Preconditioning mixed finite elements for tide models

Robert C. Kirby[a,*], Tate Kernell[a]

[a] *Department of Mathematics, Baylor University, One Bear Place #97328, Waco, TX 76798-7328, United States*

Abstract

We describe a fully discrete mixed finite element method for the linearized rotating shallow water model, possibly with damping. While Crank-Nicolson time-stepping conserves energy in the absence of drag or forcing terms and is not subject to a CFL-like stability condition, it requires the inversion of a linear system at each step. We develop weighted-norm preconditioners for this algebraic system that are nearly robust with respect to the physical and discretization parameters in the system. Numerical experiments using Firedrake support the theoretical results.

Keywords: Block preconditioners, Finite element, Tide models
MSC[2010] 65N30, 65F08

1. Introduction

Accurate modeling of tides plays an important role in several disciplines. For example, geologists use tide models to help understand sediment transport and coastal flooding, while oceanographers study tides to discern mechanisms sustaining global circulation [1, 2]. Finite element methods making use of unstructured (typically triangular) meshes are attractive to handle irregular coastlines and topography [3]. In many situations, it is sufficient to use a linearized shallow water model with rotation and a parameterized drag term. In particular, the literature contains many papers [4, 5, 6, 7, 8, 9] studying mixed finite element pairs for discretization of each layer for ocean and atmosphere models, and we continue study of this case here.

Much of the literature relates to dispersion relations and enforcement of conservation principles by mixed methods, but our prior work in this area has been to focus on energy estimates. In [10], we gave a careful account of the effect of linear bottom friction in semidiscrete mixed methods, showing

[*] This work was supported by NSF 1912653.

[*] Corresponding author

Email addresses: robert_kirby@baylor.edu (Robert C. Kirby), otkernell@gmail.com (Tate Kernell)

that, absent forcing, one obtained exponential damping of a natural energy functional. This allowed estimates of long-time stability and optimal-order *a priori* error estimates. Then, we handled the (much more delicate) case of a broad family of nonlinear damping terms in [11]. In this case, the energy decay is sub-exponential (typically bounded by a power law) but still strong enough to admit long-time stability and error estimates.

While our work in [10, 11] focused on the semidiscrete mixed finite element case, we now turn to certain issues related to time-stepping. Crank-Nicolson time-stepping is second-order accurate, A-stable (not subject to CFL-like stability condition), and exactly energy conserving in the absence of forcing and damping. However, because it is implicit, it requires the solution of a system of algebraic equations at each time step. For linear damping models, this system is linear, but nonlinear otherwise. The point of this paper is to develop robust preconditioners for the linear system (or Jacobian of the nonlinear one) for use in conjunction with a Krylov method such as GMRES [12].

In addition to the mesh size and time step, our model also depends on a number of physical parameters, described in the following section. Our goal is to design a preconditioner that enables GMRES to converge with an overall iteration counts that depend as little as possible on these parameters. We follow the technique of using weighted-norm preconditioners [13]. Here, one designs an inner product with respect to which the variational problem is bounded with bounded inverse. Such bounds should depend weakly, if at all, on physical and discretization parameters.

The paper is organized as follows. We describe the particular tide model of interest and its discretization in Section 2. This includes Crank-Nicolson time-stepping and a comparison to a symplectic Euler method. Then, we turn to preconditioning the Crank-Nicolson system in Section 3. After analyzing a simple block-diagonal preconditioner with scaled mass matrices, we develop and analyze a parameter-weighted inner product on $H(\mathrm{div}) \times L^2$. Our estimate shows that the preconditioned system has an intrinsic time scale determined by the Rossby number that must be resolved by the time step. This does not seem to be a major practical constraint. After discussion and analysis of these preconditioners, we turn to numerical experiments validating the theory in Section 4 and draw some conclusions in Section 5.

2. Description of finite element tidal model

The nondimensional linearized rotating shallow water model with linear drag and forcing on a two dimensional surface Ω is given by

$$\begin{aligned} u_t + \frac{f}{\epsilon} u^{\perp} + \frac{\beta}{\epsilon^2} \nabla \left(\eta - \eta' \right) + C u = 0, \\ \eta_t + \nabla \cdot (H u) = 0. \end{aligned} \tag{1}$$

Here, the unknowns are u, which is the nondimensional velocity field tangent to Ω, and η is the nondimensional free surface elevation above the height at state of rest. The quantity $u^{\perp} = (-u_2, u_1)$ is just the velocity rotated by $\pi/2$. The system is driven by the quantity $\nabla \eta'$, which is the (spatially varying) tidal forcing. Several physical parameters also appear in the system. The Coriolis forces are given by f, a non-dimensional parameter which is equal to the sine of the latitude (which can be approximated by a linear or constant profile for local area models). The Rossby number, ϵ, measures the ratio of inertial to Coriolis forces and tends to be small but not singularly so for global tides ($\mathcal{O}(10^{-3})$ – $\mathcal{O}(10^{-1})$). The Burger number β measures the ratio of vertical forces arising from density stratification to horizontal rotational forces. In many applications it is $\mathcal{O}(1)$. C is the (spatially varying) nondimensional drag coefficient modeling bottom friction. H is the (spatially varying due to bathymetry of the ocean floor) nondimensional fluid depth at rest, and ∇ and $\nabla \cdot$ are the intrinsic gradient and divergence operators on Ω, respectively. On the boundary $\partial\Omega$, we specify no-flux boundary conditions $u \cdot n = 0$. We must also specify initial conditions $u(\cdot, 0) = u_0$ and zero-mean $\eta(\cdot, 0) = \eta_0$.

Prior energy and error analysis in [10] assumes that the bottom friction satisfies some $0 < C_* \leq C(\mathbf{x}) \leq C^*$. The strict lower bound allows one to show an exponential damping of the energy. However, even without the lower bound, the model is well-posed and, absent forcing, has non-increasing energy. Since we are not working with energy estimates, it is sufficient for us to merely assume the upper bound $C(\mathbf{x}) \leq C^*$. Also, we require bounds on the bathymetry of $0 < H_* \leq H(\mathbf{x}) < H^*$.

As in [10], we arrive at a form suitable for discretization by mixed methods by working with the linearized momentum $\widetilde{u} = Hu$ rather than velocity.

After making this substitution and dropping the tildes, we obtain

$$\begin{aligned} \frac{1}{H}u_t + \frac{f}{H\epsilon}u^\perp + \frac{\beta}{\epsilon^2}\nabla\eta + \frac{C}{H}u = F, \\ \eta_t + \nabla \cdot u = 0. \end{aligned} \tag{2}$$

Here, we introduce F as shorthand for the forcing term $F = \frac{\beta}{H\epsilon^2}\nabla\eta'$. A natural weak formulation of this equations is to seek $u \in H(\mathrm{div})$ and $\eta \in L^2$ so that

$$\begin{aligned} \left(\frac{1}{H}u_t, v\right) + \frac{1}{\epsilon}\left(\frac{f}{H}u^\perp, v\right) - \frac{\beta}{\epsilon^2}\left(\eta, \nabla \cdot v\right) + \left(\frac{C}{H}u, v\right) = \left(F, v\right), \\ \left(\eta_t, w\right) + \left(\nabla \cdot u, w\right) = 0 \end{aligned} \tag{3}$$

for all $v \in H(\mathrm{div})$ and $w \in L^2$.

We select suitable mixed finite element spaces $V_h \subset H(\mathrm{div})$ and $W_h \subset L^2$ of order k satisfying the commuting projection and having divergence mapping V_h onto W_h [14]. We will need to make use of the *inverse assumption* that there exists some C_I (typically depending on the polynomial degree and mesh shape but not element size) such that

$$\|\nabla \cdot u\| \leq \tfrac{C_I}{h}\|u\| \tag{4}$$

for all $u \in V_h$.

Our theory for preconditioners does not depend on the particular choice of finite element spaces, and so our numerical results will include various instances of triangular and rectangular elements. A classic element pair on triangular meshes uses the Raviart-Thomas elements [15] for V_h together with discontinuous piecewise polynomials for W_h. In this case, the space V_h consists of elements that, restricted to a given triangle K, live in the space

$$V^k(K) = P_{k-1}(K)^2 + \begin{bmatrix} x \\ y \end{bmatrix} P_{k-1}(K),$$

where $P_k(K)$ is the space of polynomials of degree k over K and $P_k(K)^2$ consists of all 2-vectors of such polynomials. The Raviart-Thomas space is the smallest possible space that the divergence can map onto $P_{k-1}(K)$. Here, we follow the ordering of the Periodic Table of Finite Elements [16] summarizing the Finite Element Exterior Calculus [17] instead of the original ordering of Raviart and Thomas – the lowest order RT space is indexed with

1 rather than zero. The global spaces consist of functions locally in $V^k(K)$ with continuous normal components between edges for V_h and discontinuous piecewise polynomials of degree $k-1$ for W_h.

Raviart-Thomas elements can be defined on rectangular domains in a similar way – choosing something the divergence maps onto tensor-products of polynomials. In this case, the local flux space consists of vectors whose x-components are tensor products of polynomials of degree k in the x variable with polynomials of degree $k-1$ in the y variable and vice versa for the y component. If such vectors, pieced together with continuous normal components, form the global V_h space, they are combined with discontinuous tensor-product polynomials of degree k on each cell for W_h.

On rectangular elements, using tensor products of degree k polynomials for W_h delivers accuracy of order $k+1$, but at a higher cost than strictly necessary. It is only necessary that W_h contain polynomials of total degree k for this accuracy to hold. Classically, the Brezzi-Douglas-Marini elements [18] do just this, taking V_h to have vectors of polynomials of total degree k suitably enriched to allow normal continuity. More recently, these elements have been interpreted in the context of serendipity differential forms, and related, but smaller spaces of "trimmed" serendipity forms are known [19]. We also use the second-order instance of these spaces. In the lowest order case, RT elements with piecewise constants give 4 + 1 degrees of freedom per cell. In the next case, RT elements give 12 + 4 degrees of freedom per cell. The second-order trimmed serendipity element with linears gives just 10 + 3 degrees of freedom per cell.

We define $u_h \subset V_h$ and $\eta_h \subset W_h$ as solutions of the discrete variational problem

$$\begin{aligned}\left(\frac{1}{H}u_{h,t}, v_h\right) + \frac{1}{\epsilon}\left(\frac{f}{H}u_h^{\perp}, v_h\right) - \frac{\beta}{\epsilon^2}\left(\eta_h, \nabla \cdot v_h\right) + \left(\frac{C}{H}u_h, v_h\right) &= \left(F, v_h\right), \\ \left(\eta_{h,t}, w_h\right) + \left(\nabla \cdot u_h, w_h\right) &= 0.\end{aligned} \tag{5}$$

In previous work [10], we analyzed the semi-discrete form of this method, and in [20] we analyzed a symplectic Euler time discretization of mixed methods for the simpler (obtained from our current model putting $f = 0, C = 0$ and possibly allowing β, ϵ to vary spatially) acoustic wave equation. For each time step, this method only requires the inversion of a mass matrix for V_h and another for W_h. However, it requires a CFL-like time step constraint with $\Delta t = \mathcal{O}(h)$ (the constant depends somehow on the shape of mesh elements), is only first-order accurate, and only conserves a quantity close to the actual system energy in the undamped case. Moreover, in the finite element context even explicit methods require the inversion of mass matrices, unless the mesh and approximating spaces admit some kind of diagonal approximation (e.g. lumping).

In this paper, we turn to implicit methods, especially Crank-Nicolson. This method is second-order accurate in time, does not require a CFL condition for stability, and exactly conserves the system energy for the undamped equations. We also point out that, for linear problems it is equivalent to the implicit midpoint rule, which is the lowest-order Gauss-Legendre implicit Runge-Kutta method. In addition to their A-stability, these methods are both symplectic and B-stable, which makes them seem quite appropriate for problems based on a energy conservation principle plus some damping mechanism. (Note: for nonlinear problems, Crank-Nicolson is actually the lowest-order LobattoIIIA method, which is still A-stable but not symplectic.) On the down side, it requires the solution of a more complicated system of equations at each time step than symplectic Euler. Error analysis goes through following standard techniques; our goal here is the design and analysis of an effective preconditioner.

Selecting time levels $0 = t_0 < t_1 < \cdots < t_N = T$ with $t_n = t_0 + n\Delta t$, we seek a sequence of $\{(u_h^n, \eta_h^n)\}_{n=0}^N$ such that for each $n \geq 1$,

$$
\begin{aligned}
\left(\frac{1}{H}\frac{u_h^{n+1} - u_h^n}{\Delta t}, v_h\right) + \frac{1}{\epsilon}\left(\frac{f}{2H}\left((u_h^{n+1})^\perp + (u_h^n)^\perp\right), v_h\right) \\
-\frac{\beta}{2\epsilon^2}\left(\eta_h^{n+1} + \eta_h^n, \nabla\cdot v_h\right) + \left(\frac{C}{2H}\left(u_h^{n+1} + u_h^n\right), v_h\right) &= \left(F^{n+\frac{1}{2}}, v_h\right), \\
\left(\frac{\eta_h^{n+1} - \eta_h^n}{\Delta t}, w_h\right) + \left(\tfrac{1}{2}\nabla\cdot(u_h^{n+1} + u_h^n), w_h\right) &= 0.
\end{aligned}
\tag{6}
$$

for all $v_h \in V_h$ and $w_h \in W_h$.

Given u_h^n and η_h^n, this defines a linear system for u_h^{n+1} and η_h^{n+1} that must be solved at each time step. Dropping the superscripts and subscripts, multiplying through by Δt and putting $k \equiv \frac{\Delta t}{2}$, we arrive at a canonical equation to be solved at each time step:

$$
\begin{aligned}
\left(\frac{1}{H}u, v\right) + \left(\frac{fk}{\epsilon H}u^\perp, v\right) - \frac{\beta k}{\epsilon^2}\left(\eta, \nabla\cdot v\right) + \left(\frac{Ck}{H}u, v\right) &= (F, v), \\
(\eta, w) + k\left(\nabla\cdot u, w\right) &= (G, w),
\end{aligned}
\tag{7}
$$

where the solution $u \in V_h$ and $\eta \in W_h$ and similar for test functions. Equivalently, we can define a bilinear form on the product space $V_h \times W_h$. Adding together the first equation and $\frac{\beta}{\epsilon^2}$ times the second, we let $\mathbf{u} = (u, \eta)$ and $\mathbf{v} = (v, w)$, to define

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) = \left(\frac{1}{H} u, v \right) + \left(\frac{fk}{\epsilon H} u^{\perp}, v \right) - \frac{\beta k}{\epsilon^2} \left(\eta, \nabla \cdot v \right) \\ + \left(\frac{Ck}{H} u, v \right) + \frac{\beta}{\epsilon^2} \left(\eta, w \right) + \frac{\beta k}{\epsilon^2} \left(\nabla \cdot u, w \right). \end{aligned} \tag{8}$$

Before proceeding, we remark that many other single-stage methods (e.g. backward Euler or the implicit midpoint rule) would give variational problems of this form as well. Now, solving a variational problem associated with this bilinear form gives rise to a block-structured linear system

$$\begin{bmatrix} \check{M} & -\frac{\beta k}{\epsilon^2} D^T \\ \frac{\beta k}{\epsilon^2} D & \frac{\beta}{\epsilon^2} M \end{bmatrix} \begin{bmatrix} \mathrm{u} \\ \eta \end{bmatrix} = \begin{bmatrix} \mathrm{f} \\ \mathrm{g} \end{bmatrix}, \tag{9}$$

where for finite element bases $\{\psi_i\}_{i=1}^{\dim V_h}$ and $\{\phi_i\}_{i=1}^{\dim W_h}$, we have matrices

$$\begin{aligned} \check{M}_{ij} &= \left(\frac{1 + Ck}{H} \psi_j, \psi_i \right) + \left(\frac{fk}{\epsilon H} \psi_j^{\perp}, \psi_i \right), \\ D_{ij} &= \left(\nabla \cdot \psi_j, \phi_i \right), \\ M_{ij} &= \left(\phi_j, \phi_i \right). \end{aligned} \tag{10}$$

Note that $\check{M}$ is not just a weighted mass matrix. It is nonsymmetric owing to the skew off-diagonal terms and the skew term involving $\cdot^{\perp}$ in the top left block. It is the diagonal skew term rather than the off-diagonals that lead to the parameter-dependence in our weighted-norm estimate.

3. Preconditioning

Now, we turn to developing a preconditioner for (8). Here, we concretize the abstract approach taken in [21, 22] for our particular tide model. Essentially, a bounded bilinear form a on a Hilbert space V is equivalent to a linear operator $\mathcal{A}$ from V into its topological dual V'. Classical Galerkin discretization restricts this bilinear form and operator to some finite-dimensional subspace $V_h \subset V$. Moreover, the discrete operator $\mathcal{A}_h : V_h \to V_h'$ is encoded by the usual finite element stiffness matrix A obtained by substituting each member of a basis for V_h into each argument of a.

When one seeks to solve the linear system for the discrete solution by means of an iterative method such as GMRES [12], the *conditioning* of the matrix A plays a critical role. As the condition number, and hence number of iterations required, of A degrades under mesh refinement, it is critical to *precondition* the linear system by means of (at least morally) pre-multiplying the system

$$Ax = b$$

by some linear operator P^{-1}. Thus, one obtains the equivalent system

$$P^{-1}Ax = P^{-1}b,$$

and if the conditioning of $P^{-1}A$ is much better than that of A, the iterative method should converge much faster. Of course, the cost of applying P^{-1} at each iteration must not offset the reduction in iteration count for the preconditioner to be successful.

One can think of the matrix P as discretizing some simpler operator $\mathcal{P} : V \to V'$ so that the product $P^{-1}A$ encodes a bounded operator from V_h onto itself. In the simplest case this is the *Riesz map*, which isometrically identifies each $f \in V_h'$ uniquely with some $v \in V_h$ so that $f(u) = (u, v)$ for all $u \in V_h$. Bounded operators have bounded spectra, and functional-analytic bounds obtained on $\mathcal{P}^{-1}\mathcal{A}$ mean that the matrices $P^{-1}A$ will inherit mesh-independent bounds on their spectra. We refer to [21, 22] for further discussion of this approach.

In addition to mesh refinement, variation in physical parameters can also contribute adversely to the conditioning of discrete problems. While the standard Riesz map serves as a simple preconditioner that eliminates mesh dependence, it does not address physical constants. Increasingly, attempts are made to design *parameter-robust* preconditioners, meaning that they also eliminate or at least mitigate the dependence of the conditioning on system parameters.

In this section, we present two preconditioners. One is based on inverting weighted mass matrices. This utilizes an inverse assumption in $H(\mathrm{div})$ to work in purely a discrete L^2 inner product, so that the bounds depend on the mesh parameter h in such a way that conditioning (as expected) degrades as $h \searrow 0$. However, this dependence can be offset by taking $k = \mathcal{O}(h)$, a

CFL-like criterion that enforces conditioning rather than stability of the time-discretization. Our second approach better respects the functional analytic structure, working in a weighted $H(\mathrm{div}) \times L^2$ inner product. Here, we obtain a mesh-independent bound that is also far less dependent on other parameters at the expense of a more complicated operator to invert as a preconditioner.

3.1. Mass matrices: block diagonal

A simple approach that may help for small time steps is to precondition the linear system with the block diagonal matrix

$$P_M = \begin{bmatrix} \tilde{M} & 0 \\ 0 & \frac{\beta}{\epsilon^2} M \end{bmatrix}, \tag{11}$$

where

$$\tilde{M}_{ij} = \left(\frac{1}{H} \psi_j, \psi_i \right) \tag{12}$$

is the mass-like matrix obtained from the $\frac{1}{H}$-weighted inner product of the V_h basis functions and M is as in (10).

This is motivated by the observing that the bilinear form a from (8) is continuous and coercive on discrete subspaces of $(L^2)^2 \times L^2$, although the constants depend on the discretization parameters h and k as well as the physical parameters. We define the norm

$$\|\mathbf{u}\|_2^2 = \|u\|_{\frac{1}{H}}^2 + \frac{\beta}{\epsilon^2} \|\eta\|^2, \tag{13}$$

where $\|u\|_{\frac{1}{H}}^2 = \left(\frac{1}{H} u, u \right)$. The the inner product for this norm generates the matrices in (11).

Establishing well-posedness of variational problems for the bilinear form a follows from demonstrating continuity and inf-sup estimates in $H(\mathrm{div}) \times L^2$. However, we can study the mass matrix preconditioner (11) by means of establishing continuity and coercivity of a on finite element subspaces equipped with the L^2 norms. This analysis is somewhat nonstandard, but it establishes an alternate proof of solvability of the discrete system and more importantly, allows us to demonstrate mesh-independence of (11) as a preconditioner subject to a CFL-like restriction on k.

Proposition 3.1. *Let*

$$\kappa = 4 \max \left\{ 1 + C^* k + \frac{f^* k}{\epsilon}, \frac{\sqrt{\beta} k C_I}{\epsilon H_* h} \right\}. \tag{14}$$

Then, the bilinear form a satisfies

$$a(\mathbf{u}, \mathbf{v}) \leq \kappa \|\mathbf{u}\|_2 \|\mathbf{v}\|_2, \tag{15}$$

and

$$a(\mathbf{u}, \mathbf{u}) \geq \|\mathbf{u}\|_2^2. \tag{16}$$

Proof. We begin with the continuity estimate (15).

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) &= \left(\frac{1}{H}u, v\right) + \left(\frac{fk}{\epsilon H}u^\perp, v\right) - \frac{\beta k}{\epsilon^2}\left(\eta, \nabla \cdot v\right) \\ &\quad + \left(\frac{Ck}{H}u, v\right) + \frac{\beta}{\epsilon^2}\left(\eta, w\right) + \frac{\beta k}{\epsilon^2}\left(\nabla \cdot u, w\right) \\ &\leq \left(1 + C^* k + \frac{f^* k}{\epsilon}\right) \|u\|_{\frac{1}{H}} \|v\|_{\frac{1}{H}} \\ &\quad + \frac{\beta}{\epsilon^2}\|\eta\|\|w\| + \frac{\beta k}{\epsilon^2}\|\nabla \cdot u\|\|w\| + \frac{\beta k}{\epsilon^2}\|\eta\|\|\nabla \cdot v\|. \end{aligned}$$

Applying the inverse estimate to the divergences and converting to the $\frac{1}{H}$-weighted norm now gives

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) &\leq \left(1 + C^* k + \frac{f^* k}{\epsilon}\right) \|u\|_{\frac{1}{H}} \|v\|_{\frac{1}{H}} \\ &\quad + \frac{\beta}{\epsilon^2}\|\eta\|\|w\| + \frac{\beta k C_I}{h H_* \epsilon^2}\|u\|_{\frac{1}{H}}\|w\| + \frac{\beta k C_I}{h H_* \epsilon^2}\|\eta\|\|v\|_{\frac{1}{H}}. \end{aligned}$$

The result follows by absorbing $\sqrt{\beta}/\epsilon$ into the norm of $\|w\|$ and $\|\eta\|$ in the third and fourth term and then recognizing each term as bounded by $\kappa\|\mathbf{u}\|\|\mathbf{v}\|$.

The rescaling of the second equation to produce the bilinear form a makes the coercivity estimate rather simple. Noting that $u^\perp \cdot u = 0$ pointwise and that the divergence terms in $a(\mathbf{u}, \mathbf{u})$ cancel, we have

$$\begin{aligned} a(\mathbf{u}, \mathbf{u}) &= \left(\frac{1}{H}u, u\right) + \left(\frac{Ck}{H}u, u\right) + \frac{\beta k}{\epsilon^2}\left(\eta, \eta\right) \\ &\geq \left(1 + C_* k\right) \|u\|_{\frac{1}{H}}^2 + \frac{\beta}{\epsilon}^2 \|\eta\|^2 \\ &\geq \|\mathbf{u}\|_2^2. \end{aligned}$$

$\square$

It is possible to achieve slightly better constants (e.g. through more careful use of discrete Cauchy-Schwarz), but the main issue remains: the conditioning of the system (continuity divided by coercivity constants) depends on the discretization (as well as physical) parameters, scaling like $\frac{k}{h}$. For a fixed time step, the conditioning degrades like h^{-1}, and so preconditioning with weighted mass matrices is only scalable if one also imposes a CFL-like time step restriction. Moreover, even including some weights in the norm, we still have quite a bit of parameter dependence in our estimate.

3.2. Weighted-norm preconditioning

The mesh-dependence in our estimate comes from invoking the inverse assumption in order to obtain L^2 estimates. Our bilinear form is not coercive on subspaces of $H(\mathrm{div}) \times L^2$, but we can prove that it still defines a bounded operator with bounded inverse in a weighted norm that nearly eliminates parameter dependence. Such techniques appear for other applications [23, 24] as well, and are based on defining a suitable (parameter-dependent) inner product in which the problem is well-behaved rather than algebraic considerations such as merely selecting the block diagonal or triangular part of the system matrix [25, 26, 27]

We can equip $H(\mathrm{div})$ with the following weighted norm

$$\|u\|_a^2 = \|(1 + Ck)u\|_{\frac{1}{H}}^2 + \frac{k^2\beta}{\epsilon^2}\|\nabla \cdot u\|^2 \tag{17}$$

and, as previously, L^2 with the norm

$$\|\eta\|_b^2 = \frac{\beta}{\epsilon^2}\|\eta\|^2. \tag{18}$$

We then equip the product space $H(\mathrm{div}) \times L^2$ with the norm

$$\begin{aligned} |||\mathbf{u}|||^2 = |||(u, \eta)|||^2 &= \|u\|_a^2 + \|\eta\|_b^2 \\ &= \|(1 + Ck)u\|_{\frac{1}{H}}^2 + \frac{k^2\beta}{\epsilon^2}\|\nabla \cdot u\|^2 + \frac{\beta}{\epsilon^2}\|\eta\|^2 \end{aligned} \tag{19}$$

This norm is derived from a weighted inner product

$$((\mathbf{u}, \mathbf{v})) = ((1 + Ck)\, u, v)_{\frac{1}{H}} + \frac{k^2\beta}{\epsilon^2}\,(\nabla \cdot u, \nabla \cdot v) + \frac{\beta}{\epsilon^2}\,(\eta, w)\,. \tag{20}$$

Discretizing this bilinear form on mixed function spaces $V_h \times W_h$ yields a block-diagonal preconditioning matrix:

$$P = \begin{bmatrix} P_{V_h} & 0 \\ 0 & P_{W_h} \end{bmatrix}, \tag{21}$$

where the first block handles the parameter-weighted $H(\mathrm{div})$ inner product and the second is just the standard W_h mass matrix scaled by $\frac{\beta}{\epsilon^2}$. We have

$$\begin{aligned} (P_{V_h})_{ij} &= ((1 + Ck)\,\psi_j, \psi_i)_{\frac{1}{H}} + (\psi_j, \psi_i) + \frac{k^2\beta}{\epsilon^2}\,(\nabla \cdot \psi_j, \nabla \cdot \psi_i)\,, \\ (P_{W_h})_{ij} &= \frac{\beta}{\epsilon^2}\,(\phi_j, \phi_i)\,. \end{aligned} \tag{22}$$

In the lowest-order case (either on triangles or squares), W_h consists of piecewise constants so that P_{W_h} is simply a diagonal matrix. Since the top left block discretizes a differential operator, applying $P_{V_h}^{-1}$ will constitute the bulk of the cost in applying the preconditioner. Options based on multigrid are available, and we discuss these more later.

The following result shows the boundedness of a in this norm, with mild dependence on parameters, which we discuss below.

Theorem 3.1. *For all $\mathbf{u} = (u, \eta)$ and $\mathbf{v} = (v, w)$ in $H(\mathrm{div}) \times L^2$, the bilinear form a satisfies*

$$a(\mathbf{u}, \mathbf{v}) \leq K |||\mathbf{u}||| |||\mathbf{v}|||, \tag{23}$$

where constant $K = K_{k,\epsilon} = \max\left\{2, 1 + \frac{k}{\epsilon}\right\}$.

Proof. The proof is a direct calculation using Cauchy-Schwarz, the isometry of $\cdot^{\perp}$, and upper bounds on some of the spatially varying coefficients

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) &= \left(\frac{1}{H}u, v\right) + \left(\frac{fk}{\epsilon H}u^{\perp}, v\right) - \frac{\beta k}{\epsilon^2}\,(\eta, \nabla \cdot v) \\ &\quad + \left(\frac{Ck}{H}u, v\right) + \frac{\beta}{\epsilon^2}\,(\eta, w) + \frac{\beta k}{\epsilon^2}\,(\nabla \cdot u, w) \\ &\leq \|(1 + Ck)u\|_{\frac{1}{H}} \|(1 + Ck)v\|_{\frac{1}{H}} + \frac{f^* k}{\epsilon}\|u\|_{\frac{1}{H}} \|v\|_{\frac{1}{H}} \\ &\quad + \frac{\beta k}{\epsilon^2}\|\eta\|\|\nabla \cdot v\| + \frac{\beta}{\epsilon^2}\|\eta\|\|w\| + \frac{\beta k}{\epsilon^2}\|\nabla \cdot u\|\|w\|. \end{aligned} \tag{24}$$

Now, we can write

$$\|u\|_{\frac{1}{H}} \leq \frac{1}{\sqrt{1+C_* k}} \|(1+Ck)u\|_{\frac{1}{H}} \leq \|(1+Ck)u\|_{\frac{1}{H}}$$

and recalling that $|f^*| \leq 1$,

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) \leq & \left(1 + \frac{k}{\epsilon}\right) \|(1+Ck)u\|_{\frac{1}{H}} \|(1+Ck)v\|_{\frac{1}{H}} \\ & + \frac{\beta k}{\epsilon^2} \|\eta\| \|\nabla \cdot v\| + \frac{\beta}{\epsilon^2} \|\eta\| \|w\| + \frac{\beta k}{\epsilon^2} \|\nabla \cdot u\| \|w\|. \end{aligned} \tag{25}$$

Now, we recognize the right-hand side as the dot product of

$$\begin{bmatrix} \sqrt{(1+\frac{k}{\epsilon})} \|(1+Ck)u\|_{\frac{1}{H}} \\ \frac{\sqrt{\beta}}{\epsilon} \|\eta\| \\ \frac{\sqrt{\beta}}{\epsilon} \|\eta\| \\ \frac{k\sqrt{\beta}}{\epsilon} \|\nabla \cdot u\| \end{bmatrix}^t \begin{bmatrix} \sqrt{(1+\frac{k}{\epsilon})} \|(1+Ck)v\|_{\frac{1}{H}} \\ \frac{k\sqrt{\beta}}{\epsilon} \|\nabla \cdot v\| \\ \frac{\sqrt{\beta}}{\epsilon} \|w\| \\ \frac{\sqrt{\beta}}{\epsilon} \|w\| \end{bmatrix},$$

whence discrete Cauchy-Schwarz gives

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) \leq & \left[\left(1 + \frac{k}{\epsilon}\right) \|(1+Ck)u\|^2_{\frac{1}{H}} + \frac{k^2 \beta}{\epsilon^2} \|\nabla \cdot u\|^2 + \frac{2\beta^2}{\epsilon} \|\eta\|^2 \right]^{\frac{1}{2}} \\ & \times \left[\left(1 + \frac{k}{\epsilon}\right) \|(1+Ck)v\|^2_{\frac{1}{H}} + \frac{k^2 \beta}{\epsilon^2} \|\nabla \cdot v\|^2 + \frac{2\beta^2}{\epsilon} \|w\|^2 \right]^{\frac{1}{2}}, \end{aligned} \tag{26}$$

and the result follows from a simple bound. □

Note that the Coriolis term $\frac{fk}{\epsilon}(u^\perp, v)$, which is skew and on the diagonal leaves the term scaled by $\frac{k}{\epsilon}$ so that we do not obtain total parameter-independence. We can interpret this bound as saying that the Rossby number ϵ induces a time scale, independent of h, that must be resolved in order to obtain a robust continuity estimate. More precisely,

Corollary 3.1. *For any $M \geq 2$ and $\epsilon > 0$, there exists k_0 such that for any $k \leq k_0$,*

$$a(\mathbf{u}, \mathbf{v}) \leq M \|\|\mathbf{u}\|\| \|\|\mathbf{v}\|\|.$$

Next, we bound the inverse of the operator induced by a by means of an inf-sup condition. Unlike our continuity estimate, this is completely parameter-independent.

Theorem 3.2. *The bilinear form a satisfies the estimate*

$$\inf_{\mathbf{u}\neq 0}\sup_{\mathbf{v}} a(\mathbf{u},\mathbf{v}) \geq \frac{\sqrt{3}}{6}. \tag{27}$$

Proof. We let $\mathbf{u} = (u, \eta)$ be given and put $\mathbf{v} = (v, w) = (u, \eta + k\nabla \cdot u)$ so that

$$\begin{aligned} a(\mathbf{u},\mathbf{v}) &= \left(\frac{1}{H}u, u\right) + \left(\frac{fk}{\epsilon H}u^\perp, u\right) - \frac{\beta k}{\epsilon^2}(\eta, \nabla\cdot u) \\ &\quad + \left(\frac{Ck}{H}u, u\right) + \frac{\beta}{\epsilon^2}(\eta, \eta + k\nabla\cdot u) + \frac{\beta k}{\epsilon^2}(\nabla\cdot u, \eta + k\nabla\cdot u) \\ &= \|(1+Ck)u\|^2_{\frac{1}{H}} + \frac{\beta}{\epsilon^2}\|\eta\|^2 + \frac{k^2\beta}{\epsilon^2}\|\nabla\cdot u\|^2 + \frac{k\beta}{\epsilon^2}(\eta, \nabla\cdot u). \end{aligned} \tag{28}$$

The last term is readily bounded below by $-\frac{\beta}{2\epsilon^2}\left(\|\eta\|^2 + k^2\|\nabla\cdot u\|^2\right)$ so that

$$a(\mathbf{u},\mathbf{v}) \geq \|(1+Ck)u\|^2_{\frac{1}{H}} + \frac{\beta}{2\epsilon^2}\|\eta\|^2 + \frac{k^2\beta}{2\epsilon^2}\|\nabla\cdot u\|^2 \geq \frac{1}{2}|||\mathbf{u}|||^2. \tag{29}$$

Now, we have that

$$\begin{aligned} |||\mathbf{v}|||^2 &= \|(1+Ck)u\|^2_{\frac{1}{H}} + \frac{k^2\beta}{\epsilon^2}\|\nabla\cdot u\|^2 + \frac{\beta}{\epsilon^2}\|\eta + k\nabla\cdot u\|^2 \\ &\leq \|(1+Ck)u\|^2_{\frac{1}{H}} + 3\frac{\beta}{\epsilon^2}\|\nabla\cdot u\|^2 + 2\frac{\beta}{\epsilon^2}\|\eta\|^2 \\ &\leq 3|||\mathbf{u}|||^2, \end{aligned} \tag{30}$$

and combining this with (29) gives the result. □

Because the spectral radius of a matrix is bounded above by any natural norm, these results prove that the moduli of the eigenvalues of $P^{-1}A$ are bounded below by a constant (in fact, $\frac{\sqrt{3}}{6}$) independently of the mesh size and all the physical constants. The moduli of the eigenvalues of $P^{-1}A$ are further bounded above the greater of 2 and $1 + \frac{k}{\epsilon}$, which can degrade as the Rossby number decreases.

3.3. Dropping the damping term from the preconditioner

In [11], we consider energy and error analysis of a possibly degenerate nonlinear damping term, where the term Cu in (2) is replaced by a more general $g(u)$. Typical use cases have a power law such as $g(u) = |u|^{p-1}u$, modified to have linear growth for large u (at least as a technical assumption). In this case, $g(u)$ tends to zero as $|u|$ does so that the effective damping decays.

Carrying out the same manipulations that leads to (8) for nonlinear damping leads to the nonlinear variational form

$$\begin{aligned} F(\mathbf{u};\mathbf{v}) = &\left(\frac{1}{H}u, v\right) + \left(\frac{fk}{\epsilon H}u^{\perp}, v\right) - \frac{\beta k}{\epsilon^2}\left(\eta, \nabla\cdot v\right) \\ &+ \left(\frac{k}{H}g(u), v\right) + \frac{\beta}{\epsilon^2}\left(\eta, w\right) + \frac{\beta k}{\epsilon^2}\left(\nabla\cdot u, w\right). \end{aligned} \tag{31}$$

Newton-type methods require the Jacobian of this system. Linearizing about some state $\mathbf{u}_0 = (u_0, \eta_0)$, we have

$$\begin{aligned} J_{\mathbf{u}_0}(\mathbf{u};\mathbf{v}) = &\left(\frac{1}{H}u, v\right) + \left(\frac{fk}{\epsilon H}u^{\perp}, v\right) - \frac{\beta k}{\epsilon^2}\left(\eta, \nabla\cdot v\right) \\ &+ \left(\frac{k}{H}g'(u_0)u, v\right) + \frac{\beta}{\epsilon^2}\left(\eta, w\right) + \frac{\beta k}{\epsilon^2}\left(\nabla\cdot u, w\right). \end{aligned} \tag{32}$$

All of the analysis carried out in [11] required monotonicity of g, so that $g' > 0$. In this case, (32) takes the same form as (8) with $C \leftrightarrow g'(u_0)$. As a result, our theory carries over directly to preconditioning each Newton step provided that the Riesz map (20) is updated at each iteration (of each time step).

On the other hand, many preconditioners such as algebraic multigrid can be relatively expensive to initialize, so that it is helpful to reuse the same bilinear form between successive linear solves as the damping changes. We drop the damping term in the bilinear form in (20) to define

$$((\mathbf{u},\mathbf{v}))_* = (u, v)_{\frac{1}{H}} + \frac{k^2\beta}{\epsilon^2}(\nabla\cdot u, \nabla\cdot v) + \frac{\beta}{\epsilon^2}(\eta, w), \tag{33}$$

and an associated norm

$$|||\mathbf{u}|||_*^2 \equiv ((\mathbf{u},\mathbf{u}))_*. \tag{34}$$

This norm is, at the cost of some dependence on C^*, equivalent to $|||\cdot|||$

Proposition 3.2. *For all* $\mathbf{u} = (u, \eta) \in V_h \times W_h$*,*

$$\tfrac{1}{1+C^*k}|||\mathbf{u}|||^2 \leq |||\mathbf{u}|||_*^2 \leq |||\mathbf{u}|||^2 \tag{35}$$

Proof. The proof is elementary and uses that $\frac{1+Ck}{1+C^*k} \leq 1 \leq 1 + Ck$ in the definition of $((\cdot, \cdot))$. $\square$

Theorems 3.1 and 3.2 can be readily restated using this norm:

Corollary 3.2. *For all* $\mathbf{u}, \mathbf{v} \in H(\mathrm{div}) \times L^2$,

$$a(\mathbf{u}, \mathbf{v}) \leq K_* |||\mathbf{u}|||_* |||\mathbf{v}|||_*, \tag{36}$$

where $K_* = (1 + C^*k) \max\{2, 1 + \frac{k}{\epsilon}\}$*, and the inf-sup constant of* a *with respect to* $|||\cdot|||_*$ *is also at least* $\frac{\sqrt{3}}{6}$.

Typically, the linear damping is small compared to the other effects in the equation so that the effective bounds on the preconditioner are essentially unchanged. Much as with the Rossby number, the time step can be reduced to accommodate large C^* if it becomes a problem.

4. Numerical results

We have implemented a mixed finite element discretization of the tide model and developed all of our preconditioners within the Firedrake framework [28]. Firedrake is an automated system for the solution of PDE using the finite element method. It allows users to specify the variational form of their problems using the Unified Form Language (UFL) in Python [29], generates efficient low-level code for the evaluation of operators, and interfaces tightly with PETSc for scalable algebraic solvers. Firedrake also allows users to specify UFL for preconditioning operator that is distinct from that for the problem being solved, and we make use of this facility. A sample listing is shown in Figure 1. The solver/preconditioner is typically configured by passing a dictionary into `solve` function as a keyword argument.

Before proceeding with the numerical investigation of our weighted-norm preconditioner, we confirm that the mass matrix preconditioner (11) in fact performs poorly as suggested by Proposition 3.1. In this experiment, we fix exemplary values of $C = f = H = 1$ and $\beta = \epsilon = 0.01$. We consider several values of k and a sequence of refined meshes. We subdivide the unit square into an $N \times N$ mesh of squares, each of which is further subdivided into two right triangles. On each mesh, we approximate u in the lowest-order Raviart-Thomas space and η in the space of discontinuous piecewise constants. Figure 2 shows the GMRES(100) iteration count required to solve

```python
from firedrake import *

mesh = UnitSquareMesh(16, 16)
V = FunctionSpace(mesh, "RT", 1)
Q = FunctionSpace(mesh, "DG", 0)

Z = V * Q

x, y = SpatialCoordinate(mesh)
k = Constant(k)
Eps = Constant(0.1)
Beta = Constant(0.1)
C = Constant(1.0)
f = Constant(1.0)
beps2 = Beta / Eps**2

up = Function(Z)
v, q = TestFunctions(Z)

u, p = split(up)

F = (inner(u, v) * dx
    + k / Eps * f * inner(perp(u),v) * dx
    - k * beps2 * inner(p, div(v)) * dx
    + C * k * inner(u,v) * dx
    + beps2 * inner(p, q) * dx
    + k * beps2 * inner(div(u), q) * dx
    - beps2 * inner(sin(pi*x)*cos(pi*y),q)*dx)

uu, pp = TrialFunctions(Z)
Jpc = ((Constant(1.0) + C * k) * inner(uu,v)*dx
        + k**2 * beps2 * inner(div(uu),div(v)) *dx
        + beps2 * inner(pp,q)*dx)

bcs = [DirichletBC(Z.sub(0), 0, 'on_boundary')]

solve(F==0, up, bcs=bcs, Jp=Jpc)
solver.solve()
```

Figure 1: Sample Firedrake code for solving the tide model using the Riesz map (20) as a preconditioner. The user can optionally pass a Python dictionary containing PETSc options into the solve function.

the linear system using (11) as a preconditioner. As expected, for fixed k, the iteration count increases under mesh refinement. For a fixed mesh, increasing k also dramatically increases the iteration count. We should hope for lower iteration counts and greater parameter robustness from our weighted-norm preconditioner.

In our next sequence of experiments, we test the results of Theorems 3.1 and 3.2. We keep the damping coefficient C, Coriolis parameter f, and bathymetry H all equal 1, the Burger number $\beta = 0.1$ and Rossby number $\epsilon = 0.01$, and continue a division of the unit square into an $N \times N$ mesh. To demonstrate the independence of our preconditioning technique with respect to the particular discretization, we consider both triangular Raviart-Thomas elements and rectangular Raviart-Thomas and serendipity $H(\mathrm{div})$ elements [30] in our experiments. Figures 3, 4, 5, and 6 plot GMRES(100) iteration counts using preconditioners (21) and (33) for various values of N and k for these discretizations. While the particular iteration counts vary slightly for different elements in the figures, several trends are consistent. The curves showing mesh refinement for a given k are largely flat, indicating a high degree of mesh-independence.

We do see some dependence of the iteration count on k for a fixed mesh. For the smaller and larger values of k, the iteration count is smaller, and it is largest for intermediate values of k. We explain this as follows. Comparing the preconditioning bilinear form (20) to the bilinear form (8) when $k \searrow 0$ decreasing, we observe that the bilinear form and the preconditioner approach the same limit, explaining the iteration counts dropping to 1 or 2. The case for k large is more subtle. However, we observe that (8) approaches a variable-coefficient Laplace operator (scaled by k) and that the preconditioner is just k times a Riesz map with respect to which that operator is known to be well-conditioned. The case of intermediate k is somewhere between these limiting cases, and the observed behavior is closer to the worst-case estimate.

As a final note on this discussion, we compare the left and right hand plots in each of the Figures 3, 4, 5, and 6. We typically see a slight increase in iteration count, although this could become more significant for strongly heterogeneous problems.

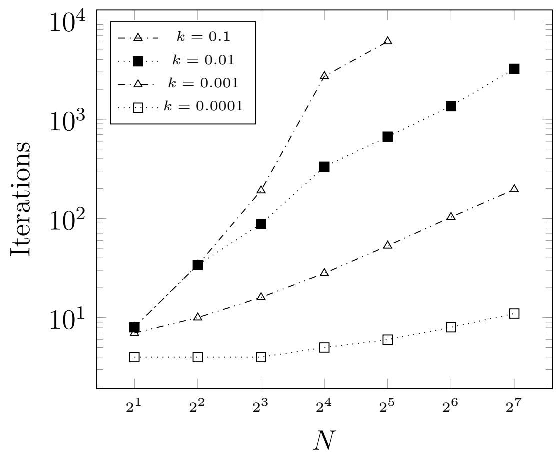
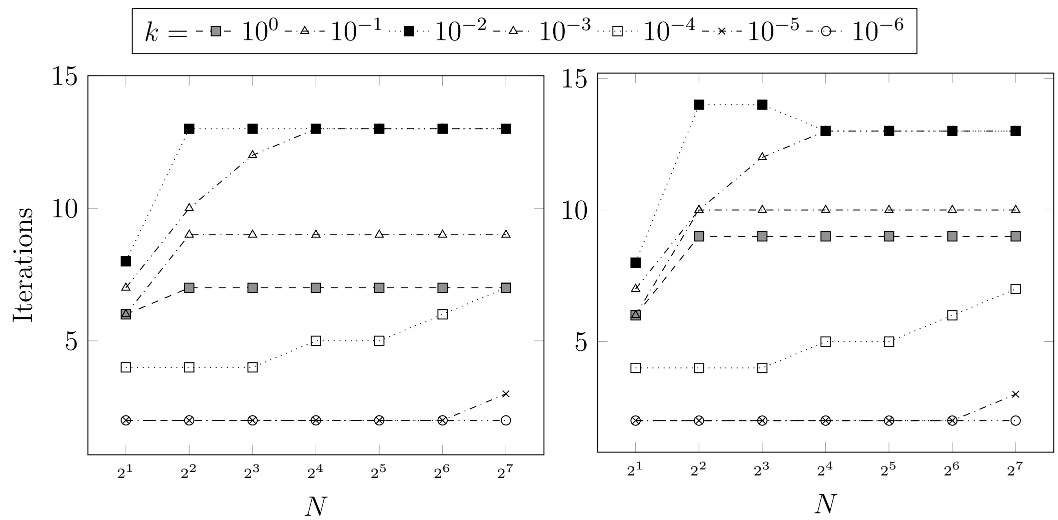


Figure 2: Iteration count versus mesh refinement under various k values for $C = f = 1$, $\beta = 0.1$, and $\epsilon = 0.1$. The unit square is divided into an $N \times N$ mesh of squares, each subdivided into two right triangles. Lowest-order Raviart-Thomas discretization is used, with mass matrix (11) as a preconditioner.

To further confirm our theorem, we now fix the mesh at $N = 128$ and studying the iteration count as a function of ϵ and k. These results are shown in Figures 7, 8, and 9. This shows that, for fixed k, increasing ϵ also increases the iteration count. On the other hand, for fixed ϵ, one finds the largest iteration counts for intermediate values of the time step. Much as the mesh-dependence study, we remark that varying the discretization order and cell shape has little effect on the results.

While these numerical results demonstrate the theoretical robustness of our preconditioner, they have been obtained by applying the preconditioner with a sparse direct solver. Using an iterative solver with sufficiently tight tolerance should give identical iteration counts, but it is also interesting to consider an inexact application of the top left block of (21) or (33). Here, we use the $H(\text{div})$ geometric multigrid algorithm of Arnold, Falk, and Winther [31]. (As an alternative, one could employ an algebraic approach such as that in [32].) Rather than pointwise smoothers, multigrid in $H(\text{div})$ requires the

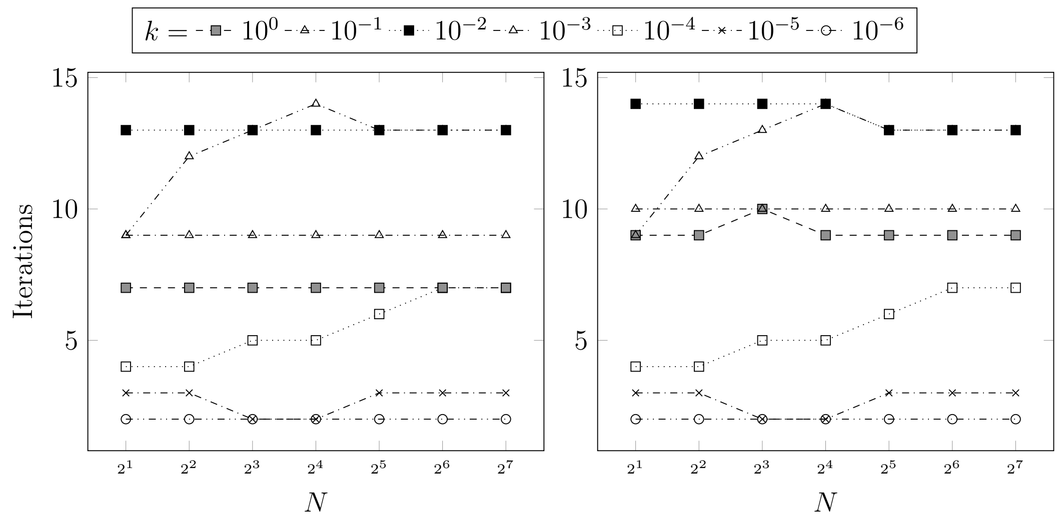


(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 3: Iteration count versus mesh refinement under various k values for $C = f = 1$, $\beta = 0.1$, and $\epsilon = 0.01$. The unit square is divided into an $N \times N$ mesh of squares, each subdivided into two right triangles. Lowest-order Raviart-Thomas discretization is used. The iteration counts are largest for moderate k and decrease as k is either very large or small. Also, removing the damping term (right) from the weighted inner product leads to a small increase in iteration count.

solution of local problems associated with vertex patches on each level. This approach is accessible in Firedrake through the high-level solver interface described in [33] and the `pcpatch` package [34]. We use full multigrid with a sparse direct coarse-grid solve. Comparing Figures 10 and 11 to Figure 3, we see larger iteration counts when a multigrid sweep replaces the sparse direct method. Since the preconditioner is applied inexactly, an increase in iteration counts is not surprising. On the other hand, the trend lines in Figures 10 and 11 are mostly downward with respect to mesh refinement. We conjecture this is due to using a fixed number of multigrid levels with a sparse direct coarse grid solve. As the mesh is refined, a finer coarse grid and hence smaller perturbation in the preconditioner application is obtained.

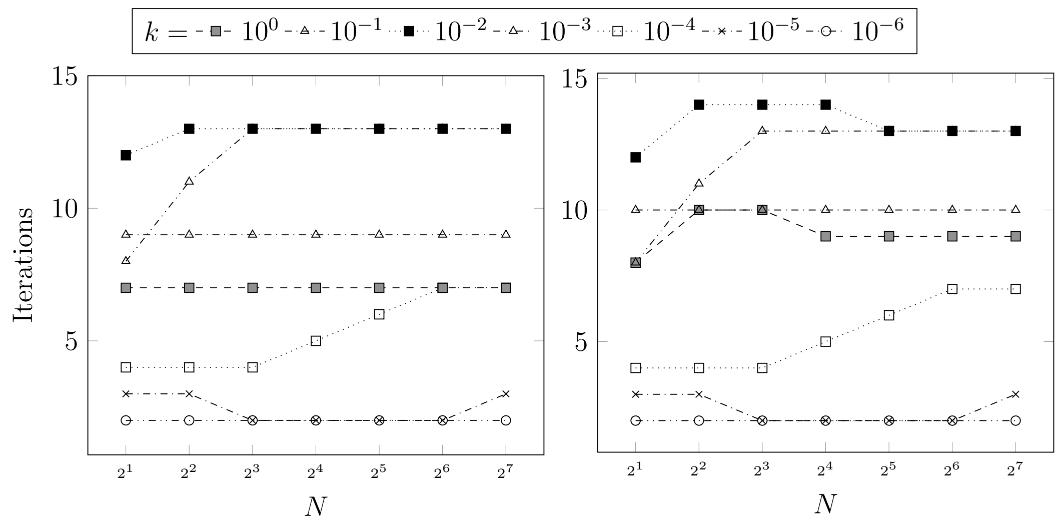


(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 4: Experiment in Figure 3 is repeated, except with the next-to-lowest Raviart-Thomas elements. Since the bilinear form (20) is also discretized in this space, very little changes relative to the lowest-order case.

As a final example, we consider the effect of nonlinear damping on our preconditioner. In particular, we choose $g(u) = |u|^2 u$ in (31) (this bypasses numerical wrinkles in differentiating through the singularity of quadratic damping). Our typical use case is within a time-stepping loop, where the solution at the previous time step serves as an initial guess for Newton iteration. To imitate having such a suitable initial guess, we seed Newton's method with the solution of the linear, undamped problem. In this case, we observed that Newton requires but a single iteration to converge. Iteration counts for the linear solve in this situation are presented in Figure 12, and similar results are obtained for other discretizations. For more rapidly-varying Jacobians, one might require adaptive time stepping or more iterations, but robustly handling time discretizations is beyond the scope of our present investigation.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

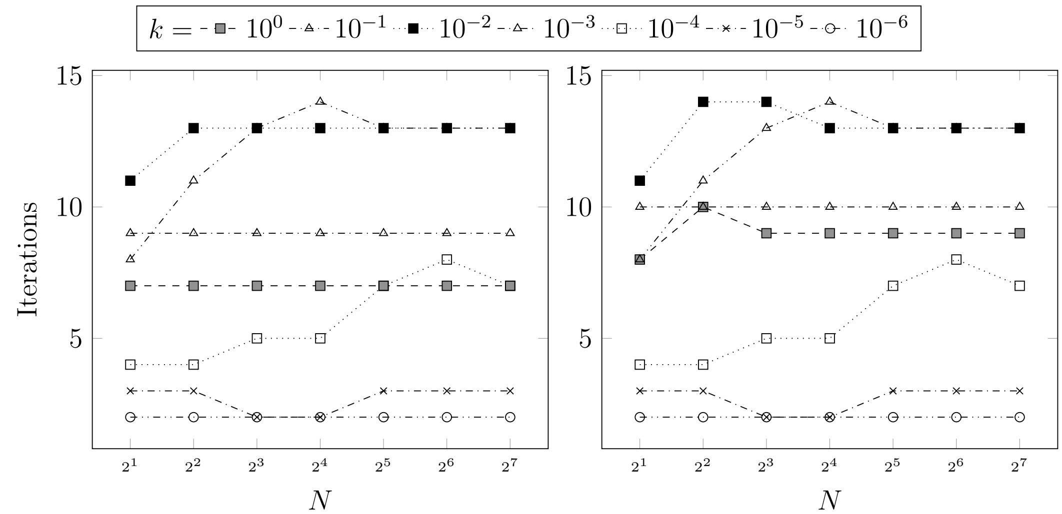
(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 5: Experiment in Figure 3 is again repeated, except with lowest Raviart-Thomas elements on squares. Again, little changes relative to the triangular case.

5. Conclusions

We have developed effective weighted-norm preconditioners for a mixed finite element/Crank-Nicolson discretization of the linearized rotating shallow water equations with (possibly nonlinear) damping. These preconditioners are based on defining a suitable inner product in which the operators are bounded with bounded inverse in a relatively parameter-independent way. These estimates in turn control the spectrum of the preconditioned operator. Our estimates remain dependent on the ratio $\frac{k}{\epsilon}$, although this seems relatively benign in practice. Moreover, inexactly applying the preconditioner through a multigrid sweep and neglecting damping terms in the inner product lead to further simplifications with only mild effects on iteration count.

This work suggests many future research directions. Since our theory and numerical observations both seem independent of mesh type and discretization order, we hope to apply these preconditioners to unstructured quadrilateral elements such as Arbogast-Correa [35]. Moreover, our techniques should be applicable to more complex tide models that might include additional nonlinearities or layering.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 6: Experiment in Figure 3 is again repeated, except second-order trimmed serendipity elements on squares. Again, little changes relative to the triangular case.

References

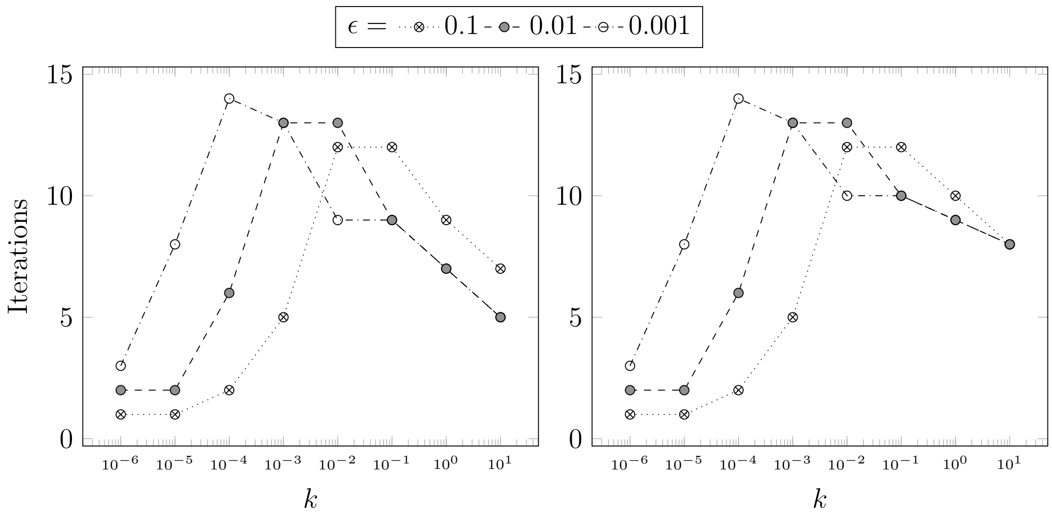
[1] C. Garrett, E. Kunze, Internal tide generation in the deep ocean, Annu. Rev. Fluid Mech. 39 (2007) 57–87.

[2] W. Munk, C. Wunsch, Abyssal recipes II: energetics of tidal and wind mixing, Deep-Sea Research Part I 45 (12) (1998) 1977–2010.

[3] H. Weller, T. Ringler, M. Piggott, N. Wood, Challenges facing adaptive mesh modeling of the atmosphere and ocean, Bulletin of the American Meteorological Society 91 (1) (2010) 105–108.

[4] R. Comblen, J. Lambrechts, J.-F. Remacle, V. Legat, Practical evaluation of five partly discontinuous finite element pairs for the non-conservative shallow water equations, Int. J. Num. Meth. Fluid. 63 (6) (2010) 701–724.

[5] C. Cotter, D. Ham, Numerical wave propagation for the triangular P1DG-P2 finite element pair, Journal of Computational Physics 230 (8) (2011) 2806 – 2820. doi:DOI: 10.1016/j.jcp.2010.12.024.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 7: Iteration count with weighted-norm preconditioning as a function of k and ϵ on a 128×128 mesh divided into right triangles using lowest-order Raviart-Thomas elements. Note that for a fixed k, the iteration count increases with decreasing ϵ. As in Figure 3, removing the damping term from the preconditioner leads to a very slight increase in iteration count.

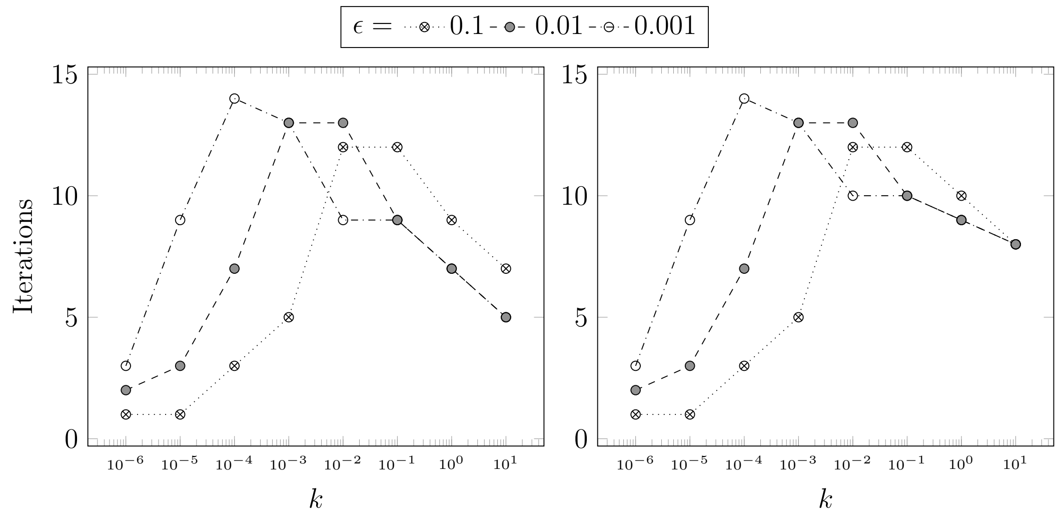
[6] D. Y. Le Roux, Dispersion relation analysis of the $P_1^{NC} - P_1$ finite-element pair in shallow-water models, SIAM Journal on Scientific Computing 27 (2) (2005) 394–414.

[7] D. Le Roux, V. Rostand, B. Pouliot, Analysis of numerically induced oscillations in 2D finite-element shallow-water models part I: Inertia-gravity waves, SIAM J. Sci. Comput. 29 (1) (2007) 331–360.

[8] D. Y. Le Roux, B. Pouliot, Analysis of numerically induced oscillations in two-dimensional finite-element shallow-water models part II: Free planetary waves, SIAM journal on scientific computing 30 (4) (2009) 1971–1991.

[9] V. Rostand, D. Le Roux, Raviart-Thomas and Brezzi-Douglas-Marini finite-element approximations of the shallow-water equations, Int. J. Num. Meth. Fluids 57 (8) (2008) 951–976.

[10] C. J. Cotter, R. C. Kirby, Mixed finite elements for global tide models, Numerische Mathematik 133 (2) (2016) 255–277.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 8: Repeating experiment in Figure 7 with next-to-lowest order elements, showing little change in results.

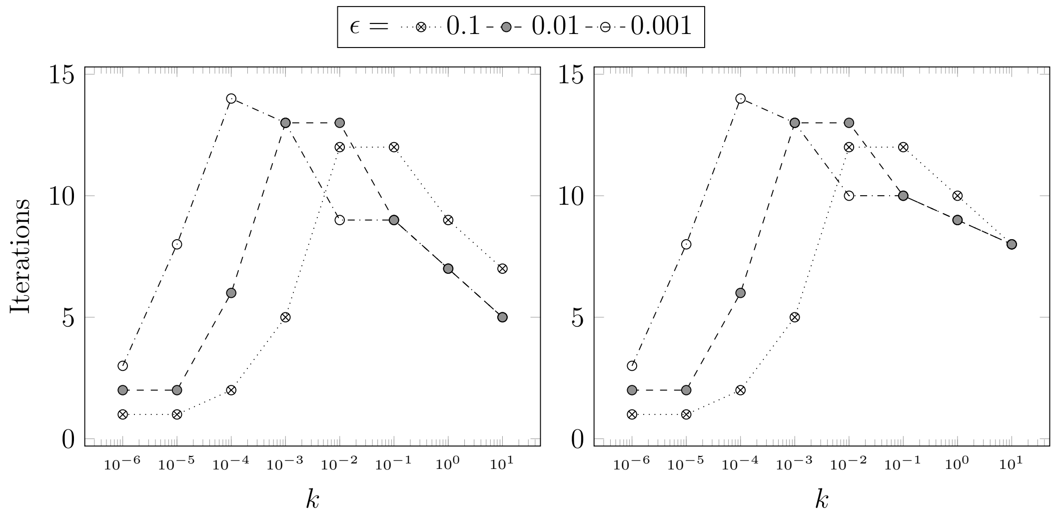
[11] C. J. Cotter, P. J. Graber, R. C. Kirby, Mixed finite elements for global tide models with nonlinear damping, Numerische Mathematik 140 (4) (2018) 963–991.

[12] Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (3) (1986) 856–869.

[13] D. N. Arnold, R. S. Falk, R. Winther, Preconditioning in $H(\mathrm{div})$ and applications, Math. Comput. 66 (219) (1997) 957–984. doi:10.1090/S0025-5718-97-00826-0.
URL `http://dx.doi.org/10.1090/S0025-5718-97-00826-0`

[14] F. Brezzi, M. Fortin, Mixed and hybrid finite element methods, Springer-Verlag New York, Inc., 1991.

[15] P. A. Raviart, J. M. Thomas, A mixed finite element method for 2nd order elliptic problems, in: Mathematical aspects of finite element methods (Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975), Springer, Berlin, 1977, pp. 292–315. Lecture Notes in Math., Vol. 606.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 9: Iteration count with weighted-norm preconditioning as a function of k and ϵ on a 128×128 mesh of squares using lowest-order Raviart-Thomas elements. Again, results are nearly identical to the two triangular cases.

[16] D. N. Arnold, A. Logg, Periodic table of the finite elements, SIAM News 47 (9) (2014) 212.

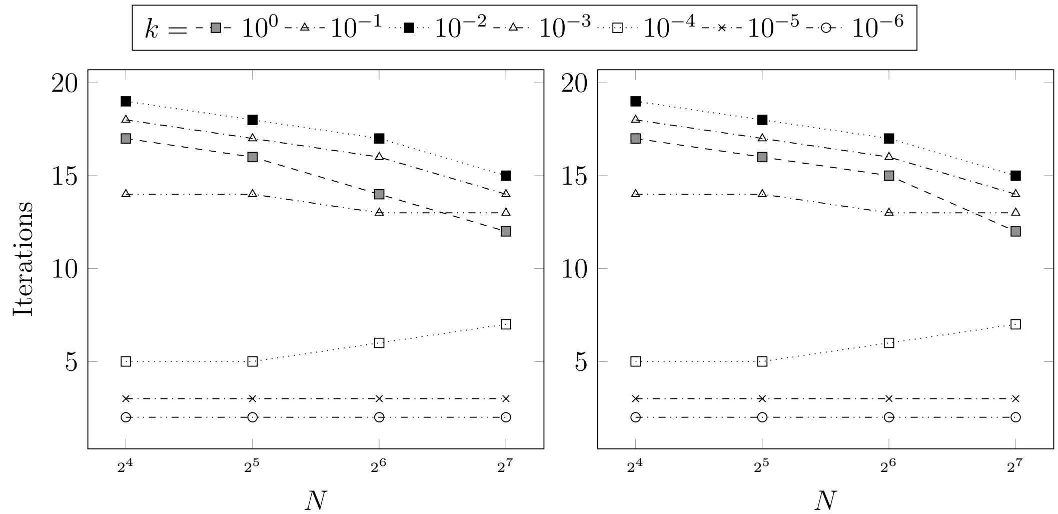
[17] D. N. Arnold, R. S. Falk, R. Winther, Finite element exterior calculus, homological techniques, and applications, Acta Numerica 15 (1) (2006) 1–155.

[18] F. Brezzi, J. D. Jr., L. D. Marini, Two families of mixed finite elements for second order elliptic problems, Numerische Mathematik 47 (2) (1985) 217–235.

[19] A. Gillette, T. Kloefkorn, Trimmed serendipity finite element differential forms, Mathematics of Computation 88 (316) (2019) 583–606.

[20] R. C. Kirby, T. T. Kieu, Symplectic-mixed finite element approximation of linear acoustic wave equations, Numerische Mathematik 130 (2) (2015) 257–291.

[21] R. C. Kirby, From functional analysis to iterative methods, SIAM Review 52 (2) (2010) 269–293.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 10: Iteration count versus mesh refinement under various k values for $C = f = 1$, $\beta = 0.1$, and $\epsilon = 0.01$ using lowest-order triangular Raviart-Thomas elements. Instead of inverting P_{V_h} by LU factorization, however, a single full multigrid cycle is used. Comparing to Figure 3 reveals a slight increase in iteration count in exchange for forgoing the sparse direct factorization.

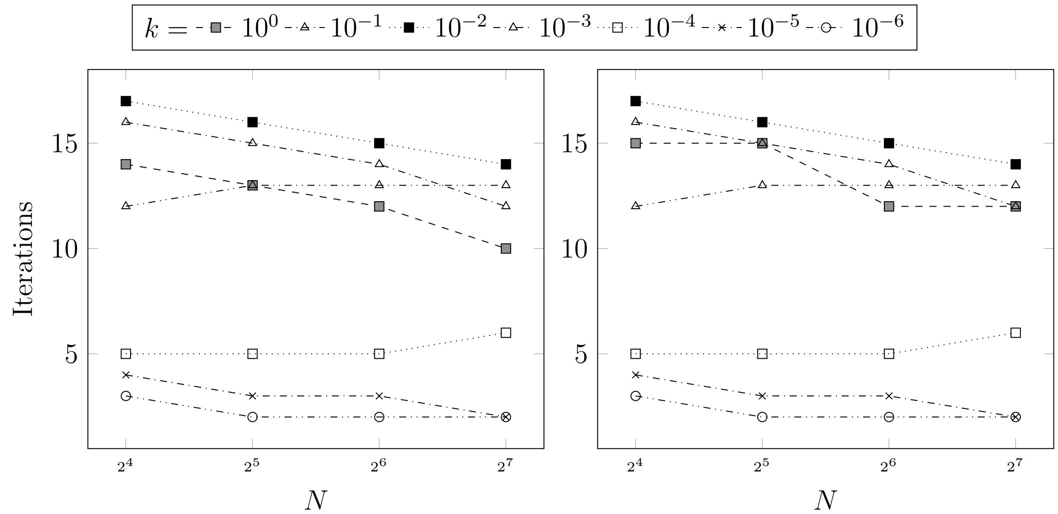
[22] K.-A. Mardal, R. Winther, Preconditioning discretizations of systems of partial differential equations, Numerical Linear Algebra with Applications 18 (1) (2011) 1–40.

[23] J. H. Adler, F. J. Gaspar, X. Hu, C. Rodrigo, L. T. Zikatanov, Robust block preconditioners for Biot's model, in: International Conference on Domain Decomposition Methods, Springer, 2017, pp. 3–16.

[24] T. Bærland, J. J. Lee, K.-A. Mardal, R. Winther, Weakly imposed symmetry and robust preconditioners for Biot's consolidation model, Computational Methods in Applied Mathematics 17 (3) (2017) 377–396.

[25] K.-A. Mardal, B. F. Nielsen, X. Cai, A. Tveito, An order optimal solver for the discretized bidomain equations, Numerical Linear Algebra with Applications 14 (2) (2007) 83–98.

[26] A. Wathen, D. Silvester, Fast iterative solution of stabilised Stokes systems. Part I: Using simple diagonal preconditioners, SIAM Journal on Numerical Analysis 30 (3) (1993) 630–649.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 11: Iteration count versus mesh refinement under various k values for $C = f = 1$, $\beta = 0.1$, and $\epsilon = 0.01$ using lowest-order Raviart-Thomas elements on squares. A full multigrid cycle using four levels, as in Figure 10, is used instead of sparse direct factorization of P_{V_h}, again resulting in a slight increase in iteration count.

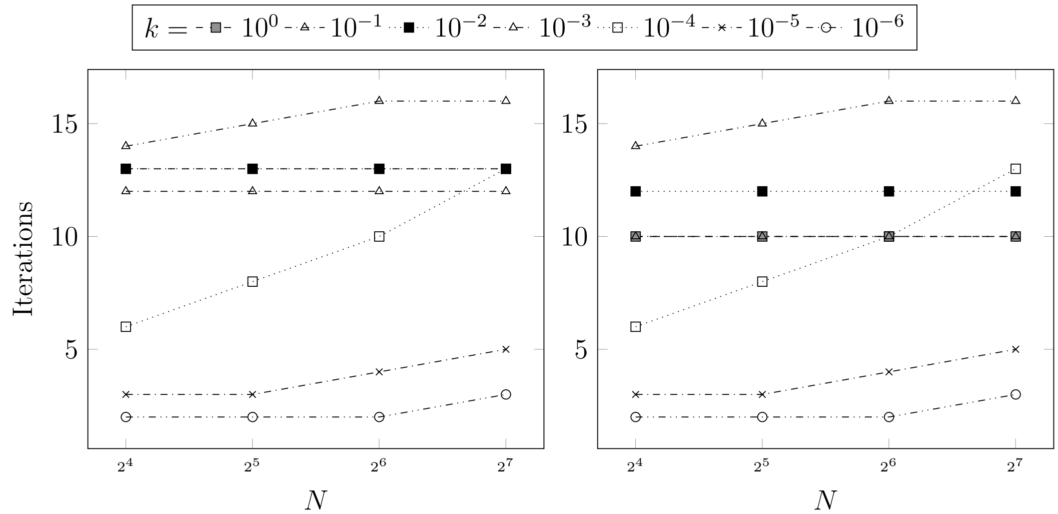
[27] V. E. Howle, R. C. Kirby, Block preconditioners for finite element discretization of incompressible flow with thermal convection, Numerical Linear Algebra with Applications 19 (2) (2012) 427–440.

[28] F. Rathgeber, D. A. Ham, L. Mitchell, M. Lange, F. Luporini, A. T. T. McRae, G.-T. Bercea, G. R. Markall, P. H. J. Kelly, Firedrake: automating the finite element method by composing abstractions, ACM Transactions on Mathematical Software 43 (3) (2016) 24:1–24:27. arXiv:1501.01809, doi:10.1145/2998441.

[29] M. S. Alnæs, A. Logg, K. B. Ølgaard, M. E. Rognes, G. N. Wells, Unified form language: A domain-specific language for weak formulations of partial differential equations, ACM Transactions on Mathematical Software (TOMS) 40 (2) (2014) 1–37.

[30] A. Gillette, T. Kloefkorn, V. Sanders, Computational serendipity and tensor product finite element differential forms, The SMAI Journal of Computational Mathematics 5 (2019) 1–21.

[31] D. N. Arnold, R. S. Falk, R. Winther, Multigrid in $H(\mathrm{div})$ and $H(\mathrm{curl})$, Numerische Mathematik 85 (2) (2000) 197–217.



(a) Using the bilinear form (20) (includes damping) as a preconditioner

(b) Using the bilinear form (33) (without damping) as a preconditioner

Figure 12: Iteration count versus mesh refinement under various k values for $f = 1$, $\beta = 0.1$, and $\epsilon = 0.01$ for the nonlinear damping law $g(u) = |u|^2 u$.

[32] T. V. Kolev, P. Vassilevski, Parallel H^1-based auxiliary space AMG solver for H(curl) problems, Tech. rep., Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States) (2006).

[33] R. C. Kirby, L. Mitchell, Solver composition across the PDE/linear algebra barrier, SIAM Journal on Scientific Computing 40 (1) (2018) C76–C98. doi:10.1137/17M1133208.

[34] P. E. Farrell, M. G. Knepley, F. Wechsung, L. Mitchell, PCPATCH: software for the topological construction of multigrid relaxation methods, arXiv preprint arXiv:1912.08516 (2019).

[35] T. Arbogast, M. R. Correa, Two families of H(div) mixed finite elements on quadrilaterals of minimal dimension, SIAM Journal on Numerical Analysis 54 (6) (2016) 3332–3356.