September 16^{th} 2019
CIRM 2019 - Marseille

On the design and interface of parallel scalable sparse hybrid linear solvers

Emmanuel Agullo
joint work with
L. Giraud, M. Kuhn, G. Marait, L. Poirel

HiePACS Project Team
Inria Bordeaux Sud-Ouest

Context

Real System: *tire, airplane, earth atmosphere...*

Information: *temperature, velocity, pressure...*

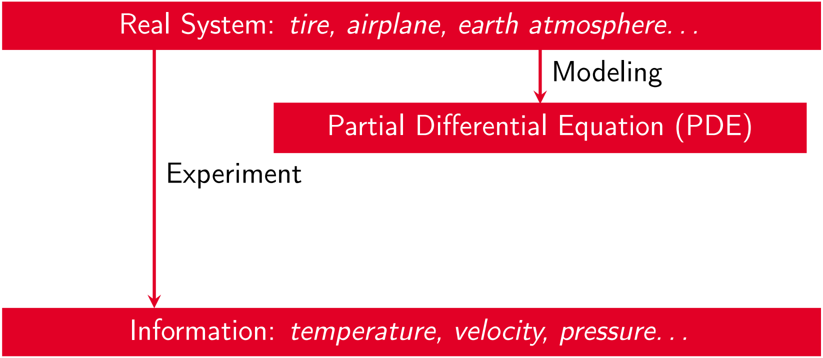
Context

Real System: *tire, airplane, earth atmosphere. . .*

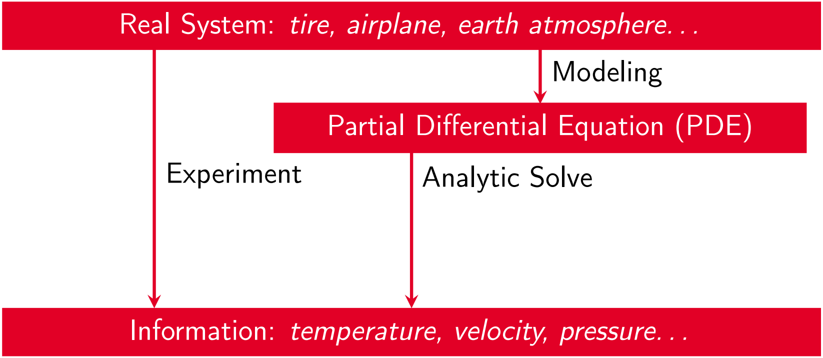
↓ Experiment

Information: *temperature, velocity, pressure. . .*

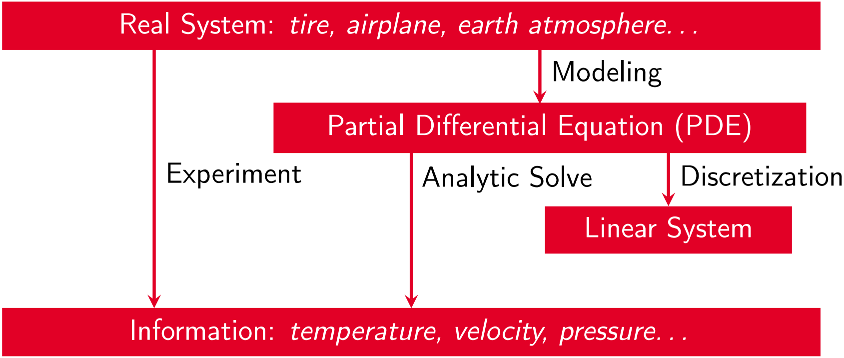
Context


Real System: *tire, airplane, earth atmosphere...*
Modeling
Partial Differential Equation (PDE)
Experiment
Information: *temperature, velocity, pressure...*

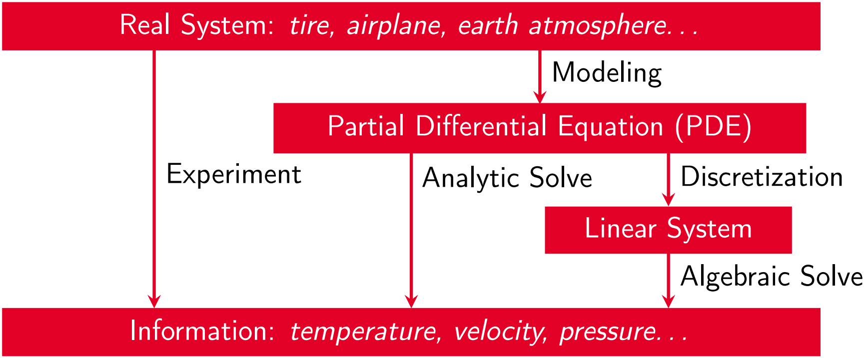
Context


Real System: *tire, airplane, earth atmosphere. . .*
Modeling
Partial Differential Equation (PDE)
Experiment
Analytic Solve
Information: *temperature, velocity, pressure. . .*

Context

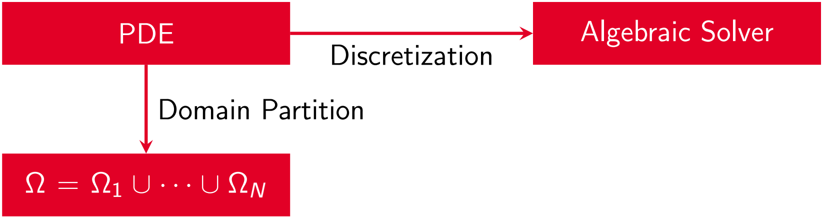

Real System: *tire, airplane, earth atmosphere...*
Modeling
Partial Differential Equation (PDE)
Experiment
Analytic Solve
Discretization
Linear System
Information: *temperature, velocity, pressure...*

Context

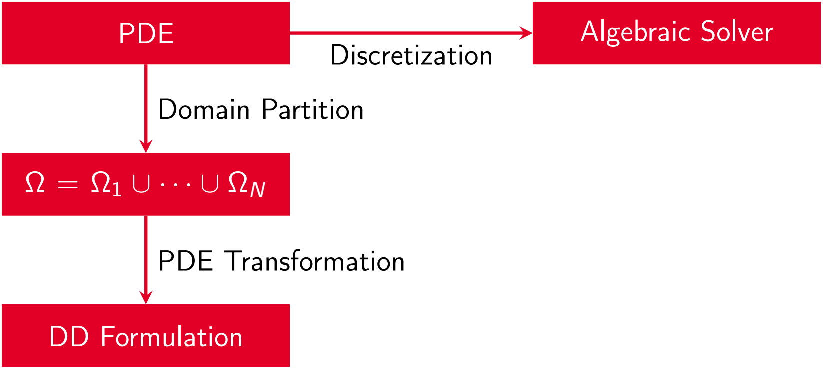

Real System: *tire, airplane, earth atmosphere. . .*
Modeling
Partial Differential Equation (PDE)
Experiment
Analytic Solve
Discretization
Linear System
Algebraic Solve
Information: *temperature, velocity, pressure. . .*

Domain Decomposition Methods

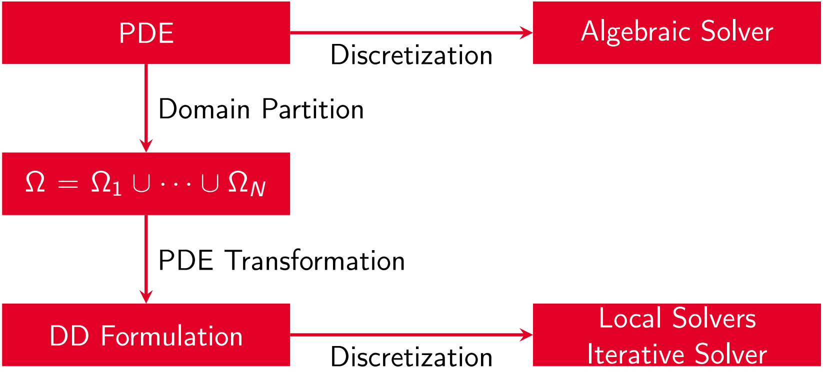
Domain Decomposition Methods



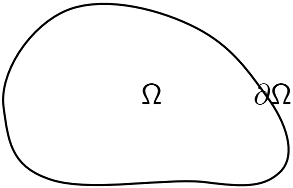
Domain Decomposition Methods



Domain Decomposition Methods



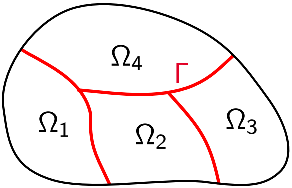
Domain Decomposition Methods (DDM)



Domain Decomposition Methods

- The domain is decomposed into subdomains

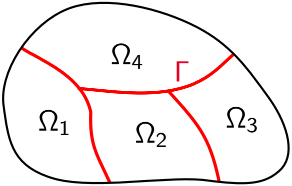
Domain Decomposition Methods (DDM)



Domain Decomposition Methods

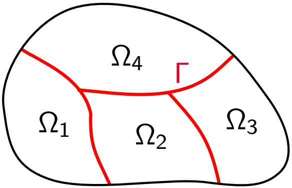
- The domain is decomposed into subdomains
- The model is transformed to include interface conditions

Domain Decomposition Methods (DDM)



Domain Decomposition Methods

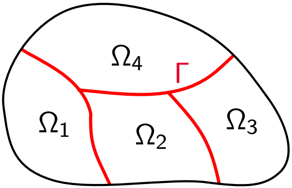
- The domain is decomposed into subdomains
- The model is transformed to include interface conditions
- The problem is solved in parallel
  - Choose an initial guess u in Ω or u_Γ on the interface Γ
  - Solve local problems in each subdomain Ω_i
  - Update u (or u_Γ) and iterate



The local problems in Ω_i

- Inside Ω_i and on $\partial\Omega$, the problem remains the same

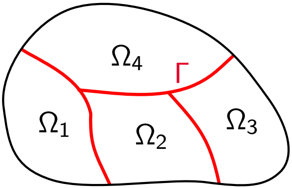
Domain Decomposition Methods (DDM)



The local problems in Ω_i

- Inside Ω_i and on $\partial\Omega$, the problem remains the same
- What boundary condition should we impose on Γ?

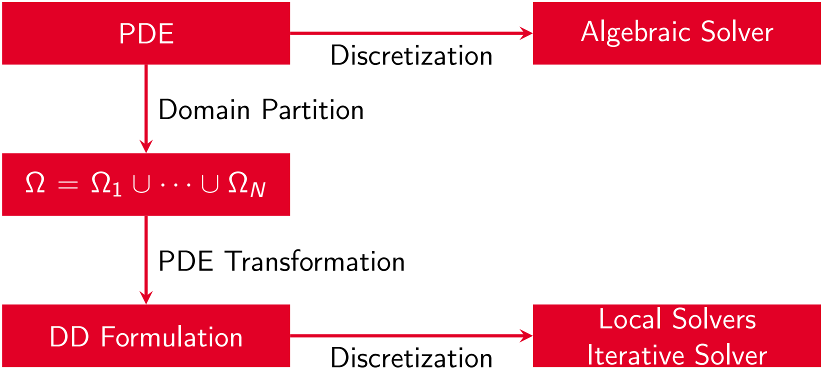
Domain Decomposition Methods (DDM)



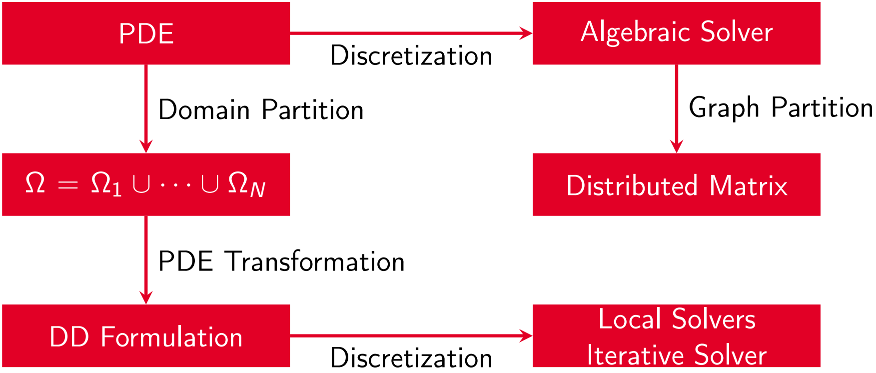
The local problems in Ω_i

- Inside Ω_i and on $\partial\Omega$, the problem remains the same
- What boundary condition should we impose on Γ?
  - Dirichlet BC → *imposed temperature*
  - Neumann BC → *imposed heat flow*
  - Robin BC → *heat flow depending on the temperature*

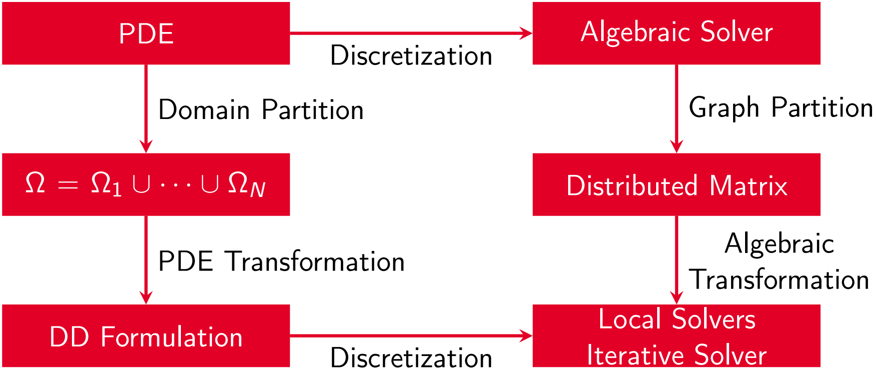
Domain Decomposition Methods and Hybrid Solvers



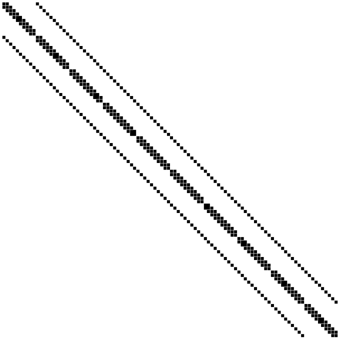
Domain Decomposition Methods and Hybrid Solvers



Domain Decomposition Methods and Hybrid Solvers



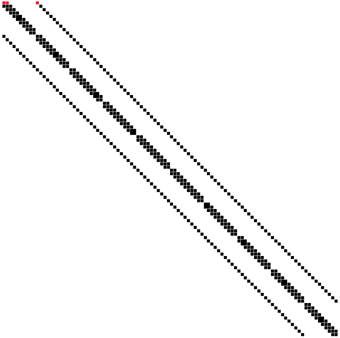
Global Matrix $\mathcal{K}$



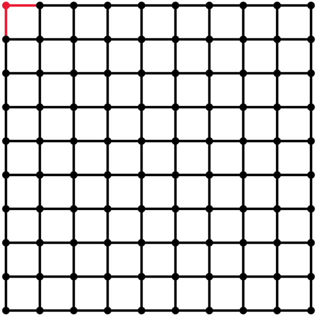
- $\mathcal{K}$ is a sparse matrix. We want to solve $\mathcal{K}u = f$.

The Hybrid Solver approach

Global Matrix $\mathcal{K}$



Adjacency graph G



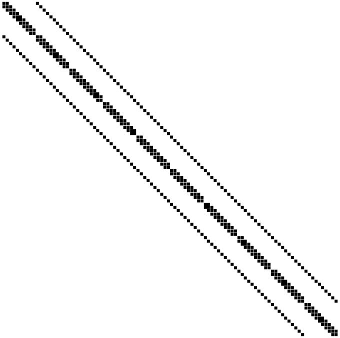
- The adjacency graph of $\mathcal{K}$ $(n \times n)$ is used as an algebraic mesh:

$$G = (\{1, \ldots, n\}, \; \{(i,j), \;\; i \neq j, \;\; a_{ij} \neq 0 \mid a_{ji} \neq 0\})$$

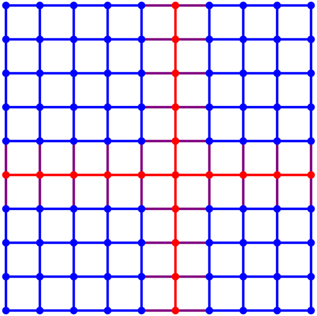
  - On the first row of $\mathcal{K}$, $k_{1,1}$, $k_{1,2}$ and $k_{1,11} \neq 0$
    $\Rightarrow (1,2)$ and $(1,11) \in G$

The Hybrid Solver approach

Global Matrix $\mathcal{K}$



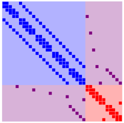
Adjacency graph G



- A graph partitioner is used to split the graph

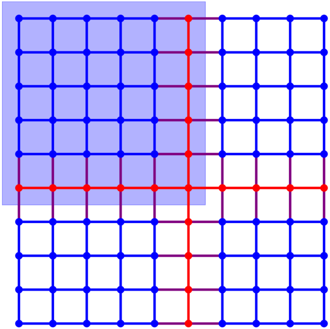
The Hybrid Solver approach

Local Matrices $\mathcal{K}_i$



$$\mathcal{K}_i = \begin{pmatrix} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i} & \mathcal{K}_{\mathcal{I}_i\Gamma_i} \\ \mathcal{K}_{\Gamma_i\mathcal{I}_i} & \mathcal{K}_{\Gamma_i\Gamma_i} \end{pmatrix}$$

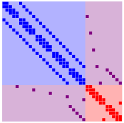
Adjacency graph G

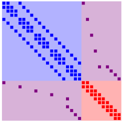


$$\mathcal{K} = \sum_{i=1}^{N} \mathcal{R}_{\Omega_i}^T \mathcal{K}_i \mathcal{R}_{\Omega_i}$$

The Hybrid Solver approach

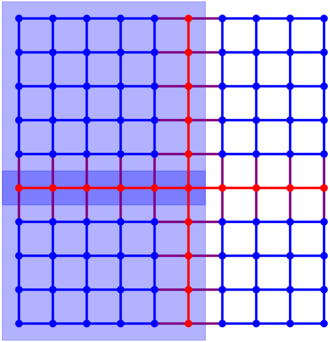
Local Matrices $\mathcal{K}_i$





$$\mathcal{K}_i = \begin{pmatrix} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i} & \mathcal{K}_{\mathcal{I}_i\Gamma_i} \\ \mathcal{K}_{\Gamma_i\mathcal{I}_i} & \mathcal{K}_{\Gamma_i\Gamma_i} \end{pmatrix}$$

Adjacency graph G

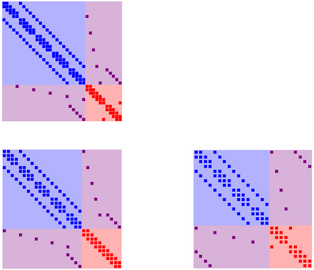


$$\mathcal{K} = \sum_{i=1}^{N} \mathcal{R}_{\Omega_i}^T \mathcal{K}_i \mathcal{R}_{\Omega_i}$$

- We have to split the interface non-zeros

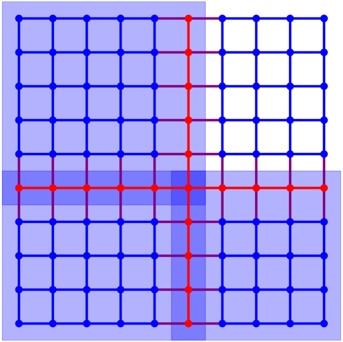
The Hybrid Solver approach

Local Matrices $\mathcal{K}_i$



$$\mathcal{K}_i = \begin{pmatrix} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i} & \mathcal{K}_{\mathcal{I}_i\Gamma_i} \\ \mathcal{K}_{\Gamma_i\mathcal{I}_i} & \mathcal{K}_{\Gamma_i\Gamma_i} \end{pmatrix}$$

Adjacency graph G

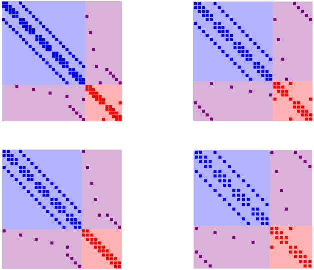


$$\mathcal{K} = \sum_{i=1}^{N} \mathcal{R}_{\Omega_i}^T \mathcal{K}_i \mathcal{R}_{\Omega_i}$$

- We have to split the interface non-zeros

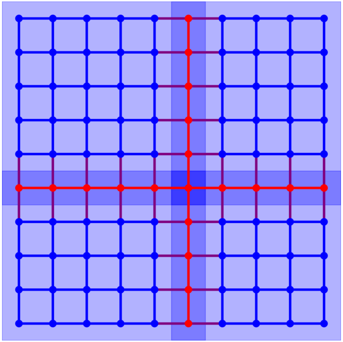
The Hybrid Solver approach

Local Matrices $\mathcal{K}_i$



$$\mathcal{K}_i = \begin{pmatrix} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i} & \mathcal{K}_{\mathcal{I}_i\Gamma_i} \\ \mathcal{K}_{\Gamma_i\mathcal{I}_i} & \mathcal{K}_{\Gamma_i\Gamma_i} \end{pmatrix}$$

Adjacency graph G

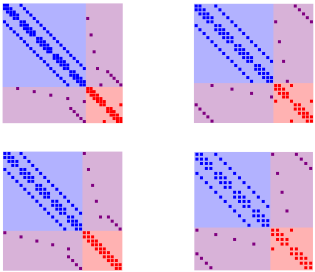


$$\mathcal{K} = \sum_{i=1}^{N} \mathcal{R}_{\Omega_i}^T \mathcal{K}_i \mathcal{R}_{\Omega_i}$$

- We have to split the interface non-zeros

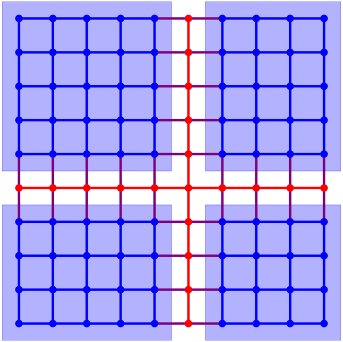
The substructuring approach

Local Matrices $\mathcal{K}_i$



$$\mathcal{K}_i = \begin{pmatrix} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i} & \mathcal{K}_{\mathcal{I}_i\Gamma_i} \\ \mathcal{K}_{\Gamma_i\mathcal{I}_i} & \mathcal{K}_{\Gamma_i\Gamma_i} \end{pmatrix}$$

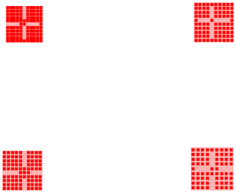
Adjacency graph G



$$\mathcal{K} = \sum_{i=1}^{N} \mathcal{R}_{\Omega_i}^T \mathcal{K}_i \mathcal{R}_{\Omega_i}$$

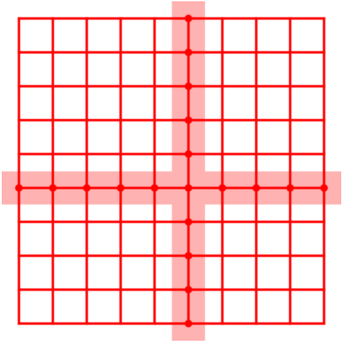
The substructuring approach

Local Schur Matrices $\mathcal{S}_i$



$$\mathcal{S}_i = \mathcal{K}_{\Gamma_i \Gamma_i} - \mathcal{K}_{\Gamma_i \mathcal{I}_i} \mathcal{K}_{\mathcal{I}_i \mathcal{I}_i}^{-1} \mathcal{K}_{\mathcal{I}_i \Gamma_i}$$

Adjacency graph G



$$\mathcal{S} = \sum_{i=1}^{N} \mathcal{R}_{\Gamma_i}^T \mathcal{S}_i \mathcal{R}_{\Gamma_i}$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{K}u = f \qquad \mathcal{K} = \sum_{i=1}^{N} \mathcal{R}_{\Omega_i}^T \mathcal{K}_i \mathcal{R}_{\Omega_i}$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{S}u = \widetilde{f}_\Gamma \qquad \mathcal{S} = \sum_{i=1}^{N} \mathcal{R}_{\Gamma_i}^T \mathcal{S}_i \mathcal{R}_{\Gamma_i}$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T {\mathcal{A}_i^{(aS)}}^{-1} \mathcal{R}_i$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T {\mathcal{A}_i^{(aS)}}^{-1} \mathcal{R}_i$$

- Additive Schwarz (AS)

$$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)^{-1}} \mathcal{R}_i$$

- Additive Schwarz (AS)

$$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

- Neumann Neumann (NN)

$$\mathcal{A}_i^{(aS)} = D_i^{-1} \mathcal{A}_i \; D_i^{-1} \qquad \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{R}_i = I$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)^{-1}} \mathcal{R}_i$$

- Additive Schwarz (AS)

$$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

- Neumann Neumann (NN)

$$\mathcal{A}_i^{(aS)} = D_i^{-1} \mathcal{A}_i^{\star} D_i^{-1} \qquad \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{R}_i = I$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T {\mathcal{A}_i^{(aS)}}^{-1} \mathcal{R}_i$$

- Additive Schwarz (AS)

$$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

- Neumann Neumann (NN)

$$\mathcal{A}_i^{(aS)} = D_i^{-1} \mathcal{A}_i{}^{\star} D_i^{-1} \qquad \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{R}_i = I$$

  - not fully algebraic: $\mathcal{A}_i^{\star} \to \mathcal{A}$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T {\mathcal{A}_i^{(aS)}}^{-1} \mathcal{R}_i$$

- Additive Schwarz (AS)

$$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

- Neumann Neumann (NN)

$$\mathcal{A}_i^{(aS)} = D_i^{-1} \mathcal{A}_i{}^{\star} D_i^{-1} \qquad \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{R}_i = I$$

  - not fully algebraic: $\mathcal{A}_i^{\star} \not\leftarrow \mathcal{A}$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T {\mathcal{A}_i^{(aS)}}^{-1} \mathcal{R}_i$$

- Additive Schwarz (AS)

$$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

- Neumann Neumann (NN)

$$\mathcal{A}_i^{(aS)} = D_i^{-1} \mathcal{A}_i{}^{\star} D_i^{-1} \qquad \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{R}_i = I$$

  - not fully algebraic: $\mathcal{A}_i^{\star} \not\leftarrow \mathcal{A}$
  - singular?

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)^{-1}} \mathcal{R}_i$$

- Additive Schwarz (AS)

  $$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

- Neumann Neumann (NN)

  $$\mathcal{A}_i^{(aS)} = D_i^{-1} \mathcal{A}_i{}^{\star} D_i^{-1} \qquad \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{R}_i = I$$

  - not fully algebraic: $\mathcal{A}_i^{\star} \not\leftarrow \mathcal{A}$
  - singular?
- Shifted (Sh)

  $$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{\star} + \lambda I$$

abstract Schwarz (aS) Preconditioners - [Dryja, Widlund, 1990]

$$\mathcal{A}x = b \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i \mathcal{R}_i \qquad \mathcal{A}^{-1} \approx \mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)^{-1}} \mathcal{R}_i$$

- Additive Schwarz (AS)

  $$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{(AS)} = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$$

- Neumann Neumann (NN)

  $$\mathcal{A}_i^{(aS)} = D_i^{-1} \mathcal{A}_i{}^{\star} D_i^{-1} \qquad \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{R}_i = I$$

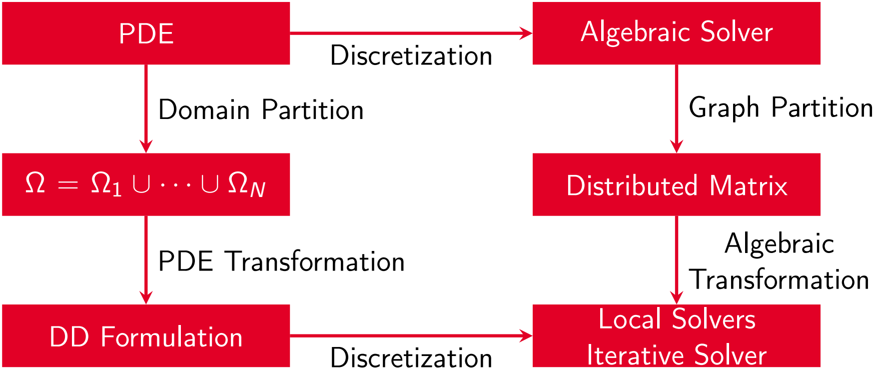
  - not fully algebraic: $\mathcal{A}_i^{\star} \not\leftarrow \mathcal{A}$
  - singular?
- Shifted (Sh)

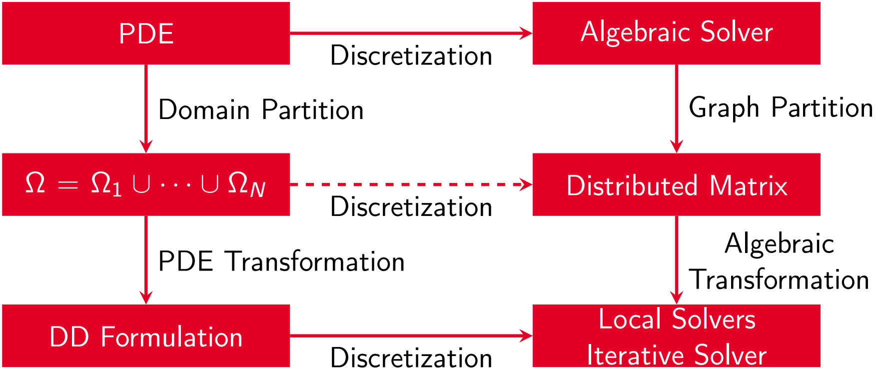
  $$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{\star} + \lambda I$$

- Robin (Ro)

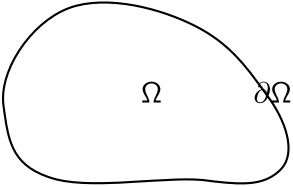
  $$\mathcal{A}_i^{(aS)} = \mathcal{A}_i^{\star} + \mathcal{T}_{\Gamma_i}$$

Domain Decomposition Methods and Hybrid Solvers



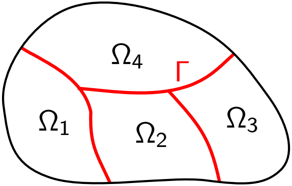

PDE
Discretization
Algebraic Solver
Domain Partition
Graph Partition
$\Omega = \Omega_1 \cup \cdots \cup \Omega_N$
Discretization
Distributed Matrix
PDE Transformation
Algebraic Transformation
DD Formulation
Discretization
Local Solvers
Iterative Solver

DDM at the algebraic level



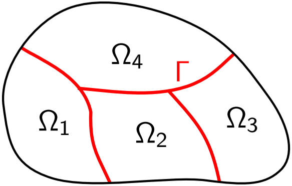
1. PDE defined on the global domain Ω

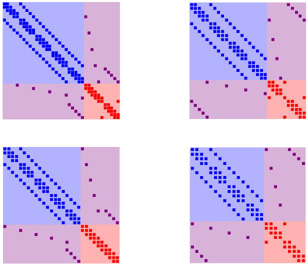
DDM at the algebraic level



1. PDE defined on the global domain Ω
2. Partition into subdomains Ω_i

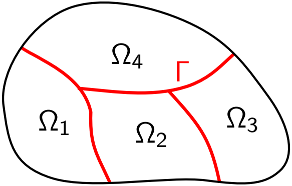
DDM at the algebraic level

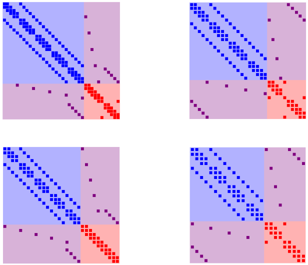




1. PDE defined on the global domain Ω
2. Partition into subdomains Ω_i
3. Discretization at the subdomain level $\Omega_i \rightarrow \mathcal{K}_i^{\star}$
   - Neumann boundary condition on Γ

DDM at the algebraic level





1. PDE defined on the global domain Ω
2. Partition into subdomains Ω_i
3. Discretization at the subdomain level $\Omega_i \rightarrow \mathcal{K}_i^\star$
   - Neumann boundary condition on Γ
4. DDM at the algebraic level

Statement:

Domain Decomposition Methods
can be implemented efficiently
at the algebraic level

Context: Coarse Space Correction for aS methods

Choice of a coarse space $V_0 = \sum_{i=1}^{N} \mathcal{R}_i^T V_0^i$

- Constant per subdomain [Nicolaides, 1987]
- Partition of unity [Sarkis, 2003]

Context: Coarse Space Correction for aS methods

Choice of a coarse space $V_0 = \sum_{i=1}^{N} \mathcal{R}_i^T V_0^i$

- Constant per subdomain [Nicolaides, 1987]
- Partition of unity [Sarkis, 2003]
- Adaptative (GenEO) [Spillane et al., 2013]
  - based on element matrices a_τ and topological information

Context: Coarse Space Correction for aS methods

Choice of a coarse space $V_0 = \sum_{i=1}^{N} \mathcal{R}_i^T V_0^i$

- Constant per subdomain [Nicolaides, 1987]
- Partition of unity [Sarkis, 2003]
- Adaptative (GenEO) [Spillane et al., 2013]
  - based on element matrices a_τ and topological information

Coarse Space Correction for abstract Schwarz

V_0	Coarse space
$\mathcal{M}_0 = V_0 (V_0^T \mathcal{A} V_0)^\dagger V_0^T$	Coarse solve
$\mathcal{P}_0 = \mathcal{M}_0 \mathcal{A}$	Projector onto the coarse space

Context: Coarse Space Correction for aS methods

Choice of a coarse space $V_0 = \sum_{i=1}^{N} \mathcal{R}_i^T V_0^i$

- Constant per subdomain [Nicolaides, 1987]
- Partition of unity [Sarkis, 2003]
- Adaptative (GenEO) [Spillane et al., 2013]
  - based on element matrices a_τ and topological information

Coarse Space Correction for abstract Schwarz

V_0	Coarse space
$\mathcal{M}_0 = V_0(V_0^T \mathcal{A} V_0)^\dagger V_0^T$	Coarse solve
$\mathcal{P}_0 = \mathcal{M}_0 \mathcal{A}$	Projector onto the coarse space

- one-level aS: $\mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)^\dagger} \mathcal{R}_i$

Choice of a coarse space $V_0 = \sum_{i=1}^{N} \mathcal{R}_i^T V_0^i$

- Constant per subdomain [Nicolaides, 1987]
- Partition of unity [Sarkis, 2003]
- Adaptative (GenEO) [Spillane et al., 2013]
  - based on element matrices a_τ and topological information

Coarse Space Correction for abstract Schwarz

V_0	Coarse space
$\mathcal{M}_0 = V_0 (V_0^T \mathcal{A} V_0)^\dagger V_0^T$	Coarse solve
$\mathcal{P}_0 = \mathcal{M}_0 \mathcal{A}$	Projector onto the coarse space

- one-level aS: $\mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)^\dagger} \mathcal{R}_i$
- deflated aS: $\mathcal{M}_{aS,D} = (I - \mathcal{P}_0) \mathcal{M}_{aS} (I - \mathcal{P}_0)^T + \mathcal{M}_0$

Context: Coarse Space Correction for aS methods

Choice of a coarse space $V_0 = \sum_{i=1}^{N} \mathcal{R}_i^T V_0^i$

- Constant per subdomain [Nicolaides, 1987]
- Partition of unity [Sarkis, 2003]
- Adaptative (GenEO) [Spillane et al., 2013]
  - based on element matrices a_τ and topological information

Coarse Space Correction for abstract Schwarz

V_0	Coarse space
$\mathcal{M}_0 = V_0(V_0^T \mathcal{A} V_0)^\dagger V_0^T$	Coarse solve
$\mathcal{P}_0 = \mathcal{M}_0 \mathcal{A}$	Projector onto the coarse space

- one-level aS: $\mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)^\dagger} \mathcal{R}_i$
- deflated aS: $\mathcal{M}_{aS,D} = (I - \mathcal{P}_0)\mathcal{M}_{aS}(I - \mathcal{P}_0)^T + \mathcal{M}_0$

Convergence of aS methods

Conjugate Gradient Error-estimate

$$\frac{\|x_k - x^*\|_{\mathcal{A}}}{\|x_0 - x^*\|_{\mathcal{A}}} \leq 2 \left(\frac{\sqrt{\kappa(\mathcal{M}\mathcal{A})} - 1}{\sqrt{\kappa(\mathcal{M}\mathcal{A})} + 1} \right)^k \qquad \kappa(\mathcal{M}\mathcal{A}) = \frac{\lambda_{\max}(\mathcal{M}\mathcal{A})}{\lambda_{\min}(\mathcal{M}\mathcal{A})}$$

Convergence of aS methods

Conjugate Gradient Error-estimate

$$\frac{\|x_k - x^*\|_{\mathcal{A}}}{\|x_0 - x^*\|_{\mathcal{A}}} \leq 2 \left(\frac{\sqrt{\kappa(\mathcal{M}\mathcal{A})} - 1}{\sqrt{\kappa(\mathcal{M}\mathcal{A})} + 1} \right)^k \qquad \kappa(\mathcal{M}\mathcal{A}) = \frac{\lambda_{\max}(\mathcal{M}\mathcal{A})}{\lambda_{\min}(\mathcal{M}\mathcal{A})}$$

Algebraic bounds

$$1 \leq \lambda_{\min}(\mathcal{M}_{NN}\mathcal{A}) \qquad \lambda_{\max}(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$$

Bound $\kappa \leq \chi(\mathcal{A}_i^\star, \mathcal{A}_i^{(aS)}, V_0)$ (L. Poirel's thesis - 2018)

Deflated aS Preconditioner: $\mathcal{A}_i^{(aS)}$ and V_0^i given

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

where $V_0 = \sum_{i=1}^N \mathcal{R}_i^T V_0^i$ $\quad \mathcal{M}_{aS,D} = \mathcal{M}_0 + (I - \mathcal{P}_0) \sum_{i=1}^{N} \mathcal{R}_i^T \mathcal{A}_i^{(aS)\dagger} \mathcal{R}_i \, (I - \mathcal{P}_0)^T$

Flavor of the proof

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Flavor of the proof

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i}\right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Upper bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$

Flavor of the proof

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1\leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1\leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Upper bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$

$\lambda(\mathcal{M}_{aS}\mathcal{A})$

Flavor of the proof

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Upper bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$

$$\lambda(\mathcal{M}_{aS}\mathcal{A}) \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Flavor of the proof

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1\leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1\leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Upper bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$

$$\lambda(\mathcal{M}_{aS}\mathcal{A}) \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(aS)} u_i} \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(AS)} u_i} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Upper bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$

$$\lambda(\mathcal{M}_{aS}\mathcal{A}) \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(aS)} u_i} \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(AS)} u_i} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i} \lesssim N_c \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Flavor of the proof

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1\leq i\leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1\leq i\leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Upper bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$

$$\lambda(\mathcal{M}_{aS}\mathcal{A}) \approx \frac{u^T\mathcal{A}u}{u_i^T \mathcal{A}_i^{(aS)} u_i} \approx \frac{u^T\mathcal{A}u}{u_i^T \mathcal{A}_i^{(AS)} u_i} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i} \lesssim N_c \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Lower bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{NN}\mathcal{A}) \geq 1$

Flavor of the proof

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Upper bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{AS}\mathcal{A}) \leq N_c$

$$\lambda(\mathcal{M}_{aS}\mathcal{A}) \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(aS)} u_i} \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(AS)} u_i} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i} \lesssim N_c \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Lower bound for $\lambda(\mathcal{M}_{aS}\mathcal{A})$

$\lambda(\mathcal{M}_{NN}\mathcal{A}) \geq 1$

$$\lambda(\mathcal{M}_{aS}\mathcal{A}) \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(aS)} u_i} \approx \frac{u^T \mathcal{A} u}{u_i^T \mathcal{A}_i^{(NN)} u_i} \frac{u_i^T \mathcal{A}_i^{(NN)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i} \gtrsim 1 \frac{u_i^T \mathcal{A}_i^{(NN)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Extending GenEO for aS

Choose $\mathcal{A}_i^{(aS)}$ and χ, build $V_0(\mathcal{A}_i^\star, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \le \chi$

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \le \left(1 + \sup_{\substack{1 \le i \le N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \le i \le N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

Extending GenEO for aS

Choose $\mathcal{A}_i^{(aS)}$ and χ, build $V_0(\mathcal{A}_i^{\star}, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \leq \chi$

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^{\perp}}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

1. Choose two thresholds α and β

Extending GenEO for aS

Choose $\mathcal{A}_i^{(aS)}$ and χ, build $V_0(\mathcal{A}_i^\star, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \leq \chi$

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

1. Choose two thresholds α and β
2. Solve locally the generalized eigenproblems

$$\mathcal{A}_i^{(aS)} p = \lambda\, \mathcal{A}_i^{(NN)} p \qquad \text{and} \qquad \mathcal{A}_i^{(AS)} p = \eta\, \mathcal{A}_i^{(aS)} p$$

for eigenvalues $\lambda \geq \alpha$ and $\eta \geq N_c \beta$

Extending GenEO for aS

Choose $\mathcal{A}_i^{(aS)}$ and χ, build $V_0(\mathcal{A}_i^\star, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \leq \chi$

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

1. Choose two thresholds α and β
2. Solve locally the generalized eigenproblems

   $$\mathcal{A}_i^{(aS)} p = \lambda\, \mathcal{A}_i^{(NN)} p \qquad \text{and} \qquad \mathcal{A}_i^{(AS)} p = \eta\, \mathcal{A}_i^{(aS)} p$$

   for eigenvalues $\lambda \geq \alpha$ and $\eta \geq N_c \beta$
3. Assemble the resulting coarse space $V_0 = \left(\mathcal{R}_1^T V_0^1 \; \cdots \; \mathcal{R}_N^T V_0^N\right)$

Extending GenEO for aS

Choose $\mathcal{A}_i^{(aS)}$ and χ, build $V_0(\mathcal{A}_i^\star, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \leq \chi$

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq \left(1 + \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(NN)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(aS)} u_i}{u_i^T \mathcal{A}_i^{(NN)} u_i} \right) N_c \sup_{\substack{1 \leq i \leq N \\ u_i \in \left(\mathcal{A}_i^{(aS)} V_0^i\right)^\perp}} \frac{u_i^T \mathcal{A}_i^{(AS)} u_i}{u_i^T \mathcal{A}_i^{(aS)} u_i}$$

1. Choose two thresholds α and β
2. Solve locally the generalized eigenproblems

$$\mathcal{A}_i^{(aS)} p = \lambda \, \mathcal{A}_i^{(NN)} p \qquad \text{and} \qquad \mathcal{A}_i^{(AS)} p = \eta \, \mathcal{A}_i^{(aS)} p$$

   for eigenvalues $\lambda \geq \alpha$ and $\eta \geq N_c \beta$
3. Assemble the resulting coarse space $V_0 = \left(\mathcal{R}_1^T V_0^1 \ \cdots \ \mathcal{R}_N^T V_0^N \right)$

Then:

$$\kappa(\mathcal{M}_{aS,D}\mathcal{A}) \leq (1 + \alpha)\, \beta = \chi$$

Shortcut of the results in an abstract Schwarz framework

Convergence theorem

1. Choose $\mathcal{A}_i^{(aS)}$ and V_0
2. Bound $\kappa \leq \chi(\mathcal{A}_i^{\star}, \mathcal{A}_i^{(aS)}, V_0)$

Algebraic GenEO-like coarse space

1. Choose $\mathcal{A}_i^{(aS)}$ and χ
2. Build $V_0(\mathcal{A}_i^{\star}, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \leq \chi$

Shortcut of the results in an abstract Schwarz framework

Convergence theorem

1. Choose $\mathcal{A}_i^{(aS)}$ and V_0
2. Bound $\kappa \leq \chi(\mathcal{A}_i^{\star}, \mathcal{A}_i^{(aS)}, V_0)$

Algebraic GenEO-like coarse space

1. Choose $\mathcal{A}_i^{(aS)}$ and χ
2. Build $V_0(\mathcal{A}_i^{\star}, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \leq \chi$

- For any SPSD local preconditioner $\mathcal{A}_i^{(aS)}$

Shortcut of the results in an abstract Schwarz framework

Convergence theorem

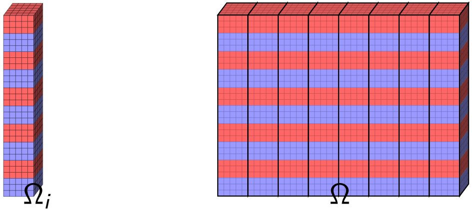
1. Choose $\mathcal{A}_i^{(aS)}$ and V_0
2. Bound $\kappa \leq \chi(\mathcal{A}_i^\star, \mathcal{A}_i^{(aS)}, V_0)$

Algebraic GenEO-like coarse space

1. Choose $\mathcal{A}_i^{(aS)}$ and χ
2. Build $V_0(\mathcal{A}_i^\star, \mathcal{A}_i^{(aS)}, \chi)$ such that $\kappa \leq \chi$

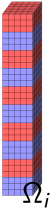
- For any SPSD local preconditioner $\mathcal{A}_i^{(aS)}$
- Need only algebraic input at the subdomain level $\mathcal{A}_i^\star$

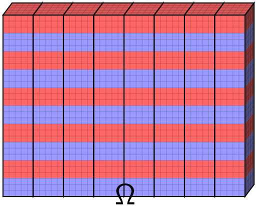
Heterogeneous diffusion in a 3D stratified medium



- Each subdomain Ω_i has $5 \times 30 \times 5$ Q_1 elements
- Ω is decomposed in $N \times 1 \times 1$ subdomains

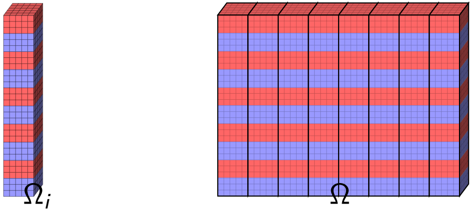
Heterogeneous diffusion in a 3D stratified medium



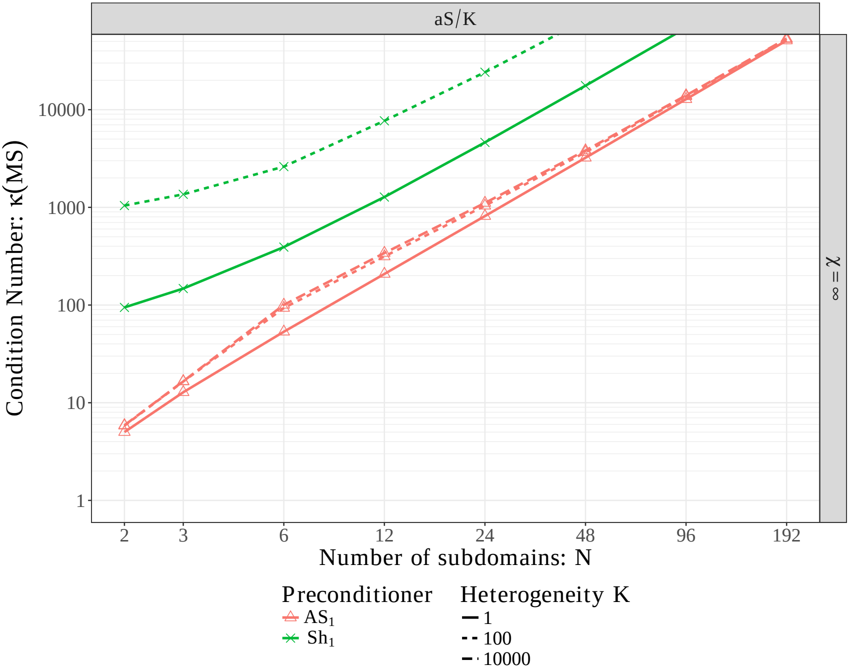


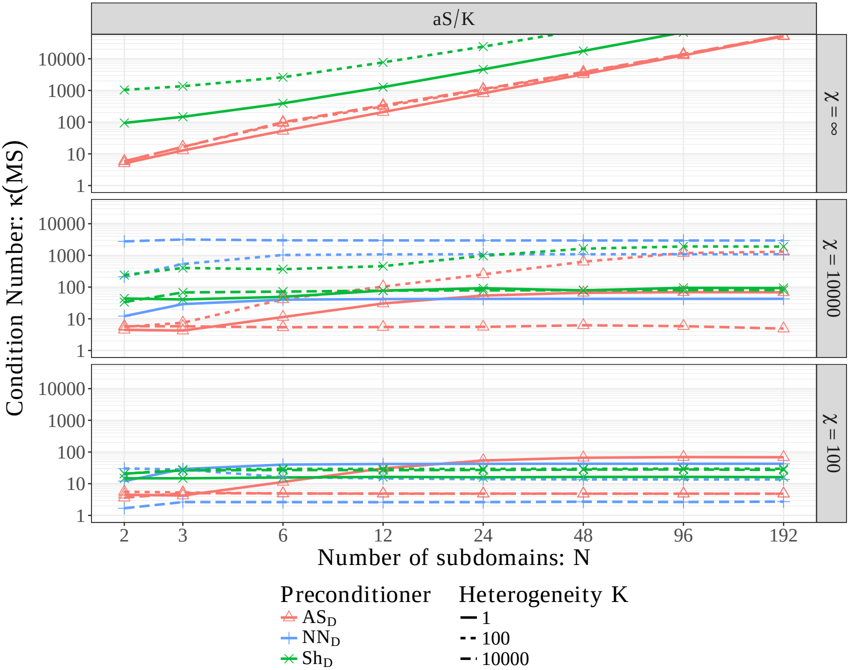
- Each subdomain Ω_i has $5 \times 30 \times 5$ Q_1 elements
- Ω is decomposed in $N \times 1 \times 1$ subdomains
- $\nabla(k \nabla u) = 1$ with $k = 1$ (blue) or $k = K$ (red)
- BC: Dirichlet on the left, Neumann elsewhere

Heterogeneous diffusion in a 3D stratified medium

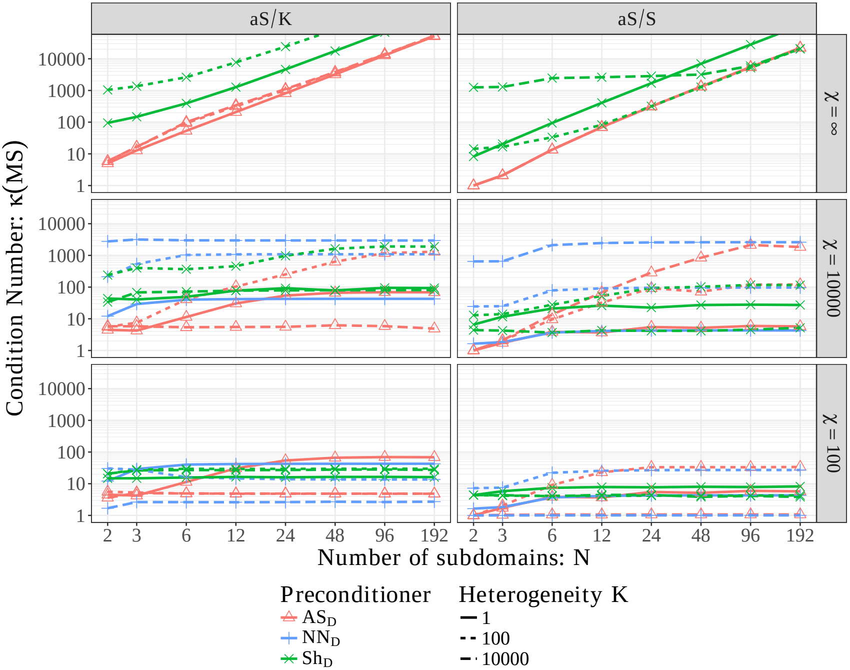


- Each subdomain Ω_i has $5 \times 30 \times 5$ Q_1 elements
- Ω is decomposed in $N \times 1 \times 1$ subdomains
- $\nabla(k\nabla u) = 1$ with $k = 1$ (blue) or $k = K$ (red)
- BC: Dirichlet on the left, Neumann elsewhere
- *A*dditive *S*chwarz, *N*eumann-*N*eumann or *Sh*ifted preconditioner, with ($aS/\mathcal{S}$) or without substructuring ($aS/\mathcal{K}$)
- Condition number: $\kappa \leq \chi$

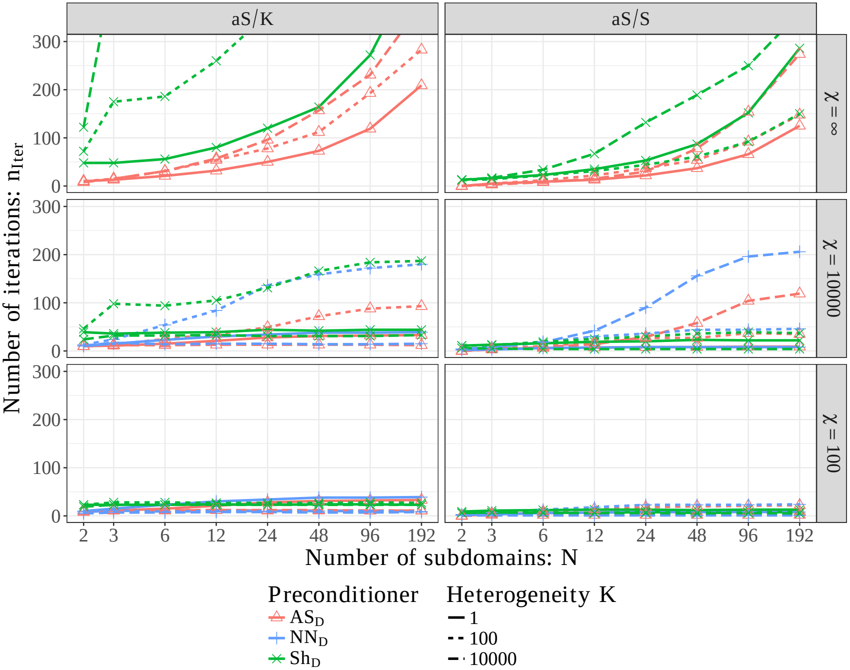

aS/K
χ = ∞
Condition Number: κ(MS)
1
10
100
1000
10000
2
3
6
12
24
48
96
192
Number of subdomains: N
Preconditioner
AS1
Sh1
Heterogeneity K
1
100
10000

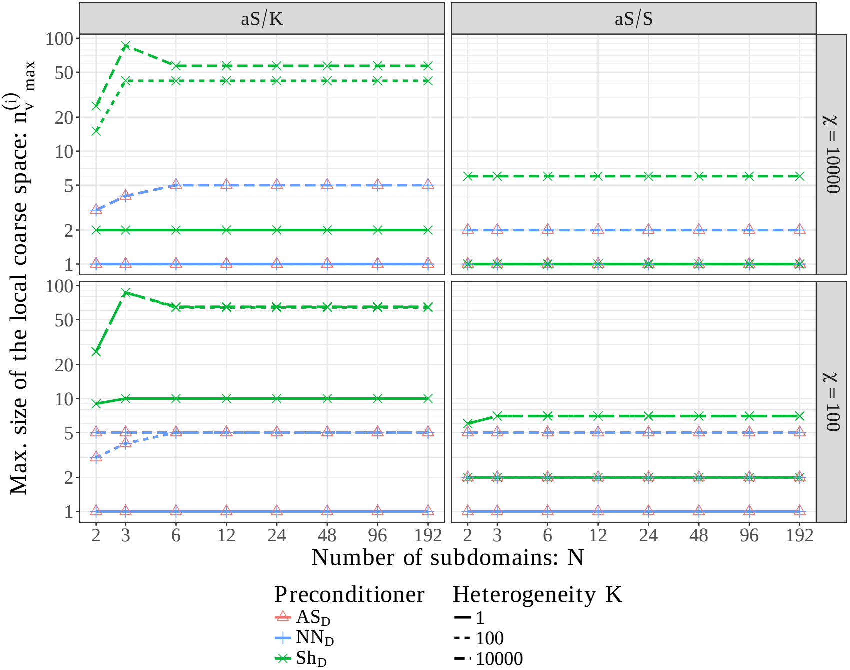


aS/K
Condition Number: κ(MS)
1
10
100
1000
10000
χ = ∞
χ = 10000
χ = 100
2
3
6
12
24
48
96
192
Number of subdomains: N
Preconditioner
AS_D
NN_D
Sh_D
Heterogeneity K
1
100
10000

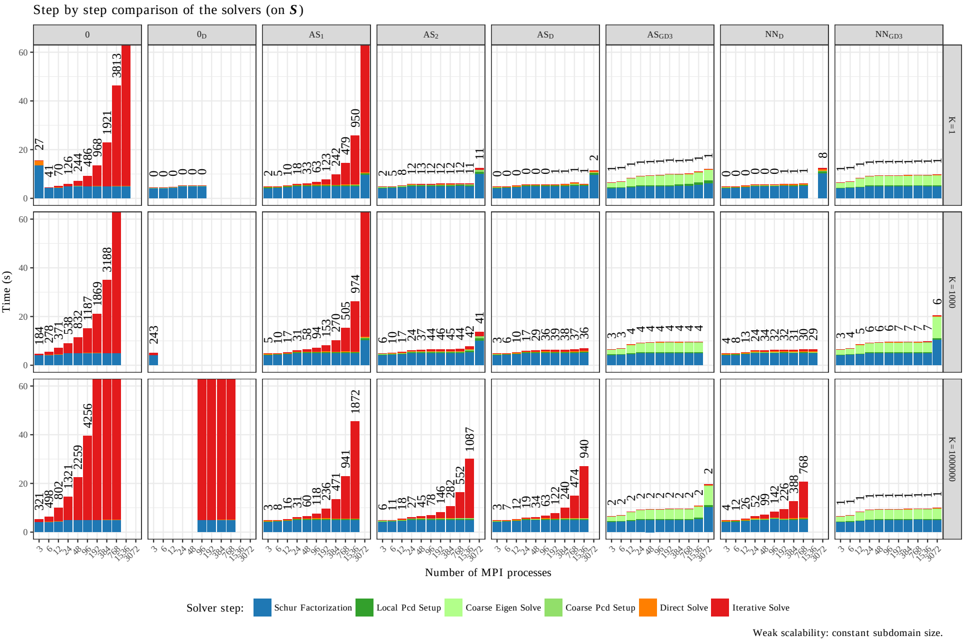


aS/K
aS/S
χ = ∞
χ = 10000
χ = 100
Condition Number: κ(MS)
10000
1000
100
10
1
2 3 6 12 24 48 96 192
Number of subdomains: N
Preconditioner
AS_D
NN_D
Sh_D
Heterogeneity K
1
100
10000

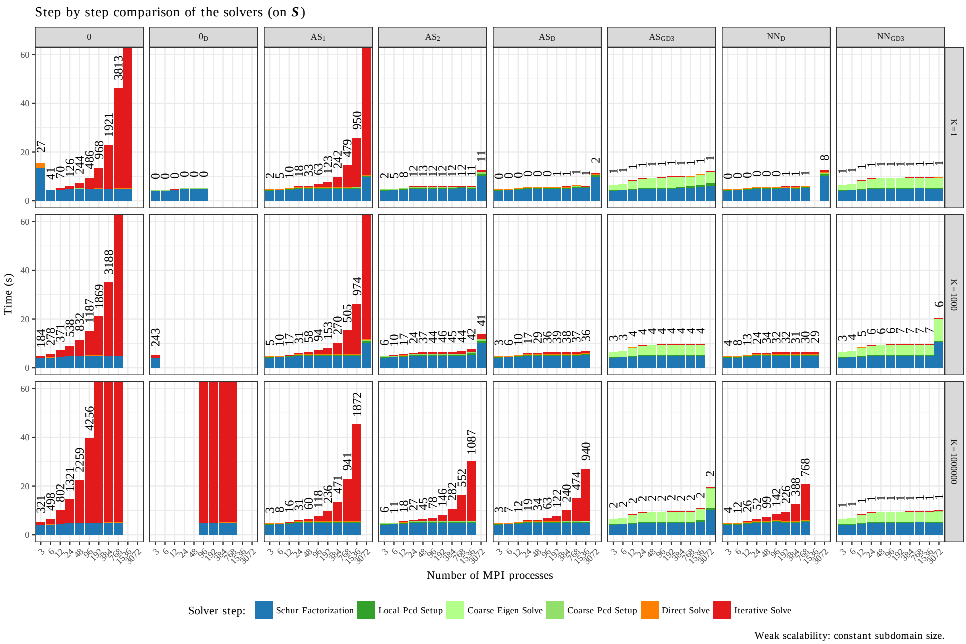

aS/K
aS/S
χ = ∞
χ = 10000
χ = 100
Number of iterations: n_Iter
300
200
100
0
2
3
6
12
24
48
96
192
Number of subdomains: N
Preconditioner
AS_D
NN_D
Sh_D
Heterogeneity K
1
100
10000



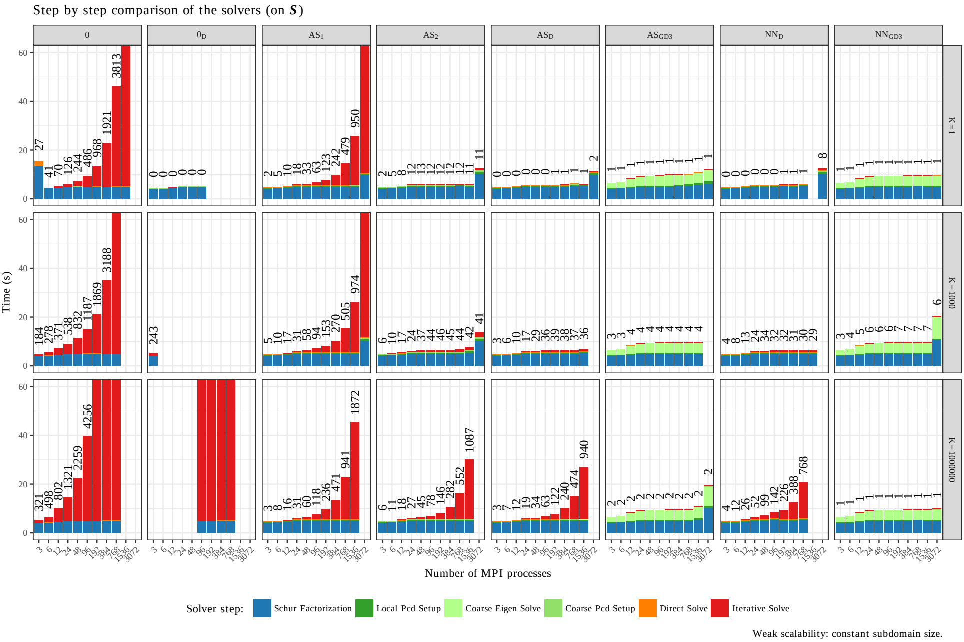
aS/K
aS/S
χ = 10000
χ = 100
Max. size of the local coarse space: $n_{v\ max}^{(i)}$
100
50
20
10
5
2
1
2
3
6
12
24
48
96
192
Number of subdomains: N
Preconditioner
AS_D
NN_D
Sh_D
Heterogeneity K
1
100
10000



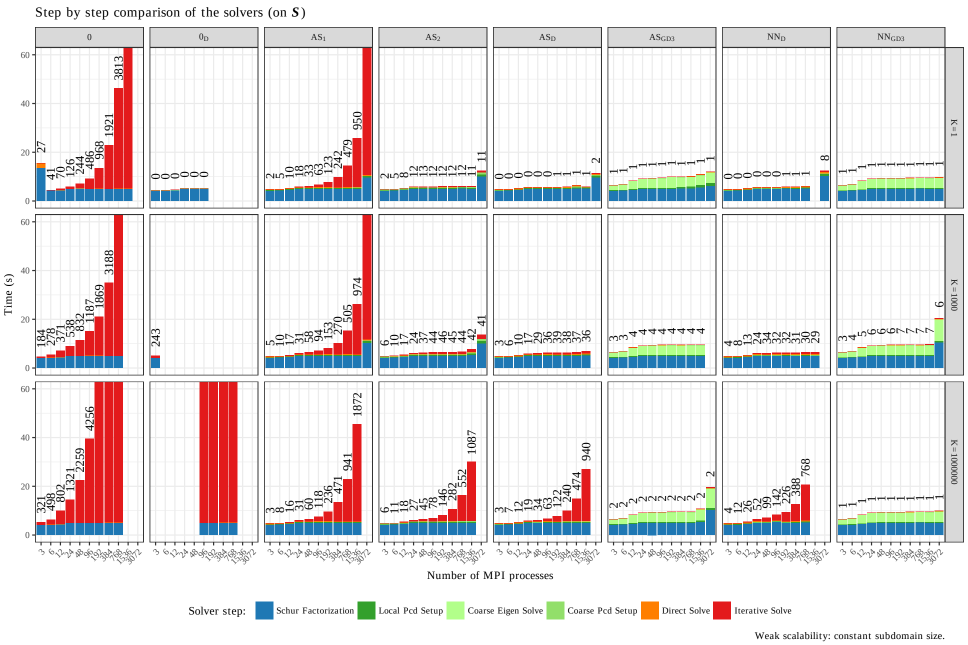
Time (s) - Number of subdomains/MPI processes $(3 - 3,072)$



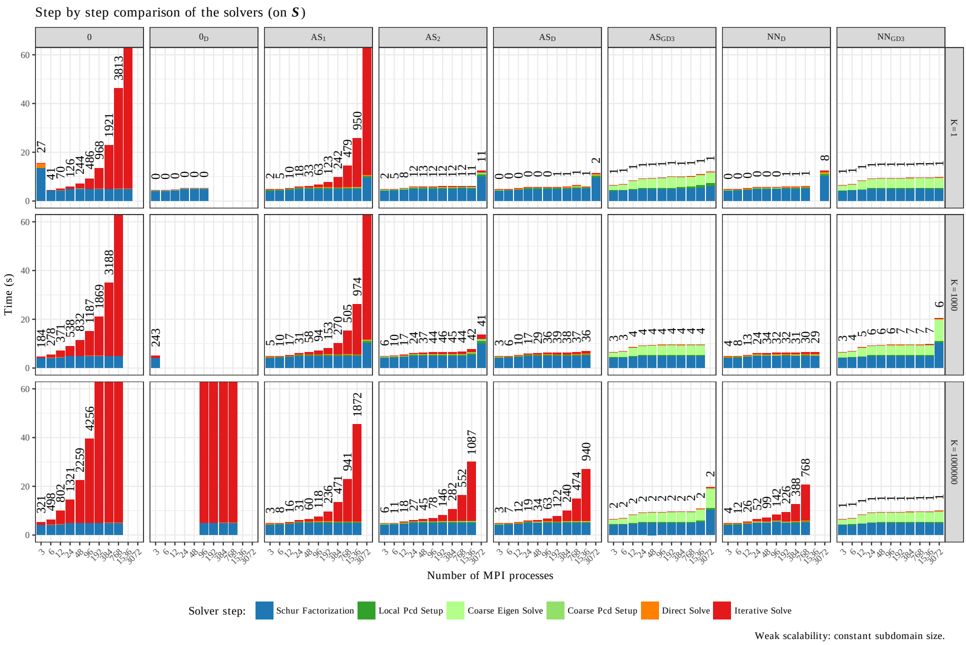
8 preconditioners



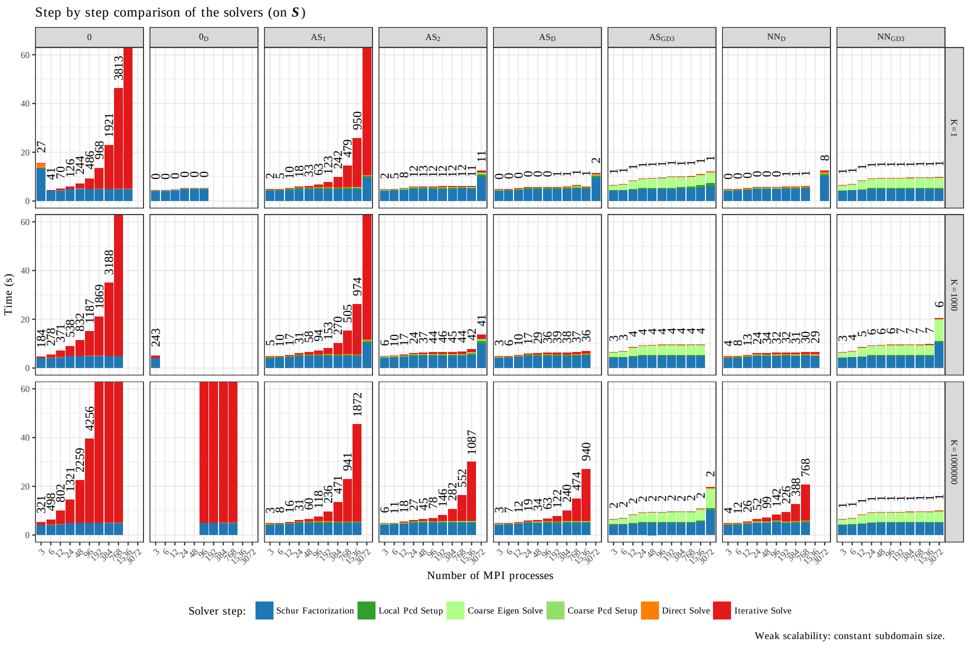
Heterogeneity K : 1, 1 000, 1 000 000



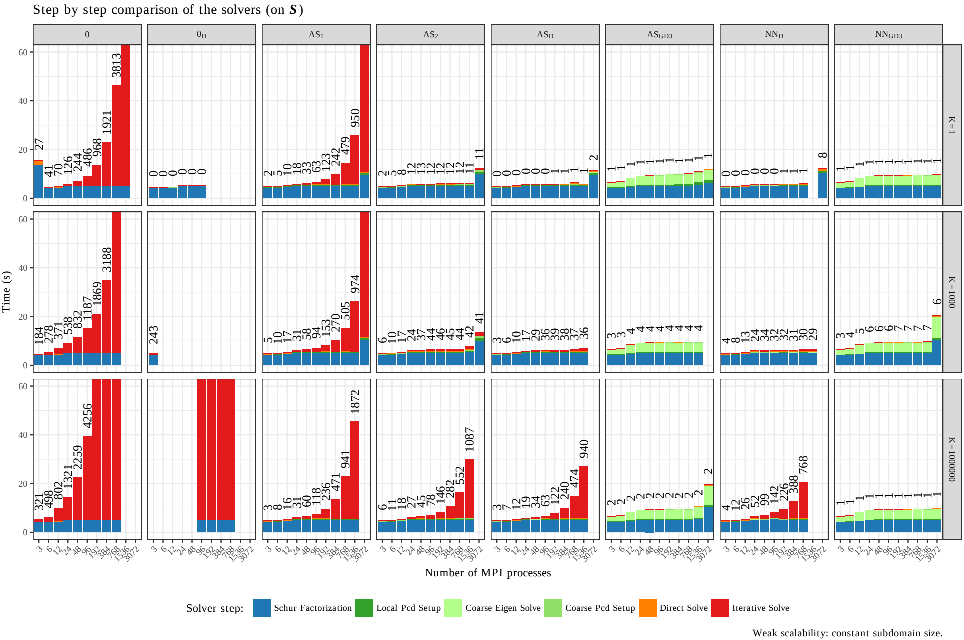
Total time to solution divided in 6 steps



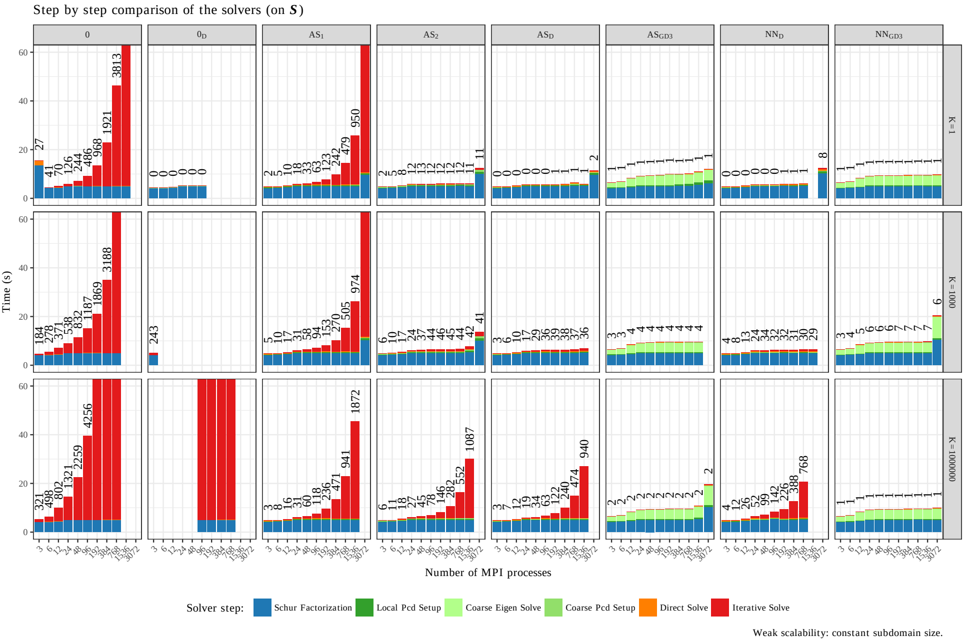
Number of iterations



One-level methods are not scalable (numerically *and* computationally)

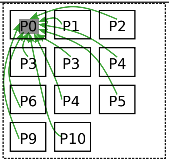
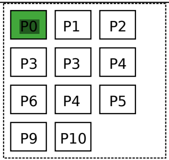


Two-level methods are scalable for $K \leq 1\,000$ (weak scalability)

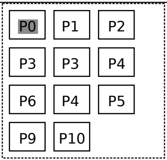
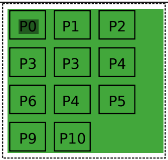


Two-level methods with an adaptative coarse space are fully scalable

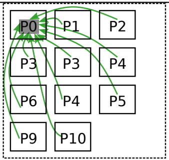
Parallel computing of $\mathcal{S}_0^{-1}$ - dense centralized

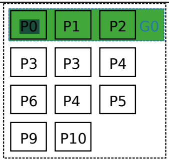
	Gather $\mathcal{S}_0$	Factorize $\mathcal{S}_0$
DC(1)	 gather(main)	 facto(master)

Parallel computing of $\mathcal{S}_0^{-1}$ - sparse distributed

	Gather $\mathcal{S}_0$	Factorize $\mathcal{S}_0$
SD(11)	 $\emptyset$	 facto(main)

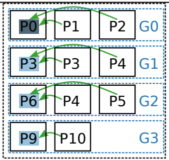
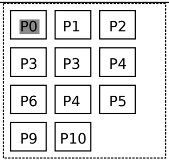
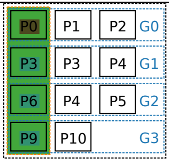
Parallel computing of $\mathcal{S}_0^{-1}$ - sparse centralized

<table>
<tr><td></td><td>Gather $\mathcal{S}_0$</td><td>Factorize $\mathcal{S}_0$</td></tr>
<tr><td>SC(3)</td><td>

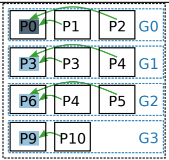
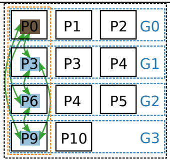
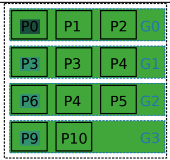
gather(main)</td><td>

facto(group)</td></tr>
<tr><td></td><td></td><td></td></tr>
</table>

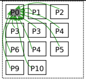
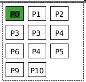
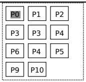
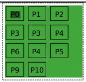
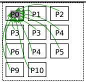
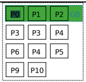
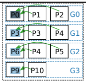
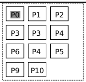
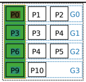
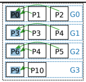
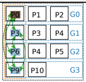
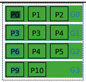
Parallel computing of $\mathcal{S}_0^{-1}$ - sparse hierarchical distributed

	Gather $\mathcal{S}_0$		Factorize $\mathcal{S}_0$
SHD(4)	 gather(groups)	 $\emptyset$	 facto(masters)

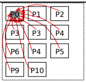
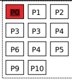
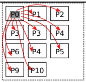
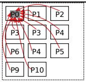
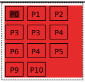
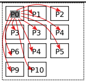
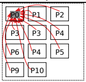
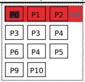
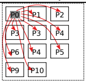
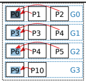
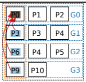
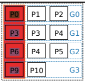
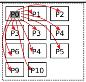
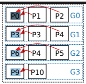
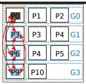
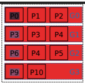
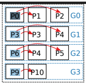
Parallel computing of $\mathcal{S}_0^{-1}$ - sparse replicated centralized

	Gather $\mathcal{S}_0$		Factorize $\mathcal{S}_0$
SRC(3)	 gather(groups)	 allgather(masters)	 facto(groups)

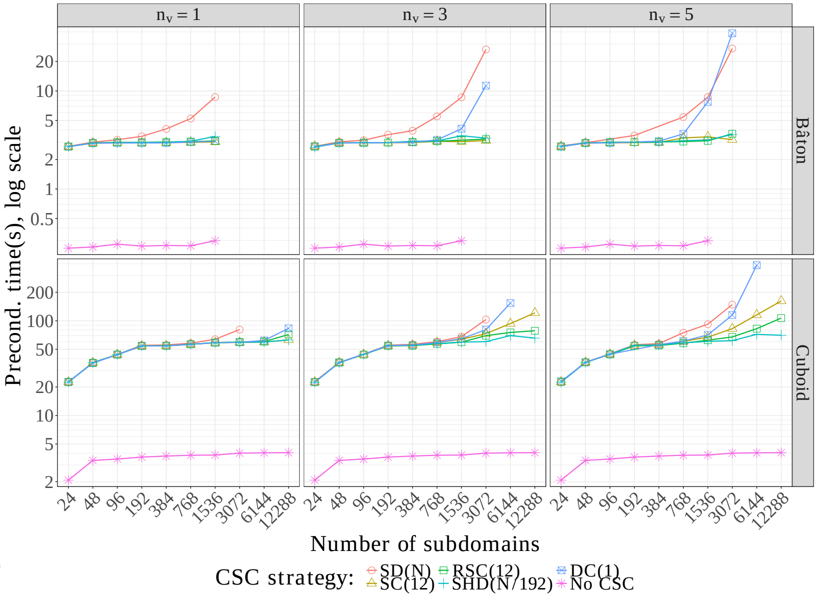
Parallel strategies for computing $\mathcal{S}_0^{-1}$

	Gather $\mathcal{S}_0$		Factorize $\mathcal{S}_0$
DC(1) dense centralized	 gather(main)		 facto(master)
SD(11) sparse distributed	 $\emptyset$		 facto(main)
SC(3) sparse centralized	 gather(main)		 facto(group)
SHD(4) sparse hierarchical distributed	 gather(groups)	 $\emptyset$	 facto(masters)
SRC(3) sparse replicated centralized	 gather(groups)	 allgather(masters)	 facto(groups)

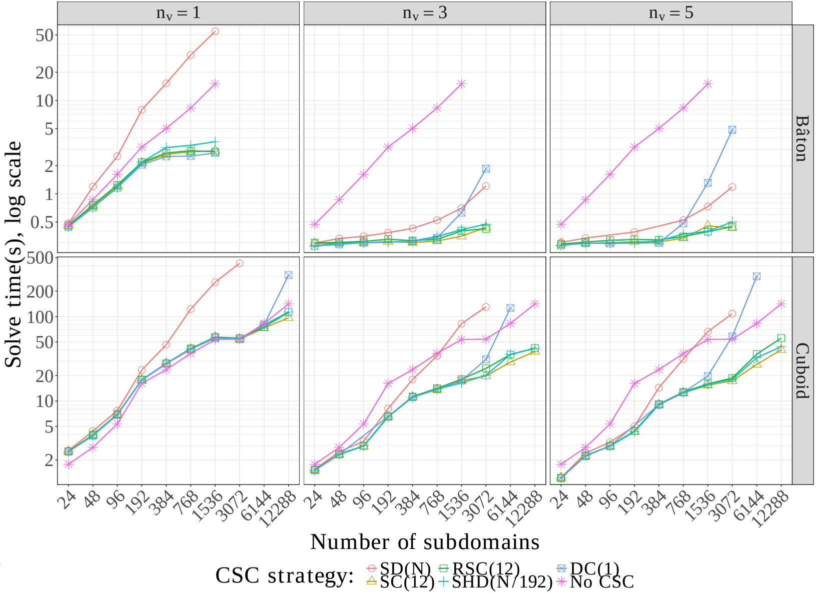
Parallel strategies for applying $\mathcal{S}_0^{-1}$

	Gather r_0		Solve $z_0 = \mathcal{S}_0^{-1} r_0$	Broadcast z_0
DC(1) dense centralized	 gather(main)		 solve(master)	 broadcast(main)
SD(11) sparse distributed	 gather(main)		 facto(main)	 broadcast(main)
SC(3) sparse centralized	 gather(main)		 solve(group)	 broadcast(main)
SHD(4) sparse hierarchical distributed	 gather(groups)	 gather(masters)	 solve(masters)	 broadcast(main)
SRC(3) sparse replicated centralized	 gather(groups)	 allgather(masters)	 solve(groups)	 broadcast(groups)

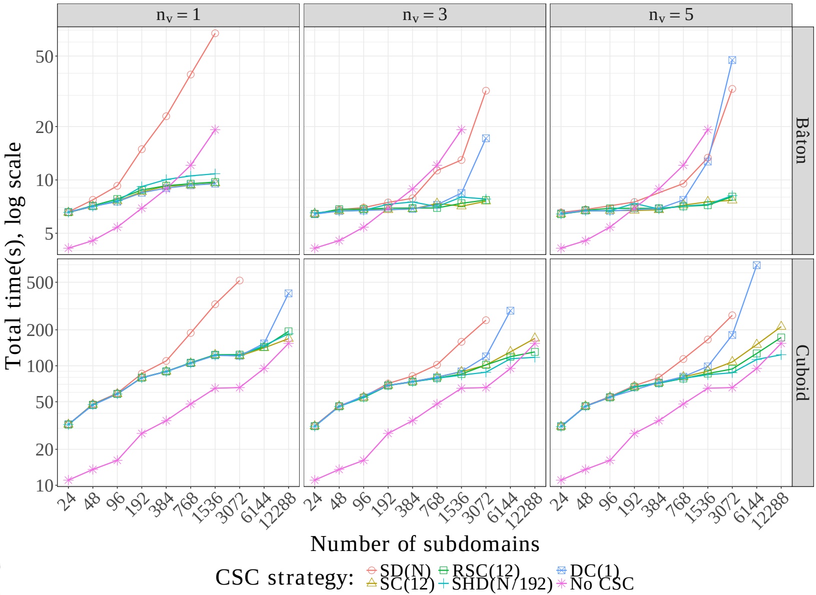
Heterogeneous ($k = 1/k = 10,000$) diffusion - Precond



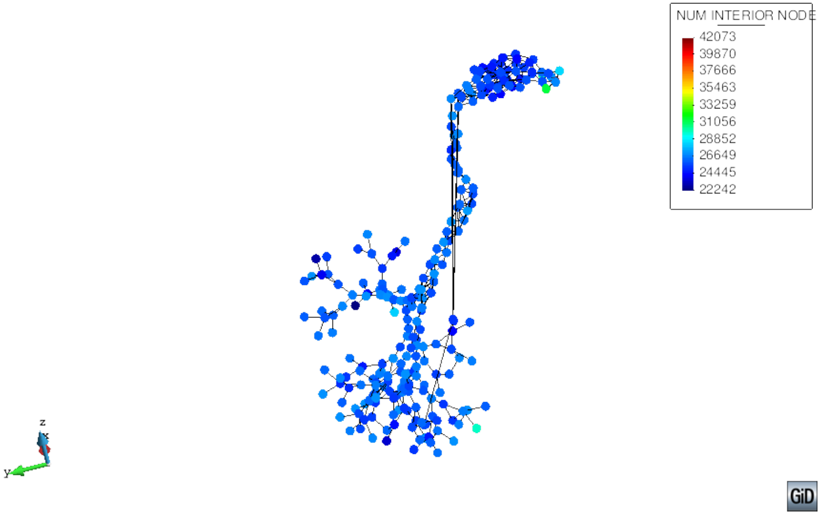
Heterogeneous ($k = 1/k = 10,000$) diffusion - Solve



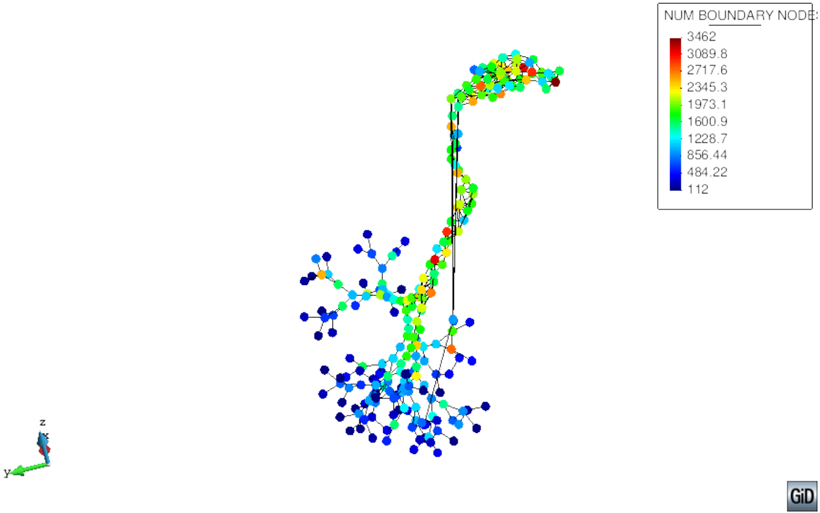
Heterogeneous ($k = 1/k = 10,000$) diffusion - Total (1 rhs)



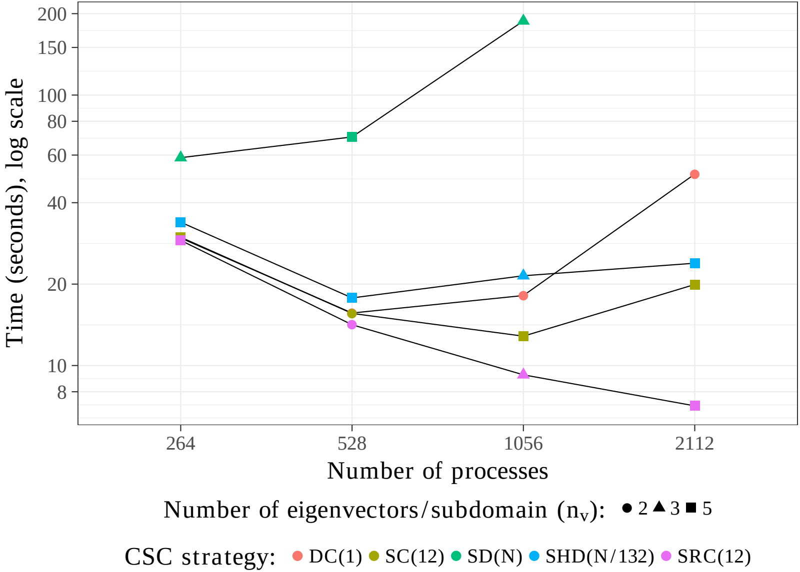
Respiratory test case (Alya@BSC) - interiors per subdomain



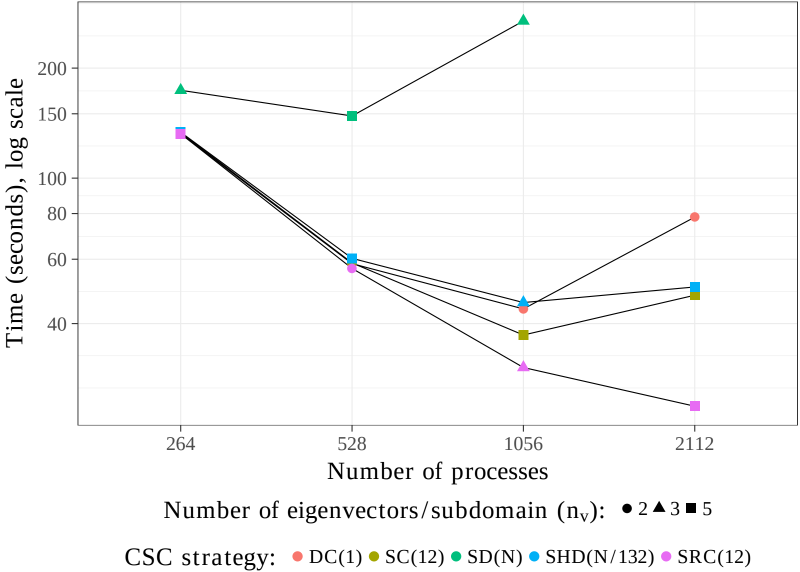
Respiratory test case - interface vertices per subdomain



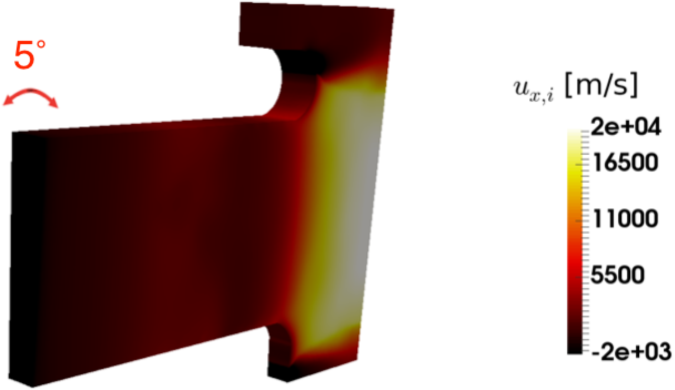
Respiratory test case - precond. application time



Respiratory test case - total time

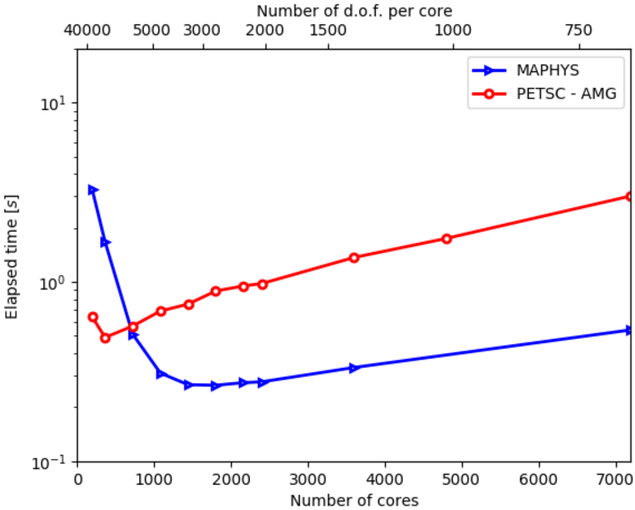


Avip: Plasma Propulsion Simulation (4.5 Mdof)



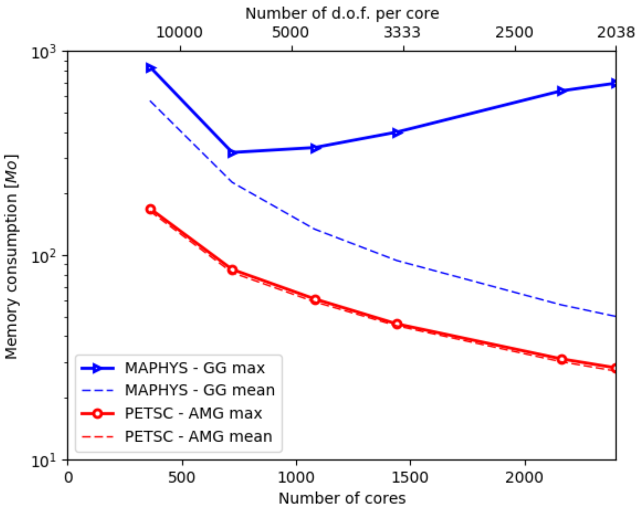
Courtesy: CERFACS/CFD

Avip: Plasma Propulsion Simulation (4.5 Mdof)



Courtesy: CERFACS/CFD

Avip: Plasma Propulsion Simulation (4.5 Mdof)



Courtesy: CERFACS/CFD

Thanks for your attention

- MaPHyS: F90, MPI+threads
- ddmpy: python, MPI
- MaPHyS++: C++, MPI, threads, task-based

https://gitlab.inria.fr/solverstack/maphys

Questions ?