Efficient Variable-Coefficient Finite-Volume Stokes Solvers

Mingchao Cai[1], Andy Nonaka[2], John B. Bell[2], Boyce E. Griffith[3,1,4] and Aleksandar Donev[1,*]

[1] *Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, USA.*
[2] *Center for Computational Sciences and Engineering, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA.*
[3] *Leon H. Charney Division of Cardiology, Department of Medicine, New York University School of Medicine, NY, USA.*
[4] *Department of Mathematics, University of North Carolina, Chapel Hill, NC 27599, USA.*

Abstract. We investigate several robust preconditioners for solving the saddle-point linear systems that arise from spatial discretization of unsteady and steady variable-coefficient Stokes equations on a uniform staggered grid. Building on the success of using the classical projection method as a preconditioner for the coupled velocity-pressure system [*B. E. Griffith, J. Comp. Phys., 228 (2009), pp. 7565–7595*], as well as established techniques for steady and unsteady Stokes flow in the finite-element literature, we construct preconditioners that employ independent generalized Helmholtz and Poisson solvers for the velocity and pressure subproblems. We demonstrate that only a single cycle of a standard geometric multigrid algorithm serves as an effective inexact solver for each of these subproblems. Contrary to traditional wisdom, we find that the Stokes problem can be solved nearly as efficiently as the independent pressure and velocity subproblems, making the overall cost of solving the Stokes system comparable to the cost of classical projection or fractional step methods for incompressible flow, even for steady flow and in the presence of large density and viscosity contrasts. Two of the five preconditioners considered here are found to be robust to GMRES restarts and to increasing problem size, making them suitable for large-scale problems. Our work opens many possibilities for constructing novel unsplit temporal integrators for finite-volume spatial discretizations of the equations of low Mach and incompressible flow dynamics.

*Corresponding author. *Email addresses:* cmchao2005@gmail.com (M. Cai), ajnonaka@lbl.gov (A. Nonaka), jbbell@lbl.gov (J. B. Bell), boyceg@email.unc.edu (B. E. Griffith), donev@courant.nyu.edu (A. Donev)

1   Introduction

Many numerical methods for solving the time-dependent (unsteady) incompressible [1, 3, 24, 27] or low Mach number [14, 43] equations require the solution of a linear unsteady Stokes flow subproblem. The linear steady Stokes problem is of particular interest for low Reynolds number flows [26, 42] or flow in viscous boundary layers. In this work, we investigate efficient linear solvers for the unsteady and steady Stokes equations in the presence of variable density and viscosity. Specifically, we consider the coupled velocity-pressure Stokes system [20, 49]

$$\begin{cases} \rho u_t + \nabla p = \nabla \cdot \tau(u) + f, \\ \nabla \cdot u = g, \end{cases} \tag{1.1}$$

where $\rho(r)$ is the density, $u(r,t)$ is the velocity, $p(r,t)$ is the pressure, $f(r,t)$ is a force density, and $\tau(u)$ is the viscous stress tensor. A nonzero velocity-divergence $g(r,t)$ arises, for example, in low Mach number models because of compositional or temperature variations [43]. The viscous stress $\tau(u)$ is $\mu \nabla u$ for constant viscosity incompressible flow, $\mu[\nabla u + (\nabla u)^T]$ when $g = 0$ (incompressible flow), and $\mu[\nabla u + (\nabla u)^T] + (\gamma - \frac{2}{3}\mu)(\nabla \cdot u)I$ when $g \neq 0$, where $\mu(r,t)$ is the shear viscosity and $\gamma(r,t)$ is the bulk viscosity. When the inertial term is neglected, $\rho u_t = 0$, (1.1) reduces to the time-independent (steady) Stokes equations. In this work we consider periodic boundary conditions and physical boundary conditions that involve velocity only, notably no-slip and free-slip physical boundaries[†].

Spatial discretization of (1.1) can be carried out using standard finite-volume or finite-element techniques. Applying the backward Euler scheme to solve the spatially-discretized equations with time step size Δt gives the following discrete system for the velocity u^{n+1} and the pressure p^{n+1} at the end of time step n,

$$\begin{cases} \rho \left(\dfrac{u^{n+1} - u^n}{\Delta t} \right) + \nabla p^{n+1} = \nabla \cdot \tau \left(u^{n+1} \right) + f^{n+1}, \\ \nabla \cdot u^{n+1} = g^{n+1}, \end{cases} \tag{1.2}$$

where f^{n+1} contains external forcing terms such as gravity and any explicitly-handled terms such as, for example, advection. Similar linear systems are obtained with other implicit and semi-implicit temporal discretizations [1, 3, 27]. In the limit $\rho/\Delta t \to 0$, the system (1.2) reduces to the steady Stokes equations. Here we will assume that the spatial discretization is stable, more precisely, that the Stokes system (1.2) is "uniformly solvable" as the spatial discretization becomes finer, i.e., that a suitable measure of the condition number of the Schur complement of (1.2) remains bounded as the grid spacing

---

[†]When the normal component of velocity is specified on the whole boundary of the computational domain Ω, a compatibility condition $\int_{\partial\Omega} u \cdot n \, dS = \int_{\Omega} g \, dr$ needs to be imposed.

$h \to 0$. In the context of finite-element methods, this is typically implied by the well-known $\inf - \sup$ or Ladyženskaja-Babuška-Brezzi (LBB) condition. Here we employ the classical staggered-grid [32] discretization on a uniform grid, which can be thought of as a rectangular analog of the lowest-order Raviart-Thomas element and is known to be a stable discretization [41, 48]. We expect that it will be relatively straightforward to generalize the preconditioners developed here to recently-developed adaptive mesh staggered schemes [26, 28]. Note, however, that collocated finite-volume discretizations of the Navier-Stokes equations do not provide a stable discretization, which motivates the development of approximate-projection methods [2].

Historically, there have been significant differences in the treatment of (1.2) in the finite-volume and finite-element literature. In the finite-element literature, there is a long history of numerical methods for solving the Stokes equations, especially in the time-independent (steady) context [20, 49]. By contrast, in the context of high-resolution finite-volume methods, the dominant paradigm has been to use a splitting (fractional-step) or projection method [7, 11] to separate the pressure and velocity updates. In part, this choice has been motivated by the target applications, which are often high Reynolds number, or even inviscid, flows. In the inviscid limit, the splitting error associated with projection methods vanishes, and for sufficiently large Reynolds number flows the time step size dictated by advective stability constraints makes the splitting error relatively small. At the same time, the preference for splitting methods stems, in large part, from the perception that solving the saddle-point problem (1.2) is much more difficult than solving the pressure and velocity subproblems; to quote the authors of Ref. [7], "Spatially discretized versions of the coupled Eqs. ⋯ are cumbersome to solve directly." In fact, one of the first second-order projection methods [3] was developed by starting with a Crank-Nicolson variant of (1.2) and then approximating the resulting Stokes system using a velocity-pressure splitting that was motivated by the perceived difficulty in solving the coupled system.

Fractional-step approaches, however, suffer from several significant shortcomings. It is well-known, for example, that the splitting introduces a commutator error that leads to the appearance of spurious or "parasitic" modes [7, 15] in the presence of physical boundaries. Furthermore, it is generally not possible to impose the true boundary conditions of the Stokes system in a fractional-step scheme; instead, "artificial" boundary conditions must be imposed in the velocity and pressure subsystems. This has motivated the construction of methods that *approximately* solve the Stokes system (1.2) using block-triangular factorizations [44] similar to those employed here to construct preconditioners for iterative methods that solve (1.2) *exactly*. This is crucial at small Reynolds numbers because the splitting error becomes larger as viscous effects become more dominant, and projection or approximate factorization methods do not apply in the steady Stokes regime for problems with physical boundary conditions.

Recognizing these problems, one of us investigated the use of projection-like methods as preconditioners for a Krylov method for solving the coupled system (1.2) [27].

It was found that, contrary to traditional finite-volume wisdom and in agreement with extensive experience in the finite-element context, the saddle-point problem (1.2) can be efficiently solved using standard multigrid techniques for the velocity and pressure sub-problems, for a broad range of parameters. Here we improve and generalize the preconditioners developed in Ref. [27] to account for variable density and variable viscosity, as well as to robustly handle small or zero Reynolds number (steady) flows. Our primary motivation for this work is the development of semi-implicit integrators for the low Mach number equations of fluctuating hydrodynamics for multicomponent fluid mixtures [14]. For these applications it is important to treat the viscosity implicitly (including the limit of steady Stokes flow [13]) due to the large separation of time scales between momentum and mass diffusion (i.e., large Schmidt number). In order to properly include thermal fluctuations with implicit viscous handling it is also necessary to use a coupled Stokes formulation instead of split (projection method) approaches [12, 50].

The preconditioners we investigate here numerically are drawn from the large finite-element literature on Stokes solvers [4, 5, 18–22, 34, 35, 37–40, 42, 49]. We will not attempt to review the extensive finite-element literature on preconditioners for Stokes flow here; instead, we will point out the similarities and differences with prior work for each of the preconditioners that we study. When necessary, we generalize the existing preconditioners to finite Reynolds numbers (unsteady Stokes equations) and to variable density and variable viscosity problems. We investigate several alternative preconditioners that solve the velocity and pressure subproblems in different orders. In the finite-element context, variable-viscosity steady Stokes solvers based on several of the preconditioners we investigate here have been developed by several groups [8, 25], and have already been successfully scaled to massively-parallel architectures and very difficult large-contrast geophysical problems. In the finite-volume context, the work most closely related to our work is Ref. [24], which focuses on steady Stokes flow in the presence of large viscosity contrast (i.e., discontinuities) for geodynamic applications. Notably, both our work and the work presented in Ref. [24] are based on a staggered finite-volume discretization and geometric multigrid solvers.

Our primary contribution in this work is that we investigate in detail the computational performance of a collection of four standard preconditioners over a broad range of parameters in the context of a specific but very efficient (in terms of number of degrees of freedom per grid cell) finite-volume discretization. By carefully designing and optimizing the parameters for all of the key components of the solvers, ranging from the geometric multigrid smoothers to the restart frequency of the GMRES solver, we construct a complete solver that can readily be employed to construct novel unsplit temporal integrators for finite-volume (conservative) spatial discretizations of the equations of low Mach and incompressible flow dynamics. Importantly, these unsplit schemes can use the same building blocks (e.g., geometric multigrid solvers and high-resolution advection techniques) and achieve a similar computational complexity as traditional projection methods, as we will demonstrate in future work.

The preconditioners that we investigate are built using two crucial subsolvers. The first of these is a linear solver for the inviscid problem

$$
\begin{cases}
\rho\left(\dfrac{u^{n+1}-u^{n}}{\Delta t}\right)+\nabla p^{n+1}=f^{n+1}, \\
\nabla\cdot u^{n+1}=g^{n+1},
\end{cases}
\tag{1.3}
$$

the solution of which requires solving a density-weighted pressure Poisson equation

$$
-\Delta t\,\nabla\cdot\left(\rho^{-1}\nabla p^{n+1}\right)=g^{n+1}-\nabla\cdot\left(u^{n}+\rho^{-1}f^{n+1}\Delta t\right).
$$

For the staggered-grid finite-volume discretization we employ here, this Poisson problem can efficiently be solved using standard geometric multigrid techniques [1]. The second subsolver required by the preconditioners is a linear solver for the unconstrained variable-coefficient velocity equation,

$$
\rho\left(\dfrac{u^{n+1}-u^{n}}{\Delta t}\right)=\nabla\cdot\tau\left(u^{n+1}\right)+f^{n+1}.
\tag{1.4}
$$

Note that both (1.3) and (1.4) use the same boundary conditions for velocity as the coupled problem, and that natural boundary conditions are required for the pressure when solving (1.3) on a staggered grid. For constant viscosity incompressible flow $\nabla\cdot\tau(u)=\mu\nabla^{2}u$ and therefore (1.4) is a system of d uncoupled Helmholtz equations, where d is the dimensionality. These can be solved efficiently using standard geometric multigrid techniques. For variable viscosity flows or when $g\neq 0$ the different components of velocity are coupled. Here we develop an effective geometric multigrid method for solving (1.4) that generalizes the classical red-black coloring smoother for the scalar Poisson equation. Since the solution of either (1.3) or (1.4) is itself a costly iterative process, it is crucial that the preconditioners require only approximate subsolvers. More precisely, preconditioning should only require the application of linear operators that are spectrally-equivalent [20] to the exact solution operators for (1.3) or (1.4). Here we use one or a few cycles of geometric multigrid as approximate solvers for these subproblems.

The preconditioners investigated in this work can be easily generalized to other spatial discretizations and boundary condition types by simply modifying the approximate subsolvers for (1.3) and (1.4). For example, boundary conditions that couple pressure and viscous stress can be handled by imposing approximate boundary conditions for the subsolvers. In Ref. [27], at physical boundaries on which normal tractions (normal components of the stress tensor) are prescribed, Neumann conditions are imposed on the normal velocity component when solving (1.4) and Dirichlet conditions are imposed for the pressure when solving (1.3). For adaptively-refined meshes [26, 28], multilevel geometric multigrid techniques can be used to solve the pressure and velocity subproblems [1, 28]. To the best of our knowledge, however, there is presently no (LBB) stable conservative

discretization of the *steady* Stokes equations on block-structured multi-resolution staggered grids; existing schemes do not maintain at least one of the key properties of uniform discretization: conservation, uniform solvability, and symmetry. We therefore do not consider here non-uniform staggered grids and focus our attention on uniform staggered grids.

The organization of this paper is as follows. In Section 2, we introduce several preconditioners based on approximating the inverse of the Schur complement. In Section 3 we specialize to a particular staggered-grid second-order finite-volume discretization and give details of our numerical implementation. In Section 4 we perform a detailed study of the efficiency and robustness of the various preconditioners, and select the optimal values for several algorithmic parameters. Finally, we offer some conclusions in Section 5, and then give several technical derivations in an extensive Appendix.

2   Preconditioners

In this section we present several preconditioners for solving the saddle-point linear system (1.2) that arises after spatio-temporal discretization of (1.1). Much of the discussion presented here has already appeared scattered through many diverse works in the literature; for the benefit of the reader we provide a condensed but complete summary of the key derivations. For increased generality, we write this system in the form,

$$M \begin{pmatrix} x_u \\ x_p \end{pmatrix} = \begin{pmatrix} A & G \\ -D & 0 \end{pmatrix} \begin{pmatrix} x_u \\ x_p \end{pmatrix} = \begin{pmatrix} b_u \\ b_p \end{pmatrix}, \tag{2.1}$$

where $(x_u, x_p)^T$ denote the velocity and pressure degrees of freedom, $(b_u, b_p)^T$ are the velocity and pressure right hand sides, D denotes a discrete divergence operator, and G is a discrete gradient operator. Note that for the staggered-grid discretization that we describe in Section 3, the gradient and divergence operators are negative adjoints of each other for periodic, no-slip, and free-slip boundary conditions, $G = (-D)^*$, where star denotes adjoint, making $M = M^\star$ a self-adjoint matrix (non-symmetric saddle-point systems can also be considered [10]). Here the linear velocity operator $A = \theta \rho - L_\mu$ combines inertial and viscous effects, where θ is a parameter that is zero for steady Stokes flow, and $\theta \sim \Delta t^{-1}$ for unsteady flow. The operator ρ is a mass density matrix (distinct from the standard finite element mass matrix), such that ρx_u is a spatially-discrete (conserved) momentum field. The viscous operator is denoted with L_μ, with $L_\mu u$ being a spatial discretization of $\nabla \cdot \tau(u)$.

The saddle-point problem (2.1) can formally be solved by using the inverse of the Schur complement,

$$S^{-1} = \left(-DA^{-1}G \right)^{-1},$$

to obtain the exact solution for the pressure,

$$x_p = -S^{-1}(DA^{-1}b_u + b_p), \tag{2.2}$$

and for the velocity degrees of freedom,

$$x_u = A^{-1}(b_u - Gx_p) = A^{-1}b_u + A^{-1}GS^{-1}(DA^{-1}b_u + b_p). \qquad (2.3)$$

These formal solutions are not useful in practice because the Schur complement cannot be formed explicitly for large three-dimensional grids, nor inverted efficiently. In Ref. [24], the authors investigate evaluating the action of S^{-1} in (2.2) by an outer Krylov solver, which itself relies on evaluating the action of A^{-1} in an inner (nested) Krylov solver. We do not investigate this approach here and instead focus on what the authors of Ref. [24] call the "fully coupled preconditioned approach", in which an approximation of the Schur complement solution is used to construct an effective preconditioner for a Krylov solver applied to the saddle-point problem (2.1). The key part in designing preconditioners for (2.1) is approximating the (inverse of the) Schur complement, specifically, constructing an operator $\mathcal{S}^{-1}$ that is spectrally-equivalent to S^{-1} [19].

To motivate the approximation of S^{-1}, let us consider the case of constant viscosity μ_0 and constant density ρ_0. In this case $A = \theta\rho_0 I - \mu_0 L$, where I denotes an identity matrix and L is a discrete vector Laplacian operator, constructed taking into account the imposed velocity boundary conditions. We then have

$$S^{-1} = \left[-D(\theta\rho_0 I - \mu_0 L)^{-1}G \right]^{-1} \approx \left[(-DG)\left(\theta\rho_0 I - \mu_0 L_p\right)^{-1} \right]^{-1} = -\theta\rho_0 L_p^{-1} + \mu_0 I, \quad (2.4)$$

where $L_p = DG$ denotes a scalar (pressure) discrete Laplacian operator, and have assumed the commuting property $LG \approx GL_p$, which is an exact identity for the staggered grid discretization applied to periodic systems. This approximation to the Schur complement inverse has been used in the finite-element context in Ref. [36] and in the finite-volume approach in Ref. [27]; an in-depth discussion of the use of approximate commutators for constructing preconditioners can be found in Ref. [17].

Here we generalize (2.4) to variable density and viscosity through a simple construction. The basic idea is that the first part of the Schur complement approximation, $\theta\rho_0 L_p^{-1}$, corresponds to the inviscid limit. For variable density, this term becomes θL_ρ^{-1}, where

$$L_\rho = D\rho^{-1}G$$

is a discretization of the density-weighted Poisson operator $\nabla \cdot \rho^{-1}\nabla$ that also appears in traditional variable-density projection methods [1]. Therefore, for variable-density, constant-viscosity flow, $\nabla \cdot \tau = \mu_0 \nabla^2 u$, and we employ the approximation

$$S^{-1} \approx \mathcal{S}^{-1} = -\theta L_\rho^{-1} + \mu_0 I. \qquad (2.5)$$

The term $\mu_0 I$ in (2.4) is an analogue of the viscous operator L_μ that acts on pressure-like degrees of freedom instead of velocity-like degrees of freedom. This has to be constructed on a case-by-case basis, and in the constant viscosity setting it corresponds to the viscous pressure-correction term proposed by Brown, Cortez and Minion [7] in the context of second-order projection methods. For incompressible flow, $\tau(u) = \mu[\nabla u + (\nabla u)^T]$,

the Fourier-space calculation described in Appendix A suggests replacing the term $\mu_0 \boldsymbol{I}$ with $2\boldsymbol{\mu}$, where $\boldsymbol{\mu}$ is a diagonal matrix of viscosities corresponding to each pressure degree of freedom. This gives the Schur complement inverse approximation

$$S^{-1} \approx \mathcal{S}^{-1} = -\theta L_\rho^{-1} + 2\boldsymbol{\mu}, \tag{2.6}$$

which is called the "local viscosity" preconditioner in Ref. [24]. Note however that the prefactor of two suggested by the analysis in Appendix A is not included in Eq. (36) in Ref. [24]. When bulk viscosity is included, $\boldsymbol{\tau}(\boldsymbol{u}) = \mu[\boldsymbol{\nabla u} + (\boldsymbol{\nabla u})^T] + (\gamma - \frac{2}{3}\mu(\boldsymbol{\nabla} \cdot \boldsymbol{u}))\boldsymbol{I}$, we take

$$S^{-1} \approx \mathcal{S}^{-1} = -\theta L_\rho^{-1} + \left(\gamma + \frac{4}{3}\mu\right), \tag{2.7}$$

where γ is the diagonal matrix of bulk viscosities. As we demonstrate in Appendix A, these approximations are exact for periodic systems if the density and viscosity are constant. In all other cases they are approximations that are expected to be good in regions far from boundaries where the coefficients do not vary significantly. Our numerical experiments support this intuition.

We have investigated the alternative approximations

$$S^{-1} \approx -\theta L_\rho^{-1} - L_\rho^{-1} D\rho^{-1} L_\mu \rho^{-1} G L_\rho^{-1}, \tag{2.8}$$

as well as

$$S^{-1} \approx -\theta L_\rho^{-1} - L_p^{-1}\left(DL_\mu G\right)L_p^{-1},$$

which is similar to the so-called BFBt preconditioner of Elman [19] in the steady-state case, and which is also investigated in Ref. [24]. These approximations utilize the velocity boundary conditions since they involve the viscous operator L_μ, unlike the pressure-space viscous operator in (2.6) which does not make use of the velocity boundary conditions. We have observed similar behavior for the more expensive approximation (2.8) as with the simpler and significantly more efficient approximation (2.6). We therefore do not investigate BFBt-type preconditioners in this work.

As explained in Appendix B, the spectrum of the preconditioned operators for the preconditioners we consider next is determined by the spectrum of $\mathcal{S}^{-1}S$. In that appendix, we demonstrate with a combination of analytical techniques and numerical computation that this operator has a very clustered spectrum even in the presence of nontrivial boundary conditions and large variations in viscosity.

2.1 Projection preconditioner

In the first preconditioner we consider, which we will denote with P_1, we use one step of the classical projection method [3, 11, 27] as a preconditioner. In P_1, we use (2.2) to estimate the pressure, and make a commuting assumption in (2.3),

$$A^{-1}GS^{-1} = A^{-1}G(-DA^{-1}G)^{-1} \approx -A^{-1}A\rho^{-1}GL_\rho^{-1} = -\rho^{-1}GL_\rho^{-1},$$

which gives the velocity estimate

$$x_u \approx A^{-1}b_u - \rho^{-1}GL_\rho^{-1}(DA^{-1}b_u + b_p). \tag{2.9}$$

Note that this velocity estimate (2.9) satisfies the divergence condition exactly, $Dx_u = -b_p$. More precisely, x_u is the L_2 projection of the unconstrained velocity estimate $A^{-1}b_u$ onto the divergence constraint.

In practical implementations, the exact subproblem solvers need to be replaced by approximations. Specifically, A^{-1} is approximated by the inexact velocity solver $\widetilde{A}^{-1}$, L_ρ^{-1} is approximated by the inexact pressure Poisson solver $\widetilde{L}_\rho^{-1}$, and S^{-1} is replaced by $\widetilde{\mathcal{S}}^{-1}$, which is an approximation to the approximate Schur complement inverse $\mathcal{S}^{-1}$ given by (2.6) for incompressible flow. In summary, for the variable-coefficient Stokes problem, the projection preconditioner P_1 is defined by the block factorization

$$P_1^{-1} = \begin{pmatrix} I & \rho^{-1}G\widetilde{L}_\rho^{-1} \\ 0 & \widetilde{\mathcal{S}}^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -D & -I \end{pmatrix} \begin{pmatrix} \widetilde{A}^{-1} & 0 \\ 0 & I \end{pmatrix}. \tag{2.10}$$

This factorization clearly shows the main steps in the application of the preconditioner. First, a velocity subproblem is solved inexactly (right-most block) to compute $x_u^* = \widetilde{A}^{-1}b_u$. Second, $b_c = Dx_u^* + b_p$ is computed (middle block). Third, a Poisson problem is solved approximately to compute $\widetilde{L}_\rho^{-1}b_c$ and, lastly, the pressure and velocity estimates are evaluated (first block). For constant-coefficient periodic problems with exact subsolvers, the projection preconditioner is an exact solver for the coupled Stokes equations since both (2.4) and (2.9) are exact.

For the constant viscosity and density Stokes problem, a projection preconditioner very similar to P_1 was first proposed by one of us in Ref. [27]. In this work we generalize the projection preconditioner to the case of variable viscosity and density. Even in the constant-coefficient case, there is a small but important difference between P_1 and the previous projection preconditioner in Ref. [27], which uses the following approximation of the Schur complement inverse,

$$S^{-1} \approx \widetilde{\mathcal{S}}^{-1} = -\left(\theta\rho_0 I - \mu_0 L_p\right)\widetilde{L}_p^{-1},$$

rather than the approximation (2.5) used here, $\widetilde{\mathcal{S}}^{-1} = -\theta\rho_0\widetilde{L}_p^{-1} + \mu_0 I$, which we have found to give a slightly more efficient solver. The two approximations are identical when exact Poisson solvers are used, $\widetilde{L}_p^{-1} = L_p^{-1}$, but not when an approximate solver is employed.

2.2 Lower triangular preconditioner

For our second preconditioner, which we denote with P_2, we use (2.2) for the pressure estimate, but the velocity estimate takes the simpler form

$$x_u \approx A^{-1}b_u, \tag{2.11}$$

which is obtained by discarding the second part in (2.3). If we further approximate the matrix inverses with inexact solves, namely, replacing A^{-1} by $\widetilde{A}^{-1}$, L_ρ^{-1} by $\widetilde{L}_\rho^{-1}$, and S^{-1} by $\widetilde{S}^{-1}$, the second preconditioner is given by the block factorization

$$P_2^{-1} = \begin{pmatrix} I & 0 \\ 0 & -\widetilde{S}^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ D & I \end{pmatrix} \begin{pmatrix} \widetilde{A}^{-1} & 0 \\ 0 & I \end{pmatrix}. \tag{2.12}$$

By combing all the terms in the right hand side of (2.12), we see that P_2 is actually an approximation of the inverse of the lower triangular preconditioner previously studied by several other groups [9,34,35,37,38],

$$P_2^{-1} \approx \begin{pmatrix} A & 0 \\ -D & -S \end{pmatrix}^{-1}. \tag{2.13}$$

Notice that for steady Stokes flow, $\theta = 0$, the application of P_2^{-1} does not require any pressure Poisson solvers, unlike the projection preconditioner. Therefore, a single application of P_2^{-1} can be significantly less expensive computationally than an application of P_1^{-1}. For unsteady flows P_1 and P_2 involve nearly the same operations and applying them has similar computational cost.

2.3 Upper triangular preconditioner

Alternatively, one can assume $DA^{-1}b_u \approx 0$ to obtain $x_p = -S^{-1}b_p$ and

$$x_u = A^{-1}(b_u + GS^{-1}b_p).$$

Replacing the exact solvers with inexact solvers, we obtain our third preconditioner in block factorization form,

$$P_3^{-1} = \begin{pmatrix} \widetilde{A}^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -G \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -\widetilde{S}^{-1} \end{pmatrix}, \tag{2.14}$$

which is exactly the same as the "fully coupled" approach with the "local viscosity" preconditioner studied in Ref. [24] and also the block-triangular preconditioner of Ref. [25], generalized here to time-dependent problems. If we combine all the terms in the right hand side of (2.14), then we see that P_3 is actually an approximation of the inverse of the upper triangular preconditioner [9,34,35,37,38],

$$P_3^{-1} \approx \begin{pmatrix} A & G \\ 0 & -S \end{pmatrix}^{-1}. \tag{2.15}$$

The computational cost of applying P_3^{-1} is very similar to that of applying P_2^{-1}.

2.4   Other preconditioners

In addition to the three main preconditioners (projection, lower and upper triangular) we study here, we have investigated some other preconditioners. The simplest Schur-complement based preconditioner one can construct is the block diagonal preconditioner [9, 35, 37, 38]

$$P_4^{-1} = \begin{pmatrix} \widetilde{A}^{-1} & \mathbf{0} \\ \mathbf{0} & -\widetilde{\mathcal{S}}^{-1} \end{pmatrix}. \tag{2.16}$$

This preconditioner has the lowest computational cost of all the preconditioners per Krylov iteration, but it also yields the poorest approximation to the exact solution (2.2)-(2.3). Note, however, that the use of a diagonal preconditioner can make the preconditioned operator symmetric and thus allow for the use of more efficient (short-recurrence) Krylov solvers such as MINRES. This is exploited in Ref. [8] to construct a robust and highly-scalable finite-element discretization of the variable-viscosity steady Stokes equations, using a single cycle of algebraic multigrid for a Laplacian approximation to A as an approximate velocity solver.

In Appendix B, we show that P_1, P_2 and P_3 all give the same spectrum for the preconditioned linear operator. It is also well-known that P_1, P_2, P_3, and P_4 are all spectrally-equivalent if exact solvers are used [20]. Furthermore, if an exact Schur complement inverse is employed, it can be shown for P_2, P_3, and P_4 that any Krylov subspace iterative method with a Galerkin property will require only a small number of iterations (two or three) to converge to the exact answer [39].

As an alternative approximation to (2.2)-(2.3) that is more accurate than the previous approximations, we consider a fifth preconditioner closely-related to the Uzawa method [33], denoted by P_5. The action of the inverse of this preconditioner P_5^{-1} cannot easily be written in block-factorization form so we present in the form of pseudo-code:

1. Solve for $x_u^* = \widetilde{A}^{-1} b_u$ using multigrid with initial guess $\mathbf{0}$.
2. Estimate pressure as $x_p \approx -\widetilde{\mathcal{S}}^{-1}(D x_u^* + b_p)$.
3. Estimate velocity as $x_u \approx \widetilde{A}^{-1}(b_u - G x_p)$ using a multigrid solver, starting with x_u^* as an initial guess.

If exact solvers are employed the only approximation made in P_5 is the approximation $S^{-1} \approx \widetilde{\mathcal{S}}^{-1}$, and as such we expect it to be the best approximation to M^{-1}. It is, however, also the most expensive of the five preconditioners because it involves two applications of $\widetilde{A}^{-1}$. Our goal will be to investigate how well these preconditioners perform in practice with inexact subsolvers.

3   Numerical implementation

In this section we specialize the relatively general preconditioners from the previous section to a specific second-order conservative finite-volume discretization of the time-

dependent Stokes equations on a uniform rectangular grid. We do not discuss here the inclusion of advection in the full Navier-Stokes equations. Schemes that handle advection explicitly using a non-dissipative spatial discretization are described in detail in Refs. [14, 50], and Ref. [27] describes a particular higher-order upwind scheme for uniform staggered-grids.

3.1 Staggered-grid discretization

For our numerical investigations of the various preconditioners we employ the well-known staggered-grid or MAC discretization of the Stokes equations [31, 32]. This is a conservative discretization that is uniformly div-stable [41, 48]. The scheme defines the degrees of freedom at staggered locations. Specifically, scalar variables including pressure and density are defined at cell centers, while components of vector variables including velocity components are defined at the corresponding faces of the grid [27, 50]. For illustration, we assume that the domain Ω is rectangular and there are n_x cells along the x direction and n_y cells along the y direction, with periodic, no-slip (e.g., $\boldsymbol{u}=0$ along a boundary) or free-slip (e.g., $v=0$ and $\partial u/\partial y=0$ along the south boundary) boundary conditions specified at each of the domain boundaries. For simplicity, we further assume that the grid spacing along the different directions is constant, $h_x=h_y=h$.

The divergence of $\boldsymbol{u}=(u,v)^T$ is approximated at cell centers by $\boldsymbol{Du}=D^xu+D^yv$ with

$$(D^xu)_{i,j}=h^{-1}\left(u_{i+\frac{1}{2},j}-u_{i-\frac{1}{2},j}\right), \qquad (D^yv)_{i,j}=h^{-1}\left(v_{i,j+\frac{1}{2}}-v_{i,j-\frac{1}{2}}\right).$$

The gradient of p is approximated at the x and y edges of the grid cells (faces in three dimensions) by $\boldsymbol{G}p=(G^xp,G^yp)^T$ with

$$(G^xp)_{i-\frac{1}{2},j}=h^{-1}\left(p_{i,j}-p_{i-1,j}\right), \qquad (G^yp)_{i,j-\frac{1}{2}}=h^{-1}\left(p_{i,j}-p_{i,j-1}\right).$$

For periodic domains or where a homogeneous Dirichlet condition is specified for the normal component of velocity at physical boundaries, the staggered discretization satisfies $\boldsymbol{D}=-\boldsymbol{G}^*$. Note that $\boldsymbol{DG}=\boldsymbol{L}_p$, where $\boldsymbol{L}_p$ is the standard (five-point in two dimensions, seven-point in three dimensions) centered finite difference Laplacian.

For constant viscosity, the finite difference approximation to the vector Laplacian $\boldsymbol{\nabla}^2\boldsymbol{u}$ is denoted as $\boldsymbol{Lu}=(L^xu,L^yv)$. In the interior of the domain, $\boldsymbol{\nabla}^2u$ is discretized using the standard five-point discrete Laplacian. In the presence of physical boundaries, L^xu is defined at all interior edges/faces where u are defined, and L^yv is defined at all interior edges/faces where v are defined. The finite-difference stencils for tangential velocities next to no-slip and free-slip boundaries are modified to account for the boundary conditions, as described in Refs. [27, 50]. Note that for constant viscosity, if one uses the Laplacian form of the viscous term, the different components of velocity are uncoupled.

When the viscosity is not a constant, the strain tensor form of the viscous term is

needed, for which

$$L_\mu u = \nabla \cdot \tau(u) = \begin{bmatrix} 2\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) \\ 2\frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial y}\right) + \frac{\partial}{\partial x}\left(\mu\frac{\partial v}{\partial x} + \mu\frac{\partial u}{\partial y}\right) \end{bmatrix}. \tag{3.1}$$

The discretization of $\nabla \cdot \tau(u)$ is constructed using standard (staggered) centered second-order differences to give the discrete viscous operator L_μ. Note that even for constant viscosity, there is coupling between the velocity components in (3.1). For the staggered discretization that we employ here, it can be shown that for constant viscosity μ_0, the viscous operator degenerates to a Laplacian, $L_\mu u = \mu_0 L u$, if $Du = 0$. That is, for constant viscosity incompressible flow the solution of the Stokes system is not affected by the choice of the form of the viscous term. (By contrast, in fractional step methods, the unprojected velocity and therefore the projected velocity is affected by this choice.) However, the Stokes solver is in general affected by the choice of the viscous term, even for constant viscosity. As described in Ref. [14], centered differences for the viscous fluxes that require values outside of the physical domain are replaced by one-sided differences that only use values from the interior cell bordering the boundary and boundary values. The tangential momentum flux is set to zero for any faces of the corresponding control volume that lie on a free-slip boundary, and values in cells outside of the physical domain are never required. The overall discretization is spatially globally second-order accurate.

We build the discrete velocity operator $A = \theta\rho - L_\mu$ from the above centered finite-difference operators. We assume that the density ρ is specified at the cell centers. The density matrix ρ is constructed by defining the discrete momentum density ρu at the cell faces, where the corresponding velocity components are defined. Here we follow Ref. [14] and average the density from cell centers to cell faces,

$$(\rho u)_{i+\frac{1}{2},j} = \left(\frac{\rho_{i,j} + \rho_{i+1,j}}{2}\right)u_{i+\frac{1}{2},j} \quad \text{and} \quad (\rho u)_{i,j+\frac{1}{2}} = \left(\frac{\rho_{i,j} + \rho_{i,j+1}}{2}\right)v_{i,j+\frac{1}{2}},$$

giving a diagonal density matrix ρ with the interpolated face-centered densities along the diagonal. We will assume here that the shear μ and bulk γ viscosities are specified at the cell centers; typically they are an explicit function of other scalar variables such as density, temperature, and/or composition. The matrices μ and γ that appear in the approximation to the Schur complement (e.g. Eq. (2.7)) are diagonal matrices containing the cell-centered values of the shear and bulk viscosities. The discretization of the viscous operator L_μ requires a shear viscosity at both cell-centers and nodes (edges in three dimensions). The value of μ at a node is set to be the average of the four neighboring cell-centered values [14]. Note that the Schur complement approximation uses only the cell-centered and not the node-centered viscosities.

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 itera-

tive Krylov solver. The basic operation required by the Krylov solver is computing Mx, which amounts to a straightforward direct evaluation of the appropriate finite-difference stencils at every interior face and every cell center in the computational grid.

Application of any of the preconditioners requires implementing approximate solvers for the pressure and velocity subproblems (i.e., application of $\widetilde{L}_\rho^{-1}$ and $\widetilde{A}^{-1}$). Here we employ geometric multigrid techniques to implement these solvers. For the cell-centered pressure solver we use standard variable-coefficient Poisson multigrid solvers [1]. For the face-centered velocity solver we develop a vector variant of a standard scalar Helmholtz solver based on a generalized red-black Gauss-Seidel smoother. The details of the multigrid algorithms are given in Appendix C. Our implementation is based on the Fortran version of the BoxLib library [45].

Note that because the preconditioners in Krylov methods are applied to a residual-correction system, zero is an appropriate initial guess for the multigrid subsolvers. Note also that for certain choices of boundary conditions the pressure subproblem has a null space of constant vectors. 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 implementation of 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). When non-homogeneous velocity boundary conditions are specified, the viscous operator is an affine operator because the viscous stencils near the boundary use the specified boundary values. 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 subtracting the boundary terms in a pre-processing step.

Here we employ left preconditioning and apply the iterative solver to the preconditioned system $P^{-1}Mx = P^{-1}b$. The convergence criterion for the Krylov solver is therefore most naturally expressed in terms of either the absolute or relative reduction in the magnitude of the preconditioned residual $r_P = \|P^{-1}(Mx-b)\|_2$. A more robust alternative is to base convergence criteria on the value of the unpreconditioned (true) residual $r = \|Mx-b\|_2$. For the problems studies here we observe r and r_P to exhibit similar convergence for well-scaled problems.

3.3   Rescaling of the linear system

Another issue that arises when solving saddle-point problems is that of scaling of the system to minimize the loss of floating point precision occurs when adding the different terms. This is particularly important when the equations are solved in dimensional form (with physical units), but can also be important even when the equations are non-dimensionalized. We consider rescaling the velocity equations by some constant c and rescaling the pressure unknowns by the same factor, to obtain the rescaled system

$$\begin{pmatrix} cA & G \\ -D & 0 \end{pmatrix} \begin{pmatrix} x_u \\ cx_p \end{pmatrix} = \begin{pmatrix} cb_u \\ b_p \end{pmatrix}. \tag{3.2}$$

Intuitively, a well-scaled Stokes system is one in which both velocity-like and pressure-like unknowns have elements of similar typical magnitude. In order not to lose precision when evaluating the left-hand-side of the velocity equation, we would like the viscous $c\boldsymbol{A}\boldsymbol{x}_u$ and pressure $\boldsymbol{G}\boldsymbol{x}_p$ contributions 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. Numerical experiments confirm that rescaling the viscosity from a typical value μ_0 to $c\mu_0 \sim h$ can dramatically improve the numerical conditioning of the Stokes system. Note that no rescaling of the divergence constraint is necessary since $\boldsymbol{D}\boldsymbol{x}_u$ has magnitude $\sim h^{-1}$ as the rest of the terms. Similarly, using equal weighting for velocity and pressure residuals in the residual-space inner product will not pose any problems because the two components of the residual $c\boldsymbol{b}_u$ and $\boldsymbol{b}_p$ already have similar magnitude. 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. [24] for a specific formulation. To avoid loss of accuracy, in extreme cases extended precision arithmetic may need to be used in the solver [24].

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. We have verified that after the rescaling (3.2) we get similar results for other choices of reference values for the viscosity and grid spacing. We emphasize that if the rescaling is not applied (i.e., $c\!=\!1$ is used), multiple linear algebra issues arise when solving the Stokes problem. For example, GMRES may not converge, or if it converges, there may be a large difference between the preconditioned and unpreconditioned residuals, and lastly, the computed solution may have a large error due to ill-conditioning.

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 [46, 47]. A more robust and flexible method is FGMRES (Flexible GMRES). In particular, FGMRES allows the use preconditioners that are not necessarily constant linear operators (e.g., another Krylov solver or a variable number of multigrid cycles). A notable drawback of FGMRES is that it requires twice the storage of GMRES. Since one of our goals is to develop solvers for large-scale calculations, we do not consider FGMRES here and use the less memory-intensive GMRES method. GMRES requires the storage of m vectors like $\boldsymbol{x}$. 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. For testing purposes, we generate a random x and then compute $b = Mx$, with zero velocity along all non-slip boundaries. Similar convergence (not shown) is observed for other right-hand sides or boundary velocities.

The multigrid algorithms used in the pressure and velocity subsolvers iteratively apply V cycles, each of which consists of successive hierarchical restriction (coarsening), smoothing, and prolongation [6]. In our tests, we will use the same 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 $\widetilde{L}_\rho^{-1}$) is n scalar V cycles, and the cost of the velocity subsolver (application of $\widetilde{A}^{-1}$) is $d \cdot n$ 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 steady flow.

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 n_c cells with no-slip boundaries along all domain boundaries. For the bubble problem, the viscous stress is taken to be $\tau(u) = \mu[\nabla u + (\nabla u)^T]$, and the diagonal elements of the viscosity matrix μ and the density matrix ρ at cell centers are generated from the spatially-dependent functions $\mu(x) = \mu_0 f(x; r_\mu)$ and $\rho(x) = \rho_0 f(x; r_\rho)$ respectively, where

$$f(x;r) = \frac{1}{2}(r+1) + \frac{1}{2}(r-1)\tanh\left(\frac{\mathrm{d}(x,\Gamma)}{\epsilon}\right) + 0.1R. \tag{4.1}$$

Here r_μ and r_ρ are the viscosity and density contrast ratios, Γ is the interface, a sphere of radius $L/4$ placed at the center of a cube with side of length $L = n_c h$, $\mathrm{d}(x,\Gamma)$ is the distance function to the interface, $\epsilon = h$ is a smoothing width used to avoid discontinuous jumps in the coefficients, and R is a random number uniformly distributed in $(0,1)$. Here we focus on the case of no bulk viscous term; we have also done tests including a bulk viscous term and observed similar behavior. Unless otherwise indicated, the bubble test is steady-state ($\theta = 0$), and we use a contrast ratio $r_\mu = r_\rho = 100$; we have observed similar behavior with a ratio of 1000, at least for sufficiently smooth jumps (i.e., sufficiently large ϵ). It is important to note that the types of preconditioners we use here have been shown to effective even with much larger viscosity contrasts ($\sim 10^6$) in the context of geophysical flow problems [8, 24, 25]. For the target applications we have in mind, such as phase-field models of fluid mixtures, viscosity contrasts of $10^2 - 10^3$ are more relevant (for example, the viscosity ratio of water and air is only 55, while the density ratio is 1000).

4.1　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.

4.1.1　The effect of the 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, and, more importantly, makes the smoother highly parallelizable. 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. 1 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 part dominated by smoother, henceforth we use two applications of the smoother at each level of the multigrid hierarchy in each V cycle.

The speed of convergence of the multigrid iteration for the component solvers is the standard against which one should measure convergence of the Krylov solver for the Stokes problem. As we can see in Fig. 1, each V cycle reduces the residual by at least an order of magnitude, 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 then 10 orders of magnitude should be considered excellent.

4.1.2　The effect of the number of multigrid cycles

As detailed in Appendix B, for constant-coefficient Stokes problems with periodic boundaries, GMRES will converge in a single iteration with preconditioner P_1 and in two iterations with P_2 and P_3. The same holds for any choice of boundary condition for the time-dependent case in the inviscid limit. 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
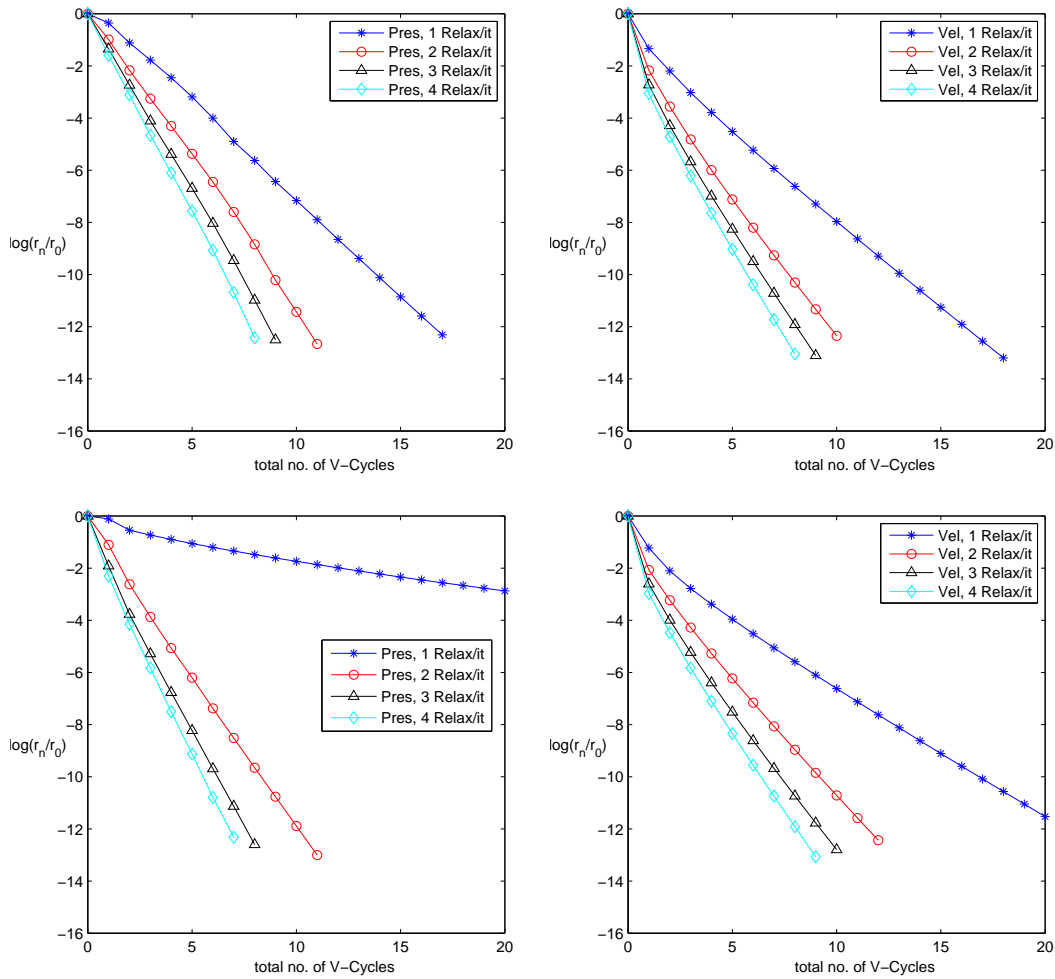
Figure 1: The log of the relative residual for the pressure (left) and velocity (right) multigrid solvers as a function of the number of multigrid V cycles, for different numbers of smoothing (relaxation) sweeps. A constant coefficient steady Stokes problem is solved on a 512^2 grid in two dimensions (top panels), and 128^3 grid in three dimensions (bottom panels), with no-slip conditions at all domain boundaries.

preconditioner. We use the preconditioner P_1 for these tests; similar results are observed for all of the preconditioners.

In the left panels of Fig. 2 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
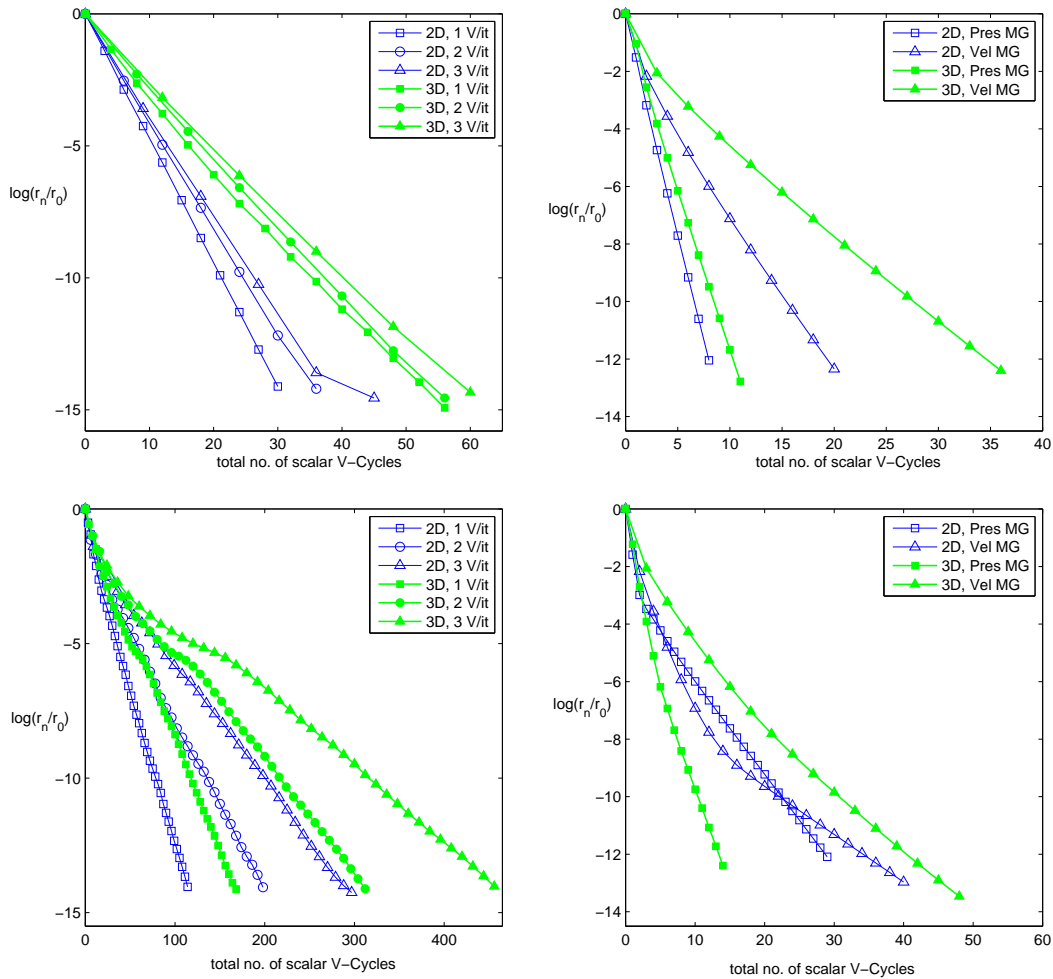
Figure 2: The relative residual (on a log scale) as a function of the total number of scalar multigrid V cycles, for different number of multigrid cycles per application of the pressure and velocity subsolvers in preconditioner P_1. GMRES convergence is shown in the left panels, and pressure (squares) 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.

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 serve as a reference point for what one may expect for a projection-like split solver.

The top left panel in Fig. 2 shows that for periodic constant coefficient problems there is no significant difference between using an exact subsolver (in effect, 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 because the standard multigrid algorithm takes the form of a simple Richardson iterative solver, and we expect GMRES to perform at least as well as Richardson iteration. 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 multigrid alone, see for example the discussion in Ref. [24].

In the bottom left panel in Fig. 2 we show the convergence of GMRES for the variable-coefficient bubble problem, which is typical of the behavior we observe when there are non-periodic boundary conditions or variable coefficients. Similar behavior is observed for the other preconditioners (not shown). The results clearly demonstrate that when using exact subsolvers in the preconditioner does not yield an exact solver for the Stokes problem, 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, except for nearly inviscid problems where performing more than one pressure cycle may be somewhat beneficial. By comparing the left and right panels, 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 ~ 170 cycles for coupled solver as seen in the left panel).

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

4.2 Comparison of preconditioners

Having empirically 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.

4.2.1 The effect of the sign of Schur complement

In the literature [34, 39], the following Schur complement based preconditioners have been proposed and studied,

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

where the sign of the lower diagonal block can be either positive or negative. It was proven that $T_+ = P_+^{-1}M$ satisfies $(T_+ - I)(T_+ + I) = 0$ and $T_- = P_-^{-1}M$ satisfies $(T_- - I)^2 = 0$ [34, 39]. If a block-diagonal preconditioner such as P_4 is used, by changing the sign it is

possible to either make the preconditioned operator symmetric but indefinite (allowing the use of methods such as MINRES), or non-symmetric but positive semi-definite [23].

Because the GMRES method possesses a Galerkin property [16], the total number of GMRES iterations is equal to the degree of the characteristic polynomials of the preconditioned systems. Therefore, using both $P_+^{-1}M$ and $P_-^{-1}M$, GMRES converges in 2 iterations if the inverses of A and the Schur complement are exact [39]. However, when inexact subsolvers are employed, we observe significant difference between the two choices of the sign of the Schur complement. Our numerical results (not shown) indicate that the preconditioners with "-" sign in front of Schur complement give almost twice faster GMRES convergence than those with "+" sign. This is consistent with our original choice in Eqs. (2.13) and (2.15).

4.2.2 The effect 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. 3 we compare the behavior of P_1, P_2 and P_3 for restart intervals of 5 or 10 GMRES iterations. In the left panel of the figure we show the behavior for an inviscid time-dependent bubble test 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 bubble 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. 3 we see that the performance of P_3 significantly deteriorates for the small restart interval for the inviscid problem. In the right panel of the figure we see some deterioration of the convergence for the small restart interval for all three preconditioners.

Based on these results, henceforth we use a restart interval of 10 iterations.

4.2.3 Comparisons of different preconditioners

Having empirically 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 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 robust convergence but is less efficient for the steady flow case because it requires one additional 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 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.
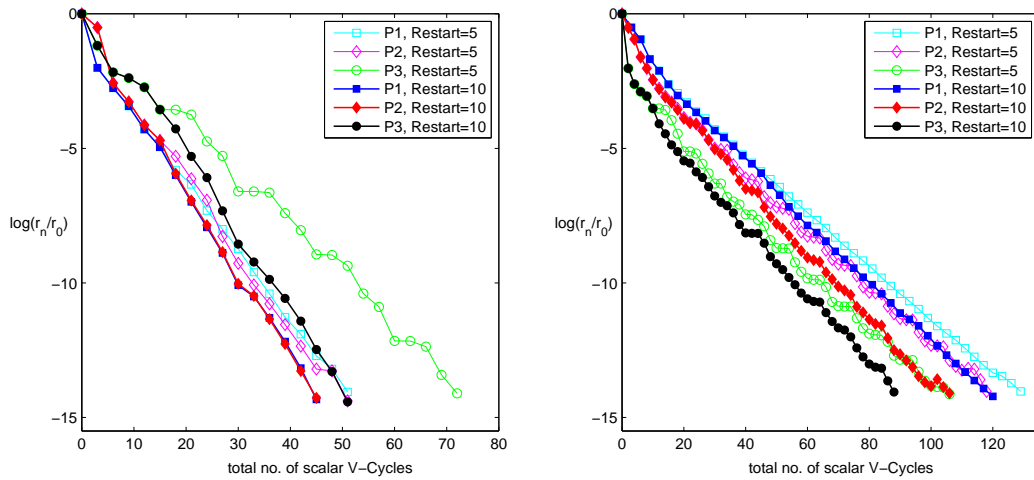
Figure 3: The GMRES convergence history for preconditioners P_1, P_2 and P_3 in two dimensions (512^2 grid) for the bubble test problem, for GMRES restart frequency 5 (open symbols) and 10 (filled symbols). In the left panel we set the viscosity to zero (unsteady inviscid flow) and in the right panel we set density to zero (steady viscous flow).
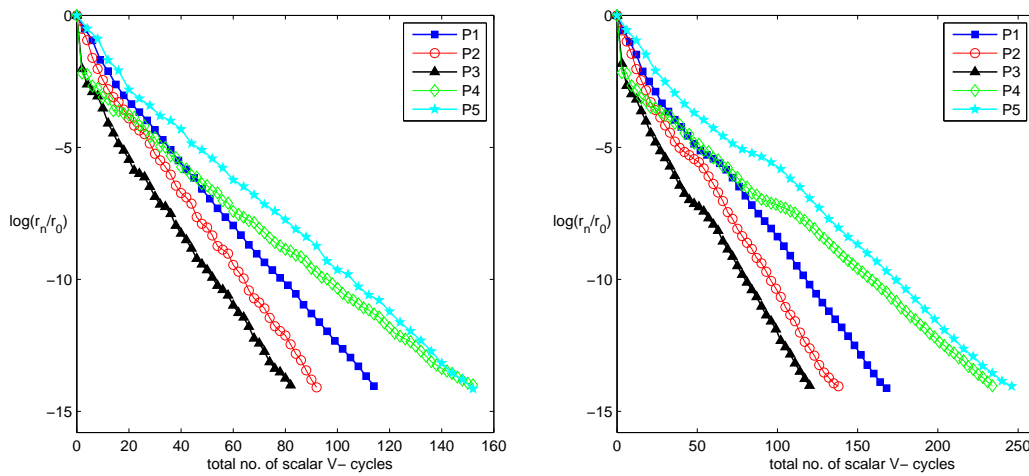

Figure 4: The GMRES convergence history for preconditioners P_1, P_2, P_3, P_4 and P_5 in two (left panel, 512^2 grid) and three (right panel, 128^3 grid) dimensions, for the bubble test problem. The restart frequency is 10 GMRES iterations.

Based on these observations, henceforth we do not consider P_4 and P_5. Since we find very similar behavior between P_2 and P_3, while P_3 shows poorer behavior with frequent restarts, henceforth we focus on examining in more detail the performance of P_1 and P_2.

4.3 Robustness

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

4.3.1 Effects of viscous CFL 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 $\beta > n_c^2$ indicates a viscous-dominated problem, where n_c is the grid size, with the hardest case being a steady-state problem $\beta \to \infty$. In Fig. 5 we study the performance of the GMRES Stokes solver for varying viscous CFL numbers for the bubble test problem, in both two and three dimensions, for both preconditioners P_1 and P_2. As expected, we see most rapid convergence for $\beta = 0$, and slowest convergence for $\beta \to \infty$. For the steady state case $\theta = 0$, we do not need a pressure Poisson solve for P_2 and therefore this preconditioner is somewhat more efficient than P_1. For intermediate β's we get somewhat better convergence for P_1, although the difference is small. In our experience both preconditioners show rather robust behavior for varying
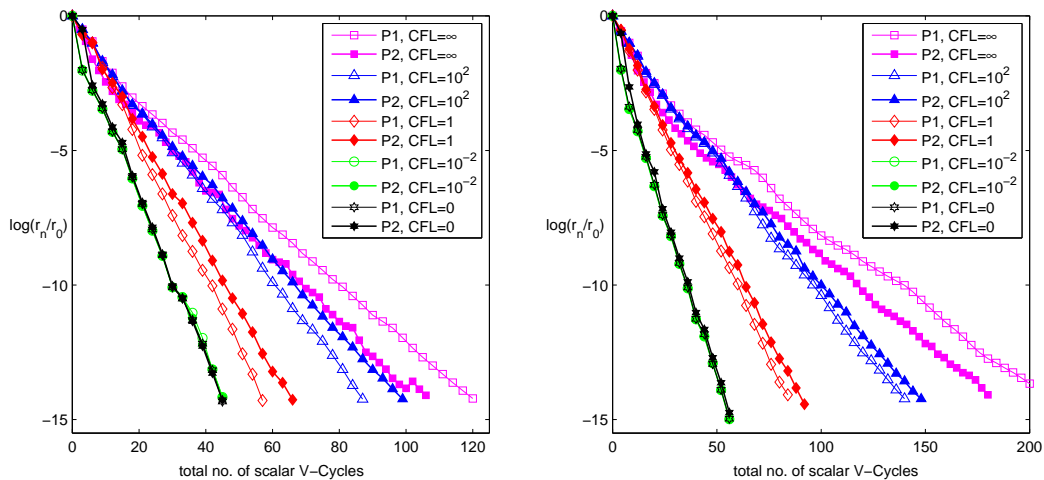

Figure 5: The GMRES convergence history for preconditioners P_1 (empty symbols) and P_2 (filled symbols) in two dimensions (left panel, 512^2 grid) and in three dimensions (right panel, 128^3 grid) for the bubble test problem. We vary θ to change the viscous CFL number β from the inviscid limit $\beta = 0$ to the steady limit $\beta \to \infty$.

viscous CFL number, viscosity and density contrast ratios, and different combinations of boundary conditions (periodic, free-slip, or no-slip).

4.3.2 Effects of problem size

An important goal in designing solvers suitable for large-scale calculations is to ensure that the total number of multigrid cycles remains essentially independent of the system size (or, equivalently, under grid refinement). In Fig. 6 we show convergence histories of GMRES for varying grid sizes for the steady state bubble problem in both two and three dimensions. In the left panels we show results for P_1 and the right panels for P_2. For this challenging variable-viscosity problem (recall that the viscosity and density con-
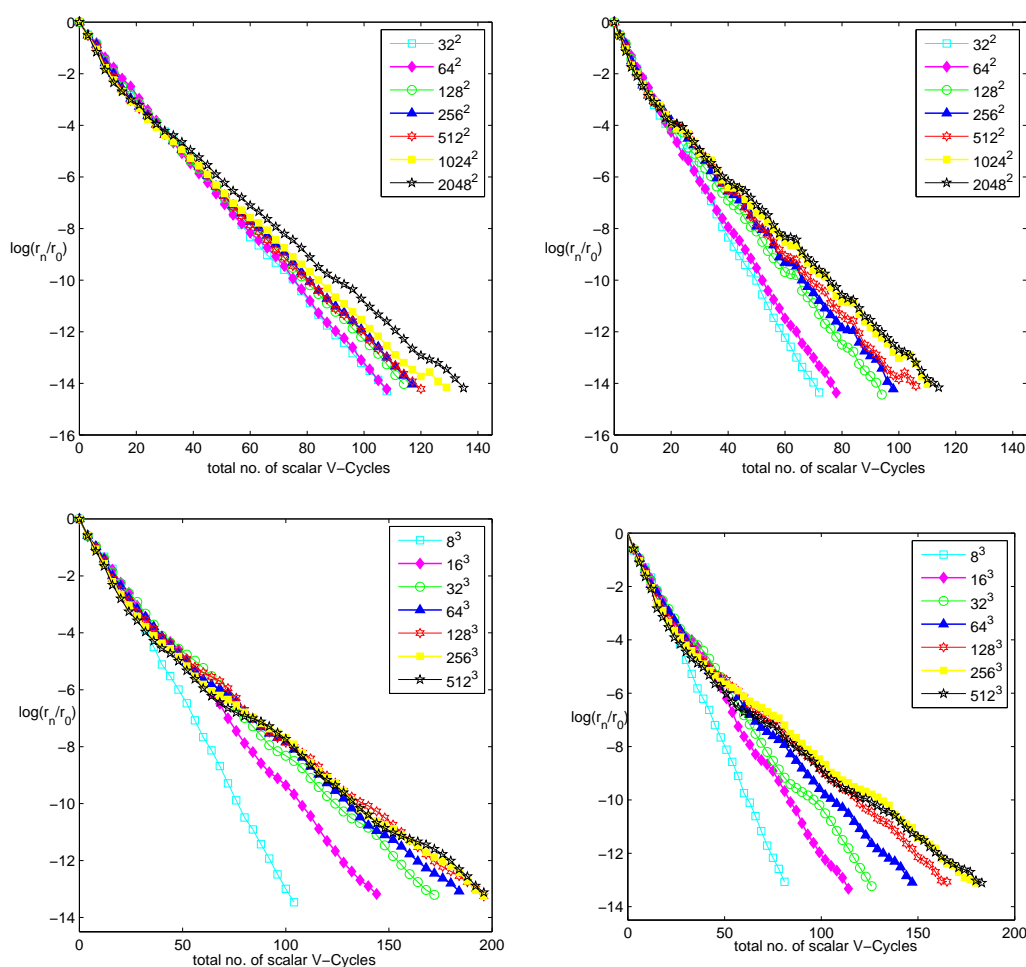

Figure 6: The GMRES convergence history for preconditioners P_1 (left panels) and P_2 (right panels) in two dimensions (top panels) and in three dimensions (bottom panels) for the steady-state bubble test problem with contrast ratio $r_\mu = r_\rho = 100$, as the grid size is varied.
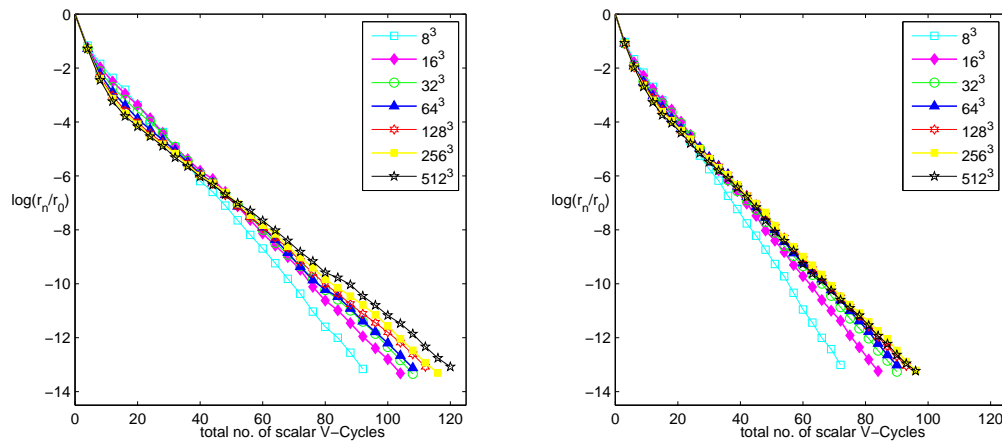
Figure 7: Same as bottom two panels of Fig. 6 but for contrast ratio $r_\mu = r_\rho = 2$.

trast ratio is $r_\mu = r_\rho = 100$), P_1 shows robust convergence for all of the grid sizes tested here in both two and three dimensions, requiring no more than 200 multigrid V cycles (i.e., no more than $200/4 = 50$ GMRES iterations) to reduce the residual to essentially roundoff tolerance even for a 512^3 grid. The convergence for preconditioner P_2 shows a very mild deterioration with increasing system size, although the overall efficiency is still somewhat higher than P_1 for all system sizes tested here. We have confirmed that making the subsolvers nearly exact does not aid the overall GMRES convergence, despite the substantial increase in the computational cost (data not shown).

It is important to point out that the exact convergence and its behavior on system size depends sensitively on the details of the multigrid algorithm (e.g., how the bottom level of the multigrid hierarchy is handled, which is typically affected by parallelization), the restart interval (here set to 10 iterations), and, most importantly, on the contrast ratio. In Fig. 7 we show scaling results in three dimensions for a much weaker contrast ratio $r_\mu = r_\rho = 2$. In this case we see little to no effect of the system size on the convergence rate, and the total number of GMRES iterations is less than 30.

5 Conclusions

We studied several preconditioners for solving time-dependent and steady discrete Stokes problems arising when solving fluid flow problems on a staggered finite-volume grid. All of the preconditioners we studied are based on approximating the inverse of the Schur complement with a simple local operator and have been proposed before, though often limited to either constant coefficient or steady flow. By suitably approximating the inverse of the Schur complement in the case of time-dependent variable-viscosity flow we were able to easily generalize these preconditioners and thus substantially enlarge

their practical applicability. Herein, we modified and extended a previously proposed projection-based preconditioner P_1 to variable-coefficient flows [27], we generalized a well-known lower triangular preconditioner P_2 to variable-coefficient flow, and we extended a previously-studied "fully coupled" solver with a "local viscosity" preconditioner [24] to time-dependent flows to obtain an upper triangular preconditioner P_3. The preconditioners investigated here can be generalized to other stable or stabilized spatial discretizations of the time-dependent Stokes equations, such as finite-element schemes or adaptive mesh finite-volume discretizations.

Our primary focus was on studying the performance of these preconditioners when the pressure and velocity subsolvers are performed on a uniform staggered grid using geometric multigrid algorithms. We showed that optimal convergence rates of the GMRES Stokes solver were obtained when a single multigrid V cycle is employed as an inexact subdomain solver. We numerically observed that all three preconditioners are effective for both time-dependent and steady flow problems, with the lower and upper triangular preconditioners being more efficient for steady problems and P_1 being somewhat more efficient for time-dependent problems, which is consistent with the findings of Ref. [27]. All three preconditioners were found to handle variable-coefficient problems rather well, with little deterioration in convergence from the case of constant-coefficient problems. Our observations are consistent with the conclusion of the authors of Ref. [24], who "find that it is advantageous to use the FC [fully-coupled] approach utilizing relaxed tolerances for solution of the sub-problems, combined with the LV [local viscosity] preconditioner."

All of our empirical observations are consistent with the general observation that solving the coupled Stokes problem is comparable to a single step of a fractional step method, and not more than 2-3 times more expensive than a fractional step even for difficult cases of large contrast ratio, large viscous CFL number and non-trivial boundary conditions. We believe that this mild increase in cost is more than justified given the important advantages of the coupled approach when solving the Navier-Stokes equations. Furthermore, we observed robust behavior of the projection and lower triangular preconditioners for large systems with relatively frequent restarts. This demonstrates that GMRES with preconditioners P_1 and P_2 provides a robust solver for large-scale computations. In future studies, the robustness of these preconditioners with respect to the variability of viscosity and density should be studied more carefully. One aspect of this is whether the spectrum of the preconditioned operator can be provably bounded for arbitrary contrast ratios. More importantly, however, the performance of the preconditioners in practical applications, should be accessed. Experience with steady Stokes geodynamics applications, which have extreme viscosity contrasts, are very promising [24].

Acknowledgments

A Fourier analysis of Schur complement

The most important element in the preconditioners we study here is the approximation of the Schur complement inverse. In previous work, Fourier analysis [9], operator mapping properties and PDE theory in [37, 38], and commutator properties (2.4) [17, 27, 35] have been used to justify approximations to the Schur complement inverse. Here we use Fourier analysis to justify our approximation to the Schur complement inverse for the stress form of the viscous operator (3.1). This analysis assumes periodic boundaries but should also inform the case with physical boundary conditions.

For simplicity, we use two dimensional steady state Stokes equations for illustration but extensions to three dimensions and time-dependent flow are trivial. We denote the discrete Fourier transform of velocity as $\hat{v} = [\hat{u}, \hat{v}]^T$, and denote the (purely imaginary) Fourier symbol of the staggered divergence operator as $\hat{D} = [\hat{D}_x \ \hat{D}_y]$, where D_x and D_y represent the staggered finite difference operator along the x and y axes. The Fourier transform of the staggered gradient operator is $\hat{G} = -\hat{D}^\star = [\hat{D}_x \ \hat{D}_y]^T$, and similarly,

$$\hat{L}_p = \hat{D}\hat{G} = \hat{D}_x^2 + \hat{D}_y^2 = -\left(|\hat{D}_x|^2 + |\hat{D}_y|^2 \right).$$

Our goal is to approximate the Schur complement inverse with a Laplacian-like local operator L_S, i.e., to find $(DL_\mu^{-1}G)^{-1} = L_S$. This is only an approximation in general but should be exact for periodic constant-coefficient problems. In Fourier space,

$$\hat{L}_S = \left(\hat{D}\hat{L}_\mu^{-1}\hat{G} \right)^{-1}. \tag{A.1}$$

When the Laplacian form of the viscous term is used, $L_\mu = \mu_0 L$, we have

$$\hat{L}_\mu = \mu_0 \begin{bmatrix} \hat{D}_x^2 + \hat{D}_y^2 & 0 \\ 0 & \hat{D}_x^2 + \hat{D}_y^2 \end{bmatrix},$$

which combined with (A.1) gives $\hat{L}_S = \mu_0$. Applying an inverse Fourier transform gives the well-known result $L_S = \mu_0 I$.

When L_μ is the discrete operator for the stress tensor form of the viscous term (3.1) and the viscosity is constant, we have

$$\hat{L}_\mu = \mu_0 \begin{bmatrix} 2\hat{D}_x^2 + \hat{D}_y^2 & \hat{D}_y\hat{D}_x \\ \hat{D}_x\hat{D}_y & \hat{D}_x^2 + 2\hat{D}_y^2 \end{bmatrix},$$

which gives $\hat{L}_S = 2\mu_0$, and therefore $L_S = 2\mu_0 I$. This motivates our variable-viscosity generalization (2.6). When L_μ is the discrete operator for the viscous term with bulk viscosity and assuming both the shear viscosity and the bulk viscosities are constant, we have

$$\hat{L}_\mu = \begin{bmatrix} (\frac{4}{3}\mu_0 + \gamma_0)\hat{D}_x^2 + \mu_0\hat{D}_y^2 & (\frac{1}{3}\mu_0 + \gamma_0)\hat{D}_y\hat{D}_x \\ (\frac{1}{3}\mu_0 + \gamma_0)\hat{D}_x\hat{D}_y & \gamma_0\hat{D}_x^2 + (\frac{4}{3}\mu_0 + \gamma_0)\hat{D}_y^2 \end{bmatrix},$$

which gives $\hat{L}_S = ((4/3)\mu_0 + \gamma_0)$ and therefore $L_S = ((4/3)\mu_0 + \gamma_0)I$. This motivates our variable-viscosity generalization (2.7).

B Analysis of preconditioners with exact subsolvers

In this Appendix we give some analysis of the spectrum of the preconditioned operators when exact pressure and velocity subsolvers are used. To see how well the different preconditioners approximate the original saddle point form (2.1), we formally calculate

$$P_1^{-1}M = \begin{pmatrix} I & (I - \rho^{-1}GL_\rho^{-1}D)A^{-1}G \\ 0 & \mathcal{S}^{-1}S \end{pmatrix}, \tag{B.1}$$

$$P_2^{-1}M = \begin{pmatrix} I & A^{-1}G \\ 0 & \mathcal{S}^{-1}S \end{pmatrix}, \tag{B.2}$$

$$P_3^{-1}M = \begin{pmatrix} I - A^{-1}G\mathcal{S}^{-1}D & A^{-1}G \\ \mathcal{S}^{-1}D & 0 \end{pmatrix}, \tag{B.3}$$

and lastly

$$P_4^{-1}M = \begin{pmatrix} I & A^{-1}G \\ \mathcal{S}^{-1}D & 0 \end{pmatrix}. \tag{B.4}$$

Recall that for constant-coefficient problems with exact subsolvers, $\mathcal{S}^{-1} = -\theta\rho_0 L_p^{-1} + \mu_0 I$. For periodic domains, the finite-difference operators G, D, L and L_p commute,

$$GL_p = LG \quad \text{and} \quad L_pD = DL, \tag{B.5}$$

and therefore $P_1^{-1}M$ is exactly the discrete identity operator, and similarly, the (1,1) diagonal block of $P_3^{-1}M$ is zero.

From (B.1), we see that the preconditioned system is block upper triangular. Therefore, the eigenvalues of the preconditioned system are either unity or the eigenvalues of $\mathcal{S}^{-1}S$. Similarly, we can derive the eigenvalues of the preconditioned system using (B.2) and (B.3). Alternatively, one can write down the generalized eigenvalue system, for instance,

$$M \begin{pmatrix} u \\ p \end{pmatrix} = \lambda P_3 \begin{pmatrix} u \\ p \end{pmatrix}.$$

Again, one can see that the eigenvalues are either 1 or the eigenvalues of $\mathcal{S}^{-1}S$.

When $\mu_0=0$, or equivalently, $\Delta t\to 0$, we have that $P_1^{-1}M=I$, regardless of the boundary conditions. If P_2 is used, we have

$$P_2^{-1}M=\begin{pmatrix} I & \frac{\Delta t}{\rho_0}G \\ 0 & I \end{pmatrix}, \tag{B.6}$$

and therefore $(P_2^{-1}M-I)^2=0$. This proves that in the inviscid case, the GMRES algorithm converges in 1 iteration when preconditioner P_1 is used, and in 2 iterations when P_2 or P_3 are used. When inexact subsolvers are used our numerical results showed that all three preconditioners exhibit exactly the same convergence rate in the inviscid case.

Furthermore, for constant viscosity ($\mu=1$) steady state ($\theta=0$) problems on a two-dimensional domain of $n_x\times n_y$ grid cells with no-slip boundaries, one can prove the following property for the eigenvalues of the Schur complement $S=DL^{-1}G$:

1. $\lambda(S)\in\{0\}\cup[\eta^2,1]$, where η is the inf-sup constant independent of grid size [20,41].
2. The multiplicity of the 0 eigenvalue is 1.
3. There are at most $2(n_x-1)+2(n_y-1)$ non-unit eigenvalues of S.

This is a quantitative statement of the intuitive expectation that a few cells away from the boundaries S is close to an identity operator, just as for a periodic system (see Eq. (B.5)). The lower bound of the eigenvalues is a consequence of the uniform div-stability [19,20,51]. From (B.1) (and also (B.2) and (B.3)), we see that the same conclusions hold for the preconditioned systems. This analysis explains the good performance of the simple Schur complement approximation even in the presence of nontrivial boundary conditions [27]. Below we numerically compute the spectrum of the eigenvalues of $\mathcal{S}^{-1}S$ for both constant and variable viscosity steady flow, and confirm the theoretical predictions above.

B.1 Spectrum of the 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, it is generally accepted and widely observed that having closely clustered eigenvalues of the preconditioned operator $P^{-1}M$ 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 n_c^2 cells with four no-slip boundaries. We consider using exact subdomain solvers, $\widetilde{A}^{-1}=A^{-1}$ and $\widetilde{L}_\rho^{-1}=L_\rho^{-1}$ instead of multigrid, relying on the fact that a well-designed multigrid
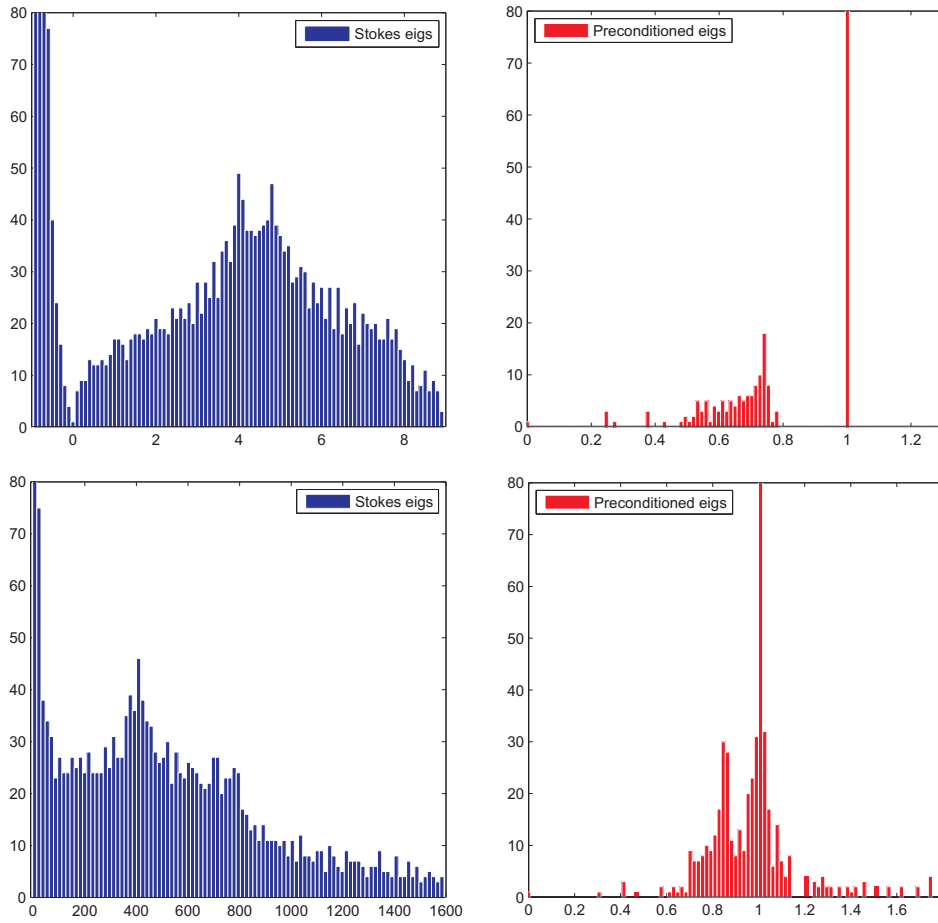
Figure 8: Histogram of the eigenvalues of M (left panels) and $P^{-1}M$ (more precisely, $\mathcal{S}^{-1}S$) (right panels) for a steady Stokes problem on a grid of 32×32 cells with four no-slip boundaries. The vertical axis gives the number of eigenvalues, truncated for the eigenvalue $\lambda = 1$ 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.

cycle is (essentially) spectrally-equivalent to an exact solver. We explicitly form M and $\mathcal{S}^{-1}S$ in MATLAB as dense matrices and compute their eigenvalues.

In Fig. 8, we show a histogram of the eigenvalues of the unpreconditioned and preconditioned operators for a square domain of length $n_c = 32$ cells with no-slip boundaries. In the upper row in Fig. 8 we study the constant viscosity case. The total number of DOFs is $N_{\text{dof}} = n_c^2 + 2n_c(n_c - 1) = 3008$. Since the original Stokes system is of saddle point type, M has both positive eigenvalues and negative eigenvalues, and there are $n_c^2 = 1024$ 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 $4(n_c - 1)/N_{\text{dof}} \approx 4\%$ nonzero eigenvalues remaining well-clustered.

In the lower row in Fig. 8 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. By contrast, 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. [29,30] to prove that $\mathcal{S}^{-1}S$ is spectrally-equivalent to the identity matrix for the staggered grid discretization we employ here.

C   Multigrid algorithms

We employ a standard V-cycle multigrid approach [6] for both the cell-centered multigrid subsolver for the weighted Poisson operator L_ρ and the staggered velocity multigrid subsolver for the viscous operator L_μ. We use the standard residual formulation, so that on all coarsened levels we are solving for the error in the coarsened residual from the next-finer level. In our implementation, the multigrid coarsening factor is 2, and coarsening continues until the problem domain represented on the coarsest grid contains two grid points (with respect to cell-centers) in any given spatial direction. At the coarsest level of the multigrid hierarchy, we perform a large (8 or more) number of relaxations, to ensure that the preconditioner is a constant linear operator.

Multigrid consists of 3 major components: (i) choice of relaxation at a particular level, (ii) coarsening/restriction operator, and (iii) interpolation/prolongation operator.

Relaxation. Both the staggered and cell-centered solvers use multicolored Gauss-Seidel smoothing. The cell-centered solver uses standard red-black relaxation, whereas the staggered solver uses a $2d$-colored relaxation, where d is the dimensionality of the problem. Because the coupling between the degrees of freedom corresponding to a given component of velocity is the same as for the cell-centered Poisson equation, by coloring each component of the velocity separately, as in the standard red-black coloring (i.e., coloring odd grid points with a different color from the even grid points), we obtain decoupling between the $2d$ colors so that each color can be relaxed separately (improving convergence and aiding parallelization). We relax the components of velocity in turn (i.e., in three dimensions, we order the relaxations as red-x, black-x, red-y, black-y, red-z, black-z), although other orderings of the colors are possible. Refer to Fig. 9 for a physical representation of the viscous operator stencils. Given a cell-centered operator of the form, $\nabla \cdot \beta \nabla \phi \equiv \mathcal{L}\phi = r$, or a staggered operator of the form, $\alpha\phi - \nabla \cdot \beta[\nabla\phi + (\nabla\phi)^T] \equiv \mathcal{L}\phi = r$, the relaxation takes the form

$$\phi^{k+1} = \phi^k + \omega\mathcal{D}^{-1}(r - \mathcal{L}\phi^k), \tag{C.1}$$

for each color in turn, where the superscript represents the iterate, and $\mathcal{D}^{-1}$ is the inverse
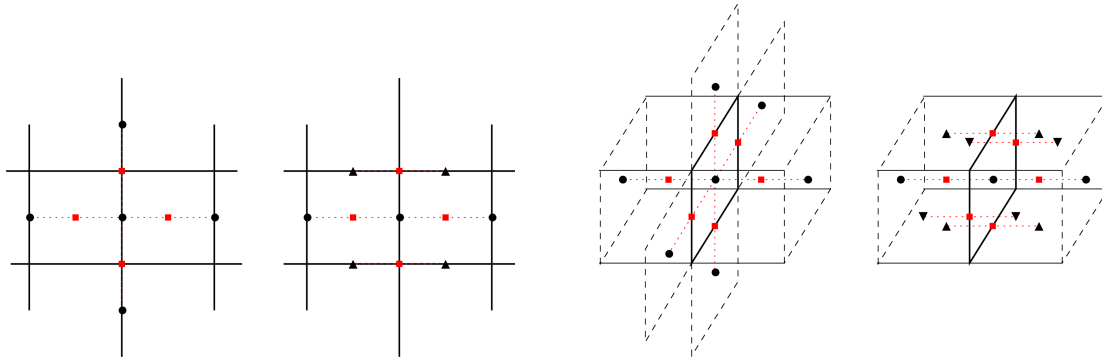
Figure 9: (*Left panel*) The stencils for the x-component of $\nabla \cdot \beta \nabla \phi$ (left) and $\nabla \cdot \beta (\nabla \phi)^T$ (right), in two dimensions. The black circles indicate locations of u. The black triangles indicate locations of v. The red dots indicate the location of the β and the gradients of velocity. (*Right panel*) The stencils for the x-component of $\nabla \cdot \beta \nabla v$ (left) and $\nabla \cdot \beta (\nabla v)^T$ (right), in three dimensions. The black circles indicate locations of u. The black triangles indicate locations of v and w. The red dots indicate the location of the β and the gradients of velocity.

of the diagonal elements of $\mathcal{L}$. We use unit weighting factor[‡], $\omega = 1$ (suggested to be near-optimal in numerical experiments) for both subsolvers.

Restriction. For the cell-centered solver, restriction is a simple averaging of the 2^d fine cells. For the staggered solver, we use a slightly more complicated 6-point ($d = 2$) or 12-point ($d = 3$) stencil. For example, for x-faces we use

$$\phi_{i,j}^{c} = \frac{1}{8} \left(\phi_{2i-1,2j}^{f} + \phi_{2i-1,2j+1}^{f} + \phi_{2i+1,2j}^{f} + \phi_{2i+1,2j+1}^{f} \right) + \frac{1}{4} \left(\phi_{2i,2j}^{f} + \phi_{2i,2j+1}^{f} \right). \qquad \text{(C.2)}$$

As seen in Fig. 9, for the staggered solver we require α at faces, and β at both cell-centers and nodes ($d = 2$) or edges ($d = 3$). When creating coefficients at coarser levels, we obtain α by averaging the overlaying fine faces, cell-centered β by averaging the overlaying fine cell-centered values, β on nodes through direct injection, and β on edges by averaging the overlaying fine edges.

Prolongation. For the cell-centered solver, prolongation is simply direct injection from the coarse cell to the overlaying 2^d fine cells. For the staggered solver, we use a slightly more complicated stencil that involves linear interpolation for fine faces that overlay coarse faces, and bilinear interpolation for fine faces that do not overlay coarse faces. For example, for x-faces we use

$$\phi_{i,j}^{f} = \frac{3}{4} \phi_{i/2,j/2}^{c} + \frac{1}{4} \phi_{i/2,j/2-1}^{c}, \quad \text{for } i \text{ and } j \text{ both even,} \qquad \text{(C.3)}$$

$$\phi_{i,j}^{f} = \frac{3}{8} \left(\phi_{i/2,j/2}^{c} + \phi_{i/2+1,j/2}^{c} \right) + \frac{1}{8} \left(\phi_{i/2,j/2-1}^{c} + \phi_{i/2+1,j/2-1}^{c} \right), \quad \text{for } i \text{ odd and } j \text{ even,} \qquad \text{(C.4)}$$

where we use integer division in the index subscripts.

[‡]Note that for Jacobi relaxation with the stress form of the viscous operator, a standard analysis suggests $\omega = 1/2$ as the optimal relaxation parameter (ensuring damping of all modes).

References

[1] A. S. Almgren, J. B. Bell, P. Colella, L. H. Howell, and M. L. Welcome, A conservative adaptive projection method for the variable density incompressible Navier-Stokes equations, Journal of Computational Physics, 142 (1998), pp. 1–46.

[2] A. S. Almgren, J. B. Bell, and W. G. Szymczak, A numerical method for the incompressible Navier-Stokes equations based on an approximate projection, SIAM Journal on Scientific Computing, 17 (1996), pp. 358–369.

[3] J. B. Bell, P. Colella, and H. M. Glaz, A second order projection method for the incompressible Navier-Stokes equations, Journal of Computational Physics, 85 (1989), pp. 257–283.

[4] M. Benzi, Preconditioning techniques for large linear systems: A survey, Journal of Computational Physics, 182 (2002), pp. 418–477.

[5] M. Benzi, G. H. Golub, and J. Liesen, Numerical solution of saddle point problems, Acta Numerica, 14 (2005), pp. 1–137.

[6] W. L. Briggs, V. Henson, and S. McCormick, A Multigrid Tutorial Society for Industrial and Applied Mathematics, Philadelphia, PA, (1987).

[7] D. L. Brown, R. Cortez, and M. L. Minion, Accurate projection methods for the incompressible Navier-Stokes equations, Journal of Computational Physics, 168 (2001), pp. 464–499.

[8] C. Burstedde, O. Ghattas, G. Stadler, T. Tu, and L. C. Wilcox, Parallel scalable adjoint-based adaptive solution of variable-viscosity stokes flow problems, Computer Methods in Applied Mechanics and Engineering, 198 (2009), pp. 1691–1700.

[9] J. Cahouet and J.-P. Chabard, Some fast 3D finite element solvers for the generalized stokes problem, International Journal for Numerical Methods in Fluids, 8 (1988), pp. 869–895.

[10] Z.-H. Cao, Constraint Schur complement preconditioners for nonsymmetric saddle point problems, Applied Numerical Mathematics, 59 (2009), pp. 151–169.

[11] A. J. Chorin, Numerical solution of the Navier-Stokes equations, Journal of Computational Mathematics, 22 (1968), pp. 745–762.

[12] S. Delong, B. E. Griffith, E. Vanden-Eijnden, and A. Donev, Temporal integrators for fluctuating hydrodynamics, Physical Review E, 87 (2013), p. 033302.

[13] A. Donev, T. G. Fai, and E. Vanden-Eijnden, A reversible mesoscopic model of diffusion in liquids: From giant fluctuations to Fick's law, Journal of Statistical Mechanics: Theory and Experiment, 2014 (2014), p. P04004.

[14] A. Donev, A. J. Nonaka, Y. Sun, T. G. Fai, A. L. Garcia, and J. B. Bell, Low mach number fluctuating hydrodynamics of diffusively mixing fluids, Communications in Applied Mathematics and Computational Science, 9 (2014), pp. 47–105.

[15] W. E and J. Liu, Gauge method for viscous incompressible flows, Communications in Mathematical Sciences, 1 (2003), pp. 317–332.

[16] M. Eiermann and O. G. Ernst, Geometric aspects of the theory of Krylov subspace methods, Acta Numerica, 10 (2001), pp. 251–312.

[17] H. Elman, V. E. Howle, J. Shadid, R. Shuttleworth, and R. Tuminaro, Block preconditioners based on approximate commutators, SIAM Journal on Scientific Computing, 27 (2006), pp. 1651–1668.

[18] ——, A taxonomy and comparison of parallel block multi-level preconditioners for the incompressible Navier-Stokes equations, Journal of Computational Physics, 227 (2008), pp. 1790–1808.

[19] H. C. Elman, Preconditioning for the steady-state Navier-Stokes equations with low viscosity, SIAM Journal on Scientific Computing, 20 (1999), pp. 1299–1316.

[20] H. C. Elman, D. J. Silvester, and A. J. Wathen, Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics: with Applications in Incompressible Fluid Dynamics, OUP Oxford, 2005.

[21] X. Feng and Y. He, Modified homotopy perturbation method for solving the Stokes equations, Computers & Mathematics with Applications, 61 (2011), pp. 2262–2266.

[22] X. Feng and L. Shao, On the generalized Sor-like methods for saddle point problems, Journal of Applied Mathematics and Informatics, 28 (2010), pp. 663–677.

[23] B. Fischer, A. Ramage, D. J. Silvester, and A. J. Wathen, Minimum residual methods for augmented systems, BIT Numerical Mathematics, 38 (1998), pp. 527–543.

[24] M. Furuichi, D. A. May, and P. J. Tackley, Development of a stokes flow solver robust to large viscosity jumps using a Schur complement approach with mixed precision arithmetic, Journal of Computational Physics, 230 (2011), pp. 8835–8851.

[25] T. Geenen, C. Vuik, G. Segal, S. MacLachlan, et al., On iterative methods for the incompressible Stokes problem, International Journal for Numerical Methods in Fluids, 65 (2011), pp. 1180–1200.

[26] T. V. Gerya, D. A. May, and T. Duretz, An adaptive staggered grid finite difference method for modeling geodynamic Stokes flows with strongly variable viscosity, Geochemistry, Geophysics, Geosystems, 14 (2013), pp. 1200–1225.

[27] B. Griffith, An accurate and efficient method for the incompressible Navier-Stokes equations using the projection method as a preconditioner, Journal of Computational Physics, 228 (2009), pp. 7565–7595.

[28] B. Griffith, Immersed boundary model of aortic heart valve dynamics with physiological driving and loading conditions, International Journal for Numerical Methods in Biomedical Engineering, 28 (2012), pp. 317–345.

[29] P. Grinevich, An iterative solution method for the stokes problem with variable viscosity, Moscow University Mathematics Bulletin, 65 (2010), pp. 119–122.

[30] P. Grinevich and M. Olshanskii, An iterative method for the Stokes-type problem with variable viscosity, SIAM Journal on Scientific Computing, 31 (2009), pp. 3959–3978.

[31] J. Guermond, P. Minev, and J. Shen, An overview of projection methods for incompressible flows, Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 6011–6045.

[32] F. Harlow and J. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluids with free surfaces, Physics of Fluids, 8 (1965), pp. 2182–2189.

[33] Q. Hu and J. Zou, Nonlinear inexact Uzawa algorithms for linear and nonlinear saddle-point problems, SIAM Journal on Optimization, 16 (2006), pp. 798–825.

[34] I. C. Ipsen, A note on preconditioning nonsymmetric matrices, SIAM Journal on Scientific Computing, 23 (2001), pp. 1050–1051.

[35] D. Kay, D. Loghin, and A. Wathen, A preconditioner for the steady-state Navier-Stokes equations, SIAM Journal on Scientific Computing, 24 (2002), pp. 237–256.

[36] D. A. Kay, P. M. Gresho, D. F. Griffiths, and D. J. Silvester, Adaptive time-stepping for incompressible flow Part II: Navier-Stokes equations, SIAM Journal on Scientific Computing, 32 (2010), pp. 111–128.

[37] K.-A. Mardal and R. Winther, Uniform preconditioners for the time dependent stokes problem, Numerische Mathematik, 98 (2004), pp. 305–327.

[38] ——, Preconditioning discretizations of systems of partial differential equations, Numerical Linear Algebra with Applications, 18 (2011), pp. 1–40.

[39] M. F. Murphy, G. H. Golub, and A. J. Wathen, A note on preconditioning for indefinite linear

　　　　systems, SIAM Journal on Scientific Computing, 21 (2000), pp. 1969–1972.

[40] M. Olshanskii, Multigrid analysis for the time dependent stokes problem, Mathematics of Computation, 81 (2012), pp. 57–79.

[41] M. A. Olshanskii and E. V. Chizhonkov, On the best constant in the inf-sup-condition for elongated rectangular domains, Mathematical Notes, 67 (2000), pp. 325–332.

[42] M. A. Olshanskii, J. Peters, and A. Reusken, Uniform preconditioners for a parameter dependent saddle point problem with application to generalized stokes interface equations, Numerische Mathematik, 105 (2006), pp. 159–191.

[43] R. Pember, L. Howell, J. Bell, P. Colella, W. Crutchfield, W. Fiveland, and J. Jessee, An adaptive projection method for unsteady, low-Mach number combustion, Combustion Science and Technology, 140 (1998), pp. 123–168.

[44] A. Quarteroni, F. Saleri, and A. Veneziani, Factorization methods for the numerical approximation of Navier-Stokes equations, Computer Methods in Applied Mechanics and Engineering, 188 (2000), pp. 505–526.

[45] C. Rendleman, V. Beckner, M. Lijewski, W. Crutchfield, and J. Bell, Parallelization of structured, hierarchical adaptive mesh refinement algorithms, Computing and Visualization in Science, 3 (2000), pp. 147–157. Software available at `https://ccse.lbl.gov/BoxLib`.

[46] Y. Saad, A flexible inner-outer preconditioned GMRES algorithm, SIAM Journal on Scientific Computing, 14 (1993), pp. 461–469.

[47] Y. Saad and M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing, 7 (1986), pp. 856–869.

[48] D. Shin and J. C. Strikwerda, Inf-sup conditions for finite-difference approximations of the Stokes equations, Journal of the Australian Mathematical Society-Series B, 39 (1997), pp. 121–134.

[49] S. Turek, Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computional Approach, vol. 6, Springer Verlag, 1999.

[50] F. B. Usabiaga, J. B. Bell, R. Delgado-Buscalioni, A. Donev, T. G. Fai, B. E. Griffith, and C. S. Peskin, Staggered schemes for fluctuating hydrodynamics, SIAM Journal of Multiscale Modeling and Simulation, 10 (2012), pp. 1369–1408.

[51] R. Verfürth, A multilevel algorithm for mixed problems, SIAM Journal on Numerical Analysis, 21 (1984), pp. 264–271.