Preconditioning Techniques for Saddle Point Problems

Michele Benzi

Department of Mathematics and Computer Science
Emory University
Atlanta, GA

Woudschoten, 5 October 2012

Outline

Outline

Outline

Outline

Outline

The need for efficient and robust solvers

Many scientific and engineering applications require the solution of large, sparse, ill-conditioned saddle point problems.

The need for efficient and robust solvers

Many scientific and engineering applications require the solution of large, sparse, ill-conditioned saddle point problems.

Much progress has been made in the last 20 years, but challenges remain.

The need for efficient and robust solvers

Many scientific and engineering applications require the solution of large, sparse, ill-conditioned saddle point problems.

Much progress has been made in the last 20 years, but challenges remain.

For example, solving the Navier-Stokes equations still presents serious difficulties. Steady-state problems with low viscosity discretized on stretched meshes are still very hard to solve efficiently.

The need for efficient and robust solvers

Many scientific and engineering applications require the solution of large, sparse, ill-conditioned saddle point problems.

Much progress has been made in the last 20 years, but challenges remain.

For example, solving the Navier-Stokes equations still presents serious difficulties. Steady-state problems with low viscosity discretized on stretched meshes are still very hard to solve efficiently.

Research in this area is still very much ongoing!

Results for Kay, Loghin and Wathen preconditioner

Test problems: steady Oseen, homogeneous Dirichlet BCs, two choices of the wind function.

Results for Kay, Loghin and Wathen preconditioner

Test problems: steady Oseen, homogeneous Dirichlet BCs, two choices of the wind function.

- A constant wind problem: $\mathbf{w} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

Results for Kay, Loghin and Wathen preconditioner

Test problems: steady Oseen, homogeneous Dirichlet BCs, two choices of the wind function.

- A constant wind problem: $\mathbf{w} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
- A recirculating flow (vortex) problem: $\mathbf{w} = \begin{pmatrix} 4(2y-1)(1-x)x \\ -4(2x-1)(1-y)y \end{pmatrix}$

Results for Kay, Loghin and Wathen preconditioner

Test problems: steady Oseen, homogeneous Dirichlet BCs, two choices of the wind function.

- A constant wind problem: $\mathbf{w} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
- A recirculating flow (vortex) problem: $\mathbf{w} = \begin{pmatrix} 4(2y-1)(1-x)x \\ -4(2x-1)(1-y)y \end{pmatrix}$

Uniform FEM discretizations: isoP2-P0 and isoP2-P1. These discretizations satisfy the inf-sup condition: no pressure stabilization is needed.
SUPG stabilization is used for the velocities.

Results for Kay, Loghin and Wathen preconditioner

Test problems: steady Oseen, homogeneous Dirichlet BCs, two choices of the wind function.

- A constant wind problem: $\mathbf{w} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
- A recirculating flow (vortex) problem: $\mathbf{w} = \begin{pmatrix} 4(2y-1)(1-x)x \\ -4(2x-1)(1-y)y \end{pmatrix}$

Uniform FEM discretizations: isoP2-P0 and isoP2-P1. These discretizations satisfy the inf-sup condition: no pressure stabilization is needed.
SUPG stabilization is used for the velocities.

The Krylov subspace method used is Bi-CGSTAB. This method requires two matrix-vector multiplies with $\mathcal{A}$ and two applications of the preconditioner at each iteration.

Results for Kay, Loghin and Wathen preconditioner

Test problems: steady Oseen, homogeneous Dirichlet BCs, two choices of the wind function.

- A constant wind problem: $\mathbf{w} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
- A recirculating flow (vortex) problem: $\mathbf{w} = \begin{pmatrix} 4(2y-1)(1-x)x \\ -4(2x-1)(1-y)y \end{pmatrix}$

Uniform FEM discretizations: isoP2-P0 and isoP2-P1. These discretizations satisfy the inf-sup condition: no pressure stabilization is needed.
SUPG stabilization is used for the velocities.

The Krylov subspace method used is Bi-CGSTAB. This method requires two matrix-vector multiplies with $\mathcal{A}$ and two applications of the preconditioner at each iteration.

A preconditioning step requires two convection-diffusion solves (three in 3D) and one Poisson solve at each iteration, plus some mat-vecs.

Results for Kay, Loghin, and Wathen preconditioner (cont.)

Results for KLW preconditioner.

mesh size h	viscosity ν				
	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}
constant wind					
1/16	6 / 12	8 / 16	12 / 24	30 / 34	100 / 80
1/32	6 / 10	10 / 16	14 / 24	24 / 28	86 / 92
1/64	6 / 10	8 / 14	16 / 24	22 / 32	64 / 66
1/128	6 / 10	8 / 12	16 / 26	24 / 36	64 / 58
rotating vortex					
1/16	6 / 8	10 / 12	30 / 40	$>$ 400 / 188	
1/32	6 / 8	10 / 12	30 / 40	$>$ 400 / 378	
1/64	4 / 6	8 / 12	26 / 40	$>$ 400 / $>$ 400	
1/128	4 / 6	8 / 10	22 / 44	228 / $>$ 400	

Number of Bi-CGSTAB iterations
(**Note**: exact solves used throughout. Stopping criterion: $\|\mathbf{b} - \mathcal{A}\mathbf{x}_k\|_2 < 10^{-6}\|\mathbf{b}\|_2$).

Results with Vanka-type MG preconditioner

Results for Vanka-MG-BiCGStab approach, isoP2-P0 FEM.

mesh size h	viscosity $= \nu$				
	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}
constant wind					
1/16	4	4	3	4	5
1/32	4	4	3	4	4
1/64	4	4	4	3	5
1/128	4	4	4	3	4
rotating vortex					
1/16	4	5	5	8	14
1/32	4	4	6	9	17
1/64	4	4	5	8	40
1/128	4	4	4	9	> 400

Better than KLW preconditioner, but still not robust for low viscosity. Can we do better than this?

Outline

Augmented Lagrangian formulation of saddle point problems

Consider the equivalent augmented Lagrangian formulation (Fortin, Glowinski, 1982) given by

$$\begin{pmatrix} A+\gamma B^T W^{-1} B & B^T \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f+\gamma B^T W^{-1} g \\ g \end{pmatrix}, \tag{1}$$

where $\gamma > 0$ and W is symmetric positive definite.

Augmented Lagrangian formulation of saddle point problems

Consider the equivalent augmented Lagrangian formulation (Fortin, Glowinski, 1982) given by

$$\begin{pmatrix} A+\gamma B^T W^{-1} B & B^T \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f+\gamma B^T W^{-1} g \\ g \end{pmatrix}, \tag{1}$$

where $\gamma > 0$ and W is symmetric positive definite.
Letting $A_\gamma := A + \gamma B^T W^{-1} B$ and $f_\gamma := f + \gamma B^T W^{-1} g$,

$$\begin{pmatrix} A_\gamma & B^T \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f_\gamma \\ g \end{pmatrix}, \quad \text{or} \quad \widehat{\mathcal{A}}\,\mathbf{x} = \hat{\mathbf{b}}. \tag{2}$$

Augmented Lagrangian formulation of saddle point problems

Consider the equivalent augmented Lagrangian formulation (Fortin, Glowinski, 1982) given by

$$\begin{pmatrix} A+\gamma B^T W^{-1} B & B^T \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f+\gamma B^T W^{-1} g \\ g \end{pmatrix}, \tag{1}$$

where $\gamma > 0$ and W is symmetric positive definite.
Letting $A_\gamma := A + \gamma B^T W^{-1} B$ and $f_\gamma := f + \gamma B^T W^{-1} g$,

$$\begin{pmatrix} A_\gamma & B^T \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f_\gamma \\ g \end{pmatrix}, \quad \text{or} \quad \widehat{\mathcal{A}}\,\mathbf{x} = \hat{\mathbf{b}}. \tag{2}$$

B. and Olshanskii (SISC 2006) introduced the following block preconditioner for (2)

$$\mathcal{P} = \begin{pmatrix} A_\gamma & B^T \\ O & \widehat{S} \end{pmatrix}, \quad \widehat{S}^{-1} = -\nu \widehat{M}_p^{-1} - \gamma W^{-1}. \tag{3}$$

Analysis for the Oseen problem

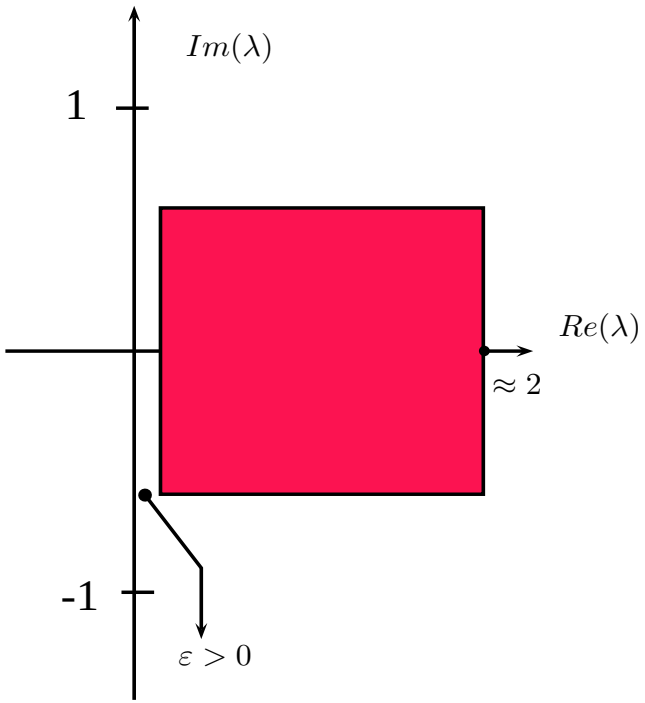
Theorem (B./Olshanskii, SISC 2006)

Setting $W = M_p$, the preconditioned matrix $\mathcal{P}^{-1}\widehat{\mathcal{A}}$ has the eigenvalue 1 of multiplicity n; the remaining m eigenvalues are contained in a rectangle in the right half plane with sides independent of the mesh size h, and bounded away from 0. Moreover, for $\gamma = O(\nu^{-1})$ the rectangle does not depend on ν. When $\gamma \to \infty$, all the eigenvalues tend to 1.

Analysis for the Oseen problem

Theorem (B./Olshanskii, SISC 2006)

Setting $W = M_p$, the preconditioned matrix $\mathcal{P}^{-1}\widehat{\mathcal{A}}$ has the eigenvalue 1 of multiplicity n; the remaining m eigenvalues are contained in a rectangle in the right half plane with sides independent of the mesh size h, and bounded away from 0. Moreover, for $\gamma = O(\nu^{-1})$ the rectangle does not depend on ν. When $\gamma \to \infty$, all the eigenvalues tend to 1.



Analysis for the Oseen problem (cont.)

Using field of values analysis, we can prove the following stronger parameter-independent convergence result for preconditioned GMRES:

Analysis for the Oseen problem (cont.)

Using field of values analysis, we can prove the following stronger parameter-independent convergence result for preconditioned GMRES:

Theorem (B./Olshanskii, SINUM 2011)

For $\nu < 1$, if $\gamma = \|(BA^{-1}B^T)^{-1}M_p\|_{M_p}$ the residual norms in GMRES with the original AL preconditioner satisfy

$$\|\hat{\mathbf{b}} - \widehat{\mathcal{A}}\mathbf{x}_k\| \leq q^k \|\hat{\mathbf{b}} - \widehat{\mathcal{A}}\mathbf{x}_0\|,$$

where $q < 1$ is independent of problem parameters h, ν and α.

Analysis for the Oseen problem (cont.)

Using field of values analysis, we can prove the following stronger parameter-independent convergence result for preconditioned GMRES:

Theorem (B./Olshanskii, SINUM 2011)

For $\nu < 1$, if $\gamma = \|(BA^{-1}B^T)^{-1}M_p\|_{M_p}$ the residual norms in GMRES with the original AL preconditioner satisfy

$$\|\hat{\mathbf{b}} - \widehat{\mathcal{A}}\mathbf{x}_k\| \leq q^k \|\hat{\mathbf{b}} - \widehat{\mathcal{A}}\mathbf{x}_0\|,$$

where $q < 1$ is independent of problem parameters h, ν and α.

Recall that the field of values of an $n \times n$ matrix $\mathcal{B}$ is the subset of $\mathbf{C}$ defined by

$$\mathcal{F}(\mathcal{B}) := \{\mathbf{x}^* \mathcal{B}\mathbf{x} \mid x \in \mathbf{C}^n,\ \mathbf{x}^*\mathbf{x} = 1\}.$$

We proved that $\mathcal{F}(\mathcal{P}^{-1}\mathcal{A})$ is bounded and bounded away from 0 for all h, ν and α. This implies the above convergence result for GMRES.

Practical considerations

Applying $\mathcal{P}^{-1}$ to a vector requires one solve with A_γ and one with $\widehat{S}$.

Practical considerations

Applying $\mathcal{P}^{-1}$ to a vector requires one solve with A_γ and one with $\widehat{S}$.

- In practice we use $W = \widehat{M}_p = \text{diag}(M_p)$ in A_γ

Practical considerations

Applying $\mathcal{P}^{-1}$ to a vector requires one solve with A_γ and one with $\widehat{S}$.

- In practice we use $W = \widehat{M}_p = \text{diag}(M_p)$ in A_γ
- The solve with A_γ can be approximated by a suitable geometric multigrid method for elliptic systems (similar to the one by Schöberl, NM 1999)

Practical considerations

Applying $\mathcal{P}^{-1}$ to a vector requires one solve with A_γ and one with $\widehat{S}$.

- In practice we use $W = \widehat{M}_p = \text{diag}(M_p)$ in A_γ
- The solve with A_γ can be approximated by a suitable geometric multigrid method for elliptic systems (similar to the one by Schöberl, NM 1999)
- For $\widehat{S}^{-1}$, a few Richardson iterations preconditioned with the diagonal of M_p can be used to solve the linear system with M_p.

Practical considerations

Applying $\mathcal{P}^{-1}$ to a vector requires one solve with A_γ and one with $\widehat{S}$.

- In practice we use $W = \widehat{M}_p = \text{diag}(M_p)$ in A_γ
- The solve with A_γ can be approximated by a suitable geometric multigrid method for elliptic systems (similar to the one by Schöberl, NM 1999)
- For $\widehat{S}^{-1}$, a few Richardson iterations preconditioned with the diagonal of M_p can be used to solve the linear system with M_p.

Though the previous theorems suggest that $\gamma = O(\nu^{-1})$, in practice, $\gamma = O(1)$ is sufficient for (near) parameter-independent convergence.

The multigrid solver for A_γ

The main multigrid components are the prolongation and restriction operators and the smoother.

The multigrid solver for A_γ

The main multigrid components are the prolongation and restriction operators and the smoother.

Let $\mathbf{V}_h$, $\mathbf{Q}_h$ denote the finite element velocity and pressure fine spaces. Also, let $P_h : L_0^2(\Omega) \longrightarrow \mathbf{Q}_h$ be the orthogonal projection. Consider the bilinear form

$$a_h(\mathbf{u}, \mathbf{v}) = \nu \langle \nabla \mathbf{u}, \nabla \mathbf{v} \rangle + \langle P_h \mathrm{div}\, \mathbf{u}, \mathrm{div}\, \mathbf{v} \rangle .$$

The multigrid solver for A_γ

The main multigrid components are the prolongation and restriction operators and the smoother.

Let $\mathbf{V}_h$, $\mathbf{Q}_h$ denote the finite element velocity and pressure fine spaces. Also, let $P_h : L_0^2(\Omega) \longrightarrow \mathbf{Q}_h$ be the orthogonal projection. Consider the bilinear form

$$a_h(\mathbf{u}, \mathbf{v}) = \nu \langle \nabla \mathbf{u}, \nabla \mathbf{v} \rangle + \langle P_h \mathrm{div}\, \mathbf{u}, \mathrm{div}\, \mathbf{v} \rangle .$$

Let $\tilde{\mathbf{V}}_h \subset \mathbf{V}_h$ and $\tilde{\mathbf{Q}}_h \subset \mathbf{Q}_h$ denote subspaces such that there is a constant c independent of h with

$$\inf_{\tilde{q}_h \in \tilde{\mathbf{Q}}_h} \sup_{\tilde{\mathbf{v}}_h \in \tilde{\mathbf{V}}_h} \frac{\langle \tilde{q}_h, \mathrm{div}\, \tilde{\mathbf{v}}_h \rangle}{\|\nabla \tilde{\mathbf{v}}_h\| \|\tilde{q}_h\|} \geq c > 0$$

and $\tilde{\mathbf{V}}_h \subset \ker (P_H \mathrm{div})$.

The multigrid solver for A_γ (cont.)

For given $\tilde{\mathbf{u}}_H \in \tilde{\mathbf{V}}_H$ (coarse space), let $\tilde{\mathbf{u}}_h$ be the solution of the problem

$$a_h(\tilde{\mathbf{u}}_h, \tilde{\mathbf{v}}_h) = a_h(\mathbf{u}_H, \tilde{\mathbf{v}}_h) \quad \forall\, \tilde{\mathbf{v}}_h \in \tilde{\mathbf{V}}_h\,. \tag{4}$$

The multigrid solver for A_γ (cont.)

For given $\tilde{\mathbf{u}}_H \in \tilde{\mathbf{V}}_H$ (coarse space), let $\tilde{\mathbf{u}}_h$ be the solution of the problem

$$a_h(\tilde{\mathbf{u}}_h, \tilde{\mathbf{v}}_h) = a_h(\mathbf{u}_H, \tilde{\mathbf{v}}_h) \quad \forall\, \tilde{\mathbf{v}}_h \in \tilde{\mathbf{V}}_h\,. \tag{4}$$

The prolongation $p : \mathbf{V}_H \longrightarrow \mathbf{V}_h$ is defined by

$$p\,\mathbf{u}_H = \mathbf{u}_H - \tilde{\mathbf{u}}_h\,.$$

The multigrid solver for A_γ (cont.)

For given $\tilde{\mathbf{u}}_H \in \tilde{\mathbf{V}}_H$ (coarse space), let $\tilde{\mathbf{u}}_h$ be the solution of the problem

$$a_h(\tilde{\mathbf{u}}_h, \tilde{\mathbf{v}}_h) = a_h(\mathbf{u}_H, \tilde{\mathbf{v}}_h) \quad \forall\, \tilde{\mathbf{v}}_h \in \tilde{\mathbf{V}}_h\,. \tag{4}$$

The prolongation $p : \mathbf{V}_H \longrightarrow \mathbf{V}_h$ is defined by

$$p\,\mathbf{u}_H = \mathbf{u}_H - \tilde{\mathbf{u}}_h\,.$$

For an appropriate choice of $\tilde{\mathbf{V}}_h, \tilde{\mathbf{Q}}_h$, solving (4) requires solving N_p independent linear systems of dimension 6 (in 2D) or 9 (in 3D) where N_p is the number of elements in the coarse grid triangulation T_H.

The multigrid solver for A_γ (cont.)

For example, for isoP2-P0 or P2-P0 finite elements we can take

$$\begin{aligned} \tilde{\mathbf{Q}}_h &= \left\{ \tilde{q}_h \in \mathbf{Q}_h : \int_\tau \tilde{q}_h \, dx = 0 \quad \forall\, \tau \in T_H \right\}, \\ \tilde{\mathbf{V}}_h &= \left\{ \tilde{\mathbf{v}}_h \in \mathbf{V}_h : \tilde{\mathbf{v}}_h|_{\partial\tau} = 0 \quad \forall\, \tau \in T_H \right\}. \end{aligned}$$

The multigrid solver for A_γ (cont.)

For example, for isoP2-P0 or P2-P0 finite elements we can take

$$\begin{aligned} \tilde{\mathbf{Q}}_h &= \left\{ \tilde{q}_h \in \mathbf{Q}_h : \int_\tau \tilde{q}_h \, dx = 0 \quad \forall\, \tau \in T_H \right\}, \\ \tilde{\mathbf{V}}_h &= \left\{ \tilde{\mathbf{v}}_h \in \mathbf{V}_h : \tilde{\mathbf{v}}_h|_{\partial\tau} = 0 \quad \forall\, \tau \in T_H \right\}. \end{aligned}$$

For the restriction, we simply take the L^2-orthogonal projection from $\mathbf{V}_h$ to $\mathbf{V}_H$.

The multigrid solver for A_γ (cont.)

For example, for isoP2-P0 or P2-P0 finite elements we can take

$$\begin{aligned} \tilde{\mathbf{Q}}_h &= \left\{ \tilde{q}_h \in \mathbf{Q}_h : \int_\tau \tilde{q}_h \, dx = 0 \quad \forall\, \tau \in T_H \right\}, \\ \tilde{\mathbf{V}}_h &= \left\{ \tilde{\mathbf{v}}_h \in \mathbf{V}_h : \tilde{\mathbf{v}}_h|_{\partial\tau} = 0 \quad \forall\, \tau \in T_H \right\}. \end{aligned}$$

For the restriction, we simply take the L^2-orthogonal projection from $\mathbf{V}_h$ to $\mathbf{V}_H$.

Finally, the smoother is chosen so as to effectively eliminate the oscillatory components in $\ker(P_h \mathrm{div})$. We use a multiplicative Schwarz-type iteration with overlapping blocks, where each block contains degrees of freedom supporting a basis function from $\ker(P_h \mathrm{div})$.

The multigrid solver for A_γ (cont.)

We found that this smoother is fairly insensitive to the block ordering.

The multigrid solver for A_γ (cont.)

We found that this smoother is fairly insensitive to the block ordering.

In our implementation, blocks are ordered lexicographically.

The multigrid solver for A_γ (cont.)

We found that this smoother is fairly insensitive to the block ordering.

In our implementation, blocks are ordered lexicographically.

One smoothing iteration consists of two relaxation steps: one with left-right/top-bottom ordering, the other one in the reverse order.

The multigrid solver for A_γ (cont.)

We found that this smoother is fairly insensitive to the block ordering.

In our implementation, blocks are ordered lexicographically.

One smoothing iteration consists of two relaxation steps: one with left-right/top-bottom ordering, the other one in the reverse order.

The block size is 14 in 2D and 39 in 3D.

The multigrid solver for A_γ (cont.)

We found that this smoother is fairly insensitive to the block ordering.

In our implementation, blocks are ordered lexicographically.

One smoothing iteration consists of two relaxation steps: one with left-right/top-bottom ordering, the other one in the reverse order.

The block size is 14 in 2D and 39 in 3D.

See B. and Olshanskii (SISC, 2006) for additional details.

Numerical results

Table: Bi-CGSTAB iterations (isoP2-P0 FEM, SUPG, $\gamma = 1$)

Viscosity ν	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}
Mesh size	Constant wind				
1/16	7	5	5	6	6
1/32	7	5	6	7	8
1/64	5	5	6	5	7
1/128	5	5	5	5	6
Mesh size	Rotating vortex				
1/16	5	5	6	10	15
1/32	4	4	5	10	21
1/64	4	4	5	9	18
1/128	4	5	5	7	14

The rate of convergence of Krylov subspace method with this preconditioner is nearly optimal:

- Independent of the grid; almost independent of viscosity
- Cost is $O(n + m)$ per iteration
- Similar results with isoP2-P1 FEM

Numerical results (cont.)

Results for AL approach, isoP2-P1 FEM.

mesh size h	viscosity ν				
	1.	10^{-1}	10^{-2}	10^{-3}	10^{-4}
	parameter γ				
	1.	1.	1.	0.1	0.02
constant wind					
1/16	6	6	7	8	24
1/32	7	6	10	8	22
1/64	7	6	8	7	19
1/128	7	6	9	9	18
rotating vortex					
1/16	6	6	7	13	25
1/32	5	6	9	11	32
1/64	4	5	10	11	37
1/128	4	4	10	12	34

Number of preconditioned BiCGStab iterations
($\hat{A}_\gamma^{-1}$ is one W(1,1)-cycle).

Outline

The modified augmented Lagrangian-based preconditioner

Motivation: Circumvent sophisticated geometric multigrid techniques in A_γ^{-1}, so as to be able to handle unstructured grids and more complex geometries.

The modified augmented Lagrangian-based preconditioner

Motivation: Circumvent sophisticated geometric multigrid techniques in A_γ^{-1}, so as to be able to handle unstructured grids and more complex geometries. From $A = \text{diag}(A_1, A_2)$ and $B = (B_1, B_2)$, we find

$$A_\gamma = A + \gamma B^T W^{-1} B = \begin{pmatrix} A_1 + \gamma B_1^T W^{-1} B_1 & \gamma B_1^T W^{-1} B_2 \\ \gamma B_2^T W^{-1} B_1 & A_2 + \gamma B_2^T W^{-1} B_2 \end{pmatrix} := \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}.$$

The modified augmented Lagrangian-based preconditioner

Motivation: Circumvent sophisticated geometric multigrid techniques in A_γ^{-1}, so as to be able to handle unstructured grids and more complex geometries. From $A = \mathrm{diag}(A_1, A_2)$ and $B = (B_1, B_2)$, we find

$$A_\gamma = A + \gamma B^T W^{-1} B = \begin{pmatrix} A_1 + \gamma B_1^T W^{-1} B_1 & \gamma B_1^T W^{-1} B_2 \\ \gamma B_2^T W^{-1} B_1 & A_2 + \gamma B_2^T W^{-1} B_2 \end{pmatrix} := \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}.$$

The modified AL preconditioner (B., Olshanskii and Wang, IJNMF 2011) is defined as

$$\widetilde{\mathcal{P}} = \begin{pmatrix} \widetilde{A}_\gamma & B^T \\ O & \widehat{S} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & B_1^T \\ O & A_{22} & B_2^T \\ O & O & \widehat{S} \end{pmatrix} \tag{5}$$

The modified augmented Lagrangian-based preconditioner

Motivation: Circumvent sophisticated geometric multigrid techniques in A_γ^{-1}, so as to be able to handle unstructured grids and more complex geometries. From $A = \mathrm{diag}(A_1, A_2)$ and $B = (B_1, B_2)$, we find

$$A_\gamma = A + \gamma B^T W^{-1} B = \begin{pmatrix} A_1 + \gamma B_1^T W^{-1} B_1 & \gamma B_1^T W^{-1} B_2 \\ \gamma B_2^T W^{-1} B_1 & A_2 + \gamma B_2^T W^{-1} B_2 \end{pmatrix} := \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}.$$

The modified AL preconditioner (B., Olshanskii and Wang, IJNMF 2011) is defined as

$$\widetilde{\mathcal{P}} = \begin{pmatrix} \widetilde{A}_\gamma & B^T \\ O & \widehat{S} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & B_1^T \\ O & A_{22} & B_2^T \\ O & O & \widehat{S} \end{pmatrix} \tag{5}$$

- $A_{ii} = A_i + \gamma B_i^T W^{-1} B_i$ $(i = 1, 2)$ can be interpreted as discrete scalar anisotropic convection-diffusion operators with anisotropy ratio $\approx 1 + \frac{\gamma}{\nu}$.

The modified augmented Lagrangian-based preconditioner

Motivation: Circumvent sophisticated geometric multigrid techniques in A_γ^{-1}, so as to be able to handle unstructured grids and more complex geometries. From $A = \mathrm{diag}(A_1, A_2)$ and $B = (B_1, B_2)$, we find

$$A_\gamma = A + \gamma B^T W^{-1} B = \begin{pmatrix} A_1 + \gamma B_1^T W^{-1} B_1 & \gamma B_1^T W^{-1} B_2 \\ \gamma B_2^T W^{-1} B_1 & A_2 + \gamma B_2^T W^{-1} B_2 \end{pmatrix} := \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}.$$

The modified AL preconditioner (B., Olshanskii and Wang, IJNMF 2011) is defined as

$$\widetilde{\mathcal{P}} = \begin{pmatrix} \widetilde{A}_\gamma & B^T \\ O & \widehat{S} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & B_1^T \\ O & A_{22} & B_2^T \\ O & O & \widehat{S} \end{pmatrix} \tag{5}$$

- $A_{ii} = A_i + \gamma B_i^T W^{-1} B_i$ $(i = 1, 2)$ can be interpreted as discrete scalar anisotropic convection-diffusion operators with anisotropy ratio $\approx 1 + \frac{\gamma}{\nu}$.
- They can be solved with standard algebraic multigrid (AMG) methods, in particular, parallel AMG solvers.

Analysis

Note that

$$\widehat{\mathcal{A}}\widetilde{\mathcal{P}}^{-1} = \left(\begin{array}{c|cc} I_{n/2} & O & O \\ \hline * & I_{n/2} - D & E \\ * & F & I_m - G \end{array}\right).$$

The eigenvalues of $\widehat{\mathcal{A}}\widetilde{\mathcal{P}}^{-1}$ are $\lambda = 1$ of multiplicity $n/2$, plus the eigenvalues of

$$\begin{pmatrix} I_{n/2} - D & E \\ F & I_m - G \end{pmatrix} = I_{n/2+m} - \begin{pmatrix} D & -E \\ -F & G \end{pmatrix}.$$

In general, the multiplicity of $\lambda = 1$ is only $n/2$.

Analysis

Note that

$$\widehat{\mathcal{A}}\widetilde{\mathcal{P}}^{-1} = \left(\begin{array}{c|cc} I_{n/2} & O & O \\ \hline * & I_{n/2} - D & E \\ * & F & I_m - G \end{array}\right).$$

The eigenvalues of $\widehat{\mathcal{A}}\widetilde{\mathcal{P}}^{-1}$ are $\lambda = 1$ of multiplicity $n/2$, plus the eigenvalues of

$$\begin{pmatrix} I_{n/2} - D & E \\ F & I_m - G \end{pmatrix} = I_{n/2+m} - \begin{pmatrix} D & -E \\ -F & G \end{pmatrix}.$$

In general, the multiplicity of $\lambda = 1$ is only $n/2$.
However, letting $\widehat{S}^{-1} = -\gamma W^{-1}$, the matrix on the right-hand side is rank deficient by $n/2$, so $\widehat{\mathcal{A}}\widetilde{\mathcal{P}}^{-1}$ has the eigenvalue $\lambda = 1$ of multiplicity at least n.

Using field of values analysis, we can prove that the convergence rate of GMRES with the modified AL preconditioner is h-independent, with a moderate dependence on ν.

The choice of γ

The value of the augmentation parameter γ is determined by local Fourier analysis (LFA):

The choice of γ

The value of the augmentation parameter γ is determined by local Fourier analysis (LFA):

1. 'discretize' the diffusion and (frozen) convection terms in A by centered differences, assuming periodic BCs;

The choice of γ

The value of the augmentation parameter γ is determined by local Fourier analysis (LFA):

1. 'discretize' the diffusion and (frozen) convection terms in A by centered differences, assuming periodic BCs;
2. 'discretize' the gradient and divergence by one-sided differences;

The choice of γ

The value of the augmentation parameter γ is determined by local Fourier analysis (LFA):

1. 'discretize' the diffusion and (frozen) convection terms in A by centered differences, assuming periodic BCs;
2. 'discretize' the gradient and divergence by one-sided differences;
3. note that $W = \widehat{M}_p$ scales as h^2;

The choice of γ

The value of the augmentation parameter γ is determined by local Fourier analysis (LFA):

1. 'discretize' the diffusion and (frozen) convection terms in A by centered differences, assuming periodic BCs;
2. 'discretize' the gradient and divergence by one-sided differences;
3. note that $W = \widehat{M}_p$ scales as h^2;
4. express $\widetilde{\mathcal{P}}$ and $\widehat{\mathcal{A}}$ in terms of "Fourier eigenvalues", and find the γ that minimizes the average distance of the non-unit eigenvalues $\lambda(\gamma)$ of the preconditioned matrix $\widehat{\mathcal{A}}\widetilde{\mathcal{P}}^{-1}$ from 1.

Note that γ only depends on h and ν. Hence it can be pre-computed, so no overhead is imposed. The discretizations are symbolic.

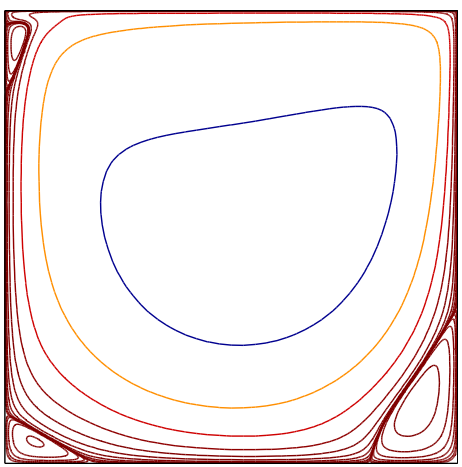
Details in B. and Wang (SISC, 2011).

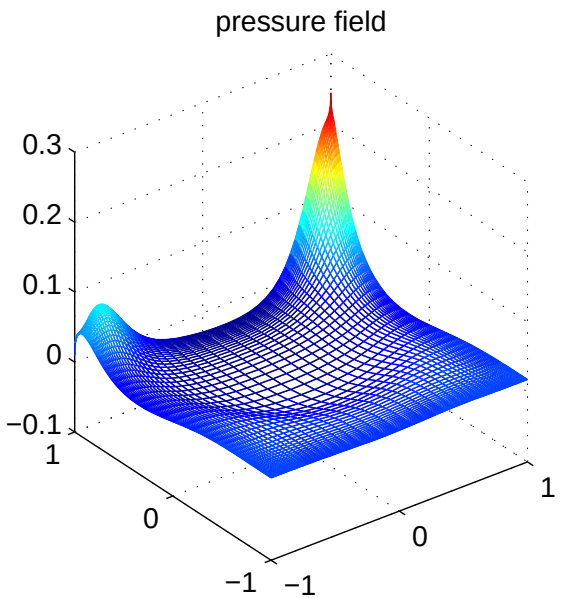
Example: a regularized lid driven cavity problem

All 2D experiments are done using IFISS package (Elman, Silvester & Ramage).

In the lid driven cavity problem, the flow is enclosed in a square with $u_1 = 1 - x^4, u_2 = 0$ on the top to represent the moving lid.

Figure: Regularized lid driven cavity (Q2-Q1, $\nu = 0.001$, stretched 128×128 grid)





Iteration counts for the lid driven cavity problem

Table: GMRES iterations with modified AL preconditioner (cavity, Q2-Q1, uniform grids)

Viscosity	0.1		0.01		0.005		0.001	
Grid	LFA	Opt	LFA	Opt	LFA	Opt	LFA	Opt
16×16	9	9	12	12	26	15	42	23
32×32	10	9	11	11	20	14	37	29
64×64	9	9	11	10	13	13	33	27
128×128	9	9	10	10	13	12	25	24

Observations:

- The number of GMRES iterations with γ chosen by LFA is almost the same as for the optimal γ, especially on the finest grid
- The iteration counts with both sets of γ are independent of grid size and only mildly dependent on viscosity

Values of γ

Table: The values of γ chosen by LFA and optimal values (cavity, Q2-Q1, uniform grids)

Viscosity	0.1		0.01		0.005		0.001	
Grid	LFA	Opt	LFA	Opt	LFA	Opt	LFA	Opt
16×16	0.42	0.45	0.075	0.085	0.270	0.068	0.220	0.063
32×32	0.29	0.38	0.056	0.050	0.098	0.043	0.067	0.035
64×64	0.32	0.32	0.055	0.045	0.032	0.032	0.037	0.022
128×128	0.28	0.28	0.036	0.046	0.022	0.032	0.020	0.017

Eigenvalues of preconditioned matrices

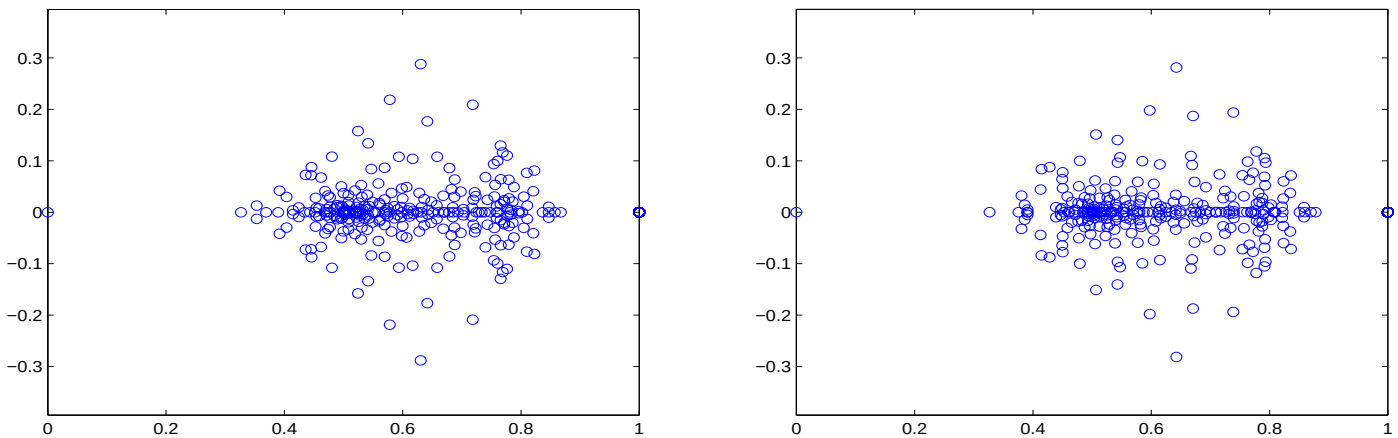


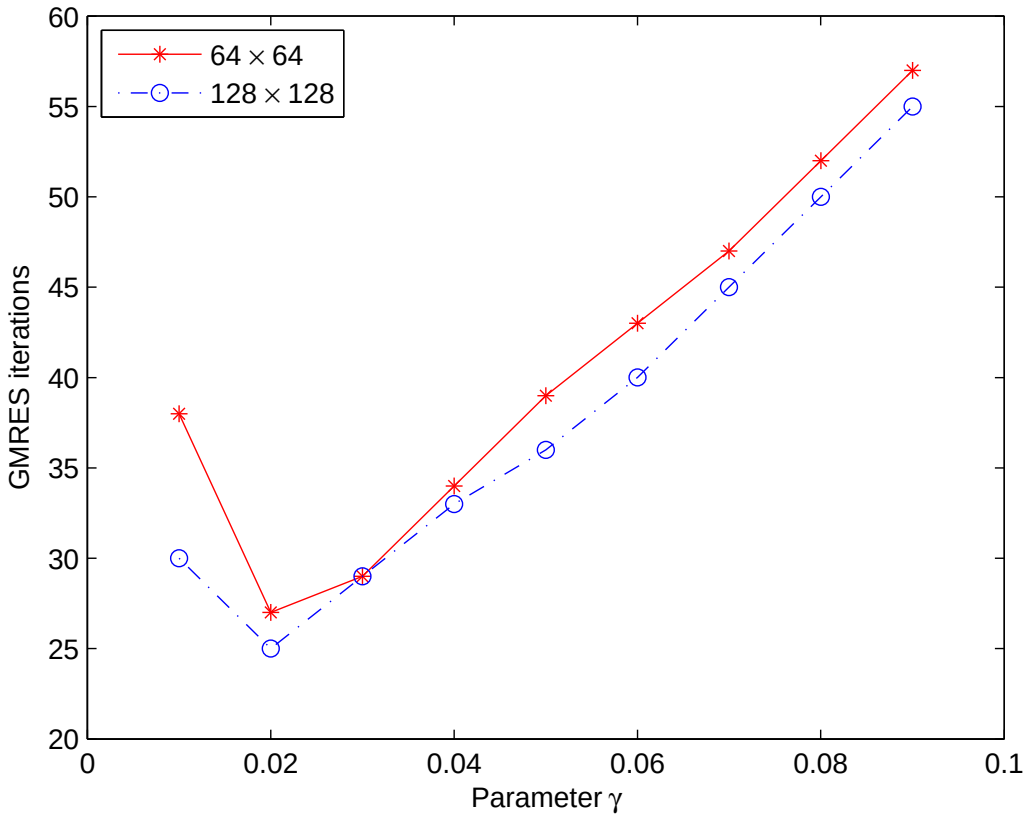
Figure: Plots of the eigenvalues of the preconditioned Oseen matrix (lid driven cavity, Q2-Q1, 32×32 uniform grid, $\nu = 0.01$). Left: with optimal γ. Right: with γ chosen by Fourier analysis.

The two values of γ are very close: 0.050 vs. 0.056.

The eigenvalue $\lambda = 1$ has multiplicity n (for all γ).

Iteration counts with various values of γ

Figure: GMRES iterations with modified AL preconditioner (cavity, Q2-Q1, uniform grids, $\nu = 0.001$)



Iteration counts with Newton linearization

Use the same values of γ as for Picard linearization.

Table: GMRES iterations with modified AL preconditioner (cavity, Q2-Q1, stretched grids, Newton)

Viscosity	0.1		0.01		0.005		0.001	
Grid	LFA	Opt	LFA	Opt	LFA	Opt	LFA	Opt
16×16	13	13	21	21	20	25	99	62
32×32	14	14	23	23	31	30	71	60
64×64	14	14	24	23	35	33	84	72
128×128	15	14	26	23	40	34	95	82

Not quite h-independent for small ν.

Results for stretched grids and comparison with PCD/LSC/mPCD

Table: GMRES with modified AL preconditioner (cavity, Q2-Q1, stretched grids)

Viscosity	0.1		0.01		0.005		0.001	
Grid	FA	Opt	FA	Opt	FA	Opt	FA	Opt
16×16	9	9	11	11	21	13	35	20
32×32	9	9	11	11	17	14	31	23
64×64	8	8	11	11	14	14	29	25
128×128	8	7	11	11	14	13	26	26

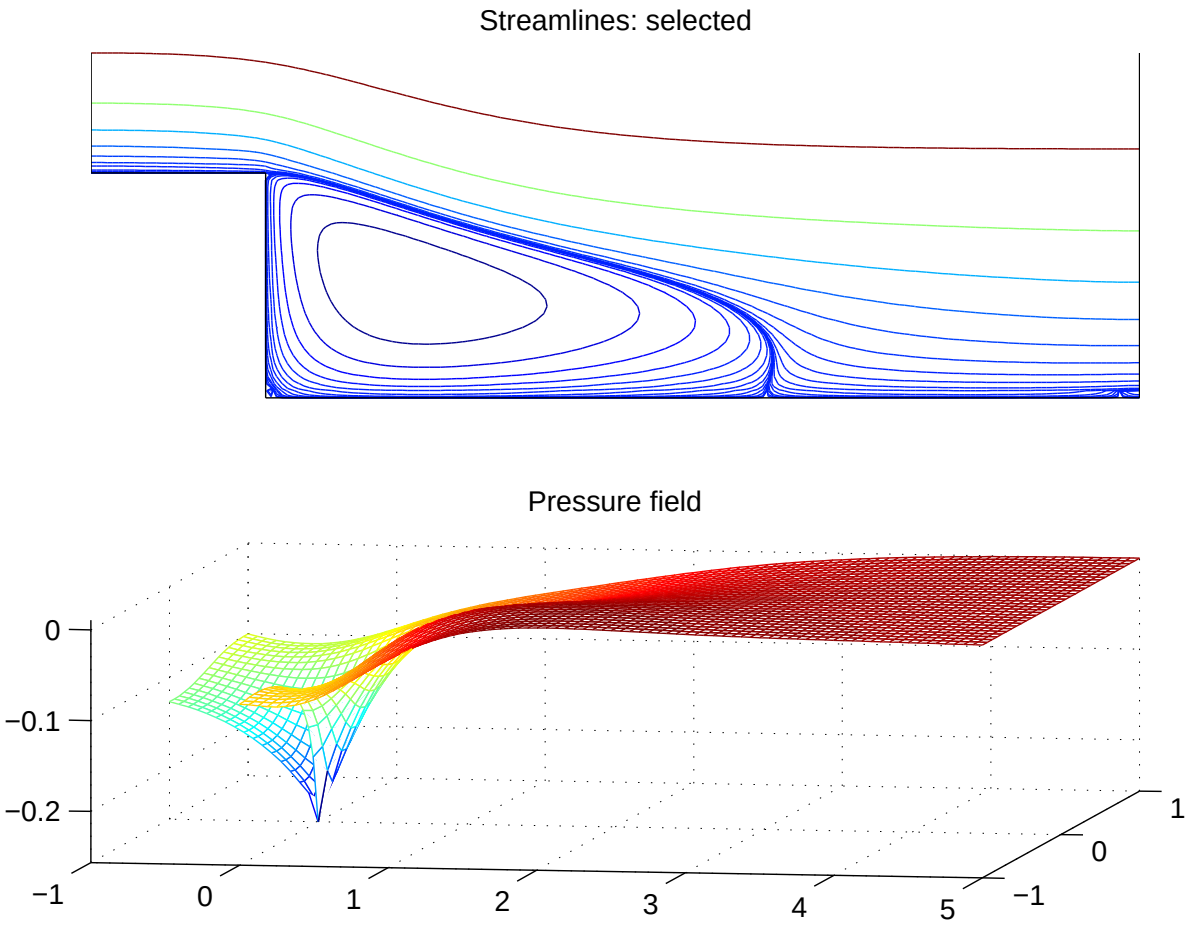
Table: GMRES iterations with PCD, LSC and mPCD preconditioners of Elman, Silvester, Wathen (cavity, Q2-Q1, stretched grids, $\nu = 0.001$)

Grid	PCD	LSC	mPCD
16×16	79	50	81
32×32	105	78	201
64×64	117	117	135
128×128	117	174	144

Note: All methods have similar costs per iteration.

A backward facing step test problem

Figure: Backward facing step problem (Q2-Q1, $\nu = 0.005$, uniform 64×192 grid)



Iteration counts for the backward facing step problem

Table: GMRES iterations with modified AL preconditioner (step, Q2-Q1, uniform grids)

Viscosity	0.1		0.01		0.005	
Grid	LFA	Opt	LFA	Opt	LFA	Opt
16×48	15	12	46	16	59	19
32×96	12	12	24	17	38	20
64×192	12	11	17	16	26	19
128×384	11	11	15	15	19	19

Comparison of exact and inexact solves

In the following Table we present a comparison of modified AL preconditioning with exact and inexact inversion of diagonal blocks A_{ii}.

For the 'exact' solves we use the sparse LU factorization with column AMD reordering available in Matlab.

For the inexact solves we use one iteration (V-cycle) of AMG using the HSL_MI20 code developed by Boyle, Mihajlovic and Scott (IJNME 2010).

We perform tests for both Picard and Newton linearizations of the lid driven cavity problem discretized with Q2-Q1 elements (Newton is harder), using the same value of γ from Fourier analysis in both cases. The viscosity is $\nu = 0.005$.

The experiments are performed in Matlab 7.9.0 on a Sun Microsystems SunFire.

The upshot:

- Using inexact solves does not affect the convergence rates
- Inexact solves result in much faster solution times

Iteration counts and timings of exact solve and AMG (MI20)

Table: Comparison of exact and inexact inner solvers. GMRES iterations and timings with modified AL preconditioner (cavity, Q2-Q1, uniform grids, $\nu = 0.005$)

Grid	Picard		Newton	
Timings	Exact	MI20	Exact	MI20
64×64	13	13	35	36
Set-up time	1.93	0.31	1.95	0.26
Iter time	0.62	2.76	1.75	5.76
Total time	2.55	3.07	3.70	6.02
128×128	13	13	39	39
Setup time	34.90	1.29	34.34	1.25
Iter time	4.44	12.00	10.94	29.38
Total time	39.34	13.29	45.28	30.63
256×256	13	13	43	43
Setup time	856.74	5.86	673.29	6.07
Iter time	40.22	58.84	85.84	152.05
Total time	896.96	64.70	759.12	158.12

The modified AL preconditioner for 3D problems

For 3D Oseen problems $A = \mathrm{diag}(A_1, A_2, A_3)$ and $B = (B_1, B_2, B_3)$.

$$
\begin{aligned}
A_\gamma &= A + \gamma B^T W^{-1} B \\
&= \begin{pmatrix} A_1 & O & O \\ O & A_2 & O \\ O & O & A_3 \end{pmatrix} + \gamma \begin{pmatrix} B_1^T \\ B_2^T \\ B_3^T \end{pmatrix} W^{-1} \begin{pmatrix} B_1 & B_2 & B_3 \end{pmatrix} \\
&= \begin{pmatrix} A_1 + \gamma B_1^T W^{-1} B_1 & \gamma B_1^T W^{-1} B_2 & \gamma B_1^T W^{-1} B_3 \\ \gamma B_2^T W^{-1} B_1 & A_2 + \gamma B_2^T W^{-1} B_2 & \gamma B_2^T W^{-1} B_3 \\ \gamma B_3^T W^{-1} B_1 & \gamma B_3^T W^{-1} B_2 & A_3 + \gamma B_3^T W^{-1} B_3 \end{pmatrix} \\
&=: \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix},
\end{aligned}
$$

so the modified AL preconditioner is

$$
\widetilde{P} = \begin{pmatrix} \widetilde{A}_\gamma & B^T \\ O & \widehat{S} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & A_{13} & B_1^T \\ O & A_{22} & A_{23} & B_2^T \\ O & O & A_{33} & B_3^T \\ O & O & O & \widehat{S} \end{pmatrix}.
$$

Parallel results

These preliminary runs are done on a cluster at ORNL. The cluster consists of eighty Cray XK6 nodes, and each node has a 16-core AMD 2.2 GHz 6274 Opteron processor and 32 GB of GDDR3 RAM.

The subproblems in the modified AL preconditioner are solved inexactly by one AMG iteration ('ML' solver in Trilinos).

Parallel results

These preliminary runs are done on a cluster at ORNL. The cluster consists of eighty Cray XK6 nodes, and each node has a 16-core AMD 2.2 GHz 6274 Opteron processor and 32 GB of GDDR3 RAM.

The subproblems in the modified AL preconditioner are solved inexactly by one AMG iteration ('ML' solver in Trilinos).

Steady 2D Oseen problem for the lid driven cavity ($\nu = 0.01$), with γ chosen by Fourier analysis. Q2-Q1 elements, 256×256 grid, $148,739$ unknowns.

The table reports GMRES iterations and timings.

Parallel results

These preliminary runs are done on a cluster at ORNL. The cluster consists of eighty Cray XK6 nodes, and each node has a 16-core AMD 2.2 GHz 6274 Opteron processor and 32 GB of GDDR3 RAM.

The subproblems in the modified AL preconditioner are solved inexactly by one AMG iteration ('ML' solver in Trilinos).

Steady 2D Oseen problem for the lid driven cavity ($\nu = 0.01$), with γ chosen by Fourier analysis. Q2-Q1 elements, 256×256 grid, $148,739$ unknowns.

The table reports GMRES iterations and timings.

No. of cores	2	4	8	16	32	64
Iterations	29	28	28	25	26	30
Set-up time	2.04	1.18	0.67	0.43	0.31	0.25
Iter time	7.11	4.00	1.87	0.78	0.43	0.34
Total time	10.15	5.18	2.54	1.21	0.74	0.59

Parallel results (cont.)

3D enclosed flow problem, $\nu = 0.01$, Marker-and-Cell discretization.

- $1,036,288$ DOFs on $64 \times 64 \times 64$ grid
- $8,442,624$ DOFs on $128 \times 128 \times 128$ grid.

No. of cores	2	4	8	16	32	64
$64 \times 64 \times 64$	23	22	22	23	24	27
Set-up time	4.12	2.53	1.42	0.91	0.79	0.96
Iter time	11.20	6.51	3.06	1.56	0.98	0.89
Total time	15.32	9.04	4.48	2.47	1.77	1.85
$128 \times 128 \times 128$	26	26	26	26	28	32
Set-up time	36.78	20.89	11.36	6.43	3.86	3.04
Iter time	115.12	78.00	39.42	19.35	10.05	6.09
Total time	151.90	101.37	54.44	25.78	13.91	9.13

Parallel results (cont.)

3D driven cavity, $\nu = 0.05$, P2-P1 discretization. Grad-div stabilization already present in the formulation of the NSE.

No. of cores	2	4	8	16	32	64
$16 \times 16 \times 16$	28	26	25	24	23	23
Set-up time	1.31	1.09	1.06	0.75	0.55	0.42
Iter time	3.46	2.18	1.44	0.82	0.54	0.41
Total time	4.77	3.27	2.50	1.57	1.09	0.83
$24 \times 24 \times 24$	26	24	24	23	24	22
Set-up time	4.63	3.92	3.81	2.37	1.64	1.10
Iter time	11.82	7.65	5.30	2.81	1.79	1.06
Total time	16.45	11.57	9.11	5.18	3.43	2.16
$32 \times 32 \times 32$	24	22	21	22	21	20
Set-up time	11.31	9.75	9.46	5.70	3.55	2.44
Iter time	26.48	17.38	11.44	6.64	3.79	2.11
Total time	37.78	27.13	20.90	12.34	7.34	4.55

Note: The Trilinos set-up does not scale for a small number of cores.

Outline

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy
- Suitably modified and combined with AMG-type inner solvers, the AL approach results in fairly robust preconditioners

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy
- Suitably modified and combined with AMG-type inner solvers, the AL approach results in fairly robust preconditioners
- Both stable and stabilized discretizations can be accommodated

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy
- Suitably modified and combined with AMG-type inner solvers, the AL approach results in fairly robust preconditioners
- Both stable and stabilized discretizations can be accommodated
- Stretched grids do not pose any difficulties to the AL approach

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy
- Suitably modified and combined with AMG-type inner solvers, the AL approach results in fairly robust preconditioners
- Both stable and stabilized discretizations can be accommodated
- Stretched grids do not pose any difficulties to the AL approach
- The Fourier-based approach gives very good estimates of the optimal parameter γ

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy
- Suitably modified and combined with AMG-type inner solvers, the AL approach results in fairly robust preconditioners
- Both stable and stabilized discretizations can be accommodated
- Stretched grids do not pose any difficulties to the AL approach
- The Fourier-based approach gives very good estimates of the optimal parameter γ
- Clearly better than competing methods on difficult problems

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy
- Suitably modified and combined with AMG-type inner solvers, the AL approach results in fairly robust preconditioners
- Both stable and stabilized discretizations can be accommodated
- Stretched grids do not pose any difficulties to the AL approach
- The Fourier-based approach gives very good estimates of the optimal parameter γ
- Clearly better than competing methods on difficult problems
- AL approach works well also as a smoother for coupled multigrid

Conclusions and future work

- Large linear systems of saddle point type still pose a significant challenge for modern preconditioned iterative solvers
- Computing steady solutions to incompressible flow problems for small values of the viscosity and on stretched grids is not easy
- Suitably modified and combined with AMG-type inner solvers, the AL approach results in fairly robust preconditioners
- Both stable and stabilized discretizations can be accommodated
- Stretched grids do not pose any difficulties to the AL approach
- The Fourier-based approach gives very good estimates of the optimal parameter γ
- Clearly better than competing methods on difficult problems
- AL approach works well also as a smoother for coupled multigrid
- Current and future work: implementation using the Teko framework in Trilinos; application to real problems.

References

1. M. Benzi and M. A. Olshanskii, *An augmented Lagrangian-based approach to the Oseen problem*, SIAM J. Sci. Comput., 28 (2006), pp. 2095–2113.
2. M. Benzi and M. A. Olshanskii, *Field-of-values analysis of augmented Lagrangian preconditioners for the linearized Navier–Stokes problem*, SIAM J. Numer. Anal., 49 (2011), pp. 770–788.
3. M. Benzi, M. A. Olshanskii and Z. Wang, *Modified augmented Lagrangian preconditioners for the incompressible Navier–Stokes equations*, Intern. J. Numer. Meth. Fluids, 66 (2011), pp. 6185–6202.
4. M. Benzi and Z. Wang, *Analysis of augmented Lagrangian-based preconditioners for the steady incompressible Navier–Stokes equations*, SIAM J. Sci. Comput., 33 (2011), pp. 2761–2784.
5. M. Benzi and Z. Wang, *A parallel implementation of the modified augmented Lagrangian preconditioner for the Navier–Stokes equations*, to appear in Numerical Algorithms.
6. S. Hamilton, M. Benzi, and E. Haber, *New multigrid smoothers for the Oseen problem*, Numer. Linear Algebra Appl., 17 (2010), pp. 557–576.