ADAPTIVE OPTIMIZED SCHWARZ METHODS

Conor McCoid

Felix Kwok

Université Laval

SCHWARZ METHODS

Consider the following sample problem:

$$\Delta u(x, y) = f(x, y), \quad (x, y) \in \Omega = [-1, 1] \times [-1, 1]$$

$$u(x, y) = h(x, y), \quad (x, y) \in \partial\Omega$$

Consider the following sample problem:

$$\Delta u(x,y) = f(x,y), \quad (x,y) \in \Omega = [-1,1] \times [-1,1]$$

$$u(x,y) = h(x,y), \quad (x,y) \in \partial\Omega$$

We'll discretize this into

Consider the following sample problem:

$$\Delta u(x,y) = f(x,y), \quad (x,y) \in \Omega = [-1,1] \times [-1,1]$$

$$u(x,y) = h(x,y), \quad (x,y) \in \partial\Omega$$

We'll discretize this into

$$A\vec{u} = \vec{f}, \quad A \in \mathbb{R}^{N \times N}$$

$$A\vec{u} = \vec{f}, \quad A \in \mathbb{R}^{N \times N}$$

If N is very large, this can take a long time to find $\vec{u}$

$$A\vec{u} = \vec{f}, \quad A \in \mathbb{R}^{N \times N}$$

If N is very large, this can take a long time to find $\vec{u}$

Schwarz methods divide the domain into smaller problems

Schwarz methods divide the domain into smaller problems

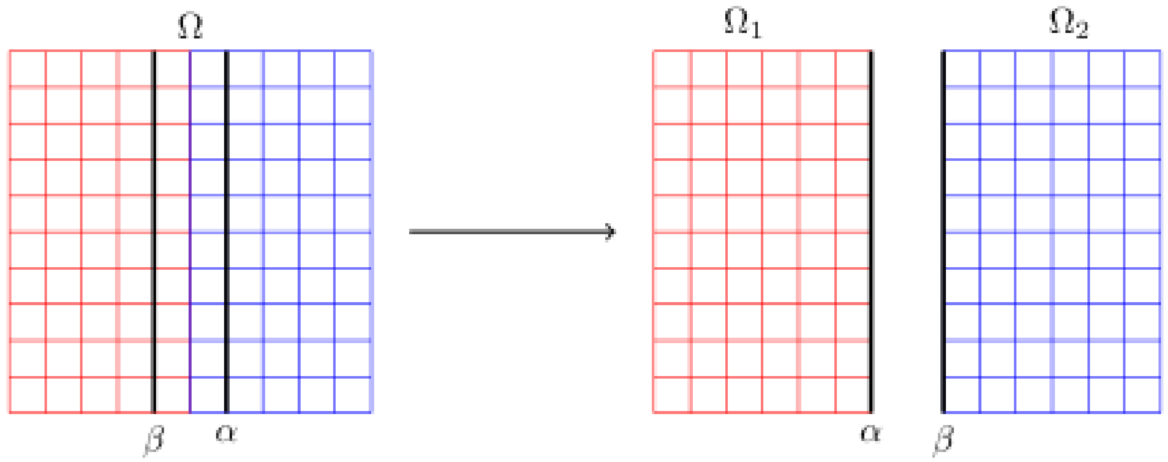
$$\Delta u_1^n(x,y) = f(x,y), \quad (x,y) \in \Omega_1 = [-1,\alpha] \times [-1,1]$$

$$u_1^n(\alpha,y) = u_2^{n-1}(\alpha,y)$$

$$\Delta u_2^n(x,y) = f(x,y), \quad (x,y) \in \Omega_2 = [\beta,1] \times [-1,1]$$

$$u_2^n(\beta,y) = u_1^{n-1}(\beta,y)$$

Schwarz methods divide the domain into smaller problems



Schwarz methods divide the domain into smaller problems

Multiplicative Schwarz: solve one subdomain after the other

Schwarz methods divide the domain into smaller problems

Multiplicative Schwarz: solve one subdomain after the other

Additive Schwarz: solve both subdomains at the same time

Schwarz methods divide the domain into smaller problems

$$\begin{bmatrix} A_{11} & A_{1\Gamma} & \\ A_{\Gamma 1} & A_{\Gamma\Gamma} & A_{\Gamma 2} \\ & A_{2\Gamma} & A_{22} \end{bmatrix} \begin{bmatrix} \vec{u}_1 \\ \vec{u}_\Gamma \\ \vec{u}_2 \end{bmatrix} = \begin{bmatrix} \vec{f}_1 \\ \vec{f}_\Gamma \\ \vec{f}_2 \end{bmatrix}$$

Schwarz methods divide the domain into smaller problems

$$\begin{bmatrix} A_{11} & A_{1\Gamma} \\ A_{\Gamma 1} & A_{\Gamma\Gamma} + T_{2\to 1} \end{bmatrix} \begin{bmatrix} \vec{u}_1^{n+1} \\ \vec{u}_{1\Gamma}^{n+1} \end{bmatrix} = \begin{bmatrix} \vec{f}_1 \\ \vec{f}_\Gamma \end{bmatrix} + \begin{bmatrix} \\ -A_{\Gamma 2}\vec{u}_2^n + T_{2\to 1}\vec{u}_{2\Gamma}^n \end{bmatrix}$$

$$\begin{bmatrix} A_{22} & A_{2\Gamma} \\ A_{\Gamma 2} & A_{\Gamma\Gamma} + T_{1\to 2} \end{bmatrix} \begin{bmatrix} \vec{u}_2^{n+1} \\ \vec{u}_{2\Gamma}^{n+1} \end{bmatrix} = \begin{bmatrix} \vec{f}_2 \\ \vec{f}_\Gamma \end{bmatrix} + \begin{bmatrix} \\ -A_{\Gamma 1}\vec{u}_1^n + T_{1\to 2}\vec{u}_{1\Gamma}^n \end{bmatrix}$$

OPTIMIZED SCHWARZ METHODS

Boundary conditions transmit information between the two subdomains

Boundary conditions transmit information between the two subdomains

We've used Dirichlet BCs

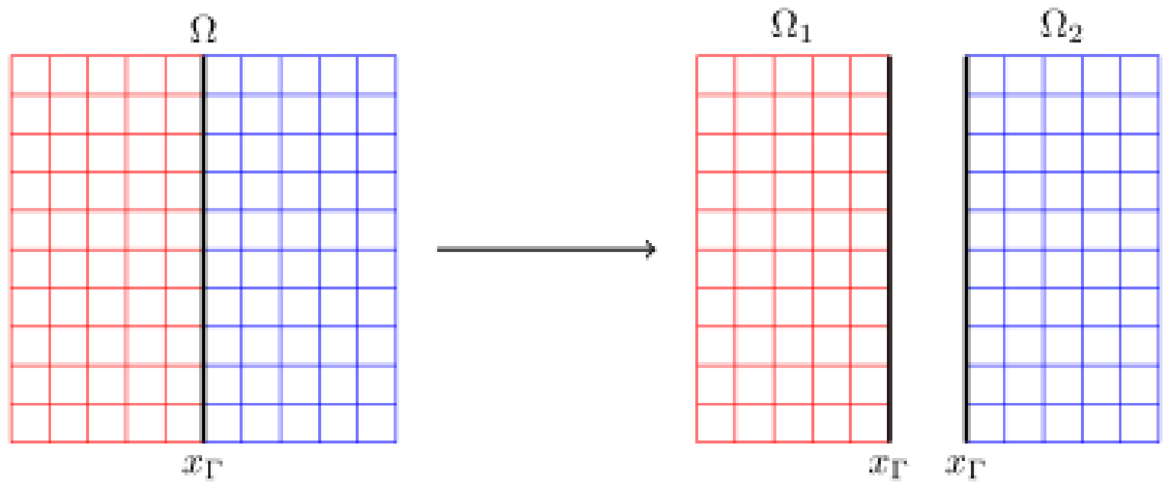
Boundary conditions transmit information between the two subdomains

We've used Dirichlet BCs

We could use Robin BCs instead

We could use Robin BCs instead

We could use Robin BCs instead



Now there's a choice in Robin parameter p, and this choice can be optimized

Now there's a choice in Robin parameter p, and this choice can be optimized

Optimization is done using Fourier analysis and finds the best p for all Fourier modes

Now there's a choice in Robin parameter p, and this choice can be optimized

Optimization is done using Fourier analysis and finds the best p for all Fourier modes

If you have a specific Fourier mode in mind, you can also pick the best p for just that mode

Robin BCs aren't the only option for optimized BCs

Robin BCs aren't the only option for optimized BCs

Tangential BCs are also a popular choice, and give a second parameter q to optimize

$$\frac{\partial u_1^n}{\partial x} - p u_1^n(0,y) + q \frac{\partial u_1^n}{\partial y} =$$

$$\frac{\partial u_2^{n-1}}{\partial x} - p u_2^{n-1}(0,y) + q \frac{\partial u_2^n}{\partial y}$$

But the best BCs are absorbing BCs, which are used in perfectly matched layers

These are dense, and correspond to Schur complements

$$T_{2\to 1} \to S_{2\to 1} = -A_{\Gamma 2} A_{22}^{-1} A_{2\Gamma},$$

$$T_{1\to 2} \to S_{1\to 2} = -A_{\Gamma 1} A_{11}^{-1} A_{1\Gamma}$$

These are dense, and correspond to Schur complements

$$T_{2\to 1} \to S_{2\to 1} = -A_{\Gamma 2} A_{22}^{-1} A_{2\Gamma},$$

$$T_{1\to 2} \to S_{1\to 2} = -A_{\Gamma 1} A_{11}^{-1} A_{1\Gamma}$$

This means they have about the same computation time as M iterations, where M is the size of the overlap

ADAPTIVE TRANSMISSION CONDITIONS

Finding optimized BCs requires Fourier analysis on each given problem

Finding optimized BCs requires Fourier analysis on each given problem

And the optimal BCs need Schur complements which are expensive to calculate

Finding optimized BCs requires Fourier analysis on each given problem

And the optimal BCs need Schur complements which are expensive to calculate

We want cheap, black box BCs

Finding optimized BCs requires Fourier analysis on each given problem

And the optimal BCs need Schur complements which are expensive to calculate

We want cheap, black box BCs

To get them, we'll find them adaptively

Let's look at the differences between iterates for a single sudomain

$$\begin{bmatrix} A_{11} & A_{1\Gamma} \\ A_{\Gamma 1} & A_{\Gamma\Gamma} + T_{2\to 1} \end{bmatrix} \left(\begin{bmatrix} \vec{u}_1^{n+1} \\ \vec{u}_{1\Gamma}^{n+1} \end{bmatrix} - \begin{bmatrix} \vec{u}_1^{n} \\ \vec{u}_{1\Gamma}^{n} \end{bmatrix} \right) =$$

$$\begin{bmatrix} \vec{f}_1 \\ \vec{f}_\Gamma \end{bmatrix} + \begin{bmatrix} \\ -A_{\Gamma 2}\vec{u}_2^n + T_{2\to 1}\vec{u}_{2\Gamma}^n \end{bmatrix} - \left(\begin{bmatrix} \vec{f}_1 \\ \vec{f}_\Gamma \end{bmatrix} + \begin{bmatrix} \\ -A_{\Gamma 2}\vec{u}_2^{n-1} + T_{2\to 1}\vec{u}_{2\Gamma}^{n-1} \end{bmatrix} \right)$$

Let's look at the differences between iterates for a single sudomain

$$\begin{bmatrix} A_{11} & A_{1\Gamma} \\ A_{\Gamma 1} & A_{\Gamma\Gamma} + T_{2\to 1} \end{bmatrix} \begin{bmatrix} \vec{d}_1^{n+1} \\ \vec{d}_{1\Gamma}^{n+1} \end{bmatrix} = \begin{bmatrix} \\ -A_{\Gamma 2}\vec{d}_2^n + T_{2\to 1}\vec{d}_{2\Gamma}^n \end{bmatrix}$$

We then perform what's known as static condensation by noting that

$$A_{11}\vec{d}_1^{n+1} = -A_{\Gamma 1}\vec{d}_{1\Gamma}^{n+1}$$

This leads to

$$(A_{\Gamma\Gamma} + S_{1\to 2} + T_{2\to 1})\vec{d}_{1\Gamma}^{n+1} = (T_{2\to 1} - S_{2\to 1})\vec{d}_{2}^{n}$$

This leads to

$$(A_{\Gamma\Gamma} + S_{1\to2} + T_{2\to1})\vec{d}_{1\Gamma}^{\,n+1} = (T_{2\to1} - S_{2\to1})\vec{d}_{2\Gamma}^{\,n}$$

$$\vec{y}^{\,n+1} = E_2\vec{d}_{2\Gamma}^{\,n} = -A_{\Gamma 2}\vec{d}_{2}^{\,n} + T_{2\to1}\vec{d}_{2\Gamma}^{\,n}$$

$$\vec{y}^{n+1} = E_2 \vec{d}_{2\Gamma}^{n} = -A_{\Gamma 2} \vec{d}_{2}^{n} + T_{2\to 1} \vec{d}_{2\Gamma}^{n}$$

At every other iteration, we're going to update E_2

$$\vec{y}^{n+1} = E_2\vec{d}_{2\Gamma}^{n} = -A_{\Gamma 2}\vec{d}_2^{n} + T_{2\to 1}\vec{d}_{2\Gamma}^{n}$$

At every other iteration, we're going to update E_2

$$E_2 \to E_2 - \frac{\vec{y}\vec{d}^{\top}}{||\vec{d}||^2}$$

$$E_2 \to E_2 - \frac{\vec{y}\vec{d}^\top}{||\vec{d}||^2}$$

With each iteration, there's a new pair of $\vec{y}$ and $\vec{d}$

$$E_2 \rightarrow E_2 - \frac{\vec{y}\vec{d}^\top}{||\vec{d}||^2}$$

With each iteration, there's a new pair of $\vec{y}$ and $\vec{d}$

We can apply a modified Gram-Schmidt process to the vectors $\vec{d}$, making the vectors $\vec{w}$

$$E_2 \rightarrow E_2 - \frac{\vec{y}\vec{d}^{\top}}{||\vec{d}||^2}$$

With each iteration, there's a new pair of $\vec{y}$ and $\vec{d}$

We can apply a modified Gram-Schmidt process to the vectors $\vec{d}$, making the vectors $\vec{w}$

Through this process, the vectors $\vec{y}$ get modified as well, to the vectors $\vec{v}$

$$E_2 \to E_2 - VW^\top$$

$$E_2 \to E_2 - VW^\top$$

Since $V = E_2 W$, this is equivalent to

$$E_2 \to E_2\left(I - WW^\top\right)$$

$$E_2 \to E_2 - VW^\top$$

Since $V = E_2 W$, this is equivalent to

$$E_2 \to E_2 \left(I - WW^\top\right)$$

We generate a low rank approximation of E_2

$$E_2 \to E_2 - VW^\top$$

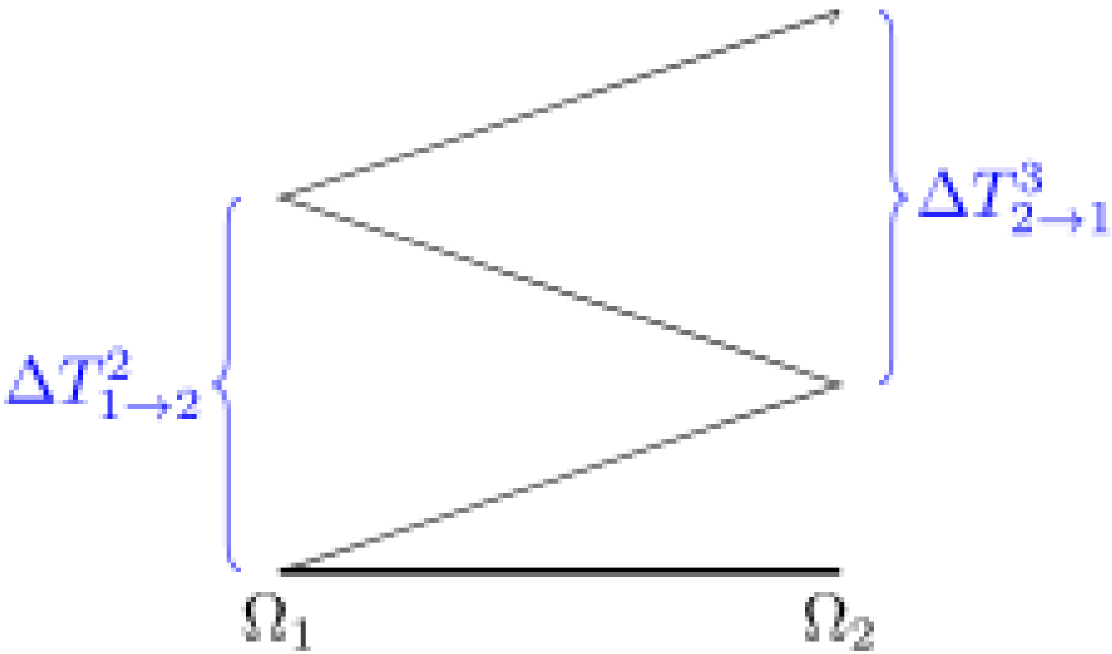
Since $V = E_2 W$, this is equivalent to

$$E_2 \to E_2 \left(I - WW^\top\right)$$

We generate a low rank approximation of E_2

To make the optimized BCs, we subtract this from $T_{2\to1}$

The transmission conditions $T_{2\to 1}$ and $T_{1\to 2}$ now change iteratively



Recall:

$$(A_{\Gamma\Gamma} + S_{1\to 2} + T_{2\to 1})\vec{d}_{1\Gamma}^{n+1} = (T_{2\to 1} - S_{2\to 1})\vec{d}_{2\Gamma}^{n}$$

The vectors $\vec{d}$ lie in a Krylov subspace

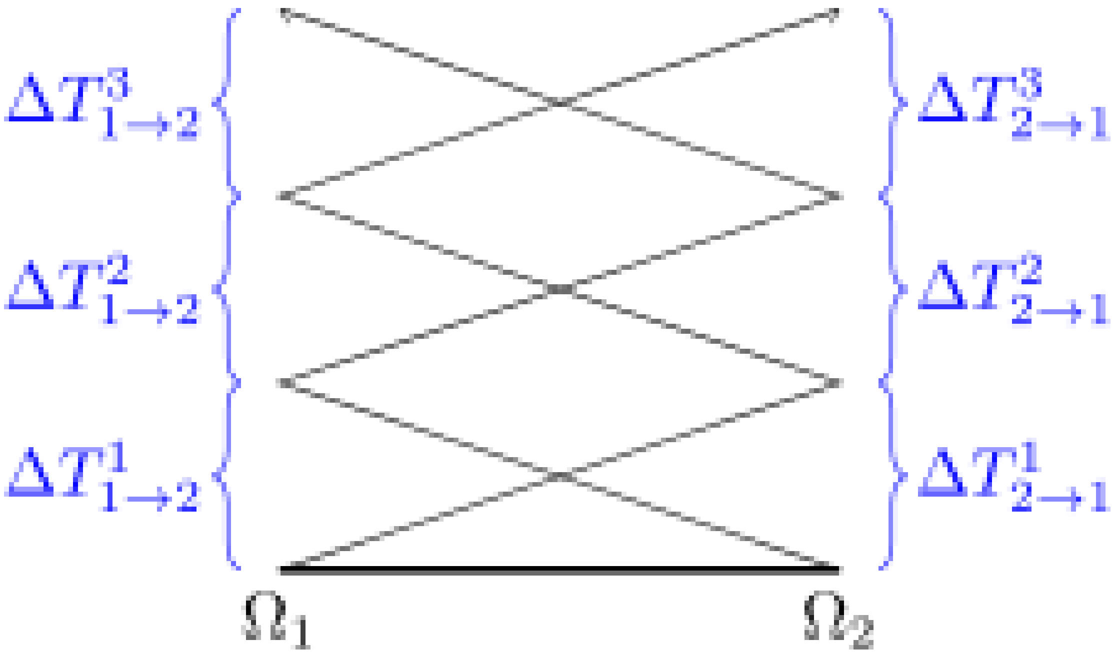
Recall:

$$(A_{\Gamma\Gamma} + S_{1\to 2} + T_{2\to 1})\vec{d}_{1\Gamma}^{n+1} = (T_{2\to 1} - S_{2\to 1})\vec{d}_{2\Gamma}^{n}$$

The vectors $\vec{d}$ lie in a Krylov subspace

They satisfy an implicit Galerkin condition

There's also the option to update the BCs at every iteration



ADAPTIVE OPTIMIZED SCHWARZ METHODS (AOSMS)

1. INITIAL GUESSES

1. INITIAL GUESSES

Make initial choices of $\vec{u}^0_{1\Gamma}, T^1_{1\to 2}$ and $T^1_{2\to 1}$

1. INITIAL GUESSES

Make initial choices of $\vec{u}_{1\Gamma}^{0}, T_{1\to 2}^{1}$ and $T_{2\to 1}^{1}$

Find $\vec{u}_1^0 = A_{11}^{-1}\left(\vec{f}_1 - A_{1\Gamma}\vec{u}_{1\Gamma}^{0}\right)$

1. INITIAL GUESSES

Make initial choices of $\vec{u}_{1\Gamma}^{0}, T_{1\to 2}^{1}$ and $T_{2\to 1}^{1}$

Find $\vec{u}_1^0 = A_{11}^{-1}\left(\vec{f}_1 - A_{1\Gamma}\vec{u}_{1\Gamma}^0\right)$

Solve

$$\begin{bmatrix} A_{22} & A_{2\Gamma} \\ A_{\Gamma 2} & A_{\Gamma\Gamma} + T_{1\to 2}^{1} \end{bmatrix} \begin{bmatrix} \vec{u}_2^1 \\ \vec{u}_{2\Gamma}^1 \end{bmatrix} = \begin{bmatrix} \vec{f}_2 \\ \vec{f}_\Gamma \end{bmatrix} + \begin{bmatrix} \\ -A_{\Gamma 1}\vec{u}_1^0 + T_{1\to 2}^{1}\vec{u}_{1\Gamma}^0 \end{bmatrix}$$

2. SEED KRYLOV SUBSPACE

2. SEED KRYLOV SUBSPACE

Solve

$$\begin{bmatrix} A_{11} & A_{1\Gamma} \\ A_{\Gamma 1} & A_{\Gamma\Gamma} + T^1_{2\to 1} \end{bmatrix} \begin{bmatrix} \vec{u}^2_1 \\ \vec{u}^2_{1\Gamma} \end{bmatrix} = \begin{bmatrix} \vec{f}_1 \\ \vec{f}_\Gamma \end{bmatrix} + \begin{bmatrix} \\ -A_{\Gamma 2}\vec{u}^1_2 + T^1_{2\to 1}\vec{u}^1_{2\Gamma} \end{bmatrix}$$

2. SEED KRYLOV SUBSPACE

Solve

$$\begin{bmatrix} A_{11} & A_{1\Gamma} \\ A_{\Gamma 1} & A_{\Gamma\Gamma} + T^1_{2\to 1} \end{bmatrix} \begin{bmatrix} \vec{u}^2_1 \\ \vec{u}^2_{1\Gamma} \end{bmatrix} = \begin{bmatrix} \vec{f}_1 \\ \vec{f}_\Gamma \end{bmatrix} + \begin{bmatrix} \\ -A_{\Gamma 2}\vec{u}^1_2 + T^1_{2\to 1}\vec{u}^1_{2\Gamma} \end{bmatrix}$$

Calculate $\vec{d}^2_{1\Gamma} = \vec{u}^2_{1\Gamma} - \vec{u}^0_{1\Gamma}$ and $\vec{d}^2_1 = \vec{u}^2_1 - \vec{u}^0_1$

2. SEED KRYLOV SUBSPACE

Calculate $\vec{d}_{1\Gamma}^{2} = \vec{u}_{1\Gamma}^{2} - \vec{u}_{1\Gamma}^{0}$ and $\vec{d}_{1}^{2} = \vec{u}_{1}^{2} - \vec{u}_{1}^{0}$

2. SEED KRYLOV SUBSPACE

Calculate $\vec{d}^2_{1\Gamma} = \vec{u}^2_{1\Gamma} - \vec{u}^0_{1\Gamma}$ and $\vec{d}^2_1 = \vec{u}^2_1 - \vec{u}^0_1$

Normalize $\vec{d}^2_{1\Gamma}$ using α^2_1 such that $\vec{w}^2_1 = \alpha^2_1 \vec{d}^2_{1\Gamma}$ and calculate

$$\vec{v}^2_1 = \alpha^2_1 \left(-A_{\Gamma 1} \vec{d}^2_1 + T^1_{1\rightarrow 2} \vec{d}^2_{1\Gamma} \right)$$

2. SEED KRYLOV SUBSPACE

Calculate $\vec{d}_{1\Gamma}^{2} = \vec{u}_{1\Gamma}^{2} - \vec{u}_{1\Gamma}^{0}$ and $\vec{d}_{1}^{2} = \vec{u}_{1}^{2} - \vec{u}_{1}^{0}$

Normalize $\vec{d}_{1\Gamma}^{2}$ using α_1^2 such that $\vec{w}_1^2 = \alpha_1^2 \vec{d}_{1\Gamma}^{2}$ and calculate

$$\vec{v}_1^2 = \alpha_1^2 \left(-A_{\Gamma 1} \vec{d}_1^{2} + T_{1\to 2}^{1} \vec{d}_{1\Gamma}^{2} \right)$$

Update $T_{1\to 2}^{1}$:

$$T_{1\to 2}^{2} = T_{1\to 2}^{1} + \Delta T_{1\to 2}^{2} = T_{1\to 2}^{1} - \vec{v}_1^2 \left(\vec{w}_1^2 \right)^{\top}$$

2. SEED KRYLOV SUBSPACE

Solve

$$\begin{bmatrix} A_{22} & A_{2\Gamma} \\ A_{\Gamma 2} & A_{\Gamma\Gamma} + T^2_{1\to 2} \end{bmatrix} \begin{bmatrix} \vec{d}^3_2 \\ \vec{d}^3_{2\Gamma} \end{bmatrix}$$

$$= \begin{bmatrix} \\ -A_{\Gamma 1}\vec{d}^2_1 + T^2_{1\to 2}\vec{d}^2_{1\Gamma} - \Delta T^2_{1\to 2}\left(\vec{u}^1_{2\Gamma} - \vec{u}^0_{1\Gamma}\right) \end{bmatrix}$$

2. SEED KRYLOV SUBSPACE

Solve

$$\begin{bmatrix} A_{22} & A_{2\Gamma} \\ A_{\Gamma 2} & A_{\Gamma\Gamma} + T^2_{1\to 2} \end{bmatrix} \begin{bmatrix} \vec{d}^3_2 \\ \vec{d}^3_{2\Gamma} \end{bmatrix}$$

$$= \langle \vec{w}^2_1, \vec{u}^1_{2\Gamma} - \vec{u}^0_{1\Gamma} \rangle \begin{bmatrix} \\ \vec{v}^2_1 \end{bmatrix}$$

2. SEED KRYLOV SUBSPACE

Normalize $\vec{d}^3_{2\Gamma}$ using α^3_2 such that $\vec{w}^3_2 = \alpha^3_2 \vec{d}^3_{2\Gamma}$ and calculate

$$\vec{v}^3_2 = \alpha^3_2 \left(-A_{\Gamma 2} \vec{d}^3_2 + T^1_{2\to 1} \vec{d}^3_{2\Gamma} \right)$$

Update $T^1_{2\to 1}$:

$$T^3_{2\to 1} = T^1_{2\to 1} - \vec{v}^3_2 \left(\vec{w}^3_2 \right)^\top$$

3. ITERATE

3. ITERATE

Solve for $\vec{d}_i^n$ and $\vec{d}_{i\Gamma}^n$

3. ITERATE

Solve for $\vec{d}_i^n$ and $\vec{d}_{i\Gamma}^n$

Apply modified Gram-Schmidt to find $\vec{v}_i^n$ and $\vec{w}_i^n$

3. ITERATE

Solve for $\vec{d}_i^n$ and $\vec{d}_{i\Gamma}^n$

Apply modified Gram-Schmidt to find $\vec{v}_i^n$ and $\vec{w}_i^n$

Update $T_{i\rightarrow j}^n$ using $\vec{v}_i^n \left(\vec{w}_i^n\right)^\top$

WOODBURY MATRIX IDENTITY

The updates to $T^n_{1\to 2}$ and $T^n_{2\to 1}$ are low rank

The updates to $T_{1\to2}^n$ and $T_{2\to1}^n$ are low rank

In order to preserve any matrix factorizations, we can use the Woodbury matrix identity to apply these low rank updates

The updates to $T^n_{1\to 2}$ and $T^n_{2\to 1}$ are low rank

In order to preserve any matrix factorizations, we can use the Woodbury matrix identity to apply these low rank updates

$$(A - VW^\top)\vec{u} = \vec{f},$$

$$\vec{u} = A^{-1}\vec{f} + A^{-1}V(I_{k\times k} - W^\top A^{-1}V)^{-1}W^\top A$$

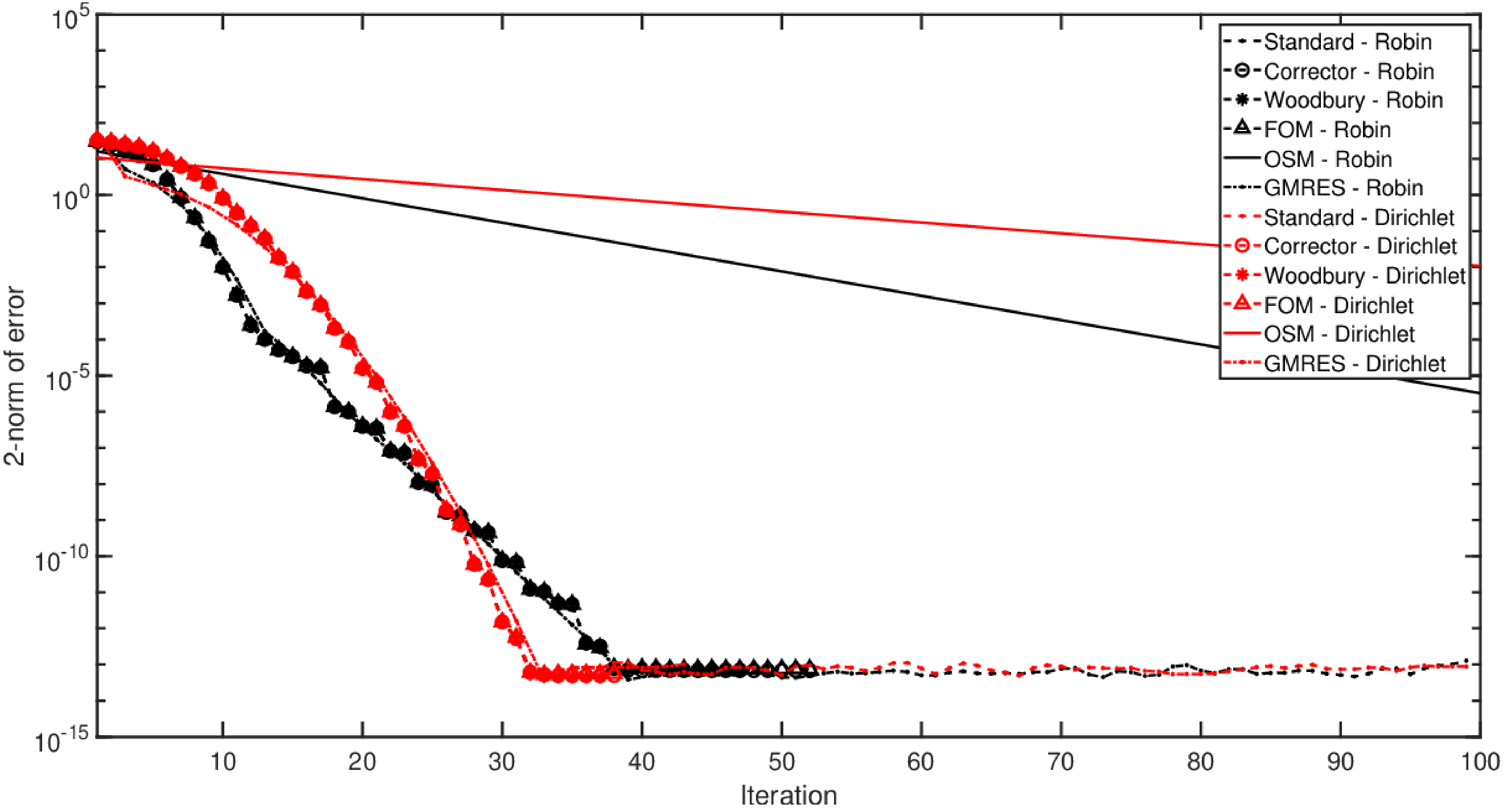
NUMERICAL EXPERIMENTS

Recall the sample problem:

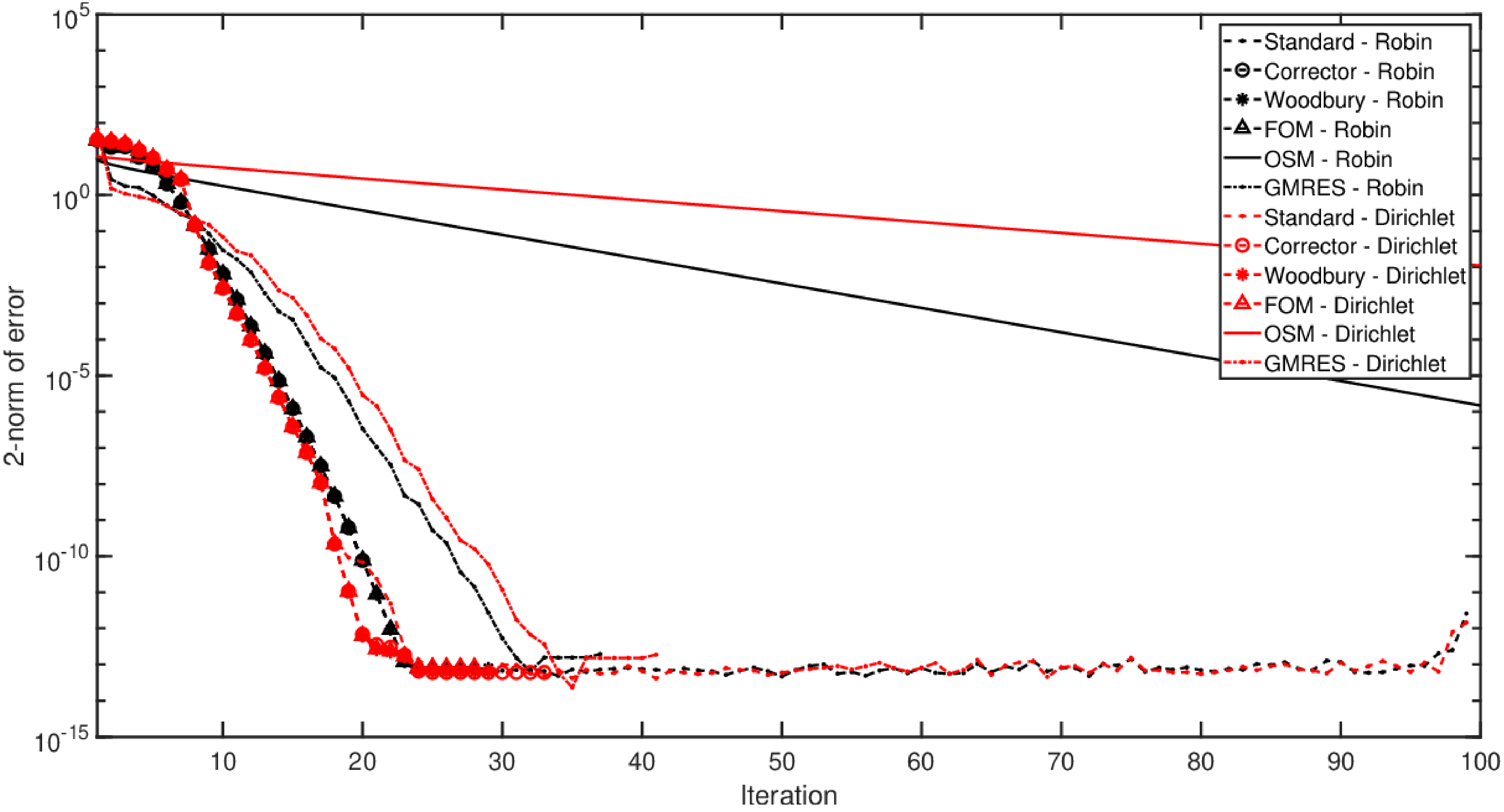
$$\Delta u(x, y) = f(x, y), \quad (x, y) \in \Omega = [-1, 1] \times [-1, 1]$$

$$u(x, y) = h(x, y), \quad (x, y) \in \partial\Omega$$

Let's apply AOSMs to this problem



Standard - Robin
Corrector - Robin
Woodbury - Robin
FOM - Robin
OSM - Robin
GMRES - Robin
Standard - Dirichlet
Corrector - Dirichlet
Woodbury - Dirichlet
FOM - Dirichlet
OSM - Dirichlet
GMRES - Dirichlet
2-norm of error
Iteration
10^{5}
10^{0}
10^{-5}
10^{-10}
10^{-15}
10
20
30
40
50
60
70
80
90
100



Standard - Robin
Corrector - Robin
Woodbury - Robin
FOM - Robin
OSM - Robin
GMRES - Robin
Standard - Dirichlet
Corrector - Dirichlet
Woodbury - Dirichlet
FOM - Dirichlet
OSM - Dirichlet
GMRES - Dirichlet
2-norm of error
Iteration

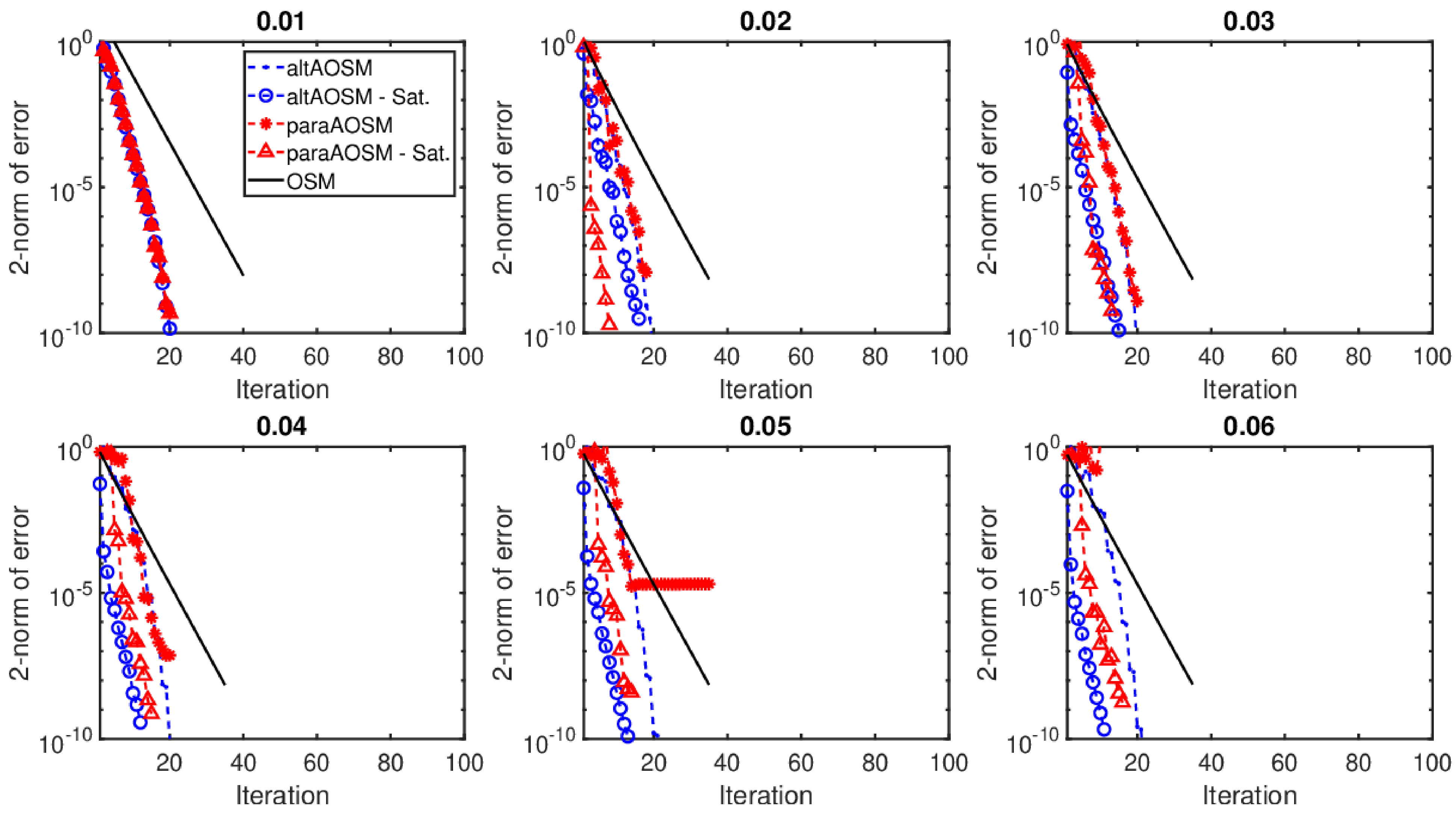
$$u_t(x,y,t) = \Delta u(x,y,t), \quad (x,y) \in \Omega = [-1,1] \times [-1,1],\ t \in [0,T]$$

$$u(x,y,0) = u_0(x,y), \quad (x,y) \in \Omega,$$

$$u(x,y,t) = h(x,y), \quad (x,y) \in \partial\Omega,\ t \in [0,T]$$

$$\frac{u_{n+1} - u_n}{\Delta t} = A u_{n+1}$$

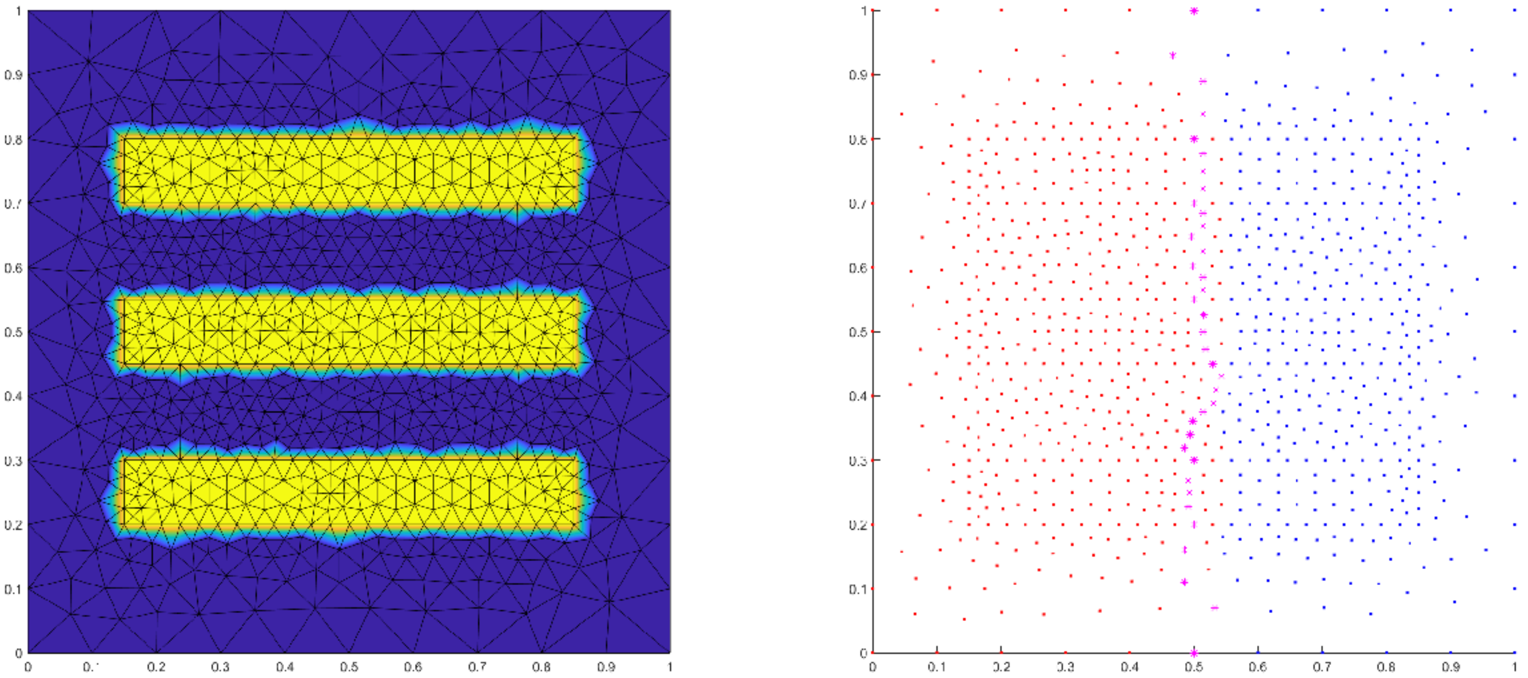
$$(I - \Delta t A) u_{n+1} = u_n$$


0.01
0.02
0.03
0.04
0.05
0.06
altAOSM
altAOSM - Sat.
paraAOSM
paraAOSM - Sat.
OSM
2-norm of error
Iteration
10^{0}
10^{-5}
10^{-10}
20
40
60
80
100

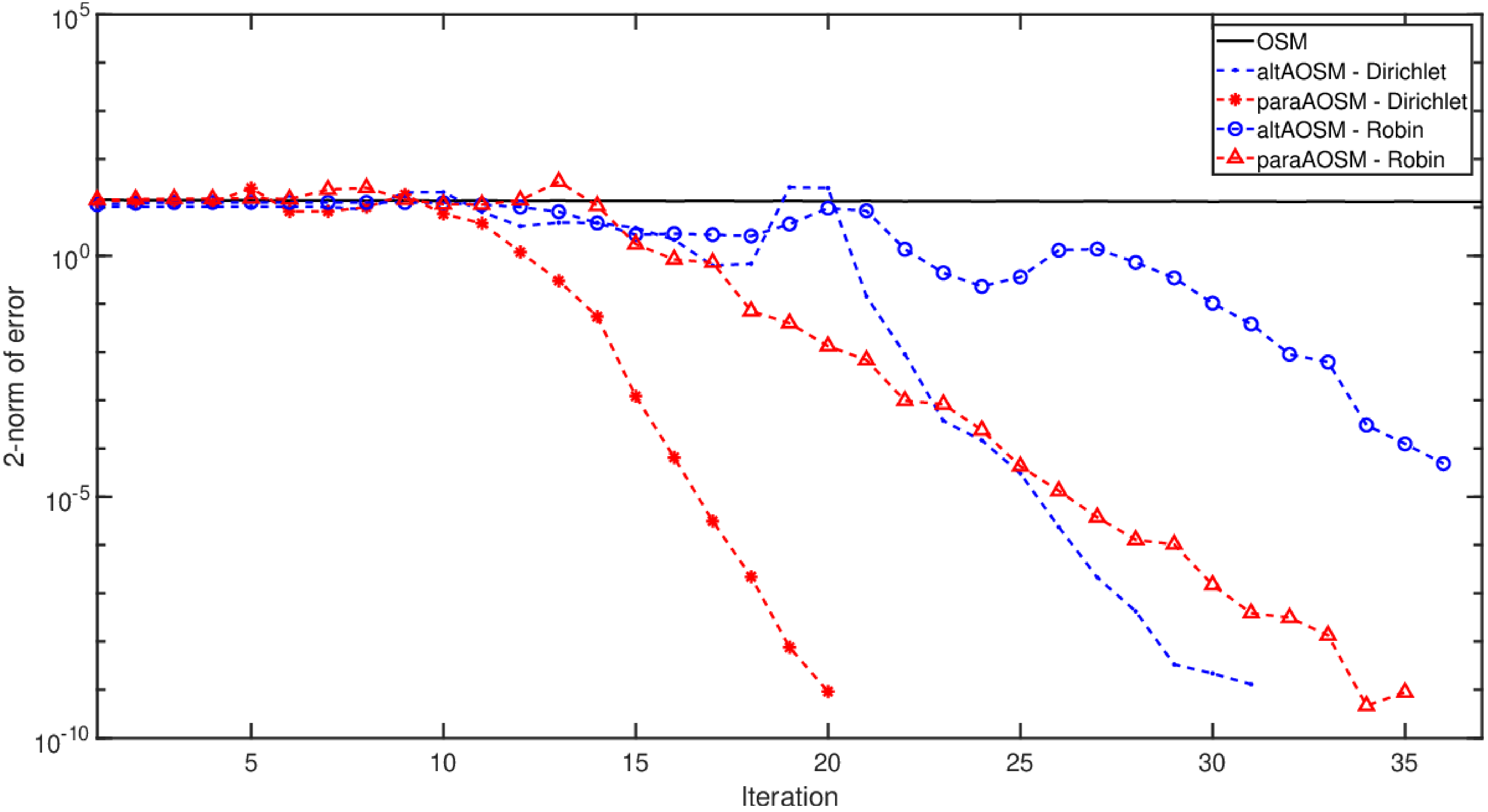
$$\nabla(\alpha(x,y)\cdot\nabla u(x,y)) = f(x,y), \quad (x,y)\in\Omega = [-1,1]\times[-1,1],$$

$$u(x,y) = h(x,y), \quad (x,y)\in\partial\Omega,$$

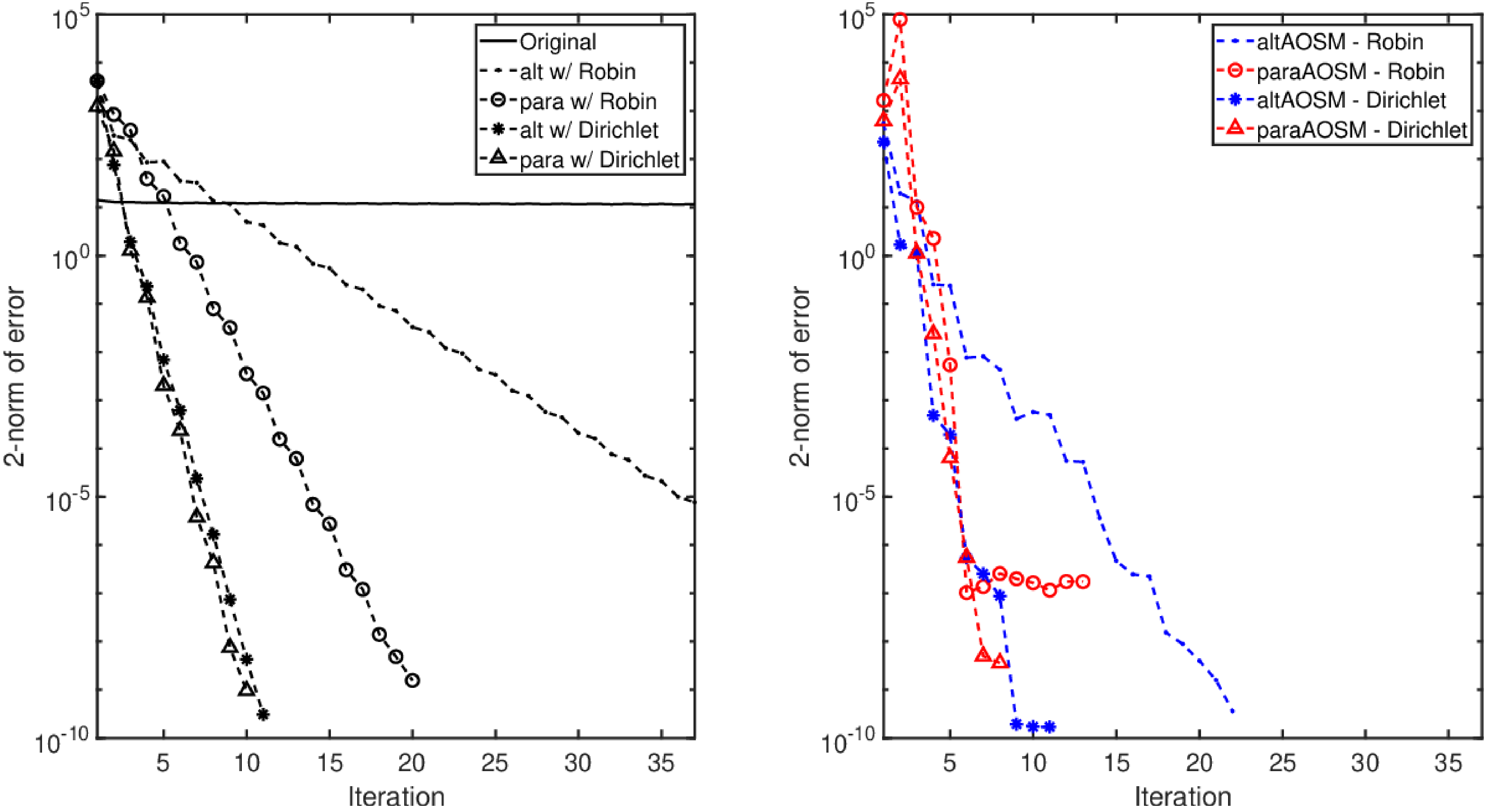
Solve this using FEM software



0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1



OSM
altAOSM - Dirichlet
paraAOSM - Dirichlet
altAOSM - Robin
paraAOSM - Robin
10^5
10^0
10^{-5}
10^{-10}
2-norm of error
5
10
15
20
25
30
35
Iteration



Original
alt w/ Robin
para w/ Robin
alt w/ Dirichlet
para w/ Dirichlet
2-norm of error
Iteration
altAOSM - Robin
paraAOSM - Robin
altAOSM - Dirichlet
paraAOSM - Dirichlet
2-norm of error
Iteration

CONCLUSIONS

CONCLUSIONS

- AOSMs give GMRES convergence without GMRES cost

CONCLUSIONS

- AOSMs give GMRES convergence without GMRES cost
- Transmission conditions can be re-used, for restarts and time steps

CONCLUSIONS

- AOSMs give GMRES convergence without GMRES cost
- Transmission conditions can be re-used, for restarts and time steps

FUTURE WORK

CONCLUSIONS

- AOSMs give GMRES convergence without GMRES cost
- Transmission conditions can be re-used, for restarts and time steps

FUTURE WORK

- Track down stability issues

CONCLUSIONS

- AOSMs give GMRES convergence without GMRES cost
- Transmission conditions can be re-used, for restarts and time steps

FUTURE WORK

- Track down stability issues
- Test out other choices of adaptive transmission conditions