Numerical methods for bifurcation problems

Laurette S. Tuckerman
Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur (LIMSI-CNRS), BP 133, 91403 Orsay Cedex, France
email: laurette@limsi.fr

Cristian Huepe
James Franck Institute, University of Chicago
5640 S. Ellis Ave., Chicago IL 60637, USA

Marc-Etienne Brachet
Laboratoire de Physique Statistique de l'Ecole Normale Supérieure
24 Rue Lhomond, 75231 Paris, France

Introduction

Most physical systems are governed by evolution equations of the general form

$$\frac{\partial \Psi}{\partial t} = L\Psi + W(\Psi) \tag{1}$$

where L is the Laplacian operator and W represents some combination of multiplicative and nonlinear terms. Some examples are:

$$\frac{\partial U}{\partial t} = -(U\cdot\nabla)U - \nabla P + \nu\nabla^2 U \qquad \text{(Navier-Stokes)} \tag{2}$$

$$\frac{\partial A}{\partial t} = \mu A - |A|^2 A + \nabla^2 A \qquad \text{(Ginzburg-Landau)} \tag{3}$$

$$-i\frac{\partial \Psi}{\partial t} = \left[\tfrac{1}{2}\nabla^2 + \mu - V(\mathbf{x}) - a|\Psi|^2\right]\Psi \qquad \text{(Nonlinear Schrödinger)} \tag{4}$$

as well as many other systems, such as the usual Schrödinger equation, reaction-diffusion equations, and the complex Ginzburg-Landau equation. Although the physical system evolves according to the time-dependent equations (1), valuable insight may be gained by studying the closely related equations

$$0 = L\Psi + W(\Psi) \tag{5}$$

and

$$\lambda\psi = L\psi + DW(\Psi)\psi \tag{6}$$

where $DW(\Psi)$ is the linearization or Jacobian of W evaluated at Ψ. (5) describes the steady states of (1) while (6) describes the eigenmodes of (1) about a steady state Ψ.

This contribution is a tutorial describing the physical motivations and the numerical methods for solving (5) and (6). Additional details concerning these methods, as well as various physical applications, can be found in [1, 2, 3]. Here, we will use the Nonlinear Schrödinger equation (4), also called the Gross-Pitaevskii equation [4, 5], as an illustrative example. The Nonlinear Schrödinger equation has been used to described the behavior of a Bose-Einstein condensate [6, 7, 8, 9, 10], in which atoms are cooled so drastically that they populate the same quantum-mechanical state. The steady states, solutions of (5), are a stable (elliptic) and unstable (hyperbolic/elliptic) pair which meet at a Hamiltonian saddle-node bifurcation [11, 12, 13]. The rate at which the critical eigenvalue $|\lambda|$ described by (6) approaches and recedes from zero determines the rate at which the Bose-Einstein condensate decays [12, 13].

Steady states via Newton's method

While simulation of the time-dependent equations (1) converges to steady states of (5), there are two advantages to be gained in solving (5) rather than (1). The first is practical: the solutions of (5) can be obtained far more rapidly and accurately than by time-dependent simulation of (1). The second advantage is theoretical: (5) describes the steady states independent of their stability, whereas (1) converges only to stable steady states. Unstable steady states can give valuable information concerning the bifurcation structure of a problem, much as the extension of a real function into the complex plane explains the origin of singularities.

The best way to solve (5) is by Newton's method or one of its variants. Starting with an approximation Ψ, we search for a decrement ψ which will make $\Psi - \psi$ into a steady state. Linearizing (5) about Ψ, we write

$$\begin{aligned} 0 &= L(\Psi - \psi) + W(\Psi - \psi) \\ &\approx L\Psi - L\psi + W(\Psi) - DW(\Psi)\psi \qquad (7a) \\ \Psi &\leftarrow \Psi - \psi \qquad (7b) \end{aligned}$$

where $DW(\Psi)$ is the Jacobian of W linearized about Ψ. Solving the linear system

$$(L + DW(\Psi))\psi = L\Psi + W(\Psi) \qquad (8)$$

while straightforward in principle, poses the main computational challenge. We assume that the vectors Ψ have been spatially discretized on M gridpoints or basis functions, as well as the operators L, W, and DW. We define the M by M Jacobian matrix

$$A \equiv L + DW(\Psi) \qquad (9)$$

When A results from the spatial discretization of a partial differential equation such as (2), (3), or (4), M can be quite large. Each spatial dimension D requires between 10 and 100 points or modes. Hence, $10^D \leq M \leq 10^{2D}$, i.e. $10^2 \leq M \leq 10^4$ for problems with two spatial dimensions and $10^3 \leq M \leq 10^6$ for three-dimensional problems. The two ways to solve linear equations are termed direct and iterative. A direct method essentially uses Gaussian elimination, also called LU decomposition and backsolving, to solve (8), and requires a time proportional to M^3. An iterative method acts carries out an inexpensive procedure repeatedly in order to gradually improve an approximation to the solution.

Here we confine ourselves to conjugate gradient methods and their variants. The principle of these methods is simple: one acts repeatedly with the matrix A an initial vector, typically the right-hand-side, and constructs the approximate solution as a superposition of the resulting vectors. For symmetric definite systems, i.e. matrices which are self-adjoint under the inner product used and whose eigenvalues are all of the same sign, the conjugate gradient algorithm and its properties are well known. However, for matrices which violate one or both of these properties, there exist many variants of the conjugate gradient method, none of which is demonstrably superior in all cases to all the others [14]. The analysis of conjugate gradient methods for non-symmetric definite systems is very complicated and small variations in the algorithm may have far-reaching consequences. This is one of the major problems in numerical linear algebra. We use the BiCGSTAB (Bi Conjugate Gradient Stabilized) [15], which has enabled us obtain accurate solutions of many problems of the type (8) in on the order of several hundred iterations.

In the worst case, a matrix-vector multiplication requires a time proportional to M^2 and M multiplications are required in order to generate M linearly independent directions. Thus, conjugate gradient and direct methods should in general take the same time. However, the time taken by conjugate gradient methods to solve (8) is usually much less, for two

reasons. The first reason arises from the structure of the matrix. The matrix resulting from the spatial discretisation of a partial differential equation is sparse and/or regular. Thus, multiplication by it typically requires on the order of M, rather than M^2 operations. The second reason lies in the distribution of eigenvalues and eivenvectors of the matrix. If the eigenvalues of a matrix are clustered, a property called *well-conditioned*, then a small number of vectors generated by matrix-vector multiplication is sufficient to represent the solution to (8). However, the matrix resulting from the spatial discretisation of a partial differential equation is usually *not* well-conditioned, causing solution of (8) by a conjugate gradient method to take on the order of M iterations, or even to break down altogether.

The remedy for poor conditioning of a matrix is *preconditioning*. Both sides of (8) are multiplied by a matrix P which resembles the inverse of A, but is inexpensive to act with. Extreme illustrative examples are $P = I$, which is trivial to act with, but ineffectual and $P = A^{-1}$, which is extremely effective, but imposssibly expensive by assumption. One seeks a P between these two extremes.

In the case of operators such as (9), it is the Laplacian operator which is responsible for the poor conditioning. Preconditioning by the inverse L^{-1} of the Laplacian can thus 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. For the case of a Cartesian pseudospectral representation, solution to the Poisson equation is trivial, since each Fourier component is merely divided by the square of its wavenumber:

$$\nabla^2 \sum_{k_x,k_y,k_z} f_{k_x,k_y,k_z} \exp(ik_x x + ik_y y + ik_z z) = - \sum_{k_x,k_y,k_z} |k|^2 f_{k_x,k_y,k_z} \exp(ik_x x + ik_y y + ik_z z) \tag{10}$$

(The $|k| = 0$ 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.)

Another possibility for a preconditioner is the inverse Helmholtz operator $(L + cI)^{-1}$. With $c = -1/\Delta t$, this is the operator used in implicit time stepping of the Laplacian and is thus already available in most time-dependent codes. The timestep Δt must be taken to be large in order for preconditioning to be effective, unlike the small values of Δt appropriate for timestepping. More precisely, consider the first-order mixed explicit-implicit algorithm for time-stepping (1):

$$\Psi(t + \Delta t) = (I - \Delta t L)^{-1}(I + \Delta t W)\Psi(t) \tag{11}$$

For the linearized equation

$$\frac{\partial \psi}{\partial t} = (L + DW(\Psi))\psi = A\psi \tag{12}$$

whose exact solution is

$$\psi(t + \Delta t) = \exp((L + DW(\Psi))\Delta t)\psi(t) = \exp(A\Delta t)\psi(t) \tag{13}$$

algorithm (11) is:

$$\psi(t + \Delta t) = (I - \Delta t L)^{-1}(I + \Delta t DW(\Psi))\psi(t) \tag{14}$$

We now note that system (8) with the preconditioner $(\Delta t)^{-1}(I - \Delta t L)^{-1}$

$$(\Delta t)^{-1}(I - \Delta t L)^{-1}(L + DW(\Psi))\psi = (\Delta t)^{-1}(I - \Delta t L)^{-1}(L + W)\Psi \tag{15}$$

is equivalent to

$$\left[(I - \Delta t L)^{-1}(I + \Delta t DW(\Psi)) - I\right]\psi = \left[(I - \Delta t L)^{-1}(I + \Delta t W) - I\right]\Psi \tag{16}$$

The right-hand-side is the difference $\Psi(t+\Delta t) - \Psi(t)$ between two consecutive (but widely spaced) timesteps (11) while the left-hand-side is the difference $\psi(t+\Delta t) - \psi(t)$ between two linearized timesteps (14).

Newton's method can only be used when a good initial estimate is available, as is the case when mapping out a branch of solutions which depends on some control parameter, and when an initial solution on the branch is available by some other means, such as time integration. Newton's method converges either very quickly – on the order of 3–5 iterations of (7) – or not at all. If convergence is not achieved, then smaller steps in the control parameter must be taken. Specialized algorithms are used to go around a saddle-node bifurcation or to switch branches; see [1, 2, 3] and references therein.

Linear stability analysis via the Arnoldi method

The eigenmodes described by equation (6) provide quantitative information about the transitions undergone by (1). In contrast to the search for steady states of a nonlinear equation (5), there exists a direct algorithm for solving (6) to find their eigenmodes. The direct algorithm is construction and diagonalization of the Jacobian matrix via the QR algorithm. This algorithm, like the direct LU decomposition for matrix inversion, takes a time proportional to M^3, where M is the size of the matrix A of (9). When A is derived from discretization of a partial differential equation with one spatial dimension, then it can be diagonalized directly. If there are three spatial dimensions, then direct solution is not possible. In the intermediate case of two spatial dimensions, the feasibility of direct diagonalization at the present time depends on the details of the problem and on the computational resources available.

Here, we describe the Arnoldi method, which iteratively calculates several eigenvalue--eigenvector pairs. Like the conjugate gradient method for solving linear systems, and like the elementary *power method* for calculating a single eigenpair, the Arnoldi method requires only repeated multiplication by matrix A. Starting from an initial arbitrary vector u_1, a sequence of vectors

$$\{u_k \equiv A^{k-1}u_1; \quad k = 1, \ldots K\} \tag{17}$$

is generated, where K is on the order of 2 to 20, usually 4 or 6. This sequence is orthonormalized to yield a basis $\{v_k; \quad k = 1, \ldots K\}$ for what is termed the *Krylov subspace*. The representation of A in the K-dimensional Krylov subspace is generated as

$$H_{jk} \equiv \langle v_j, Av_k \rangle \tag{18}$$

The matrix H is then (directly) diagonalized, yielding

$$H\phi_i = \lambda_i \phi_i. \tag{19}$$

Direct diagonalization of H is inexpensive, since its size K is quite small. The eigenvalues λ_i are approximations to those of A. The corresponding approximations to the eigenvectors are superpositions of the vectors v_k, whose coefficients are contained in the vectors ϕ_i via $\sum_{k=1}^{K} \phi_i(k) v_k$. The quality of the approximate eigenpairs is measured by the residue

$$r \equiv (A - \lambda_i) \sum_{k=1}^{K} \phi_i(k)\, v_k \tag{20}$$

The Arnoldi procedure is carried out until $|r|$ is sufficiently small. This can be done in several ways, which may also be combined. The simplest is to take a moving sequence, i.e. $\{u_1, \ldots u_K\} \leftarrow \{u_2, \ldots u_{K+1}\}$. More generally, some number of powers of A may be taken between construction of Krylov spaces, i.e. $\{u_1, \ldots u_K\} \leftarrow \{A^p u_1, \ldots A^p u_K\}$. Yet another way is to use a power A^p of A as the basic iteration matrix, i.e. $u_k \equiv A^{(k-1)p} u_1$.

For bifurcation problems, only the *leading eigenvalues*, i.e. those with largest real part, are of interest; bifurcations occur as these traverse zero or the imaginary axis. However, repeated multiplication by A polarizes the vectors u or v in the direction of the *dominant eigenvalues*, i.e. those of largest magnitudes. For matrices such as (9) derived from spatial discretization of partial differential equations, the dominant eigenvalues resemble those of the Laplacian and are the spatially oscillating modes corresponding to the highest Fourier component or to the spatial grid. To find the leading eigenvalues, we use the property that for eigenpairs (λ, ψ) of A,

$$f(A)\psi = f(\lambda)\psi \tag{21}$$

and apply Arnoldi's method to a matrix $f(A)$ whose dominant eigenvalues $f(\lambda)$ are the leading eigenvalues of A.

Three candidates for functions for transforming the spectrum of A are

$$\text{the exponential:} \qquad f(A) = \exp(A\Delta t) \tag{22}$$

$$\text{a polynomial:} \qquad f(A) = \sum_i c_i A^i \tag{23}$$

$$\text{the inverse:} \qquad f(A) = A^{-1} \tag{24}$$

To understand the relative effectiveness of these transformations, we use the error analysis appropriate for the power method. We assume that a single real leading eigenvalue λ_1 of A is sought, and that this eigenvalue is close to zero, and use as a measure of the error the component corresponding to the second leading eigenvalue λ_2. Each action by $f(A)$ multiplies this component by the factor $f(\lambda_2)/f(\lambda_1)$.

The exponential (22) would in theory be ideal, since the leading eigenvalues of the Jacobian are sought precisely because they are the dominant eigenvalues of the evolution operator $\exp(A\Delta t)$ of the linearized equation (12). However, the exponential operator is not readily available; indeed constructing it is equivalent to direct diagonalization. What is available is instead the approximation (14) to $\exp(A\Delta t)$, valid for small Δt. The error at each step is multiplied by the factor $\exp((\lambda_2 - \lambda_1)\Delta t)$, which is near one for $\Delta t \ll 1$.

A polynomial operator (23) is used by the software package `Arpack` [16]. The coefficients c_i are chosen in such a way as to increase the components of eigenvectors whose eigenvalues are in a specified portion of the complex plane and to reduce unwanted components. These coefficients are refined as the calculation progresses and reveals more information about the spectrum of the matrix. The basic problem remains the same as that for A, however: multiplication by A tends to polarize vectors in the direction of the dominant eigenvalues of A; accessing leading eigenvalues, especially those close to zero, is far more difficult.

The inverse operator (24) is the most effective way to calculate eigenvalues near zero, since the multiplication by A^{-1} reduces the error by a factor λ_1/λ_2, which is very small for $|\lambda_1|$ small. The use of a shift s

$$f(A) = (A - sI)^{-1} \tag{25}$$

allows the inverse method to be used to direct the calculation towards any eigenvalue, including those with large imaginary parts or away from eigenvalues already calculated.

The exact inverse is, like the exponential, also not available. However, the equation

$$(L + DW(\Psi) - sI)u_{k+1} = u_k \tag{26}$$

may be solved by the conjugate gradient method, as described in the previous section, and this may be accomplished efficiently by preconditioning by the inverse Laplacian

$$L^{-1}(L + DW(\Psi) - sI)u_{k+1} = L^{-1}u_k \tag{27}$$

or by

$$(\Delta t)^{-1}(I-\Delta t L)^{-1}(L+DW(\Psi)-sI)u_{k+1} = (\Delta t)^{-1}(I-\Delta t L)^{-1}u_k \tag{28}$$

Equation (28) can be implemented by adapting a mixed implicit-explicit time-stepping code, again as described in the previous section.

The operators used to generate the Krylov subspace and the matrix representation on the Krylov subspace need not be the same. Since the action of $(A-sI)^{-1}$ via conjugate gradient iteration is far less accurate than direct multiplication by A, we use (27) in order to generate a Krylov subspace which is polarized in the direction of the desired leading eigenvectors and we use (18) to generate the small matrix representation H on the resulting subspace.

Since conjugate gradient iteration constructs the solution to (26) by successive multiplications by A, both (23) and (24) essentially construct approximations to the eigenvectors as superpositions of powers of A on an initial vector. However, when the inverse Laplacian is available, (24) is more efficient than (23) in assembling the information gathered about the matrix as the iterative methods progress. We note, however, that implementation of the inverse Arnoldi method via conjugate gradient iteration suffers an inherent contradiction. The convergence of the Arnoldi method depends on separation of the eigenvalues, while that of the conjugate gradient method depends on their clustering. The shift s may be adjusted empirically to facilitate convergence of the conjugate gradient iteration. Computation of a leading eigenpair generally requires 3–10 iterations of the inverse Arnoldi method.

Application to the Gross-Pitaevskii equation

We have applied the methods described above to the problem of computing the steady states and eigenmodes of equation (4), which we write in the abbreviated form:

$$-i\frac{\partial \Psi}{\partial t} = L\Psi + W(\Psi) \tag{29}$$

where

$$L\Psi \equiv \frac{1}{2}\nabla^2\Psi \tag{30}$$

$$W(\Psi) \equiv \left[\mu - V(\mathbf{x}) - a|\Psi|^2\right]\Psi \tag{31}$$

The low temperature needed for Bose-Einstein condensation is modeled by a confining harmonic potential with cylindrical symmetry:

$$V(\mathbf{x}) = \frac{1}{2}|\omega\cdot\mathbf{x}|^2 = \frac{1}{2}(\omega_r r^2 + \omega_z z^2) \tag{32}$$

The operators L and W defined in (30) and (31) are spatially discretized using the pseudospectral method. We use a three-dimensional periodic Cartesian domain, on which Ψ and V are expanded as three-dimensional trigonometric (Fourier) series. The resolution in each direction is 50 or 100, so the total number of gridpoints or trigonometric functions is as high as $M = 10^6$. Actions and inversions of the Laplacian L are carried out on the Fourier representations, while the actions of the multiplicative operator W are carried out on the grid representations; all of these operations scale approximately linearly in M. Fourier transforms are used to pass between the Fourier and grid representations in a time proportional to $M \log M$.

The particle number $\mathcal{N}$ and energy $\mathcal{E}$ are defined by:

$$\mathcal{N} = \int d^3x |\Psi|^2 \tag{33}$$

$$\mathcal{E} = \int d^3x \left[\frac{1}{2}|\nabla\Psi|^2 + V(\mathbf{x})|\Psi|^2 + \frac{a}{2}|\Psi|^4\right] \tag{34}$$

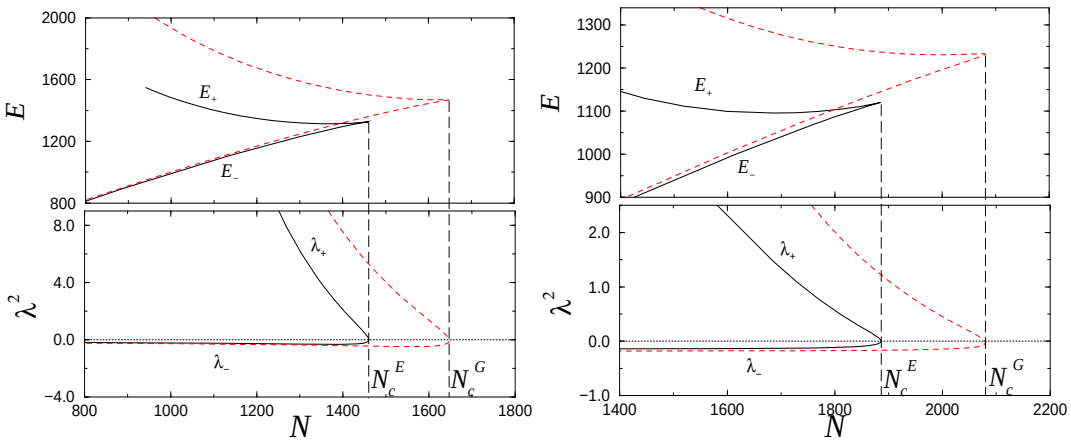

Figure 1: Stationary solutions of the GP equation versus the particle number $\mathcal{N}$ for two non-isotropic potentials with $(\omega_r, \omega_z) = (\hat{\omega}, \hat{\omega}/5)$ (cigar, left) and $(\omega_r, \omega_z) = (\hat{\omega}/5, \hat{\omega})$ (pancake, right). Top: value of the energy functional $\mathcal{E}_+$ on the stable (elliptic) branch and $\mathcal{E}_-$ on the unstable (hyperbolic) branch. Bottom: square of the bifurcating eigenvalue ($\lambda_\pm^2$); $|\lambda_-|$ is the energy of small excitations around the stable branch. Solid lines: exact solution of the GP equation. Dashed lines: Gaussian approximation.

The control parameter is the particle number, which is conserved by the Nonlinear Schrödinger equation. The solutions can also be indexed by μ.

The results for two geometries, a *cigar* ($\omega_z = \omega_r/5$), and a *pancake* ($\omega_r = \omega_z/5$) [17, 18, 19]. are presented in figure 1. Also shown are approximate results from the analytic Gaussian variational technique [11, 12, 13, 19].

In each case, the steady states are calculated using Newton's method according to the methods described above. Critical values exist for $\mathcal{N}$ and for μ, at which the system undergoes a Hamiltonian saddle-node bifurcation. For $\mathcal{N} < \mathcal{N}_c$, a fixed value of $\mathcal{N}$ corresponds to two values of μ, one greater and one less than μ_c. That is, an elliptic stationary state and a mixed elliptic-hyperbolic stationary state coexist. The phase space is separated into two regions by a separatrix which is a homoclinic orbit linking the hyperbolic stationary state to itself. Trajectories inside the orbit remain bounded near the elliptic fixed point, while those outside fall into unbounded hyperbolic trajectories corresponding to collapse of the condensate. As $\mathcal{N}$ is increased, the hyperbolic and elliptic stationary states approach one another, and the homoclinic orbit inside which orbits are bounded is reduced. The two stationary states join at $\mathcal{N} = \mathcal{N}_c$ and no stationary state exists for $\mathcal{N} > \mathcal{N}_c$.

In order to arrive at a correct formulation of the linear stability problem, it is necessary to decompose $\psi = \psi^R + i\psi^I$. We have

$$\begin{aligned} DW(\Psi)\psi &= [\mu - V(\mathbf{x})]\,(\psi^R + i\psi^I) - a\Psi^2(3\psi^R + i\psi^I) \\ &= DW^R\psi^R + iDW^I\psi^I \end{aligned} \tag{35}$$

where

$$DW^R \equiv \mu - V(\mathbf{x}) - 3a\Psi^2 \tag{36a}$$

$$DW^I \equiv \mu - V(\mathbf{x}) - a\Psi^2 \tag{36b}$$

The equation governing the eigenmodes of (29) is:

$$\lambda \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix} = \begin{bmatrix} 0 & -(L + DW^I) \\ L + DW^R & 0 \end{bmatrix} \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix} \tag{37}$$

It is more convenient to work with the square of the matrix in (37):

$$\lambda^2 \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix} = \begin{bmatrix} -(L+DW^I)(L+DW^R) & 0 \\ 0 & -(L+DW^R)(L+DW^I) \end{bmatrix} \begin{pmatrix} \psi^R \\ \psi^I \end{pmatrix} \quad (38)$$

Because (38) is block diagonal, it can be separated into the two problems:

$$\lambda^2 \psi^R = -(L+DW^I)(L+DW^R)\psi^R \quad (39a)$$
$$\lambda^2 \psi^I = -(L+DW^R)(L+DW^I)\psi^I \quad (39b)$$

Problems (39a) and (39b) are closely related, since if ψ^R is an eigenvector of (39a) with eigenvalue λ, then $(L+DW^R)\psi^R$ is an eigenvector of (39b) with the same eigenvalue. Thus, we solve only (39a).

Since the operators L, DW^I, and DW^R are all self-adjoint under the standard Euclidean inner product, so are the operators in (39a)-(39b). The eigenvalues λ^2 of (39a)-(39b) are therefore all real; in fact almost all are negative, perturbed only slightly from the eigenvalues of $-L^2$. The unstable hyperbolic branch has one positive eigenvalue λ^2. We use the inverse Arnoldi method with Laplacian preconditioning, adjusting the shift s to follow this leading eigenvalue as it changes sign through the stable elliptic and unstable hyperbolic branches. The leading eigenvalues λ^2 are presented in figure 1, along with those calculated by the Gaussian approximation.

Acknowledgments

The computations were carried out on the NEC-SX5 computer of the Institut du Développement et des Ressources en Informatique Scientifique (IDRIS) of the Centre National pour la Recherche Scientifique (CNRS). The Gaussian approximations were calculated by S. Métens.

References

[1] C.K. Mamun & L.S. Tuckerman, *Asymmetry and Hopf bifurcation in spherical Couette flow*, Phys. Fluids **7**, 80 (1995).

[2] L.S. Tuckerman & D. Barkley, *Bifurcation analysis for time-steppers*, in Numerical Methods for Bifurcation Problems and Large-Scale Dynamical Systems, ed. by E. Doedel & L.S. Tuckerman (Springer, New York, 2000), p. 452–466.

[3] L.S. Tuckerman, F. Bertagnolio, O. Daube, P. Le Quéré & D. Barkley, *Stokes preconditioning for the inverse Arnoldi method*, in *Continuation Methods for Fluid Dynamics* (Notes on Numerical Fluid Mechanics, Vol. 74), pp. 241–255, ed. by D. Henry & A. Bergeon (Vieweg, 2000).

[4] E.P. Gross, Nuovo Cimento **20** 454 (1961).

[5] L.P. Pitaevskii, *Vortex lines in an imperfect Bose gas*, Zh. Eksp. Teor. Fiz. **40**, 646 (1961) [Sov. Phys. JETP **13**, 451 (1961)].

[6] S. Bose, *Plancks Gesetz und Lichtquantenhypothese*, Z. Phys. **26**, 178 (1924).

[7] A. Einstein, *Quantentheorie des einatomigen idealen gases: Zweite Abhandlung*, Sitzungber. Preuss. Akad. Wiss. **1925**, 3 (1925).

[8] M.H. Anderson, J.R. Ensher, M.R. Matthews, C.E. Wieman & E.A. Cornell, *Observation of Bose-Einstein condensation in a dilute atomic vapor*, Science **269**, 198 (1995).

[9] K.B. Davis, M.-O. Mewes, M.R. Andrews, N.J. van Druten, D.S. Durfee, D.M. Kurn & W. Ketterle, *Bose-Einstein condensation in a gas of sodium atoms*, Phys. Rev. Lett. **75**, 3969 (1995).

[10] C.C. Bradley, C.A. Sackett, J.J. Tollett & R.G. Hulet, *Evidence of Bose-Einstein condensation in an atomic gas with attractive interactions*, Phys. Rev. Lett. **75**, 1687 (1995).

[11] P.A. Ruprecht, M.J. Holland, K. Burnett, & M. Edwards, *Time-dependent solution of the nonlinear Schrödinger equation for Bose-condensed trapped neutral atoms*, Phys. Rev. A **51**, 4704 (1995).

[12] M. Ueda & A.J. Leggett, *Macroscopic quantum tenneling of a Bose-Einstein condensate with attractive interaction*, Phys. Rev. Lett. **80**, 1576 (1998).

[13] C. Huepe, S. Métens, G. Dewel, P. Borckmans & M.E. Brachet, *Decay rates in attractive Bose-Einstein condensates*, Phys. Rev. Lett. **82**, 1616 (1999).

[14] L.N. Trefethen, N.M. Nachtigal & S.C. Reddy, *How fast are nonsymmetric matrix iterations?*, SIAM J. Matrix Anal. Appl. **13**, 778 (1992).

[15] H.A. van der Vorst, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. **13**, 631 (1992).

[16] R.B. Lehoucq, D.C. Sorensen & C. Yang C., *ARPACK User's Guide, Philadelphia*, (SIAM, Philadelphia, 1998).

[17] J.L. Roberts, N.R. Claussen, S.L. Cornish, E.A. Donley, E.A. Cornell & C.E. Wieman, *Controlled collapse of a Bose-Einstein condensate*, Phys. Rev. Lett. **86**, 4211 (2001).

[18] A. Gammal, T. Frederico & L. Tomio, *Critical number of atoms for attractive Bose-Einstein condensates with cylindrically symmetrical traps*, Phys. Rev. A **64** 055602 (2001).

[19] C. Huepe, L.S. Tuckerman, S. Metens & M.E. Brachet, *Decay rates in non-isotropic attractive Bose-Einstein condensates*, to be submitted to Phys. Rev. A (2002).