Pergamon

0045-7930(94)00042-5

Computers & Fluids Vol. 24, No. 4, pp. 415–434, 1995
Copyright © 1995 Elsevier Science Ltd
Printed in Great Britain. All rights reserved
0045-7930/95 $9.50 + 0.00

AN EFFICIENT CODE TO COMPUTE NON-PARALLEL STEADY FLOWS AND THEIR LINEAR STABILITY

HENK A. DIJKSTRA,[1] M. JEROEN MOLEMAKER,[1]
AUKE VAN DER PLOEG[2] and EUGEN F. F. BOTTA[2]

[1]Institute for Marine and Atmospheric Research, University of Utrecht, Utrecht, The Netherlands

[2]Department of Mathematics, University of Groningen, Groningen, The Netherlands

(Received 14 February 1994; in revised form 30 November 1994)

Abstract—A simple, fast and efficient algorithm to compute steady non-parallel flows and their linear stability in parameter space is described. The pseudo-arclength continuation method is used to trace branches of steady states as one of the parameters is varied. To determine the linear stability of each state computed, a generalized eigenvalue problem of large order is solved. Only a prescribed number of eigenvalues, those closest to the imaginary axis, are calculated by a combination of a complex mapping and the Simultaneous Iteration Technique. The underlying linear systems are solved with preconditioned Bi-CGSTAB. It is shown that it is possible to deal efficiently with (discretized) problems with $O(10^5)$ degrees of freedom. As an application, the bifurcation structure of steady two-dimensional Rayleigh–Bénard flows in large rectangular containers (up to aspect ratio 20) is computed. We show how the results connect up with those obtained with weakly nonlinear theory and extend these into the nonlinear regime. Main aim is to investigate whether pattern selection occurs through the occurrence of saddle node bifurcations creating intervals of unique steady states. It turns out that these intervals do not exist; multiple stable states continue to exist at large aspect ratio over a large range of Rayleigh numbers. In addition, the bifurcation structure provides no answer why the 'preferred' wavelength increases with increasing Rayleigh number, as observed in experiments.

1. INTRODUCTION

In contrast to the situation for parallel flows, relatively little is known on the linear stability of non-parallel flows. This is not because the latter flows are not interesting; the mathematical problem is just much harder to solve. For parallel flows, the problem of linear stability can be reduced to a 2-point boundary (eigen) value problem. However, for non-parallel flows the problem becomes, in most cases, an elliptic eigenvalue problem. The latter is much more difficult to solve, even numerically. The computation of the linear stability of non-parallel flows is particularly important when studying the multiplicity of steady confined flows (depending on several parameters) in parameter space. Transitions to different steady patterns or to time-dependent flows may occur as a value of a (forcing) parameter is changed. A well-known example is the appearance of Taylor vortices between rotating concentric cylinders [1].

In this paper, we consider a relatively old problem which has been studied for many decades, i.e. pattern selection due to buoyancy driven convection in a layer of a Newtonian liquid heated from below. We were stimulated by the recent book of Koschmieder [2], wherein he states that "we do not seem to know why or by what mechanism the convective motions select, out of a continuum of unstable wavenumbers available according to linear theory and weakly nonlinear theory, one particular wavenumber, or perhaps a narrow band of wavenumbers at a particular supercritical Rayleigh number". In his book, he describes that one of the main discrepancies between weakly nonlinear theory and experiments in large aspect ratio boxes is, that weakly nonlinear theory predicts a decrease in wavelength of the 'preferred' (i.e. realized) pattern, whereas in experiments the opposite is observed. It is questioned, whether a 'preferred' pattern—which suggests a certain uniqueness—really exists. Getling, [3], for instance, argued that the realized state may not be related to stability but to the combined action of selective and anti-selective factors, determined by the initial and boundary conditions. Attempts have been made to characterize the 'preferred' pattern through extremals based on simple properties of the flow, for example maximum heat transfer. However, no such principle has been convincingly demonstrated to exist [4].

There are basically two different theoretical approaches to the pattern selection problem at slightly supercritical Rayleigh numbers Ra. The starting point of the modulation equation approach is the critical state of the infinite layer (the minimum of the neutral curve). For values of Ra slightly larger than the critical value, Ra_c, a band of wavenumbers becomes unstable. Modulation equations describe the evolution of the flow due to interaction of these unstable modes. By careful analysis, the effect of distant no-slip lateral boundaries (a distance L apart) can be taken into account. In a series of papers [5–7], the structure of steady states and their linear stability was investigated for the two-dimensional problem with stress free horizontal boundaries. Near onset, formally when $\mathrm{Ra} - \mathrm{Ra}_c = O(L^{-2})$, two pairs of (symmetry related) solutions branch off from the conduction solution. Only the pair of solutions which arises at the first bifurcation is stable. At larger forcing, formally when $\mathrm{Ra} - \mathrm{Ra}_c = O(L^{-1})$, other stable states appear (the so-called phase winding solutions) and their structure and stability depend on the Prandtl number Pr [7]. Either there is a smooth evolution through the range $\mathrm{Ra} - \mathrm{Ra}_c = O(L^{-1})$ with little change in the wavelength (at large Pr) or there is a discontinuous evolution in which to preserve stability, the solution must jump from one solution branch to another (at small Pr). In both cases, sidewalls restrict the number of stable flow patterns compared to those possible in an infinite layer.

Another approach is through local bifurcation analysis of steady states in finite aspect ratio containers. For small aspect ratio containers the paths of primary bifurcation points is quite different for 'slippery' than for rigid sidewalls [8]. The neutral curve is composed of parts of two intertwining curves, along which the most unstable pattern changes (see [9] for a related case) as the aspect ratio is varied. Although the curves closely approach as the aspect ratio becomes large, the bifurcation picture remains that of two closely spaced pitchfork bifurcations, of which the first secondary branches are stable, whereas the other secondary branches are unstable. Secondary bifurcations might introduce stable mixed modes and/or change the stability of both secondary branches, depending on the Prandtl number. The analysis in [8] was done for rigid horizontal walls and was probably not continued to large aspect ratio because the eigenfunctions (at primary bifurcation points) had to be computed numerically.

The results from modulation equations (where no-slip sidewalls are taken into account through a boundary layer analysis) and local bifurcation theory connect up at large aspect ratio A. Some 'phase winding' solutions must correspond to the mixed modes which appear through secondary bifurcation. However, both approaches have the drawback that they can only be justified close to the critical state. In this study, we address again the question of pattern selection in two-dimensional, large aspect ratio, rectangular containers by computing the bifurcation structure of steady two-dimensional flow patterns numerically for Rayleigh numbers far into the nonlinear domain. The two-dimensionality of the flow (at not too small Pr and at large A), for Ra up to about $10 * \mathrm{Ra}_c$ is reasonable and hence our results should be able to describe laboratory experiments quite well. To be able to check the range of validity of the asymptotic results provided by weakly nonlinear theory and to extend these into the full nonlinear range, one must be able to compute steady states as a function of Ra for large A. Time integration is not very suited to compute these states, because the approach to steady state is very slow [10]. Another disadvantage of time integration is that unstable steady states cannot be computed. Although these states are not physically relevant, transitions of these states through secondary bifurcations, often lead to stable states.

Continuation techniques [11, 12] have been used successfully in many examples to follow branches of unstable and stable steady states in parameter space. The calculation of the linear stability is important since otherwise a false picture of the physically relevant flows may arise. For example, in [13] a multitude of steady (double diffusive) patterns was found in a laterally heated stratified liquid. However, it was shown [14] that most of these flows are unstable leaving a very simple picture of realizable flows. Dijkstra [15] investigated the bifurcation structure of Rayleigh–Bénard–Marangoni flows in small aspect ratio containers using pseudo-arclength continuation. The linear stability of each steady state was simultaneously determined by solving the associated generalized eigenvalue problem The most time consuming part of the code was the solution of the underlying linear systems. Two types of linear systems have to be solved: one type arising after Newton linearization of the nonlinear algebraic systems of equations, the other type within the eigenvalue solver. In [15] this was done with a direct (band) solver which severely constrained the number of degrees of freedom of the particular system. This limited the study of the multiplicity of two-dimensional Rayleigh–Bénard–Marangoni flows to rectangular containers with a maximum aspect ratio A of 4.

To compute a sufficiently accurate bifurcation structure for containers with A up to 20, the direct linear systems solver used in [15] is replaced by an iterative solver. Here, we use an efficient version of a preconditioned gradient like method as presented in Van der Ploeg [16]. The type of preconditioner has already proved to be very efficient for a class of standard problems. It turns out to work well if applied to the class of less conditioned linear systems which are met in continuation problems involving a large class of non-parallel flows. This leads to a very efficient method to compute steady non-parallel flows and their linear stability in parameter space. It appears that this is the first version of a continuation algorithm in which iterative linear solvers are used and therefore can deal with systems having very large degrees of freedom. All similar codes known to the authors are designed for systems with relatively small degrees of freedom [17–19] since direct solvers are used and full eigenvalue problems are solved with standard techniques. We therefore present the main numerical methods in some detail in Section 2 of this paper, together with some elementary results on the performance of the eigenvalue solver.

The Rayleigh–Bénard problem is considered in Section 3. The results near onset confirm most of the results of weakly nonlinear theory. The main new results are the computation of the bifurcation structure of the steady states in the nonlinear regime. For A = 10, velocity profiles computed show very good agreement with measurements. Finally, we investigate whether pattern selection is induced through the occurrence of saddle node bifurcations far above onset. In small aspect ratio containers, it was shown in [15] that these saddle node bifurcations lead to intervals in the stability parameter, with in each a unique stable steady state. For A = 20 and Pr = 7, the result is a negative one: these intervals do not exist. The pattern selection as observed in experiments is not related to this type of bifurcation structure.

Our eventual goal is to apply these techniques to study the multiplicity of patterns of the large scale ocean circulation. Although large scale numerical ocean models have been developed over the last decades, they all use time-integration and are therefore not very suited (because of the large spin-up times) for these type of studies. Although some technical difficulties (memory usage, choice of parameters in the preconditioner, choice of testfunctions) will probably arise from application to application, the algorithm used here has the potential to perform these type of bifurcation studies in the near future.

2. DESCRIPTION OF THE ALGORITHM

The algorithm is comprised of two main pieces: the continuation method to advance one step on a branch of steady states as a parameter is varied and an eigenvalue solver to determine the linear stability of the new computed steady state.

2.1. Computation of steady non-parallel flows in parameter space

When a set of time-independent partial differential equations is discretized by some finite difference, finite volume or finite element method, a system of nonlinear algebraic equations

$$\mathbf{F}(\mathbf{u}, \mathbf{p}) = 0 \tag{1}$$

emerges. Here $\mathbf{u}$ is a d-dimensional vector consisting of the unknowns at the gridpoints, $\mathbf{p}$ is the n_p-dimensional vector of parameters and $\mathbf{F}$ is a nonlinear mapping from $R^d \times R^{n_p} \rightarrow R^d$. To determine branches of steady solutions of (1) as one of the parameters (say μ) is varied, the pseudo-arclength method is used. This method is well described in [20]; the branches $(\mathbf{u}(s), \mu(s))$ are parametrized by an 'arclength' parameter s. An additional equation is obtained by 'normalizing' the tangent

$$\dot{\mathbf{u}}_0^{\mathrm{T}}(\mathbf{u} - \mathbf{u}_0) + \dot{\mu}_0(\mu - \mu_0) - \Delta s = 0 \tag{2}$$

where $(\mathbf{u}_0, \mu_0)$ is an analytically known starting solution or a previously computed point on a particular branch and Δs is the steplength. To solve the system of equations (1, 2) Euler–Newton continuation is used. The Jacobian matrix $\mathscr{J}(s)$ of (1, 2) along a branch is given by

$$\mathscr{J}(s) \equiv \begin{bmatrix} \Phi & \mathbf{F}_\mu \\ \dot{\mathbf{u}}_0^{\mathrm{T}} & \dot{\mu}_0 \end{bmatrix} \tag{3}$$

where Φ is the Jacobian $\mathbf{F}_u$. During one Newton iteration, linear systems of the form

$$\Phi \mathbf{z} = \mathbf{y} \tag{4}$$

have to be solved where $\mathbf{z}$ and $\mathbf{y}$ are $(d \times 2)$ matrices.

To monitor singularities on a particular branch, several indicator functions are used. For example, limit points are detected by following $\dot{\mu}$. Other singularities, like Hopf bifurcation points, must be detected by solving the linear stability problem. Only for simple bifurcation points (i.e. transcritical—or pitchfork bifurcations) there are cheaper alternatives, for example the sign of $\det(\boldsymbol{\Phi})$. A family of testfunctions τ_{pq} is obtained as follows [21]: let $\boldsymbol{\Phi}_{pq}$ be the matrix $\boldsymbol{\Phi}$ in which p^{th} row is replaced by the q^{th} unit vector. If we solve the linear system

$$\boldsymbol{\Phi}_{pq} \mathbf{v} = \mathbf{e}_p \tag{5}$$

for $\mathbf{v}$, where $\mathbf{e}_p$ is the p^{th} unit vector, then it can be shown [21] that

$$\tau_{pq} = \mathbf{e}_p^{\mathrm{T}} \boldsymbol{\Phi} \mathbf{v} \tag{6}$$

changes sign as $\boldsymbol{\Phi}$ becomes singular. In principle, the choices of q and p are arbitrary as long as $\boldsymbol{\Phi}_{pq}$ is nonsingular. Of course, for any solution method, it is advantageous that $\boldsymbol{\Phi}_{pq}$ and $\boldsymbol{\Phi}$ have the same structure. However, in specific problems, not all values of q and p can be chosen and it is advisable to make a choice based on the knowledge of the (symmetry properties of the) solutions of the particular problem. For each testfunction τ_{pq} computed, we have to solve one additional linear system.

2.2. *Computation of the linear stability of a particular steady state*

In most applications, discretization of the equations governing the evolution of infinitesimal disturbances on a particular steady state leads to a generalized eigenvalue problem

$$\mathscr{A} \mathbf{x} = \sigma \mathscr{B} \mathbf{x} \tag{7}$$

where $\mathscr{A}$ is a nonsingular, non-symmetric $d \times d$ matrix [typically $d = O(10^4 - 10^5)$]. Through Dirichlet boundary conditions and/or the incompressibility condition, $\mathscr{B}$ may become singular.

The eigenvalues of the pensil (the set of all matrices of the form $\mathscr{A} - z\mathscr{B}$ with $z \in C$) are elements of the set

$$\sigma(\mathscr{A}, \mathscr{B}) = \{z \in C \mid \det(\mathscr{A} - z\mathscr{B}) = 0\} \tag{8}$$

If $\mathscr{B}$ is nonsingular, the problem reduces to an ordinary eigenvalue problem for the matrix $\mathscr{B}^{-1}\mathscr{A}$. Because only real matrices are considered, there are d eigenvalues which are either real or occur as complex conjugate pairs. However, if $\mathscr{B}$ is singular, the eigenvalue structure may be more complicated; the set $\sigma(\mathscr{A}, \mathscr{B})$ may be finite, empty or even the whole complex plane [22].

Traditional eigenvalue solvers (e.g. the QZ algorithm [22], p. 251ff) which determine all eigenvalues and, if desired, all eigenvectors are impossible to use. In many hydrodynamic stability problems, the instability of a certain steady flow pattern occurs only through a few number of modes. In the discretized linear stability problem the set $\sigma(\mathscr{A}, \mathscr{B})$ is finite and one is only interested to compute a few eigenmodes, i.e. those with eigenvalues closest to the imaginary axis (the 'most dangerous' modes). This motivated several studies to develop specific algorithms for this task. Goldhirsch *et al.* [23] present three different versions of such an algorithm suited for (ordinary) non-symmetric eigenvalue problems. Christodoulou and Scriven [24] use a combination of spectral transformations and the Arnoldi algorithm [25] and apply this to compute the linear stability of steady (coating) flows.

A variant of the methods in [24] was suggested in Dijkstra [15], where a combination of a spectral transformation and the Simultaneous Iteration Technique [26], abbreviated from now on as SIT, was used. As in [24], the idea of the algorithm is to transform the eigenvalue problem in such a way that the most dangerous modes become the most dominant modes (i.e. those with eigenvalues with largest norm). In this way, generalized power methods can be used on the transformed problem.

The first step of the eigenvalue algorithm is the application of a complex mapping

$$\sigma = b + a\frac{\lambda - 1}{\lambda + 1} \tag{9}$$

where $b \in R$, $a \in R^+$. The parameter b introduces a shift of the spectrum over the real axis, whereas the parameter a stretches the spectrum. The left complex plane $\mathscr{R}(\sigma - b) < 0$ is mapped within the unit circle $|\lambda| < 1$ (see Fig. 1 for $a = 1$, $b = 0$) and $\mathscr{R}(\sigma - b)$ is mapped onto the unit circle.

The eigenvalue problem (7) transforms with (9) to

$$(\mathscr{A} + (a - b)\mathscr{B})\mathbf{x} = -(\mathscr{A} - (a + b)\mathscr{B})\lambda\mathbf{x} \tag{10}$$

Let $\mathscr{C} = \mathscr{A} + (a - b)\mathscr{B}$ and $\mathscr{D} = -\mathscr{A} + (a + b)\mathscr{B}$. Although $\mathscr{B}$ is singular, the matrices $\mathscr{C}$ and $\mathscr{D}$ are generically not singular and we therefore consider the problem

$$\mathscr{D}^{-1}\mathscr{C}\mathbf{x} = \lambda\mathbf{x} \tag{11}$$

The properties of the mapping (9) are such that the most dangerous modes of (7) are mapped onto the most dominant modes (i.e. with largest norm) of (11). The eigenvalues $\sigma = \pm\infty$ are mapped onto $\lambda = -1$.

To determine a prescribed number of dominant modes, the SIT is particularly suited. This algorithm is well described in [26] and we only repeat the essentials. Let the desired number of accurate eigenvalues and eigenvectors be $n_f - 2$, then one starts with n_f vectors of length d, the starting vectors; let the $d \times n_f$ matrix be $\mathbf{U}^0$. The SIT consists of a filtering stage and a reorientation stage. During the filtering stage, components in the direction of the dominant eigenmodes increase in amplitude. A prescribed number, say l, of matrix vector products of the form

$$\mathbf{U}^{n+1} = \mathscr{D}^{-1}\mathscr{C}\mathbf{U}^n \tag{12}$$

and thereto computed. Hence, linear systems of the form

$$\mathscr{D}\mathbf{U}^{n+1} = \mathscr{C}\mathbf{U}^n \tag{13}$$

have to be solved. After each filtering stage, there is a reorientation stage in which an approximation to the eigenvalues λ is found by solving a small eigenvalue problem of order n_f with standard methods, for instance QR. The whole process of filtering and reorientation has to be performed a number of times, say k, until convergence (details in [26]). When converged according to a certain stopping criterion, the matrix $\mathbf{U}$ gives the eigenvectors corresponding to the n_f eigenvalues λ of which $n_f - 2$ are accurate. The eigenvalues σ are found from (9).

2.3. *The solution of the linear systems*

For both the computation of the steady states as well as the determination of its linear stability, linear systems $A\mathbf{x} = \mathbf{b}$ have to be solved. The Bi-CGSTAB method [27] is used as an iterative method for the solution of these linear systems. This is a conjugate gradient-like method which can be used for systems of linear equations in which the coefficient matrix is non-symmetric. Its

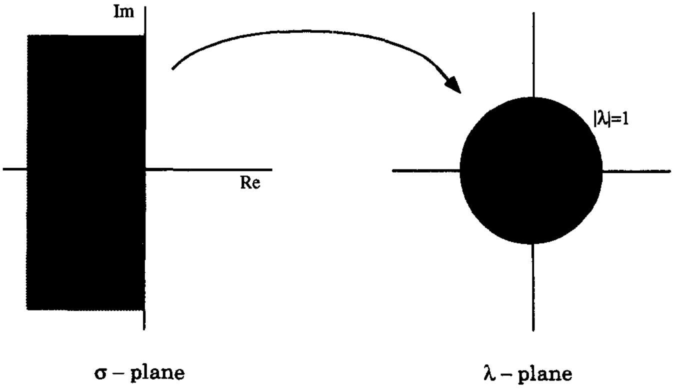

Fig. 1. The complex mapping (9) for $a = 1$, $b = 0$ in the complex plane.

convergence behavior is strongly influenced by the location of the eigenvalues of the coefficient matrix, and it appears to be very important that the spectral condition number of this matrix is small [28, 29]. For this reason we apply Bi-CGSTAB not to the $d \times d$ system $\mathscr{A}\mathbf{x} = \mathbf{b}$, but to the preconditioned system

$$\mathscr{P}^{-1}A\mathbf{x} = \mathscr{P}^{-1}\mathbf{b}$$

The non-singular matrix $\mathscr{P}$ is called the preconditioner. There is a wide choice of preconditioners, see for example [30]. The matrix $\mathscr{P}$ should have the following properties.

1. It should be a proper approximation of A such that $\mathscr{P}^{-1}A$ resembles the identity matrix. The most important quality of a preconditioner is to reduce the spectral condition number of the preconditioned matrix $\mathscr{P}^{-1}A$.
2. The preconditioner should be cheap to compute, and it should be possible to solve the system $\mathscr{P}\mathbf{y} = \mathbf{c}$ for given $\mathbf{c}$ in $O(d)$ operations.
3. It should not require a large amount of storage.

For sparse matrices which are diagonally dominant, a very effective way of obtaining a proper preconditioning matrix is to proceed with an LU-decomposition, but to preserve the sparsity in the factors L and U by ignoring some or all elements causing fill-in additional to that of A. The matrix $\mathscr{P}$ is then chosen equal to LU. In this way one obtains the splitting $(LU, -R)$, in which $R = A - LU$ is the so-called residual matrix. In principle, any conjugate gradient like method suitable for non-symmetric systems can be applied to the preconditioned system

$$U^{-1}L^{-1}A\mathbf{x} = U^{-1}L^{-1}\mathbf{b}$$

In this paper, we use an incomplete LU-decomposition in which the sparsity pattern of $L + U$ is based on a drop tolerance ϵ_p. First the matrix is scaled in such a way that

$$\sum_{j=1}^{d} |a_{ij}| = 1 \quad \text{for } 1 \leqslant i \leqslant d$$

Next, we construct a splitting $(LU, -R)$ in which all entries of R are in absolute value smaller than ϵ_p. Herein, ϵ_p is a threshold parameter which has to be chosen in advance. When neglecting the effect of roundoff errors, we make an exact decomposition of the matrix $A - R$. No restriction is made with respect to the sparsity pattern of the coefficient matrix, so the method can be used even when a complicated domain or an irregular node numbering is used. The incomplete decomposition is constructed row by row. Given the first $i - 1$ rows of L and U, we construct row i of L and U from $R = A - LU$ as

$$r_{ik} = a_{ik} - \sum_{j=1}^{\min(i,k)} l_{ij}u_{jk} \tag{14}$$

Suppose l_{ij} has been calculated for $j < k$. If $k < i$ one obtains from (14)

$$r_{ik} + l_{ik}u_{kk} = a_{ik} - \sum_{j=1}^{k-1} l_{ij}u_{jk}, \quad k < i \tag{15}$$

If the absolute value of the right-hand side of this equation is less than ϵ_p, fill-in on position (i, k) is neglected; otherwise l_{ik} is calculated from (15) together with $r_{ik} = 0$. With $l_{ii} = 1$, u_{ii} can be calculated from

$$u_{ii} = a_{ii} - \sum_{j=1}^{i-1} l_{ij}u_{ji} \tag{16}$$

If $k > i$ one obtains from (14) together with $l_{ii} = 1$

$$r_{ik} + u_{ik} = a_{ik} - \sum_{j=1}^{i-1} l_{ij}u_{jk}, \quad k > i \tag{17}$$

If the absolute value of the right-hand side of (17) is greater than or equal to ϵ_p, we demand r_{ik} to be zero; otherwise fill-in position (i, k) is neglected. The construction of the factors L and U is described in more detail in [16]. The associated CPU-time required is relatively small, not more than the time for one or two iterations with Bi-CGSTAB. From numerical experiments it is found that the choice of ϵ_p is not very critical, and all values in the range 0.001–0.01 perform well. When several systems of linear equations have to be solved in which the coefficient matrices do not differ very much, as in continuation methods, it is possible to use the same preconditioner several times. In that case it may be advantageous to choose a relatively small value for ϵ_p.

In order to increase the efficiency of the incomplete decomposition, it is possible to perform a renumbering of the unknowns which is based on the same basic idea as in multigrid methods: many iterative methods can eliminate high-frequency errors very effectively, but they are inefficient at eliminating a long wavelength error. A couple of iteration steps result in an approximation with a smooth error. This error can therefore be well corrected on a coarser grid. Solving the equations on the coarse grid gives the two-grid method. Applying this idea recursively on coarse and coarser grids leads to the multigrid method. Our preconditioning technique uses a partition of the unknowns based on a similar sequence of grids as in multigrid. Renumbering the unknowns according to this partition enables us to construct an incomplete LU-decomposition which can be used in eliminating effectively both high- and low-frequency errors. The renumbering of the unknowns as described above results in a system of linear equations which can be written as

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

where x_1 is the vector containing the unknowns of the first level, and x_2 those of higher levels. This partitioning of the matrix can be repeated for the matrix in the lower-right corner until we arrive at the coarsest grid. The preconditioning technique consists now of making a splitting $(LU, -R)$ of A, in which the elements r_{ij} of the residual matrix $R = A - LU$ satisfy

$$|r_{ij}| \leqslant \epsilon_{ij} \quad \text{for} \quad 1 \leqslant i,j \leqslant d$$

The drop tolerance ϵ_{ij} is chosen in such a way that one obtains a residual matrix with relatively small elements r_{ij} in the lower-right corner. For simple constructed problems, an expression for this drop tolerance can be derived in such a way that the condition number of the preconditioned matrix has a small upperbound independent of the dimension of the system [31]. Several numerical experiments demonstrate that this preconditioning technique is of interest in much more general cases, for example, when the mesh size is far from constant, or when the partial differential equation contains dominating convective parts. In summary, the computational work to solve a linear system of equations consists of three parts: the construction of the preconditioner (with or without renumbering), its application and the number of iterations with Bi-CGSTAB. From numerical experiments [16] we conclude that the computational time used to construct the preconditioner grows linearly with the number of unknowns. Its application is linear with the fill-in. This fill-in is about a factor two or three larger than that of A, which is modest.

2.4. *Summary of the total algorithm*

The total algorithm combining the continuation method with the eigenvalue solver and the iterative linear systems solver is summarized as follows. Suppose a point $(\mathbf{u}_0, \mu_0)$, the tangent $(\dot{\mathbf{u}}_0, \dot{\mu}_0)$ and n_f eigenvectors (and eigenvalues $\sigma_1, \ldots, \sigma_{n_f}$) are computed, then

1. Compute the Euler guess: $\mathbf{u} = \mathbf{u}_0 + \Delta s \dot{\mathbf{u}}_0$; $\mu = \mu_0 + \Delta s \dot{\mu}_0$
2. Compute the Jacobian $\boldsymbol{\Phi}$ and the preconditioning matrix $\mathscr{P}_N$ with a specific choice of the drop tolerance ϵ_p.
3. Solve the system (1, 2) in a Newton iteration with initial guess from step 1 until convergence in m steps. This requires the solution of $2 * m$ systems of linear equations with (fixed) preconditioning matrix $\mathscr{P}_N$.
4. Compute the matrix $\mathscr{D}$ in (13) and determine the preconditioning matrix $\mathscr{P}_E$ with a certain choice of ϵ_p. Start the Simultaneous Iteration Technique with the n_f starting (eigen)vectors. With filtering index l, and k reorientations until convergence, this requires the solution of $n_f * l * k$ linear systems with (fixed) preconditioning matrix $\mathscr{P}_E$.
5. Compute a desired number, say n, of testfunctions τ_{pq}. For each testfunction τ, a linear system must be solved with preconditioning matrix $\mathscr{P}_N$.

For each problem, the user has to supply the system of nonlinear equations (1) and the Jacobian matrix of that system ($\boldsymbol{\Phi}$) in a particular form as desired by the Bi-CGSTAB algorithm [16]. Apart from the choice of numerical parameters, such as ϵ_p and a suitable choice of the testfunctions τ_{pq} the remainder of the code is totally application independent.

It is not a priori clear that the iterative method is superior, with respect to CPU time, to the original algorithm [15] in which direct linear systems solvers were used. In total $n_f * l * k + 2 * m + n$ linear systems have to be solved for one complete step on a branch. The convergence of the Bi-CGSTAB algorithm combined with the preconditioner should therefore be fast. Obviously, with respect to memory usage, the iterative algorithm is strongly favorable. The fill-in during the incomplete decomposition (dependent on ϵ_p) is much smaller than that of a complete decomposition. In the following test problem we compare the performance of the direct and iterative solution technique of the linear systems as applied in the generalized eigenvalue solver. This is the most expensive part of the code, since the largest number of linear has systems to be solved. To avoid complications arising due to infinite eigenvalues we choose a very simple ordinary eigenvalue problem of which the eigenvalues are analytically known.

2.5. Performance of the eigenvalue solver

Consider the following Poisson problem on [0, 1] × [0, 1] for the function $u(x, y)$

$$\nabla^2 u = f(x, y) \tag{18}$$

with Dirichlet boundary conditions $u = 0$ on all boundaries. When the problem is discretized using central differences on a uniform grid with grid size h and k, we get the following linear algebraic problem for $i = 1, \ldots, N-1; j = 1, \ldots, M-1$ with

$$a_N u_{i,j+1} + a_S u_{i,j-1} - u_{i,j} + a_W u_{i-1,j} + a_E u_{i+1,j} = b_{i,j} \tag{19}$$

with

$$a_N = a_S = \frac{h^2}{2(k^2 + h^2)}$$
$$a_W = a_E = \frac{k^2}{2(k^2 + h^2)}$$
$$b_{i,j} = \frac{f_{i,j} h^2 k^2}{(k^2 + h^2)}$$

This can be written as $\mathcal{E}\mathbf{u} = \mathbf{b}$ where $\mathbf{u}$ is a $(N-1) * (M-1)$ vector of unknown values of u at the internal gridpoints. If this linear system of equations is solved using the Jacobi method, we get an iteration matrix $\mathcal{F} = \mathcal{E} - \mathcal{I}$. It is well known that the eigenvalues σ of the ordinary eigenvalue problem

$$\mathcal{F}\mathbf{x} = \sigma \mathbf{x} \tag{20}$$

are

$$\sigma_{i,j} = \frac{h^2}{k^2 + h^2} \cos(i\pi/N) + \frac{k^2}{k^2 + h^2} \cos(j\pi/M) \tag{21}$$

for $i = 1, \ldots, N-1; j = 1, \ldots, M-1$.

The test problem consists of determining the largest eigenvalues of the matrix $\mathcal{F}$ numerically with the generalized eigenvalue solver. Equation (20) describes an ordinary eigenvalue problem and the matrix $\mathcal{B}$ in (7) is the identity matrix. Note that it is possible to determine the largest eigenvalues directly with the SIT code without any complex mapping. However, as already mentioned above, we want to compare the performance of the linear system solvers within the generalized eigenvalue solver. Also no special use is made of the symmetry of the system of equations. For the parameters in the complex mapping (9), we take $a = 1$, $b = 0$.

To determine the accuracy of the calculated eigenvalues and eigenvectors each eigensolution is backsubstituted into (20) and divided by the L_2 norm of the corresponding eigenvector. The L_∞ norm of this residue is then taken as a measure of the accuracy of the specific eigensolution. For

Table 1(a). First 4 eigenvalues of the iteration matrix $\mathscr{F}$ with $N = M = 50$

p	q	Computed	Relative accuracy
1	1	0.99810	8×10^{-9}
2	1	0.99526	2×10^{-6}
1	2	0.99526	2×10^{-6}
2	2	0.99241	3×10^{-3}

Table 1(b). Timing and memory requirements of the eigenvalue solver

			Direct solver			
N	M	d	Time	Time*	El. in $\mathscr{P}$*	Mem (Mw)
25	25	6.3×10^{2}	3	4.80×10^{-3}	—	2
50	50	2.5×10^{3}	33	1.32×10^{-2}	—	24
100	100	1.0×10^{4}	—	—	—	396•
			Iterative solver, no renumbering			
N	M	d	Time	Time*	El. in $\mathscr{P}$*	Mem (Mw)
25	25	6.3×10^{2}	1.4	2.24×10^{-3}	23.7	—
50	50	2.5×10^{3}	9	3.60×10^{-3}	28.7	1
100	100	1.0×10^{4}	50	5.00×10^{-3}	31.6	2
200	200	4.0×10^{4}	415	1.04×10^{-2}	33.3	8
			Iterative solver, renumbering			
N	M	d	Time	Time*	El. in $\mathscr{P}$*	Mem (Mw)
25	25	6.3×10^{2}	0.9	1.44×10^{-3}	20.7	—
50	50	2.5×10^{3}	4	1.60×10^{-3}	22.3	—
100	100	1.0×10^{4}	19	1.90×10^{-3}	24.8	2
200	200	4.0×10^{4}	90	2.25×10^{-3}	26.4	8
300	300	9.0×10^{4}	220	2.44×10^{-3}	27.0	18
400	400	1.6×10^{5}	416	2.60×10^{-3}	27.2	32

*Per unknown, • obtained by extrapolation (out of memory).

the described testproblem the number of computed eigenvalues n_f is set to 4. In all cases random vectors (with absolute magnitude smaller than 1) were used as an initial guess for the eigenvectors. The required accuracy is set to 10^{-5} for the first two eigenvectors. To accomplish this accuracy 10 ($l = 10$) iterations were needed during the filtering stage of the SIT, together with one reorientation cycle ($k = 1$). The same accuracy is also obtained with ($l = 5, k = 2$) and takes about the same amount of CPU time, since the number of linear systems of equations to solve is the same.

The eigenvalue results for the particular case $N = M = 50$ are shown in Table 1(a). The algorithm is able to determine multiple eigenvalues; the accuracy of the last computed eigensolution is slightly less than the others. The timing results for this test problem using both a direct solver, with a complete LU-decomposition computed with a bandsolver (the NAG routines F01LBF and F04LDF), and Bi-CGSTAB are shown in Table 1(b) for several resolutions $N = M$. For a complete LU-decomposition of a banded matrix (F01LBF was only called once) with bandwidth m_d and order d, the CPU time increases with $(m_a)^2d$. The preconditioning was also performed only once with $\epsilon_p = 10^{-2}$. Since $k = 1$, we are in fact comparing the time it takes to solve the l^*n_f linear systems with both methods.

The code for the iterative solver is only partially vectorized because the preconditioning technique is not very well suited for vectorization. However, even with a relative low grade of vectorization, a relatively good efficiency is obtained in terms of computational time and use of core memory. Time and memory requirements depend on the amount of fill-in that is allowed in the preconditioning matrix $\mathscr{P}$ and thereby on ϵ_p. Also shown in the table is the number of elements (per unknown) in $\mathscr{P}$. Here an optimum is sought between time needed for the preconditioning and the resulting iterative solution of the linear systems. The initial preconditioning is sufficient for the rest of the filtering iterations and the underlying linear solver Bi-CGSTAB takes only three iterations for the chosen value of ϵ_p. Clearly, the use of the iterative solver, even without renumbering, is a significant improvement over the direct solver. For large systems the gain in CPU time is already very good [$O(10)$ for 2500 unknowns]. For even larger systems [$O(10^4)$ or larger] direct methods are simply not possible on current computer systems because of memory limitations. The results using the renumbering technique are far better than those without renumbering. The three different cases are summarized in Fig. 2, where the CPU time per unknown (t/d) is plotted versus the number of degrees of freedom d. For the preconditioner with renumbering, t/d is proportional to a very small power of d.

3. RAYLEIGH–BÉNARD FLOWS IN LARGE ASPECT RATIO CONTAINERS

As a typical application of the code, we consider a liquid layer heated from below in a two-dimensional rectangular box of aspect ratio A (ratio of length to height). The temperature on top and bottom is constant, the sidewalls are perfectly isolated and all walls satisfy no-slip conditions. If the liquid is motionless, the only solution is the conduction solution for which the temperature is a linear function of z. If the horizontal and vertical velocities are u and w, respectively, the governing dimensionless equations, with the usual scaling (see e.g. [2]), in streamfunction ψ (with $u = \partial\psi/\partial z$ and $w = -\partial\psi/\partial x$)-vorticity ω ($\omega = \partial w/\partial x - \partial u/\partial z$) formulation are

$$\mathrm{Pr}^{-1}\left(\frac{\partial \omega}{\partial t} + u\frac{\partial \omega}{\partial x} + w\frac{\partial \omega}{\partial z}\right) = \nabla^2\omega + \mathrm{Ra}\frac{\partial T}{\partial x} \tag{22}$$

$$\omega = -\nabla^2\psi \tag{23}$$

$$\frac{\partial T}{\partial t} + u\frac{\partial T}{\partial x} + w\frac{\partial T}{\partial z} = \nabla^2 T + w \tag{24}$$

with boundary conditions

$$x = 0, \mathrm{A}: \ \psi = \frac{\partial \psi}{\partial x} = \frac{\partial T}{\partial x} = 0 \tag{25}$$

$$z = 0, 1: \ \psi = \frac{\partial \psi}{\partial z} = T = 0 \tag{26}$$

where T is the temperature deviation from the condition solution $\bar{T} = 1 - z$, Pr is the Prandtl number and Ra the Rayleigh number.

The governing stationary equations and boundary conditions are discretized using a control volume method as in [15] on a grid for $i = 0, \ldots, N$; $j = 0, \ldots, M$. By using the mapping

$$y = \tfrac{1}{2}(1 + \tanh(q(\bar{y} - \tfrac{1}{2}))/\tanh(q/2)) \tag{27}$$

a non-equidistant grid in y is obtained from an equidistant one in $\bar{y}$. These non-equidistant grids were employed in both the x- and z-direction; the stretching parameters q are indicated below as q_x and q_z. Discretization gives a nonlinear system of algebraic equations of the form

$$\mathbf{F}_{N,M}(\mathbf{u}, \mathbf{p}) = 0 \tag{28}$$

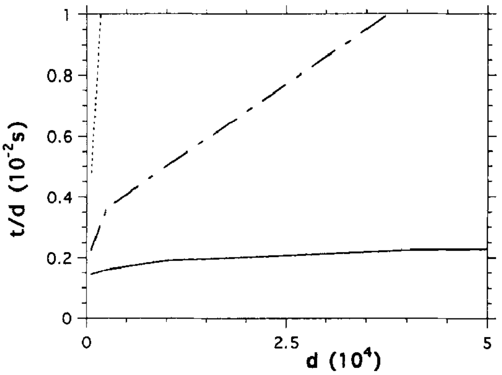

Fig. 2. CPU time (CRAY-YMP) per unknown (t/d) as a function of the degrees of freedom d for the direct solver (·····) and the iterative solver with (——) and without (– – – – – –) renumbering.

Table 2. Values of Ra_1 and Ra_2, the first two primary bifurcation points at $\mathrm{A} = 10$ for several resolutions and (non-equidistant) grids. A value $q_x = 1$ indicates an equidistant grid, a value of 3 indicates a stretching of the grid to create more points near the sidewalls, according to the mapping (27)

q_x	q_z	N	M	Ra_1	Ra_2
3	3	128	16	1787.3	1788.8
3	3	256	16	1745.4	1748.2
3	3	512	16	1734.8	1737.7
3	3	∞	16	1731.3	1734.1
1	3	128	16	1757.3	1759.2
1	3	256	16	1737.8	1740.4
1	3	256	32	1735.7	1738.9
1	3	512	16	1732.9	1735.7
1	3	∞	16	1731.3	1734.1
1	1	256	16	1744.4	1747.7

where $\mathbf{u}$ is the $d = 3(N + 1)(M + 1)$ dimensional solution vector consisting of the unknowns at the gridpoints. The 3-dimensional vector $\mathbf{p}$ consists of the values of the parameters (Ra, Pr, A). The linear stability of a particular solution of (10) is determined by solving the generalized eigenvalue problem of the form (7) where $\mathscr{A}$ is the (in this case symmetric) Jacobian of $\mathbf{F}_{N,M}$ and $\mathscr{B}$ is a singular diagonal matrix. In this case, infinite eigenvalues occur, but components of the corresponding eigenvectors are first filtered out, by using inverse iteration [15, 24]; this takes little extra computational time. For the parameters in the mapping (9), we again take standard values $a = 1$, $b = 0$.

The results are divided over three sections. In the first, we consider the influence of (distant) sidewalls of the container on the critical Rayleigh number for the onset of convection. The critical Rayleigh number is here determined by the value of the Rayleigh number at the first primary bifurcation point. In the second section, we focus on $\mathrm{A} = 10$ and study the stable (and unstable) steady solutions and the associated convective heat transfer, through the Nusselt number Nu, for different Prandtl numbers. These results can be compared with experimental work. In the third section, we consider $\mathrm{A} = 20$ and investigate whether any 'pattern selection', in the form of intervals in Ra where unique stable steady states exist, is found directly from the bifurcation diagrams.

3.1. The onset of convection

The primary bifurcation points could be easily detected although they are closely spaced. All primary bifurcations are of pitchfork type, because of the $Z_2 \times Z_2$ symmetry of the problem. Due to this symmetry, some testfunctions τ_{pq} will not detect certain bifurcation points for specific choices of p and q. Since we are only interested in large aspect ratio containers, the only relevant symmetry is reflection through the mid-axis of the container. If one chooses the value of p corresponding to a streamfunction value at a location where the eigenvector at the singularity is such that the streamfunction (at that particular location) is zero, then the testfunction will not detect the singularity. This is due to the fact that the eigenvector is supposed to be normalized with its p^{th} component being unity [21]. For example, if the eigenvector consists of an even cell pattern, a value of p corresponding to the streamfunction on the mid-axis will not detect the singularity associated with this eigenvector. The solution is simple, we choose two values of p corresponding to both temperature and streamfunction at the same location. When the streamfunction is an odd function, the temperature will be even and vice versa. One of the testfunctions will detect a singularity. Note that singularities are always noticed through the eigenvalue solver.

In Table 2, the values of the first and the second primary bifurcation point for $\mathrm{A} = 10$ are shown for several grids. The extrapolated value (it is observed that quadratic convergence is attained for the finer grids) is 1731.3. Note that an equidistant grid in x gives better results than a grid with a stretching $q_x = 3$ (more points near the sidewalls). For an infinite layer the critical value is 1708. The sidewalls have a stabilizing influence, as is well known from similar problem [7, 9] and for finite aspect ratio containers the critical value Ra_c is therefore larger. The value found here is slightly below the range of values reported experimentally in [32]. The experimental values are likely to be larger because the roll-axes may not be perfectly aligned to the sidewalls.

3.2. *Finite amplitude convection, $A = 10$*

We first show timing results (Table 3) for the computation of one regular point and its linear stability on a particular branch in the bifurcation diagram. The number of Newton iterations is for each resolution equal to 3 (the step size was chosen accordingly), and the number of dominant eigenvalues $n_f = 4$. The values in Table 3 are typical for the bulk of the computations, where good approximations to the eigensolutions are available. In this case the SIT converges fast, usually within 2 reorientations. Renumbering was used in computing the preconditioning matrix, with a straightforward extension of the procedure described above to more than one unknown per point. Although it speeds up the code somewhat this probably can be improved; work on this is in progress.

The data in Table 3 indicate that the eigenvalue solver is much more expensive than the determination of the steady state itself. In the computations below, we therefore did not determine the stability at each point but at every 5th point on a branch. The total computational time does not increase linearly with the number of unknowns but somewhat faster. Table 3 also illustrates that a good choice of ϵ_p is worthwhile.

If we pose as criterion that the numerical value of the first two bifurcation points may only differ 1% from the extrapolated value, then a 256 × 16 grid (both equidistant and stretched) is adequate (see Table 2) to determine a correct bifurcation structure. These bifurcation structures were computed for three different Prandtl numbers (Pr = 0.7, 5.5 and 1000). In these computations, the grid was stretched in x-direction ($q_x = 3$), because it was expected that changes in the number of cells would occur along one of the sidewalls, requiring extra resolution. The bifurcation structure for Pr = 5.5 is plotted in the weakly nonlinear regime in Fig. 3(a). On the vertical axis, the vertical velocity at the gridpoint (3, 12)—near the upper left corner—is plotted. The slightly supercritical patterns near the primary bifurcation points are shown in Figs 3(b–d). In the nonlinear regime, larger Ra, the bifurcation structure is shown in Fig. 4(a). Patterns at labelled points in Fig. 4(a) are presented in Figs 4(b–f). At the first primary bifurcation point (Ra = 1745.4) the motionless solution becomes unstable to the 10-cell pattern [Fig. 3(b)] which stabilizes in a supercritical pitchfork bifurcation. This 10-cell pattern remains stable up to large Rayleigh number [Fig. 4(a, b)]. At the second bifurcation, a 9-cell solution branches off [Fig. 3(c)]. This solution remains unstable up to the secondary bifurcation point at Ra = 1835, but is stable for larger Ra [Fig. 4(d)]. The mixed mode branch appearing at this secondary bifurcation point, consists of an asymmetric pattern where a new cell develops near the left wall of the container [Fig. 4(c)]. This pattern remains unstable [Fig. 4(a)]. At the third primary bifurcation point, the motionless solution becomes unstable to an 11-cell pattern [Fig. 3(d)]. Although there is a secondary bifurcation point at Ra = 2043 at which asymmetric solutions branch off [Fig. 4(e)], the solution along this branch remains unstable over the computational domain considered [Fig. 4(f)].

We next compare these results with experiments, in particular the work by Dubois and Bergé [33] and Walden *et al.* [32]. In [33], a container with A = 10 is used and the Prandtl number is about 1000. Velocity profiles (vertical as well as horizontal) are measured and presented in [33] for several values of supercritical Rayleigh numbers. The 10-cell solution branches are almost identical for Pr = 1000 and Pr = 5.5. Both 10-cell patterns are stable up to large Ra. This is consistent with that found in [33] (in these experiments both symmetry related 10-cell patterns could be induced through

Table 3. CPU-time (CRAY-YMP) and memory usage for Rayleigh–Bénard problem for one typical step on a branch, excluding/including the computation of the linear stability through $n_f = 4$ eigensolutions. The number of Newton iterations was in each case $m = 3$, the number of filtering steps was $l = 5$ and the number of reorientations $k = 2$

N	M	Eigenvalues	ϵ_p	CPU time (s)	Mem (Mw)
256	16	no	5.0×10^{-5}	58	6.6
256	16	yes	5.0×10^{-5}	141	6.6
256	16	yes	5.0×10^{-4}	249	6.6
256	16	yes	5.0×10^{-6}	147	6.6
128	16	no	5.0×10^{-5}	34	3.4
128	16	yes	5.0×10^{-5}	64	3.4
512	16	no	5.0×10^{-5}	101	13.1
512	16	yes	5.0×10^{-5}	308	13.1

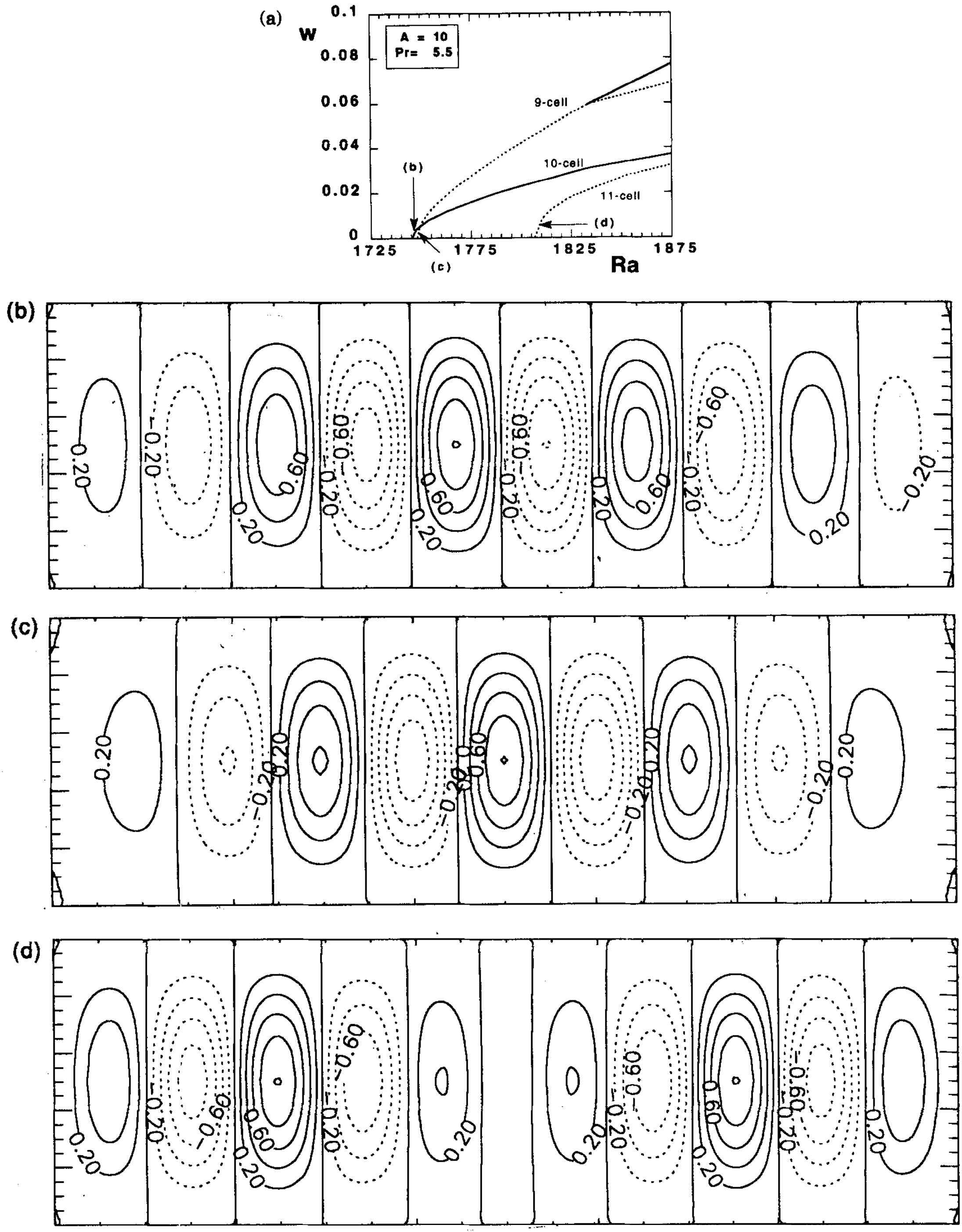

Fig. 3. (a) Bifurcation picture for A = 10 and Pr = 5.5 in the weakly nonlinear regime. Solid (····) curves indicate (un)stable states. Selected patterns of the streamfunction at labelled points in the figure are shown in the panels (b–d). Contour levels are labelled in the plots.

different initial conditions) where the 10-cell patterns could be realized for Ra up to $10\mathrm{Ra}_c$. In Fig. 5(a), we show the vertical velocity $w(x, z = 0.5)$ over the interval $x\epsilon[3.5, 6.5]$ for two values of supercritical Rayleigh numbers close to the values of Fig. 3 in [33]. Using the experimental values of thermal diffusivity and layer height, we calculate for $\mathrm{Ra} = 6.7 \times \mathrm{Ra}_c$ a maximum velocity of 419 $\mu\mathrm{m\,s}^{-1}$. This value is close to the experimental value shown in Fig. 3(b) of [33]. Also, the change in the shape of the velocity profile, as observed in experiments, with increasing Ra is clearly seen. Just as in [33], we decomposed the vertical velocity field at $z = 0.5$ into two Fourier components, one (w_1) with wavelength 2.0 and one (w_3) higher harmonic of wavelength 2/3. The maximum amplitudes of these components are shown in Fig. 5(b) as a function of Ra. The behavior is in excellent correspondence with the measurements in [33], the first harmonic has a square root dependence on Ra, whereas the higher harmonic behaves like $\mathrm{Ra}^{3/2}$. Best fits, when the dimensional values are considered, are

$$w_1 = 154\epsilon^{0.5}\,\mu\mathrm{m\,s^{-1}} \quad w_3 = 4.1\epsilon^{1.5}\,\mu\mathrm{m\,s^{-1}} \tag{29}$$

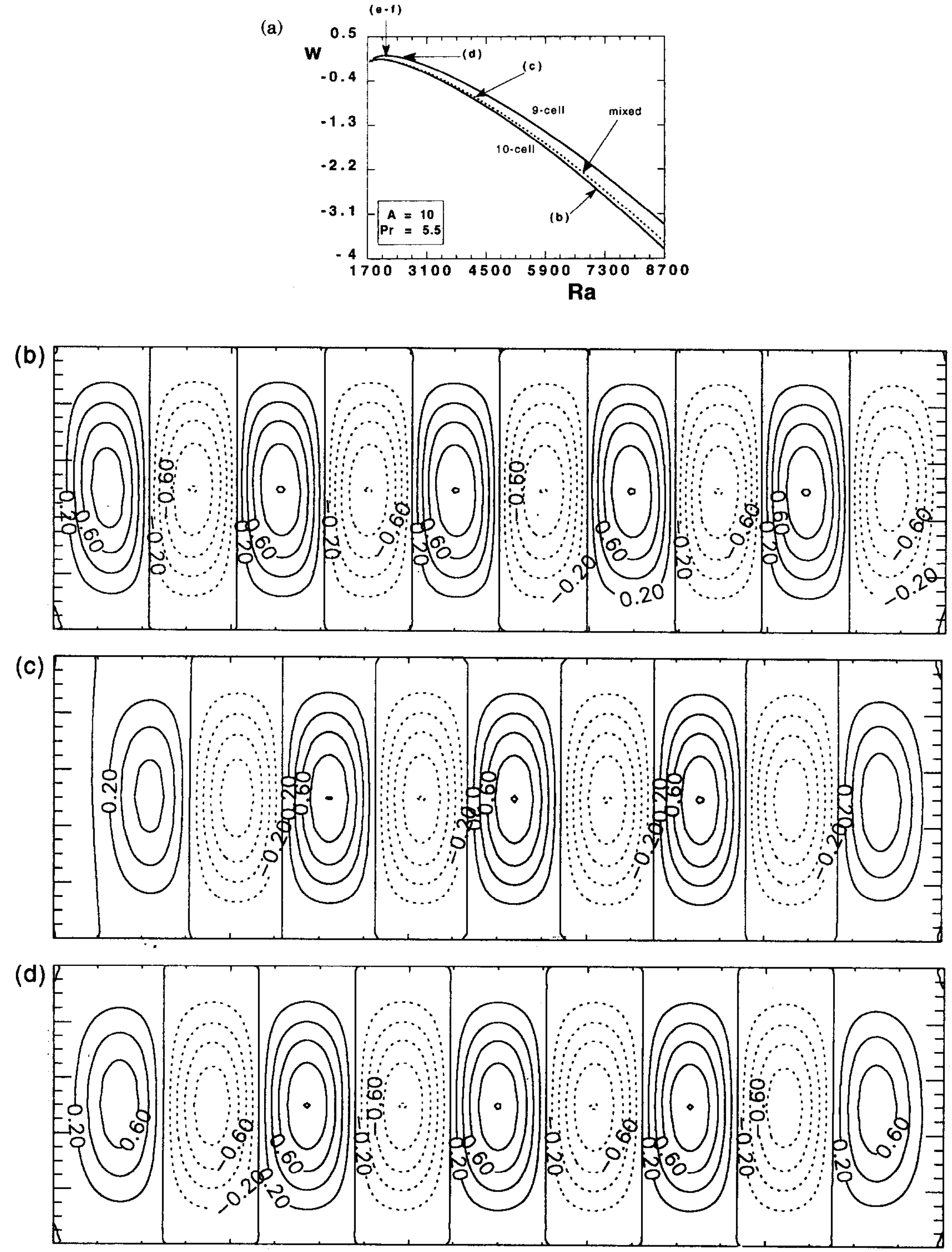

Fig. 4(a-d)—*Caption opposite.*

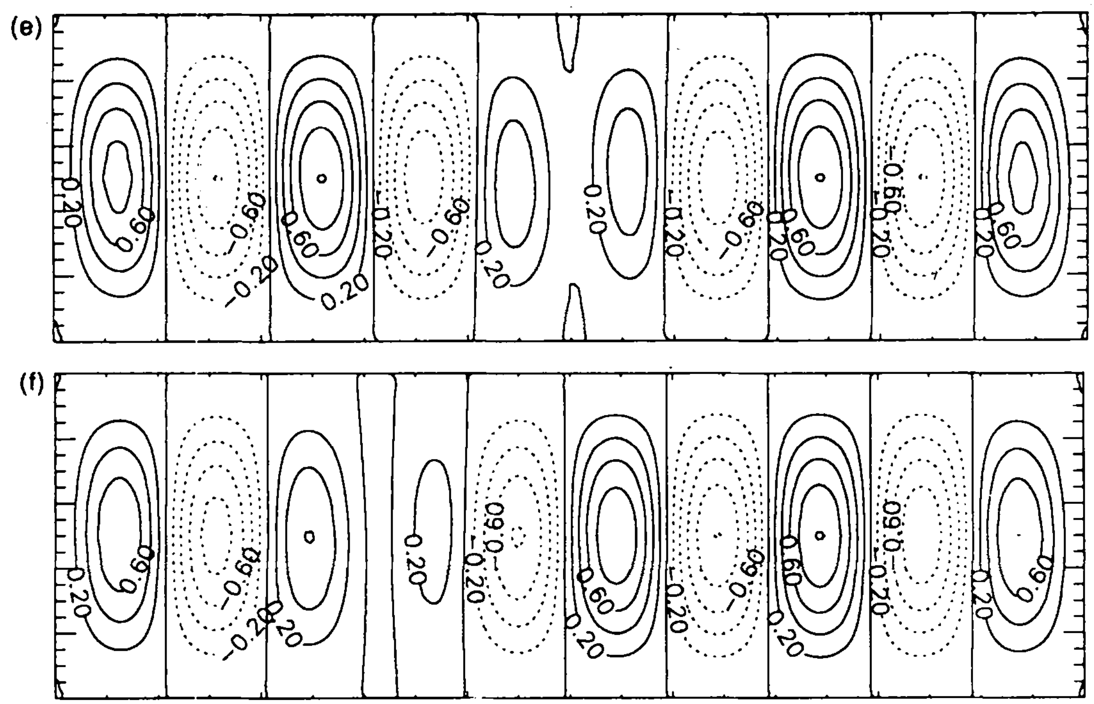

Fig. 4(e-f).

Fig. 4. (a) Bifurcation picture for A = 10 and Pr = 5.5 in the nonlinear regime. Again, selected patterns at labelled points in the figure are shown in the panels (b–f).

where $\epsilon = (\mathrm{Ra} - \mathrm{Ra}_c)/\mathrm{Ra}_c$. This is quite close to the relations found in the experiments. The dependence of Nu on Ra along the stable 10-cell branch is for the three Prandtl numbers shown in Fig. 6(a). There is hardly any difference between the heat transfer associated with the slightly different convection patterns. Clearly Nu is linear with Ra close to onset, but levels off at larger Ra, in correspondence with experiments. Walden *et al.* [32] consider a container with an aspect ratio 10.61 × 5.32 (their container A) and measure the Nusselt number Nu as a function of increasing Rayleigh number. They observe several steady (nearly) 2-dimensional patterns each having its different curve Nu–Ra (see their Fig. 3). The Nusselt number of the linearly stable patterns are shown in Fig. 6(b) in the same way as Fig. 3 of [32]. Because of the internal symmetry, symmetry related cells have the same Nusselt number. The 9-cell and 10-cell solutions have nearly the same Nu number and the values are within the range of the experimental results, although they are slightly larger. Again, the non-perfect alignment of the roll-axes in the experiments may be responsible for the lower experimental values of Nu.

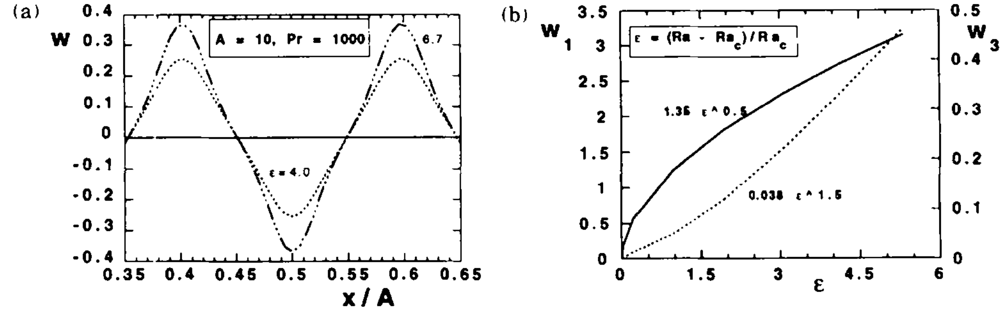

Fig. 5. (a) Vertical velocity at the midheight of the layer ($z = 0.5$) as a function of x/A for Pr = 1000 and the two indicated values of $\epsilon = (\mathrm{Ra} - \mathrm{Ra}_c)/\mathrm{Ra}_c$. (b) Fourier components of the vertical velocity along the 10-cell branch as a function of ϵ for the same values of A and Pr as in (a).

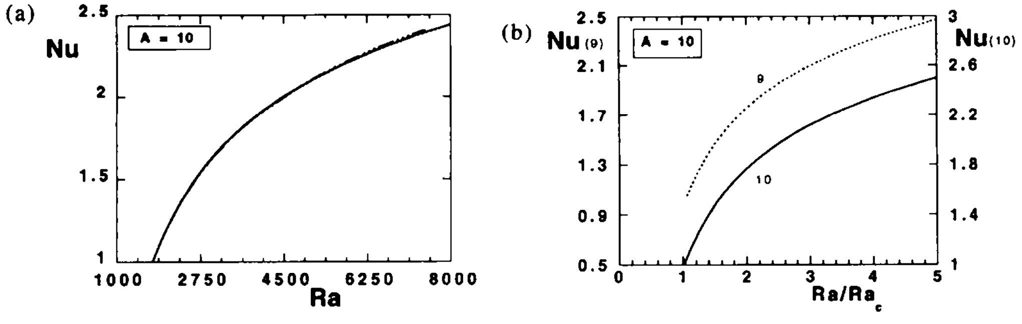

Fig. 6. The Nusselt number Nu as a function of Ra. (a) For the three different Prandtl numbers along the 10-cell branch. (b) For Pr = 5.5 along the 9- and 10-cell branches.

3.3. *The cell pattern selection problem, A = 20*

In this last section, we compute bifurcation diagrams at a fairly large aspect ratio (A = 20) using a stretched 512 × 16 grid. First we compare these qualitatively with the asymptotic results of weakly nonlinear theory [7] for a container with distant sidewalls and stress free horizontal boundaries. In fact, one can view these results as extensions of the weakly nonlinear theory both with respect to the type of horizontal boundaries and into the full nonlinear regime. Second, we test the hypothesis that pattern selection is induced through saddle node bifurcations far from onset. In support of this hypothesis we would like to find several intervals in Ra where unique stable steady states exist.

The bifurcation picture for Pr = 7, A = 20 is presented in Fig. 7(a) for Rayleigh numbers close to onset (the weakly nonlinear regime). The corresponding steady patterns of slightly supercritical values of Ra are given in the Figs 7(b–e). The first two solutions bifurcating at Ra = 1729.2 [the 20-cell solution in Fig. 7(b) and the 19-cell solution in Fig. 7(c)] have constant phase (of the complex amplitude function which modulates the size of the individual rolls) and only the first (the 20-cell solution) is stable. For the other two [Figs 7(d) and (e)], the amplitude function goes to zero at locations within the container (these type of solutions were also found by Daniels [7]) and both solutions are unstable. Hence, the picture of steady states close to onset is qualitatively similar to that of stress free horizontal walls. For larger values of Ra the bifurcation structure in Fig. 7(a) shows that the secondary bifurcation point on the 19-cell branch (at which the solution stabilizes) already occurs close to the onset, at Ra = 1733.5.

Both the 20-cell and 19-cell remain stable over the rest of the computational domain [Fig. 8(a)]. For the infinite layer, the critical wave number is 3.117; this corresponds to 19.84 cells at onset. Figure 8(b) shows that 20 cells fit in 'comfortably' if the wavelength of the cells near the boundary is slightly larger than the critical value whereas the wavelength over the remainder of the layer is slightly smaller. Also 19 cells fit in 'well' [Fig. 8(c)], with the wavelength over the layer being slightly larger than the critical wavelength. At the secondary bifurcation point, Ra = 1733.5, new unstable solutions appear. This branch is labelled 'mixed' in both Figs 7(a) and 8(a). These are probably the 'phase winding' solutions found in weakly nonlinear theory.

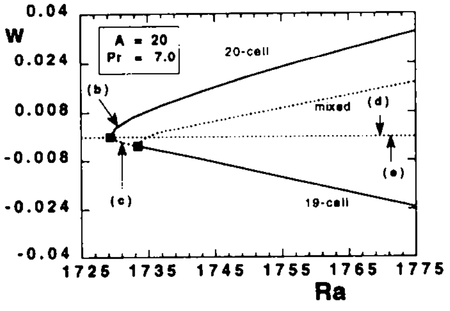

Fig. 7(a)—*Caption opposite.*

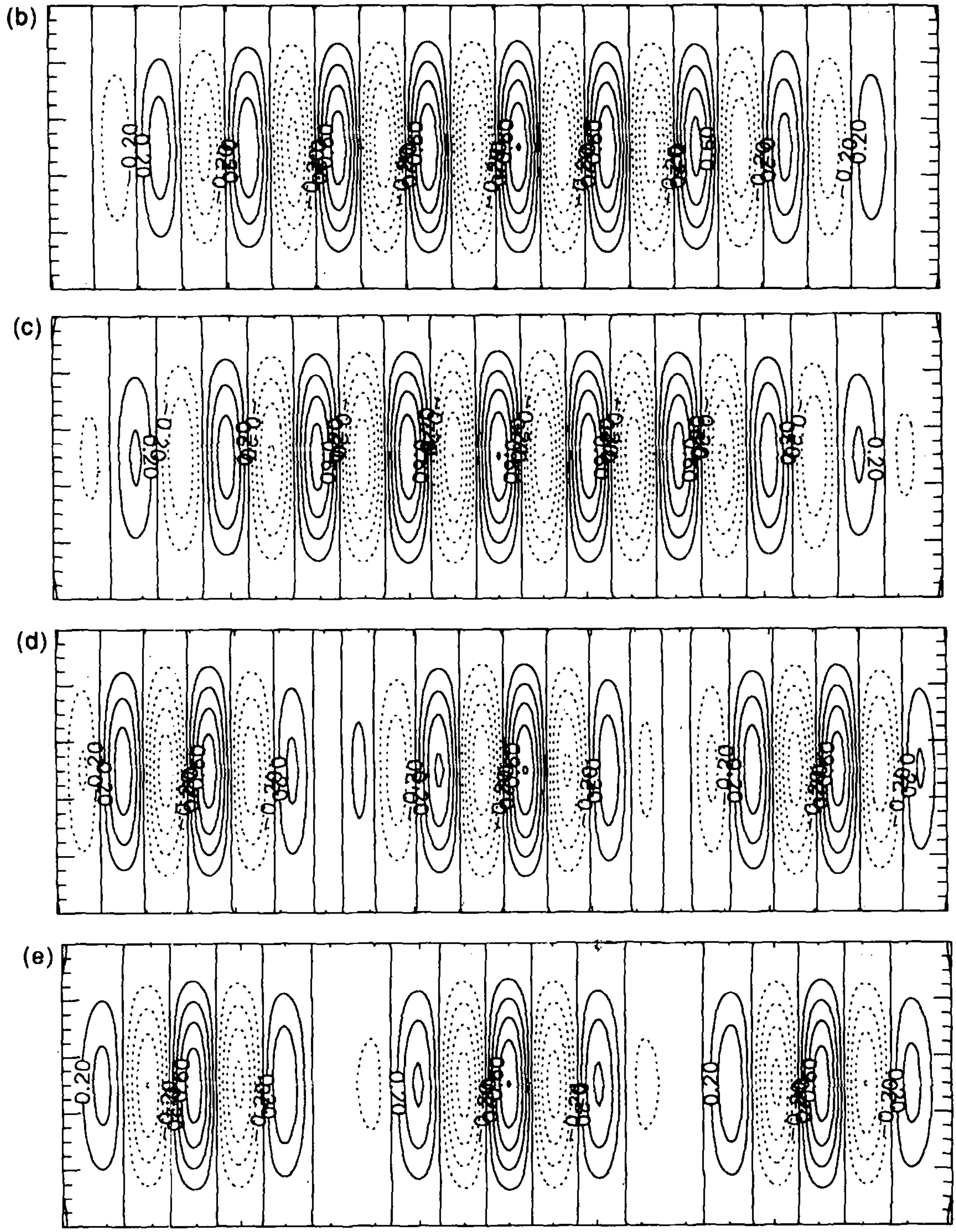

Fig. 7(b-e).

Fig. 7. (a) Bifurcation picture for A = 20 and Pr = 7.0 in the weakly nonlinear regime. Bifurcation points are shown as squares. Selected patterns at labelled points in the figure are shown in the panels (b – e).

Pictures of two solutions on this branch are presented in the Figs 8(d) and (e). With respect to our hypothesis, we observe that in Fig. 8(a), no saddle node bifurcations appear far from onset. Hence, the bifurcation picture clearly shows that for this value of Pr there is no discontinuous change between unique states. Instead more than one stable state is possible in large aspect ratio containers.

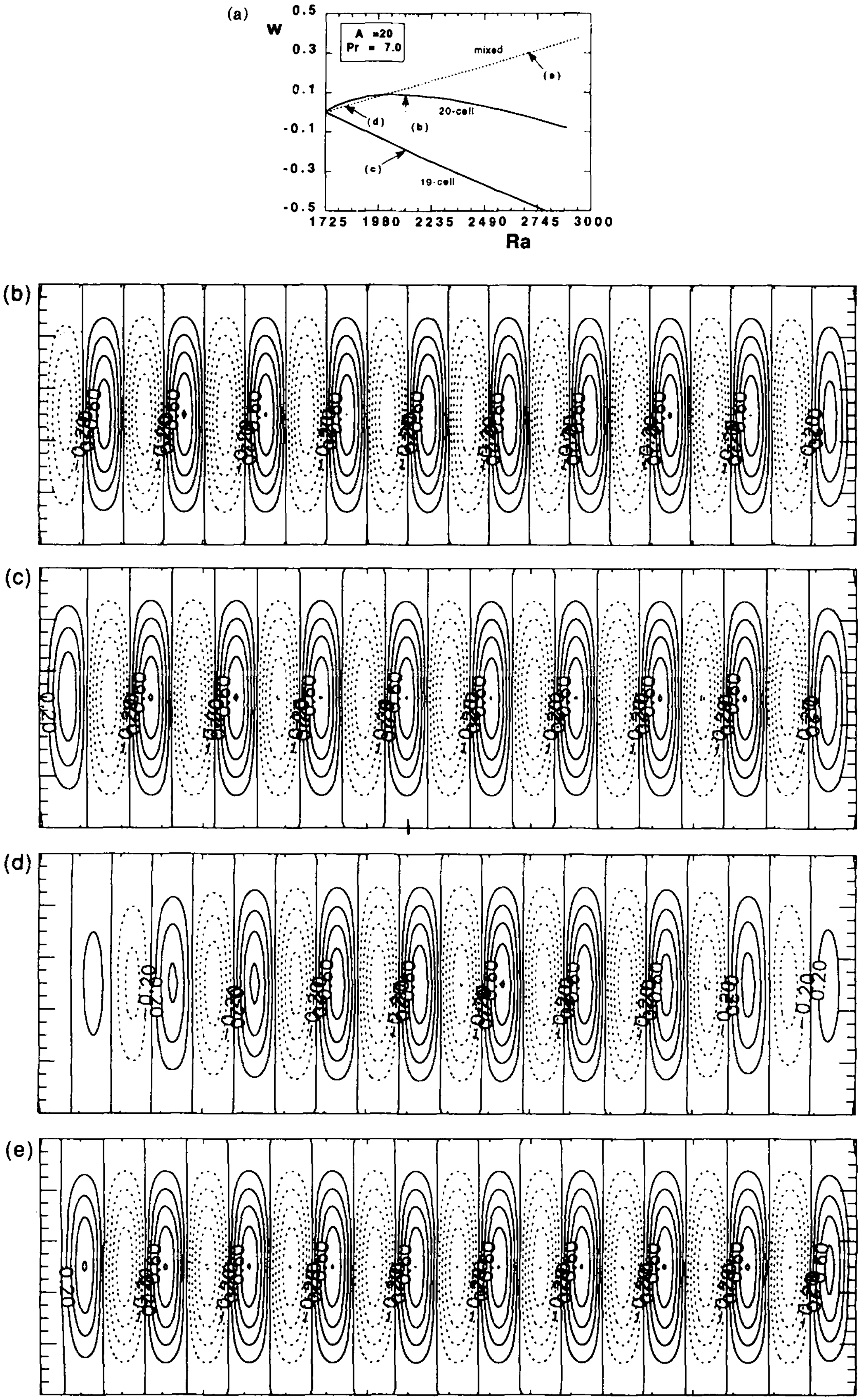

Fig. 8. (a) Bifurcation picture for A = 20 and Pr = 7.0 in the nonlinear regime. Again, selected patterns at labelled points in the figure are shown in the panels (b–e).

4. CONCLUSIONS

The numerical technique which is presented in this paper is useful to investigate the structure of solutions of large systems of nonlinear algebraic equations in parameter space. These large algebraic systems, for example, arise through discretization of systems of steady partial differential equations describing, for example, confined flows. In this way, it becomes possible to effectively analyse non-parallel flows and their linear stability in parameter space. The advantage of this type of analysis is that unstable solution branches can be found, which is hardly possible (only with special tricks) using time-integration techniques. Although these unstable solutions have no physical relevance they are often necessary to construct the right transitions between stable patterns.

Results of a test problem have been presented to show that the eigenvalue solver performs well and that (for the particular problem) the computational effort per unknown hardly increases with the number of degrees of freedom d, if d is large. This efficiency is obtained through the type of preconditioner (with renumbering) used; the major part of the computational work in the eigenvalue solver is to solve linear systems. Although Bi-CGSTAB is used here, many other standard methods to solve the preconditioned system would give nearly the same results. A preconditioned conjugate gradient like method combined with the SIT therefore enables one to treat large generalized eigenvalue problems in this way.

Application of the code to the classical Rayleigh–Bènard problem in large aspect ratio containers shows that indeed high resolution is necessary to compute a sufficiently accurate bifurcation diagram (Table 2). The aim of the present paper is to demonstrate that the present code can do this job. Thereby it opens the gate to compute these type of bifurcation diagrams for flows for which, up to now, only theoretical information in the weakly nonlinear regime through asymptotic techniques or in the full nonlinear regime using time-integration could be obtained. The results for $A = 10$ were shown to compare well with measurements. The results for even larger aspect ratio $A = 20$ compare (near onset) qualitatively well with those of weakly nonlinear theory. Our hypothesis about the selection induced by saddle node bifurcations fails in the particular case investigated. The bifurcation results as presented therefore give no indication why the wavelength of the experimentally observed pattern increases with increasing Rayleigh number. A detailed answer to the question of the pattern selection problem as found in experiments and pointed out in the introduction requires more (computational) effort and is outside the scope of this paper.

Finally, the numerical method is not without its drawbacks and can be improved in several ways. First, the test functions used to detect simple bifurcation points are not robust and have to be reconsidered for each application. More robust test functions are currently developed. Second, the linear solver works very well for linear systems which are diagonally dominant but for systems which are not, the amount of fill-in in the preconditioning matrix and thereby the required CPU time and use of memory increases. We are currently experimenting with the parameters in the complex mapping (9) to improve this (at least in the eigenvalue solver). When the CPU time becomes excessively large or the amount of fill-in too large, then time integration remains as the only option to determine any dynamics of the system.

Although there may remain technical problems from application to application, the code enables one to deal with bifurcation problems defined by large systems [$d = O(10^5)$] of non-linear algebraic equations in an efficient way. The code has recently been applied to a pattern selection problem in 3D Marangoni convection [34]. It has the potential to study a new range of problems, for instance the stability of the large scale ocean circulation, with continuation techniques in the near future.

Acknowledgements—All computations were performed on the CRAY Y-MP at the Academic Computing Centre (SARA), Amsterdam, the Netherlands within the project SC212. Use of these computing facilities was sponsored by the Stichting Nationale Supercomputerfaciliteiten (National Computing Facilities Foundation, NCF) with financial support from the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (Netherlands Organization for Scientific Research, NWO).

REFERENCES

1. K. A. Cliffe and T. Mullin, A numerical and experimental study of anomalous modes in the Taylor experiment. *J. Fluid Mech.* **153**, 243–258 (1985).
2. E. L. Koschmieder, *Bénard Cells and Taylor Vortices.* Cambridge Univ. Press (1993).

3. A. V. Getling, Formation of spatial structures in Rayleigh–Bénard convection. *Sov. Phys. Usp.* **34,** 737–776 (1991).
4. I. C. Catton, Wavenumber selection in Bénard convection. *Trans. ASME* **110,** 1154–1163 (1988).
5. M. C. Cross, P. C. Hohenberg and S. A. Safran, Wavenumber selection in Rayleigh–Bénard convection: a numerical study. *Physica D* **5,** 75–82 (1982).
6. M. C. Cross, P. G. Daniels, P. C. Hohenberg and E. D. Siggia, Phase winding solutions in a finite container above the convective threshold. *J. Fluid Mech.* **127,** 155–183 (1983).
7. P. G. Daniels, The effect of distant side-walls on the evolution and stability of finite-amplitude Rayleigh–Bénard convection. *Proc. R. Soc. Lond.* **378,** 539–566 (1981).
8. P. Metzener, The effect of rigid sidewalls on nonlinear two-dimensional Bénard convection. *Phys. Fluids* **29,** 1373–1377 (1986).
9. A. I. v.d. Vooren and H. A. Dijkstra, A finite element stability analysis of the Marangoni problem in a two-dimensional container with rigid sidewalls. *Comp. Fluids* **17,** 467–485 (1989).
10. W. Arter, A. Bernoff and A. C. Newell, Wavenumber selection of convection rolls in a box. *Phys. Fluids* **30,** 3840–3842 (1987).
11. K. A. Cliffe, Numerical calculations of two-cell and single-cell Taylor flows. *J. Fluid Mech.* **135,** 219–233 (1983).
12. D. S. Riley and K. H. Winters, Modal exchange mechanisms in Lapwood convection. *J. Fluid Mech.* **204,** 325–358 (1989).
13. N. Tsitverblit and E. Kit, The multiplicity of steady flows in confined double-diffusive convection with lateral heating. *Phys. Fluids A* **5,** 1062–1064 (1993).
14. E. J. Kranenborg and H. Dijkstra, The structure of (linearly) stable double diffusive flow patterns in a laterally heated stratified liquid. *Phys. Fluids* (1995), in press.
15. H. A. Dijkstra, On the structure of cellular solutions in Rayleigh–Bénard–Marangoni flows in small-aspect-ratio containers. *J. Fluid Mech.* **243,** 73–102 (1992).
16. A. v.d. Ploeg, Preconditioning techniques for non-symmetric matrices with application to temperature calculations of cooled concrete. *Int. J. Num. Methods Eng.* **35(6),** 1311–1328 (1992).
17. E. J. Doedel, AUTO: a program for the automatic bifurcation analysis of autonomous systems. In *Proc. 10th Manitoba Conf. on Numerical Math. and Comput.* **30,** 265–274 (1980).
18. A. I. Khibnik, LINLBF: a program for continuation and bifurcation analysis of equilibria up to codimension three. In *Continuation and Bifurcations: Numerical Techniques and Applications* (Edited by D. Rosse *et al.*), pp. 283–296. Kluwer (1990).
19. M. Kubiček and M. Marek, *Computational Methods in Bifurcation Theory and Dissipative Structures.* Springer (1983).
20. H. B. Keller, Numerical solution of bifurcation and nonlinear eigenvalue problems. In *Applications of Bifurcation Theory* (Edited by P. H. Rabinowitz). Academic (1977).
21. R. Seydel, Numerical computation of branch points in nonlinear equations. *Num. Math.* **33,** 339–352 (1979).
22. G. Golub and C. F. Van Loan, *Matrix Computations.* The Johns Hopkins Univ. Press (1983).
23. I. Goldhirsch, S. A. Orszag and B. K. Maulik, An efficient method for computing leading eigenvalues and eigenvectors of large asymmetric matrices. *J. Sci. Comput.* **2,** 33–58 (1987).
24. K. N. Christodoulou and L. E. Scriven, Finding leading modes of a viscous free surface flow: an asymmetric generalized eigenproblem. *J. Sci. Comput.* **3,** 355–406 (1988).
25. Y. Saad, Variations on Arnoldi's method for computing eigenelements of large unsymmetric matrices. *Lin. Alg. Appl.* **34,** 269–295 (1980).
26. W. J. Steward and A. Jennings, A simultaneous iteration algorithm for real matrices. *ACM Trans. Math. Software* **7,** 184–198 (1981).
27. H. A. v.d. Vorst, BI-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.* **13(2),** 631–644 (1992).
28. D. Axelsson, Solution of linear systems of equations: iterative methods. In *Sparse Matrix Techniques: Copenhagen* (Edited by V. A. Barker), pp. 1–51. Springer, Berlin (1977).
29. A. v.d. Sluis and H. A. v.d. Vorst, The rate of convergence of conjugate gradients. *Num. Math.* **48,** 543–560 (1986).
30. Y. Saad, Krylov subspace methods on supercomputers. *SIAM J. Sci. Stat. Comput.* **10(6),** 200–1232 (1989)
31. A. v.d. Ploeg, E. F. F. Botta and F. W. Wubs, Grid-independent convergence based on preconditioning techniques. Report W-9310, Department of Mathematics, Groningen (1993).
32. R. W. Walden, P. Kolonder, A. Passner and C. M. Surko, Heat transport by parallel-roll convection in a rectangular container. *J. Fluid Mech.* **185,** 205–234 (1987).
33. M. Dubois and P. Bergé, Experimental study of the velocity field in Rayleigh–Bénard convection. *J. Fluid Mech.* **85,** 641–653 (1978).
34. H. A. Dijkstra, Surface tension driven celullar patterns in three-dimensional boxes. Linear stability. *Microgravity Sci. Technol.* VII, 4, 307–312 (1995).