Many physical processes are described by elliptic or parabolic partial differential equations. For linear stability problems associated with such equations, the inverse Laplacian provides a very effective preconditioner. In addition, it is also readily available in most scientific calculations in the form of a Poisson solver or an implicit diffusive timestep. We incorporate Laplacian preconditioning into the inverse Arnoldi method, using BiCGSTAB to solve the large linear systems. Two successful implementations are described: spherical Couette flow described by the Navier-Stokes equations and Bose-Einstein condensation described by the Nonlinear Schrödinger equation.

1 Introduction

Many physical systems are governed by parabolic evolution equations of the general form

\[ \partial_t U = LU + N(U) \]  

where \( L \) is the Laplacian operator and \( N \) represents some combination of nonlinear terms or a multiplicative potential. Two examples which we will consider are the Navier-Stokes equations

\[ \partial_t U = -(U \cdot \nabla)U - \nabla P + \nu \nabla^2 U \] 
\[ \nabla \cdot U = 0 \]  

and the Nonlinear Schrödinger equation

\[ -i \partial_t \Psi = \left[ \frac{1}{2} \nabla^2 + \mu - V(x) - a|\Psi|^2 \right] \Psi \]  

The linear stability of a steady solutions \( U \) of (1) satisfying

\[ 0 = LU + N(U) \]  

is determined by the leading eigenpairs of

\[ \lambda u = Lu + Nu \]  

where \( Nu \) is the linearization or Jacobian of \( N \) evaluated at \( U \).

We use the inverse Arnoldi method [1] to calculate several leading eigenpairs, if these eigenvalues are those of smallest magnitude. Defining the Jacobian matrix

\[ A \equiv L + Nu \]  

we generate the sequence \( \{u_k \equiv A^{-(k-1)}u_1; \ k = 1, \ldots, K\} \) by solving
\[
Au_{k+1} = u_k
\]  
(7)

This sequence is orthonormalized by the usual Arnoldi process to yield the basis \( \{v_k\} \) for the Krylov space and the upper Hessenberg matrix
\[
H_{jk} \equiv \langle v_j, A^{-1}v_k \rangle
\]
(8)

\( H \) is directly diagonalized, yielding
\[
H\phi_k = \lambda_k\phi_k.
\]
(9)

with estimated eigenpairs \( \lambda_k^{-1}, V\phi_k \) for \( A \).

A shift \( s \) can be used to accelerate convergence of the Arnoldi method to a desired eigenvalue. In this case, we solve
\[
(A - sI)u_{k+1} = u_k
\]
(10)

If seeking the leading eigenvalues for a branch of steady states which depends on a parameter such as a Reynolds number, then a very accurate estimate \( s \) of the desired eigenvalue is that obtained for a neighboring parameter value.

If (7) cannot be solved to high accuracy, then we replace (8) by
\[
H_{jk} \equiv \langle v_j, Av_k \rangle
\]
(11)
to construct a Krylov space representation of \( A \) rather than of \( A^{-1} \); the eigenvalues of \( H \) are then estimates of those of \( A \). Note that we continue to generate the Krylov vectors by acting with \( A^{-1} \) via (7). Thus, we continue to target the method to eigenvalues closest to 0 or to \( s \). Assuming that action with \( A \) is inexpensive, carrying out additional multiplications by \( A \) to generate \( H \) adds little to the cost, while eliminating an additional source of inaccuracy.

Our implementation of the Arnoldi method is as follows. Choosing a small value of \( K \), typically \( 2 \leq K \leq 6 \), and an initial random vector \( u_1 \), we take \( K \) Arnoldi steps, generating the \( K \) Krylov vectors \( \{u_k\} \), the \( K \times K \) matrix \( H \), and the \( K \) eigenpair estimates. To continue, we take one additional Arnoldi step, discard \( u_1 \), redefine \( u_k \leftarrow u_{k+1} \), and generate an updated \( H \) and eigenpair estimates. The procedure is halted when the the residual \( \| (A - \lambda_k^{-1}I)V\phi_k \| \) or \( \| (A - \lambda_kI)V\phi_k \| \) of the eigenpair sought is sufficiently small.

2 Laplacian preconditioning

Our method for solving (7) is based on the BiCGSTAB variant of the conjugate gradient method [2]. Since \( A \) results from the spatial discretization of a partial differential equation, its size may be quite large. Denoting by \( M \) the number of points or modes necessary to represent the variation in each spatial dimension \( D \), we assume that \( 10 \leq M \leq 1000 \). The size of \( A \) is then \( 10^D \leq M^D \leq 10^{3D} \), i.e. \( 10^2 \leq M^2 \leq 10^6 \) for problems with two spatial dimensions and \( 10^3 \leq M^3 \leq 10^9 \) for three-dimensional problems. In addition, \( A \) is poorly conditioned, primarily because of the wide range of eigenvalues of the Laplacian \( L \). The smallest and largest eigenvalues of \( L \) can be estimated roughly as \( -1 \) and \( -D M^2 \), yielding a condition number for \( L \) of \( 3 \times 10^4 \) for a three-dimensional case with \( M = 100 \). Thus, preconditioning by the inverse \( L^{-1} \) of the Laplacian will
be very effective. In addition, multiplication by the inverse Laplacian, i.e. solution of the Poisson equation, is a ubiquitous problem for which a great deal of computational technology has been developed, for all sorts of spatial discretizations.

In the two examples we have implemented, we have used a pseudospectral spatial discretization [3]. Functions are represented both as series of basis functions, such as Fourier series or Chebyshev polynomials (spectral representation), and also by their values on a spatial grid. Actions and inversions of the Laplacian $L$ are carried out in the spectral representation, while the actions of the multiplicative operator $N$ or $N_U$ are carried out on the grid representations; all of these operations scale approximately linearly in $M^D$, the number of gridpoints or basis functions. Fourier or Chebyshev transforms are used to pass between the spectral and grid representations in a time proportional to $D M^D \log M$.

The problem we solve instead of (7) uses as a preconditioner either the Poisson operator $L^{-1}$:

$$L^{-1}(L + N_U)u_{k+1} = L^{-1}u_k$$

or the Helmholtz operator $(I - \Delta tL)^{-1}$:

$$(I - \Delta tL)^{-1}(L + N_U)u_{k+1} = (I - \Delta tL)^{-1}u_k$$

(13)

Appropriate boundary conditions must be imposed on either equation; this is assumed to be encompassed in the notation $L^{-1}$ or $(I - \Delta tL)^{-1}$. Use of (13) instead of (12) is motivated by the utilization of timestepping schemes in which the evolution of the diffusive terms is calculated implicitly. This is necessitated precisely because the wide range of eigenvalues of $L$ leading to poor conditioning in the context of the linear system (7), leads to stiffness of the evolution equation (1).

The simplest such timestepping scheme is the backward-Euler/forward Euler first-order algorithm:

$$U(t + \Delta t) = (I - \Delta tL)^{-1}(I + \Delta tN)U(t)$$

(14)

We adapt (14) to the linearized operator by substituting $N_U$ for $N$. The difference between two consecutive linearized timesteps can be written:

$$U(t + \Delta t) - U(t) = (I - \Delta tL)^{-1}(I + \Delta tN_U)U(t) - U(t)$$

$$= (I - \Delta tL)^{-1}[(I + \Delta tN_U) - (I - \Delta tL)]U(t)$$

$$= (I - \Delta tL)^{-1}\Delta t(N_U + L)U(t)$$

(15)

which is seen to be the action of the operator on the left hand side of (13). In order for the Helmholtz operator $(I - \Delta tL)^{-1}$ to be an effective preconditioner, $\Delta t$ must be set to a large value, in contrast to the small value required for timestepping. Varying $\Delta t$ can also provide a way of testing the preconditioning, since $\Delta t \to 0$ is equivalent to no preconditioning, while $\Delta t \to \infty$ is equivalent to preconditioning by $L^{-1}$. See also [4, 5, 6, 7, 8].

When a shift is used, equations (12) and (13) are replaced by

$$L^{-1}(L + N_U - sI)u_{k+1} = L^{-1}u_k$$

(16)

$$I - \Delta tL)^{-1}(L + N_U - sI)u_{k+1} = (I - \Delta tL)^{-1}u_k$$

(17)

This method then consists of a sequence of outer Arnoldi iterations, each of which requires a sequence of inner BiCGSTAB iterations. There exists an inherent conflict between the outer Arnoldi
and inner BiCGSTAB iterations: the Arnoldi method should converge fastest when the eigenvalues differ most, while BiCGSTAB should converge fastest when the matrix is well conditioned. We will see in the applications that this conflict posed no practical difficulty in the case of our spherical Couette flow problem (Navier-Stokes equations), but may be responsible for problems encountered in the case of our Bose-Einstein condensation problem (the Nonlinear Schrödinger Equation).

3 Application to spherical Couette flow

3.1 Physical description of flow and instabilities

Spherical Couette flow is the flow between two concentric differentially rotating spheres. When the outer sphere is held fixed, spherical Couette flow is characterized by two dimensionless quantities, the Reynolds number \( Re \equiv \Omega_1 r_1^2 / \nu \) and the gap ratio \( \sigma \equiv (r_2 - r_1) / r_1 \) where \( r_1, r_2 \) are the inner and outer radii, \( \Omega_1 \) is the angular velocity of the inner sphere, and \( \nu \) is the kinematic viscosity. Like the better known cylindrical Couette flow, spherical Couette flow undergoes an instability as \( Re \) is increased, which leads to the formation of vortices. The physical mechanism responsible for the instability is the radial gradient in angular momentum, which is decreased by the radial mixing of fluid engendered by the vortices. One measure of this gradient is the torque required to rotate the inner sphere at angular velocity \( \Omega_1 \) or, equivalently for a steady flow, to keep the outer sphere stationary.

Extensive studies [9, 10, 11, 12, 13, 14] of the case \( \sigma = 0.18 \) have led to the following conclusions: In the range \( Re < 850 \), all steady states are axisymmetric, i.e. independent of the angle about the axis of rotation, and reflection-symmetric about the equator. For brevity, we describe as symmetric or antisymmetric the steady states and eigenvectors which are reflection-symmetric or antisymmetric about the equator, as well as the corresponding eigenvalues. Those which are neither are called asymmetric. There exist three types of steady states: the zero-vortex state, with no vortices, the one-vortex state, with one vortex in each hemisphere, and the two-vortex state with two vortices in each hemisphere.

The steady solutions obtained by gradually increasing \( Re \) from \( Re = 0 \) are located on what can be termed the basic branch. Along the basic branch, the zero-vortex state evolves continuously into the two-vortex state. Since the vortices are infinitesimal at onset, it is difficult to define a precise criterion for when this occurs, but \( Re \approx 735 \). The zero- and two- vortex states along the basic branch are unstable over the range \( 650 < Re < 775 \). The eigenvalue responsible for this instability is real, and the corresponding eigenvector is antisymmetric. The endpoints of this interval correspond to subcritical pitchfork bifurcations, which means in this case that the asymmetric bifurcating branches originating at \( Re = 650, 775 \) are unstable and not the final destinations of the transitions triggered by the instability. Instead, evolution via a sequence of asymmetric transient states terminates at a steady stable symmetric one-vortex state.

Figure 1 shows up to four leading eigenvalues of the basic flow, calculated using the inverse Arnoldi method. The eigenvalue which is positive over the range \( 650 < Re < 775 \) is that responsible for the subcritical pitchfork bifurcations initiating transition to the one-vortex state described above. This is the leading eigenvalue and it is antisymmetric. For \( Re < 744 \), the next leading eigenvalues are a symmetric complex conjugate pair whose real and imaginary parts are both shown in figure 1. At \( Re = 744 \), these coalesce and become two real eigenvalues, the lower of
which decreases so rapidly with $Re$ that it is no longer visible on figure 1 for $Re > 755$. The last eigenvalue shown on figure 1 belongs to a second antisymmetric eigenvector.

The leading eigenvalue attains a maximum at $Re = 735$, precisely at the value where the torque is minimum and very near the value at which the basic flow evolves from a zero-vortex to a two-vortex flow. All of the other leading eigenvalues (or, in the case of the complex conjugate pair, their real part) also have maxima near, though not exactly at, this Reynolds number. This reflects the fact that the angular momentum gradient responsible for the instability has been alleviated by the radial fluid mixing of the vortices. In evolving from a zero-vortex to a two-vortex flow, the basic branch becomes less unstable and the eigenvalues decrease.

Figure 1: Four leading eigenvalues for spherical Couette flow. Corresponding to antisymmetric eigenvectors are two eigenvalues (circles), one of which is positive over $650 < Re < 775$. Corresponding to symmetric eigenvectors are two eigenvalues (squares) which form a complex conjugate pair for $Re < 744$; both real (filled squares) and imaginary (hollow squares) parts are shown.
3.2 Numerical results

We now describe the results of eigenvalue computations for this case of spherical Couette flow. Approximately 50 lines were added to an existing time-stepping program [13] to implement the method. This program uses a tensor-product basis set (Chebyshev polynomials in radius multiplied by trigonometric functions of meridional angle) to represent fields. This leads to a Laplacian which is highly structured (although not sparse) and thus to rapid action with $(I - \Delta t L)^{-1}$. The azimuthal velocity and the meridional streamfunction are used to represent the axisymmetric fields and incompressibility is imposed to machine accuracy via the influence matrix technique. The program was previously adapted to calculate the leading eigenpair via the simple power method [14] or Arnoldi’s method [4] on the approximate exponential (14), and also to calculate steady states by Newton’s method with Laplacian preconditioning [4] (called Stokes preconditioning in this context). The numerical resolution usually used is $16 \times 128$ which, with two fields, leads to matrices of size 4096.

We focus on the calculation of the eigenvalue of smallest magnitude at Reynolds number 750, whose value is $\lambda = -0.15181122$. Krylov spaces of dimension $K = 2$ are used for the Arnoldi iterations. To solve the linear systems, BiCGSTAB is given a stopping criterion of

$$\|Au_{k+1} - u_k\|/\|u_k\| \leq 10^{-7},$$

with a maximum number of iterations of 2000. We measure CPU time by the number of matrix-vector multiplications, each approximately equivalent to a timestep. We recall that for this pseudospectral code, the cost of such a multiplication increases only slightly faster than linearly in the number of gridpoints of basis functions. Figure 2 shows the convergence of the error $|\lambda - \bar{\lambda}|$ as a function of the number $n$ of matrix-vector multiplications. Each point corresponds to a single Arnoldi iteration, i.e. a single solution of (7) or (13), and thus to a sequence of BiCGSTAB iterations. The timestep $\Delta t$ of the Helmholtz preconditioner $(I - \Delta t L)^{-1}$ is varied from 0.01 to 100. Very fast convergence is seen for $\Delta t = 10$. Six-digit accuracy is obtained after 7 Arnoldi steps, requiring $n = 754$ matrix-vector multiplies. Further increase of $\Delta t$ has little effect, as shown by the similar values corresponding to $\Delta t = 100$; this shows that the preconditioning operator is essentially the inverse Laplacian $L^{-1}$. As $\Delta t$ is decreased, the preconditioning also decreases drastically. For $\Delta t = 1$, the number of matrix-vector multiplications required by BiCGSTAB to converge to the same accuracy increases from 100 to 300, while for $\Delta t = 0.1$, this number is on the order of 600. For $\Delta t = 0.01$, convergence would require more than the maximum number of iterations we have allowed for BiCGSTAB.

We quantify the dependence of convergence on $\Delta t$ further by plotting the same data logarithmically in the number of matrix-vector multiplications. Each sequence $|\lambda - \bar{\lambda}|(n)$ has approximately the same slope. The sequences are displaced rightwards by the same interval as $\Delta t$ is decreased by factors of 10 from 10 to 1 to 0.1. This implies that

$$|\lambda - \bar{\lambda}|(n, \Delta t) \sim (n\Delta t^\alpha)^{-\beta}$$

and figure 2 yields the estimated dependence

$$|\lambda - \bar{\lambda}|(n, \Delta t) \sim (n\Delta t^{1/4})^{-14}$$

Figure 3 shows the convergence of $|\lambda - \bar{\lambda}|$ as shifts progressively closer to $\bar{\lambda}$ are employed, more specifically $s = 0$, $s = -0.1$, $s = -0.15$, and $s = -0.152$. Two unexpected conclusions can
be drawn from figure 3. The first is that as \( s \) approaches \( \hat{\lambda} \), convergence continues to improve, despite the fact that the matrix must become less well conditioned as \( s \to \hat{\lambda} \). The second is that convergence does not improve monotonically as \( s \) approaches \( \hat{\lambda} \); convergence for \( s = -0.1 \) is slower than that for \( s = 0 \). In seeking to understand this, we notice that BiCGSTAB requires fewer matrix-vector multiplications for each Arnoldi iteration for the case \( s = -0.1 \) than for the case \( s = 0 \). We therefore forced BiCGSTAB to increase the number of matrix-vector multiplications by reducing its stopping criterion from \( 10^{-7} \) to the more stringent value of \( 10^{-9} \). Figure 3 shows that this change greatly improves the convergence of the \( s = -0.1 \) case but has little effect on the \( s = 0 \) case. This suggests that the outer Arnoldi and inner BiCGSTAB iterations are intermeshed in a way which is more complicated to capture than by the simple value of the stopping criterion. We have not explored the effect of varying \( s \) from one Arnoldi iteration to the next.

Figure 4 shows that convergence is almost unaffected by an increase in the size of the Krylov space from \( K = 2 \) to \( K = 4 \). We also increased the spatial resolution from \( 2 \times 16 \times 128 \) to \( 2 \times 32 \times 256 \). This increases the size of the matrix by a factor of 4 and the cost of each matrix-vector multiplication by slightly more than a factor of 4, and changes the exact value of the eigenvalue to \( \lambda = -0.15197992 \). We see, however, that the dependence of \( \| \lambda - \hat{\lambda} \| \) on the number of matrix-vector multiplications is almost unaffected. This demonstrates our claim that the cost of this method is approximately linear in the number of gridpoints or modes, i.e. in the size of the matrix.

Figure 2: Left: Convergence of error \( |\lambda - \hat{\lambda}| \) as a function of the number of matrix-vector multiplications for different values \( \Delta t \) in the Helmholtz preconditioner \((I - \Delta t L)^{-1}\). For \( \Delta t = 100 \) (solid circles) or \( \Delta t = 10 \) (hollow circles), inverse Laplacian preconditioning is achieved and convergence is rapid. As \( \Delta t \) is decreased through 1 (solid triangles) and 0.1 (hollow triangles), convergence decreases dramatically. For \( \Delta t = 0.01 \), the preconditioner has become so ineffective that BiCGSTAB does not converge to the requested precision in the maximum number of iterations allowed. Right: Same data in log-log representation indicates that \( |\lambda - \hat{\lambda}| \propto (n, \Delta t)^{\alpha} \) with \( \alpha \approx 1/4 \) and \( \beta \approx 14 \).
Figure 3: Left: Convergence of error $|\lambda - \bar{\lambda}|$ as a function of the number of matrix-vector multiplications for different values of shift $s$, with $\Delta t$ fixed at 100 and BiCGSTAB stopping criterion fixed at $10^{-7}$. As the shift approaches $\bar{\lambda} = -0.15181122$ from $s = 0$ (solid circles), through $s = -0.1$ (hollow circles), $s = -0.15$ (solid triangles), to $s = -0.152$ (hollow triangles), convergence greatly accelerates except for $s = -0.1$. Right: Convergence of error $|\lambda - \bar{\lambda}|$ for $s = 0$ (solid) and $s = -0.1$ (hollow) and for BiCGSTAB stopping criterion $10^{-7}$ (circles) and $10^{-9}$ (squares). Reducing the stopping criterion improves convergence for $s = -0.1$ but has little effect for $s = 0$.

Figure 4: Convergence of error $|\lambda - \bar{\lambda}|$ as Krylov space is increased from $K = 2$ to $K = 4$ (hollow triangles) and as spatial resolution is increased from $2 \times 16 \times 128$ to $2 \times 32 \times 256$ (solid triangles).
4 Bose-Einstein Condensation

The Nonlinear Schrödinger equation (3), also called the Gross-Pitaevskii equation [15, 16], has been used to described the behavior of a Bose-Einstein condensate [17, 18, 19, 20, 21], in which atoms are cooled so drastically that they populate the same quantum-mechanical state.

The steady states are a stable (elliptic) and unstable (hyperbolic/elliptic) pair which meet at a Hamiltonian saddle-node bifurcation [22, 23, 24] described as follows. The eigenvalues occur in pairs $\pm \lambda$ or $\pm i\lambda$. Along the elliptic branch, all are imaginary. As this branch is followed towards the Hamiltonian saddle-node bifurcation, one imaginary eigenvalue pair $\pm i\lambda$ approaches zero, becoming zero at the saddle-node bifurcation. As we leave the bifurcation along the unstable hyperbolic branch, the eigenvalue pair $\pm \lambda$ is real, with absolute value increasing along the branch. The rate at which the critical eigenvalue $|\lambda|$ approaches and recedes from zero determines the rate at which the Bose-Einstein condensate decays [23, 24].

The low temperature needed for Bose-Einstein condensation is modeled by a confining harmonic potential $\frac{1}{2}|\omega \cdot x|^2$. Two types of calculations of the steady states and eigenvalues have previously been carried out. First, a variational technique which approximates steady states as Gaussians yields analytic estimates of the critical eigenvalues [22, 23, 24, 25]. Second, if the potential is isotropic (spherically symmetric), and this assumption is made throughout, then the problem has effectively only one spatial dimension and the stability matrix is small enough to be directly diagonalized [24]. We describe an implementation of the inverse Arnoldi method with Laplacian preconditioning which calculates the eigenvalues in a general geometry and present results for the two cylindrically symmetric potentials, termed a cigar and a pancake [25, 26, 27].

We write the Nonlinear Schrödinger equation in the abbreviated form:

$$-i\partial_t \Psi = L\Psi + W(\Psi) \quad (21)$$

where

$$L\Psi \equiv \frac{1}{2}\nabla^2 \Psi \quad (22)$$

$$W(\Psi) = \left[\mu - \frac{1}{2}|\omega \cdot x|^2 - a|\Psi|^2\right] \Psi \quad (23)$$

$$|\omega \cdot x|^2 = \omega_r r^2 + \omega_z z^2 \quad (24)$$

We begin by presenting our results for two geometries, a cigar ($\omega_z = \omega_r/5$), and a pancake ($\omega_z = \omega_r/5$) [25, 26, 27] in figure 5. The particle number $N$ and energy $E$ are defined by:

$$N = \int d^3x |\Psi|^2 \quad (25)$$

$$E = \int d^3x \left[\frac{1}{2}|\nabla \Psi|^2 + \frac{1}{2}|\omega \cdot x|^2|\Psi|^2 + \frac{a}{2}|\Psi|^4\right] \quad (26)$$

The control parameter is the particle number, which is conserved by the Nonlinear Schrödinger equation. (The solutions can also be indexed by $\mu$.) When the particle number is below $N_c$, there exist two solutions, one stable (elliptic) and the other unstable (hyperbolic). The Hamiltonian saddle-node bifurcation takes place at $N_c$, and for a particle number exceeding $N_c$, no Bose-Einstein condensate exists.
Figure 5: Stationary solutions of the GP equation as a function of particle number $N$ for two non-isotropic potentials with $\omega_z = \omega_r/5$ (cigar, left) and $\omega_r = \omega_z/5$ (pancake, right). Top: value of the energy functional $\mathcal{E}_+$ on the unstable (hyperbolic) branch and $\mathcal{E}_-$ on the stable (elliptic) branch. Bottom: square of the bifurcating eigenvalue ($\lambda_2^2$); $|\lambda_-|$ is the energy of small excitations around the stable branch. Solid lines: exact solution of the GP equation. Dashed lines: Gaussian approximation.

The operators $L$ and $W$ defined in (22) and (23) are spatially discretized using the pseudospectral method. We assume a three-dimensional periodic Cartesian domain, on which $\Psi$ and $|\omega \cdot \mathbf{x}|^2$ are expanded as three-dimensional trigonometric (Fourier) series. In this representation, solution to the Poisson equation is trivial, since each Fourier component is merely divided by the square of its wavenumber:

$$
\nabla^2 \sum_{l_x,l_y,l_z} f_{l_x,l_y,l_z} \exp(\imath l_x x + \imath l_y y + \imath l_z z) = - \sum_{l_x,l_y,l_z} |l|^2 f_{l_x,l_y,l_z} \exp(\imath l_x x + \imath l_y y + \imath l_z z) \quad (27)
$$

(The $|l| = 0$ component is determined by the boundary conditions in solving the Poisson equation and can be treated arbitrarily when $L^{-1}$ is used merely as a preconditioner.) The resolution $M$ in each direction is 50 or 100, so the total number of gridpoints or trigonometric functions is as high as $10^6$. The time required for action by $L$ or $L^{-1}$ is again proportional to the number of gridpoints or modes, while action by $W$ includes a Fourier transform and so takes a time proportional to $M^3 \log M$. Stable and unstable steady states were previously obtained [24] by adapting a time-stepping code to carry out Newton’s method, using Laplacian preconditioning and BiCGSTAB to solve the resulting linear systems.

In order to correctly formulate the linear stability problem, it is necessary to decompose the eigenvector $\psi = \psi^R + \imath \psi^I$. We have

$$
W \psi = W^R \psi^R + \imath W^I \psi^I \quad (28)
$$

where

$$
W^R = \mu - \frac{1}{2} |\omega \cdot \mathbf{x}|^2 - 3a\Psi^2 \quad (29a)
$$

$$
W^I = \mu - \frac{1}{2} |\omega \cdot \mathbf{x}|^2 - a\Psi^2 \quad (29b)
$$
The equation governing the eigenmodes of (21) is:

\[
\lambda \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix} = \begin{bmatrix} 0 & -(L + W^I_\psi) \\ L + W^R_\psi & 0 \end{bmatrix} \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix}
\]  
(30)

It is more convenient to work with the square of the matrix in (30):

\[
\lambda^2 \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix} = \begin{bmatrix} -(L + W^I_\psi)(L + W^R_\psi) & 0 \\ 0 & -(L + W^R_\psi)(L + W^I_\psi) \end{bmatrix} \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix}
\]  
(31)

Because (31) is block diagonal, it can be separated into the two problems:

\[
\lambda^2 \psi^R = -(L + W^I_\psi)(L + W^R_\psi)\psi^R  
\]  
(32a)

\[
\lambda^2 \psi^I = -(L + W^R_\psi)(L + W^I_\psi)\psi^I  
\]  
(32b)

Problems (32a) and (32b) are closely related, since if \( \psi^R \) is an eigenvector of (32a) with eigenvalue \( \lambda \), then \( (L + W^R_\psi)\psi^R \) is an eigenvector of (32b) with the same eigenvalue. Thus, we solve only (32a). As the critical eigenvalue pair \( \pm \lambda \) makes the transition from imaginary to real, \( \lambda^2 \) passes from negative to positive.

To generate the Krylov space, we shift the operator in (32a) and precondition with the square of the inverse Laplacian:

\[
L^{-2}[-(L + W^I_\psi)(L + W^R_\psi) - sI]\psi_{k+1} = L^{-2}\psi_k
\]  
(33)

For this problem, we find that it is better to construct \( H \) as an approximation to \( A \) via (11) rather than as an approximation to \( A^{-1} \) via (8). The inverse Arnoldi method usually converges in 3 to 10 iterations, each of which requires several hundred BiCGSTAB iterations (matrix-vector multiplications) in order to solve its associated linear system. We adjust \( s \) empirically, both to target the critical eigenvalue and also to improve BiCGSTAB convergence. For some cases, we find that these goals are incompatible: with the shift required for the desired eigenvalue, BiCGSTAB becomes unable to converge as the Arnoldi iteration progresses, even when allowed a very large number of matrix-vector multiplies. This is the reason that the branch of eigenvalues calculated for the cigar case terminates prematurely in figure 5. We are currently exploring reasons and remedies for this phenomenon.

References


