dV - \int_B \mathbf{B} \cdot \varphi dV$$

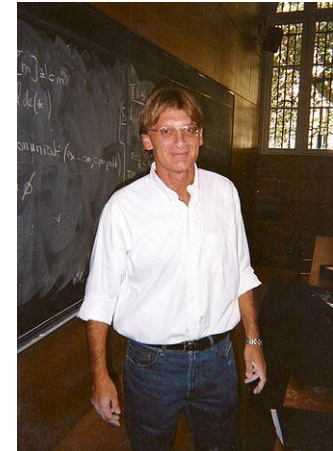
$$\nabla \cdot \mathbf{P} + \mathbf{B} = \mathbf{0}$$



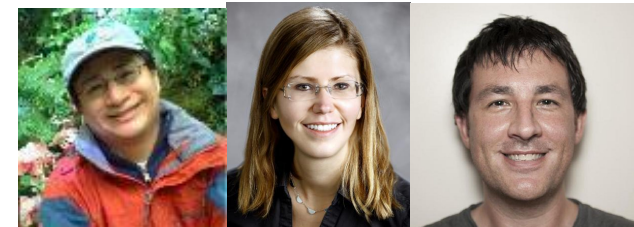
S.L. Sobolev (1908 – 1989)



S.G. Mikhlin  
(1908 – 1990)



P.-L. Lions (1956-)



A. Mota, I. Tezaur, C. Alleman

# Convergence Proof\*



## 2 Formulation of the Schwarz Alternating Method

We start by defining the standard finite-dimensional variational formulation to establish weakly lower bounding the formulation of the coupling method.

### 2.1 Variational Formulation on a Single Domain

Consider a body as the open set  $\Omega \subset \mathbb{R}^3$  undergoing a motion described by the mapping  $\varphi : \Omega \times [0, T] \rightarrow \mathbb{R}^3$ ,  $X \in \Omega$ . Assume that the boundary of the body is split into a Dirichlet boundary  $\partial_D \Omega$  and a Neumann boundary  $\partial_N \Omega$ . The prescribed boundary conditions are  $\varphi|_{\partial_D \Omega} = \bar{\varphi}$  and  $\sigma \cdot n|_{\partial_N \Omega} = \bar{\sigma}$ . The prescribed boundary traction or Neumann boundary conditions are  $\bar{\sigma} : \mathbb{R}^{3 \times 3} \rightarrow \mathbb{R}^{3 \times 3}$ . Let  $\mathcal{F}$  be the deformation gradient. Let also  $\mathcal{H} : \mathbb{R}^{3 \times 3} \rightarrow \mathbb{R}^3$  be the body force with  $\mathcal{H}$  the mass density in the reference configuration. Furthermore, introduce the energy functional

$$\Phi[\varphi] = \int_{\Omega} A(\mathcal{F}, \mathcal{Z}, \varphi) \, dx - \int_{\Omega} \mathcal{H} \cdot \varphi \, dx - \int_{\partial_N \Omega} \bar{\sigma} \cdot \varphi \, ds, \quad (1)$$

in which  $A : \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}$  is a collection of internal variables. The weak form of the problem is obtained by minimizing the energy functional  $\Phi$  over the Sobolev space  $\mathcal{S}$  that is composed of all functions that are square integrable and have square integrable first derivatives. Define

$$\mathcal{S} := \{ \varphi \in H^1(\Omega) : \varphi|_{\partial_D \Omega} = \bar{\varphi} \}, \quad (2)$$

and

$$\mathcal{V} := \{ \xi \in H^1(\Omega) : \xi|_{\partial_D \Omega} = 0 \text{ and } \xi|_{\partial_N \Omega} = 0 \}, \quad (3)$$

where  $\xi \in \mathcal{V}$  is a test function. The potential energy is minimized if and only if  $\Phi[\varphi] \leq \Phi[\varphi + \xi]$  for all  $\xi \in \mathcal{V}$  and  $\xi|_{\partial_D \Omega} = \bar{\varphi}$ . It is straightforward to show that the minimizer of  $\Phi$  is the mapping  $\varphi \in \mathcal{S}$  that satisfies

$$\langle \delta \Phi[\varphi], \xi \rangle = \int_{\Omega} P : \text{sym}(\xi) \, dx - \int_{\Omega} \text{div}(\xi) \cdot \mathcal{H} \, dx - \int_{\partial_N \Omega} \bar{\sigma} \cdot \xi \, ds = 0, \quad (4)$$

where  $P := \partial A / \partial \mathcal{F}$  denotes the first Piola-Kirchhoff stress. The Euler-Lagrange equation corresponding to the variational formulation (4) is

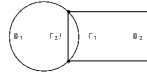


Figure 1: Two domains  $\Omega_1$  and  $\Omega_2$  with corresponding boundaries  $\Gamma_1$  and  $\Gamma_2$  used by the Schwarz alternating method.

Let  $\Omega = \Omega_1 \cup \Omega_2$ ,  $\Gamma = \Gamma_1 \cup \Gamma_2$  and  $\partial \Omega = \partial \Omega_1 \cup \partial \Omega_2$ . Introduce the following definitions for each subdomain:

- Closure  $\bar{\Omega}_i = \Omega_i \cup \Gamma_i$ .
- Dirichlet boundary  $\partial_D \Omega_i = \partial \Omega_i \cap \bar{\Omega}_i$ .
- Neumann boundary  $\partial_N \Omega_i = \partial \Omega_i \setminus \partial_D \Omega_i$ .
- Schwarz boundary  $\Gamma_i = \partial \Omega_i \cap \Omega_j$ .

Note that with these definitions we get  $\partial \Omega_1 = \partial \Omega_1 \cup \Gamma_1$ ,  $\partial \Omega_2 = \partial \Omega_2 \cup \Gamma_2$  and  $\partial \Omega = \partial \Omega_1 \cup \partial \Omega_2$ . Now define the spaces

$$\mathcal{S}_i := \{ \varphi \in H^1(\bar{\Omega}_i) : \varphi|_{\partial_D \Omega_i} = \bar{\varphi}_i \text{ and } \varphi|_{\Gamma_i} = 0 \}, \quad (5)$$

and

$$\mathcal{V}_i := \{ \xi \in H^1(\bar{\Omega}_i) : \xi|_{\partial_D \Omega_i} = 0 \text{ and } \xi|_{\Gamma_i} = 0 \}, \quad (6)$$

where the symbol  $\bar{\Omega}_i \cap \Gamma_j$  denotes the projection from the subdomain  $\bar{\Omega}_i$  onto the Schwarz boundary  $\Gamma_j$ . This projection operator plays a central role in the Schwarz alternating method. Its form and implementation are discussed in subsequent sections. For the formal proof sufficient to assure that the operator is able to project fields  $\varphi$  from one subdomain to the Schwarz boundary of the other subdomain.

The Schwarz alternating method solves a sequence of problems on  $\Omega_1$  and  $\Omega_2$ . The solution  $\varphi^k$  of the

$$\begin{aligned}
 1. & \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \Gamma_1, \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \partial \Omega_1 \setminus \Gamma_1, \\
 2. & \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \Gamma_1, \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \partial \Omega_1 \setminus \Gamma_1, \\
 3. & \text{solve } \Phi[\varphi_1] \text{ over } \mathcal{S}_1, \\
 4. & \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \Gamma_1, \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \partial \Omega_1 \setminus \Gamma_1, \\
 5. & \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \Gamma_1, \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \partial \Omega_1 \setminus \Gamma_1, \\
 6. & \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \Gamma_1, \varphi_1^{(k)} = \varphi_1^{(k-1)} \text{ on } \partial \Omega_1 \setminus \Gamma_1, \\
 7. & \text{send } \varphi_1^{(k)} \text{ to } \Omega_2.
 \end{aligned}$$

Algorithm 1: Iterative Schwarz Method.

[35, 34, 4]. Although we do not provide here formal convergence proofs for the remaining variants of the Schwarz method, we offer some numerical results illustrating their convergence in Section 4.

Consider the energy functional  $\Phi_{\varphi}$  defined in (1). We will denote by  $\varphi$  the dual of  $\Phi$  lower bound over  $\mathcal{S}$  that is

$$\varphi := \inf_{\varphi \in \mathcal{S}} \Phi[\varphi]. \quad (8)$$

For  $\varphi_1, \varphi_2 \in \mathcal{S}_1, \mathcal{S}_2$  with corresponding average  $\bar{\varphi}_1, \bar{\varphi}_2$ . The proof of the convergence of the Schwarz alternating method requires that the functional  $\Phi[\varphi]$  satisfy the following properties over the space  $\mathcal{S}$  defined in (2):

- $\Phi[\varphi]$  is convex.
- $\Phi[\varphi]$  is Fréchet differentiable with  $\Phi'[\varphi]$  denoting its Fréchet derivative.
- $\Phi[\varphi]$  is strictly convex.
- $\Phi[\varphi]$  is lower semi-continuous.
- $\Phi[\varphi]$  is uniformly continuous on  $K_{\varphi}$ , where

$$K_{\varphi} := \{ \varphi \in \mathcal{S} : \varphi|_{\partial_D \Omega} = \bar{\varphi}, \varphi|_{\partial_N \Omega} = \bar{\sigma} \}. \quad (9)$$

It can be shown that the energy functional  $\Phi[\varphi]$  defined in (1) is strictly convex in  $\mathcal{S}$  (property 1) provided

**Theorem 1.** Assume that the energy functional  $\Phi[\varphi]$  satisfies properties 1–5 above. Consider the Schwarz alternating method of Section 2 defined by (9)–(13) and its equivalent form (39). Then

- (a)  $\Phi[\tilde{\varphi}^{(0)}] \geq \Phi[\tilde{\varphi}^{(1)}] \geq \dots \geq \Phi[\tilde{\varphi}^{(n-1)}] \geq \Phi[\tilde{\varphi}^{(n)}] \geq \dots \geq \Phi[\varphi]$ , where  $\varphi$  is the minimizer of  $\Phi[\varphi]$  over  $\mathcal{S}$ .
- (b) The sequence  $\{\tilde{\varphi}^{(n)}\}$  defined in (39) converges to the minimizer  $\varphi$  of  $\Phi[\varphi]$  in  $\mathcal{S}$ .
- (c) The Schwarz minimum values  $\Phi[\tilde{\varphi}^{(n)}]$  converge monotonically to the minimum value  $\Phi[\varphi]$  in  $\mathcal{S}$  starting from any initial guess  $\tilde{\varphi}^{(0)}$ .

**Remark 1.** By the convexity of  $\Phi[\varphi]$  it follows from the Euler-Lagrange theorem that a unique minimizer to the functional over  $\mathcal{S}$  exists, i.e., the minimization of  $\Phi[\varphi]$  is well-posed.

**Remark 2.** By the Stampacchia theorem, the minimization of  $\Phi[\varphi]$  in  $\mathcal{S}$  is equivalent to finding  $\varphi \in \mathcal{S}$  such that

$$\langle \delta \Phi[\varphi], \xi \rangle \geq 0 \quad (10)$$

for all  $\xi \in \mathcal{S}$ .

**Remark 3.** Recall that the strict convexity property of  $\Phi[\varphi]$  can be written as

$$\langle \delta^2 \Phi[\varphi], \xi \rangle = \langle \delta^2 \Phi[\varphi], \xi \rangle \geq 0 \quad (11)$$

forall  $\xi \in \mathcal{S}$ . From (10), remark that if  $\Phi[\varphi]$  is strictly convex over  $\mathcal{S}$  (i.e.,  $\langle \delta^2 \Phi[\varphi], \xi \rangle > 0$ ), we can find  $\alpha > 0$  such that  $\langle \delta^2 \Phi[\varphi], \xi \rangle \geq \alpha \|\xi\|$  holds

$$\langle \delta^2 \Phi[\varphi], \xi \rangle \geq \alpha \|\xi\|, \quad \forall \xi \in \mathcal{S}. \quad (12)$$

**Remark 4.** By property 5, the uniform continuity of  $\Phi[\varphi]$ , there exists a modulus of continuity  $\omega : [0, \infty) \rightarrow [0, \infty)$  such that

$$\langle \delta^2 \Phi[\varphi], \xi \rangle \leq \omega(\|\xi\|), \quad \forall \xi \in \mathcal{S}. \quad (13)$$

forall  $\xi \in \mathcal{S}$ . By definition,  $\omega(t) \rightarrow 0$  as  $t \rightarrow 0$ .

**Remark 5.** It was shown in [35] that in the case (12),  $\forall \xi \in \mathcal{S}$ , there exist  $\zeta_1 \in \mathcal{S}$  and  $\zeta_2 \in \mathcal{S}$  such that

$$\langle \delta^2 \Phi[\varphi], \zeta_1 \rangle \leq \alpha \|\zeta_1\|, \quad (14)$$

and

$$\langle \delta^2 \Phi[\varphi], \zeta_2 \rangle \geq \alpha \|\zeta_2\|. \quad (15)$$

for some  $C_0 > 0$  independent of  $\varphi$ .

**Remark 6.** Note that (14) can be written as

$$\langle \delta^2 \Phi[\varphi], \zeta_1 \rangle \leq 0, \quad \forall \zeta_1 \in \mathcal{S}. \quad (16)$$

for  $\zeta_1 \in \{ \zeta_1 \in \mathcal{S} : \langle \delta^2 \Phi[\varphi], \zeta_1 \rangle \leq 0 \}$ . Recall from (12) the modulus of continuity  $\omega$ . This is due to the uniqueness of the solution to such non-convex problem over  $\mathcal{S}$ , and the definition of  $\omega$  as the modulus of  $\Phi[\varphi]$  over  $\mathcal{S}$ .

**Remark 7.** For  $\zeta_1 \in \mathcal{S}_1$  and  $\zeta_2 \in \mathcal{S}_2$ . By Remark 5, there exist  $\zeta_1 \in \mathcal{S}_1$  and  $\zeta_2 \in \mathcal{S}_2$  such that

$$\langle \delta^2 \Phi[\varphi], \zeta_1 \rangle = \langle \delta^2 \Phi[\varphi], \zeta_1 \rangle + \langle \delta^2 \Phi[\varphi], \zeta_2 \rangle. \quad (17)$$

Applying (15) and also (16) in (17) leads to

$$\langle \delta^2 \Phi[\varphi], \zeta_1 \rangle = \langle \delta^2 \Phi[\varphi], \zeta_1 \rangle + \langle \delta^2 \Phi[\varphi], \zeta_2 \rangle \leq \langle \delta^2 \Phi[\varphi], \zeta_1 \rangle + \langle \delta^2 \Phi[\varphi], \zeta_2 \rangle. \quad (18)$$

and substituting (16) into (18) we finally obtain that

$$\langle \delta^2 \Phi[\varphi], \zeta_1 \rangle \leq \langle \delta^2 \Phi[\varphi], \zeta_1 \rangle + \langle \delta^2 \Phi[\varphi], \zeta_2 \rangle. \quad (19)$$

$\forall \zeta_1 \in \mathcal{S}_1$ .

**Remark 8.** The part (a) of Theorem 1, recall the definition of geometric convergence:

$$\| \varphi^{(k)} - \varphi^{(k-1)} \| \leq C^k, \quad \forall k \in \mathbb{N}. \quad (20)$$

$\forall n \in \{0, 1, 2, \dots\}$ , for some  $C > 0$ , where

$$C_n := \| \varphi^{(n)} - \varphi^{(n-1)} \|, \quad \forall n \in \mathbb{N}. \quad (21)$$

**Remark 9.** Recall from the definition of convexity that if  $\delta^2 \Phi[\varphi]$  is Lipschitz continuous in  $\mathcal{S}$  near  $\varphi$ , then there exists a constant  $K > 0$  such that

$$\| \delta^2 \Phi[\varphi] - \delta^2 \Phi[\varphi] \| \leq K \| \varphi - \varphi \|, \quad (22)$$

Considering that  $\Phi[\varphi] = \Phi[\varphi]$  as the minimizer of  $\Phi[\varphi]$ , (22) is equivalent to

$$\| \delta^2 \Phi[\varphi] \| \leq K \| \varphi - \varphi \|. \quad (23)$$

**Proof of Theorem 1.**

*Proof of (a).* Let  $\varphi^{(k)} = \arg \min_{\varphi \in \mathcal{S}_k} \Phi[\varphi]$ . By (10),  $\varphi^{(k)} \in \mathcal{S}_k$ . Let  $\varphi$  be the minimizer of  $\Phi[\varphi]$  over  $\mathcal{S}$  and suppose  $\Phi[\varphi] > \Phi[\varphi^{(k)}]$ . But this is a contradiction, since we can take  $\varphi = \varphi^{(k)}$ . Hence, it cannot be that  $\Phi[\varphi^{(k)}] < \Phi[\varphi]$  where  $\varphi^{(k)} = \arg \min_{\varphi \in \mathcal{S}_k} \Phi[\varphi]$ . It follows by induction that

$$\Phi[\varphi^{(k)}] \leq \Phi[\varphi^{(k-1)}] \leq \dots \leq \Phi[\varphi]. \quad (24)$$

for  $n \in \{1, 2, 3, \dots\}$ . Now let  $\varphi$  be the minimizer of  $\Phi[\varphi]$  over  $\mathcal{S}$ . Since the problem is well-posed in  $\mathcal{S}$ , we have  $\Phi[\varphi] = \Phi[\varphi]$ . Hence,  $\Phi[\varphi] = \Phi[\varphi^{(k)}]$ , for all  $k \in \{1, 2, 3, \dots\}$ .  $\square$

$$\| \varphi^{(k)} - \varphi^{(k-1)} \| \leq C^k. \quad (25)$$

from which we can conclude that  $\varphi^{(k)} - \varphi^{(k-1)} \rightarrow 0$  as  $n \rightarrow \infty$ .

We now show that  $\varphi^{(k)}$  converges to  $\varphi$ , the minimizer of  $\Phi[\varphi]$  on  $\mathcal{S}$ . By (15) with  $\varphi_1 = \varphi$  and  $\varphi_2 = \varphi^{(k)}$ , we have

$$\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \leq \frac{1}{2} \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle + \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle. \quad (26)$$

Since  $\varphi$  is the minimizer of  $\Phi[\varphi]$  by (10) we have that  $\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \geq 0$ . It follows that

$$\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \leq \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle + \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle. \quad (27)$$

Substituting (27) into (26) we have

$$\| \varphi - \varphi^{(k)} \|^2 \leq \frac{1}{2} \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle. \quad (28)$$

Now by (22) (Remark 9)

$$\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \leq K \| \varphi - \varphi^{(k)} \| = K \| \varphi^{(k)} - \varphi \|. \quad (29)$$

Substituting (29) into (28) leads to

$$\| \varphi - \varphi^{(k)} \|^2 \leq \frac{K}{2} \| \varphi^{(k)} - \varphi \|. \quad (30)$$

Applying the uniform continuity hypothesis (14), we obtain

$$\| \varphi^{(k)} - \varphi \| \leq \frac{K}{2} \| \varphi^{(k)} - \varphi \|. \quad (31)$$

By (31),  $\| \varphi^{(k)} - \varphi \| \rightarrow 0$  as  $n \rightarrow \infty$ . From this we obtain the result, namely that  $\varphi^{(k)} \rightarrow \varphi$  as  $n \rightarrow \infty$ .  $\square$

*Proof of (b).* This follows immediately from (a) and (16).

*Proof of (c).* By (15), for large enough  $n$ , there exists some  $C_1 > 0$  independent of  $n$  such that

$$\| \varphi^{(k)} - \varphi \| \leq \frac{C_1}{2} \| \varphi^{(k)} - \varphi \|. \quad (32)$$

Let us choose  $C_1$  such that  $C_1 > \frac{1}{2} K$ , where  $K$  is the Lipschitz continuity constant in (22). Combining (32) with (31) leads to

$$\frac{1}{2} \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \geq \frac{C_1}{2} \| \varphi^{(k)} - \varphi \|. \quad (33)$$

**Remark 10.**  $\varphi_k = \varphi^{(k-1)}|_{\Omega_1}$  for  $\varphi^{(k)} \in \mathcal{S}_2$ ,  $\varphi_k = \varphi^{(k-1)}|_{\Omega_2}$  for  $\varphi^{(k)} \in \mathcal{S}_1$ .  $\square$

**Theorem 1.** Assume that the energy functional  $\Phi[\varphi]$  satisfies properties 1–5 above. Consider the Schwarz alternating method of Section 2 defined by (9)–(13) and its equivalent form (39). Then

(a)  $\Phi[\tilde{\varphi}^{(0)}] \geq \Phi[\tilde{\varphi}^{(1)}] \geq \dots \geq \Phi[\tilde{\varphi}^{(n-1)}] \geq \Phi[\tilde{\varphi}^{(n)}] \geq \dots \geq \Phi[\varphi]$ , where  $\varphi$  is the minimizer of  $\Phi[\varphi]$  over  $\mathcal{S}$ .

(b) The sequence  $\{\tilde{\varphi}^{(n)}\}$  defined in (39) converges to the minimizer  $\varphi$  of  $\Phi[\varphi]$  in  $\mathcal{S}$ .

(c) The Schwarz minimum values  $\Phi[\tilde{\varphi}^{(n)}]$  converge monotonically to the minimum value  $\Phi[\varphi]$  in  $\mathcal{S}$  starting from any initial guess  $\tilde{\varphi}^{(0)}$ .

(d) If  $\varphi \in \mathcal{S}$  is Lipschitz continuous in a neighborhood of  $\varphi$ , then the sequence  $\{\tilde{\varphi}^{(n)}\}$  converges geometrically to the minimizer  $\varphi$ .  $\square$

*Proof.* See Appendix A.  $\square$

Finally, while most of works cited above present their analysis for the specific case of one subdomain, extension to multiple subdomains is a general straightforward. The case of multiple subdomains was consistently explicitly in [16] (Suk, Baskin, and Li-Shan), [14] (Shan and Li-Shan), and [15] (Suk).

## 4 Numerical Examples

In this section, we present general examples of the behavior of the Schwarz alternating method for two different implementations. First, we briefly describe the two implementations, one in MATLAB and the other in C. We then show that  $\varphi^{(k)}$  converges to  $\varphi$ , the minimizer of  $\Phi[\varphi]$  on  $\mathcal{S}$ . Next, we discuss the convergence speed through numerical examples. Then, we continue with four examples that demonstrate different features of the Schwarz alternating method and our implementations. The first example, a one-dimensional singular bar, is used to demonstrate the behavior of the first Schwarz variant of Section 2. The second example, a cuboid body of square base, aims to study the effect of the size of the overlap region on the convergence of the method. The objective of the third example is to analyze the numerical error in a two-subdomain

$$\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \leq \frac{1}{2} \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle + \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle. \quad (34)$$

since  $\varphi$  is the minimizer of  $\Phi[\varphi]$  by (10) we have that  $\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \geq 0$ . It follows that

$$\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \leq \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle + \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle. \quad (35)$$

Substituting (35) into (34) we have

$$\| \varphi - \varphi^{(k)} \|^2 \leq \frac{1}{2} \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle. \quad (36)$$

Now by (22) (Remark 9)

$$\langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \leq K \| \varphi - \varphi^{(k)} \| = K \| \varphi^{(k)} - \varphi \|. \quad (37)$$

Substituting (37) into (36) leads to

$$\| \varphi - \varphi^{(k)} \|^2 \leq \frac{K}{2} \| \varphi^{(k)} - \varphi \|. \quad (38)$$

Applying the uniform continuity hypothesis (14), we obtain

$$\| \varphi^{(k)} - \varphi \| \leq \frac{K}{2} \| \varphi^{(k)} - \varphi \|. \quad (39)$$

By (39),  $\| \varphi^{(k)} - \varphi \| \rightarrow 0$  as  $n \rightarrow \infty$ . From this we obtain the result, namely that  $\varphi^{(k)} \rightarrow \varphi$  as  $n \rightarrow \infty$ .  $\square$

*Proof of (b).* This follows immediately from (a) and (16).

*Proof of (c).* By (15), for large enough  $n$ , there exists some  $C_1 > 0$  independent of  $n$  such that

$$\| \varphi^{(k)} - \varphi \| \leq \frac{C_1}{2} \| \varphi^{(k)} - \varphi \|. \quad (40)$$

Let us choose  $C_1$  such that  $C_1 > \frac{1}{2} K$ , where  $K$  is the Lipschitz continuity constant in (22). Combining (40) with (39) leads to

$$\frac{1}{2} \langle \delta^2 \Phi[\varphi], \varphi - \varphi^{(k)} \rangle \geq \frac{C_1}{2} \| \varphi^{(k)} - \varphi \|. \quad (41)$$

Applying the uniform continuity hypothesis (14), we obtain

$$\| \varphi^{(k)} - \varphi \| \leq \frac{K}{2} \| \varphi^{(k)} - \varphi \|. \quad (42)$$

By (42),  $\| \varphi^{(k)} - \varphi \| \rightarrow 0$  as  $n \rightarrow \infty$ . From this we obtain the result, namely that  $\varphi^{(k)} \rightarrow \varphi$  as  $n \rightarrow \infty$ .  $\square$





- Like for quasistatics, dynamic alternating Schwarz method converges provided each single-domain problem is **well-posed** and **overlap region** is **non-empty**, under some **conditions** on  $\Delta t$ .
- **Well-posedness** for the dynamic problem requires that action functional  $S[\boldsymbol{\varphi}] := \int_I \int_{\Omega} L(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) dV dt$  be **strictly convex** or **strictly concave**, where  $L(\boldsymbol{\varphi}, \dot{\boldsymbol{\varphi}}) := T(\dot{\boldsymbol{\varphi}}) + V(\boldsymbol{\varphi})$  is the Lagrangian.
  - This is studied by looking at its second variation  $\delta^2 S[\boldsymbol{\varphi}_h]$
- We can show assuming a **Newmark** time-integration scheme that for the **fully-discrete** problem:

$$\delta^2 S[\boldsymbol{\varphi}_h] = \mathbf{x}^T \left[ \frac{\gamma^2}{(\beta \Delta t)^2} \mathbf{M} - \mathbf{K} \right] \mathbf{x}$$

- $\delta^2 S[\boldsymbol{\varphi}_h]$  can always be made positive by choosing a **sufficiently small**  $\Delta t$
- Numerical experiments reveal that  $\Delta t$  requirements for **stability/accuracy** typically lead to automatic satisfaction of this bound.

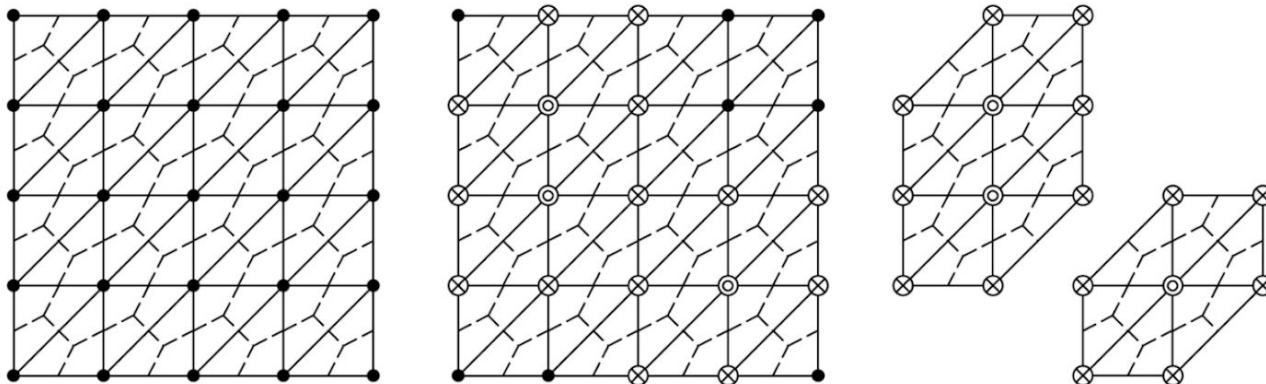
# Energy-Conserving Sampling and Weighting (ECSW)



- **Project-then-approximate** paradigm (as opposed to approximate-then-project)

$$\begin{aligned} r_k(q_k, t) &= W^T r(\tilde{u}, t) \\ &= \sum_{e \in \mathcal{E}} W^T L_e^T r_e(L_e \tilde{u}, t) \end{aligned}$$

- $L_e \in \{0,1\}^{d_e \times N}$  where  $d_e$  is the **number of degrees of freedom** associated with each mesh element (this is in the context of meshes used in first-order hyperbolic problems where there are  $N_e$  mesh elements)
- $L_{e+} \in \{0,1\}^{d_e \times N}$  selects degrees of freedom necessary for **flux reconstruction**
- Equality can be **relaxed**



Augmented reduced mesh:  $\odot$  represents a selected node attached to a selected element; and  $\otimes$  represents an added node to enable the full representation of the computational stencil at the selected node/element

# ECSW: Generating the Reduced Mesh and Weights



- Using a subset of the same snapshots  $u_i, i \in 1, \dots, n_h$  used to generate the **state basis**  $V$ , we can train the reduced mesh
- Snapshots are first **projected** onto their associated basis and then **reconstructed**

$$c_{se} = W^T L_e^T r_e \left( L_e + \left( u_{ref} + V V^T (u_s - u_{ref}) \right), t \right) \in \mathbb{R}^n$$

$$d_s = r_k(\tilde{u}, t) \in \mathbb{R}^n, \quad s = 1, \dots, n_h$$

- We can then form the **system**

$$\mathbf{C} = \begin{pmatrix} c_{11} & \dots & c_{1N_e} \\ \vdots & \ddots & \vdots \\ c_{n_h 1} & \dots & c_{n_h N_e} \end{pmatrix}, \quad \mathbf{d} = \begin{pmatrix} d_1 \\ \vdots \\ d_{n_h} \end{pmatrix}$$

- Where  $\mathbf{C}\xi = \mathbf{d}$ ,  $\xi \in \mathbb{R}^{N_e}$ ,  $\xi = \mathbf{1}$  must be the solution
- Further relax the equality to yield **non-negative least-squares problem**:

$$\xi = \arg \min_{x \in \mathbb{R}^n} \|\mathbf{C}x - \mathbf{d}\|_2 \text{ subject to } x \geq \mathbf{0}$$

- Solve the above optimization problem using a **non-negative least squares solver** with an **early termination condition** to **promote sparsity** of the vector  $\xi$