INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2012; **68**:83–98
Published online 10 December 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/fld.2495

$\mathscr{H}$-LU factorization in preconditioners for augmented Lagrangian and grad-div stabilized saddle point systems

Steffen Börm[1] and Sabine Le Borne[2,*,†]

[1]*Department of Computer Science, Christian Albrechts University Kiel, Kiel, Germany*
[2]*Department of Mathematics, Box 5054, Tennessee Technological University, Cookeville, TN 38505, U.S.A.*

SUMMARY

The (mixed finite element) discretization of the linearized Navier–Stokes equations leads to a linear system of equations of saddle point type. The iterative solution of this linear system requires the construction of suitable preconditioners, especially in the case of high Reynolds numbers. In the past, a stabilizing approach has been suggested which does not change the exact solution but influences the accuracy of the discrete solution as well as the effectiveness of iterative solvers. This stabilization technique can be performed on the continuous side before the discretization, where it is known as 'grad-div' (GD) stabilization, as well as on the discrete side where it is known as an 'augmented Lagrangian' (AL) technique (and does not change the discrete solution). In this paper, we study the applicability of $\mathscr{H}$-LU factorizations to solve the arising subproblems in the different variants of stabilized saddle point systems. We consider both the saddle point systems that arise from the stabilization in the continuous as well as on the discrete setting. Recently, a modified AL preconditioner has been proposed for the system resulting from the discrete stabilization. We provide a straightforward generalization of this approach to the GD stabilization. We conclude the paper with numerical tests for a variety of problems to illustrate the behavior of the considered preconditioners as well as the suitability of $\mathscr{H}$-LU factorization in the preconditioners.

Received 25 June 2010; Revised 14 October 2010; Accepted 16 October 2010

KEY WORDS: saddle point problems; Oseen problem, grad-div stabilization; augmented Lagrangian method; hierarchical matrices; $\mathscr{H}$-LU

1. INTRODUCTION

The ability to solve large, sparse systems arising from the (linearized) Navier–Stokes equations is critical to the simulation of incompressible fluid flow. Linear systems of equations are typically solved (approximately) by iterative methods that have linear storage and computational complexity (per iteration step) in the number of unknowns. However, the rate of convergence may be unacceptably slow, and one needs to accelerate the convergence by suitable preconditioning techniques. The design of robust and efficient preconditioners for linear systems arising in flow simulations is still a challenge.

The continuous problem to be solved reads as follows: Given an open, bounded domain $\Omega \subset \mathbb{R}^d$ ($d = 2, 3$) with boundary $\partial\Omega$, a time interval $[0, T]$, a force function $\mathbf{f}$, boundary data $\mathbf{g}$ and an initial

*Correspondence to: Sabine Le Borne, Department of Mathematics, Box 5054, Tennessee Technological University, Cookeville, TN 38505, U.S.A.
†E-mail: sleborne@tntech.edu

condition $\mathbf{u}_0$, find a velocity field $\mathbf{u}$ and pressure p that satisfy the incompressible Navier–Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega\times(0,T], \tag{1a}$$

$$\operatorname{div}\mathbf{u} = 0 \quad \text{in } \Omega\times[0,T], \tag{1b}$$

$$\mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega\times[0,T], \tag{1c}$$

$$\mathbf{u}(x,0) = \mathbf{u}_0(x) \quad \text{on } \Omega. \tag{1d}$$

Here, ν denotes the kinematic viscosity. Implicit time discretization and (Picard fixed-point) linearization result in a sequence of Oseen problems of the form

$$\sigma\mathbf{u} - \nu\Delta\mathbf{u} + (\mathbf{b}\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega\times(0,T], \tag{2a}$$

$$\operatorname{div}\mathbf{u} = 0 \quad \text{in } \Omega\times[0,T], \tag{2b}$$

$$\mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega\times[0,T], \tag{2c}$$

where $\mathbf{b}$ denotes a divergence-free convection field (namely, the solution of the previous step in the Picard iteration). Spatial discretization by stable mixed finite element techniques results in large, sparse saddle point problems of the type

$$\begin{pmatrix} A & B \\ B^{\mathrm{T}} & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}, \tag{3}$$

where u and p now denote the discrete velocity and pressure, respectively. The matrix block A is a block diagonal matrix with d (spatial dimension) identical blocks along the diagonal representing convection-diffusion operators. The off-diagonal blocks B^{T} and B represent the discrete divergence and gradient operators, respectively.

Numerous solution techniques have been proposed in the literature for this type of a problem. A comprehensive survey [1] reviews many of the most promising solution methods with an emphasis on the iterative solution of these large, sparse, indefinite problems. Several of these preconditioners are based on block approaches which require approximate solves for auxiliary velocity as well as pressure Schur complement problems [2–7]. The constraint preconditioner (and its variants) also employ the given block structure and yield iterates that satisfy the constraint of (discretely) divergence-free velocity exactly [8–10].

In this paper, we will focus on preconditioning techniques based on the grad-div (GD)-stabilized [11–13] or augmented Lagrangian (AL) approach [7, 14, 15]. The underlying idea is to add the term $\gamma\nabla\operatorname{div}\mathbf{u}$ to the left-hand side of the momentum equation of the Navier–Stokes (1a) or Oseen (2a) equations, respectively. On the discrete side, such an addition is represented by adding $\gamma BW^{-1}B^{\mathrm{T}}$ to the first diagonal block A in (3). The matrix W is arbitrary (as long as it is invertible) and will be specified later. In view of the incompressibility constraint (1b), (2b), which is represented by the second block row $B^{\mathrm{T}}u = 0$ in (3), such stabilizations or augmentations in the first equation do not change the exact solution of the saddle point problem.

One of the advantages of such a stabilized formulation is its positive effect on a subsequent iterative solution; the solution of a Schur complement problem $B^{\mathrm{T}}(A_\gamma)^{-1}B$ where $A_\gamma = A + \gamma BW^{-1}B^{\mathrm{T}}$ denotes the augmented diagonal block, is no longer the bottleneck of the iterative solution as it is for many block preconditioning approaches to the original system. In the augmented approach, this Schur complement's inverse is identical to $(B^{\mathrm{T}}A^{-1}B)^{-1} + \gamma W^{-1}$ which yields the basis for robust (Schur complement) preconditioning. Instead, the main difficulty now becomes the robust solution of the augmented diagonal block A_γ itself. The addition of $\gamma BW^{-1}B^{\mathrm{T}}$ creates new entries in the matrix and, depending on the discretization, substantially increases the number of nonzeros. In particular, it leads to a coupling between the previously uncoupled velocity unknowns in distinct spatial directions, i.e. the submatrix A in (3) is no longer block diagonal.

DOI: 10.1002/fld

In [12], a standard multigrid method is used for the solution of the system involving A_γ, whereas a domain-decomposition approach is pursued in [16]. In [14], a multigrid approach was developed based on ideas developed in [17]. In [7], numerical tests are presented in which the equations involving A_γ are solved via exact LU factorizations as well as inner ILU-preconditioned GMRES iterations. In the recent paper [18], the augmented block A_γ is approximated by a block upper triangular matrix. While this typically increases the number of iteration steps in the resulting modified AL preconditioner compared to solving the full system A_γ, it considerably reduces the setup time for the preconditioner as well as the time required per iteration step, leading in many cases to a faster solution method.

In this paper, we propose to use an approximate LU factorization of A_γ computed in the hierarchical ($\mathscr{H}$-)matrix format [19, 20] to precondition this augmented block. Hierarchical matrices divide the matrix into subblocks in which matrix data is approximated by low rank representations. An efficient arithmetic has been developed in this format which permits the computation of an $\mathscr{H}$-LU factorization in almost optimal complexity [21–23]. Here, we will use $\mathscr{H}$-matrices to compute $\mathscr{H}$-LU factorizations of augmented blocks A_γ obtained from the discretization of GD-stabilized systems as well as those obtained from a discrete augmentation $A_\gamma = A + \gamma B W^{-1} B^{\mathrm{T}}$. The application of the $\mathscr{H}$-matrix technique to the related Lamé equations has also been analyzed in a PhD thesis [24].

In addition, we will consider the modified AL approach introduced in [18] as well as its continuous (modified GD) counterpart, and again use $\mathscr{H}$-LU factorizations to approximate the arising subproblems. It turns out that all the considered approaches display the expected almost optimal complexity of $\mathscr{H}$-LU factorization, with the continuous (GD-stabilized) approach leading to much faster $\mathscr{H}$-LU factorizations compared to the discrete (AL) approach.

The remainder of this paper is organized as follows: in Section 2 we introduce the four preconditioners that will be compared with respect to their suitability for $\mathscr{H}$-LU factorization. We present some results on eigenvalues of the preconditioned saddle point systems with references to the literature. In Section 3, we briefly review the $\mathscr{H}$-matrix techniques which we used. Since $\mathscr{H}$-matrix techniques have become rather standard, and no new modifications needed to be developed for the applications considered in this paper, we mostly provide references to the literature for further detail. Section 4 concludes the paper with numerical tests to illustrate the behavior of the various preconditioners using varying accuracies in the $\mathscr{H}$-LU factorizations. Finally, we summarize our conclusions in Section 5.

2. GD AND AL PRECONDITIONERS

We first introduce some notation to distinguish between the continuously GD-stabilized system and the discretely AL-stabilized system. To this end, we define the three saddle point matrices

$$\mathscr{A} := \begin{pmatrix} A & B \\ B^{\mathrm{T}} & 0 \end{pmatrix}, \tag{4}$$

$$\mathscr{A}_{\mathrm{AL}} := \begin{pmatrix} A + \gamma B W^{-1} B^{\mathrm{T}} & B \\ B^{\mathrm{T}} & 0 \end{pmatrix}, \tag{5}$$

$$\mathscr{A}_{\mathrm{GD}} := \begin{pmatrix} A_\gamma^{\mathrm{GD}} & B \\ B^{\mathrm{T}} & 0 \end{pmatrix}, \tag{6}$$

where the matrix $\mathscr{A}$ denotes the (stable, mixed finite element) discretization of the Oseen equations (2a)–(2c), and the matrix $\mathscr{A}_{\mathrm{AL}}$ is the AL matrix computed using the matrix blocks of $\mathscr{A}$, a parameter γ and an invertible matrix W. Whereas other choices are possible, we will exclusively

DOI: 10.1002/fld

use $W := M_{p,\mathrm{diag}}$ to be the diagonally lumped pressure mass matrix. The matrices appearing in the subblocks have the following sizes:

$$A \in \mathbb{R}^{dn \times dn}, \quad B \in \mathbb{R}^{dn \times m}, \quad W \in \mathbb{R}^{m \times m}$$

(n velocity unknowns per spatial direction and m pressure unknowns). The matrix $\mathscr{A}_{\mathrm{GD}}$ denotes the discretized GD-augmented Oseen system in which the term $\gamma \nabla \operatorname{div} \mathbf{u}$ has been added to the momentum Equation (2a) before its discretization. The first diagonal blocks of all three matrices $\mathscr{A}$, $\mathscr{A}_{\mathrm{AL}}$ and $\mathscr{A}_{\mathrm{GD}}$ have a natural $d \times d$ block structure. Here, we will only consider $d = 2$ spatial dimensions, but a generalization to $d = 3$ is straightforward. Denoting the diagonal blocks of A by A_1, A_2 (typically there holds $A_1 = A_2$) and breaking B into $B = [B_1^{\mathrm{T}}, B_2^{\mathrm{T}}]^{\mathrm{T}}$, the first diagonal block of $\mathscr{A}_{\mathrm{AL}}$ has the form

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

We use A_γ to denote the first diagonal block in either $\mathscr{A}_{\mathrm{AL}}$ or $\mathscr{A}_{\mathrm{GD}}$. In this paper we consider the block triangular preconditioner

$$\mathscr{P} := \begin{pmatrix} \widetilde{A_\gamma} & B \\ 0 & -\dfrac{1}{\gamma} \widetilde{W} \end{pmatrix} \tag{7}$$

for the stabilized systems $\mathscr{A}_{\mathrm{AL}}$ or $\mathscr{A}_{\mathrm{GD}}$. The matrices $\widetilde{A_\gamma}$ and $\widetilde{W}$ in (7) are approximations to A_γ and W, respectively. In the case of $\widetilde{A_\gamma} = A_\gamma$ and $\widetilde{W} = W$, the eigenvalues of the preconditioned system

$$\mathscr{A}_{\mathrm{AL}} \mathscr{P}^{-1} = \begin{pmatrix} I_{dn} & 0 \\ B^{\mathrm{T}} A_\gamma^{-1} & \gamma B^{\mathrm{T}} A_\gamma^{-1} B W^{-1} \end{pmatrix} \tag{8}$$

can be expressed in terms of the eigenvalues of the preconditioned Schur complement as stated in the following theorem.

Theorem 1
The preconditioned system $\mathscr{A}_{\mathrm{AL}} \mathscr{P}^{-1}$ has the eigenvalue $\lambda = 1$ of (algebraic) multiplicity at least dn, and each of the remaining eigenvalues is of the form $\lambda = \gamma\mu/(1+\gamma\mu)$, where $\mu \in \sigma(W^{-1} B A^{-1} B^{\mathrm{T}})$ is an eigenvalue of the preconditioned Schur complement.

The proof, which can be found in [14, 18], makes use of the key identity

$$(B^{\mathrm{T}} A_\gamma^{-1} B)^{-1} = S^{-1} + \gamma W^{-1}, \tag{9}$$

for the Schur complement $S := B^{\mathrm{T}} A^{-1} B$.

We note that for $\gamma \to \infty$, all eigenvalues of the preconditioned system (8) tend to one (at the expense of obtaining singularly perturbed systems (5), (6) in view of $\operatorname{rank}(B W^{-1} B^{\mathrm{T}}) \leqslant m \ll dn$).

Remark 1
We have not required the matrix block B to have full rank m. Since we have not imposed an additional condition on the pressure, e.g. of the type $\int_\Omega p \, \mathrm{d}\Omega = 1$, the pressure is only uniquely determined up to a constant by the Oseen equations (2a)–(2c). As a result, the block B is typically of rank $m-1$, which requires special considerations in some preconditioners. For example, the nullspace preconditioner [25], the implicit approximate inverse preconditioner [26], and the $\mathrm{BFB}^{\mathrm{T}}$ (or least squares commutator) preconditioner [5] all require the application of the inverse $(B^{\mathrm{T}} B)^{-1}$ which requires B to be of full rank. Such a constraint is not necessary for the block triangular preconditioner (7).

DOI: 10.1002/fld

In the following, we will not use exact matrices $\widetilde{A}_\gamma = A_\gamma$ and $\widetilde{W} = W$, but rather the following approximations.

We will use either $\widetilde{W} := W = M_{p,\mathrm{diag}}$, which denotes the diagonally lumped pressure mass matrix, or $\widetilde{W} := \mathscr{H}$-Cholesky$(M_p)$, an $\mathscr{H}$-Cholesky factorization of the (unlumped) pressure mass matrix.

The block $\widetilde{A}_\gamma$ in the preconditioner $\mathscr{P}$ is an approximation to the block A_γ in $\mathscr{A}_{\mathrm{AL}}$ (5) or $\mathscr{A}_{\mathrm{GD}}$ (6), respectively. Here, we will consider two preconditioners for each of the systems $\mathscr{A}_{\mathrm{AL}}$ or $\mathscr{A}_{\mathrm{GD}}$. In the case of $\mathscr{A}_{\mathrm{AL}}$, we will either compute an $\mathscr{H}$-LU factorization of the full matrix $\mathscr{A}_\gamma^{\mathrm{AL}}$, or, as suggested in [18], first replace $\mathscr{A}_\gamma^{\mathrm{AL}}$ by a block upper triangular matrix and then compute $\mathscr{H}$-LU factorizations only of the two diagonal blocks. We use the following notation for the resulting preconditioners:

$$\widetilde{A}_\gamma^{\mathrm{AL}} := L_{\mathscr{H}}^{\mathrm{AL}} U_{\mathscr{H}}^{\mathrm{AL}} \quad (\mathscr{H}\text{-LU factors of } A_\gamma^{\mathrm{AL}}), \tag{10}$$

$$\widetilde{A}_{\gamma,\mathrm{tri}}^{\mathrm{AL}} := L_{\mathscr{H},\mathrm{tri}}^{\mathrm{AL}} U_{\mathscr{H},\mathrm{tri}}^{\mathrm{AL}} \tag{11}$$

where $L_{\mathscr{H},\mathrm{tri}}^{\mathrm{AL}}, U_{\mathscr{H},\mathrm{tri}}^{\mathrm{AL}}$ are the $\mathscr{H}$-LU factors of the modified block triangular matrix, i.e.

$$L_{\mathscr{H},\mathrm{tri}}^{\mathrm{AL}} U_{\mathscr{H},\mathrm{tri}}^{\mathrm{AL}} \approx \begin{pmatrix} A_1 + \gamma B_1 W^{-1} B_1^{\mathrm{T}} & \gamma B_1 W^{-1} B_2^{\mathrm{T}} \\ 0 & A_2 + \gamma B_2 W^{-1} B_2^{\mathrm{T}} \end{pmatrix} =: A_{\gamma,\mathrm{tri}}^{\mathrm{AL}}.$$

We will next analyze the eigenvalues of $A_\gamma^{\mathrm{AL}}(A_{\gamma,\mathrm{tri}}^{\mathrm{AL}})^{-1}$ to gain some insight into the behavior of this modified triangular preconditioner in the case of an exact (or highly accurate $\mathscr{H}$-) LU factorization.

Theorem 2
The spectrum of the matrix $A_\gamma^{\mathrm{AL}}(A_{\gamma,\mathrm{tri}}^{\mathrm{AL}})^{-1}$ is contained in

$$\sigma(A_\gamma^{\mathrm{AL}}(A_{\gamma,\mathrm{tri}}^{\mathrm{AL}})^{-1}) \subseteq \{1\} \cup \left\{ 1 - \frac{\gamma^2}{\bar{\mu} + \gamma^2} \mid \bar{\mu} \in \sigma(W S_1^{-1}(W + \gamma S) S_2^{-1}) \right\}$$

where $S_1 := B_1^{\mathrm{T}} A_1^{-1} B_1$ and $S_2 := B_2^{\mathrm{T}} A_2^{-1} B_2$, and $S = B^{\mathrm{T}} A^{-1} B$ denotes the (full) Schur complement $S = S_1 + S_2$.

Proof
Introducing $A_{1,\gamma} := A_1 + \gamma B_1 W^{-1} B_1^{\mathrm{T}}$ and $A_{2,\gamma} := A_2 + \gamma B_2 W^{-1} B_2^{\mathrm{T}}$, we have

$$A_\gamma^{\mathrm{AL}} = \begin{pmatrix} A_{1,\gamma} & \gamma B_1 W^{-1} B_2^{\mathrm{T}} \\ \gamma B_2 W^{-1} B_1^{\mathrm{T}} & A_{2,\gamma} \end{pmatrix}, \qquad A_{\gamma,\mathrm{tri}}^{\mathrm{AL}} = \begin{pmatrix} A_{1,\gamma} & \gamma B_1 W^{-1} B_2^{\mathrm{T}} \\ 0 & A_{2,\gamma} \end{pmatrix},$$

$$(A_{\gamma,\mathrm{tri}}^{\mathrm{AL}})^{-1} = \begin{pmatrix} A_{1,\gamma}^{-1} & -\gamma A_{1,\gamma}^{-1} B_1 W^{-1} B_2^{\mathrm{T}} A_{2,\gamma}^{-1} \\ 0 & A_{2,\gamma}^{-1} \end{pmatrix}$$

and the preconditioned matrix is given by

$$\begin{aligned} A_\gamma^{\mathrm{AL}}(A_{\gamma,\mathrm{tri}}^{\mathrm{AL}})^{-1} &= \begin{pmatrix} A_{1,\gamma} & \gamma B_1 W^{-1} B_2^{\mathrm{T}} \\ \gamma B_2 W^{-1} B_1^{\mathrm{T}} & A_{2,\gamma} \end{pmatrix} \begin{pmatrix} A_{1,\gamma}^{-1} & -\gamma A_{1,\gamma}^{-1} B_1 W^{-1} B_2^{\mathrm{T}} A_{2,\gamma}^{-1} \\ 0 & A_{2,\gamma}^{-1} \end{pmatrix} \\ &= \begin{pmatrix} I & 0 \\ \gamma B_2 W^{-1} B_1^{\mathrm{T}} A_{1,\gamma}^{-1} & I - \gamma^2 B_2 W^{-1} B_1^{\mathrm{T}} A_{1,\gamma}^{-1} B_1 W^{-1} B_2^{\mathrm{T}} A_{2,\gamma}^{-1} \end{pmatrix} \end{aligned}$$

which implies that $A_\gamma^{\mathrm{AL}}(A_{\gamma,\mathrm{tri}}^{\mathrm{AL}})^{-1}$ has an eigenvalue $\lambda = 1$ of at least multiplicity dn. To determine the remaining eigenvalues, we analyze the spectrum of the second diagonal block.

DOI: 10.1002/fld

Let λ be an eigenvalue of the matrix

$$I-\gamma^2 B_2 W^{-1} B_1^{\mathrm{T}} A_{1,\gamma}^{-1} B_1 W^{-1} B_2^{\mathrm{T}} A_{2,\gamma}^{-1}.$$

We have to consider only the case $\lambda \neq 1$, since we already know that 1 is an eigenvalue.

Our equation means that $\hat{\lambda}=(1-\lambda)/\gamma^2$ is an eigenvalue of

$$B_2 W^{-1} B_1^{\mathrm{T}} A_{1,\gamma}^{-1} B_1 W^{-1} B_2^{\mathrm{T}} A_{2,\gamma}^{-1},$$

and since $\lambda \neq 1$, we have $\hat{\lambda} \neq 0$, so due to the well-known identity $\sigma(XY)\backslash\{0\}=\sigma(YX)\backslash\{0\}$, we find that $\hat{\lambda}$ also has to be an eigenvalue of

$$W^{-1} B_1^{\mathrm{T}} A_{1,\gamma}^{-1} B_1 W^{-1} B_2^{\mathrm{T}} A_{2,\gamma}^{-1} B_2.$$

Using the identity

$$B_i^{\mathrm{T}} A_{i,\gamma}^{-1} B_i = (S_i^{-1}+\gamma W^{-1})^{-1} \quad \text{for } i \in \{1,2\}$$

(cf. (9)), we obtain

$$\begin{aligned} W^{-1} B_1^{\mathrm{T}} A_{1,\gamma}^{-1} B_1 W^{-1} B_2^{\mathrm{T}} A_{2,\gamma}^{-1} B_2 &= W^{-1}(S_1^{-1}+\gamma W^{-1})^{-1} W^{-1}(S_2^{-1}+\gamma W^{-1})^{-1} \\ &= ((S_2^{-1}+\gamma W^{-1}) W (S_1^{-1}+\gamma W^{-1}) W)^{-1} \\ &= (S_2^{-1}(\gamma S+W) S_1^{-1} W + \gamma^2 I)^{-1}. \end{aligned}$$

We conclude that $\hat{\lambda}$ is an eigenvalue of

$$(S_2^{-1}(\gamma S+W) S_1^{-1} W + \gamma^2 I)^{-1},$$

therefore $1/\hat{\lambda}$ has to be an eigenvalue of

$$S_2^{-1}(\gamma S+W) S_1^{-1} W + \gamma^2 I,$$

and $\mu := 1/\hat{\lambda} - \gamma^2$ must be an eigenvalue of

$$S_2^{-1}(\gamma S+W) S_1^{-1} W.$$

This means

$$\lambda = 1+\gamma^2\hat{\lambda} = 1+\gamma^2 \frac{1}{\mu+\gamma^2} = 1+\frac{\gamma^2}{\mu+\gamma^2}$$

for $\mu \in \sigma(S_2^{-1}(\gamma S+W) S_1^{-1} W)$. □

We will use the corresponding preconditioners for the GD-stabilized system, i.e.

$$\widetilde{A}_\gamma^{\mathrm{GD}} := L_{\mathscr{H}}^{\mathrm{GD}} U_{\mathscr{H}}^{\mathrm{GD}} \quad (\mathscr{H}\text{-LU factors of } A_\gamma^{\mathrm{GD}}), \tag{12}$$

$$\widetilde{A}_{\gamma,\mathrm{tri}}^{\mathrm{GD}} := L_{\mathscr{H},\mathrm{tri}}^{\mathrm{GD}} U_{\mathscr{H},\mathrm{tri}}^{\mathrm{GD}} \tag{13}$$

where $L_{\mathscr{H},\mathrm{tri}}^{\mathrm{GD}}, U_{\mathscr{H},\mathrm{tri}}^{\mathrm{GD}}$ are the $\mathscr{H}$-LU factors of $A_{\gamma,tri}^{\mathrm{GD}}$ which results from A_γ^{GD} by replacing the 2×1 block by a 0-block, i.e. $A_{\gamma,tri}^{\mathrm{GD}}$ denotes the finite element discretization of

$$\nu\Delta\mathbf{u}+(\mathbf{b}\cdot\nabla)\mathbf{u}+\gamma \begin{pmatrix} \frac{\partial^2}{\partial x^2} & \frac{\partial^2}{\partial x \partial y} \\ 0 & \frac{\partial^2}{\partial y^2} \end{pmatrix} \mathbf{u} = \begin{pmatrix} \nu\Delta+(\mathbf{b}\cdot\nabla)+\gamma\frac{\partial^2}{\partial x^2} & \frac{\partial^2}{\partial x\partial y} \\ 0 & \nu\Delta+(\mathbf{b}\cdot\nabla)+\gamma\frac{\partial^2}{\partial y^2} \end{pmatrix} \mathbf{u}$$

where we assumed $d=2$ spatial dimensions, i.e. $\mathbf{u}=(u_1,u_2)^{\mathrm{T}}$.

DOI: 10.1002/fld

3. $\mathscr{H}$-LU PRECONDITIONING OF THE AL

In the previous section, we have proposed four preconditioners (10), (11), (12), (13) based on $\mathscr{H}$-LU factorizations of GD-stabilized or AL saddle point systems, respectively. In this section, we will specify the particular variant of $\mathscr{H}$-matrix construction which is proposed for the $\mathscr{H}$-LU factorization of these augmented systems, and point to the literature for further details on $\mathscr{H}$-matrices.

$\mathscr{H}$-matrices have been introduced in [27] and since then entered into a wide range of applications. The basic $\mathscr{H}$-matrix construction and corresponding arithmetic have reached a relatively mature state and are documented in the comprehensive lecture notes [28] and books [20, 29]. $\mathscr{H}$-matrices are based on a hierarchical subdivision of the matrix into subblocks and the low-rank approximation of matrix data within these subblocks. Originally, $\mathscr{H}$-matrices have been introduced in the context of fully populated matrices arising from solution operators of elliptic differential equations and in boundary element methods. In the finite element context, the stiffness matrix itself does not require an efficient approximation by an $\mathscr{H}$-matrix since it is sparse. Its LU-factors, however, suffer from inacceptable fill-in if computed exactly and can be efficiently computed and approximated by $\mathscr{H}$-LU factors [23, 30]. The $\mathscr{H}$-matrix construction and arithmetic as originally developed for fully populated matrices have a straightforward generalization to sparse matrices. However, there are two modifications for $\mathscr{H}$-matrices which have been designed for sparse matrices in particular and have been used here.

The first modification concerns the construction of the block structure of the $\mathscr{H}$-matrix. In the classical $\mathscr{H}$-matrix, the block structure is generated through a repeated bisection of the respective index sets, i.e. row and column index sets are divided into two subsets, respectively, which leads to four matrix subblocks. In the case of sparse matrices, the bisection has been replaced by a nested dissection approach in which row and column index sets are divided into three subsets each; two subsets S_1, S_2 of indices that are pairwise disconnected in the sense that $a_{ij}=0=a_{ji}$ if $i \in S_1,\ j \in S_2$ where $A=(a_{ij})$ denotes the stiffness matrix, and a third subset S_3 containing the remaining indices of the interior boundary. Such a subdivision results in a 3×3 matrix structure with zero blocks in the 1×2 and 2×1 positions. These zero blocks remain zero in a subsequent LU factorization which results in considerably faster $\mathscr{H}$-LU factorizations compared to bisection-based $\mathscr{H}$-matrices [21, 31].

The second modification concerns the development of a 'blackbox' clustering algorithm. The classical construction of $\mathscr{H}$-matrices requires geometric information associated with the underlying indices in order to determine a suitable block structure. For sparse matrices, the information contained in the associated matrix graph can replace the need for geometric information [22].

Whereas the classical $\mathscr{H}$-matrix uses a fixed rank for the low rank approximations within matrix subblocks, it is possible to replace it by *adaptive ranks* in order to enforce a desired accuracy within the individual blocks. In particular, given a matrix block C and a desired $\mathscr{H}$-accuracy $0<\delta_{\mathscr{H}}<1$, we set the rank k_C of the approximation to C as

$$k_C := \min\{k' \mid \sigma_{k'} \leqslant \delta_{\mathscr{H}} \sigma_1\}$$

where σ_i denotes the ith largest singular value of C. These modifications from the classical $\mathscr{H}$-matrix have led to highly efficient $\mathscr{H}$-LU preconditioners for a wide range of sparse matrices [32].

4. NUMERICAL TESTS

Numerical tests were performed on an HP ProLiant DL785 server using AMD Opteron 8400 series processors. As a model problem, we chose the stationary (i.e. $\sigma=0$) two-dimensional Oseen

DOI: 10.1002/fld

problem (2a)–(2c) with varying convection dominance $v \in [10^{-3}, 1]$, and recirculating convection

$$\mathbf{b}(x,y) = \begin{pmatrix} \dfrac{0.1\mathrm{e}^{0.1y}}{2\pi(\mathrm{e}^{0.1}-1)} \sin\left(\dfrac{2\pi(\mathrm{e}^{0.1y}-1)}{\mathrm{e}^{0.1}-1}\right)\left(1-\cos\left(\dfrac{2\pi(\mathrm{e}^{4x}-1)}{\mathrm{e}^{4}-1}\right)\right) \\ -\dfrac{4\mathrm{e}^{4x}}{2\pi(\mathrm{e}^{4}-1)} \sin\left(\dfrac{2\pi(\mathrm{e}^{4x}-1)}{\mathrm{e}^{4}-1}\right)\left(1-\cos\left(\dfrac{2\pi(\mathrm{e}^{0.1y}-1)}{\mathrm{e}^{0.1}-1}\right)\right) \end{pmatrix}.$$

We set up the discrete Oseen equations using a Taylor–Hood finite element discretization on a structured mesh on $\Omega = [-1,1] \times [-1,1]$ with Tabata's upwind triangle scheme [33, Chapter 3, Section 3.1.1]. We set the boundary conditions and force function $\mathbf{f}$ so that the exact solution of the Oseen problem (2a)–(2c) is known to be $\mathbf{u}(x,y) = (\sin x \sin y, \cos x \cos y)^{\mathrm{T}}$, $p(x,y) = 2\cos x \sin y$.

Whereas the GD stabilization does not change the continuous solution, it yields a different discrete solution compared to the discrete Oseen solution in (3). The discrete AL stabilization (5), however, does not lead to a change in the discrete solution. This difference in the accuracy of computed discrete approximations has been investigated in [11–13] to be related to the resolution of the pressure. In the model problem of the current paper, the difference in accuracy of discrete approximations appeared to be very minor, and in the following, we will focus our attention on the numerical behavior of iterative solvers for the discrete systems of equations.

As an outer iteration, we use the GMRES method without restart. We iterate until the residual is reduced by a factor of 10^{-6}, or until at most 200 iteration steps have been reached.

Our test problems are selected in order to compare the four methods considered in this paper. We use the block triangular preconditioner (7) with one of the following four approximations $\widetilde{A}_\gamma$ (and $\widetilde{W}$):

$\widetilde{\mathscr{A}}_\gamma^{\mathrm{AL}}$: the $\mathscr{H}$-LU preconditioner (11) of the full, discretely augmented matrix block A_γ^{AL} (5), referred to as 'full AL,d' ($\widetilde{W}$ in (7) is the diagonally lumped mass matrix) or 'full AL,m' ($\widetilde{W}$ in (7) is the $\mathscr{H}$-Cholesky factorization of the (exact, unlumped) mass matrix);

$\widetilde{\mathscr{A}}_{\gamma,\mathrm{tri}}^{\mathrm{AL}}$: the $\mathscr{H}$-LU preconditioner (10) of the modified triangular, discretely augmented matrix block A_γ^{AL} (5), referred to as 'tri AL,d' or 'tri AL,m', respectively;

$\widetilde{\mathscr{A}}_\gamma^{\mathrm{GD}}$: the $\mathscr{H}$-LU preconditioner (12) of the full, GD-augmented matrix block A_γ^{GD} (6), referred to as 'full GD,d' or 'full GD,m', respectively;

$\widetilde{\mathscr{A}}_{\gamma,\mathrm{tri}}^{\mathrm{GD}}$: the $\mathscr{H}$-LU preconditioner (13) of the modified triangular, GD stabilized matrix block A_γ^{GD} (6), referred to as 'tri GD,d' or 'tri GD,m', respectively.

We will compare these four preconditioners with respect to their dependence on

- the problem size $dn+m$,
- the convection dominance v,
- the stabilization parameter γ, and
- the accuracy of the solvers for the subproblems as determined by the $\mathscr{H}$-accuracy $\delta_{\mathscr{H}}$.

In order to gain a better understanding of the difference in set-up times, we begin with plots of the sparsity structures of the augmented matrix blocks A_γ^{AL} as well as A_γ^{GD} in Figure 1. The matrix A_γ^{AL} is visibly less sparse than A_γ^{GD}; in numbers, A_γ^{AL} has 235 542 non-zero entries (2.1%) compared to 39 364 non-zero entries (0.3%) in A_γ^{GD}. The $\mathscr{H}$-LU factorization exploits the sparsity structure by using a nested dissection reordering before the LU factorization. Figure 2 shows the sparsity patterns of the reordered matrices, displaying the expected 3×3 block structures with zero blocks in the 1×2 and 2×1 block positions. Since these two blocks remain zero in an ($\mathscr{H}$-)LU-factorization, one only requires ($\mathscr{H}$-)LU factorizations of the two diagonal blocks (which are in turn ordered by nested dissection) and an ($\mathscr{H}$-)LU factorization of the Schur complement in the third block. Since this Schur complement LU factorization dominates the work required for the overall LU factorization, we expect much faster setup times for the sparser matrix A_γ^{GD} for which the third diagonal block in the reordered matrix is much smaller than in A_γ^{AL}.

DOI: 10.1002/fld

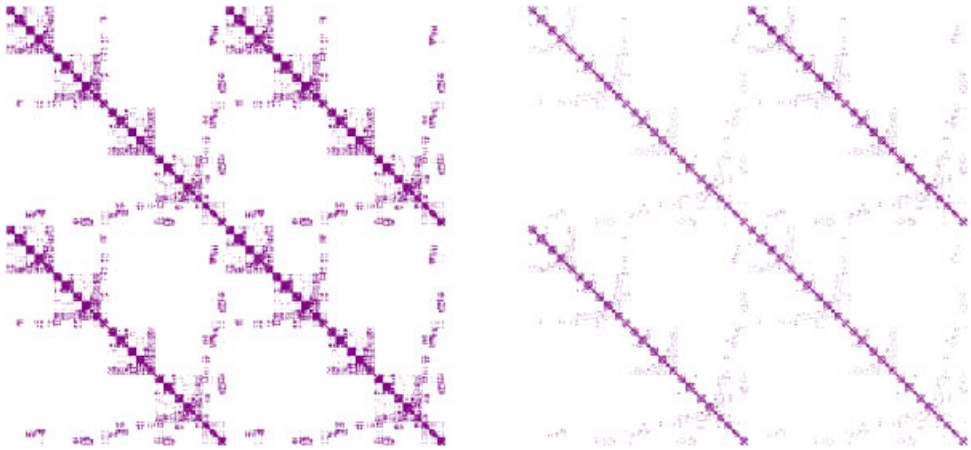

Figure 1. Sparsity patterns of A_γ^{AL} (left) and A_γ^{GD} (right), both of size 3362×3362.

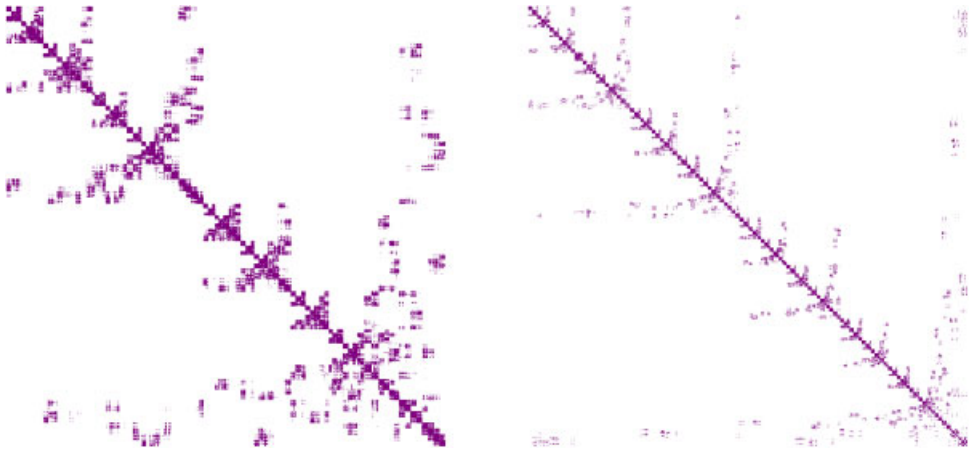

Figure 2. Sparsity patterns of re ordered A_γ^{AL} (left) and A_γ^{GD} (right), both of size 3362×3362.

Unfortunately, we cannot provide any direct comparison with setup times for alternate solvers from the literature such as geometric/algebraic multigrid solvers. However, [30] provides a comparison of $\mathscr{H}$-LU preconditioned Krylov methods and geometric multigrid methods for convection–diffusion problems. We hypothesize that the $\mathscr{H}$-LU setup costs will exceed the setup costs for multigrid methods but will stay well below the costs for an exact solver (for sufficiently large problems).

In Figure 3, we show the setup times (top) and iteration times (bottom) in seconds for various problems sizes (along the x-axis) for the four preconditioners for the stabilization parameter $\gamma = 1.0$ on logarithmic scales. The $\mathscr{H}$-accuracy is set to $\delta_{\mathscr{H}} = 10^{-4}$, and the viscosity parameter is $\nu = 0.01$. For all cases, the setup times as well as iteration times are (almost) linear with respect to the problem size. However, the setup times for the modified triangular variants are considerably shorter than the full methods, and the preconditioners based on GD stabilization are set up faster than the corresponding AL preconditioners. Overall, the setup times are significantly larger than the iteration times (different ranges are used on the y-axis), and the full methods converge faster than the corresponding modified triangular variants.

Figure 4 repeats the experiments of Figure 3 with a different stabilization parameter $\gamma = 0.01$. The setup times (top) are smaller (by different degrees) than the corresponding setup times of Figure 3. The smaller stabilization parameter slows down the iteration times of the full methods

DOI: 10.1002/fld

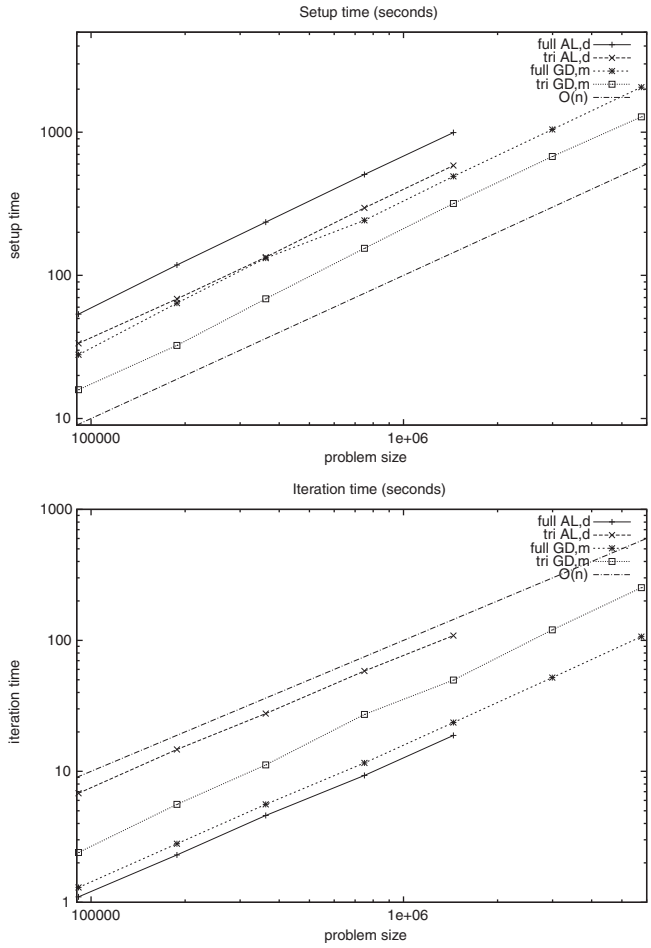

Figure 3. Setup times (top) and iteration times (bottom) for $\gamma = 1.0$.

whereas it accelerates the modified triangular AL preconditioned iteration. This is expected since as the stabilization parameter γ decreases, i.e. $\gamma \to 0$, the modified triangular and full versions both approximate the same block triangular preconditioner of the (unstabilized) Oseen problem, i.e. $\widetilde{\mathscr{A}}_{\gamma}^{\mathrm{AL}} \approx \widetilde{\mathscr{A}}_{\gamma,\mathrm{tri}}^{\mathrm{AL}}$ and $\widetilde{\mathscr{A}}_{\gamma}^{\mathrm{GD}} \approx \widetilde{\mathscr{A}}_{\gamma,\mathrm{tri}}^{\mathrm{GD}}$. In general, $\gamma = 1.0$ appears to be a good choice for the full methods, independent of problem size or convection dominance (see also Figure 5), whereas the choice of a good stabilization parameter γ in the case of the modified triangular variants appears to depend on the viscosity parameter ν (see also Figure 5).

In Figure 5 we illustrate the influence of varying convection dominance on the preconditioners. The top diagram of Figure 5 shows the number of iteration steps using the modified triangular GD preconditioner $\widetilde{\mathscr{A}}_{\gamma,\mathrm{tri}}^{\mathrm{GD}}$ with $\widetilde{W}=\mathscr{H}$-LU($M_p$). For each value of the viscosity parameter $\nu \in \{10^{-1}, 10^{-2}, 10^{-3}\}$, we performed tests for stabilization parameters $\gamma \in \{1.0, 10^{-1}, 10^{-2}, 10^{-3}\}$, but we only show the results for the choice of γ with the smallest number of iterations, resulting in the varying choices of γ in Figure 5. We observe that the performance of the modified triangular GD preconditioner deteriorates with respect to increasing convection dominance, but appears to be quite robust with respect to the problem size for fixed viscosity parameter ν. Furthermore, the optimal stabilization parameter γ depends on the viscosity parameter ν and needs to be decreased as ν decreases. We repeated the same tests with the higher $\mathscr{H}$-accuracy $\delta_{\mathscr{H}} = 10^{-6}$, but the number of iteration steps could not be reduced by more accurate subproblem solvers.

DOI: 10.1002/fld

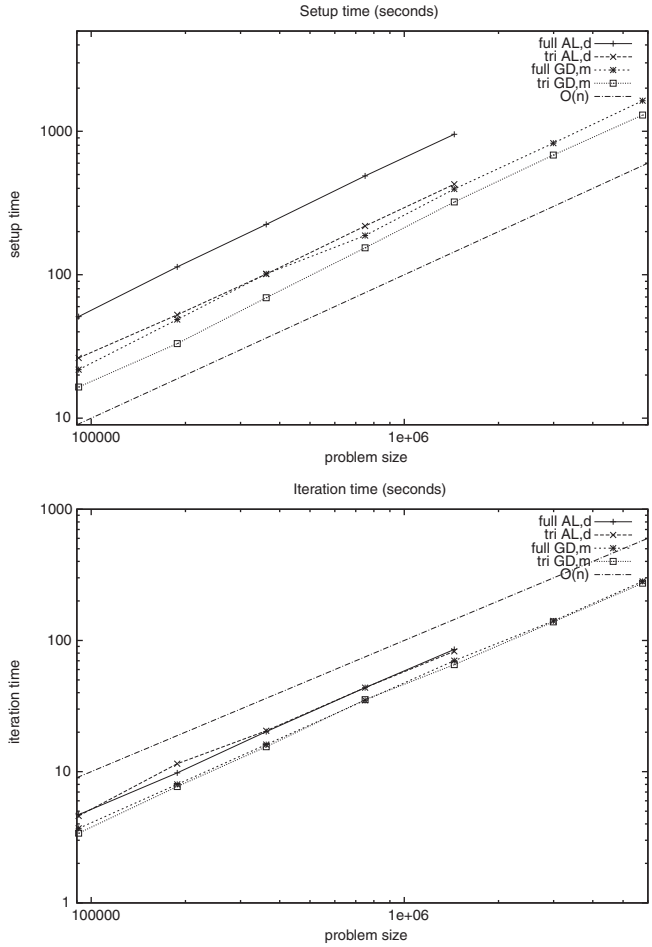

Figure 4. Setup times (top) and iteration times (bottom) for $\gamma = 0.01$.

At the bottom of Figure 5, we show the respective results for the full GD preconditioner $\widetilde{\mathscr{A}}_{\gamma}^{\mathrm{GD}}$ with $\widetilde{W}=\mathscr{H}\text{-LU}(M_p)$ and stabilization parameter $\gamma = 1.0$. Here, the increase in iteration steps with respect to problem size for the convection dominant case $v = 10^{-3}$ is due to the increased difficulty of the subproblems to be solved. We repeated these experiments with $\mathscr{H}$-accuracy $\delta_{\mathscr{H}} = 10^{-6}$ instead of $\delta_{\mathscr{H}} = 10^{-4}$, and now all numbers of iteration steps were 9 or 10, showing robustness both with respect to convection dominance v and problem size.

We also repeated this set of experiments for the full as well as modified triangular discretely augmented preconditioners $\widetilde{\mathscr{A}}_{\gamma}^{\mathrm{AL}}$, $\widetilde{\mathscr{A}}_{\gamma,\mathrm{tri}}^{\mathrm{AL}}$, but do not document these results here since they were similar in character to those for the respective GD preconditioners.

The deteriorating convergence behavior that is observed for $v \to 0$ is only of theoretical interest but not a drawback of the method since only values $v \in [10^{-3}, 0]$ are physically meaningful.

Figure 6 illustrates the influence of the choice of $\widetilde{W}$ in the preconditioner (7). On the top, $\widetilde{W}$ equals the diagonally lumped mass matrix, whereas at the bottom $\widetilde{W}$ equals the $\mathscr{H}$-Cholesky factorization of the (unlumped) mass matrix. We use the stabilization parameter $\gamma = 1.0$ for the full preconditioners and $\gamma = 0.1$ for the modified triangular variants. Whereas all preconditioners are optimal with respect to the problem size, regardless of the choice of $\widetilde{W}$ in (7), both the full and modified triangular AL preconditioners require fewer steps when the diagonally lumped mass matrix is used, i.e. when $\widetilde{W}$ equals the matrix W used in the discrete stabilization (5). On the other hand, both the full and modified triangular GD preconditioners require fewer steps when the $\mathscr{H}$-Cholesky factorization of the full mass matrix is used.

DOI: 10.1002/fld

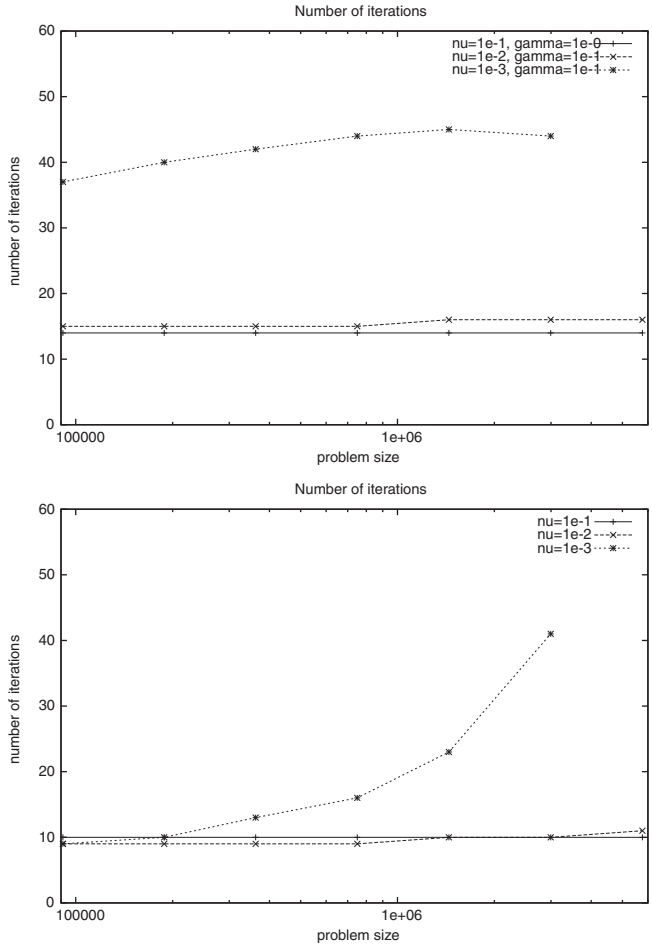

Figure 5. Number of iteration steps for varying convection dominance for modified triangular grad-div (top) and full grad-div (bottom) preconditioners ($\delta_{\mathscr{H}} = 10^{-4}$, varying γ (top), $\gamma = 1.0$ (bottom)).

We also performed tests to analyze the influence of the $\mathscr{H}$-accuracy $\delta_{\mathscr{H}}$ used in the subproblem solvers. In Figure 7 we show the results for the full GD preconditioner $\widetilde{A}_{\gamma}^{\mathrm{GD}}$ for the convection-dominated case $\nu = 10^{-3}$. On the top, we used $\widetilde{W} = \mathscr{H}$-Cholesky($M_p$) whereas at the bottom, $\widetilde{W}$ is the diagonally lumped mass matrix. In either case, $\delta_{\mathscr{H}} > 10^{-4}$ was typically not accurate enough to obtain optimal convergence rates, but no improvement was gained by reducing $\delta_{\mathscr{H}}$ below 10^{-5}. For less convection dominance, $\delta_{\mathscr{H}} = 10^{-4}$ was already sufficient. In particular, only the full variants of the preconditioners required $\delta_{\mathscr{H}} < 10^{-4}$ for best convergence in the higher dimensions—see also the comments on Figure 5 (bottom).

Finally, we provide (very limited) results comparing the performances of the modified triangular GD preconditioner for the augmented problem (6) with two preconditioners for the original Oseen problem (3). Table I shows the set-up times, iteration times and number of iteration steps to solve the Oseen problem with $\nu = 10^{-2}$ and recirculating convection. In the GD preconditioner $\widetilde{A}_{\gamma,\mathrm{tri}}^{\mathrm{GD}}$, we used stabilization parameter $\gamma = 10^{-2}$. We compare with the BFB$^{\mathrm{T}}$ preconditioner [2, 4, 5] as well as the implicit approximate inverse preconditioner [26]. These two preconditioners have the same setup times since they both require preconditioners for the first diagonal block A in (3) as well as to the product $B^{\mathrm{T}}B$ of the discrete divergence and gradient. In Table I, we used $\mathscr{H}$-LU factorizations accuracies $\delta_{\mathscr{H}} = 10^{-4}$ to precondition A, and we used an $\mathscr{H}$-LU factorization of $B^{\mathrm{T}}B$ of $\mathscr{H}$-accuracy 10^{-8}. Since setup times and iteration times vary significantly with varying parameters (i.e. $\mathscr{H}$-accuracy, stabilization parameter γ, problem size, convection dominance, etc.),

DOI: 10.1002/fld

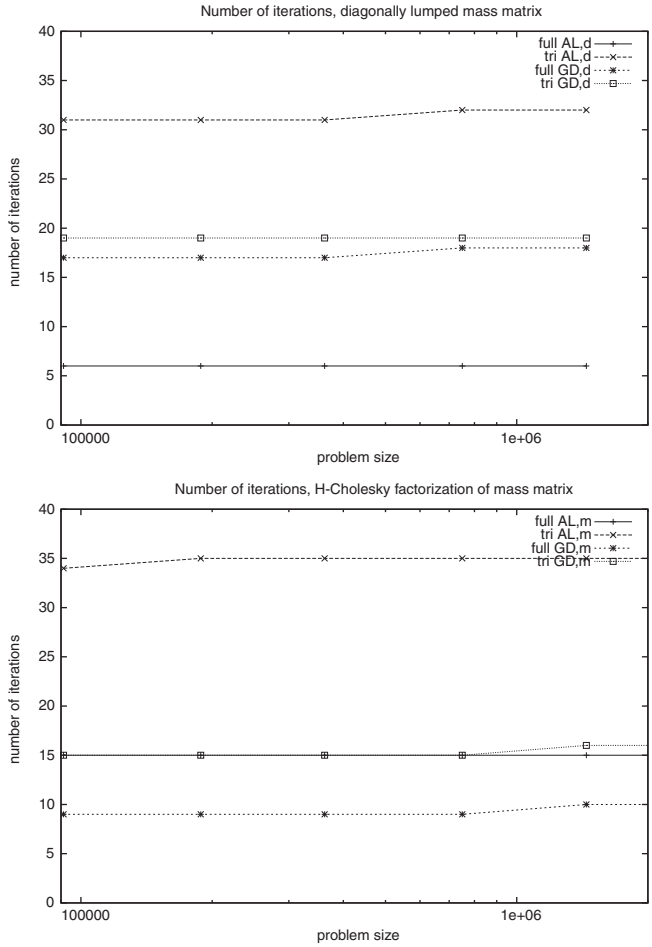

Figure 6. Number of iteration steps for different choices of $\widetilde{W}$ in (7) ($\delta_{\mathscr{H}} = 10^{-4}$, $\nu = 10^{-2}$).

these results and the following interpretation should be treated with caution. The setup times for the GD-stabilized preconditioner are almost twice as high as those for the original system. This can be explained by the fact that the majority of the set-up time lies in the $\mathscr{H}$-LU factorization of the first diagonal block A or A_γ^{GD}, resp. Since A is a block-diagonal matrix with two identical blocks on the diagonal, we factor only one of them. The GD-stabilized matrix A_γ^{GD}, however, has two different blocks on the diagonal, both of which have to be factored, thus about twice the amount of time. On a parallel machine, both blocks could be factored in parallel, leading to more favorable setup times for the triangular GD preconditioner. In terms of iteration time, the triangular GD preconditioner leads to the best results. The iteration time is the smallest, and here we have robustness with respect to the problem size whereas for the other two preconditioners, the number of iterations increases, although only mildly, as the problem size increases.

5. CONCLUSIONS

The novelty of this paper lies in the application of $\mathscr{H}$-LU factorizations for preconditioning the stabilized Oseen problems, and in the generalization of the recently introduced modified triangular AL preconditioner to the modified triangular GD preconditioner. The main results, as demonstrated

DOI: 10.1002/fld

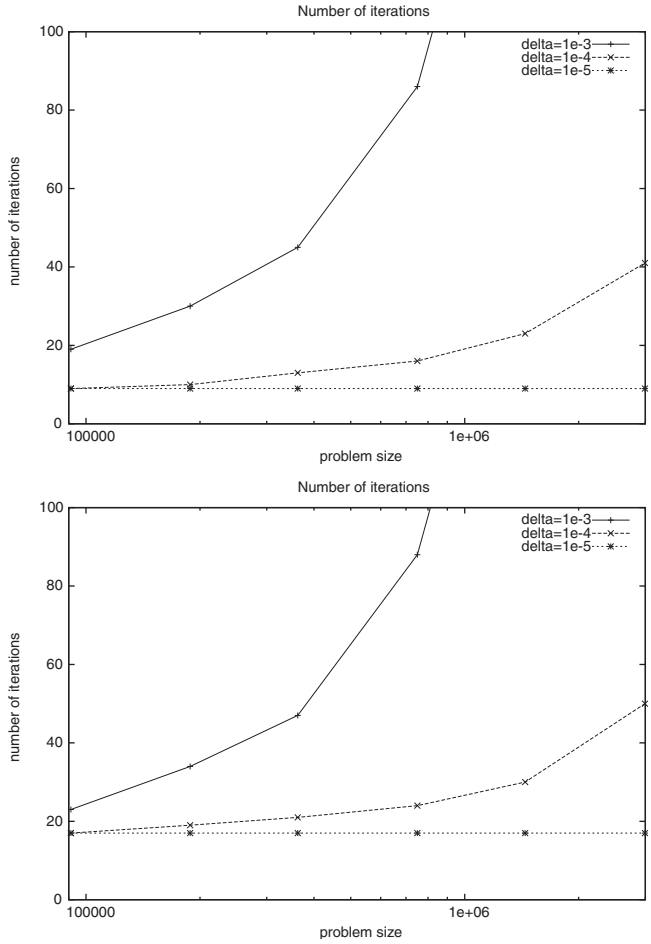

Figure 7. Number of iteration steps for different $\mathscr{H}$-accuracies $\delta_{\mathscr{H}}$, $\mathscr{H}$-Cholesky mass matrix (top) and diagonally lumped mass matrix (bottom) ($\nu = 10^{-3}$).

through our numerical tests, are the following:

- $\mathscr{H}$-matrices are suited for the computation of approximate LU factors of the augmented systems, i.e. setup times are linear with respect to the problem sizes. If the $\mathscr{H}$-accuracy, given by $\delta_{\mathscr{H}}$, is chosen sufficiently small (here, $\delta_{\mathscr{H}} = 10^{-4}$), then very robust solvers are obtained. The major drawbacks are rather large constants in these almost linear complexities, resulting in setup times clearly dominating the actual iteration times.
- Adding setup and iteration times, the fastest method appears to be the newly introduced modified triangular GD preconditioner.
- The setup times are much smaller if we first stabilize, then discretize (GD) rather than first discretize and then stabilize (AL). The convergence behavior of the subsequent iteration is similar for both approaches and both converge to the same solution.
- The full variants of the preconditioners have larger setup times than the modified triangular variants. However, the full preconditioners lead to optimal solvers that are robust with respect to problem size as well as convection dominance. The modified triangular variants are no longer robust with respect to convection dominance, and here the question of the suitable selection of the stabilization parameter γ arises. This question is also discussed in a recent paper [34]. In this paper, a Fourier analysis is used to devise an effective method for the choice of the augmentation parameter γ, and numerical results demonstrate robustness with respect to the mesh size and only slight dependence on the viscosity for this choice of γ. In

DOI: 10.1002/fld

Table I. Comparison with preconditioning of the original Oseen system (3)

	Triangular grad-div		BFB^T		Implicit inverse	
n	Setup	iter/steps	Setup	iter/steps	Setup	iter/steps
1 444 806	305s	108s/34	164s	154s/44	164s	164s/39
2 890 002	684s	223s/34	357s	357s/49	357s	366s/42

particular, it is stated in [34] that for all interesting problem settings, the optimal parameter γ lies in the interval $[0.001, 1]$. This statement agrees with the numerical results of this current paper.

ACKNOWLEDGEMENTS

This work was supported in part by the National Science Foundation under grant No. DMS-0913017.

REFERENCES

1. Benzi M, Golub GH, Liesen J. Numerical solution of saddle point problems. *Acta Numerica* 2005; **14**:1–137.
2. Olshanskii MA, Vassilevski YV. Pressure Schur complement preconditioners for the discrete Oseen problem. *SIAM Journal on Scientific Computing* 2007; **29**:2686–2704.
3. Kay D, Loghin D, Wathen A. A preconditioner for the steady-state Navier–Stokes equations. *SIAM Journal on Scientific Computing* 2002; **24**:237–256.
4. Elman HC. Preconditioning of the steady-state Navier–Stokes equations with low viscosity. *SIAM Journal on Scientific Computing* 1999; **20**:1299–1316.
5. Elman HC, Howle VE, Shadid J, Shuttlerworth R, Tumarino R. Block preconditioners based on approximate commutators. *SIAM Journal on Scientific Computing* 2005; **27**:1651–1668.
6. Le Borne S. Hierarchical matrix preconditioners for the Oseen equations. *Computing and Visualization in Science* 2008; **11**:147–158.
7. de Niet AC, Wubs FW. Two preconditioners for saddle point problems in fluid flows. *International Journal for Numerical Methods in Fluids* 2007; **54**:355–377.
8. Keller C, Gould NIM, Wathen AJ. Constraint preconditioning for indefinite linear systems. *SIAM Journal on Matrix Analysis and Applications* 2000; **21**:1300–1317.
9. Haws JC, Meyer CD. Preconditioning KKT systems. *Technical Report M&CT-Tech-01-021*, The Boing Company, 2003.
10. Gould NIM, Hribar ME, Nocedal J. On the solution of equality constrained quadratic programming problems arising in optimization. *SIAM Journal on Scientific Computing* 2001; **23**:1376–1395.
11. Olshanskii MA. A low order Galerkin finite element method for the Navier–Sokes equations of steady incompressible flow: a stabilization issue and iterative methods. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**:5515–5536.
12. Olshanskii MA, Reusken A. Grad-div stabilization for Stokes equations. *Mathematics of Computation* 2004; **73**:1699–1718.
13. Olshanskii MA, Lube G, Heister T, Löwe J. Grad-div stabilization and subgrid pressure models for incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 2009; **198**:3975–3988.
14. Benzi M, Olshanskii MA. An augmented Lagrangian approach to the Oseen problem. *SIAM Journal on Scientific Computing* 2006; **28**:1095–2113.
15. Olshanskii MA, Benzi M. An augmented Lagrangian approach to linearized problems in hydrodynamic stability. *SIAM Journal on Scientific Computing* 2008; **30**:1459–1473.
16. Dohrmann C, Lehoucq R. A primal-based penalty preconditioner for elliptic saddle point systems. *SIAM Journal on Numerical Analysis* 2006; **44**:270–282.
17. Schöberl J. Multigrid methods for a parameter dependent problem in primal variables. *Numerische Mathematik* 1999; **84**:97–119.
18. Benzi M, Olshanskii MA, Wang Z. Modified augmented Lagrangian preconditioners for the incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 2010.
19. Grasedyck L, Hackbusch W. Construction and arithmetics of $\mathscr{H}$-matrices. *Computing* 2003; **70**:295–334.
20. Hackbusch W. *Hierarchische Matrizen*. Springer: Berlin, 2009 (in German).
21. Grasedyck L, Kriemann R, Le Borne S. Domain decomposition based $\mathscr{H}$-LU preconditioning. *Numerische Mathematik* 2009; **112**:565–600.
22. Grasedyck L, Kriemann R, Le Borne S. Parallel black box $\mathscr{H}$-LU preconditioning for elliptic boundary value problems. *Computing and Visualization in Science* 2008; **4-6**:273–291.

DOI: 10.1002/fld

23. Bebendorf M. Why finite element discretizations can be factored by triangular hierarchical matrices. *SIAM Journal on Numerical Analysis* 2007; **45**:1472–1494.
24. Schreittmiller R. Zur Approximation der Lösungen elliptische Systeme partieller Differentialgleichungen mittels Finiter Elemente und $\mathscr{H}$-Matrizen. *Ph.D. Thesis*, Technische Universität München, 2005.
25. Le Borne S. Preconditioned nullspace method for the two-dimensional Oseen problem. *SIAM Journal on Scientific Computing* 2009; **31**:2492–2509.
26. Le Borne S, Ngufor C. An implicit approximate inverse preconditioner for saddle point problems. *ETNA* 2010; **37**:173–188.
27. Hackbusch W. A sparse matrix arithmetic based on $\mathscr{H}$ matrices. Part I: Introduction to $\mathscr{H}$-matrices. *Computing* 1999; **62**:89–108.
28. Börm S, Grasedyck L, Hackbusch W. Hierarchical matrices. Lecture Notes No. 21, Max-Planck-Institute for Mathematics in the Sciences, Leipzig, Germany, 2003. Available from: www.mis.mpg.de/preprints/ln/ [Revised June 2006].
29. Bebendorf M. *Hierarchical Matrices. A Means to Efficiently Solve Elliptic Boundary Value Problems*, Lecture Notes in Computational Science and Engineering, vol. 63. Springer: Berlin, 2008.
30. Grasedyck L, Le Borne S. $\mathscr{H}$-matrix preconditioners in convection-dominated problems. *SIAM Journal on Mathematical Analysis* 2006; **27**:1172–1183.
31. Ibragimov I, Rjasanow S, Straube K. Hierarchical Cholesky decomposition of sparse matrices arising from curl-curl-equations. *Journal of Numerical Mathematics* 2007; **15**:31–58.
32. Grasedyck L, Hackbusch W, Kriemann R. Performance of $\mathscr{H}$-LU preconditioning for sparse matrices. *Computational Methods in Applied Mathematics* 2008; **8**:336–349.
33. Roos HG, Stynes M, Tobiska L. Numerical methods for singularly perturbed differential equations: convection diffusion and flow problems. *Computational Mathematics*, vol. 24. Springer: Berlin, 1996.
34. Benzi M, Wang Z. Analysis of augmented Lagrangian-based preconditioners for the steady incompressible Navier–Stokes equations. *Technical Report TR-2010-011*, Mathematics and Computer Science Department, Emory University, 2010.

DOI: 10.1002/fld