

# A Continuation and Bifurcation Technique for Navier–Stokes Flows

J. Sanchez,\* F. Marques,\* and J. M. Lopez†

\**Departament de Física Aplicada, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain; and*

†*Department of Mathematics and Statistics, Arizona State University, Tempe, Arizona 85287-1804*

E-mail: sanchez@fa.upc.es, marques@fa.upc.es, lopez@math.asu.edu

Received October 10, 2001; revised April 3, 2002

---

An efficient numerical bifurcation and continuation method for the Navier–Stokes equations in cylindrical geometries is presented and applied to a nontrivial fluid dynamics problem, the flow in a cylindrical container driven by differential rotation. The large systems that result from discretizing the Navier–Stokes equations, especially in regimes where inertia is important, necessitate the use of iterative solvers which in turn need preconditioners. We use incomplete lower–upper decomposition (ILU) as an effective preconditioner for such systems and show the significant gain in efficiency when an incomplete LU of the full Jacobian is used instead of using only the Stokes operator. The computational cost, in terms of CPU time, grows with the size of the system (i.e., spatial resolution) according to a power law with exponent around 1.7, which is very modest compared to direct methods, indicating the appropriateness of the schemes for large nonlinear partial differential equation problems. © 2002 Elsevier Science (USA)

*Key Words:* Krylov; Arnoldi; subspace iteration; incomplete LU; preconditioner; continuation.

---

## 1. INTRODUCTION

The development of modern linear algebra techniques and the availability of fast processors and large amounts of memory allow one to contemplate numerical bifurcation and continuation studies of large extended systems such as in fluid dynamics. The main difficulty with this program of study is that it is not practical to solve the resulting large linear algebra problems by direct methods. Iterative methods tailored according to the problem at hand are needed. They are based on subspace iteration, Krylov methods, and efficient preconditioners. In this paper, we put a combination of these components together and apply them to a nontrivial fluid dynamics problem—the flow in a cylindrical container driven by the rotation of the endwalls. The base state is nontrivial, and in different regions of parameter

space it undergoes saddle-node and Hopf bifurcations. We also locate a codimension-2 cusp bifurcation.

For low-dimensional systems—e.g., models of population dynamics, chemical reactions, neuron models, coupled oscillators, and lasers (see [11] for specific references)—there are readily available continuation and bifurcation packages such as AUTO, CONTENT, and DSTOOL. Many of these low-dimensional systems are governed by a large number of bifurcation parameters, and these packages have been used to deal with high-codimension bifurcations (mainly local), as well as continuation of invariant manifolds (e.g., periodic, homoclinic, and heteroclinic orbits). Due to the small size of the associate linear systems these are designed for, they implement direct solvers. These types of methods are based fundamentally on two key ideas, arclength continuation and the use of extended systems (e.g., see [12]).

For fluid dynamics problems, the dynamical system is large in dimension (due to the discretization in space) and typically has a small number of bifurcation parameters (e.g., Reynolds numbers describing governing forces and geometric parameters). Continuation of steady states or rotating wave solutions in, for example, the Taylor–Couette problem, have been performed using finite differences and/or Fourier expansions in space with AUTO (e.g., [5, 19, 23]), and with finite elements and extended systems techniques (e.g., [2, 26]). The spatial resolution available to these approaches is limited by the use of direct solvers.

Recent developments in iterative methods (e.g., [20, 22]) allow for a marked increase in the spatial resolution that can be utilized. However, for these iterative methods to be robust and efficient, care must be taken with preconditioning. The specifics of the preconditioner depend critically on how space is discretized. For spectral discretization, Edwards *et al.* [7] advocate the use of the Stokes operator in spectral space as the preconditioner. Example fluid dynamics problems close to the bifurcation point have been successfully solved (e.g., [17]). Canuto *et al.* [1] also suggest the use of finite-difference or finite-element-based preconditioners within a spectral method in general. Demaret and Deville [3] have taken this approach for the Stokes equation ( $Re \rightarrow 0$ ). However, this approach is cumbersome, as the governing equations must be discretized in two distinct ways but within the same formulation (e.g., velocity–pressure and streamfunction–vorticity).

In finite-difference or finite-element methods, the choice of preconditioners has been studied more extensively, and the use of incomplete lower–upper decomposition (ILU) is commonly used and is available in several packages, e.g., SPARSKIT [21]. This is the approach used in the present study.

Apart from continuation of solutions, one often also wants to determine their stability under variation of parameters. For large systems, there are essentially two practical options: subspace iterations [25] and Arnoldi methods [13], both of which are based on the power method for the eigenvalue problem. To detect bifurcation phenomena, Cayley and shift-invert transformations are almost always needed [18]. Here, we have used the Arnoldi methods.

The layout of the paper is as follows. Section 2 briefly describes the test fluid dynamics problem and the formulation of the Navier–Stokes equation and boundary conditions. Section 3 describes the discretization and sets up the various aspects of the problem (continuation of the basic state, linear stability, and nonlinear evolution). Section 4 describes the application of the numerical protocol to a series of related test cases and details the convergence and efficiency of the techniques.

## 2. FORMULATION

Dynamical systems governed by partial differential equations (PDE) give, once discretized in space, a system of differential equations (ODE) of very large dimension. The numerical methods used to deal with these problems need to take into consideration the specific details of the problem in order to be robust and efficient.

There are many fundamental issues in fluid dynamics that can be effectively addressed by the study of flows in cylindrical containers (e.g., symmetry breaking and transition to complex dynamics). The attraction of such a flow geometry is that, although the system is fully three-dimensional, the invariance of the equations and boundary conditions to arbitrary rotations in azimuth (the  $SO(2)$  symmetry) naturally provides a periodic direction (the azimuthal direction,  $\theta$ ) which can be efficiently exploited in the solution scheme. Further, there are numerous fundamental flows (e.g., Taylor–Couette and vortex breakdown in an enclosed cylinder) where not only is the basic state axisymmetric but in extensive regions of parameter space the bifurcated solutions are also axisymmetric [10, 27].

We consider the flow of an incompressible fluid in a completely filled cylinder of radius  $R$  and height  $H$  with bottom and sidewall rotating at a constant angular speed  $\Omega_b$  and driven by the differential rotation of its top lid at angular speed  $\Omega_t$ , as shown schematically in Fig. 1. Using  $R$  as the length scale and the viscous time scale  $R^2/\nu$ , we define three nondimensional parameters that govern the dynamics:

$$\begin{aligned} \text{the base Reynolds number} & \quad Re_b = \Omega_b R^2/\nu, \\ \text{the top Reynolds number} & \quad Re_t = \Omega_t R^2/\nu, \\ \text{the aspect ratio} & \quad \Lambda = H/R. \end{aligned}$$

We use cylindrical polar coordinates  $(r, \theta, z)$ , and since the problems we consider are axisymmetric, the velocity components  $(u, v, w)$  will only be functions of  $(r, z)$  and time.

For planar two-dimensional and axisymmetric problems, a convenient formulation is in terms of the streamfunction and vorticity since it is implicitly divergence-free. Here we use a biharmonic formulation in terms of the streamfunction only, in which the boundary conditions for the streamfunction are derived directly from the no-slip condition. This is the method used in Lopez *et al.* [16]; the number of variables is smaller than in a conventional streamfunction–vorticity formulation, and the boundary conditions can be very efficiently implemented for semiimplicit time evolution. In contrast, the boundary conditions on the vorticity are cumbersome in a semiimplicit formulation, and one usually resorts to explicit time integration (e.g., see [6]). However, with the biharmonic formulation, the order of

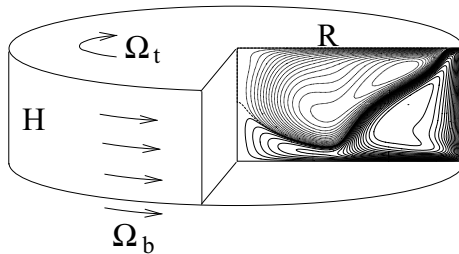


FIG. 1. Schematic of the flow geometry.

the equations increases. The numerical methods detailed below can also be implemented in other formulations of the governing equations, e.g., streamfunction–vorticity, primitive variables, and velocity potentials.

For rotating flows, the velocity has three nonzero components. The meridional components  $u$  and  $w$  are described by the streamfunction  $\psi$ , and the azimuthal component of velocity,  $v$ , is incorporated in terms of the (axial) angular momentum,  $\gamma = rv$ . In cylindrical polar coordinates, the velocity and vorticity vectors are

$$\mathbf{u} = (-\psi_z, \gamma, \psi_r)/r \quad \text{and} \quad \nabla \times \mathbf{u} = (-\gamma_z, r\eta, \gamma_r)/r,$$

where  $\psi$  is the streamfunction, and  $\eta$ , the azimuthal component of vorticity, is related to  $\psi$  via

$$\Delta_* \psi = \psi_{zz} + \psi_{rr} - \psi_r/r = -r\eta.$$

Note that contours of  $\psi$  in an  $(r, z)$  plane depict the streamlines of the flow, and likewise contours of  $\gamma$  depict the vortex lines. The governing equations are the azimuthal components of the Navier–Stokes equations and their curl:

$$\begin{aligned} \partial_t \gamma &= \Delta_* \gamma + \frac{1}{r} \psi_z \gamma_r - \frac{1}{r} \psi_r \gamma_z, \\ \partial_t \Delta_* \psi &= \Delta_*^2 \psi + \frac{1}{r} \psi_z \partial_r (\Delta_* \psi) - \frac{1}{r} \psi_r \Delta_* \psi_z - \frac{2}{r^2} \psi_z \Delta_* \psi - \partial_z (\gamma^2/r^2). \end{aligned} \quad (1)$$

The governing equations can be written in a more compact form using  $x = (\gamma, \psi)$ ,

$$\partial_t Bx = Lx + N(x, x) = f(x), \quad B = \begin{pmatrix} 1 & 0 \\ 0 & \Delta_* \end{pmatrix}, \quad L = \begin{pmatrix} \Delta_* & 0 \\ 0 & \Delta_*^2 \end{pmatrix}, \quad (2)$$

and  $N$  is the quadratic nonlinearity. The corresponding boundary conditions are:

$$\begin{array}{lll} \text{axis } (r = 0) & \psi = \psi_r = \gamma = 0 \\ \text{top } (z = \Lambda) & \psi = \psi_z = 0, & \gamma = r^2 Re_t \\ \text{bottom } (z = 0) & \psi = \psi_z = 0, & \gamma = r^2 Re_b \\ \text{sidewall } (r = 1) & \psi = \psi_r = 0, & \gamma = Re_b. \end{array} \quad (3)$$

To solve for the basic state,  $x_b = (\gamma_b, \psi_b)$ , the left-hand side of (2) is set to zero. The linear stability of the basic state is determined by linearizing (2) about the basic state, with

$$x(r, z, t) = x_b(r, z) + e^{\lambda t} x_p(r, z),$$

where the perturbation  $x_p$  satisfies homogeneous boundary conditions. The linearized equations for  $x_p$  lead to the generalized eigenvalue problem

$$\lambda Bx_p = Lx_p + N(x_b, x_p) + N(x_p, x_b) = Df(x_b)x_p, \quad (4)$$

where  $Df(x_b)$  is the Jacobian of the right-hand side of (2) evaluated at  $x_b$ .

**TABLE I**  
**Tabulation of the Maximum Order of the PDE, the Number of Variables,<sup>a</sup>**  
**and the Bandwidth of the Jacobian Matrix with Optimal Ordering<sup>b</sup>**

	Max. PDE order	Num. variables	Bandwidth
Biharmonic ( $\psi, \gamma$ )	4	$2n_r n_z$	$4 \min(n_r, n_z)$
Streamfunction–vorticity ( $\psi, \gamma, \eta$ )	2	$3n_r n_z$	$3 \min(n_r, n_z)$
Primitive variables ( $u, v, w, p$ )	2	$4n_r n_z$	$4 \min(n_r, n_z)$

<sup>a</sup> Using second-order finite differences with  $n_r$  points in radial and  $n_z$  points in axial.

<sup>b</sup> Distance between the main diagonal and the furthest diagonal.

### 3. NUMERICAL TECHNIQUES

Discretization in space is accomplished using second-order centered finite differences, both for the equations and the boundary conditions. The ordering of the variables has been chosen to minimize the bandwidth of the linear systems. It depends on  $n_r$  and  $n_z$ , the number of subintervals in each coordinate. As we have usually used equal spacing in both, the ordering depends on the aspect ratio  $\Lambda$ . If  $\Lambda \geq 1$ , the variables are ordered by rows; if  $\Lambda < 1$  they are ordered by columns; and in both cases, the values of  $\gamma$  and  $\psi$  are alternated. The bandwidth of the linear systems is then  $4 \min\{n_r, n_z\}$ . Using other formulations of the Navier–Stokes equations, the product of the number of variables times the bandwidth increases, as shown in Table I.

The boundary conditions could also have been imposed using one-sided differences, but this would increase the bandwidth without increasing the global accuracy of the solution. Sensitive measures such as the location of the codimension-2 cusp point (see Section 4) are much more sensitive to the spatial resolution ( $n_r n_z$ ) than to how the boundary condition is implemented.

#### 3.1. Continuation Method

The time-independent governing equation for the basic state,  $x_b$ , is

$$f(x, p) = Lx + N(x, x) = 0, \quad (5)$$

where  $L$  and  $N$  are the discretized versions of the continuous operators in (2) and depend on the parameters of the problem through the boundary conditions (3). The variable  $p$  in (5) is the continuation parameter to be used, and may be any of the governing parameters in the problem. Parameter and pseudo-arclength continuation methods [12] have been used to solve (5) when the parameter  $p$  is varied. Second-degree polynomial extrapolation is used as a predictor, and the Newton method is used as a corrector.

In the case of parameter continuation, the linear system to be solved in each Newton iteration, for a fixed value of the parameter, is

$$D_x f(x^i, p)(x^{i+1} - x^i) = -f(x^i, p), \quad (6)$$

where  $D_x f(x^i, p)$  acting on a vector  $y$  is

$$D_x f(x^i, p)y = Ly + N(x^i, y) + N(y, x^i).$$

In the case of pseudo-arclength continuation, which is best suited in the presence of folds, the bordered system that determines a unique solution is

$$f(x, p) = Lx + N(x, x) = 0, \quad (7)$$

$$n(x, p) \equiv v^T \cdot (x - x_0) + w(p - p_0) = 0, \quad (8)$$

with  $(x_0, p_0)$  the predicted next point along the curve of solutions and  $(v, w)$  an approximation to the tangent at  $(x_0, p_0)$ . The linear system to be solved is now

$$\begin{pmatrix} D_x f(x^i, p^i) & D_p f(x^i, p^i) \\ v^T & w \end{pmatrix} \begin{pmatrix} x^{i+1} - x^i \\ p^{i+1} - p^i \end{pmatrix} = \begin{pmatrix} -f(x^i, p^i) \\ -n(x^i, p^i) \end{pmatrix}. \quad (9)$$

In both cases, the linear systems are solved iteratively. Inexact Newton methods are known to converge, if at each iteration the residual of the linear system is kept sufficiently small [4]. They can retain quadratic convergence with a suitable selection of the tolerance of the residual for the solution of the linear system at each Newton iteration.

### 3.2. Linear Stability

The linear stability of the basic state,  $x_b$ , is determined by the eigenvalue from (4) with largest real part (growth rate). If its real part is positive (negative), the basic state is unstable (stable). To solve the eigenvalue problem, we have used subspace iteration [25] and the Arnoldi method as implemented in the ARPACK package [13]. Both iterative techniques are based on the power method and converge to the eigenvalues with largest moduli. Although convergence properties are best known for subspace iteration, we have found that, for the fluid dynamics problems presented here, Arnoldi iteration converges much more rapidly.

Since we are interested in the eigenvalues with largest real part, rather than largest moduli, we implement shift-invert or Cayley transformation converting (4) to

$$(D_x f(x_b) - \beta B)^{-1} x_p = \tilde{\lambda} x_p \quad (10)$$

or

$$(D_x f(x_b) - \beta B)^{-1} (D_x f(x_b) - \alpha B) x_p = \tilde{\lambda} x_p, \quad (11)$$

respectively (see [18] for details).

The eigenvalues of (10) and (11) are related to those of (4) by

$$\tilde{\lambda} = \frac{1}{\lambda - \beta} \quad \text{and} \quad \tilde{\lambda} = \frac{\lambda - \alpha}{\lambda - \beta}, \quad (12)$$

where  $\alpha$  and  $\beta$  are real in our implementation. We have not considered complex shifts, as they either introduce complex arithmetic or double the size of the linear system to be solved, even though they are a natural choice to detect Hopf bifurcations.

The Arnoldi method acting on (10) converges to eigenvalues of largest moduli that correspond to eigenvalues of (4) nearest to  $\beta$ . Acting on (11) with suitably chosen  $\alpha$  and  $\beta$ , Arnoldi converges to  $\tilde{\lambda}$ , which is related to the eigenvalues of (4) with largest real part, through (12). The transformations leave the eigenvectors unchanged.

### 3.3. Nonlinear Evolution Solver

In Lopez *et al.* [16], the time derivatives in (2) were approximated by a second-order backwards difference; the linear term was treated implicitly; and a second-order backwards extrapolation scheme was used for the nonlinear term, leading to

$$\left(\frac{3}{2\delta t}B - L\right)x^k = \frac{2}{\delta t}Bx^{k-1} - \frac{1}{2\delta t}Bx^{k-2} + 2N(x^{k-1}, x^{k-1}) - N(x^{k-2}, x^{k-2}),$$

where  $\delta t$  is the time step and  $x^k = x(t_0 + k\delta t)$ . This scheme was used to compute nonlinear time periodic states for  $\Lambda = 2.5$ ,  $Re_b = 0$ , and  $Re_t$  up to 4000 using a direct solver for the banded linear equations. If one replaces the direct solver with an iterative solver based on the generalized minimum residual method (GMRES) and ILU as outlined in this paper, the nonlinear evolution solver can be formulated as fully implicit. This could be of considerable advantage for very stiff nonlinear problems.

### 3.4. Preconditioning and Solving the Linear Systems

The most critical aspect in the different types of solvers listed above involves the solution of large linear systems of the form

$$(A + cB)y = a, \quad (13)$$

where  $a$  is known. The solvers differ in the specification of  $A$  and  $c$ :

continuation (parameter)	$A = D_x f(x^i, p)$	$c = 0$
continuation (pseudo-arclength)	$A = \begin{pmatrix} D_x f(x^i, p^i) & D_p f(x^i, p^i) \\ v^T & w \end{pmatrix}$	$c = 0$
linear stability	$A = D_x f(x_b, p)$	$c = -\beta$
nonlinear evolution	$A = L$	$c = -3/2\delta t$ .

For the continuation methods and linear stability analysis, the linear system (13) is solved via the preconditioned generalized minimum residual method (GMRES). GMRES is an iterative projection method. This class of methods produces sequences of approximations,  $x_m$ , to the solution of a linear system  $Ax = b$ , with an initial guess  $x_0$ , which satisfy two conditions that determine each particular method,

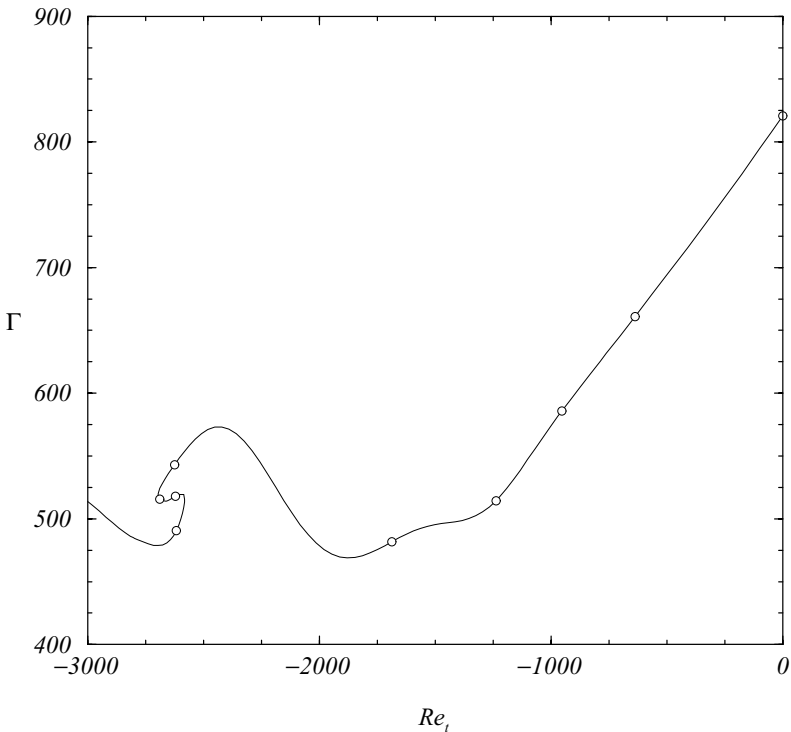
$$x_m \in x_0 + \mathcal{K}_m \quad \text{and} \quad b - Ax_m \perp \mathcal{L}_m,$$

where  $\mathcal{K}_m$  and  $\mathcal{L}_m$  are two  $m$ -dimensional linear subspaces. With these two conditions,  $x_m$  minimizes the Euclidean norm of the residual,  $b - Ax_m$ , over all the vectors in  $x_0 + \mathcal{K}_m$ . In the particular case of GMRES,  $\mathcal{L}_m = A\mathcal{K}_m$ , and  $\mathcal{K}_m$  is the Krylov subspace  $\mathcal{K}_m = \{r_0, Ar_0, A^2r_0, \dots, A^{m-1}r_0\}$ , where  $r_0 = b - Ax_0$  (see [20] for implementation details). The maximal dimension of the subspaces  $\mathcal{K}_m$  and  $\mathcal{L}_m$  used in our calculations is named *irestart*. If GMRES does not converge in *irestart* iterations, the method is restarted using the latter approximation found as initial guess  $x_0$ . For the nonlinear evolution, (13) can be solved directly because  $A$  only contains the linear part of the Navier–Stokes equations, so the linear system decouples into two independent subsystems, and further, only one initial LU decomposition is needed.

To precondition the linear systems that arise in Newton's method and the linear stability analysis, an incomplete LU decomposition (ILU) from SPARSKIT is used [21]. During the continuation process, the ILU is updated only when GMRES fails to converge within a predetermined number of iterations. This allows for many points along the continuation curves to be computed using the same ILU decomposition. In some cases, using the linearization about the zero solution ( $Re_b = Re_t = 0$ ), which corresponds to preconditioning with  $L$ , the Stokes operator, is enough for a continuation run over a long range of the control parameter. This is the case for the continuation with  $\Lambda = 2.5$ ,  $Re_b = 0$ , and  $Re_t \in [0, 2700]$  that will be used later as a test case for the computation of the spectra. In other cases, far from  $Re_b = Re_t = 0$ , the full Jacobian must be used.

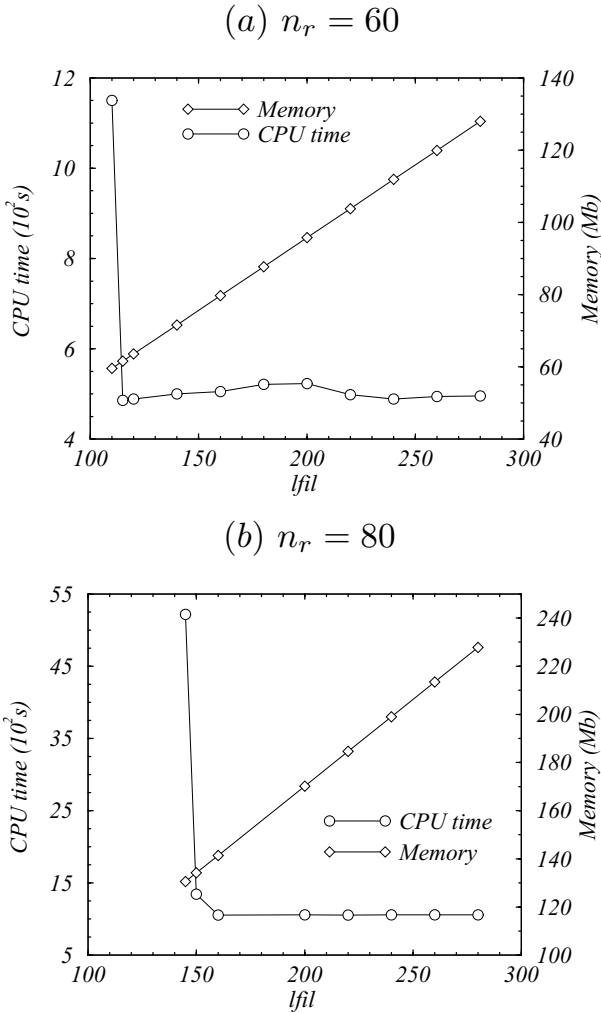
We characterize the state of the solution at a point in parameter space  $(Re_t, Re_b, \Lambda)$  by the value of  $\gamma$  at a convenient point,  $\Gamma = \gamma(r = 1/2, z = 3\Lambda/4)$ . In Fig. 2, for  $Re_b = 3000$  and  $\Lambda = 0.5$ ,  $\Gamma$  is shown over the range  $Re_t \in [-3000, 0]$ , and on the  $\Gamma$ -curve are located (open circles) the points at which incomplete LUs were computed during the continuation run in  $Re_t$ . Near the fold,  $Re_t \sim -2600$ , several ILUs were needed, due to the large variation of the velocity field along the fold. The part of the Jacobian coming from the nonlinear terms,  $N$ , undergoes significant changes (details of the fluid dynamics are given in Section 4).

The incomplete LU algorithms (ILU) perform a Gaussian elimination process on an initial sparse matrix  $A$  to find an approximation to its LU decomposition,  $A = LU$ . During the process, only certain selected entries in  $L$  and  $U$  are retained. To keep the memory requirements to a minimum, we have used the version that only retains the *lfil* elements in each column of  $L$  and each row of  $U$  of greater modulus. A tolerance can also be given



**FIG. 2.** Continuation run for  $Re_b = 3000$ ,  $\Lambda = 0.5$ . Open circles indicate the  $Re_t$  values where the ILU preconditioner was updated.

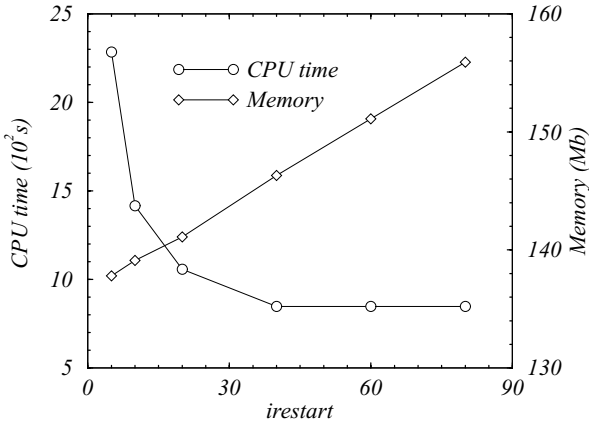




**FIG. 3.** CPU time and required memory, as a function of  $lfil$ , for continuation runs using  $n_r$ , as indicated and  $\Lambda = 2.5$ ,  $n_z = \Lambda n_r$ ,  $Re_b = 0$ , and  $Re_t \in [0, 2700]$ .

for dropping small terms in the factorization. We have not used this feature or pivoting, because we have not found them to improve the convergence for our problem. The size of  $lfil$  has proven to be critical. It cannot be small if GMRES is to converge in a reasonable number of iterations. Many tests have been made to obtain a criterion to select  $lfil$ . Two representative cases are shown in Fig. 3 for the above-mentioned continuation runs with  $n_r = 60$  and  $80$ ,  $\Lambda = 2.5$ , and  $n_z = \Lambda n_r$ . The CPU time decreases rapidly with  $lfil$  when it is small, and then it is essentially constant for  $lfil$  above approximately  $2n_r$ . The number of diagonals above and below the main diagonal of the original system is  $4n_r$  when  $\Lambda \geq 1$ . Below  $2n_r$ , the convergence is either very slow or there is no convergence at all. As the memory requirement increases linearly with  $lfil$ , we have used  $lfil \gtrsim 2 \min\{n_r, n_z\}$ , giving convergence with minimal memory requirements.

The other parameter that influences the CPU time and memory required is the dimension,  $irestart$ , of the Krylov subspace used for GMRES. Figure 4 shows how CPU time decreases with  $irestart$  for the values considered in the plot. Higher values would increase the CPU



**FIG. 4.** CPU time and required memory, as a function of  $irestart$ , for a continuation run using  $n_r = 80$ ,  $\Lambda = 2.5$ ,  $n_z = \Lambda n_r$ ,  $Re_b = 0$ , and  $Re_t \in [0, 2700]$ .

time due to the orthogonalization process used to obtain the Arnoldi basis during GMRES iterations. Again, as the memory increases with  $irestart$  we have used values as small as possible (between 20 and 50 in most cases).

When  $lfil$  and  $irestart$  are selected as described, GMRES converges in a very small number of iterations. During the continuation process it ranges from 1 to 30 except very near the points at which the ILU is updated. For the linear systems in the stability analysis, GMRES converges even faster (in about 10 iterations) because the Jacobian used in the ILU is calculated precisely at the solution for which the spectrum is sought.

#### 4. NUMERICAL RESULTS

We chose a single flow that provides stringent tests of all aspects of our computational package: spatial resolution of boundary and internal layers and the determination of spectra and eigenmodes. The swirling shear flow introduced in Section 2 (see schematic in Fig. 1) has been extensively studied for  $Re_b = 0$  (e.g., [8–10, 14, 28]). The basic state is non-trivial with structure in  $(r, z)$  consisting of an Ekman layer, a swirling wall jet, and vortex breakdown recirculations on the axis. It loses stability via a supercritical Hopf bifurcation that, for  $\Lambda$  in the neighborhood of 2.5, preserves  $SO(2)$  symmetry. This problem provides a good test case for convergence tests. The results of these tests are detailed in Section 4.1. When  $Re_b \neq 0$  and  $Re_b Re_t < 0$ , the Ekman boundary layer on the counter rotating top separates and forms a free shear layer. This free shear layer sweeps out a large range of the computational domain as parameters  $(Re_t, Re_b, \Lambda)$  are varied continuously, and as such presents a challenge for efficient continuation of solution branches. Furthermore, this problem also has a cusp bifurcation [15] leading to hysteresis as parameters are varied and to the coexistence of multiple solutions, including unstable equilibria, which also provide interesting tests for the computational methods. The results of these tests are presented in Section 4.2.

##### 4.1. The Hopf Bifurcation at $Re_b = 0$ , $\Lambda = 2.5$

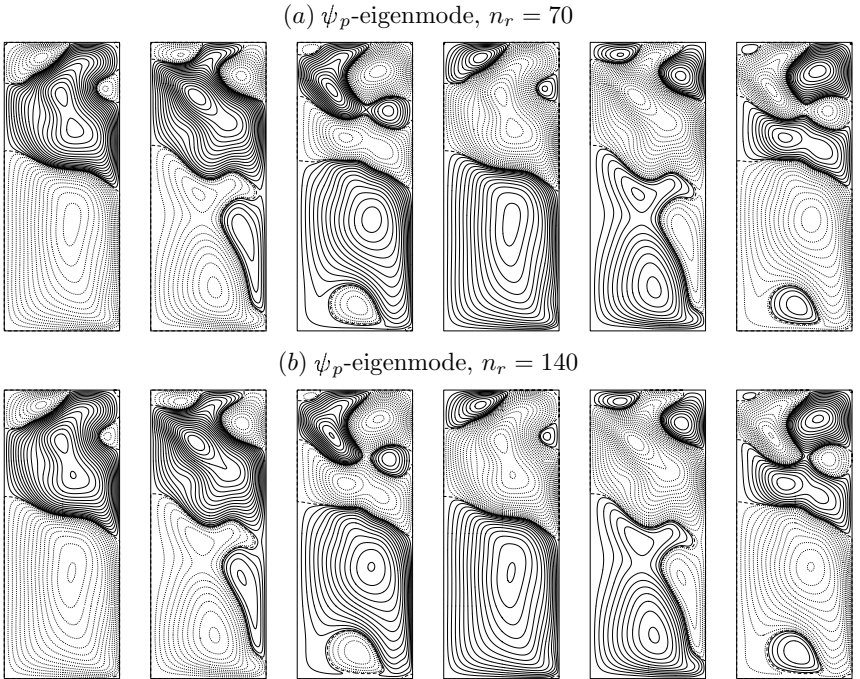
For  $Re_b = 0$  and  $\Lambda$  in the neighborhood of 2.5, Gelfgat *et al.* [10] have established that the steady axisymmetric basic state loses stability via a supercritical Hopf bifurcation

that preserves the  $SO(2)$  symmetry. In Lopez *et al.* [16], we have further studied the linear stability of the basic state at  $\Lambda = 2.5$ , showing that it undergoes successive Hopf bifurcations and that the associated eigenmodes account for the primary characteristics of the various nonlinear states that have been observed experimentally [24]. The flow physics are described in detail in Lopez *et al.* [16], and here we explore the computational issues associated with those results. Furthermore, here we examine the use of ILU, which was not utilized in Lopez *et al.* [16].

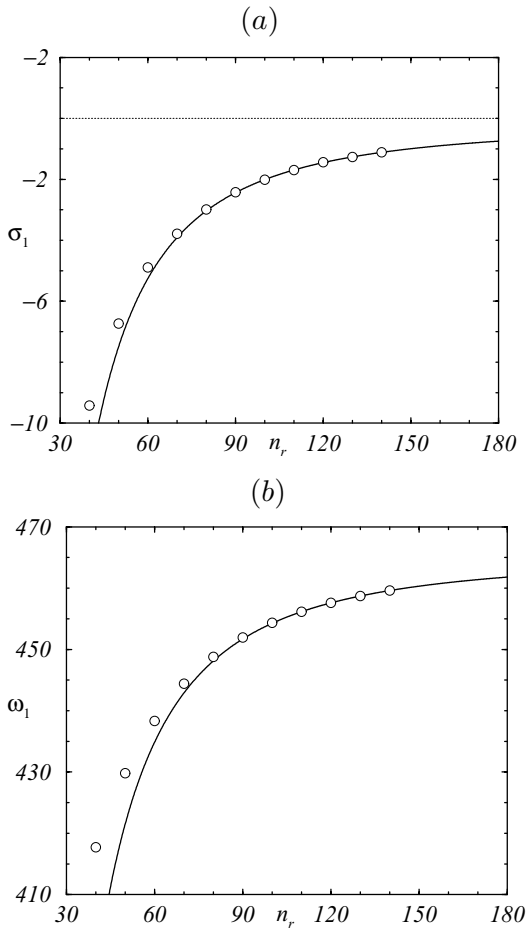
The convergence of the eigenvalue problem is by far the most sensitive computation to grid resolution. Throughout, we use a uniform grid with  $n_z/n_r = \Lambda$ , where  $n_r$  and  $n_z$  are the number of grid intervals in the radial and axial directions. We now compare the  $\psi_p$ -eigenmodes corresponding to the most dangerous eigenvalue, i.e., that with largest real part, for  $Re_b = 2700$  and  $\Lambda = 2.5$ , computed with  $n_r = 70$  and  $n_r = 140$  (the first Hopf bifurcation of the basic state for  $\Lambda = 2.5$  takes place at  $Re_b = 2706$ ). In fact, this eigenvalue is one of a pair of complex conjugates, and so the eigenvector has a harmonic temporal dependence. For the eigenvalue pair  $\sigma_1 \pm i\omega_1$  and the associated complex eigenvectors  $y_R \pm iy_I$ , the corresponding eigenmode has the form

$$\text{Real}(e^{(\sigma_1 + i\omega_1)t}(y_R + iy_I)) = e^{\sigma_1 t}(y_R \cos \omega_1 t - y_I \sin \omega_1 t).$$

Figure 5 shows six equally spaced phases of the streamlines corresponding to the eigenmode  $y_R \cos \omega_1 t - y_I \sin \omega_1 t$  over one period  $\tau_1 = 2\pi/\omega_1$ . For  $n_r = 70$ ,  $\omega_1 = 444.4$ , and for  $n_r = 140$ ,  $\omega_1 = 459.6$ . Note that the complex eigenvectors are determined up to an arbitrary phase. In the figure, we have adjusted the phase up to  $\pi/30$  in order to synchronize



**FIG. 5.** Streamlines of the eigenmode corresponding to the most dangerous eigenvalue,  $\lambda_1$ , at  $Re_b = 2700$ ,  $\Lambda = 2.5$ , over one period for  $n_r$  as indicated.



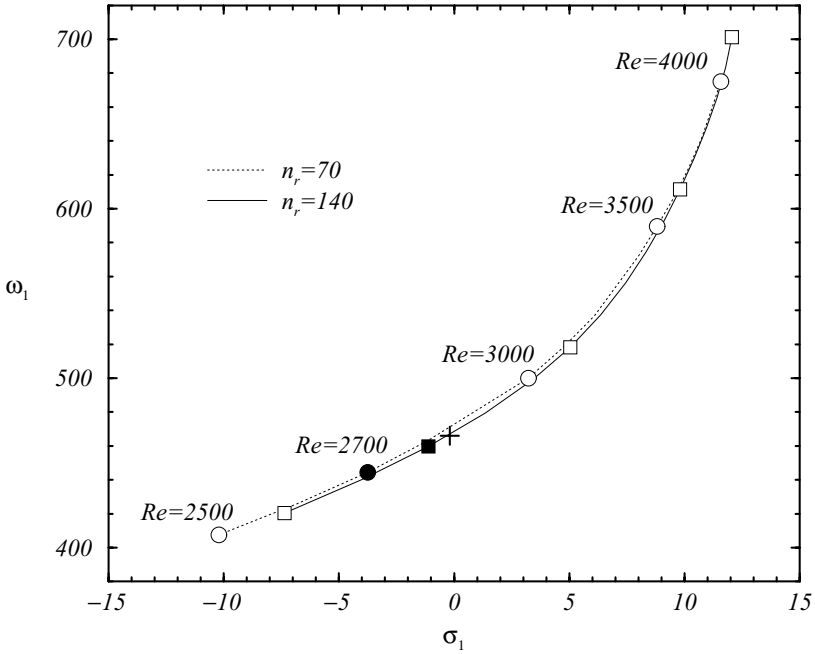
**FIG. 6.** Variations of (a)  $\sigma_1$  and (b)  $\omega_1$  of the most dangerous eigenvalue with  $n_r$ , for  $Re_b = 2700$  and  $\Lambda = 2.5$ . The solid lines are fits of the form  $a + b/r^2$  for  $n_r \geq 110$ .

the oscillations between the  $n_r = 70$  and  $n_r = 140$  cases. From this figure, we see that there is very little difference in either the temporal or spatial structure of this most dangerous mode, and so with  $n_r = 70$ , all salient features of the flow are accurately captured. To obtain a more quantitative measure of the convergence with increasing  $n_r$ , we examine how  $\sigma_1$  and  $\omega_1$  vary with  $n_r$ .

In Fig. 6,  $\sigma_1$  and  $\omega_1$  for  $Re_b = 2700$  and  $\Lambda = 2.5$  are plotted for  $n_r \in [40, 140]$  in steps of 10. We see from this figure that for  $n_r \geq 90$ , the system is well within the asymptotic regime where the computed  $\sigma_1$  and  $\omega_1$  converge as  $n_r^{-2}$ . This quadratic convergence is to be expected from our second-order scheme. The fits shown in the figure are

$$\lambda_1(n_r) = -0.18454 + 465.167i - (18230 + 108912i)/n_r^2.$$

This allows us to estimate the asymptotic value  $\lim_{n_r \rightarrow \infty} \lambda_1 = -0.18454 + 465.167i$ . Compared with the value at  $n_r = 140$ ,  $-1.1159 + 459.617i$ , we have a relative error of 1.2%. Furthermore, this error is not simply randomly oriented in the complex plane. Figure 7 shows the variation of this eigenvalue with  $Re_b$  for  $n_r = 70$  and  $n_r = 140$ ,

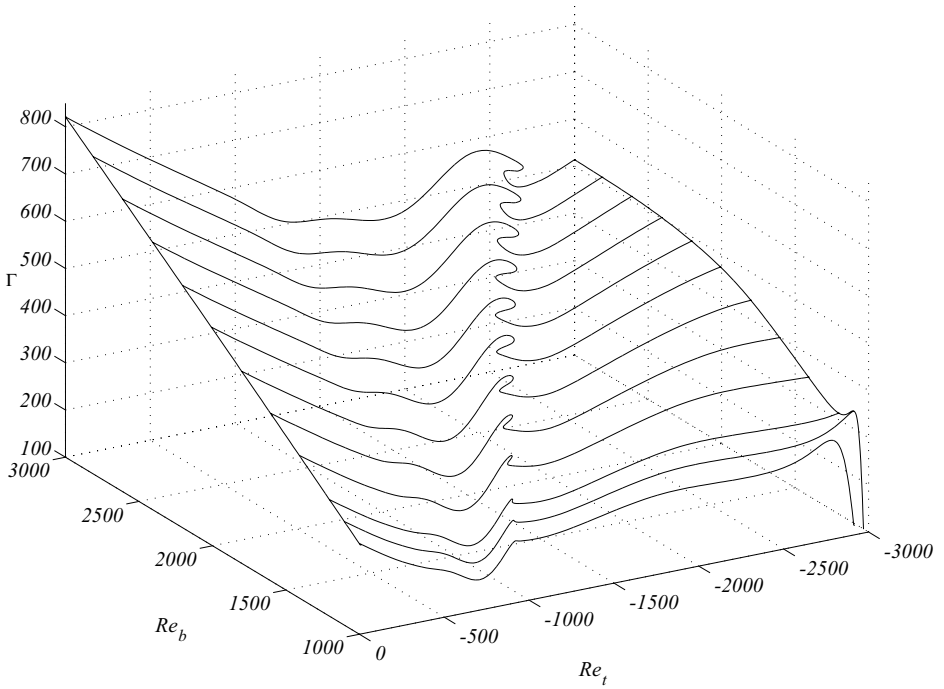


**FIG. 7.** Eigenvalues on branch 1,  $\lambda_1 = \sigma_1 + i\omega_1$ , evaluated with  $n_r = 70$  (dashed curve and circles) and  $n_r = 140$  (solid curve and squares) over a range of  $Re_t \in [2500, 4000]$  with  $\Lambda = 2.5$ . The asymptotic estimate at  $Re_t = 2700$  is the plus symbol.

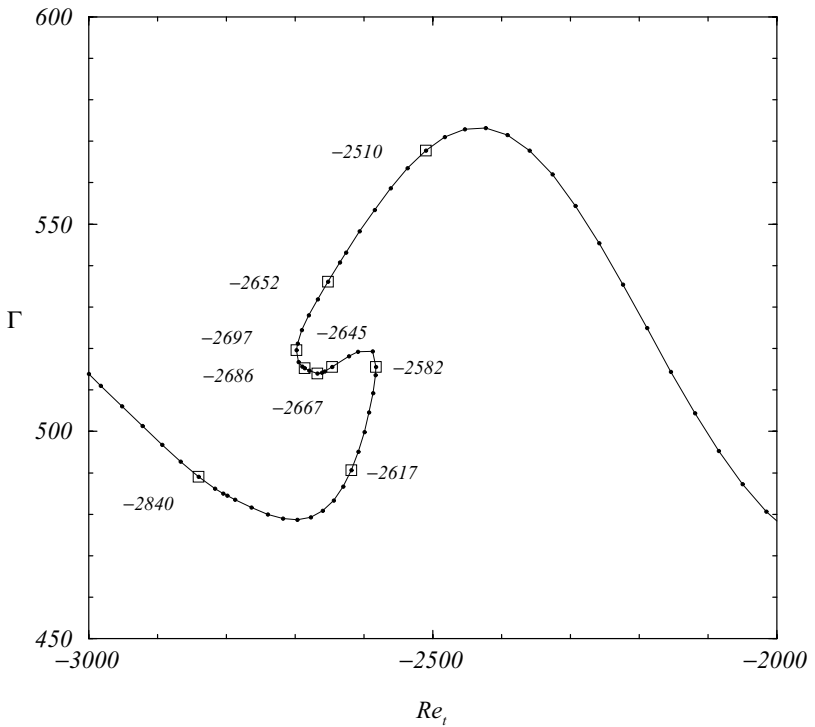
together with the asymptotic estimate at  $Re_b = 2700$ . Although there are variations due to  $n_r$  in the location of the eigenvalues in the complex plane, they are located on a well-defined curve. In essence, computations of eigenvalues using a relatively low  $n_r$  correspond to a precise computation at an effectively lower  $Re_b$ . This phenomenon can be traced back to the degree to which the Ekman layer can be resolved with finite  $n_r$ .

#### 4.2. The Cusp Bifurcation for $\Lambda = 0.5$

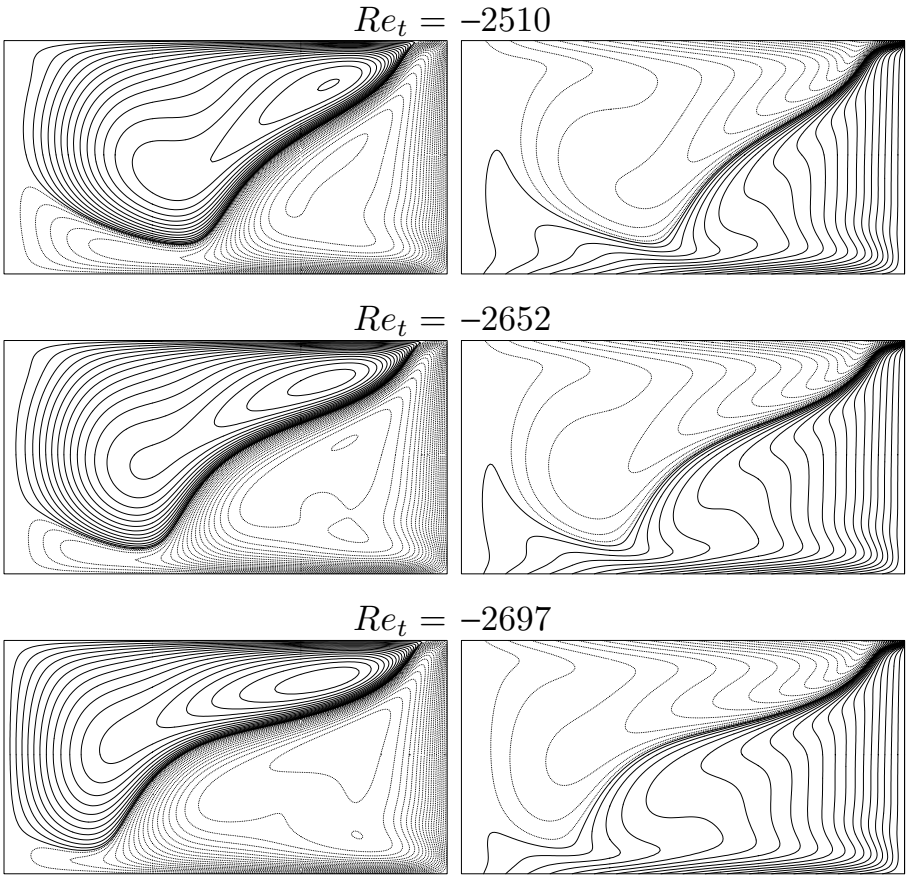
The codimension-2 cusp bifurcation for  $\Lambda = 0.5$  occurs at ( $Re_b \approx 1126$ ,  $Re_t \approx -1011$ ). From this cusp point, two lines of saddle-node bifurcations extend, with approximate equations  $Re_b = -aRe_t - b$ , with  $(a, b) = (1.304, 192.6)$  for one and  $(1.132, 18.5)$  for the other. The saddle-node lines are straight in the neighborhood of the cusp point. The loci of all these bifurcations were determined by a sequence of continuation runs, fixing  $Re_b$  and varying  $Re_t$ . The results of these runs are summarized in Fig. 8, showing the development of the fold in the solution manifold. Figure 9 shows a detail of the fold for fixed  $Re_b = 3000$ . The dots along the curve correspond to the computed steady states during the continuation run. In the fold region, there coexist three distinct equilibria on sections of the curve (branches) delimited by the two saddle-node bifurcations, two stable (top and bottom) and one unstable (middle). Note that the arclength increment is relatively small in the fold region, where the structure of the solutions changes rapidly with parameter variation. The open squares are selected equilibria on each of the three branches, and their flow structure is illustrated in Figs. 10, 11, and 12. These figures are of the streamlines (left) and vortex lines (right) projected onto a meridional ( $r, z$ )-plane. On all three branches, the most salient



**FIG. 8.** Cusp bifurcation of the basic state for  $\Lambda = 0.5$  and counterrotation ( $Re, Re_b < 0$ ). Solid lines correspond to continuation runs with fixed  $Re_b$ , with  $n_r = 200$  and  $n_z = 100$ .



**FIG. 9.** Detail of a continuation run in Fig. 8 with  $Re_b = 3000$ , showing two saddle-node bifurcations in the cusp region. Dots correspond to the steady solutions computed during the continuation, and the open squares indicate the solutions whose spatial structure is depicted in Figs. 10, 11, and 12.

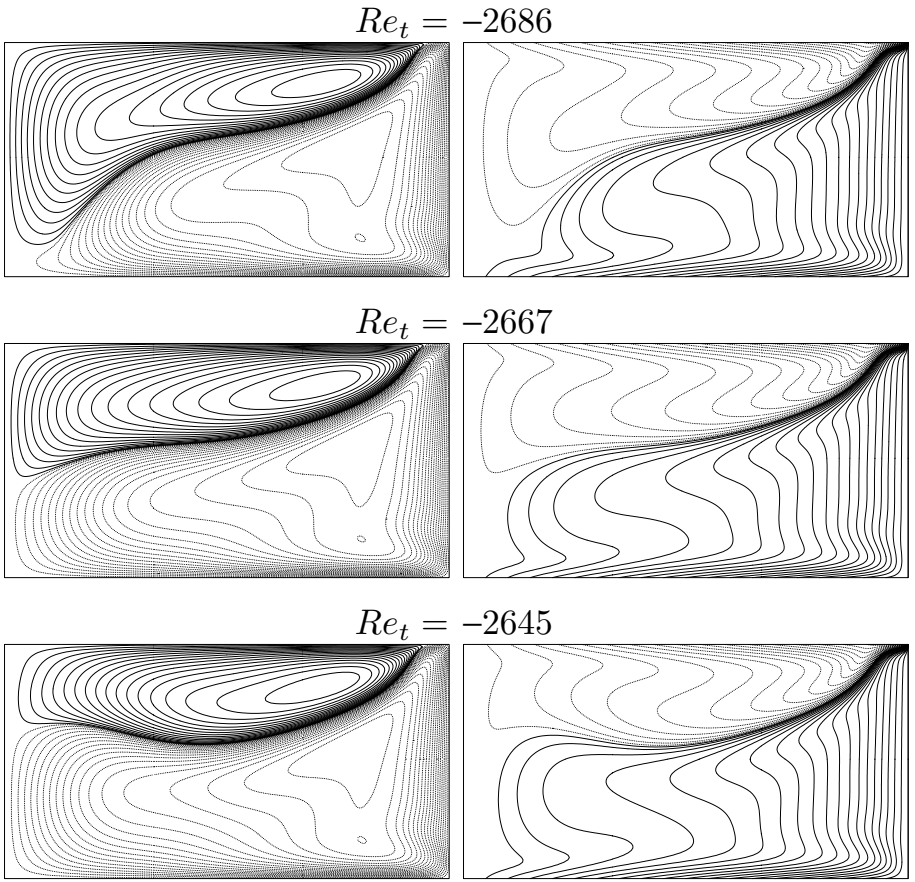


**FIG. 10.** Streamlines (left) and vortex lines (right) of the three solutions indicated on the upper branch of the fold in Fig. 9. Solid (dotted) lines correspond to positive (negative) levels.

flow feature is the shear layer originating at the corner where the rotating base meets the counterrotating top, separating fluid regions with opposite senses of azimuthal velocity. The three branches of solutions are distinguished by where the shear layer ends up.

On the top branch (Fig. 10), the shear layer attaches to the bottom Ekman layer, and on the bottom branch (Fig. 12) it reattaches to the top Ekman layer. Both of these equilibrium states are stable (in an  $SO(2)$  subspace) and can be obtained from a nonlinear temporal evolution. The middle branch (Fig. 11) is unstable; solutions near it evolve to either of the other two states depending on initial conditions. This unstable equilibrium state is characterized by the shear layer attaching to the axis, an intermediate response between those of the two competing stable equilibria.

The stability of the equilibria is rigorously determined from their corresponding spectra, depicted in Fig. 13. The development of the spectra along the continuation run shows that a single real eigenvalue crosses the imaginary axis as  $Re_t$  is varied along the top branch to the saddle-node, remains real and positive along the middle branch, and at the other saddle-node crosses back to the left of the imaginary axis. The whole spectrum is on the left-hand side of the imaginary axis along both the top and bottom branches. There is considerable variation in the spectra, corresponding to the large variation in the flow structure due to the shear layer moving across an extensive part of the enclosed cylinder throughout the fold region.



**FIG. 11.** Streamlines (left) and vortex lines (right) of the three solutions indicated on the middle branch of the fold in Fig. 9.

Several tests have been made to study the efficiency of the methods we have presented. Some have already been mentioned in relation to the determination of the optimal value of the fill-in parameter,  $l_{fil}$ , and the dimension of the Krylov subspace,  $irestart$ .

The most important factor in reducing the CPU cost of the computations is the choice of a suitable preconditioner. To solve one of the linear systems (13), a first attempt could be to use  $L$ , the Stokes operator, as a preconditioner. In the case of finite differences, including the part of the Jacobian from the nonlinear terms,  $N$ , is not difficult. Including it in a spectral code can be prohibitive. It is important to note that, in any case, the Stokes operator may not be an efficient preconditioner far from  $Re_b = Re_t = 0$ . To show this, we have computed the spectrum at the solution corresponding to  $\Lambda = 0.5$ ,  $Re_b = 3000$ ,  $Re_b = -2667$ ,  $n_r = 200$ , and  $n_z = 100$ , using three different preconditioners. The first includes the full Jacobian with the optimal value of  $l_{fil}$ , the second only contains the Stokes operator with the same  $l_{fil}$ , and the third contains the Stokes operator with the complete band in the ILU. Table II shows the CPU time needed to obtain the spectrum. The use of the full Jacobian decreases the CPU time by an order of magnitude. Preconditioning with only the Stokes operator is not efficient when the nonlinear terms in the Navier–Stokes equations are significant, and this is always the case at large Reynolds numbers. This signals a warning against naive implementations of the described methods. It can also be seen in Table II that increasing the fill-in parameter

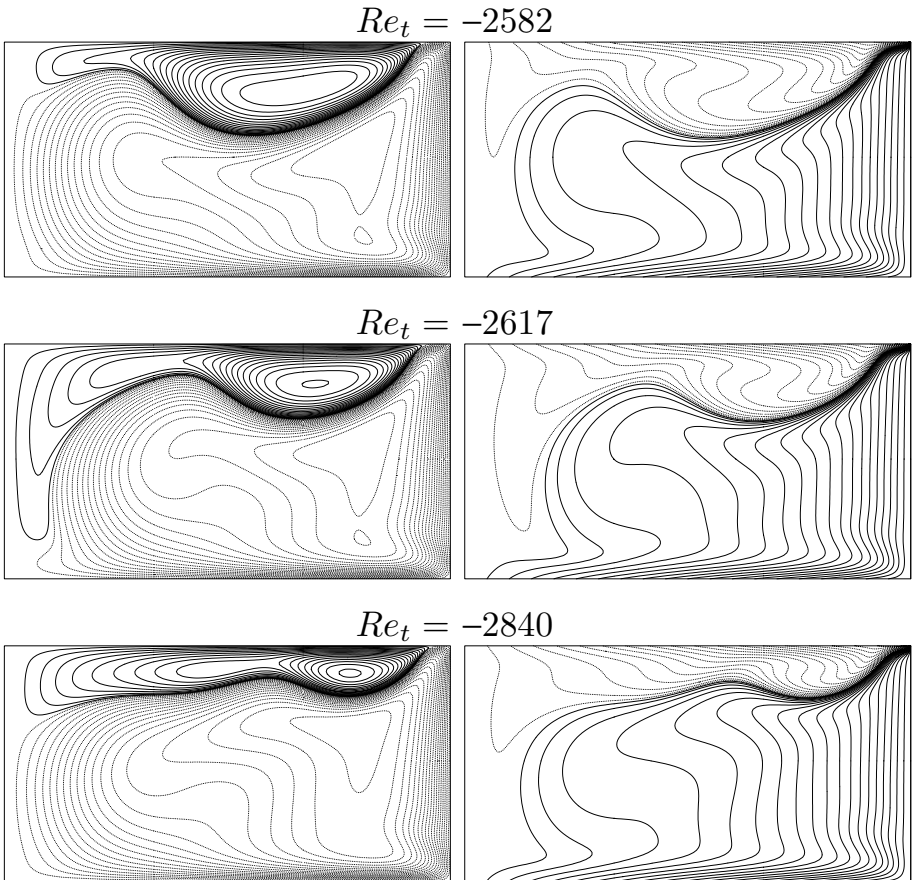


**TABLE II**  
**Comparison of the CPU Time Needed to Find a Spectrum**  
**with Different Preconditioners**

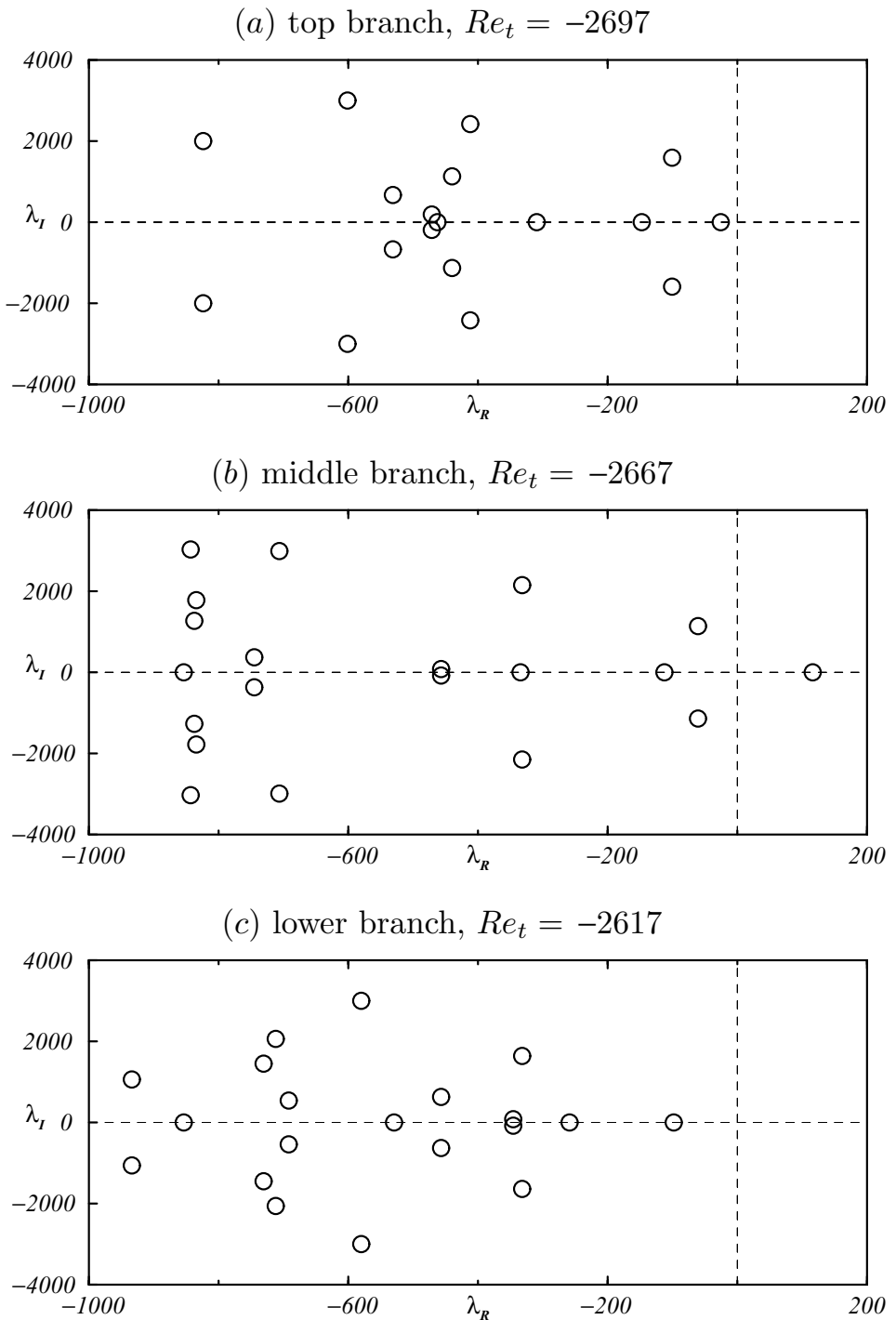
Preconditioner	CPU time (s)	<i>lfil</i>
Full Jacobian with optimal <i>lfil</i>	1,364	220
Stokes operator with optimal <i>lfil</i>	17,181	220
Stokes operator with the whole band	21,550	400

is not helpful. On the contrary, it can increase the computational cost. All computations were performed on a personal computer with an AMD Athlon processor at 800 MHz.

Finally, we have studied the dependence of the CPU time on the dimension of the system. The computational cost of solving linear systems or eigenvalue problems by direct methods grows as the third power of the dimension. Efficient methods for large problems must reduce this power noticeably. Figures 14a and b show the results for a continuation run and for the calculation of a spectrum, respectively. The first case corresponds to the continuation shown in Fig. 2. The parameters are  $\Lambda = 0.5$ ,  $Re_b = 3000$ , and  $Re_t \in [-3000, 0]$ . The computation includes several ILUs and the solution of many nonlinear systems. This would

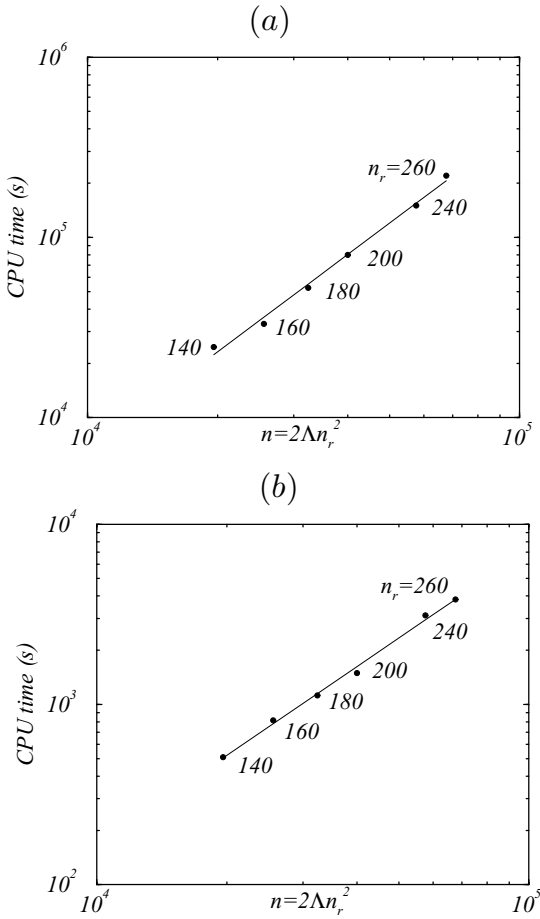


**FIG. 12.** Streamlines (left) and vortex lines (right) of the three solutions indicated on the lower branch of the fold in Fig. 9.



**FIG. 13.** Spectra for a case on each of the three branches (showing only the 20 most dangerous eigenvalues).

be a typical calculation when studying the stationary solutions of a fluid problem. The curve fit shows a power law dependence with an exponent of 1.8. For the spectrum calculations, the parameters are  $\Lambda = 0.5$ ,  $Re_b = 3000$ , and  $Re_t = -2667$ , and the power law exponent is 1.6. These results show that the methods used are suitable for problems of large dimension.



**FIG. 14.** CPU time versus (a) total number of variables, when computing the continuation curve of Fig. 2 (the slope of the adjusted straight line is 1.80), and (b) total number of variables, when computing the 20 most critical eigenvalues in the middle of the fold at  $\Lambda = 0.5$ ,  $Re_b = 3000$ , and  $Re_t = -2667$  (the slope of the adjusted straight line is 1.63).

## 5. CONCLUDING REMARKS

We have presented and shown the efficiency of a numerical package for the study of large systems. A comprehensive modular approach has been followed, and the package contains a Newton solver to determine steady states (equilibria), a linear stability solver, and a time evolution code. Any one of these components may be better suited to a particular formulation, but then the other aspects may not. In particular, if equilibria are determined via one formulation/discretization and their stability analyzed via another, the stability results may not be reliable. This is particularly the case when the flow has small-scale, dynamically important features, such as boundary and shear layers, and the small differences in their treatment using distinct formulations can, and often do, result in both quantitative and qualitative differences in the spectra.

All the linear systems in the various aspects of the problem necessitate iterative solvers due to their very large size. We have used Krylov methods, and these require efficient preconditioning. In the test cases detailed here, finite differences were used to discretize

the PDE, and the use of incomplete LU (ILU) as a preconditioner has been found to be very efficient. If spectral discretization were used, the choice of preconditioner is not so straightforward, as the associated Jacobian matrix is typically full. However, ILU can also be readily used in finite-element, finite-volume, and other schemes leading to banded and/or sparse Jacobians.

Large systems in fluid dynamics come about when the Reynolds numbers are large and the nonlinear terms (inertia) in Navier–Stokes become dynamically significant. Whereas the techniques have been applied to test problems in cylindrical geometries in this paper, they are readily applicable to problems in other geometries (the application is immediate if the coordinate system is orthogonal). Our choice of test problem for the numerical schemes is one where boundary and shear layers are dynamically significant, leading to rich and complicated bifurcations and nonlinear dynamics. These problems have been extensively studied previously, both numerically and experimentally, so there is a reliable database for comparisons. Using the present schemes, we have computed the basic states more efficiently than previously, and determined their stability with excellent agreement with previous studies. Specifically, we have studied in one case a sequence of Hopf bifurcations of the basic state (Section 4.1), and in the other (Section 4.2), a codimension-2 cusp bifurcation, with its associated saddle-nodes and hysteresis fold structure.

These stringent tests of the schemes lead to the following observations:

- It is necessary to use the full Jacobian in the preconditioner. Although using only the Stokes operator is usually sufficient for low Reynolds numbers, for the test case where inertia is important, we have observed a 10-fold speedup by using the full Jacobian. Note that our test cases actually have moderate Reynolds numbers of order  $10^3$ ; geophysical and aerodynamics applications typically have Reynolds numbers of  $10^6$  and larger, so in these cases the use of the full Jacobian is expected to have a larger impact.

- The ILU preconditioner and Arnoldi and GMRES methods have a multitude of tunable parameters. From our experience with the test cases, two stand out as needing critical attention: *lfl* that measures the maximum number of elements of each row of L and U, and *irestart* that fixes the dimension of the Krylov subspace in GMRES. Both have clearly identifiable optimal values that depend on the size of the system (i.e., the spatial resolution of the Navier–Stokes equations).

- The growth of the CPU time with the size of the system is moderate, obeying a power law with exponent around 1.7. This should be compared with direct methods, where the exponent is 3. Thus, the methods we have considered are very well suited to large systems.

## ACKNOWLEDGMENTS

This work was supported by DGICYT Grant PB97-0685 (Spain) and NSF Grants INT-9732637 and CTS-9908599 (USA).

## REFERENCES

1. C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics* (Springer-Verlag, Berlin/New York, 1998).
2. K. A. Cliffe, Numerical calculations of the primary-flow exchange process in the Taylor problem, *J. Fluid Mech.* **197**, 57 (1988).

3. P. Demaret and M. O. Deville, Chebyshev pseudospectral solution of the Stokes equations using finite element preconditioning, *J. Comput. Phys.* **83**, 463 (1989).
4. R. Dembo, S. Eisenstat, and T. Steihaug, Inexact Newton methods, *SIAM J. Numer. Anal.* **19**(2), 400 (1982).
5. N. Dinar and H. B. Keller, Computation of Taylor vortex flows using multigrid continuation methods, in *Recent Advances in Computational Fluid Dynamics*, edited by Chao, Orszag, and Shyy (Springer-Verlag, New York/Berlin, 1989). *Lecture Notes Eng.* **43**, 191.
6. W. E and J.-G. Liu, Vorticity boundary condition and related issues for finite difference schemes, *J. Comput. Phys.* **124**, 368 (1996).
7. W. S. Edwards, L. S. Tuckerman, R. A. Friesner, and D. C. Sorensen, Krylov methods for the incompressible Navier–Stokes equations, *J. Comput. Phys.* **110**, 82 (1994).
8. M. P. Escudier, Observations of the flow produced in a cylindrical container by a rotating endwall, *Expt. Fluids* **2**, 189 (1984).
9. A. Y. Gelfgat, P. Z. Bar-Yoseph, and A. Solan, Stability of confined swirling flow with and without vortex breakdown, *J. Fluid Mech.* **311**, 1 (1996).
10. A. Y. Gelfgat, P. Z. Bar-Yoseph, and A. Solan, Three-dimensional instability of axisymmetric flow in a rotating lid-cylinder enclosure. *J. Fluid Mech.* **438**, 363 (2001).
11. W. J. F. Govaerts, *Numerical Methods for Bifurcations of Dynamical Equilibria* (Soc. for Industr. & Appl. Math., Philadelphia, 2000).
12. H. B. Keller, Numerical solution of bifurcation and nonlinear eigenvalue problems, in *Applications of Bifurcation Theory*, edited by P. Rabinowitz (Academic Press, San Diego, 1977), pp. 359–384.
13. R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK User's Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods* (Soc. for Industr. & Appl. Math., Philadelphia, 1998).
14. J. M. Lopez, Axisymmetric vortex breakdown: Part 1. Confined swirling flow, *J. Fluid Mech.* **221**, 533 (1990).
15. J. M. Lopez, Characteristics of endwall and sidewall boundary layers in a rotating cylinder with a differentially rotating endwall, *J. Fluid Mech.* **359**, 49 (1998).
16. J. M. Lopez, F. Marques, and J. Sanchez, Oscillatory modes in an enclosed swirling flow, *J. Fluid Mech.* **439**, 109 (2001).
17. L. S. Mamun and L. S. Tuckerman, Asymmetry and Hopf bifurcation in spherical Couette flow, *Phys. Fluids* **7**, 80 (1995).
18. K. Meerbergen and D. Roose, Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems, *IMA J. Numer. Anal.* **16**, 297 (1996).
19. R. Meyer-Spasche and H. B. Keller, Computation of the axisymmetric flow between rotating cylinders, *J. Comput. Phys.* **35**, 100 (1980).
20. Y. Saad, *Numerical Methods for Large Scale Problems* (Manchester Univ. Press, Manchester, 1992).
21. Y. Saad, *SPARSKIT: A Basic Tool Kit for Sparse Matrix Computations, Version 2* (Univ. of Minnesota Press, Minneapolis, 1994).
22. Y. Saad, *Iterative Methods for Sparse Linear Systems* (PWS–Kent, Boston, 1996).
23. W. Schröder and H. B. Keller, Wavy Taylor-vortex flows via multigrid-continuation methods, *J. Comput. Phys.* **91**, 197 (1990).
24. J. L. Stevens, J. M. Lopez, and B. J. Cantwell, Oscillatory flow states in an enclosed cylinder with a rotating endwall, *J. Fluid Mech.* **389**, 101 (1999).
25. W. Stewart and A. Jennings, A simultaneous iteration algorithm for real matrices, *ACM Trans. Math. Soft.* **7**(2), 184 (1981).
26. S. J. Tavener and K. A. Cliffe, Primary flow exchange mechanisms in the Taylor apparatus applying impermeable stress-free boundary conditions, *IMA J. Appl. Math.* **46**, 165 (1991).
27. G. I. Taylor, Stability of a viscous liquid contained between two rotating cylinders, *Phil. Trans. R. Soc. Lond. A* **223**, 289 (1923).
28. N. Tsitverblit, Vortex breakdown in a cylindrical container in the light of continuation of a steady solution, *Fluid Dynam. Res.* **11**, 19 (1993).