INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2011; **65**:1180–1200
Published online 11 February 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/fld.2235

On iterative methods for the incompressible Stokes problem

M. ur Rehman[1,*,†], T. Geenen[2], C. Vuik[1], G. Segal[1] and S. P. MacLachlan[3]

[1]*Delft Institute of Applied Mathematics, Delft University of Technology, Faculty EEMCS, Mekelweg 4, 2628 CD, Delft, The Netherlands*
[2]*Institute of Earth Sciences, Utrecht University, Budapestlaan 4, 3584 CD, Utrecht, The Netherlands*
[3]*Department of Mathematics, Tufts University, Bromfield-Pearson Building, 503 Boston Avenue, Medford, MA 02155, U.S.A.*

SUMMARY

In this paper, we discuss various techniques for solving the system of linear equations that arise from the discretization of the incompressible Stokes equations by the finite-element method. The proposed solution methods, based on a suitable approximation of the Schur-complement matrix, are shown to be very effective for a variety of problems. In this paper, we discuss three types of iterative methods. Two of these approaches use the pressure mass matrix as preconditioner (or an approximation) to the Schur complement, whereas the third uses an approximation based on the ideas of least-squares commutators (LSC). We observe that the approximation based on the pressure mass matrix gives h-independent convergence, for both constant and variable viscosity. Copyright © 2010 John Wiley & Sons, Ltd.

Received 26 May 2009; Revised 23 September 2009; Accepted 17 October 2009

KEY WORDS: Stokes equations; block preconditioners; Krylov methods; Schur complement; variable viscosity; pressure mass matrix

1. INTRODUCTION

In this paper, we are concerned with the solution of the incompressible Stokes problem. The incompressible Stokes equations, given as

$$-\nu\nabla^2\mathbf{u}+\nabla p=\mathbf{f} \quad \text{in} \quad \Omega, \tag{1}$$

$$\nabla.\mathbf{u}=0 \quad \text{in} \quad \Omega, \tag{2}$$

are used to simulate an incompressible viscous flow. Equation (1) is known as the momentum equation, and Equation (2) is the continuity or mass conservation equation. In these equations, ν is the viscosity (inversely proportional to the Reynolds number), $\mathbf{u}$ is the velocity vector, and p is the pressure.

In order to have a unique solution, it is necessary to prescribe boundary conditions. Common types of boundary conditions are prescribed velocities and prescribed stresses, or combinations of these two.

*Correspondence to: M. ur Rehman, Delft Institute of Applied Mathematics, Delft University of Technology, Faculty EEMCS, Mekelweg 4, 2628 CD, Delft, The Netherlands.
†E-mail: M.urRehman@tudelft.nl

Contract/grant sponsor: Higher Education Commission of Pakistan and its Dutch handling agent Nuffic at the University of Technology in Delft; contract/grant number: 2-17/PM-OS/Neth/2005
Contract/grant sponsor: European Community's Sixth Framework Program; contract/grant number: MIF1-CT-2006-021927

The finite-element discretization of (1) and (2) with stable finite elements such as Taylor–Hood or Crouzeix–Raviart gives rise to a linear system of equations of the form:

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

where F corresponds to the viscous term of the PDE, B^{T} to the gradient operator, and B to minus the divergence operator. We define $\mathscr{A}$ to be the complete system matrix, n to be the number of velocity unknowns, and m to be the number of pressure unknowns with $n \geqslant m$. The system is symmetric but indefinite.

Discretization of the Stokes equations results in a so-called saddle-point problem. Saddle-point problems appear not only in fluid dynamics but also in elasticity problems and several other fields. As such, an iterative method that is developed for one type of saddle-point problem can often be applied in other areas as well [1]. Many iterative methods have been proposed for these problems, see for example [2–7]. In recent years, much of this work has focused on the development of preconditioners for the incompressible Navier–Stokes equations. As Stokes can be seen as a special case of Navier–Stokes, these preconditioners are also directly applicable to the Stokes problem. In our case, we concentrate only on the Stokes problem. In contrast to many of the Navier–Stokes preconditioners, we neither assume that the viscosity, ν, is constant, nor that it is smaller than one.

Our main goal is to investigate block preconditioners that are suitable for the Stokes problem, both for constant and varying viscosity. In general, these block preconditioners are based on splitting the equations into a velocity and a pressure part that are solved separately. Convergence of these schemes strongly depends on the approximation of the Schur complement operator, which arises from the $\mathscr{L}\mathscr{D}\mathscr{U}$ decomposition of the coefficient matrix,

$$\mathscr{A} = \mathscr{L}\mathscr{D}\mathscr{U} = \begin{bmatrix} F & B^{\mathrm{T}} \\ B & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ BF^{-1} & I \end{bmatrix} \begin{bmatrix} F & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I & F^{-1}B^{\mathrm{T}} \\ 0 & I \end{bmatrix}, \tag{4}$$

where $S = -BF^{-1}B^{\mathrm{T}}$ is the Schur-complement matrix. Most preconditioners are based on a combination of this block factorization with a suitable approximation of the Schur-complement matrix. Possible choices based on combinations of these blocks are:

- $\mathscr{D}$: with $\tilde{S} = M_p$, where M_p is the pressure mass matrix [8]. This block-diagonal preconditioner and several variants are suggested in [2] for the Stokes problem. The block-diagonal preconditioner is symmetric and positive definite and can be used as a preconditioner for MINRES.
- $\mathscr{D}\mathscr{U}$: Block-triangular preconditioner is employed with various approximations of the Schur complement. This preconditioner is no longer symmetric. For the (Navier–)Stokes problem, block-triangular preconditioners are used with various approximations of the Schur-complement matrix [1, 2, 5, 9–11]. In Section 2, we will use this block-triangular preconditioner with an approximation of the Schur-complement matrix based on the pressure mass matrix. In this case, we use GCR [12, 13] as the Krylov method to be preconditioned. With this approach, we observe h-independent convergence of the iterative scheme. Furthermore, in terms of number of iterations, the block-triangular preconditioned Krylov method shows approximately two times faster convergence than the block-diagonal preconditioner [14], [2, p. 345]. We will also discuss the least-squares commutator (LSC) approach to building block-triangular preconditioners, using both the diagonal of the velocity mass matrix (as discussed in [10]) and the diagonal of the velocity stiffness matrix (as considered for Q1-P0 elements in [15]). However, in this paper, we only use stable elements that satisfy the LBB condition.
- $\mathscr{L}\mathscr{D}\mathscr{U}$: With $S = -BD^{-1}B^{\mathrm{T}}$, where D is the diagonal of the (convection–)diffusion matrix, and where the F^{-1} terms in $\mathscr{L}$ and $\mathscr{U}$ are replaced by D^{-1}. This approach is known as the SIMPLE preconditioner in [4, 16]. In [17], D is replaced by an incomplete LL^{T} decomposition of F.
- $\mathscr{L}\mathscr{D}\mathscr{U}$ methods: In addition to using the $\mathscr{L}\mathscr{D}\mathscr{U}$ decomposition as a preconditioner, this factorization can also be used directly as an iterative method to solve the Stokes problem

DOI: 10.1002/fld

in combination with an approximation of the Schur complement [15, 18, 19]. In this case, preconditioned Krylov methods are employed only on the subsystems. The cost of these iterative methods is governed by the approximation used for the Schur-complement matrix and the preconditioner employed for the Schur subsystem. In Section 2.3, we introduce a Schur method that is also based on the $\mathscr{LDU}$ blocks. We handle terms involving the Schur-complement matrix implicitly and use the pressure mass matrix as a preconditioner to solve the subsystem corresponding to the Schur complement inexactly.

As mentioned earlier, we consider the solution of the Stokes problem with both constant and non-constant viscosity. Many of the preconditioners that have been developed to solve the Stokes problem are intended for the case of a constant-viscosity (Navier–)Stokes problem. When using the pressure mass matrix as an approximation of the Schur complement, scaling by the inverse of the viscosity is an important issue. If this scaling is not used, the pressure mass matrix acts more or less as a poorly scaled identity matrix. In Section 3, we will address this scaling in relation to the construction of the pressure mass matrix. Moreover, some aspects of convergence with and without scaling the problem/subproblems are discussed.

Numerical experiments are performed in Section 4, using Taylor Hood (Q2-Q1) elements, in which the velocity is defined by a bi-quadratic field and the pressure as bi-linear field on each element. The preconditioners are tested on problems from three different areas (fluid dynamics, extrusion, computational geodynamics) with different configurations (isoviscous, varying viscosity, sharp viscosity contrast). We compare the Schur method and the preconditioners based on the pressure mass matrix with the LSC preconditioner. The pressure mass matrix approximation shows nice convergence for quadratic elements. The Schur method seems expensive for the constant-viscosity problem; however, in problems with high-viscosity contrast, the Schur method and block-triangular preconditioner with the pressure mass matrix approximation perform equally well. Section 5 contains our conclusions.

2. SOLUTION TECHNIQUES

In this section, we discuss techniques for efficient solution of the Stokes problem. These techniques use Krylov subspace methods on both the system and subsystem level with appropriate choices of preconditioners and approximations of the pressure Schur-complement matrix. The three schemes we consider are:

1. Block-triangular preconditioner with GCR.
2. LSC block-triangular preconditioner with GCR.
3. The Schur method (a new approach).

The first two approaches are preconditioners that accelerate GCR to solve the Stokes problem. The third technique is an iterative method that uses Krylov methods on the subsystem level. The important feature of the third method is the preconditioner used to solve the implicitly constructed Schur-complement matrix $BF^{-1}B^{\mathrm{T}}$ (available in matrix–vector product form) at each step of the pressure subsystem solve.

2.1. Block-triangular preconditioner

The block-triangular preconditioner based on $\mathscr{DU}$ blocks is given in exact form as:

$$P_t = \begin{bmatrix} F & B^{\mathrm{T}} \\ 0 & S \end{bmatrix}. \tag{5}$$

If exact arithmetic is used, GMRES with this preconditioner converges into two iterations [20]. However, matrix–vector multiplications with S require solving a system with F leading to an expensive preconditioning operation. Therefore, the Schur-complement matrix is replaced by an approximation that is spectrally equivalent to the Schur-complement matrix and also cheap to

DOI: 10.1002/fld

compute. For the Stokes problem, the pressure mass matrix M_p is known to be a cheap and spectrally equivalent approximation to the Schur-complement matrix [2]. Therefore, the inexact form of the preconditioner can be written as

$$P_t = \begin{bmatrix} F & B^{\mathrm{T}} \\ 0 & -M_p \end{bmatrix}. \tag{6}$$

Solving $P_t z = r$, where $z = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$ and $r = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix}$, requires the following steps:

1. First solve $M_p z_2 = r_2$.
2. Update $r_1 = r_1 - B^{\mathrm{T}} z_2$.
3. Solve $F z_1 = r_1$.

The above steps can be performed with a lower accuracy than the accuracy required for the outer GCR iterations. In this paper, we use both exact (direct) and inexact solvers for these steps.

If the Stokes problem is scaled with a constant viscosity v, then the pressure mass matrix is also scaled with a constant viscosity $(1/v)M_p$. We will discuss scaling in detail in Section 3. The effectiveness of this approach is justified by the following results from [2, p. 270].

Theorem 2.1
For any flow problem with Dirichlet boundary conditions (prescribed velocities) discretized using a uniformly stable mixed approximation on a shape regular, quasi-uniform subdivision of $\mathbb{R}^2$, the pressure Schur-complement matrix $BF^{-1}B^{\mathrm{T}}$ is spectrally equivalent to the pressure mass matrix M_p, with

$$\mu^2 \leqslant \frac{\langle BF^{-1}B^{\mathrm{T}}\mathbf{q}, \mathbf{q}\rangle}{\langle M_p \mathbf{q}, \mathbf{q}\rangle} \leqslant 1 \quad \text{for all } \mathbf{q} \in \mathbb{R}^m, \quad \mathbf{q} \neq 0. \tag{7}$$

The inf–sup constant μ is bounded away from zero independently of h, and the condition number satisfies $k(BF^{-1}B^{\mathrm{T}}) \leqslant C/(c\mu^2)$, where C and c are the constants given by

$$ch^2 \leqslant \frac{\langle M_p \mathbf{q}, \mathbf{q}\rangle}{\langle \mathbf{q}, \mathbf{q}\rangle} \leqslant Ch^2 \quad \text{for all } \mathbf{q} \in \mathbb{R}^m, \quad \mathbf{q} \neq 0. \tag{8}$$

Similar bounds also exist for the Neumann boundary condition problem. An extra condition $\mathbf{q} \neq 1$ is required in (7) and (8) in case of enclosed flow, because the vector $\begin{pmatrix} \mathbf{0} \\ \mathbf{1} \end{pmatrix}$ corresponds to a zero eigenvalue of the Stokes matrix.

Theorem 2.2
If the Stokes problem is preconditioned with a block-triangular preconditioner, then the preconditioned system has eigenvalue $\lambda = 1$ of multiplicity n and the remaining eigenvalues depend on the approximation to the Schur-complement matrix.

In this paper, we accelerate GCR with a block-triangular preconditioner that uses the pressure mass matrix approximation for the Schur-complement matrix. For simplicity, we call this approach PMM.

2.2. LSC preconditioner

The pressure mass matrix is a good approximation of the Schur-complement matrix as long as the Reynolds number is small. For large Reynolds numbers, the effect of convection term in $BF^{-1}B^{\mathrm{T}}$ becomes important and hence convergence with the pressure mass matrix will become slow. An approximation that takes into account the effect of convection in the Navier–Stokes is the LSC preconditioner [10]. The Schur-complement approximation in LSC is based on the commutator action of a convection–diffusion operator on the velocity space ($\mathscr{L}$), multiplied by the gradient

DOI: 10.1002/fld

operator, and the gradient operator acting on the convection–diffusion operator in the pressure space ($\mathscr{L}_p$) when

$$\varepsilon = \mathscr{L}\nabla - \nabla\mathscr{L}_p, \tag{9}$$

is small, where

$$\mathscr{L} = -\nu\nabla^2 + \mathbf{w_h}.\nabla, \tag{10}$$

and $\mathbf{w_h}$ is the approximation to the discrete velocity, computed in the most recent Picard iteration. Note that the commutator would be zero if $\mathbf{w_h}$ is constant and the operators were defined on an unbounded domain [2].

Discretization and simplification (see the details given in [2]) give rise to an approximation of the form:

$$(BF^{-1}B^{\mathrm{T}})^{-1} \approx F_p(BM_1^{-1}B^{\mathrm{T}})^{-1}, \tag{11}$$

where

$$F_p = (BM_2^{-1}B^{\mathrm{T}})^{-1}(BM_2^{-1}FM_1^{-1}B^{\mathrm{T}}),$$

where M_1 and M_2 are scaling matrices.

So, applying the LSC preconditioner requires two scalar Poisson-type solves and one velocity solve. With $M_1 = M_2 = \hat{Q}_u$, where $\hat{Q}_u$ is the diagonal of the velocity mass matrix, the preconditioner shows nice convergence for solving the isoviscous problem.

In a recent paper [15], a scaling matrix has been constructed that improves the convergence considerably in the case of large viscosity contrasts. The scaling is based on the maximum row entry in the velocity matrix, $(M_1)_{ii} = (M_2)_{ii} = \max_j |F_{ij}|$. Based on this scaling, the LSC preconditioner is used to solve the Stokes problem with sharp viscosity contrasts discretized with Q1-P0 elements. Because these elements do not satisfy a discrete LBB condition [2], it may be necessary to modify the discretized equations by adding a stabilization term to the continuity equations. If such a modification is made, an appropriate adaptation of the approximate Schur complement is also needed. In this paper, we also use essentially the same scaling, by using the diagonal of the velocity matrix ($M_1 = M_2 = \mathrm{diag}(F)$). For simplicity, we call this approach LSC_D.

In the next section, we discuss a distinct solution technique than those discussed here and in Section 2.1. Usually, preconditioned Krylov methods are used to solve the Stokes problem, where the subsystems for the preconditioner are solved inaccurately with an iterative method. Therefore, the overall convergence depends on both the inner and outer iterative schemes. Next, we propose a method in which Krylov subspace methods are employed only at the subsystem level. We refer to this approach as the 'Schur method'.

2.3. The Schur method

The Schur method is based on the block factorization of Problem (3). The Schur-complement matrix, $(BF^{-1}B^{\mathrm{T}})$, present in the factorization is treated implicitly. In order to apply pressure-correction type methods, we split the coefficient matrix as follows:

$$\begin{bmatrix} F & B^{\mathrm{T}} \\ B & 0 \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} F & 0 \\ B & -BF^{-1}B^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} u^* \\ \delta p \end{bmatrix}, \tag{12}$$

where

$$\begin{bmatrix} u^* \\ \delta p \end{bmatrix} = \begin{bmatrix} I & F^{-1}B^{\mathrm{T}} \\ 0 & I \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix}. \tag{13}$$

Then the systems of equations can be solved in the following steps.

DOI: 10.1002/fld

Algorithm 1 The Schur method

Initialize $u^{(0)}$, $p^{(0)}$ and *maxiter* (maximum iterations)
Compute: $r_u = f - Fu^{(0)} - B^{\mathrm{T}} p^{(0)}$
$r_p = g - Bu^{(0)}$
For $k = 0$ to *maxiter*

1. Solve $Fu_f = r_u$
2. Solve $-BF^{-1}B^{\mathrm{T}} p_\delta = r_p - Bu_f$
3. Update $u_\delta = u_f - u_l$, where u_l is obtained by solving $Fu_l = B^{\mathrm{T}} p_\delta$
4. Update $u^{(k+1)} = u^{(k)} + u_\delta$
5. Update $p^{(k+1)} = p^{(k)} + p_\delta$
6. Update $r_u = f - Fu^{(k+1)} - B^{\mathrm{T}} p^{(k+1)}$
7. Update $r_p = g - Bu^{(k+1)}$
8. **If** converged **Exit**

End For

As already mentioned, the Stokes problem is symmetric and indefinite. However, the subsystems corresponding to the velocity (vector Poisson) and pressure used in the solves in Steps 1–3 are symmetric and definite. F^{-1} in Steps 1–3 is computed approximately by solving the velocity subsystem with an inexact solver. The best option is to use MG preconditioned CG or some multigrid technique as both of these methods are known to give optimal convergence for Poisson-type problems.

The pressure subsystem in Step 2 can in principle, be solved efficiently by CG. However, we do not construct $BF^{-1}B^{\mathrm{T}}$ explicitly, but approximately solve $(-BF^{-1}B^{\mathrm{T}})p_\delta = r_p - Bu_f$ within each step of CG. As F^{-1} is computed inexactly, CG can only be applied if the number of iterations used to do this is kept constant in each step. This is due to the fact that CG requires a constant matrix and preconditioner. This problem can be overcome by either using a stand-alone solver, such as multigrid, or a flexible Krylov methods (GCR in our case). The efficiency of the Schur method requires efficient treatment of Step 2. Because the Schur-complement matrix is not constructed explicitly, however, we need a special type of preconditioner as we cannot use preconditioners based on matrix splittings. The pressure mass matrix appears to be an efficient preconditioner for the Schur subsystem. If we use the same accuracy to solve the system with PMM and to solve Step 2 of the Schur method, the number of pressure mass matrix preconditioned GCR iterations in both methods are almost the same. These iterations govern the efficiency of both techniques. This observation also motivates the use of GCR instead of flexible CG [21].

Based on the Schur method (Algorithm 1), we propose two schemes:

1. *Schur method as direct method*: By requiring the accuracy of the subsystem solves to be the same or higher than the outer accuracy, the Schur method can be used as a direct solver. To solve the Schur pressure system $(-BF^{-1}B^{\mathrm{T}})p_\delta = r_p - Bu_f$, we use the pressure mass matrix M_p as a preconditioner. The system $(-BF^{-1}B^{\mathrm{T}})p_\delta = r_p - Bu_f$ is solved with the help of GCR in which preconditioned matrix–vector products within $(BF^{-1}B^{\mathrm{T}})p_\delta$ are obtained by computing preconditioned residual $S^{(k+1)} = Bu_m$, where u_m is obtained by solving a subsystem $Fu_m = B^{\mathrm{T}} {M_p}^{-1} r^{(k)}$ and $r^{(k)}$ is the residual computed in the previous GCR iteration.
2. *Schur as an iterative method*: When the inner systems are solved with a lower accuracy than desired of the final solution, outer iterations are required. Again, the pressure mass matrix is used as a preconditioner for the pressure subsystem.

From the above discussion, it is clear that three subsystems are solved for the velocity unknowns u_f, u_m, and u_l and one subsystem is solved for the pressure unknown p_δ in each iteration of the Schur method. The most expensive part of the algorithm is the computation of u_m as the number of times $Fu_m = r$ must be solved is equal to the total number of GCR iterations that are required to solve the pressure subsystem.

DOI: 10.1002/fld

One of the advantages that can be seen from the above algorithm is that most of the computations are done at the subsystem level. The system level computations can be reduced by a proper choice of inner accuracy. For example, if subsystems are solved as accurately as the outer tolerance requires, only one outer iteration is required. However, more outer iterations are typically required. This will be further discussed in Section 4, based on some numerical experiments.

2.4. Solving the subsystem

All of the iterative methods discussed here involve solving subsystems for the velocity and pressure. The most important issue in the pressure subsystem is its approximation as discussed above. Once it is approximated, it is easy to solve as it gives rise to a scalar problem on the pressure space that is typically easy to precondition. The velocity system, on the other hand, is large in size and requires a good preconditioner to get fast convergence.

In this paper, we apply various methods to solve these subsystems. As the velocity subsystem in the Stokes problem is symmetric and positive definite, we use AMG preconditioned CG (AMG/CG). The pressure subsystem is solved with AMG/CG when ICCG(0) becomes an expensive option. In case we are only interested in the number of outer iterations and accuracy we use Matlab in combination with a direct solver.

We use algebraic multigrid-(AMG) preconditioned CG from the library multi-level-(ML) preconditioning package [22]. The choice for multigrid is based on its optimal convergence for Poisson-type problems. Smooth aggregated multigrid, as is implemented in ML, is known to be particularly effective for the vector Poisson problems of interest here.

In general, multigrid consists of the following components: smoothing, restriction, prolongation, and a coarse-grid solver. The way they are linked to each other is shown in Algorithm 2.

Algorithm 2 Solve $A_h u_h = b_h$

where subscript h is used for the fine grid and H for the coarse grid.

1. Perform smoothing by using k iterations of an iterative method (Jacobi, Gauss Seidel, etc) on the problem $A_h u_h = b_h$
2. Compute the residual $r_h = b_h - A_h u_h$
3. Restrict the residual $r_H = P^{\mathrm{T}} r_h$
4. Solve for the coarse-grid correction, $A_H e_H = r_H$
5. Prolong and update $u_h = u_h + P e_H$
6. Perform smoothing by using l iterations of an iterative method (Jacobi, Gauss Seidel, etc) on the problem $A_h u_h = b_h$

All of these components play an important role in achieving optimal convergence. Algorithm 2 is also known as the two-grid algorithm; Step 4 can be adapted for various type of multigrid cycles (V, W, F) and choice of coarse levels. In geometric multigrid, restriction, prolongation, and coarse grids are chosen based on the geometric fine-grid information.

AMG also uses these components; however, the information that travels from finer grid levels to coarse grids is not only the grid-point locations where the variables are defined. To start the coarsening process, certain entries from matrix A_h are selected as influential in determining the solution. For example, if $a_{ij} \neq 0$ in A_h, we say that point i in the grid is connected to point j and vice-versa. The ith row of the matrix then consists of only those entries that influence the unknown u_i. The influence of unknown u_j to u_i is said to be large if a small change in u_j gives a large change in u_i [23]. The influence of one unknown on another is decided by the corresponding coefficient. A coupling between two grid points i and j is strong if

$$|a_{ij}| > \theta \sqrt{a_{ii} a_{jj}},$$

where θ is a predefined coupling parameter [24]. A coarse grid is then defined by aggregating the nodes in the graph of strong connections using a greedy algorithm [24]. Once the coarse grid has been chosen, all operators in the coarse-grid correction process, including the restriction

DOI: 10.1002/fld

and interpolation operators, are constructed based on information obtained from the coefficient matrix. Unlike MG, convergence of AMG does not require a robust smoothing strategy because the coarse-grid correction process is designed to complement simple smoothers. A piecewise constant interpolation operator $\hat{I}_H^h$ is defined that has positive non-zero entries of unity in positions determined so that its columns form a partition of unity over the aggregates. This tentative interpolation operator is then smoothed using Jacobi relaxation, given as

$$I_H^h = (I - \omega D_h^{-1} A_h^F) \hat{I}_H^h,$$

where ω is the relaxation parameter, $D_h = \mathrm{diag}(A_h^F)$ and A_h^F is the filtered matrix derived from A_h by adding all weak connections to the diagonal. The remainder of the multigrid components are formed based on the Galerkin condition [23], with restriction defined as $I_h^H = (I_H^h)^{\mathrm{T}}$ and $A_H = I_h^H A_h I_H^h$. This process is known as smoothed aggregation. AMG based on this interpolation technique shows nice convergence for problems with discontinuous coefficients and anisotropies [24, 25].

3. EFFECT OF VARIOUS TYPE OF SCALINGS

The standard pressure mass matrix is defined independently of the viscosity

$$(M_p)_{i,j} = \int_\Omega \phi_i \phi_j \, \mathrm{d}\Omega, \tag{14}$$

where ϕ_j and ϕ_i are the standard finite-element basis functions.

Experiments show that solving the variable-viscosity problem using this matrix as a preconditioner gives slow convergence. The Schur-complement matrix, however, contains F^{-1}, which means that it is proportional to the inverse of the viscosity. So, it makes sense to scale the pressure mass matrix used as a preconditioner by the inverse of the viscosity. In this section, we discuss the construction of the pressure mass matrix and scaling with the viscosity. We will consider two types of variable-viscosity problems: one with a relatively smooth viscosity (extrusion problem) and the other, a benchmark problem from geodynamics, with large viscosity jumps.

3.1. The pressure mass matrix

In case of constant viscosity, scaling of the pressure mass matrix is trivial. However, in case of variable viscosity, such scaling must be done carefully. Here, we assume that the viscosity is available at each point of the grid, and consider two alternatives to incorporate the scaling.

1. Explicit scaling of the pressure mass matrix, which implies pre- and post-multiplication of M_p by a diagonal matrix S_v : $M_{pe} = S_v^{-1} M_p S_v^{-1}$, where $S_v = \mathrm{diag}(\sqrt{v})$. This guarantees that the mass matrix remains symmetric. The evaluation of v is done either on grid points (Taylor–Hood elements) or on elements (Crouzeix–Raviart).
2. Implicitly scaling the pressure mass matrix can be done at the time of formation of the pressure mass matrix,

$$(M_{pi})_{i,j} = \int_\Omega (1/v) \phi_i \phi_j \, \mathrm{d}\Omega. \tag{15}$$

If the viscosity is defined to be piecewise constant on each element, we have multiple viscosity values to choose from at each node on the interface of two or more regions if we choose to scale M_p explicitly. Consider the grid consisting of two elements shown in Figure 1 with viscosity v_1 in element 1 and v_2 in element 2. We can see that nodes 3 and 4 of element 1 and nodes 1 and 2 of element 2 (using local numbering) are common to both elements. So, there are two different values

DOI: 10.1002/fld

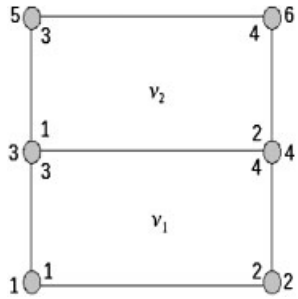

Figure 1. A grid with two elements.

of the viscosity that could be used at each node. Before constructing the global mass matrix, we define

$$\alpha_{ij}^{k} = \int_{e_k} \phi_i \phi_j \, \mathrm{d}A_k, \tag{16}$$

where α_{ij}^{k} represents the contribution to $(M_{pi})_{i,j}$ from element k where i and j represent node indices. The global pressure mass matrix from these two elements using implicit scaling is:

$$M_{pi} = \begin{bmatrix} \nu_1^{-1}\alpha_{11}^1 & \nu_1^{-1}\alpha_{12}^1 & \nu_1^{-1}\alpha_{13}^1 & \nu_1^{-1}\alpha_{14}^1 & 0 & 0 \\ \nu_1^{-1}\alpha_{21}^1 & \nu_1^{-1}\alpha_{22}^1 & \nu_1^{-1}\alpha_{23}^1 & \nu_1^{-1}\alpha_{24}^1 & 0 & 0 \\ \nu_1^{-1}\alpha_{31}^1 & \nu_1^{-1}\alpha_{32}^1 & \nu_1^{-1}\alpha_{33}^1 + \nu_2^{-1}\alpha_{11}^2 & \nu_1^{-1}\alpha_{34}^1 + \nu_2^{-1}\alpha_{12}^2 & \nu_2^{-1}\alpha_{13}^2 & \nu_2^{-1}\alpha_{14}^2 \\ \nu_1^{-1}\alpha_{41}^1 & \nu_1^{-1}\alpha_{42}^1 & \nu_1^{-1}\alpha_{43}^1 + \nu_2^{-1}\alpha_{21}^2 & \nu_1^{-1}\alpha_{44}^1 + \nu_2^{-1}\alpha_{22}^2 & \nu_2^{-1}\alpha_{23}^2 & \nu_2^{-1}\alpha_{24}^2 \\ 0 & 0 & \nu_2^{-1}\alpha_{31}^2 & \nu_2^{-1}\alpha_{32}^2 & \nu_2^{-1}\alpha_{33}^2 & \nu_2^{-1}\alpha_{34}^2 \\ 0 & 0 & \nu_2^{-1}\alpha_{41}^2 & \nu_2^{-1}\alpha_{42}^2 & \nu_2^{-1}\alpha_{43}^2 & \nu_2^{-1}\alpha_{44}^2 \end{bmatrix} \tag{17}$$

In (17), it is evident that for a high-viscosity contrast, the smaller value of ν will dominate the definition of M_{pi} (due to its inversion) at the nodes that are shared by these two elements, and large entries will be observed on the diagonal of the matrix.

3.2. Scaling for the variable-viscosity problem

We consider two variable-viscosity problems:

- *Extrusion problem with a variable viscosity*: The Stokes problem is solved in the two-dimensional (2D) domain shown on the left of Figure 2. The problem we consider is that of a round aluminum rod, which is heated and pressed through a die. In this way, a prescribed shape can be constructed. In this specific example, we consider the simple case of the construction of a small round rod. The viscosity model used describes the viscosity as function of shear stress and temperature, which are highest at the die where the aluminum is forced to flow into a much smaller region. The end rod is cut, which is modeled as a free surface. Boundary conditions are given by prescribed velocity at the inlet and a stress-free condition at the outlet. At the container surface (boundary of thick rod), we have a no-slip condition. At the die, we have friction, which is modeled as a slip condition along the tangential direction and a no-flow condition in the normal direction. The round boundary of the small rod satisfies free slip in the tangential direction and no-flow in the normal direction.
- *Geodynamic problem having a sharp viscosity contrast*: This problem is a benchmark problem known as the SINKER model in [15]. It models a geodynamic flow on a square region. Inside the region is a square with a different (but constant) viscosity, resulting in a sharp viscosity contrast. The configuration is shown in [15] and in Figure 2. Boundary conditions are no

DOI: 10.1002/fld

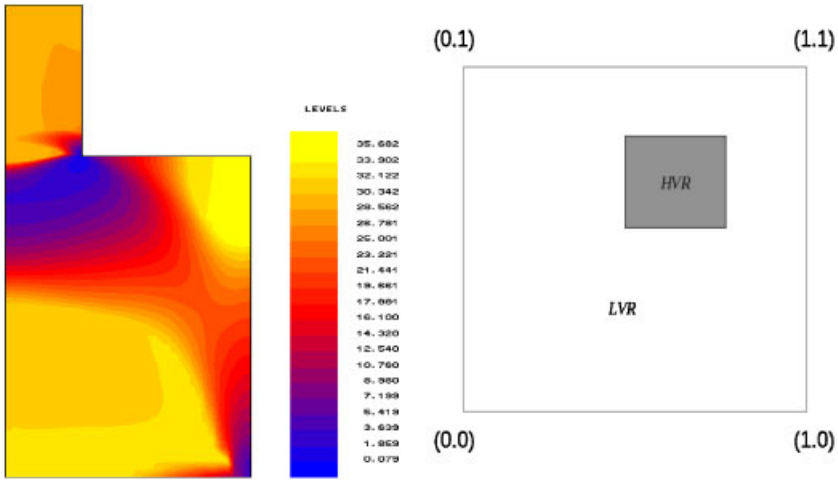

Figure 2. Two-dimensional domain for the variable-viscosity Stokes problem (left). At right, a 2D geodynamics test model: LVR represents the low-viscosity region with density $\rho_1 = 1$ and viscosity $v_1 = 1$, and HVR denotes the high-viscosity region with density $\rho_2 = 2$, and constant viscosity v_2 (1, 10^3, and 10^6).

normal flow and no shear stress (at all boundaries). For this boundary condition, pressure can be determined only up to an arbitrary additive constant.

If we try to solve the SINKER problem with high-viscosity contrast (e.g 10^6) and use an inexact solver, e.g ICCG(0), for the subsystem solves, we fail to get convergence due to stagnation of the inner solvers for high accuracies (10^{-6}). Each time, a suitable tolerance is determined for which the inner solver shows convergence. We would like to use higher accuracies to solve the subsystems because, for high-viscosity contrast problems, the number of iterations of the outer Krylov method depends on the inner accuracies. Another issue with high-viscosity contrast problems is that if we use convergence criteria based on the L_2 norm, some preconditioners e.g. PMM, lead to fewer iterations. However, an inaccurate solution is obtained with this convergence criteria. In order to achieve a suitably accurate results, we must ask for much higher accuracy from the Krylov method, which is not a good practice. For example, if we use direct solver for the high-viscosity contrast problem PMM requires much less iterations than LSC. However, the results with PMM are less accurate than those computed by LSC_D for the same tolerance. Figure 4 shows the pressure in the high-viscosity region of the SINKER model for PMM and LSC_D. The differences in these solutions can be easily seen. Note also that at other places the difference is not as visible. It has been observed that the solution obtained with LSC_D mimics that obtained with a direct solver.

Remark 3.1
If we use a preconditioner for the Schur-complement that involves the diagonal of the velocity matrix D^{-1}, the error in the iterative method using a direct method for the subsystems becomes small. This has been verified for LSC_D, $BD^{-1}B^{\mathrm{T}}$, and SIMPLE.

The above remark is true even if convergence for the velocity and pressure subsystems is achieved with an iterative method, without scaling.

To overcome issue with convergence of the subsystems and accuracy of the solution, we scale the original problem. In our case, we use a variant of scaling that has already been used with SIMPLE preconditioners [26]. S_m is a scaling matrix given by:

$$S_m = \begin{bmatrix} \sqrt{\mathrm{diag}(F)} & 0 \\ 0 & \sqrt{\mathrm{diag}(BD^{-1}B^{\mathrm{T}})} \end{bmatrix}. \tag{18}$$

DOI: 10.1002/fld

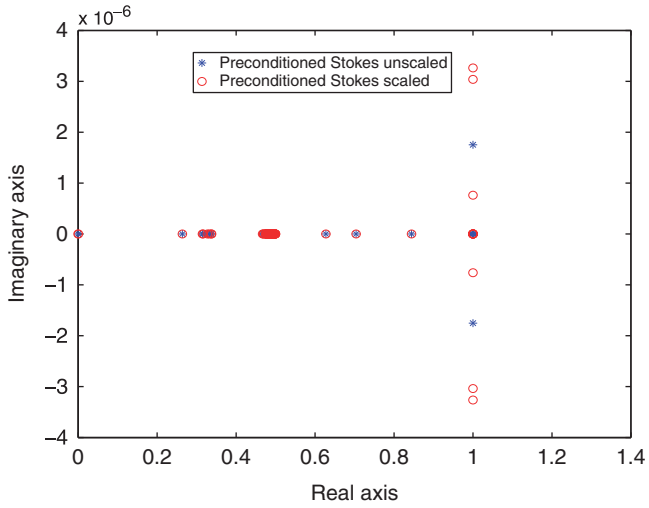

Figure 3. Eigenvalue spectrum of the Stokes problem.

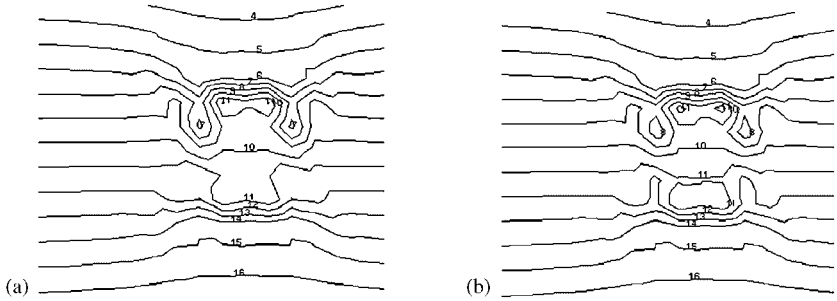

Figure 4. Solution of the variable-viscosity Stokes problem using various solution schemes: the plot shows the pressure solution in the high-viscosity region at the SINKER problem: (a) PMM and (b) LSC_D.

This scaling is applied to the complete system ($\mathscr{A}x=b$) and to the velocity and pressure subsystems (within the preconditioner), but is only well-defined when $F_{ii}>0$ and $(BD^{-1}B^{\mathrm{T}})_{ii}>0$. In our case, both conditions are satisfied and, hence, the matrix S_m is well-defined. With this scaling, convergence criteria are based on the residual in a scaled L_2 norm.

In our case, the linear system (for the SINKER model) is scaled before the iterative methods discussed here are employed. After scaling, the pressure obtained with PMM mimics the solution obtained with the exact solver. With this scaling, the number of iterations required for convergence by the preconditioned iterative method may increase. Scaling, however, only slightly changes the eigenvalue spectrum of the preconditioned system. This has been proven for the diffusion problem having extreme contrasts (up to 10^7) in the coefficients using ICCG [27]. For example, if we consider a preconditioned system, $M^{-1}Ax=M^{-1}b$, and if both the preconditioner and the coefficient matrix are scaled with the same matrix, D_s, the scaled matrices become $\hat{M}=D_s^{-1}M$ and $\hat{A}=D_s^{-1}A$, and the preconditioned system matrix after scaling is given by:

$\hat{M}^{-1}\hat{A}=M^{-1}D_sD_s^{-1}A=M^{-1}A$. So the spectrum of the preconditioned-scaled system is the same as the preconditioned-unscaled system.

Figure 3 shows the computed eigenvalue spectrum before and after scaling. The spectrum of the preconditioned system clearly remains almost unchanged. If we use scaling, so that our system, $\mathscr{A}x=b$ becomes $S_m^{-1}\mathscr{A}S_m^{-1}S_mx=S_m^{-1}b$, the most important change is the termination criterion for

DOI: 10.1002/fld

the iterative method. The relative convergence criteria for both the scaled and unscaled problems are as follows:

For the unscaled system

$$\|b-\mathscr{A}\hat{x}\|_2 \leqslant \|b\|_2 \times tol, \tag{19}$$

where $\hat{x}$ is an approximate solution and *tol* is the desired tolerance.

For the scaled system, the convergence criterion will be:

$$\|S_m^{-1}b - S_m^{-1}\mathscr{A}S_m^{-1}\hat{y}\|_2 \leqslant \|S_m^{-1}b\|_2 \times tol, \tag{20}$$

where $\hat{x} = S_m^{-1}\hat{y}$.

From the discussion above, it is clear that to get convergence (in high-viscosity contrast problems), scaling subsystems in the preconditioner is required. However, this does not guarantee that the computed solution will be accurate. For an accurate solution, the complete system also requires scaling with an appropriate scaling operator (S_m in our case).

3.3. Scaling of the velocity mass matrix

We use AMG/CG to solve the subsystems in the block preconditioners. One of the properties of the AMG method that we use is that it requires a constant number of unknowns per grid point. If the boundary conditions for the velocity are such that only a subset of the degrees of freedom is prescribed, we must define an approximation to the boundary that includes all velocity components. This is, for example, the case if the normal velocity component is prescribed in combination with the shear stress. The normal velocity is usually eliminated, leading to only 1(2D) or 2(3D) degrees of freedom on those points. To overcome this problem, we approximate the given normal velocity component by a mixed boundary condition of the form $c_n u_n + \sigma^{nt} = c_n \bar{u}$, where c_n a large number and $\bar{u}$ is the prescribed value of the normal velocity and σ^{nt} is the tangential component of the stress tensor.

Such an approximation works well except when the velocity mass matrix is used within the preconditioner, as in the Schur-complement approximation of the LSC preconditioner. In that case, the inverse of the diagonal of the velocity mass matrix is used as a scaling matrix. The combination of this preconditioner with the approximate boundary condition leads to a lack of convergence in the outer Krylov method. To overcome this difficulty, we update the velocity mass matrix by multiplying the entries corresponding to the boundary elements with the approximate boundary conditions by a factor of c_n. After this change LSC converges as quickly as it does in the case of full Dirichlet boundary conditions.

In our experiments, we observe no significant change in the number of outer/inner iterations required using either the approximate or exact boundary conditions (see Figure 5).

4. NUMERICAL EXPERIMENTS

Numerical experiments are performed for both constant- and variable-viscosity problems. The constant-viscosity problem is the 2D driven cavity problem. It simulates flow in a square cavity with enclosed boundary conditions and a lid moving from left to right, giving the boundary condition:

$$u_x = 1 \quad \text{at } y = 1; -1 \leqslant x \leqslant 1.$$

The variable-viscosity problems are the extrusion and SINKER problems discussed above.

In the tables that follow, we use the notation Schur(*eps*) to denote that all velocity and pressure subsystems are solved with an accuracy of $10^{-\text{eps}}$. The term 'iter.' gives the number of outer iterations required in the Schur method and preconditioned GCR, whereas 'inner' refers to the number of inner iterations needed to solve the Schur complement system in the Schur method. PMM stands for the block-triangular preconditioner that uses the pressure mass matrix as a preconditioner for the pressure part. The Stokes problem is solved up to relative accuracy of 10^{-6}. The iteration is stopped if the current iterate satisfies the inequality $\|r^k\|_2/\|b\|_2 \leqslant tol$, where r^k is the residual

DOI: 10.1002/fld

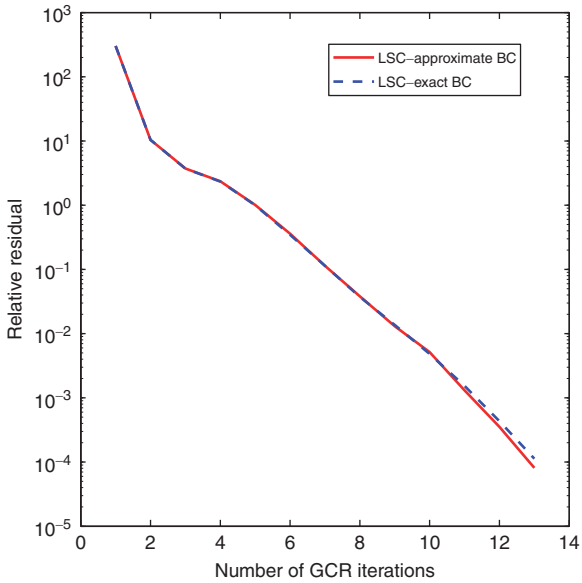

Figure 5. Convergence of LSC-preconditioned GCR, where the subsystems are solved with ICCG(0).

Table I. The Stokes driven cavity flow problem with Q2-Q1 discretization with AMG/CG for the velocity subsystem solves and ICCG(0) for the Schur subsystem solves. Solution accuracy is 10^{-6}.

	Grids			
Preconditioner	32×32	64×64	128×128	256×256
	iter.(time in seconds)			
PMM	11(1.4)	10(5.6)	9(23.6)	9(97)
LSC	10(1.38)	13(8.3)	17(54)	22(319)
LSC_D	22(3.2)	37(25)	80(275)	180(2880)
Schur(1)	6(3)	5(10.2)	5(46)	6(221)
Schur(6)	1(2)	1(10.6)	1(53)	1(251)

at the kth step of the Krylov subspace method, b is the right-hand side, and *tol* is the desired tolerance value.

4.1. Isoviscous problem

In this section, we consider the solution of the driven cavity Stokes problem with a constant viscosity. For PMM, LSC, and LSC_D, we use an accuracy of 10^{-2} for the velocity subsystem and 10^{-1} for the pressure subsystem. We start with two-level AMG for a 32×32 element problem and increment the number of levels when the grid size is doubled in each direction. In Table I, we report the number of outer iterations and CPU time required for the solution. The table shows that the PMM and the Schur method both scale well as the problem size increases. Scaling of PMM has also been observed for problems with up to 10^{8} degree of freedom [28]. Moreover, the CPU time consumed by each of these two approaches is less than that needed for LSC-type preconditioners. One iteration of the Schur method, however, is more expensive than the other preconditioned Krylov methods. In Figure 6, the number of iterations required for the pressure and velocity subsystems is plotted. The increase in the number of AMG/CG iterations required as the grid size (and, so, the number of levels) increases to solve the velocity subsystem is smaller for PMM and the Schur method than it is for the other methods. Figure 7 shows the number of AMG/CG iterations required per outer iteration for the velocity subsystem, for a number of block preconditioners. We see a small increase as the grid size increases, but this is not significant on the finer grids.

DOI: 10.1002/fld

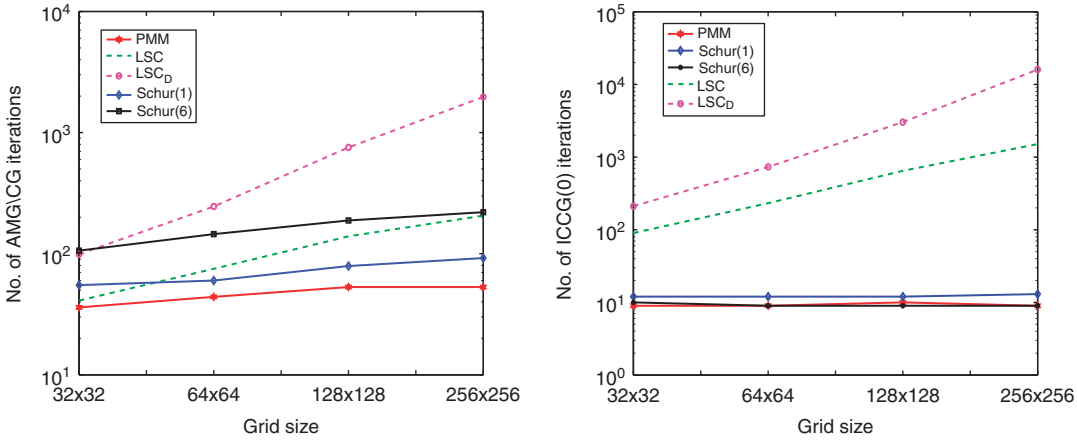

Figure 6. Solution of the constant-viscosity Stokes problem with accuracy 10^{-6}: at left, the total number of iterations required for the velocity subsystem. At right, the total number of iterations required for the pressure subsystem.

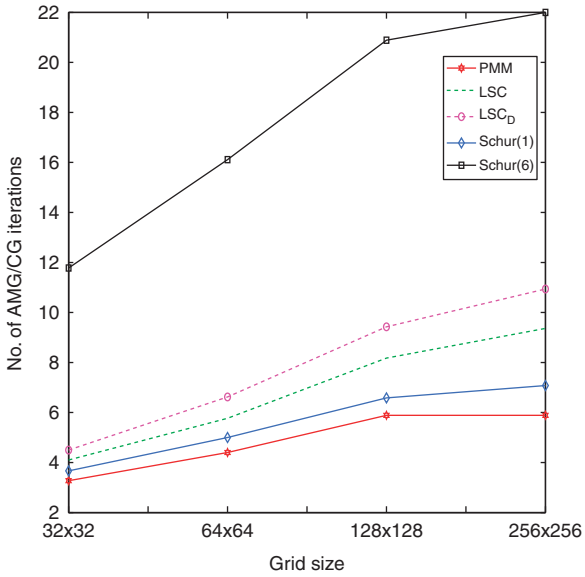

Figure 7. Number of AMG/CG iterations required to solve the velocity subsystem at each iteration of the iterative method.

If we use the Schur method as a direct method (Schur(6)), we require only one pressure step. However, as the pressure matrix requires inversion of the velocity matrix, we use an ICCG(0) for the global (outer GCR) iteration and AMG/CG to solve the velocity subsystems with high accuracy. The iterative approach with the Schur method (Schur(1)) requires more outer iterations, but the total number of velocity iterations is smaller than for the direct method, due to the lower accuracy. So, for finer grids, Schur(1) is slightly less time-consuming than Schur(6).

From Table I, we conclude that, of the methods compared, PMM shows the best convergence behavior for the Stokes driven cavity problem.

4.2. Extrusion problem with a variable viscosity

The next problem that we consider is the extrusion problem, which has a smooth variable viscosity. For this problem, we make a comparison between the Schur method and the various preconditioners.

DOI: 10.1002/fld

Table II. The variable-viscosity Stokes problem with Q2-Q1 discretization(SEPRAN) with AMG/CG for the velocity subsystem and ICCG(0) (PMM, Schur method) or AMG/CG (LSC, LSC_D) for the Schur subsystem. Solution accuracy is 10^{-6}.

Grid ↓ tol ⟶	Levels/unknowns-coarsest	PMM 10^{-3}	LSC 10^{-3}	LSC_D 10^{-6}	Schur 10^{-6}
		iter.(time in seconds)			
66*k*	$\frac{3}{394}$	19(51)	11(35)	74(357)	1(104)
195*k*	$\frac{4}{152}$	18(183)	13(188)	129(2650)	1(370)
390*k*	$\frac{5}{300}$	18(429)	14(480)	>1000	1(869)
595*k*	$\frac{5}{408}$	19(743)	15(871)	>1000	1(1478)
843*k*	$\frac{6}{112}$	19(1229)	15(1406)	>1000	1(2686)

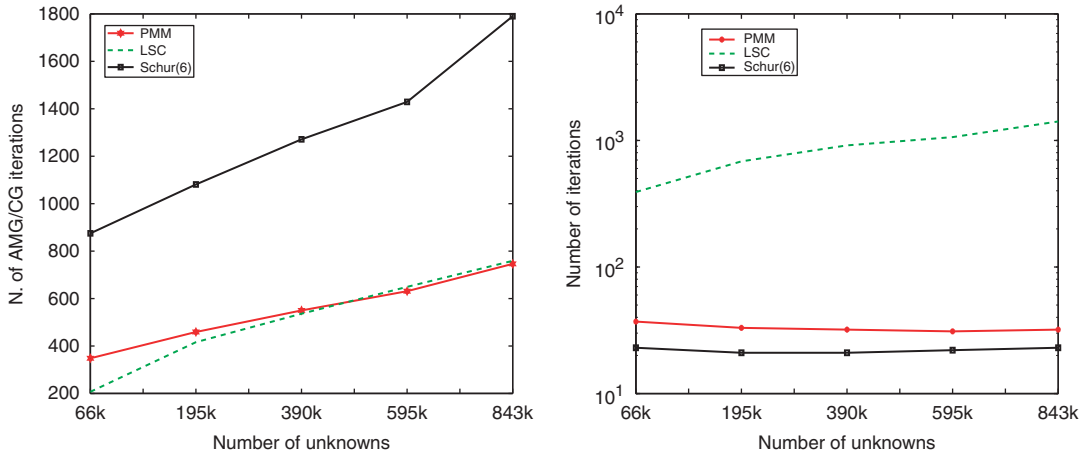

Figure 8. Solution of the variable-viscosity Stokes problem with accuracy 10^{-6}: at left, the total number of iterations required for the velocity subsystem. At right, the total number of iterations required for the pressure subsystem.

We use AMG/CG to solve the velocity subsystem in all iterative methods. In LSC and LSC_D only, AMG/CG is also employed for solving the pressure subsystems, as for PMM or the Schur method only a few iterations are required to solve the pressure subsystem with ICCG(0). In PMM and the Schur method, we use an approximation of the Schur complement matrix that has information about the variation in viscosity. LSC is different from the other preconditioners because the scaling used in LSC does not have information about the viscosity variation, as the diagonal of the velocity mass matrix is used for scaling the subfactors in the Schur complement matrix. In these experiments, we keep the tolerance required for the velocity and the pressure parts the same. Therefore, in Table II, only one tolerance is given. PMM and LSC perform best with inner accuracy 10^{-3}, whereas LSC_D and the Schur method converge well with tolerance 10^{-6}. Hence, we present results only for these accuracies.

PMM and LSC show better performance than the other two iterative methods. In terms of outer iterations, LSC requires fewer outer iterations than PMM, but one iteration of LSC is more expensive than one of PMM. Figure 8 reveals that LSC requires more pressure iterations than PMM due to the two Poisson solves required. The number of AMG/CG iterations taken by PMM and LSC for solving the velocity subsystem, are, however, comparable. Figure 9(a) shows that LSC requires more inner iterations for the velocity subsystem per outer iteration than PMM. PMM scales well with the problem size, requires fewer inner iterations for the velocity and pressure subsystems and, so, it is a better choice than LSC. The Schur method seems to be two times more expensive than PMM, because it uses an inner accuracy two times greater than PMM.

DOI: 10.1002/fld

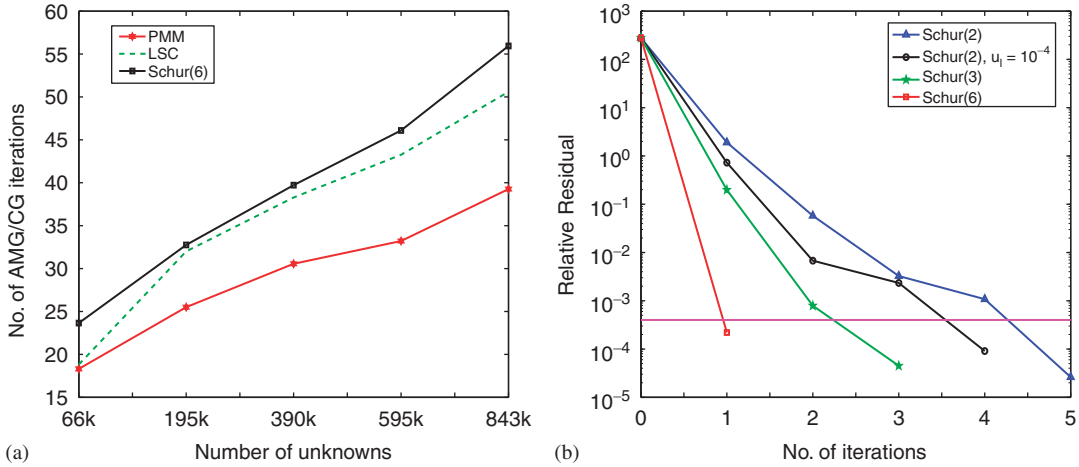

Figure 9. Performance of the algorithms for the extrusion problem: (a) number of AMG/CG iterations required to solve the velocity subsystem at each iteration of the iterative method and (b) number of outer iterations required for the solution of the Stokes problem with the Schur method.

Figure 9(b) shows convergence patterns for different subsystem accuracies in the Schur method. The horizontal line shows the tolerance for termination. The expected number of iterations for Schur(6), Schur(3), and Schur(2) are 1, 2 and 3, respectively. However, Figure 9(b) shows that for lower inner accuracies, the number of iterations is greater than expected. The main reason is that sometimes the accuracy of solutions of the subsystems does do not guarantee the satisfaction of the outer convergence criteria. In fact, in some cases, the number of outer iterations required for Schur(6) was greater than 1. The main reason for this is that the accuracy of the update $u_\delta = u_f - u_l$ computed in Step 3 of Algorithm 1 does not sufficiently reduce the overall error of the solution to the outer system, and the termination criterion is not satisfied. The same can be the case for Steps 5 and 6 in the Schur method. In those cases, we have increased the accuracy of the inner solver a little bit (multiplied by 10^{-1}), which makes the Schur method more expensive.

In the isoviscous problem, the Schur method is used either as a direct method or an iterative method. In terms of CPU time, both schemes perform well because the convergence of the subsystem solvers has a very small dependence on the inner accuracies. However, in the case of varying viscosity problems, especially those with high-viscosity contrast, the subsystems are required to be solved accurately. Therefore, in this case, a direct approach of the Schur method performs more efficiently than the iterative approach.

4.3. Geodynamic problem with a sharp viscosity contrast

The SINKER problem (Figure 10(a)) that we consider in this section has already been discussed in Section 3.1. Here, we consider a forcing term $f = (0, -\rho g)$, where $g = 9.8\,\mathrm{m/s^2}$. Figure 10 shows the three viscosity configurations that were used to check the convergence and accuracy of the iterative methods. The corresponding problems are referred to as (a), (b), (c). As LSC either diverges or converges very slowly for the high-viscosity contrasts considered here, we do not report any results for this method. The problem and subproblems are scaled as described in Section 3.2, before applying the iterative methods. The velocity solution for all preconditioners is accurate. Therefore, we only remark on the accuracy of the pressure solution. This is done by comparing with the 'exact solution', computed by a direct method.

Table III shows the number of iterations and the norm of the error in pressure for configuration (a). In the first experiments, v_1 is kept fixed at one and v_2 is increased; thereafter, v_2 is kept equal to one and v_1 is increased. We see that the number of iterations for PMM and Schur is almost the same for both the 30×30 and 60×60 grids, which suggest h-independent convergence. LSC_D

DOI: 10.1002/fld

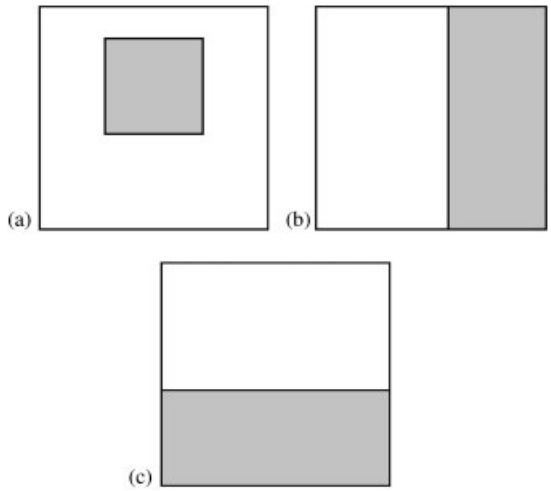

Figure 10. Geodynamic problem configurations where the dark region consists of viscosity v_2 and density ρ_2 and white region has viscosity v_1 and density ρ_1.

Table III. Iterative solution of the Stokes problem with configuration (a), accuracy $= 10^{-6}$.

	PMM		LSC_D		Schur	
v	iter.	$\|p_{\text{exact}} - p_{PMM}\|_2$	iter.	$\|p_{\text{exact}} - p_{LSC_D}\|_2$	iter. (inner)	$\|p_{\text{exact}} - p_{\text{Schur}}\|_2$
			30×30			
$v_1 = 1,\ v_2 = 10^6$	12	9×10^{-4}	26	7×10^{-6}	2(18)	2×10^{-8}
$v_1 = 1,\ v_2 = 10^3$	12	2×10^{-5}	26	3×10^{-6}	2(20)	2×10^{-10}
$v_1 = 1,\ v_2 = 10^1$	11	5×10^{-6}	24	1×10^{-6}	2(16)	2×10^{-10}
$v_1 = 1,\ v_2 = 1$	11	4×10^{-7}	25	2×10^{-6}	2(5)	5×10^{-11}
$v_1 = 10^1,\ v_2 = 1$	15	1×10^{-6}	27	2×10^{-6}	1(14)	2×10^{-6}
$v_1 = 10^3,\ v_2 = 1$	18	4×10^{-6}	26	3×10^{-6}	1(18)	2×10^{-6}
$v_1 = 10^6,\ v_2 = 1$	15	4×10^{-4}	23	1×10^{-4}	1(16)	2×10^{-5}
			60×60			
$v_1 = 1,\ v_2 = 10^6$	13	8×10^{-3}	40	6×10^{-5}	2(19)	5×10^{-8}
$v_1 = 1,\ v_2 = 10^3$	13	3×10^{-5}	40	5×10^{-6}	2(20)	3×10^{-9}
$v_1 = 1,\ v_2 = 10^1$	13	1×10^{-6}	41	3×10^{-6}	2(18)	4×10^{-10}
$v_1 = 1,\ v_2 = 1$	3	6×10^{-6}	36	3×10^{-6}	2(5)	9×10^{-10}
$v_1 = 10^1,\ v_2 = 1$	16	2×10^{-6}	41	4×10^{-6}	1(14)	4×10^{-6}
$v_1 = 10^3,\ v_2 = 1$	20	1×10^{-5}	38	7×10^{-6}	1(20)	5×10^{-6}
$v_1 = 10^6,\ v_2 = 1$	17	4×10^{-4}	35	1×10^{-4}	1(18)	3×10^{-5}

shows a clear increase with the increase of grid points. For constant v_2, the difference in accuracy between all methods is small.

However, in the problem where v_1 is constant (SINKER), the accuracy obtained with PMM is less than the other two iterative methods, even though all subsystems are solved with high accuracy (10^{-6} or 10^{-7}). The Schur method gives much more accurate results than the other two preconditioners, because the first iteration of the Schur method gives an accurate inner solve, whereas the second iteration makes the solution more accurate than the desired tolerance. From the table, we see that, with respect to accuracy and efficiency, the Schur method seems a better option than the other two. The reason is that PMM requires more iterations than Schur to get the same accuracy, whereas the costs per iteration are comparable. Similar results have been observed for a 90×90 grid.

DOI: 10.1002/fld

Table IV. Iterative solution of the Stokes problem with configuration (b), accuracy $= 10^{-6}$.

	PMM		LSC_D		Schur	
ν	iter.	$\|p_{\text{exact}} - p_{PMM}\|_2$	iter.	$\|p_{\text{exact}} - p_{LSC_D}\|_2$	iter.(inner)	$\|p_{\text{exact}} - p_{\text{Schur}}\|_2$
			$60 \times 60,\ \nu_1 = 1$			
$\nu_2 = 10^6$	11	2×10^{-4}	40	2×10^{-4}	2(15)	2×10^{-8}
$\nu_2 = 10^3$	11	1×10^{-5}	40	1×10^{-5}	2(18)	9×10^{-9}
$\nu_2 = 10^1$	11	2×10^{-6}	41	5×10^{-6}	2(16)	2×10^{-10}

Table V. Iterative solution of the Stokes problem with configuration (c), accuracy $= 10^{-6}$.

	PMM		LSC_D		Schur	
inner acc.	iter.	$\|p_{\text{exact}} - p_{PMM}\|_2$	iter.	$\|p_{\text{exact}} - p_{LSC_D}\|_2$	iter.(inner)	$\|p_{\text{exact}} - p_{\text{Schur}}\|_2$
			$\nu_1 = 1,\ \nu_2 = 10^6$			
10^{-3}	9	5×10^{-4}	39	4×10^{-4}	3(10)	5×10^{-6}
10^{-6}	9	7×10^{-4}	35	3×10^{-4}	2(10)	3×10^{-9}
10^{-8}	2	7×10^{-6}	33	2×10^{-4}	1(1)	7×10^{-6}
Direct	2	1×10^{-8}	31	2×10^{-4}	1(1)	9×10^{-9}
			$\nu_1 = 1,\ \nu_2 = 10^3$			
10^{-3}	11	1×10^{-5}	41	2×10^{-5}	3(11)	4×10^{-7}
10^{-6}	9	4×10^{-5}	36	8×10^{-6}	2(10)	3×10^{-10}
10^{-8}	2	8×10^{-7}	34	9×10^{-6}	1(1)	8×10^{-7}
Direct	2	2×10^{-8}	32	9×10^{-6}	1(1)	9×10^{-9}

Table IV gives the results for problem (b). In principle, all methods reach the same accuracy, but as the Schur method takes two outer iterations due to the fixed inner accuracy of 10^{-6}, its final accuracy is better.

Results for problem (c) with a 60×60 grid are given in Table V. We consider the convergence as a function of the inner accuracy for two values of ν_2 (10^3 and 10^6). As we already know, the number of outer iterations in these methods depends on the choice of the inner accuracy. One thing that is different from problem (a) is that, in PMM and the Schur method, inner accuracy also affects the accuracy of the solution of the complete system, whereas in LSC_D, the system accuracy remains the same with these changes in the subsystem accuracy. The reason is that problem (c) is relatively simple, because the heavy layer is at the bottom and effect of this layer (buoyancy) is much less than that in problems (a) and (b). This can be seen from the isobars in Figure 11. The pressure is smooth for problem (c), but it contains wiggles for both problems (a) and (b). In this case, the preconditioner itself is the best approximation of the problem, so increasing the inner accuracy has a large effect on the number of outer iterations required and the outer accuracy achieved. In the Schur method, we see that the accuracy obtained for solves with subsystem accuracy 10^{-3} and 10^{-8} is approximately the same. The three iterations of the Schur method with inner accuracy 10^{-3} makes the overall solution have roughly the same accuracy as one iteration of the Schur method with inner tolerance 10^{-8}. In this type of problem, PMM seems to be the better option, as we can get an accurate solution by increasing subsystem accuracy.

From all of these experiments, it is clear that, to get more accurate results, the Schur method is a better option than PMM and LSC. This property can be efficiently utilized in the solution of problems of type (a). A reason that the Schur method gives more accurate results might be the

DOI: 10.1002/fld

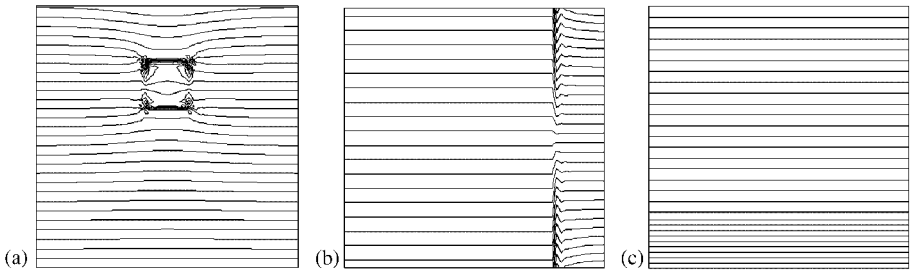

Figure 11. The pressure solution in various configurations: (a) problem (a), $\nu_1 = 1$, $\nu_2 = 10^6$; (b) problem (b), $\nu_1 = 1$, $\nu_2 = 10^6$; and (c) problem (c), $\nu_1 = 1$, $\nu_2 = 10^6$.

fact that it uses subfactors of the system iteratively in a classical way. If that is the case, we may also expect better results when using Richardson-type iterative improvement of the form:

$$x_{k+1} = x_k + PMM^{-1}(b - \mathscr{A}x_k). \tag{21}$$

We observe that (21) also gives better accuracy than PMM and LSC. As we need more iterations to apply Richardson, however, our conclusion regarding using the Schur method in problems of type (a) remains valid.

We know that for high-viscosity contrast problems, the condition number of the matrix increases with the contrast between the matrix entries. For a 10^8 contrast, the condition number of the matrix is expected to be of that same order or higher, even on small meshes. Solving the problem with a relative termination tolerance of 10^{-6} results in an error norm of roughly 10^2 (contrast*tolerance). Scaling does not change the condition number of the system (it only affects the termination criterion). So, in order to get the same order of accuracy for the non-scaled problem, we need a termination criterion that is more severe than the previous 10^{-6}. In this scenario, the use of Schur method can be a better option than other type of iterative methods.

5. CONCLUSIONS

In this paper, we consider the solution of the incompressible Stokes problem using preconditioned iterative methods. We solve three different classes of problems with various configurations and viscosity distributions using PMM, LSC, and the Schur method.

For the isoviscous problem, the PMM and the Schur method show better performance than the other preconditioners. LSC performs better for small problems. However, its dependence on the grid size makes it more expensive than the PMM and the Schur method for large problems. We note that for this type of problem based on a FEM discretization on an unstructured grid, the combination of PMM using AMG/CG to solve the subsystems leads to a number of inner and outer iterations that is essentially independent of h.

For the variable-viscosity problem arising from extrusion, the performance of PMM and the Schur method come close to one another due to the high-accuracy requirements for the subsystem solves in PMM. For the Schur method used as direct method, the accuracy of the inner subsystems must be kept equal to the outer accuracy. In general, this strategy works for isoviscous problems. However, in a variable-viscosity problem, the Schur method often needs a second iteration. This can be avoided by solving the last velocity subsystem with a higher accuracy. In some cases, all subsystems should be solved using a higher accuracy than the desired tolerance for the complete system. For a certain range of problems, LSC performs better than the PMM and the Schur method. However, due to its h-dependent convergence, its performance becomes equal to or worse than that of the PMM and the Schur method for large problems.

For a large viscosity contrast, LSC fails to converge. This suggests that for a high-viscosity contrast problem, the Schur complement matrix must be approximated by an operator that contains viscosity information for the problem. The complete system is also required to be scaled with

DOI: 10.1002/fld

proper scaling factors. The Schur method is a good choice to use in high-viscosity contrast problems because it gives a more accurate solution at lesser cost than PMM and LSC_D.

The general conclusion is that PMM is the best choice, except in case of a high-viscosity contrasts, where the Schur method is to be preferred.

ACKNOWLEDGEMENTS

The work of M. ur Rehman was supported by the Higher Education Commission of Pakistan and its Dutch handling agent Nuffic at the University of Technology in Delft, contract Ref: 2-17/PM-OS/Neth/2005.

The work of S. P. MacLachlan was partly supported by the European Community's Sixth Framework Program, through a Marie Curie International Incoming Fellowship, MIF1-CT-2006-021927.

REFERENCES

1. Benzi M, Golub GH, Liesen J. Numerical solution of saddle-point problems. *Acta Numerica* 2005; **14**:1–137.
2. Elman HC, Silvester D, Wathen AJ. *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluids Dynamics*. Oxford University Press: Oxford, 2005.
3. Benzi M. Preconditioning techniques for large linear systems: a survey. *Journal of Computational Physics* 2002; **182**(2):418–477.
4. Vuik C, Saghir A, Boerstoel GP. The krylov accelerated SIMPLE(R) method for flow problems in industrial furnaces. *International Journal for Numerical Methods in Fluids* 2000; **33**(7):1027–1040.
5. Benzi M, Olshanskii MA. An augmented Lagrangian-based approach to the Oseen problem. *SIAM Journal on Scientific Computing* 2006; **28**(6):2095–2113.
6. Wesseling P. *Principles of Computation Fluid Dynamics*. Springer Series in Computational Mathematics, vol. 29. Springer: Heidelberg, 2001.
7. Olshanskii MA, Vassilevski YV. Pressure schur complement preconditioners for the discrete Oseen problem. *SIAM Journal on Scientific Computing* 2007; **29**(6):2686–2704.
8. Verfürth R. A combined conjugate gradient-multigrid algorithm for the numerical solution of the Stokes problem. *IMA Journal of Numerical Analysis* 1984; **4**:441–455.
9. Rusten T, Winther R. A preconditioned iterative methods for saddle-point problems. *SIAM Journal on Numerical Analysis* 1992; **31**(5):1352–1367.
10. Elman H, Howle VE, Shadid J, Shuttleworth R, Tuminaro R. Block preconditioners based on approximate commutators. *SIAM Journal on Scientific Computing* 2006; **27**(5):1651–1668.
11. Kay D, Loghin D, Wathen A. A preconditioner for the steady-state Navier–Stokes equations. *SIAM Journal on Scientific Computing* 2002; **24**(1):237–256.
12. Eisenstat C, Elman HC, Schultz MH. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM Journal on Numerical Analysis* 1983; **20**(2):345–357.
13. van der Vorst HA, Vuik C. GMRESR: a family of nested GMRES methods. *Journal of Numerical Linear Algebra with Applications* 1994; **1**(4):369–386.
14. Elman H, Silvester D. Fast nonsymmetric iterations and preconditioning for Navier–Stokes equations. *SIAM Journal on Scientific Computing* 1996; **17**(1):33–46.
15. May DA, Moresi L. Preconditioned iterative methods for stokes flow problems arising in computational geodynamics. *Physics of the Earth and Planetary Interiors* 2008; **171**(1–4):33–47.
16. Elman H, Howle VE, Shadid J, Shuttleworth R, Tuminaro R. A taxonomy and comparison of parallel block multi-level preconditioners for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 2008; **227**(3):1790–1808.
17. Little L, Saad Y, Smoch L. Block LU preconditioners for symmetric and nonsymmetric saddle-point problems. *SIAM Journal on Scientific Computing* 2003; **25**(2):729–748.
18. Baggag A, Sameh A. A nested iterative scheme for indefinite linear systems in particulate flows. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(21–22):1923–1957.
19. Manguoglu M, Sameh A, Tezduyar T, Sathe S. A nested iterative scheme for computation of incompressible flows in long domains. *Computational Mechanics* 2008; **43**(1):73–80.
20. Murphy MF, Golub GH, Wathen AJ. A note on preconditioning for indefinite linear systems. *SIAM Journal on Scientific Computing* 2000; **21**(6):1969–1972.
21. Notay Y. Flexible conjugate gradients. *SIAM Journal on Scientific Computing* 2000; **22**(4):1444–1460.
22. Sala MC, Gee SM, Hu J, Tuminaro R. ML 5.0 Smoothed Aggregation User's Guide. *Sandia Report sand2006-2649 Edition*, 2006.
23. Briggs WL, Henson VE, McCormick SF. *A Multigrid Tutorial* (2nd edn). SIAM: Philadelphia, PA, 2000.
24. Vaněk P, Mandel J, Brezina M. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing* 1996; **56**(3):179–196.
25. Stüben K. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics* 2001; **128**(1–2): 281–309.

DOI: 10.1002/fld

26. Vuik C, Saghir A. The Krylov accelerated SIMPLE(R) method for incompressible flow. *Report 02-01*, Department of Applied Mathematical Analysis, Delft University of Technology, Delft, 2002.
27. Vuik C, Segal A, Meijerink JA, Wijma GT. The construction of projection vectors for a deflated ICCG method applied to problems with extreme contrasts in the coefficients. *Journal of Computational Physics* 2001; **172**: 426–450.
28. Geenen T, ur Rehman M, MacLachlan SP, Segal G, Vuik C, van den Berg AP, Spakman W. Scalable robust solvers for unstructured FE geodynamic modeling applications: solving the Stokes equation for models with large localized viscosity contrasts. *Geochemistry*, *Geophysics*, *Geosystems* 2009; **10**(9):1–12.

DOI: 10.1002/fld