

3.2. Krylov Solver. Having defined the discrete operators appearing in the Stokes system (2.1), we briefly discuss some issues that arise when solving this saddle-point problem using an iterative Krylov solver. The basic operation required the Krylov solver is the multiplication of a given vector \mathbf{x} with the matrix M . This consists of a straightforward direct evaluation of the appropriate finite-difference stencils at every interior face and every cell center in the computational grid. *inward*

Application of any of the preconditioners requires implementing approximate pressure (i.e., application of \tilde{L}_p^{-1}) and velocity (i.e., application of \tilde{A}^{-1}) solvers. Here we employ geometric multigrid techniques to implement these solvers. For the pressure solver, which estimates cell-centered pressure DOFs, we use standard variable-coefficient Poisson multigrid solvers [2]. For the velocity solver, which estimates face-centered velocity DOFs, we develop a vector variant of a standard scalar (pressure) Helmholtz solver based on a generalized red-black Gauss-Seidel smoother. The details of the multigrid algorithms are given in Appendix C, and the implementation is based on the Fortran version of the BoxLib library [34]. Note that because in Krylov methods GMRES solvers are applied to a residual correction system, zero is an appropriate initial guess for the multigrid subsolvers. Note also that the pressure subproblem has a null space of constant vectors (since only the gradient of the pressure matters) for periodic, slip and no-slip boundary conditions. Similarly, for periodic steady-state problems the velocity equation has a (d -dimensional) null-space of all constant velocity fields. In these cases some care is needed in the preconditioners to ensure that the null-space is handled consistently and the mean pressure and momentum are kept constant at certain prescribed values (e.g., zero).

Handling inhomogeneous boundary conditions, such as a prescribed non-zero tangential velocity along a given physical boundary, requires some care. When non-homogeneous velocity boundary conditions are specified, the viscous operator is ~~not a linear~~ operator because the viscous stencils near the boundary use ~~some of~~ the specified boundary values. In this case the viscous operator L_μ is an affine operator *affine, an affine operator.*

$$L_\mu(\mathbf{u}) = L_\mu^{\text{hom}}\mathbf{u} + L_\mu(\mathbf{0}),$$

where \mathbf{u} denote the velocity unknowns (interior velocity degrees of freedom), and L_μ^{hom} is the corresponding linear operator when homogeneous boundary conditions are specified. Similar considerations apply to the discrete divergence operator $D(\mathbf{u}) = D^{\text{hom}}\mathbf{u} + D(\mathbf{0})$. The discrete gradient operator acts on pressure unknowns which are restricted to the interior of the domain for the staggered-grid discretization, and is therefore unaffected by the velocity boundary conditions, $G = -(D^{\text{hom}})^*$. Because Krylov solvers require a linear rather than an affine operator, we apply the Krylov solver to the homogeneous form of the Stokes problem by correcting the right hand side in a pre-processing step, *what is this restriction? u par satisfies B.C. solve for u-u part.*

$$\begin{pmatrix} \theta\rho - L_\mu^{\text{hom}} & G \\ -D^{\text{hom}} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{x}_u \\ \mathbf{x}_p \end{pmatrix} = \begin{pmatrix} \mathbf{b}_u + L_\mu(\mathbf{0}) \\ \mathbf{b}_p + D(\mathbf{0}) \end{pmatrix}.$$

Another issue that arises when solving saddle-point problems is ~~that of~~ scaling of the system. Numerical experiments indicate that solving (2.1) when the grid-spacing h or the viscosity μ if very different from unity can lead to inaccurate solutions despite apparent convergence of the linear solver. The velocity and the pressure degrees of freedom have different physical units, and may have very different magnitudes in practice. The physical units of the right hand sides \mathbf{b}_u and \mathbf{b}_p are also different. Krylov solvers require as input a natural dot product in residual space. In our implementation, we have used the weighted dot product *the*

$$(3.8) \quad \mathbf{b}^{(1)} \cdot \mathbf{b}^{(2)} = \mathbf{b}_u^{(1)} \cdot \mathbf{b}_u^{(2)} + w_p^2 (\mathbf{b}_p^{(1)} \cdot \mathbf{b}_p^{(2)}),$$

where w_p is a weighting factor for the pressure residual. Numerically we have found this weighting factor to have little influence on the speed of the convergence or the accuracy of the result, and we have found it to be of little help in handling “badly scaled” problems. *huh*

Instead, we consider re-scaling the velocity equations by some constant c and re-scaling the pressure unknowns by the same factor, to obtain the rescaled system *See*

$$(3.9) \quad \begin{pmatrix} cA & G \\ -D & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{x}_u \\ \mathbf{x}_p \end{pmatrix} = \begin{pmatrix} c\mathbf{b}_u \\ \mathbf{b}_p \end{pmatrix}.$$

This simply amounts to rescaling the viscosity and θ and right-hand side by a factor c at the beginning of the solve, and rescaling the result for the pressure by c^{-1} after the solve. Intuitively, a well-scaled Stokes system is one in which both velocity and “pressure” unknowns have elements with the same physical units and similar typical magnitude. In order not to lose precision when evaluating the left hand-side of the velocity equation, we would like the viscous $c\mathbf{A}\mathbf{x}_u$ and pressure contributions $\mathbf{G}\mathbf{x}_p$ to have similar magnitude. This suggests choosing $c\mu_0 h^{-2} \sim h^{-1}$, giving $c \sim h/\mu_0$, where μ_0 is the typical magnitude of viscosity. Note that for $c \sim h/\mu_0$ the physical units of the rescaled pressure $c\mathbf{x}_p$ are indeed the same as the units of the velocity \mathbf{x}_u . Numerical experiments confirm that rescaling the viscosity from typical value μ_0 to $c\mu_0 \sim h$ can dramatically improve the scaling. Note that no rescaling of the divergence constraint is necessary since $\mathbf{D}\mathbf{x}_u$ has magnitude $\sim h^{-1}$ as the rest of the terms. Similarly, using equal weighting for velocity and pressure residuals, $w_p = 1$ in (3.8), will not pose any problems because the two components of the residual $c\mathbf{b}_u$ and \mathbf{b}_p already have the same units and similar magnitude.

The simple scaling correction (3.9) with $c \sim h/\mu_0$ is easy to apply as a pre/post-processing step and requires no changes to the core algorithm. If there is a very broad range of viscosities present in the problem, a uniform rescaling of the equations will not be sufficient and diagonal scaling matrices should be used to rescale the velocity and pressure separately, see Eq. (31) in Ref. [16] for a specific formulation. **[Donev: The scaling I work out here is similar but not exactly the same as the one used in (31) in Ref. [16] even for constant viscosity (the scaling of pressure is the same but they also scale velocity). I don't quite understand which one is better and why, nor do I understand their argument for the scaling they use – they actually attribute their inspiration to [35].]** To avoid loss of accuracy, in extreme cases extended precision arithmetic may need to be used in the solver [16].

[Donev: If we want to cut length much of this para can go, I just wanted to add it for completeness.] A final issue that we discuss is the choice of the convergence criterion for the iterative solver for the system (2.1). Here we employ left preconditioning and apply the iterative solver to the preconditioned system $\mathbf{P}^{-1}\mathbf{M}\mathbf{x} = \mathbf{P}^{-1}\mathbf{b}$. The convergence criterion is therefore most naturally expressed in terms of either the absolute or relative reduction in the magnitude of the preconditioned residual $r_P = \|\mathbf{P}^{-1}(\mathbf{M}\mathbf{x} - \mathbf{b})\|_2$, where the L_2 norm is defined in terms of the weighted inner product (3.8). A more robust alternative is to base convergence criteria on the value of the unpreconditioned (true) residual $r = \|\mathbf{M}\mathbf{x} - \mathbf{b}\|_2$. Each residual also contains two components, a velocity residual and a pressure residual, which may converge at different rates. These issues make it difficult to construct a general robust convergence criterion for the Krylov solver. In our implementation, which employs the GMRES method, we terminate the inner (non-restarted) GMRES iteration whenever the GMRES estimate of the preconditioned residual falls below some tolerance, $r_P \leq \epsilon \min(\|\mathbf{P}^{-1}\mathbf{b}\|_2, r_P^0)$, where r_P^0 is the initial preconditioned residual. The actual (rather than estimated) preconditioned residual is verified and the true residual computed before terminating the outer (restart) GMRES iteration. For the problems studied here we observe all three residuals to exhibit similar convergence.

4. Results. In this section we perform detailed numerical experiments to determine the most robust and efficient preconditioner over a broad range of parameters. Because the preconditioned system is not necessarily symmetric, as a Krylov solver for the saddle-point system (2.1) we use the left-preconditioned GMRES (Generalized Minimal Residual) method with a fixed restart frequency m [37, 36]. GMRES requires the storage of m vectors like \mathbf{x} . For a d -dimensional regular grid with N cells, the memory storage requirement is thus at least $(d+1)mN$ floating-point numbers since there are d velocity degrees of freedom (DOFs) and one pressure DOF per grid cell. It is therefore important to explore the use of restarts to reduce the memory requirements of the Krylov solver. A more robust and flexible method is FGMRES (Flexible GMRES), in particular, FGMRES allows the use of preconditioners that are not necessarily constant linear operators (e.g., another Krylov solver). A notable downside of FGMRES is that it requires twice the storage of GMRES. In the numerical experiments reported in the next section we utilize dimensionless well-scaled values ($h = 1, \mu_0 = 1, \rho_0 = 1$) for all of the coefficients, so that no explicit rescaling of the unknowns or the equations is required.

The multigrid algorithms used in the pressure and velocity subsolvers iteratively apply V cycles,

thought we didn't use this.

what is norm here?

how validate scaling discussion above

each of which consists of successive hierarchical restriction (coarsening), smoothing, and prolongation [6]. In our tests, we will use a constant number n of V cycles in both the pressure and the velocity solvers. This ensures that the preconditioners are constant linear operators and allows for the use of the GMRES method. The velocity (vector) multigrid V cycle has a cost very similar to d independent pressure (scalar) V cycles. Therefore, as a proxy for the CPU cost of a single application of the preconditioner we will use the number of *scalar* multigrid cycles. The cost of the pressure subsolver (application of $\tilde{\mathbf{L}}_\rho^{-1}$) is n scalar V cycles, and the cost of the velocity subsolver (application of $\tilde{\mathbf{A}}^{-1}$) is dn scalar V cycles. All preconditioners require at least one velocity solve per application, however, they differ in whether they require a pressure Poisson solve. For unsteady flow ($\theta \neq 0$), all preconditioners require the application of $\tilde{\mathbf{S}}^{-1}$, which requires the application of $\tilde{\mathbf{L}}_\rho^{-1}$. For steady flow ($\theta = 0$), \mathbf{P}_2 , \mathbf{P}_3 and \mathbf{P}_4 do not require a pressure Poisson solve.

A fundamental “easy” test problem we employ is constant coefficient steady Stokes flow in a periodic domain or a domain with no-slip condition along all boundaries. As a more challenging variable-coefficient test problem we use a *bubble* test, in which we embed a sphere (disk in two dimensions) of one fluid in another fluid with different viscosity and density. The bubble is placed in the center of a cubic (square) domain of length L_c cells with no-slip boundaries along all domain boundaries. For the bubble problem, the viscous stress is taken to be $\boldsymbol{\tau}(\mathbf{u}) = \mu [\nabla \mathbf{u} + (\nabla \mathbf{u})^T]$; and the diagonal elements of the viscosity matrix $\boldsymbol{\mu}$ and the density matrix $\boldsymbol{\rho}$ at cell centers are generated from the spatially-dependent functions

$$(4.1) \quad \mu(\mathbf{x}) / \mu_0 = \frac{1}{2} (r_\mu + 1) + \frac{1}{2} (r_\mu - 1) \tanh \left(\frac{d(\mathbf{x}, \Gamma)}{\epsilon} \right) + 0.1\mathbf{R},$$

$$(4.2) \quad \rho(\mathbf{x}) / \rho_0 = \frac{1}{2} (r_\rho + 1) + \frac{1}{2} (r_\rho - 1) \tanh \left(\frac{d(\mathbf{x}, \Gamma)}{\epsilon} \right) + 0.1\mathbf{R},$$

respectively. Here r_μ and r_ρ are the viscosity and density contrast ratios, Γ is the interface, a circle of radius $L/4$ placed at the center of a cube with side of length $L = L_c h$, $d(\mathbf{x}, \Gamma)$ is the distance function to the interface, $\epsilon = h$ is a smoothing width used to avoid discontinuous jumps in the coefficients, and \mathbf{R} is a random number uniformly distributed in $(0, 1)$. Unless otherwise indicated, we use a relatively large contrast ratio to make the problem more challenging, $r_\mu = r_\rho = 100$.

4.1. Spectrum of Preconditioned Operator. Convergence analysis of the preconditioned GMRES method is not straightforward and there is no simple link to the spectrum of the eigenvalues. Nevertheless, standard wisdom says that having closely clustered eigenvalues of the preconditioned operator $\mathbf{P}^{-1}\mathbf{M}$ leads to faster convergence. Furthermore, the ratio of the largest to the smallest eigenvalue (excluding the trivial zero eigenvalue arising from the fact pressure is only determined up to a constant) should be bounded from above by a constant essentially independent of grid size, and, possibly, viscosity and density contrast ratio.

We focus on the steady-flow case $\theta = 0$ in two dimensions, for a square domain of L_c^2 cells with four no-slip boundaries. In order to quickly gain insight into the behavior of the spectrum, we consider using exact subdomain solvers, $\tilde{\mathbf{A}}^{-1} = \mathbf{A}^{-1}$ and $\tilde{\mathbf{L}}_\rho^{-1} = \mathbf{L}_\rho^{-1}$ instead of multigrid, relying on the fact that a well-designed multigrid cycle is (essentially) spectrally-equivalent to an exact solver (this is difficult to prove however).

An analysis of the spectrum of the preconditioned operators is given in Appendix B. As demonstrated in that appendix, $\mathbf{P}_1^{-1}\mathbf{M}$ and $\mathbf{P}_2^{-1}\mathbf{M}$ have an eigenvalue $\lambda = 1$ for each velocity DOF, and the remaining eigenvalues, corresponding to the pressure DOFs, are equal to the eigenvalues of $\mathbf{S}^{-1}\mathbf{S}$. For the constant coefficient case, we find that in two dimensions $\mathbf{S}^{-1}\mathbf{S}$ has all but $4(L_c - 1)$ eigenvalues equal to unity, i.e., there is a non-zero eigenvalue for each cell that touches a single no-slip boundary. The remainder of the nonzero eigenvalues are shown to be bounded between η^2 and 1, where the inf-sup constant η is independent of the grid resolution. For the variable-coefficient case, theoretical calculations are difficult and we turn to numerical experiments to access the impact of viscosity contrast on the spectrum of the eigenvalues.

non-unity

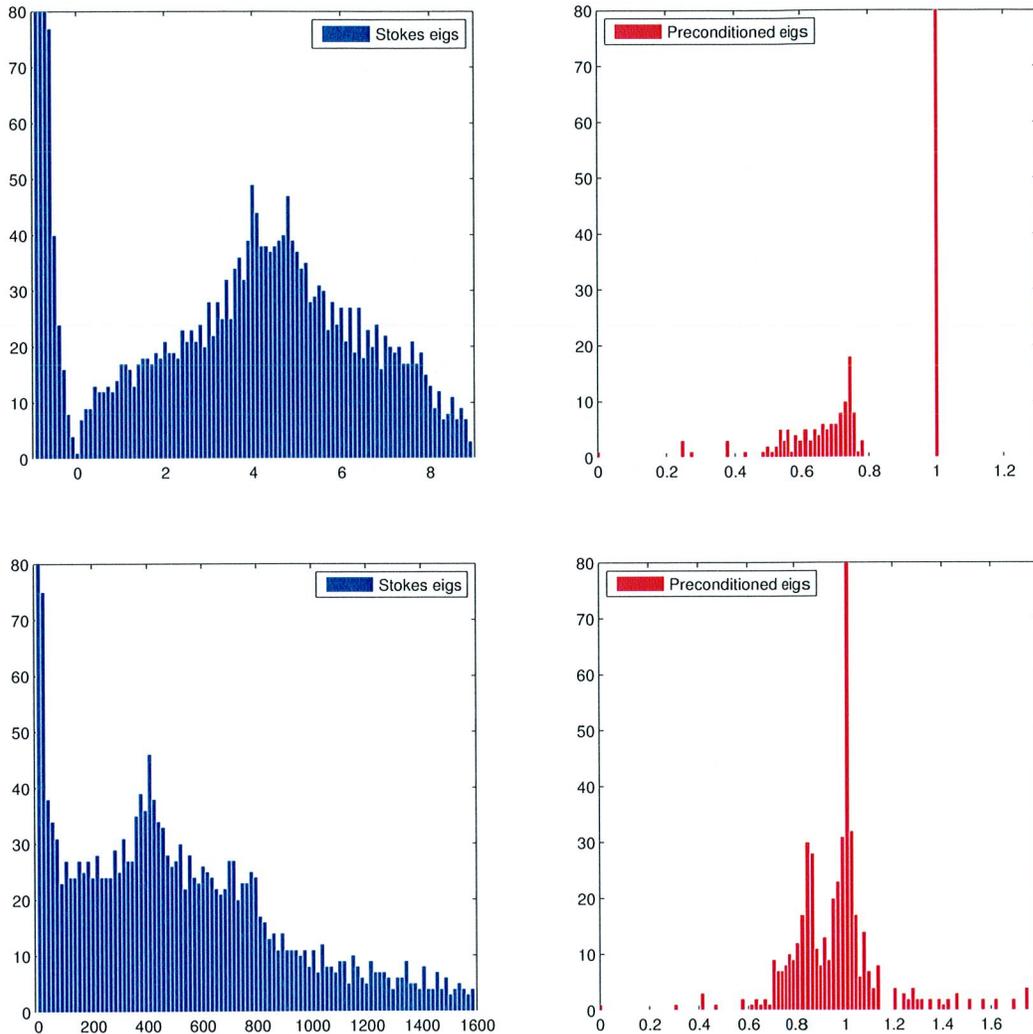


FIG. 4.1. Histogram of the eigenvalues of M (left panels) and $P^{-1}M$ (more precisely, $S^{-1}S$) (right panels) for a steady Stokes problem on a grid of 32×32 cells with four no-slip boundaries. The vertical axis gives the number of eigenvalues, truncated for the eigenvalue $\lambda = 1$ in the right panels due to the large number of unit eigenvalues. The case of constant viscosity is shown in the upper panels, and the case of variable viscosity with contrast ratio 100 is shown in the lower panels.

We explicitly form M in MATLAB as a sparse matrix, and use Cholesky factorization to factorize A and form $S^{-1}S$. We then ~~use dense linear algebra to~~ compute the eigenvalues (which are all real-valued) of M and $S^{-1}S$. This is only computationally feasible in two dimensions for relatively small grids. In Fig. 4.1, we show a histogram of the eigenvalues of the unpreconditioned and preconditioned operators for a square domain of length $L_c = 32$ cells with no-slip boundaries. In the upper row of panels in Fig. 4.1 we study the constant viscosity case. The total number of DOFs is $N_{\text{dof}} = L_c^2 + 2L_c(L_c - 1) = 3008$. Since the original Stokes system is of saddle point type, M has both positive eigenvalues and negative eigenvalues, and there are $L_c^2 = 1024$ eigenvalues that are smaller than or equal to zero. While the unpreconditioned spectrum shows a broad spectrum of eigenvalues with conditioning number that grows with the grid size, the preconditioned spectrum shows that most eigenvalues are unity, with the remaining $4(L_c - 1)/N_{\text{dof}} \approx 4\%$ nonzero eigenvalues clustered around 0.7.

In the lower row of panels in Fig. 4.1 we study the variable viscosity case for the bubble problem with viscosity contrast ratio $r_\mu = 100$. The unpreconditioned system is seen to be very badly conditioned, with a broad spectrum of eigenvalues. At the same time, the preconditioned operator is well-conditioned, with around 87% of the eigenvalues in the interval $(0.99, 1.01)$. While some eigenvalues are larger than unity in this case, the spread in the eigenvalues is not much different from the constant-coefficient case. This suggests that the spectrum remains localized around unity and bounded away from zero even for rather large contrast ratios. It may be possible to extend the finite-element theory developed in Refs. [21, 20] to prove that $\mathcal{S}^{-1}\mathcal{S}$ is spectrally-equivalent to the identity matrix for the staggered grid discretization we employ here.

4.2. Multigrid Subsolvers. Before comparing the different preconditioners, we optimize the key parameters in the multigrid pressure and velocity approximate subsolvers, specifically, the number of smoothing (relaxation) sweeps per V cycle and the number of V cycles per application of the preconditioner.

4.2.1. Number of smoothing sweeps. One of the key aspects of geometric multigrid is the smoother used to perform relaxation of the error at each level of the multigrid hierarchy. As explained in more detail in Appendix C, we employ a red-black Gauss-Seidel smoother. This ensures that all components of the error are damped to some extent for constant-coefficient problems. The optimal number of smoothing (relaxation) sweeps to be performed at each multigrid level (we use the same number of sweeps going down and up the multigrid hierarchy) has to be determined by numerical experimentation.

In Fig. 4.2 we show the convergence of the pressure (left panels) and velocity multigrid solvers (right panels) for constant viscosity but for the stress-tensor form of the viscous term (3.1). In the upper row we show results in two dimensions, and in the lower row we show results for three dimensions. Similar results are obtained for different types of boundary conditions. We see a large increase in the rate of convergence when increasing the number of smoothing sweeps from one to two, and only a modest increase thereafter. Since the cost of geometric multigrid is in large dominated by smoother, henceforth we do two applications of the smoother at each level of the multigrid hierarchy in each V cycle.

The speed of convergence of the plain multigrid iteration, which is a particular form of a (non-Krylov) iterative solver, is the standard against which one ought to measure convergence of the Krylov solver for the Stokes problem. As we can see in Fig. 4.2, each V cycle reduces the residual by an order of magnitude or more (this is the signature of “good” multigrid), so that only about a dozen V cycles are needed to reduce the residual to near roundoff. Therefore, a Stokes solver that uses only $10(d+1)$ scalar multigrid cycles to reduce the residual by more than 10 orders of magnitude should be considered excellent.

4.2.2. Number of multigrid cycles. For constant-coefficient Stokes problems with periodic boundaries, as explained in Appendix B, for both time dependent and steady state problems, if exact subdomain solvers are employed, GMRES converges in a single iteration with preconditioner P_1 and in two iterations with P_2 and P_3 . The same applies for any boundary condition for inviscid time-dependent problems. However, in the majority of cases of interest, multiple GMRES iterations will be required even if the subsolvers are exact. It is therefore important to explore the use of inexact pressure and velocity solvers. Specifically, it is important to determine the optimal number of multigrid V cycles per application of the preconditioner.

In the left panels of Fig. 4.3 we show the convergence of the relative preconditioned residual, as estimated by the GMRES algorithm, for steady Stokes problems in two and three dimensions, as a function of the total number of scalar V cycles. We recall that the number of V cycles is a good proxy for the total computational effort, so that the most rapid convergence in these plots corresponds to the most efficient solvers. In the top left panel we show results for constant viscosity but for the stress-tensor form of the viscous term (3.1) for a periodic system, and in the bottom left panel we show results for the variable-viscosity bubble problem described earlier. In the corresponding right panels we show the convergence of the pressure and velocity multigrid subsolvers on the same problem, to

not distinguishing

define a metric

which would

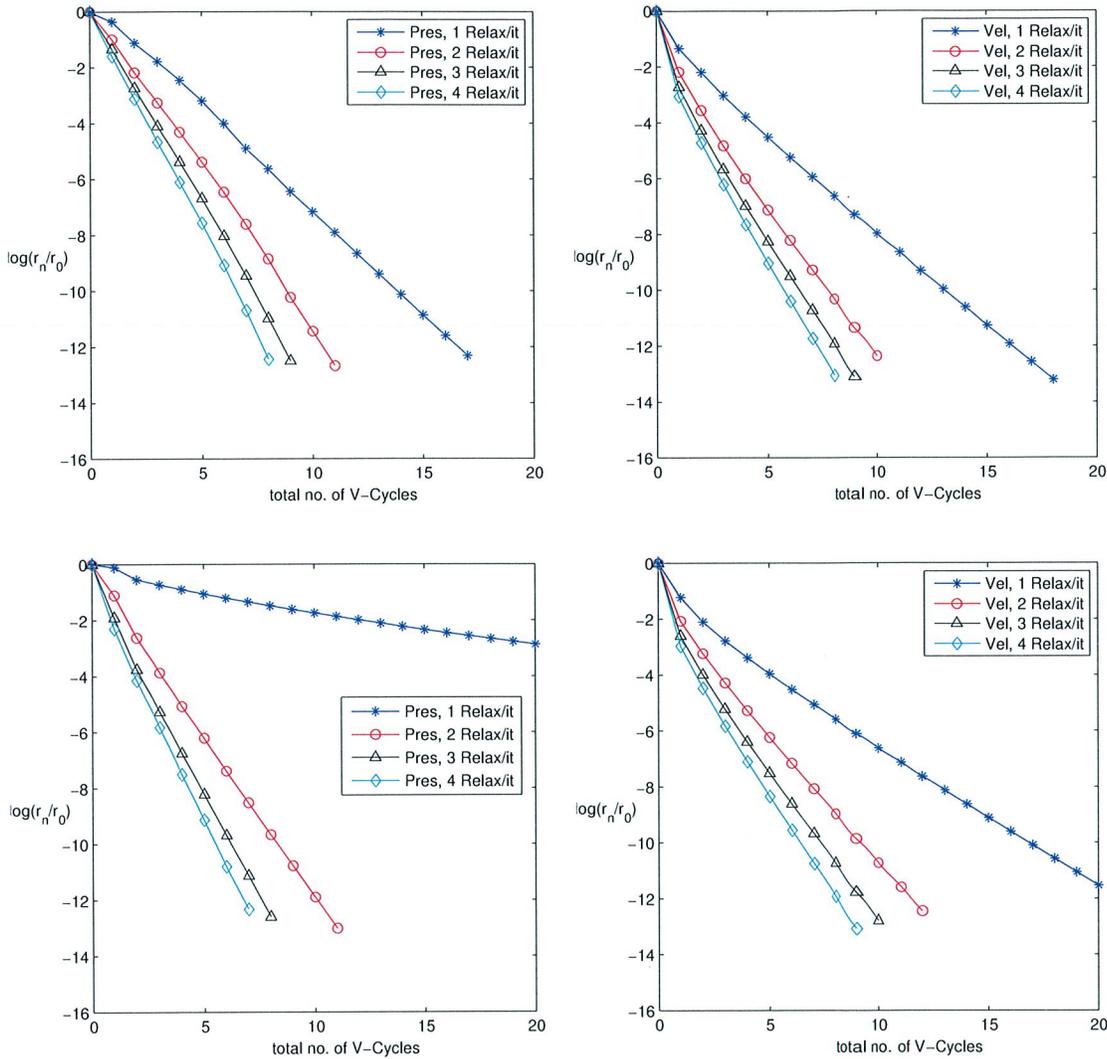


FIG. 4.2. The log of the relative residual for the pressure (left) and velocity (right) multigrid solvers as a function of the number of multigrid V cycles, for different numbers of smoothing (relaxation) sweeps. A constant coefficient steady Stokes problem is solved on a 512^2 grid in two dimensions (top panels), and 128^3 grid in three dimensions (bottom panels), with no-slip conditions at all domain boundaries.

serve as a reference point for what one may expect in terms of optimal convergence for the GMRES solver.

The top left panel in Fig. 4.3 shows that for periodic constant coefficient problems there is no real difference between using an exact subsolver (many V cycles per application of the preconditioner), and using only a single V cycle in the preconditioner but doing more GMRES iterations. This is not unexpected since multigrid itself is a form of a (non-Krylov) Richardson iterative solver and we expect GMRES to do at least as well. Note that for more difficult Poisson problems, such as problems with large jumps in the coefficients, it is well-known that a Krylov solver preconditioned with multigrid is more robust than plain multigrid, see for example the discussion in Ref. [16].

In the bottom left panel in Fig. 4.3 we show the convergence of GMRES for the variable-coefficient bubble problem, which is typical behavior we observe when there are non trivial boundary conditions or variable coefficients. Similar behavior is observed for the other preconditioners (not shown). The

] B.C. should matter

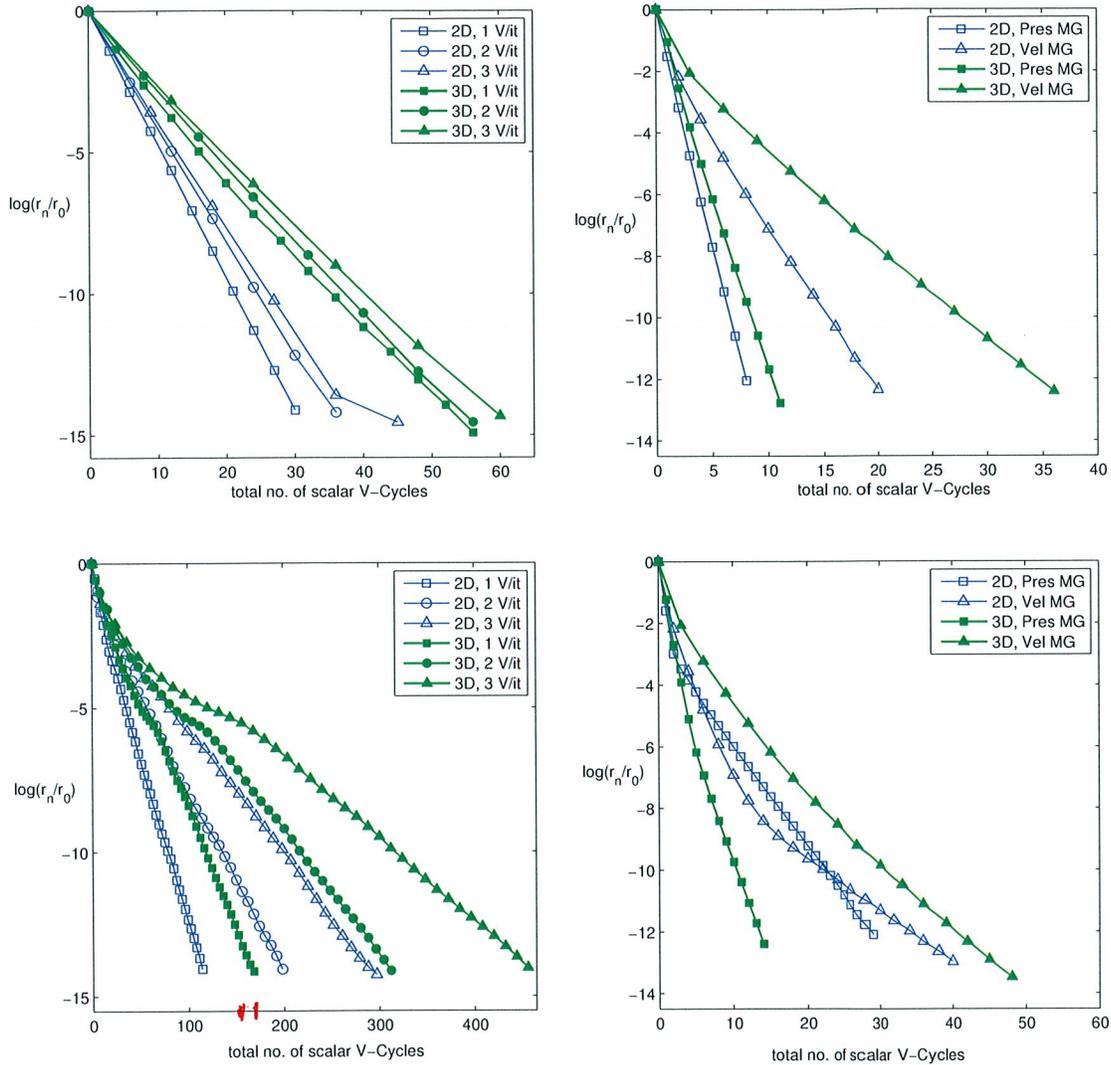


FIG. 4.3. The relative residual (on a log scale) as a function of the total number of scalar multigrid V cycles, for different number of multigrid cycles per application of the pressure and velocity subsolvers. GMRES convergence is shown in the left panels, and pressure (squares) and velocity (triangles) multigrid convergence is shown in the right panels, in both two (512^2 grid, empty symbols) and three (128^3 grid, filled symbols) dimensions. Restarts are not employed in the GMRES solver. The top panels show results for a constant-coefficient periodic steady-state Stokes problem, and the bottom panels show results for the bubble test problem.

results clearly demonstrate that when using exact subsolvers does not give an exact solver, the extra cost of performing more than a single V cycle of multigrid does not pay off in terms of overall efficiency. The optimal rate of convergence is observed when using only a single V cycle in the preconditioner. We have observed no advantage to using a different number of cycles in the pressure and velocity solvers. By comparison with the lower right panel, we see that when using a single multigrid cycle in the preconditioner the total number of scalar V cycles is at most 2-3 times larger than that used in fractional step (projection) methods (for example, $\sim 50 + 15 = 65$ in three dimensions for projection methods as seen in the right panel, and ~ 150 cycles for coupled solver as seen in the left panel).

Based on these observations, henceforth we employ only a single multigrid cycle in the subsolvers employed by the preconditioners.

4.3. Comparison of Preconditioners. Having determined the optimal settings for the pressure and velocity subsolvers, we now turn to exploring the performance of the different preconditioners. We begin by settling an issue regarding the proper choice of sign in the upper/lower triangular and block-diagonal preconditioners.

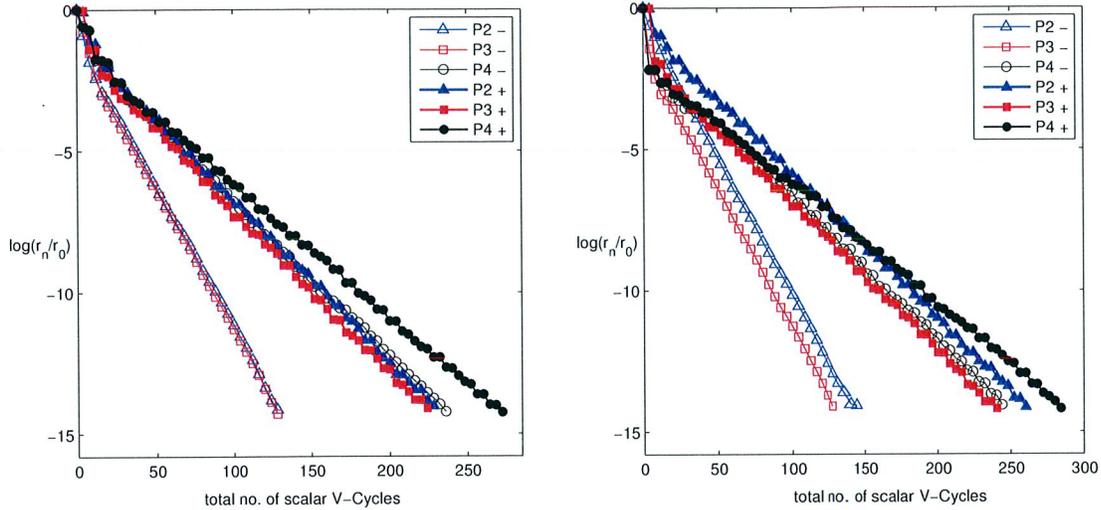


FIG. 4.4. The GMRES convergence history for preconditioners P_2 , P_3 and P_4 with negative (empty symbols) and positive (filled symbols) sign in front of the Schur complement, in three dimensions (128^3 grid). The left panel shows results for a constant-coefficient steady-state Stokes problem with no-slip boundaries, and the right panel shows results for the bubble test problem.

4.3.1. Sign of Schur complement. In this subsection, we show that the choice of sign in front of the Schur complement approximation in P_2 , P_3 and P_4 plays an important role. In the literature [24, 28], the following Schur complement based preconditioners have been proposed and studied,

$$P_{\pm} = \begin{pmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{D} & \pm(-\mathbf{D}\mathbf{A}^{-1}\mathbf{G}) \end{pmatrix},$$

where the sign of the lower diagonal block can be either positive or negative. It was proven that $T_+ = P_+^{-1}M$ satisfies $(T_+ - I)(T_+ + I) = \mathbf{0}$ and $T_- = P_-^{-1}M$ satisfies $(T_- - I)^2 = \mathbf{0}$ [24, 28]. Because the GMRES method possesses a Galerkin property [12], the total number of GMRES iterations is equal to the degrees of the characteristic polynomials of the preconditioned systems. Therefore, GMRES method, using both $P_+^{-1}M$ and $P_-^{-1}M$, converges in 2 iterations if the inverses of \mathbf{A} and the Schur complement are exact. However, when inexact subsolvers are employed, we observe significant difference between the two choices of the sign of the Schur complement. The GMRES convergence histories shown in Fig. 4.4 demonstrate that the preconditioners with "-" sign in front of Schur complement give much better convergence than those with "+" sign. This is consistent with our original choice in Eqs. (2.12) and (2.14).

4.3.2. Comparisons of different preconditioners. Having determined the optimal subsolver settings and the optimal sign of the Schur complement in the lower diagonal block of the preconditioners, we can now compare the performance of the five preconditioners on the bubble test problem in two and three dimensions. The GMRES convergence results shown in Fig. 4.5 demonstrate that for steady Stokes problem the lower and upper triangular preconditioners P_2 and P_3 yield the most efficient GMRES solver. The projection preconditioner P_1 is seen to give a robust convergence but is less efficient for steady flow case because it requires one more scalar (pressure) V cycle per GMRES iteration. The results in the figure also clearly show that P_4 and P_5 are much less efficient. This shows that including an upper or lower triangular block in the Schur complement based block preconditioners

Some reduction better 4.4 4.5

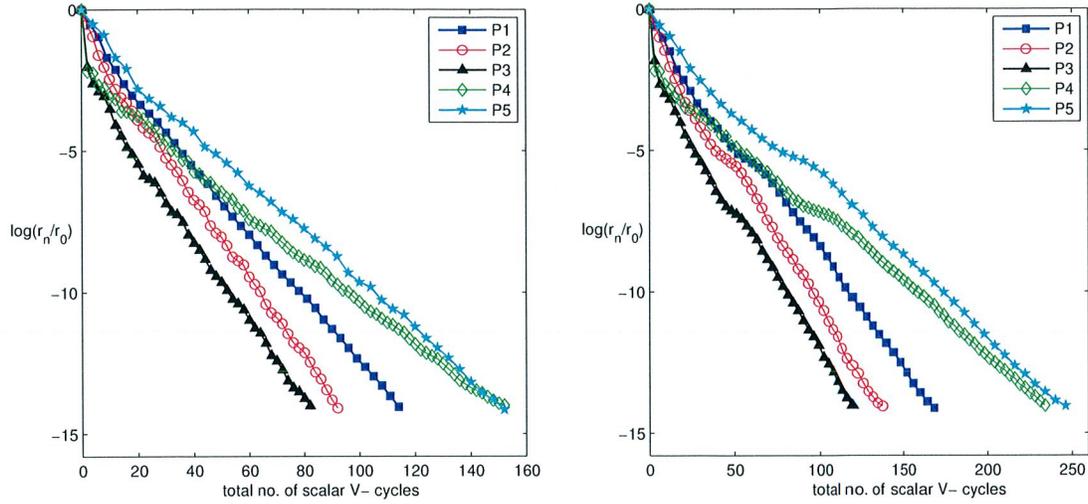


FIG. 4.5. The GMRES convergence history for preconditioners P_1 , P_2 , P_3 , P_4 and P_5 in two (left panel, 512^2 grid) and three (right panel, 128^3 grid) dimensions, for the bubble test problem.

improves convergence, and also shows that the extra work in P_5 over P_1 is not justified in terms of overall efficiency, similarly to how the additional pressure solve in P_1 does not yield improvement.

Based on these observations, henceforth we do not consider P_4 and P_5 .

4.4. Robustness. In this section we examine in more detail the robustness of P_1 and P_2 under GMRES restarts, varying importance of the viscous contribution to A , and changing problem size.

4.4.1. The effects of restarts. For large-scale problems, particularly in three dimensions, the memory requirements of the GMRES algorithm can be excessive. Restarts of the GMRES iteration offer a simple way not only to avoid convergence stalls, but also to limit the memory use. In Fig. 4.6 we compare the behavior of P_1 , P_2 and P_3 for relatively small restart frequencies, 5 or 10 GMRES iterations. In the left panel of the figure we show the behavior for an inviscid time-dependent problem ($L_\mu = 0$, relevant to simulations of large Reynolds number flows) and in the right panel we show the behavior for a steady Stokes problem (relevant to small Reynolds number flows). A two dimensional calculation is shown in the figure but similar results are observed in three dimensions as well. In the left panel of Fig. 4.6 we see that the performance of P_3 significantly deteriorates for the small restart frequency for the inviscid problem. In the right panel of the figure we see some deterioration of the convergence for the small restart frequency for P_2 and P_3 , while P_1 shows robust monotonic convergence even for frequent restarts.

Based on these results, henceforth we use a restart frequency of 10 iterations and focus on examining in more detail the performance of P_1 and P_2 .

4.4.2. Changing viscous number. One of the goals of our study is to design preconditioners that work not just in the steady state limit but also for time-dependent problems. While one can use a suitably-defined Reynolds number to measure the importance of the inertial term $\theta\rho$ in A relative to the viscous term L_μ , the best dimensionless number to use for this is the viscous CFL number

$$\beta = \frac{\nu_0}{\theta h^2} = \frac{\mu_0}{\theta \rho_0 h^2}.$$

A small $\beta \ll 1$ indicates an easier problem where inertial effects dominate, with $\beta = 0$ corresponding to inviscid flow. A large $\beta > L_c^2$ indicates a viscous-dominated problem, where L_c is the grid size, with the hardest case being a steady-state problem $\beta \rightarrow \infty$. In Fig. 4.7 we study the performance of the GMRES Stokes solver for varying viscous CFL numbers for the bubble test problem, in both two and

P_3 why not P_3
 P_3 restart 10 is best