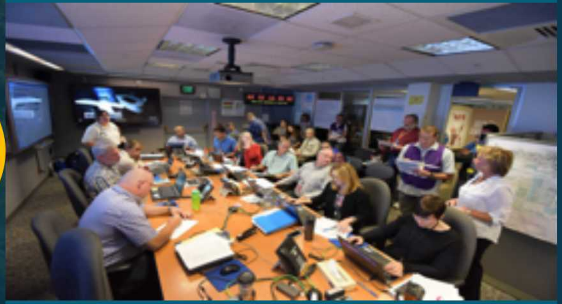
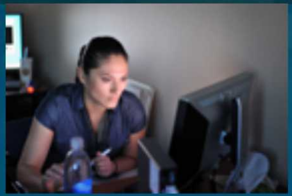
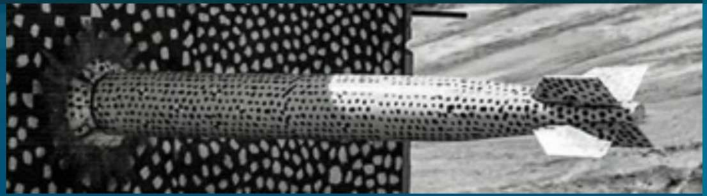
SAND2019-2040C

Domain-decomposition least-squares Petrov-Galerkin (DD-LSPG) nonlinear model reduction







PRESENTED BY

Chi Hoang and Kevin Carlberg

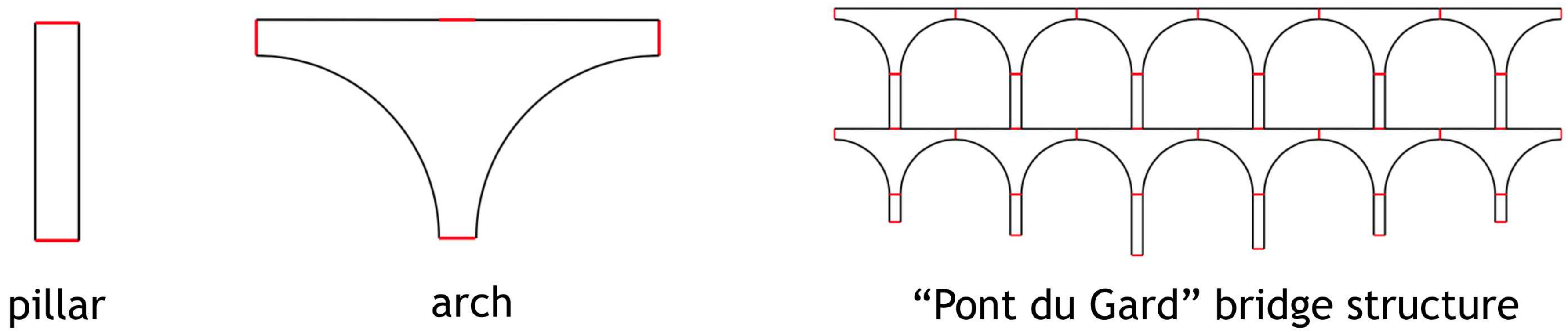
Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Motivation for DDROM

- **Global model reduction**: mature
  - + Proper Orthogonal Decomposition method [Sirovich, 1987; etc]
  - + Reduced Basis method [Prud'Homme et al., 2001; Barrault et al., 2004; Rozza et al., 2008]
  - + Proper Generalized Decomposition method [Ladeveze et al., 2009; Chinesta et al, 2010]

Motivation for DDROM

- **Global model reduction**: mature
  - + Proper Orthogonal Decomposition method [Sirovich, 1987; etc]
  - + Reduced Basis method [Prud'Homme et al., 2001; Barrault et al., 2004; Rozza et al., 2008]
  - + Proper Generalized Decomposition method [Ladeveze et al., 2009; Chinesta et al, 2010]
- **For *decomposable systems***:



(Courtesy from Huynh et al., 2013)

Motivation for DDROM

- **Global model reduction**: mature

  + Proper Orthogonal Decomposition method [Sirovich, 1987; etc]

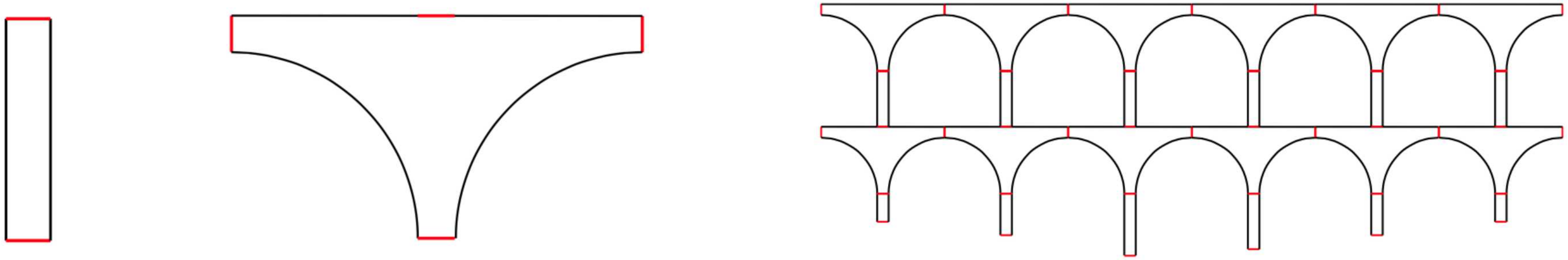
  + Reduced Basis method [Prud'Homme et al., 2001; Barrault et al., 2004; Rozza et al., 2008]

  + Proper Generalized Decomposition method [Ladeveze et al., 2009; Chinesta et al, 2010]

- **Global model reduction applied to *decomposable systems***: ineffective

  - Costly: requires training simulations for (large-scale) full-system

  - Ineffective: each different full-system configuration requires training simulations



pillar arch "Pont du Gard" bridge structure

(Courtesy from Huynh et al., 2013)

Motivation for DDROM

- **Global model reduction**: mature
  + Proper Orthogonal Decomposition method [Sirovich, 1987; etc]
  + Reduced Basis method [Prud'Homme et al., 2001; Barrault et al., 2004; Rozza et al., 2008]
  + Proper Generalized Decomposition method [Ladeveze et al., 2009; Chinesta et al, 2010]
- **Decomposable engineering systems**: ineffective
  - Costly: requires training simulations for full-system
  - Ineffective: each different full-system configuration requires training simulations
- **Main idea**: "divide and conquer" model reduction for decomposable systems
  + Divide: reduced bases constructed *separately* on each subdomain/component
  + Conquer: compute global solution using non-overlapping domain decomposition
  + General: full-system can be assembled in *arbitrary* ways
  + Weak compatibility: mitigate the need for matching meshes, freedom for interface RB
  + Applicable to nonlinear systems and multiple different solvers

Literature reviews for DDROM

Not a comprehensive list

- **Linear parameterized PDEs:** quite active topic

  + *Strong constraints*

    SCRBE [Huynh et al., 2013; Huynh et al., 2015]: heat conduction, structural analysis

    RDF [Iapichino et al., 2016]: heat conduction

  + *Weak constraints*

    RBEM [Maday et al., 2002]: potential flow analysis

    RBHM [Iapichino et al., 2012]: Stokes equation, cardiovascular networks

- **Nonlinear parameterized PDEs:** hybrid FOM-ROM approach

  Not much has been done

  + *Strong constraints*

    [Kerfriden et al., 2013]: nonlinear fracture mechanic problems

  + *Weak constraints*

    [Baiges et al., 2013]: incompressible Navier-Stokes equation with hyper-reduction

    [Corigliano et al., 2015]: elastic-plastic structural dynamic problems

Goals

+ Applicable to nonlinear systems

+ Enable hyper-reduction

+ Subdomains ROMs constructed independently (tailored bases, hyper-reduction)

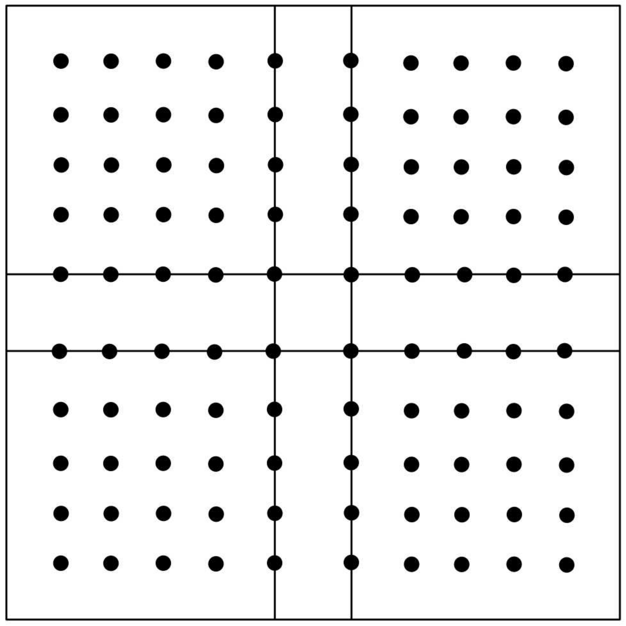
+ Benefits of weak compatibility

+ Enable different kinds of reduced bases on the interfaces: port, skeleton, interface, subdomain

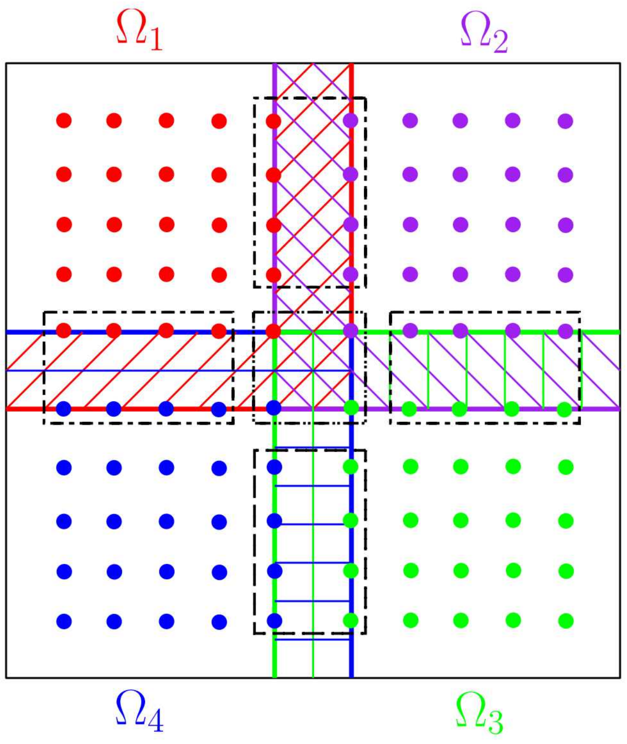
+ Various parallel numerical solvers: primal-dual monolithic*, primal-dual Schur, primal monolithic, primal Schur, nonlinear primal

+ Assembly and solve stages expose parallelism

Problem settings



Ω

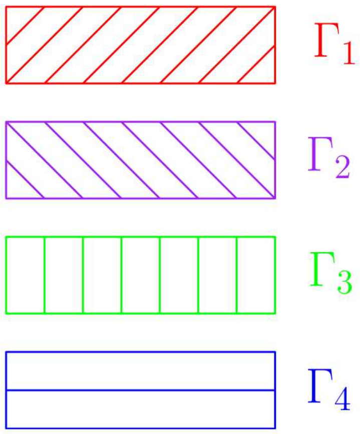


$$x_i^{\Omega} := P_i^{\Omega} x \in \mathbb{R}^{n_i^{\Omega}}$$

$$x_i^{\Gamma} := P_i^{\Gamma} x \in \mathbb{R}^{n_i^{\Gamma}}$$

$$x_i := (x_i^{\Omega}, x_i^{\Gamma})$$

$$[P_i^j]^T x_i^{\Gamma} = [P_l^j]^T x_l^{\Gamma},$$
$$i, l \in P(j)$$



Global FOM approximation

$$\boldsymbol{r}(\boldsymbol{x}) = \boldsymbol{0}$$

with residual $\boldsymbol{r} : \mathbb{R}^n \to \mathbb{R}^n$

and state $\boldsymbol{x} \in \mathbb{R}^n$

DDFOM re-formulation:

$$\boldsymbol{r}(\boldsymbol{x}) = \sum_{i=1}^{n_\Omega} [\boldsymbol{P}_i^{\boldsymbol{r}}]^T \boldsymbol{r}_i(\boldsymbol{P}_i^{\Omega} \boldsymbol{x}, \boldsymbol{P}_i^{\Gamma} \boldsymbol{x}),$$

$$\forall \boldsymbol{x} \in \mathbb{R}^n$$

$$\boldsymbol{r}_i(\boldsymbol{x}_i^{\Omega}, \boldsymbol{x}_i^{\Gamma}) = 0, \quad i = 1, \ldots, n_\Omega$$

$$\text{subject to} \sum_{i=1}^{n_\Omega} \bar{\boldsymbol{A}}_i \boldsymbol{x}_i^{\Gamma} = 0,$$

Reduced order models

- **DDROM approximation:**

Introduce reduced bases: $\boldsymbol{\Phi}_i^{\Omega} \in \mathbb{R}^{n_i^{\Omega} \times p_i^{\Omega}}, \boldsymbol{\Phi}_i^{\Gamma} \in \mathbb{R}^{n_i^{\Gamma} \times p_i^{\Gamma}}, i = 1, \ldots, n_{\Omega}$

Solution approximation: $\boldsymbol{x}_i \approx \tilde{\boldsymbol{x}}_i = (\tilde{\boldsymbol{x}}_i^{\Omega}, \tilde{\boldsymbol{x}}_i^{\Gamma}) = (\boldsymbol{\Phi}_i^{\Omega} \hat{\boldsymbol{x}}_i^{\Omega}, \boldsymbol{\Phi}_i^{\Gamma} \hat{\boldsymbol{x}}_i^{\Gamma})$

$$\underset{(\hat{\boldsymbol{x}}_1^{\Omega},\ldots,\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}),(\hat{\boldsymbol{x}}_1^{\Gamma},\ldots,\hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma})}{\text{minimize}} \frac{1}{2} \sum_{i=1}^{n_\Omega} \|\boldsymbol{r}_i(\boldsymbol{\Phi}_i^{\Omega} \hat{\boldsymbol{x}}_i^{\Omega}, \boldsymbol{\Phi}_i^{\Gamma} \hat{\boldsymbol{x}}_i^{\Gamma})\|_2^2$$

$$\text{subject to} \sum_{i=1}^{n_\Omega} \boldsymbol{A}_i \boldsymbol{\Phi}_i^{\Gamma} \hat{\boldsymbol{x}}_i^{\Gamma} = 0$$

- **DDGNAT approximation (hyper-reduction)**

$$\underset{(\hat{\boldsymbol{x}}_1^{\Omega},\ldots,\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}),(\hat{\boldsymbol{x}}_1^{\Gamma},\ldots,\hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma})}{\text{minimize}} \frac{1}{2} \sum_{i=1}^{n_\Omega} \|\boldsymbol{\Phi}_i^{r} (\boldsymbol{Z}_i \boldsymbol{\Phi}_i^{r})^{+} \boldsymbol{Z}_i \boldsymbol{r}_i(\boldsymbol{\Phi}_i^{\Omega} \hat{\boldsymbol{x}}_i^{\Omega}, \boldsymbol{\Phi}_i^{\Gamma} \hat{\boldsymbol{x}}_i^{\Gamma})\|_2^2$$

$$\text{subject to} \sum_{i=1}^{n_\Omega} \boldsymbol{A}_i \boldsymbol{\Phi}_i^{\Gamma} \hat{\boldsymbol{x}}_i^{\Gamma} = 0$$

Reduced order models…

- **Interface basis types**

"Port" bases	"Skeleton" bases	"Interface" bases	"Subdomain" bases
1. Collect snapshots for subdomain ports. 2. Compute separate SVD for each port to form port bases. 3. Combine the port bases to form interface bases.	1. Isolate global snapshots to skeleton DOFs. 2. Compute SVD for the skeleton snapshots to create skeleton bases. 3. Isolate the skeleton bases to subdomain interfaces. 4. Orthogonalize the above to form interface bases.	1. Collect snapshots for subdomain interfaces. 2. Compute separate SVD for each interface to create interface bases.	1. Collect snapshots for subdomain DOFs. 2. Compute separate SVD to create subdomain bases. 3. Isolate to interface bases.

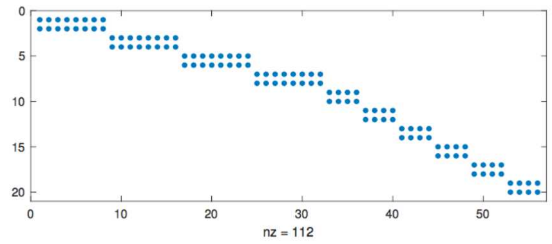
- **Constraint types**

  + Strong constraint: $\boldsymbol{A}_i = \bar{\boldsymbol{A}}_i$

  + Weak constraint: $\boldsymbol{A}_i = \boldsymbol{C}\bar{\boldsymbol{A}}_i$

$$i = 1, \ldots, n_\Omega$$

$\boldsymbol{C}$



Interface basis types: pros & cons

- **Port**: has associated global ROM when strong constraints are used

  \+ Can use both global training & subdomain training

  \- Total number of interface reduced bases are generally large

- **Skeleton**: has associated global ROM when strong constraints are used

  \+ Requires many fewer number of interface reduced bases (w.r.t. “port” type)

  \- Requires global training

- **Interface & subdomain**: do NOT have associated global ROM due to possible solution mismatch at interfaces

  \+ Can use both global training & subdomain training

  \+ Requires many fewer number of interface reduced basis (w.r.t. “port” type)

  \- Requires weak constraints

Sequential Quadratic Programming method

- **Lagrangian:**

$$L(\hat{\boldsymbol{x}}_1^\Omega, \hat{\boldsymbol{x}}_1^\Gamma, \ldots, \hat{\boldsymbol{x}}_{n_\Omega}^\Omega, \hat{\boldsymbol{x}}_{n_\Omega}^\Gamma, \boldsymbol{\lambda}) := \frac{1}{2}\sum_{i=1}^{n_\Omega} \|\boldsymbol{r}_i(\boldsymbol{\Phi}_i^\Omega \hat{\boldsymbol{x}}_i^\Omega, \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma)\|_2^2 + \sum_{i=1}^{n_\Omega} \boldsymbol{\lambda}^T \boldsymbol{A}_i \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma,$$

- **Necessary optimality conditions (KKT conditions):**

$$\hat{\boldsymbol{r}}_i^\Omega(\hat{\boldsymbol{x}}_i^\Omega, \hat{\boldsymbol{x}}_i^\Gamma) := (\boldsymbol{\Phi}_i^\Omega)^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^\Omega}(\boldsymbol{\Phi}_i^\Omega \hat{\boldsymbol{x}}_i^\Omega, \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma)^T \boldsymbol{r}_i(\boldsymbol{\Phi}_i^\Omega \hat{\boldsymbol{x}}_i^\Omega, \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma) = 0,$$

$$\hat{\boldsymbol{r}}_i^\Gamma(\hat{\boldsymbol{x}}_i^\Omega, \hat{\boldsymbol{x}}_i^\Gamma) := (\boldsymbol{\Phi}_i^\Gamma)^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^\Gamma}(\boldsymbol{\Phi}_i^\Omega \hat{\boldsymbol{x}}_i^\Omega, \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma)^T \boldsymbol{r}_i(\boldsymbol{\Phi}_i^\Omega \hat{\boldsymbol{x}}_i^\Omega, \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma) + (\boldsymbol{\Phi}_i^\Gamma)^T \boldsymbol{A}_i^T \boldsymbol{\lambda} = 0,$$

$$\sum_{i=1}^{n_\Omega} \boldsymbol{A}_i \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma = 0, \quad i = 1, \ldots, n_\Omega.$$

Note: The formulations on this and next 3 slides are for port, skeleton and interface bases.
Subdomain bases formulations are slightly different.

Primal-dual monolithic solver

Use Gauss-Newton approximation, one SQP iteration is defined as

$$
\begin{bmatrix}
\boldsymbol{H}_1^{\Omega\Omega}(\hat{\boldsymbol{x}}_1^{\Omega}, \hat{\boldsymbol{x}}_1^{\Gamma}) & \boldsymbol{H}_1^{\Omega\Gamma}(\hat{\boldsymbol{x}}_1^{\Omega}, \hat{\boldsymbol{x}}_1^{\Gamma}) & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\boldsymbol{H}_1^{\Gamma\Omega}(\hat{\boldsymbol{x}}_1^{\Omega}, \hat{\boldsymbol{x}}_1^{\Gamma}) & \boldsymbol{H}_1^{\Gamma\Gamma}(\hat{\boldsymbol{x}}_1^{\Omega}, \hat{\boldsymbol{x}}_1^{\Gamma}) & \cdots & \mathbf{0} & \mathbf{0} & (\boldsymbol{\Phi}_1^{\Gamma})^T \boldsymbol{A}_1^T \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
\mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{H}_{n_\Omega}^{\Omega\Omega}(\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}, \hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma}) & \boldsymbol{H}_{n_\Omega}^{\Omega\Gamma}(\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}, \hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma}) & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{H}_{n_\Omega}^{\Gamma\Omega}(\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}, \hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma}) & \boldsymbol{H}_{n_\Omega}^{\Gamma\Gamma}(\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}, \hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma}) & (\boldsymbol{\Phi}_{n_\Omega}^{\Gamma})^T \boldsymbol{A}_{n_\Omega}^T \\
\mathbf{0} & \boldsymbol{A}_1 \boldsymbol{\Phi}_1^{\Gamma} & \cdots & \mathbf{0} & \boldsymbol{A}_{n_\Omega} \boldsymbol{\Phi}_{n_\Omega}^{\Gamma} & \mathbf{0}
\end{bmatrix}
\begin{bmatrix}
\boldsymbol{p}_1^{\Omega} \\ \boldsymbol{p}_1^{\Gamma} \\ \vdots \\ \boldsymbol{p}_{n_\Omega}^{\Omega} \\ \boldsymbol{p}_{n_\Omega}^{\Gamma} \\ \boldsymbol{p}^{\boldsymbol{\lambda}}
\end{bmatrix}
=
-\begin{bmatrix}
\hat{\boldsymbol{r}}_1^{\Omega}(\hat{\boldsymbol{x}}_1^{\Omega}, \hat{\boldsymbol{x}}_1^{\Gamma}) \\
\hat{\boldsymbol{r}}_1^{\Gamma}(\hat{\boldsymbol{x}}_1^{\Omega}, \hat{\boldsymbol{x}}_1^{\Gamma}) \\
\vdots \\
\hat{\boldsymbol{r}}_{n_\Omega}^{\Omega}(\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}, \hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma}) \\
\hat{\boldsymbol{r}}_{n_\Omega}^{\Gamma}(\hat{\boldsymbol{x}}_{n_\Omega}^{\Omega}, \hat{\boldsymbol{x}}_{n_\Omega}^{\Gamma}) \\
\sum_{i=1}^{n_\Omega} \boldsymbol{A}_i \boldsymbol{\Phi}_i^{\Gamma} \hat{\boldsymbol{x}}_i^{\Gamma}
\end{bmatrix}
$$

Primal-dual monolithic solver...

where

$$\boldsymbol{H}_i^{\Omega\Omega}(\hat{\boldsymbol{x}}_i^{\Omega},\hat{\boldsymbol{x}}_i^{\Gamma}) := (\boldsymbol{\Phi}_i^{\Omega})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Omega}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Omega}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})\boldsymbol{\Phi}_i^{\Omega}$$

$$\boldsymbol{H}_i^{\Omega\Gamma}(\hat{\boldsymbol{x}}_i^{\Omega},\hat{\boldsymbol{x}}_i^{\Gamma}) := (\boldsymbol{\Phi}_i^{\Omega})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Omega}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Gamma}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})\boldsymbol{\Phi}_i^{\Gamma}$$

$$\boldsymbol{H}_i^{\Gamma\Gamma}(\hat{\boldsymbol{x}}_i^{\Omega},\hat{\boldsymbol{x}}_i^{\Gamma}) := (\boldsymbol{\Phi}_i^{\Gamma})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Gamma}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Gamma}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})\boldsymbol{\Phi}_i^{\Gamma}$$

$$\boldsymbol{H}_i^{\Gamma\Omega}(\hat{\boldsymbol{x}}_i^{\Omega},\hat{\boldsymbol{x}}_i^{\Gamma}) := (\boldsymbol{\Phi}_i^{\Gamma})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Gamma}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})^T \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Omega}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega},\boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})\boldsymbol{\Phi}_i^{\Omega}.$$

Update

$$\hat{\boldsymbol{x}}_i^{\Gamma} \leftarrow \hat{\boldsymbol{x}}_i^{\Gamma} + \alpha \boldsymbol{p}_i^{\Gamma}, \quad i = 1,\ldots,n_{\Omega}$$

$$\hat{\boldsymbol{x}}_i^{\Omega} \leftarrow \hat{\boldsymbol{x}}_i^{\Omega} + \alpha \boldsymbol{p}_i^{\Omega}, \quad i = 1,\ldots,n_{\Omega}$$

$$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \alpha \boldsymbol{p}^{\boldsymbol{\lambda}},$$

Primal-dual monolithic solver: online stage

Algorithm 1: Assembling procedure of primal–dual monolithic solver in parallel

1: Update the ROM state;

2: Compute residuals $\boldsymbol{r}_i(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega}, \boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})$ and Jacobians $\frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Omega}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega}, \boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})$, $\frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{x}_i^{\Gamma}}(\boldsymbol{\Phi}_i^{\Omega}\hat{\boldsymbol{x}}_i^{\Omega}, \boldsymbol{\Phi}_i^{\Gamma}\hat{\boldsymbol{x}}_i^{\Gamma})$ from each subdomain;

3: Compute all terms in [Stationary condition] from each subdomain;

4: "Stamping" all terms into linear system;

Algorithm 1: Solving procedure of primal–dual monolithic solver

5: Solve the linear system;

6: Extract search directions $\boldsymbol{p}_i^{\Omega}$, $\boldsymbol{p}_i^{\Gamma}$, $\boldsymbol{p}^{\boldsymbol{\lambda}}$;

7: Update solutions;

*Offline stage is performed in advance to create all $\boldsymbol{\Phi}_{\mathbf{i}}^{\boldsymbol{\Omega}}, \boldsymbol{\Phi}_{\mathbf{i}}^{\boldsymbol{\Gamma}}, A_i$

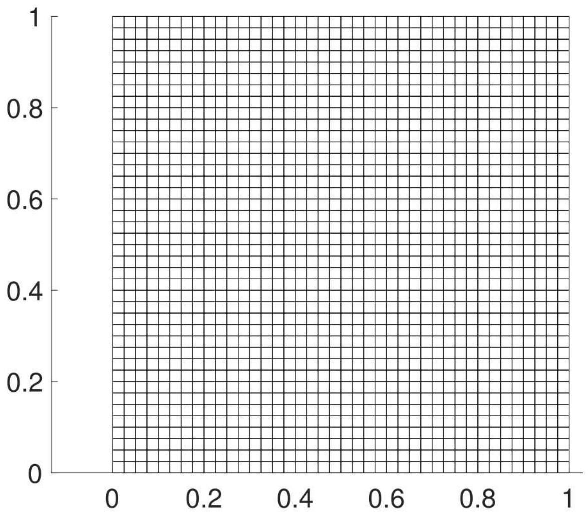
Numerical examples: nonlinear 2d heat equation

FE governing equation:

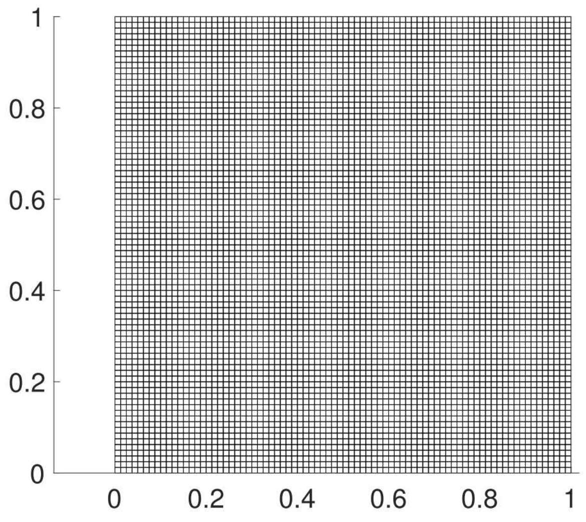
$$-\nabla^2 u + \frac{\mu_1}{\mu_2}(e^{\mu_2 u} - 1) = 100\sin(2\pi x_1)\sin(2\pi x_2)$$

[Grepl 2007]
[Chaturantabut 2010]

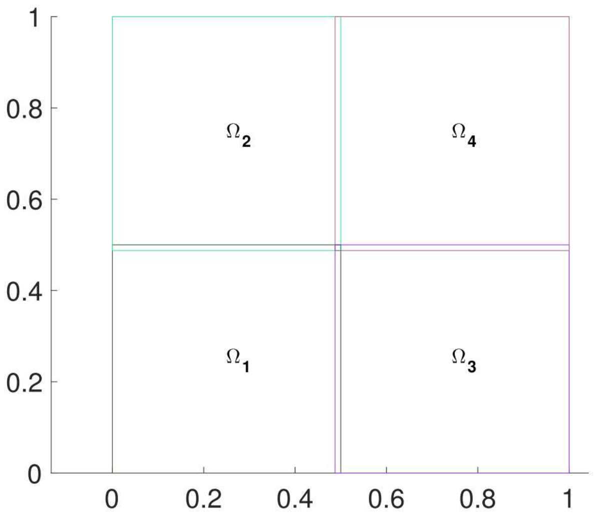
$$\mu = (\mu_1, \mu_2) \in \mathcal{D} = [0.01, 10]^2, |\Xi_{\text{train}}| = 400$$



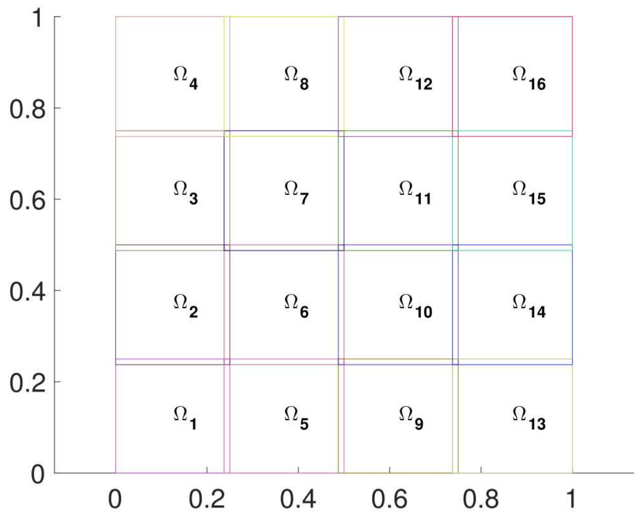
"Coarse" 40x40 elem.



"Fine" 80x80 elem.

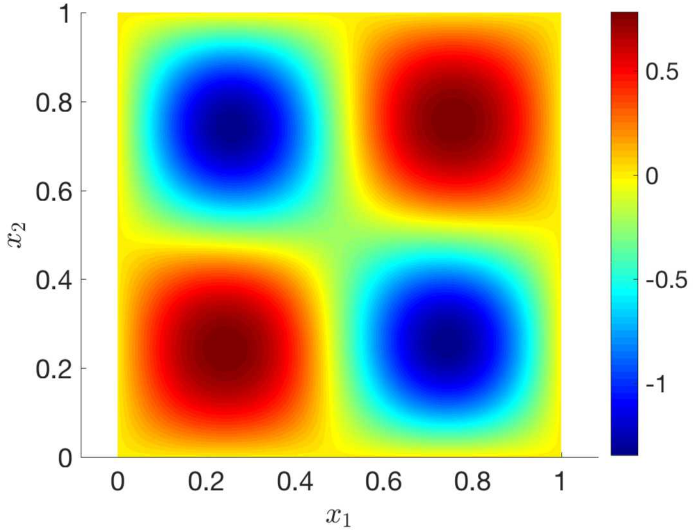


2x2 configuration

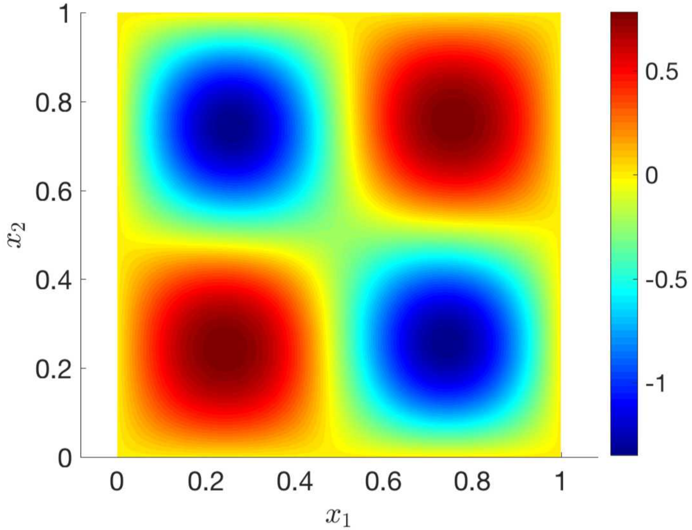


4x4 configuration

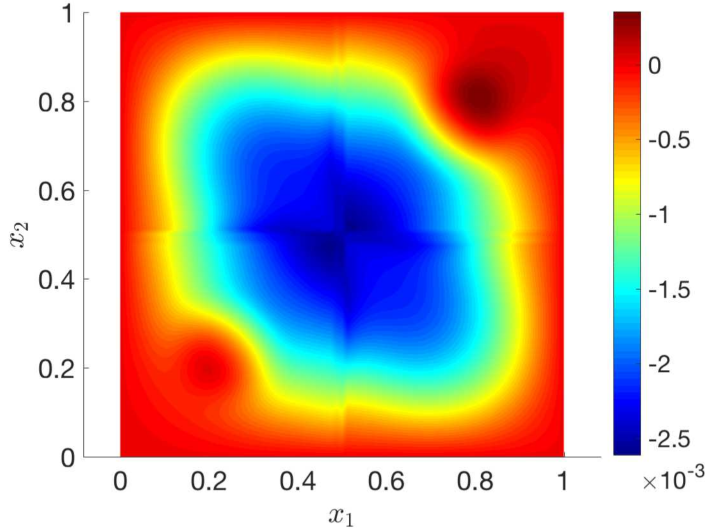
Numerical examples: DDROM and DDGNAT



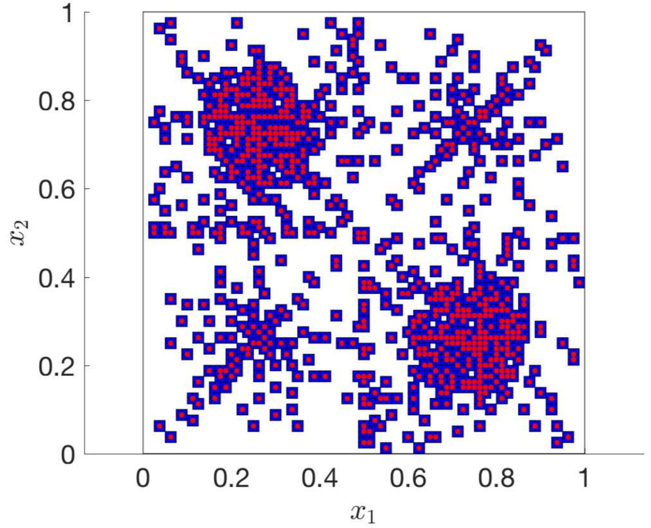
Full-order model solution



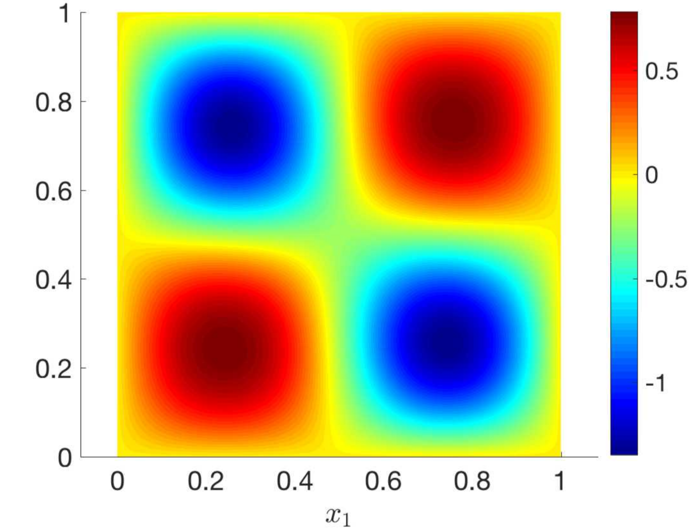
DDROM solution



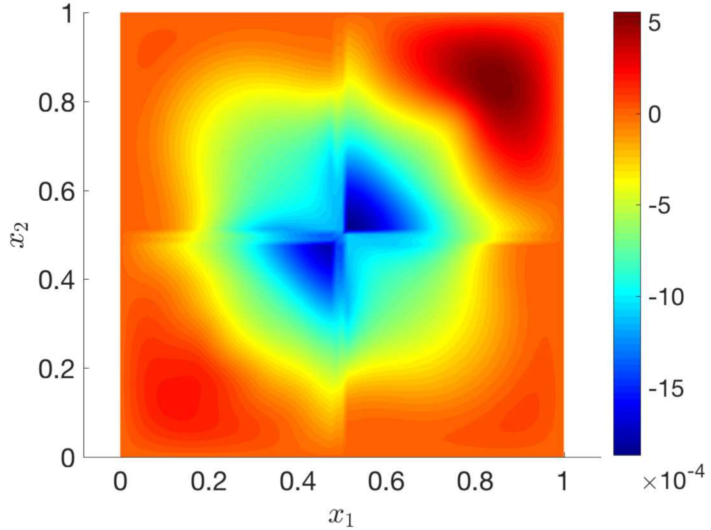
DDROM error



Sample mesh



DDGNAT solution



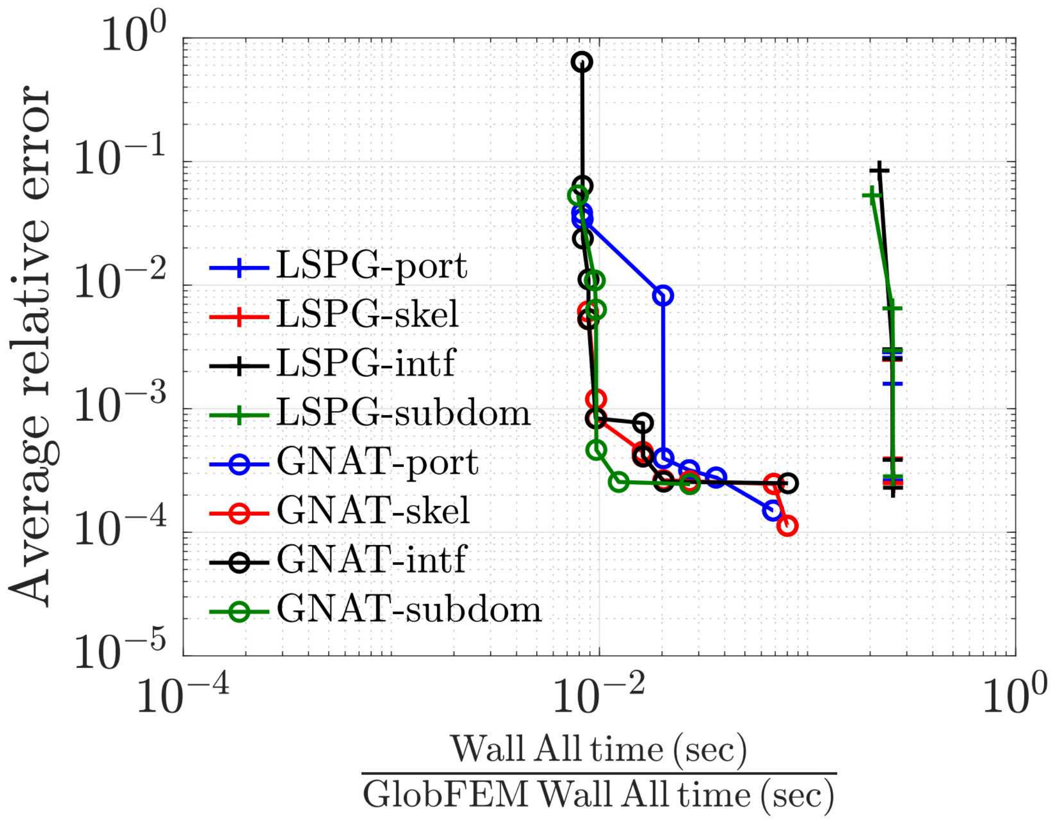
DDGNAT error

Numerical examples: many online computations

Table 1: (e.r.=energy rate, skel.=skeleton, intf.=interface, subdom.=subdomain) Heat equation, ROM method-parameters at point $\mu = (5.005, 5.005) \notin \Xi_{\text{train}}$ for many online computations

method	LSPG	GNAT
energy rate on Ω_i energy rate on Γ_i	$\{1-10^{-5}, 1-10^{-8}\}$ $\{1-10^{-5}, 1-10^{-8}\}$	$\{1-10^{-5}, 1-10^{-8}\}$ $\{1-10^{-5}, 1-10^{-8}\}$
e.r. for subdomain bases	$\{1-10^{-3}, 1-10^{-5},$ $1-10^{-7}, 1-10^{-9}\}$	$\{1-10^{-3}, 1-10^{-5},$ $1-10^{-7}, 1-10^{-9}\}$
energy rate on $\boldsymbol{r}_i$		$\{1-10^{-6}, 1-10^{-8},$ $1-10^{-10}, 1-10^{-12}\}$
$n_i^z/n_i^{\boldsymbol{r}}$ constraint type basis types	 {1, 2, 3, 4, 5, strong} {port, skel., intf., subdom.}	{1, 1.5, 2, 4} {1, 2, 3, 4, 5, strong} {port, skel., intf., subdom.}

Performance Pareto front: 2x2 "fine" configuration

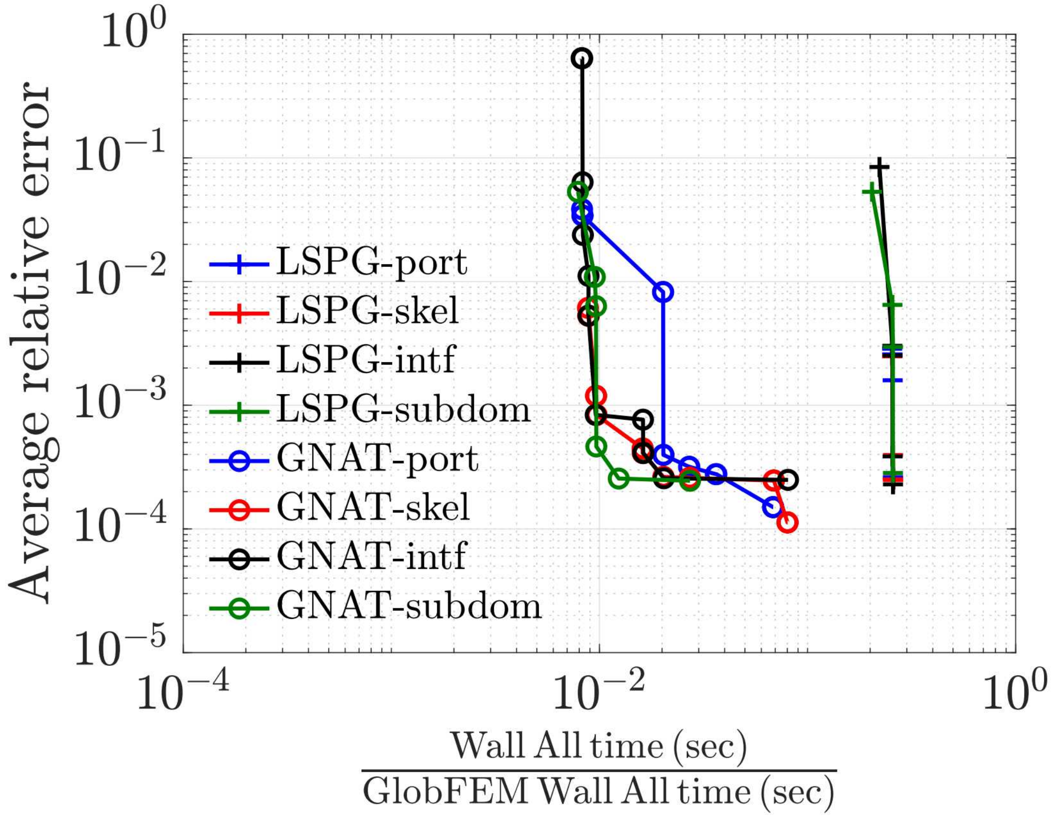


*Wall-time = timing on one processor

From: https://www.igi-global.com/dictionary/pareto-front/21878

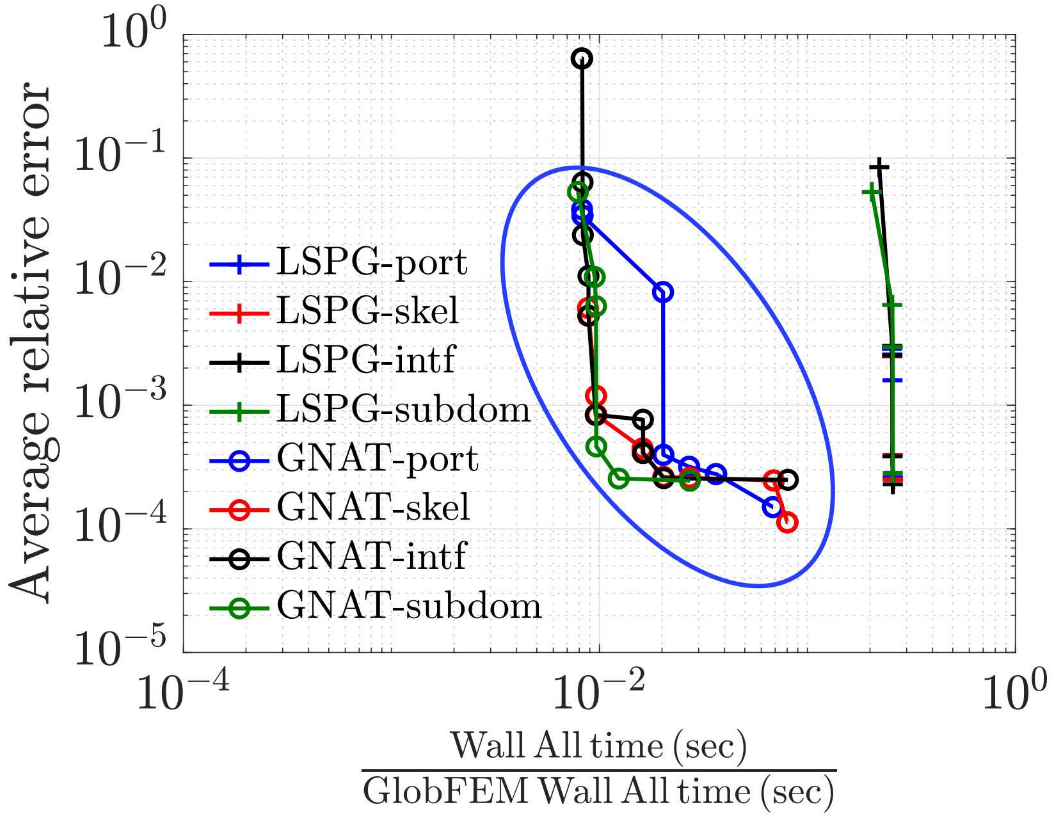
Pareto front is a set of nondominated solutions, being chosen as optimal, if no objective can be improved without sacrificing at least one other objective. On the other hand a solution x* is referred to as dominated by another solution x if, and only if, x is equally good or better than x* with respect to all objectives.

Performance Pareto front: 2x2 "fine" configuration



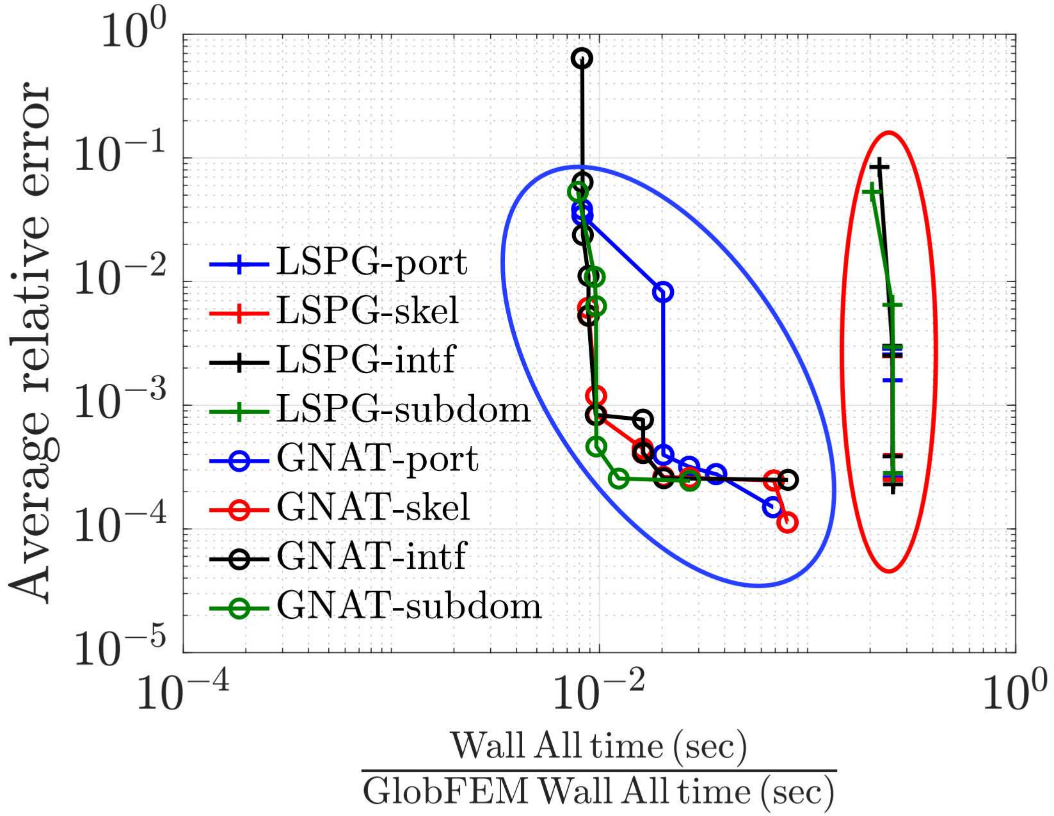
* Both DDLSPG & DDGNAT yields low errors and significant speedups w.r.t global FEM (i.e., no DD)

Performance Pareto front: 2x2 “fine” configuration



+ DDGNAT yields low errors and significant speedups (w.r.t DDLSPG)

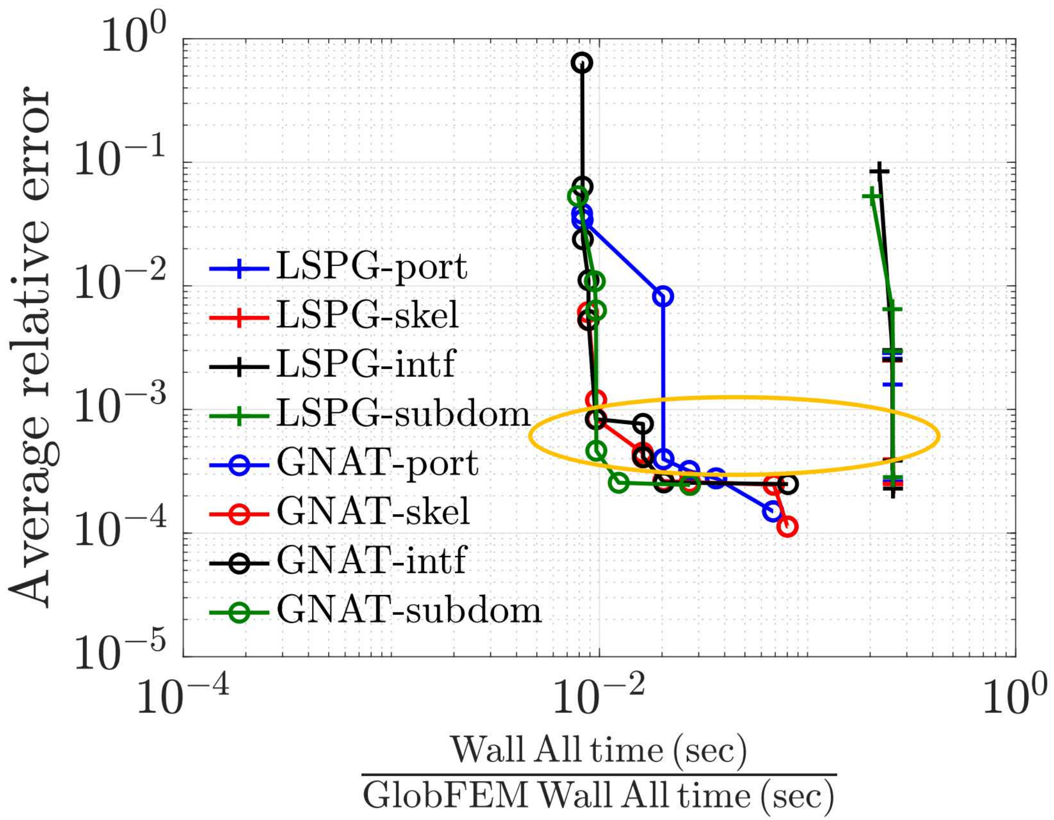
Performance Pareto front: 2x2 “fine” configuration



+ DDGNAT yields low errors and significant speedups (w.r.t DDLSPG)

* DDLSPG: can produce small errors, but incurs large relative wall time

Performance Pareto front: 2x2 “fine” configuration

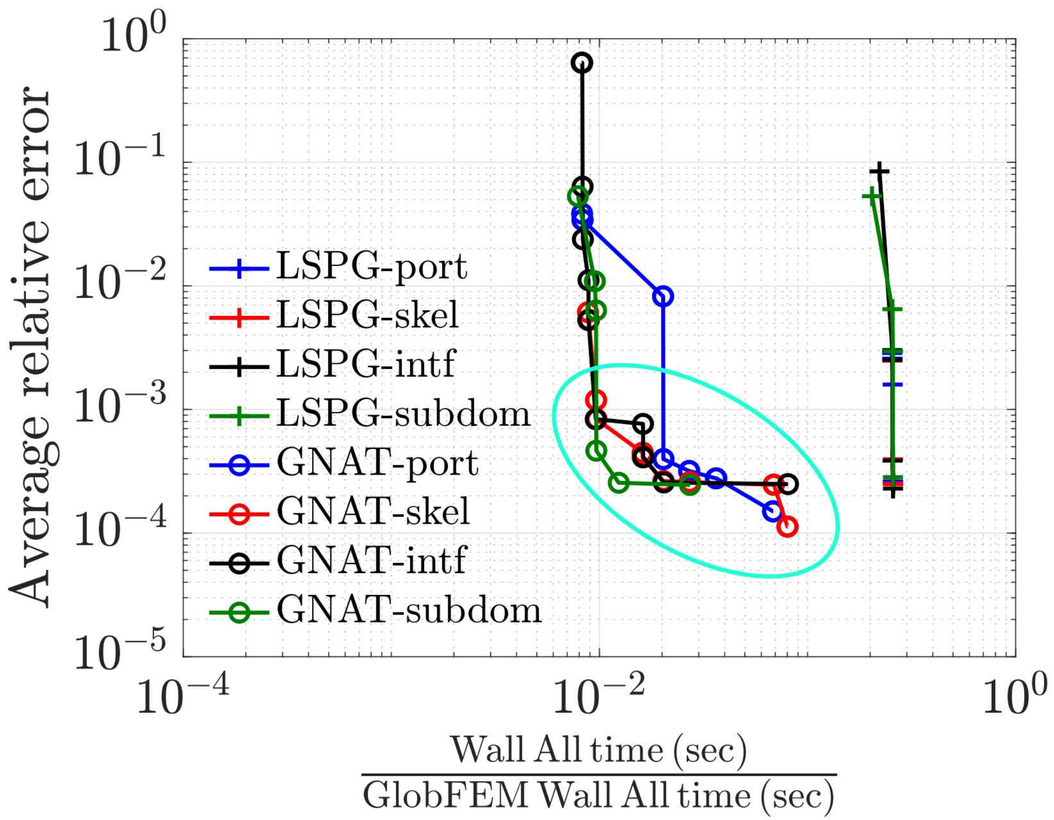


+ DDGNAT yields low errors and significant speedups (w.r.t DDLSPG)

* DDLSPG: can produce small errors, but incurs large relative wall time

+ For fixed error, DDGNAT almost 20X faster than DDLSPG

Performance Pareto front: 2x2 “fine” configuration



+ DDGNAT yields low errors and significant speedups (w.r.t DDLSPG)

* DDLSPG: can produce small errors, but incurs large relative wall time

+ For fixed error, DDGNAT almost 20X faster than DDLSPG

+ DDGNAT subdomain basis type is Pareto optimal

Performance of various constraint cases

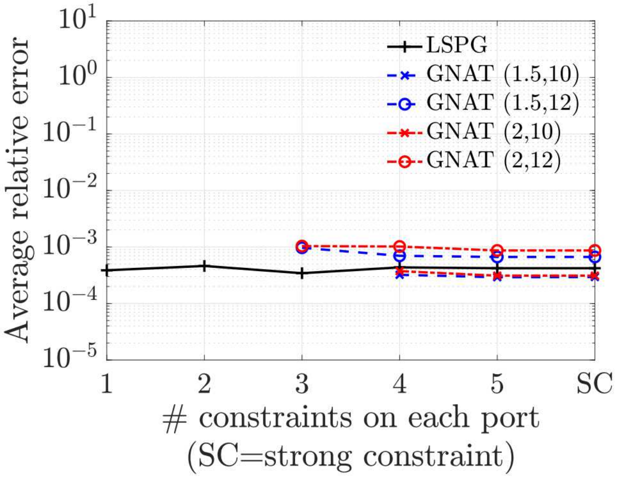
Energy rate on $\Omega_i = 1 - 10^{-8}$

Energy rate on $\Gamma_i = 1 - 10^{-8}$

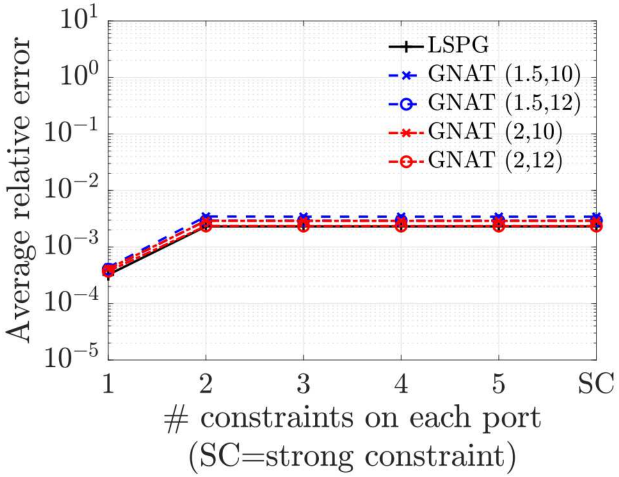
E.r. on $\Omega_i = 1 - 10^{-9}$

E.r. on $\Gamma_i = 1 - 10^{-9}$

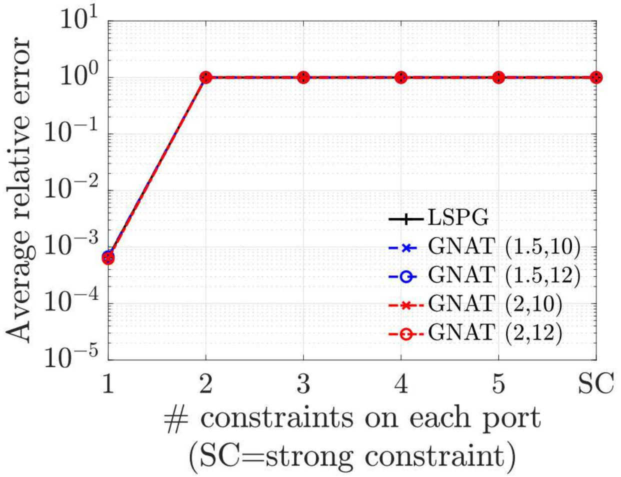
Port type



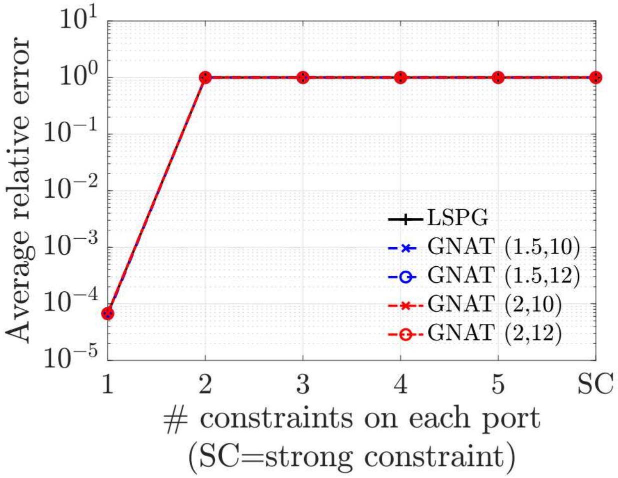
Skeleton type



Interface type



Subdomain type



+ Port & skeleton types work well with both weak & strong constraints

+ Interface & subdomain types, which gave Pareto-optimal results, require weak constraints for accuracy due to the mismatch of interface basis functions on neighboring subdomains

Conclusion

A new Domain-Decomposition Reduced-Order Modelling approach is proposed:

+ Solve nonlinear systems, with hyper-reduction

+ Subdomains ROMs constructed independently

+ Reformulate global ROM as a sum of nonlinear least-squares objectives over each subdomain with linear equality constraints and possibly weak compatibility

+ 4 kinds of reduced bases on the interfaces of subdomains: port, skeleton, interface and subdomain

+ Hyper-reduction is 20X faster than LSPG

+ Subdomain, interface and skeleton types outperform port type in performance

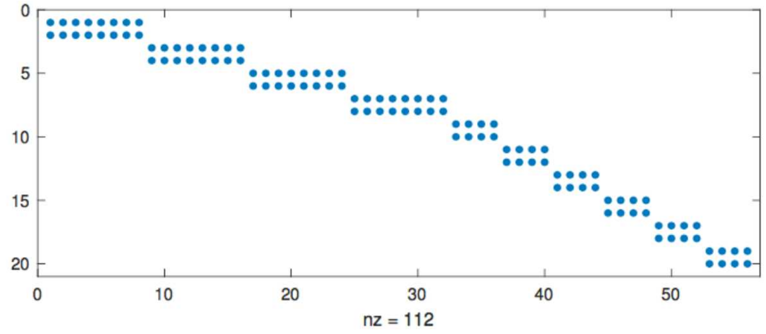
+ Skeleton is not practical for subdomain training

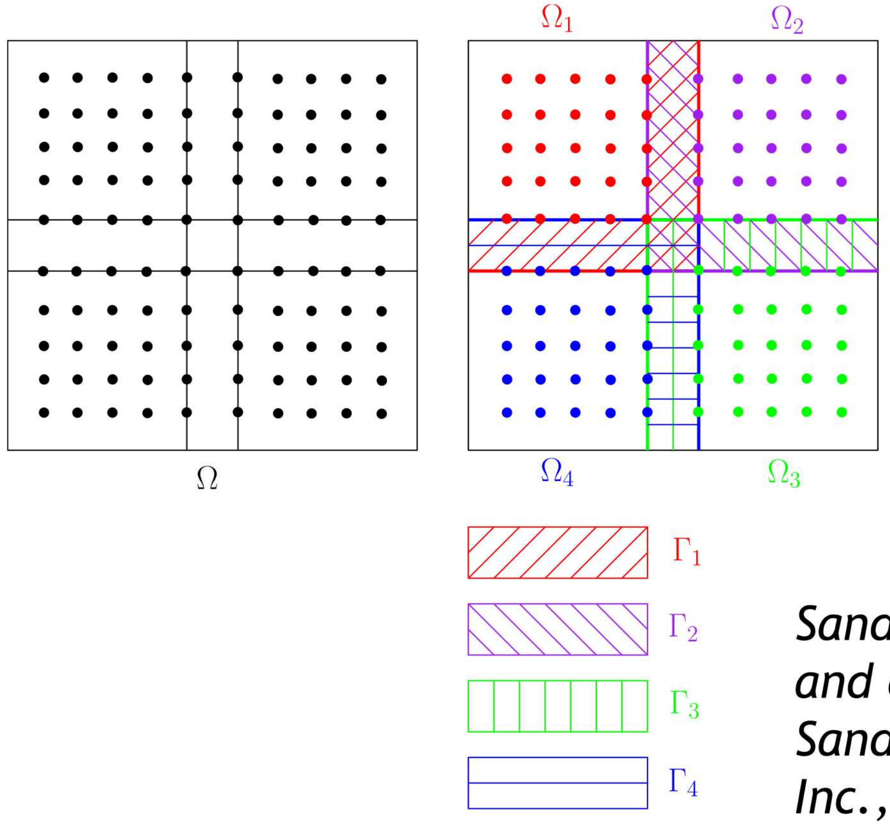
+ Subdomain and interface types are enabled thanks to weak constraints

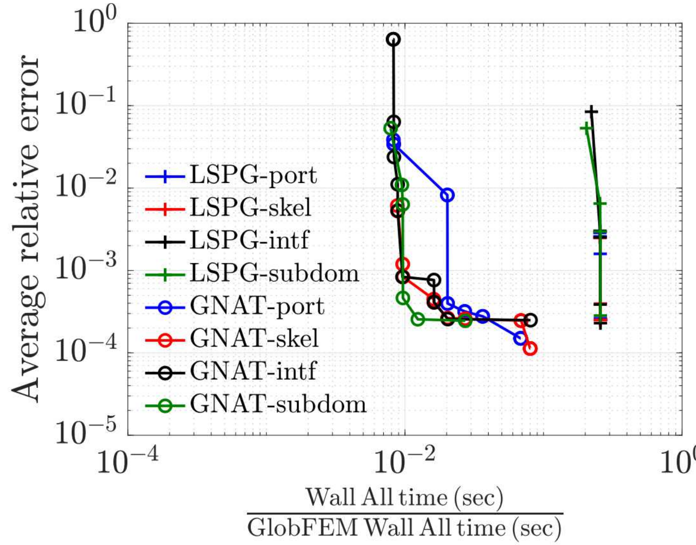
Questions?

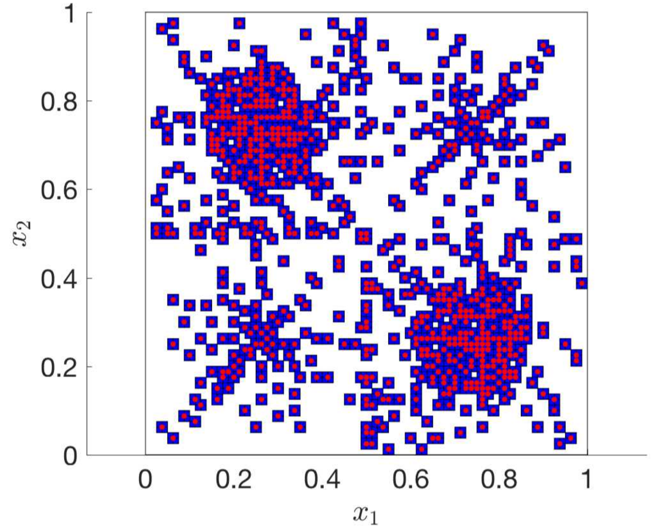
$$\underset{(\hat{\boldsymbol{x}}_1^\Omega,\ldots,\hat{\boldsymbol{x}}_{n_\Omega}^\Omega),(\hat{\boldsymbol{x}}_1^\Gamma,\ldots,\hat{\boldsymbol{x}}_{n_\Omega}^\Gamma)}{\text{minimize}} \quad \frac{1}{2}\sum_{i=1}^{n_\Omega} \|\boldsymbol{r}_i(\boldsymbol{\Phi}_i^\Omega \hat{\boldsymbol{x}}_i^\Omega, \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma)\|_2^2$$

$$\text{subject to} \quad \sum_{i=1}^{n_\Omega} \boldsymbol{A}_i \boldsymbol{\Phi}_i^\Gamma \hat{\boldsymbol{x}}_i^\Gamma = 0$$









Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.