# A Mixed Finite Difference/Galerkin Method for Three-Dimensional Rayleigh–Bénard Convection

JEFFREY C. BUELL

*Computational Fluid Dynamics Branch,*
*NASA Ames Research Center, Moffett Field, California, 94035*

Received September 19, 1986; revised April 30, 1987

We consider here a fast, accurate numerical method applicable to systems of nonlinear conservation equations with diffusion that have solutions which are periodic in two of the three space dimensions (or periodic in one dimension for two-dimensional problems). As a special case, the method is implemented for Rayleigh–Bénard convection between two rigid parallel plates in the parameter region where steady three-dimensional convection is known to be stable. High-order streamfunctions are used to reduce the system of five partial differential equations for the five primitive variables to a system of three differential equations. The new dependent variables are then expanded in Fourier series in the periodic (horizontal) directions, and the Galerkin method applied. This leaves a system of ordinary differential equations in the remaining (vertical) coordinate. These are solved by fourth-order-accurate operator compact implicit finite differencing (which is shown, for one case, to be more efficient by a factor of about five over second-order-accurate centered differencing). The calculations to evaluate the nonlinear terms are all performed in Fourier space, thus avoiding aliasing and other problems associated with collocation and Fourier transforms. Numerical tests that verify the expected convergence rates and the absolute accuracy of the method are presented.
© 1988 Academic Press, Inc.

## 1. INTRODUCTION

The two most popular and successful methods for solving systems of partial differential equations (PDEs) arising from convection-diffusion problems are finite differencing and spectral schemes. The latter are advantageous in general because relatively few degrees of freedom are needed to approximate a given function (especially smooth functions), which reduces both computer storage and execution time. Also, they are relatively easy to implement if the boundary conditions allow the use of trigonometric functions. For these reasons spectral methods have always been widely used, especially for natural convection problems. On the other hand, finite difference methods are easier to formulate for most kinds of boundary conditions and they yield banded, rather than full, matrices. Furthermore, recent advances in approximation theory have led to new finite difference schemes that are more attractive than the classical ones.

Spectral methods divide into three important classes: Galerkin, transform, and

0021-9991/88 $3.00

pseudospectral methods. (We note that in the literature there are a variety of definitions for these names; the following are most convenient here.) In the first class, which are sometimes called "traditional" Galerkin methods, the calculations are all performed in spectral space and are characterized by "interaction coefficients" and convolution summations. These arise from the nonlinear and variable-coefficient terms in the original differential system. In transform methods, the space derivatives are evaluated in spectral space and then transformed together with the dependent variables to a set of "collocation points" in physical space. The terms which would have led to interaction coefficients in the Galerkin method are evaluated in physical space and transformed back to spectral space, where the differential system is required to be satisfied. Pseudospectral (or collocation) methods differ from transform methods in that the differential system is evaluated in physical space. For many problems this is more convenient, but aliasing errors are introduced. These errors do not occur with Galerkin methods, and can be avoided with transform methods by using a sufficient number of collocation points (usually 50% more than the minimum number).

The main advantage of transform and pseudospectral methods is the reduction in the execution time due to the replacement of multidimensional transforms with sequences of one-dimensional transforms. This reduction is especially significant for three-dimensional (3D) expansions with a large number of modes, less so for two-dimensional (2D) expansions, and nonexistent in one dimension. These two methods can also reduce the execution time by a factor of about two through the use of fast transforms, but this may not make up for the use of more collocation points to reduce aliasing errors. Good analyses of spectral methods are given by Gottlieb and Orszag [1], Orszag [2], and Fletcher [3].

Veronis [4] and Busse [5] were among the first to implement the Galerkin method for finite-amplitude Rayleigh–Bénard convection. Veronis integrated the time-dependent streamfunction-vorticity form of the 2D conservation equations until the average heat transport became steady. This procedure was, and probably still is, the most common way of solving these equations, irrespective of the particular numerical method used. Busse used the fourth-order streamfunction formulation and applied Newton's method to solve the steady state problem directly. He also assumed rigid boundaries, which is reflected in the complexity of his basis functions in the vertical direction. Frick *et al.* [6] used poloidal and toroidal velocity fields to generalize Busse's work to 3D bimodal convection. This formulation is the most straightforward to implement since it avoids problems associated with solving for the pressure or vorticity. Of course, this advantage comes at the cost of introducing higher order and more complex differential equations.

Finite difference methods were used by Deardoff [7] and Fromm [8], using a formulation similar to the one used by Veronis. The fact that no-slip boundary conditions were relatively easy to implement favors these methods in general. Lipps and Somerville [9] performed 3D calculations with a similar method, but because of computational limitations, the grid was so coarse as to make the validity of the results questionable. An alternative finite difference method was developed by Chorin [10]. The primitive variables were differenced as in compressible flow problems; hence the name "artificial compressibility." This method, however, appears to be rather inefficient compared to other methods.

Methods combining the advantages of both Galerkin and finite difference methods were developed by Rogers and Beard [11] and Meyer-Spasche and Keller [12, 13] for the 2D Taylor–Couette problem (vortex flow between concentric cylinders), and by McDonough [14] for the 2D Rayleigh–Bénard problem. The basic idea of these "mixed" methods was to use the Galerkin method in the direction where it was most efficient or convenient, and to use finite differencing in the other direction. Thus, the dependent variables were expanded in Fourier series in the periodic (axial or horizontal) direction, and the Galerkin method applied. A system of ordinary differential equations (ODEs) in the other direction was obtained in each case, which was approximated by second-order-accurate centered finite differences. Meyer-Spache and Keller solved the resulting system of nonlinear algebraic equations by a full Newton's method, while McDonough first performed a modal decoupling. The former iteration scheme required fewer iterations, but the latter required less arithmetic per iteration. It appears that the total arithmetic was significantly less for the second method. Rogers and Beard solved the time-dependent problem, but they did not note the number of time steps required to reach steady state. Bourke [15] developed a specialized 3D mixed method for global weather prediction. He used a transform method with spherical harmonics in latitude and longitude for each of five "levels" in the vertical direction. The latter were coupled together through low order differencing of the governing equations in integro-differential form. Unfortunately, no analysis or discussion of the effects of the number of levels on the accuracy of the scheme was given. More recently, McDonough and Catton [16] considered 2D convection in a finite box. Because of the no-slip boundary conditions on the lateral walls, "beam" functions were used instead of trigonometric functions for the streamfunction. Among other applications of mixed methods, 3D compressible MHD simulations were carried out by Schnack *et al.* [17], where Fourier series were used in two of the directions and centered differencing with smoothing in the third.

In this paper a mixed finite difference/Galerkin method is implemented for 3D Rayleigh–Bénard convection. We consider the "classical" problem, which consists of a fluid confined between two rigid, parallel, perfectly conducting plates perpendicular to the gravity vector. The temperature of the lower one is maintained at $T_H$ and the upper one at $T_C$, with $\Delta T = T_H - T_C > 0$. All fluid properties are assumed to be constant except for density when it is multiplied by gravity. This is known as the "Boussinesq approximation," and the corresponding system of conservation equations as the "Boussinesq equations." We assume the physical parameters of the problem are such that stable steady solutions to these equations exist. (For other parameter values, one would not necessarily expect the iterations to converge.) The method of McDonough [14] is used, but with two important extensions. The first is the generalization from 2D to 3D convection, and the second is the replacement of centered finite differencing with fourth-order-accurate operator compact implicit (OCI) differencing. Solving the Boussinesq equations in poloidal–toroidal form leads to both second- and fourth-order ODEs in the vertical direction. These are solved by the OCI schemes of Stepleman [18] and Buell [19], respectively. Since only a moderate number of modes is needed here, and because of flexibility in choosing a set of basis functions, a Galerkin method is used instead of a transform or pseudospectral method. On the other hand, if some application of the present method requires a large number of modes, it would be appropriate (and not too difficult) to replace the Galerkin method as implemented here with a transform method.

## 2. Boussinesq Equations

After making the Boussinesq approximation and assuming steady flow, the nondimensional conservation equations of mass, momentum and energy are

$$\nabla \cdot \mathbf{u} = 0, \tag{1a}$$

$$\nabla^2 \mathbf{u} - \nabla p + R\theta\, \mathbf{e}_3 = \frac{1}{P} \mathbf{u} \cdot \nabla \mathbf{u}, \tag{1b}$$

$$\nabla^2 \theta + \mathbf{e}_3 \cdot \mathbf{u} = \mathbf{u} \cdot \nabla \theta, \tag{1c}$$

where we have scaled lengths with the fluid depth $d$, the velocity $\mathbf{u}$ with $\kappa/d$, the reduced pressure $p$ with $\rho_0 \kappa v/d^2$, and the temperature deviation $\theta$ from the static profile with $\Delta T$. Here, $\kappa$ is the thermal diffusivity, $v$ is the kinematic viscosity, and $\rho_0$ is the density at a reference temperature $T_0$. The unit vector in the vertical direction ($z$) is denoted by $\mathbf{e}_3$. The Rayleigh and Prandtl numbers are defined by

$$R = \frac{\alpha g\, d^3\, \Delta T}{\kappa v} \quad \text{and} \quad P = \frac{v}{\kappa},$$

where $\alpha$ is the coefficient of thermal expansion and $g$ is the acceleration of gravity. The Laplacian is given by

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}.$$

The no-slip and perfectly-conducting-plates assumptions yield the boundary conditions

$$\mathbf{u} = \theta = 0, \qquad z = 0, 1. \tag{2}$$

We can eliminate the continuity equation (1a) by replacing the velocity vector with poloidal and toroidal fields [5],

$$\mathbf{u} = \nabla \times \nabla \times (\mathbf{e}_3 \phi) + \nabla \times (\mathbf{e}_3 \psi). \tag{3}$$

both of which are solenoidal. Pressure is eliminated by performing the corresponding operations (that is, $\mathbf{e}_3 \cdot \nabla \times \nabla \times$ and $\mathbf{e}_3 \cdot \nabla \times$) on the vector momentum equation (1b). This yields the Boussinesq equations,

$$\begin{aligned}
\nabla^4 \Delta_2\phi - R\,\Delta_2\theta = \frac{1}{P}\,[&\nabla^2 \Delta_2\phi_x(\phi_{xz} + \psi_y) + \nabla^2 \Delta_2\phi_y(\phi_{yz} - \psi_x) - \nabla^2 \Delta_2\phi_z \Delta_2\phi \\
&+ 2\,\nabla^2\phi_{xy}(2\phi_{xyz} - \psi_{xx} + \psi_{yy}) \\
&- 2\psi_{xxz}(\phi_{xyz} + \psi_{yy}) + 2\psi_{yyz}(\phi_{xyz} - \psi_{xx}) \\
&- (\nabla^2\phi_{xz} - \Delta_2\psi_y + \psi_{yzz})\,\Delta_2\phi_x \\
&- (\nabla^2\phi_{yz} + \Delta_2\psi_x - \psi_{xzz})\,\Delta_2\phi_y \\
&+ (\nabla^2\phi_{xx} - \nabla^2\phi_{yy} + 2\psi_{xyz})(\phi_{xxz} - \phi_{yyz} + 2\psi_{xy})], \qquad (4a)
\end{aligned}$$

$$\begin{aligned}
\nabla^2 \Delta_2\psi = \frac{1}{P}\,[&\Delta_2\phi_x(\phi_{yzz} - \psi_{xz}) - \Delta_2\phi_y(\phi_{xzz} + \psi_{yz}) + \Delta_2\phi_z \Delta_2\psi \\
&- \Delta_2\phi\,\Delta_2\psi_z + (\phi_{xz} + \psi_y)\,\Delta_2\psi_x + (\phi_{yz} - \psi_x)\,\Delta_2\psi_y], \qquad (4b)
\end{aligned}$$

$$\nabla^2\theta - \Delta_2\phi = (\phi_{xz} + \psi_y)\theta_x + (\phi_{yz} - \psi_x)\theta_y - \Delta_2\phi\theta_z, \qquad (4c)$$

where independent variables as subscripts denote partial differentiation and

$$\Delta_2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

is the horizontal Laplacian. The boundary conditions follow directly from (1a) and (2),

$$\phi = \phi_z = \psi = \theta = 0, \qquad z = 0, 1. \qquad (5)$$

In the horizontal directions $x$ and $y$, we assume that the dependent variables are periodic with wavenumbers $a$ and $b$, respectively.

The above formulation has several advantages over the more common primitive-variable and streamfunction-vorticity formulations. The main one is the reduction of the number of dependent variables from five (primitive variables) to three, which reduces the required storage and simplifies the iteration scheme. For an infinite Prandtl-number fluid the number of variables is reduced and the iterations are simplified even more since (4b) and (5) yield $\psi \equiv 0$. If an alternative formulation is chosen, then the numerical method used depends on how either the pressure or vorticity is handled, and it usually becomes quite specialized (especially if boundaries are present). The elimination of these terms allows the implementation of almost any numerical method in a straightforward manner. The cost of these advantages is the introduction of higher order and more complex differential equations.

## 3. Numerical Method

We start by approximating the horizontal dependencies of $\phi$, $\psi$, and $\theta$ with truncated double Fourier-series expansions,

$$\phi \simeq \sum_{i=0}^{K} \sum_{j=0}^{J} \phi_{ij}(z) \cos a_i x \cos b_j y, \tag{6a}$$

$$\psi \simeq \sum_{i=1}^{K} \sum_{j=1}^{J} \psi_{ij}(z) \sin a_i x \sin b_j y, \tag{6b}$$

$$\theta \simeq \sum_{i=0}^{K} \sum_{j=0}^{J} \theta_{ij}(z) \cos a_i x \cos b_j y, \tag{6c}$$

where $a_i = ia$ and $b_j = jb$. We set $\phi_{00} \equiv 0$, since it will not appear anywhere else. For the time-dependent problem (and in general), the three other types of modes (for example, $\sin a_i x \cos b_j y$) must be included in (6). For steady flow, (6) is sufficient since the convection patterns are assumed to be mirror-symmetric around cell boundaries [6]. Nonsymmetric solutions are known for layers with stress-free boundaries [20], but it is not known if such solutions are stable for the no-slip case. (Stable nonsymmetric solutions appear to be very unlikely for moderate to large Prandtl numbers.) There is no experimental evidence for nonsymmetrical steady flow known to the author. However, general periodic patterns may be calculated with the present method by including both sine and cosine series in (6).

We allow $J$ to be a function of $i$ so that terms which are known to be small do not have to carried. In the following, $J$ is treated as a vector, $J = (J_0 J_1 \cdots J_K)$, and we replace the double summation notation in (6) with $\sum_{ij}^{KJ}$, where the starting indices are determined by the context. In standard implementations of spectral methods in two dimensions, one would use a "rectangular" truncation (where $J_i = \text{constant}$) or a "triangle" truncation (where $J_i = K - i$). The former is especially common in transform and pseudospectral schemes, and the latter in Galerkin schemes. Both truncations, however, are overly restrictive and may result in the use of more modes than is necessary for a given accuracy. In the present implementation of the Galerkin method, each element of $J$ is a separate parameter. Four possibilities for $J$ are given in Table I. These are the only ones we will use, so that if

TABLE I

The Four Truncations of the Fourier-Series Expansions Used Here

| $K$ | $M$ | $J$ |
|---|---|---|
| 7 | 20 | (3 2 2 2 1 1 1 0) |
| 9 | 31 | (5 4 3 2 2 2 1 1 1 0) |
| 11 | 46 | (7 6 5 4 3 2 2 2 1 1 1 0) |
| 13 | 62 | (7 7 7 6 5 4 3 2 2 2 1 1 1 0) |

$K$ is given, then $J$ does not need to be given. We emphasize that the appropriate "shape" of $J$ (when plotted as a histogram) can only be determined *a posteriori.* Here, we found that (for fixed $M$) the most accurate calculations obtains when half again as many $x$-direction modes are used as $y$-direction modes (that is, $K \simeq 1.5\, J_0$). For other problems, this most certainly will not be true. Lacking any information about the solution, one should use the triangle truncation as defined above. For some problems it may be more convenient to interchange the $i$ and $j$ summations in (6). In this case $J$ would be a scalar and,$K$ a vector. The total number of modes in the approximation for $\theta$ (6c) is

$$M = \sum_{i=0}^{K} J_i. \tag{7}$$

The number of modes in (6a) is $M-1$, and in (6b) there are $M-J_0-K-1$ modes.

The Galerkin method is implemented by substituting the approximations (6) into (4), multiplying by the corresponding basis functions, and integrating over the planform of one cell. The result is a coupled system of $3M-J_0-K-2$ ODEs,

$$\begin{aligned} &D^4\phi_{mn} - 2c_{mn}^2 D^2\phi_{mn} + c_{mn}^4\phi_{mn} \\ &\quad = R\theta_{mn} + \frac{f_{mn}^{(1)}}{4Pc_{mn}^2 g_{mn}}, \qquad m=0, ..., K,\ n=0, ..., J_m,\ mn \neq 00, \end{aligned} \tag{8a}$$

$$\begin{aligned} &D^2\psi_{mn} - c_{mn}^2\psi_{mn} \\ &\quad = \frac{f_{mn}^{(2)}}{4Pc_{mn}^2}, \qquad m=1, ..., K,\ n=1, ..., J_m, \end{aligned} \tag{8b}$$

$$\begin{aligned} &D^2\theta_{mn} - c_{mn}^2\theta_{mn} \\ &\quad = -c_{mn}^2\phi_{mn} + \frac{f_{mn}^{(3)}}{4g_{mn}}, \qquad m=0, ..., K,\ n=0, ..., J_m, \end{aligned} \tag{8c}$$

where $D = d/dz$, $c_{mn}^2 = a_m^2 + b_n^2$ and

$$g_{mn} = \begin{cases} 4, & m=n=0, \\ 2, & m=0 \text{ or } n=0, \\ 1, & m \neq 0 \text{ and } n \neq 0. \end{cases}$$

The last quantity came from the integrals of the linear terms. The projection of the nonlinear terms onto mode $mn$ is denoted by $f_{mn}^{(i)}$, $i=1, 2, 3$. These are given by the double summations

$$
\begin{aligned}
f^{(1)}_{mn} = \sum_{ij}^{JK} \sum_{kl}^{JK} (s^{(1)}_{ijklmn}\, D^2\phi_{ij}\, D\phi_{kl} + s^{(2)}_{ijklmn}\phi_{ij}\, D^3\phi_{kl} \\
+ s^{(3)}_{ijklmn}\phi_{ij}\, D\phi_{kl} + s^{(4)}_{ijklmn}\, D\phi_{ij}\, D\psi_{kl} \\
+ s^{(5)}_{ijklmn}\phi_{ij}\, D^2\psi_{kl} + s^{(6)}_{ijklmn}\, D^2\phi_{ij}\psi_{kl} \\
+ s^{(7)}_{ijklmn}\phi_{ij}\psi_{kl} + s^{(8)}_{ijklmn}\psi_{ij}\, D\psi_{kl}),
\end{aligned}
\tag{9a}
$$

$$
\begin{aligned}
f^{(2)}_{mn} = \sum_{ij}^{JK} \sum_{kl}^{JK} (s^{(9)}_{ijklmn}\phi_{ij}\, D^2\phi_{kl} + s^{(10)}_{ijklmn}\phi_{ij}\, D\psi_{kl} \\
+ s^{(11)}_{ijklmn}\, D\phi_{ij}\psi_{kl} + s^{(12)}_{ijklmn}\psi_{ij}\psi_{kl}),
\end{aligned}
\tag{9b}
$$

$$
f^{(3)}_{mn} = \sum_{ij}^{JK} \sum_{kl}^{JK} (s^{(0)}_{ijklmn}\, D\phi_{ij}\theta_{kl} + s^{(13)}_{ijklmn}\phi_{ij}\, D\theta_{kl} + s^{(14)}_{ijklmn}\psi_{ij}\theta_{kl}).
\tag{9c}
$$

The interaction coefficients $s^{(p)}_{ijklmn}$, $p = 0, ..., 14$, are given in Appendix A. In general, the summations in (9) would require $O(M^2)$ operations. Here, they require only $O(M)$ operations because the arrays representing the trigonometric triple-product integrals are very sparse. Using the particular form of this property allows one to eliminate the $ij$ indices and the corresponding summations, and to rewrite (9) as convolution summations [2]. We do not do this here since it is just as efficient to evaluate (9) as is, as long as the zero terms are not calculated.

The ODEs (8) are solved by finite differencing. The modal functions are defined on a uniform grid with $L$ grid points across the layer. As mentioned earlier, a full Newton's method for all the algebraic unknowns would require an excessive amount of storage and linear algebra (especially for 3D problems). Thus we apply Newton's method to each ODE with respect to only the corresponding "modal function." Since $f^{(1)}_{mn}$, $f^{(2)}_{mn}$, and $f^{(3)}_{mn}$ are linear in $\phi_{mn}$, $\psi_{mn}$, and $\theta_{mn}$, respectively, this means that the appropriate terms are simply moved from the right-hand side (RHS) to the left-hand side. This decoupling of the modal functions lowers the convergence rate of the iterations from quadratic to linear. Each ODE is now in the form of a general linear fourth-order (8a) or second-order (8b), (8c) ODE, where the coefficients and RHSs are functions of the other modal functions.

In order to maximize efficiency and ease of implementation, we put several requirements on the finite difference method to be used to solve the ODEs. First, the method should be fourth-order accurate in the mesh size $h = 1/(L-1)$, since second-order centered differencing requires about 50 to 100 grid points for good accuracy [12, 14]. Second, the method should not require values of the coefficients or the RHS at points other than grid points in order to avoid high-order interpolation. This is important because many high-order finite difference methods assume that the coefficients and RHS are continuous functions, which are not necessarily evaluated at grid points. Third, the bandwidth of the matrix approximation should be the smallest possible (tridiagonal for second-order ODEs, pentadiagonal for fourth-order ODEs). This minimizes linear algebra and problems at the boundaries. Finally, the matrix elements should be simple polynomial functions of the coefficients and the mesh size. The OCI method of Stepleman [18] for second-order ODEs, and of Buell [19] for fourth-order ODEs, satisfy all these requirements. Furthermore, they have no cell Reynolds number limitations, although this property is not needed here. Reviews of the OCI literature and details of the derivations of the difference formulae may be found in these two papers. We note only that these schemes are based on the use of the "operator" (that is, the entire differential equation) to approximate the high-order derivatives appearing in the truncation error terms of centered differences. Since Stepleman's analysis was for a nonlinear ODE, he could not give the matrix elements of his approximation explicitly. In Appendix B, we specialize his analysis to a general linear ODE and give the matrix elements. The coefficients and RHSs of (8) are functions of the derivatives of the solution and must also be evaluated with fourth-order accuracy. This is most easily accomplished for each modal function at the same time a solution is calculated for it. Appropriate difference formulae for the individual derivatives can be derived from the analyses in [18, 19].

Each "projection" in (9) is evaluated only when the corresponding ODE is to be solved. Thus, we employ Gauss–Seidel iteration for the modal functions. If Fourier transforms are used to evaluate (9), or if the computations are performed in parallel, then the corresponding method would be Jacobi iteration. Any advantages of alternative methods of evaluating (9) would have to be weighed against a possible increase in the number of iterations, as well as the considerations mentioned earlier. Decoupling of the modal functions necessitates damping to obtain convergence of the iterations. It turns out, however, that only the energy equation (8c) needs to be damped. The optimal damping factor was not determined theoretically, but numerical experiments showed that it ranges from 1.0 for [redacted] dent of the damping factor, if the iterations converge.

## 4. Convergence Tests

In this section we present results of numerical examples from which the asymptotic and absolute accuracies of the method can be ascertained, as well as its efficiency.

One of the most important physical quantities based on the solution to the Boussinesq equations is the total heat transfer across the layer. The quantity becomes the Nusselt number, $N$, when it is scaled with the conductive heat transfer. It is defined by

$$N = 1 - \frac{ab}{4\pi^2} \int_0^{2\pi/b} \int_0^{2\pi/a} \left. \frac{\partial \theta}{\partial z} \right|_{z=0} dx\, dy. \qquad (10)$$

Using (1a), (1c) and the approximation (6), this becomes

$$N = 1 - \frac{ab}{4\pi^2}\int_0^1 \int_0^{2\pi/b} \int_0^{2\pi/a} \Delta_2\phi\,\theta\,dx\,dy\,dz$$

$$\simeq 1 + \frac{1}{4}\sum_{ij}^{JK} c_{ij}^2 g_{ij} \int_0^1 \phi_{ij}\theta_{ij}\,dz. \tag{11}$$

Note that (11) is related to the $L^2$ ("weak") norm of the solution [14]. Evaluating (10) directly yields a pointwise (or "strong") measure of the solution:

$$N \simeq 1 - \left.\frac{d\theta_{00}}{dz}\right|_{z=0}. \tag{12}$$

We will use (11) here since (12) requires one-sided differencing and is, in practice, much less accurate. The integral in (11) is approximated using Simpson's method with end correction [21], which is sixth-order accurate in $h$.

As a sample problem, we solved (4) with $R = 40{,}000$, $P = 20$, $a = 2.5$, and $b = 6.0$. A pointwise iteration tolerance of $\varepsilon = 10^{-5}$ was used for all calculations. This value is sufficiently small to ensure that the final iteration error is much smaller than either the finite-difference or Fourier-series expansion truncation errors. For maximum efficiency in actual applications, $\varepsilon$ should be chosen slightly smaller than an estimate of the sum of the truncation errors (typically, $\varepsilon = 10^{-3}$).

Pointwise and Nusselt-number convergence tests are shown in Table II for decreasing $h$ and $K = 9$. In addition to OCI differencing we present results using second-order-accurate centered (SOC) differencing and Richardson extrapolation of centered (REC) differencing. The latter is implemented on those points on a given grid that are in common with points on a coarser grid. For any grid function $u_i$

TABLE II

Pointwise and Nusselt-Number Convergence Tests for Three Finite Difference Methods

| Method | $L$ | $\phi_{10}(1/2)$ | $\psi_{11}(1/2)$ | $\theta_{10}(1/2)$ | $N$ |
|---|---|---|---|---|---|
| OCI | 13 | 8.7411 | −0.6197 | 0.09862 | 3.7418 |
| | 19 | 8.7845 | −0.6445 | 0.09880 | 3.7801 |
| | 27 | 8.7922 | −0.6481 | 0.09891 | 3.7866 |
| SOC | 13 | 9.0142 | −1.1701 | 0.10376 | 4.4171 |
| | 19 | 8.9582 | −0.8908 | 0.10204 | 4.0742 |
| | 27 | 8.8833 | −0.7654 | 0.10056 | 3.9265 |
| REC | 19 | 8.9134 | −0.6674 | 0.10066 | 3.7999 |
| | 27 | 8.8143 | −0.6501 | 0.09920 | 3.7906 |

calculated with SOC differencing on a grid with mesh size $h_i$, Richardson extrapolation yields

$$u_{\mathrm{REC}} = \frac{h_2^2 u_1 - h_1^2 u_2}{h_2^2 - h_1^2}, \tag{13}$$

which is a fourth-order-accurate approximation to the true solution. A significant disadvantage of (13) is the scarcity of points common to both grids if one is interested in pointwise solutions (as opposed to, say, just the Nusselt number). From the table one sees that the OCI approximation converges at a rate between fourth and fifth order (determined by assuming the error is of the form $Ch^n$ and solving for $n$ using three values of $h$). This is due to the approximation used for the derivative boundary conditions [19]; for larger $L$ the method is closer to fourth-order accurate. Centered differencing yields results very close to the theoretically predicted second-order accuracy, but the absolute accuracy is poor compared to OCI differencing. We find that for the calcuation of the Nusselt number about 106 grid points are needed by the former to match the accuracy of the latter with 19 grid points. Since the extra arithmetic associated with OCI differencing is less than 5% of the total, this method gives a gain in efficiency of a factor of about five. Of course, the amount of gain depends on the parameters of the problem and (especially) on the absolute accuracy required. (The gain is smaller for less accurate calculations, and larger when more accuracy is required.) For 2D convection the gain in efficiency is even greater (see Buell and Catton [22, 23], where the present numerical method was used for the wavenumber selection problem in roll convection). Richardson extrapolation is much better than SOC but not as good as OCI differencing for this case. In general, REC and OCI differencing are about equally accurate; however, the former is significantly less efficient since two solutions are needed before (13) can be evaluated. The pointwise and Nusselt-number convergence tests yield essentially the same conclusions for each finite difference method. This indicates that the solutions are smooth and that it is appropriate to implement a high-order numerical method in the first place.

Convergence of the Fourier-series expansions (6) is demonstrated in Tables III

TABLE III

Residual and Nusselt-Number Convergence Tests of the Fourier-Series Expansions

| | | Residual | | | | |
|---|---|---|---|---|---|---|
| $K$ | $M$ | $\phi$ | $\psi$ | $\theta$ | $N$ | CPU |
| 7 | 20 | −4.82(+3) | −9.40(+1) | −3.64(+0) | 3.7608 | 0.039 |
| 9 | 31 | −4.74(+3) | 6.22(+1) | −2.25(+0) | 3.7418 | 0.124 |
| 11 | 46 | −1.65(+3) | −7.49(+0) | −6.57(−1) | 3.7390 | 0.381 |
| 13 | 62 | −2.09(+2) | −6.00(+0) | −7.40(−2) | 3.7378 | 0.925 |

and IV using 13 grid points. One of the best tests of the accuracy of a spectral method (or any "global" approximation) is to substitute the computed solution back into the governing equations and to evaluate the residual. The maximum of this residual over all points in space is a "strong" error norm. Here, we chose to evaluate the residual shown in Table III at $(x, y, z) = (0, 0, \frac{1}{2})$ for the $\phi$ and $\theta$ equations, and at $(x, y, z) = (\pi/2a, \pi/2b, \frac{1}{2})$ for the $\psi$ equation. We found these points to be at least qualitatively representative of all points, and thus of the

TABLE IV

Convergence in Fourier Space of the Modal Functions

| $i$ | $j$ | $\phi_{ij}$ | $\theta_{ij}$ | $\psi_{ij}$ |
|---|---|---|---|---|
| 0 | 0 |  | −2.668(−1) |  |
| 0 | 1 | 7.924(−1) | 2.031(−2) |  |
| 0 | 2 | 2.584(−2) | 2.313(−2) |  |
| 0 | 3 | 2.174(−3) | 4.685(−3) |  |
| 0 | 4 | 2.891(−4) | 2.543(−3) |  |
| 0 | 5 | 7.084(−6) | 1.009(−4) |  |
| 1 | 0 | 8.741(+0) | 9.862(−2) |  |
| 1 | 1 | −7.287(−2) | 1.266(−2) | −6.197(−1) |
| 1 | 2 | 4.244(−2) | 2.658(−2) | −4.085(−3) |
| 1 | 3 | 9.164(−5) | 1.429(−3) | −1.532(−3) |
| 1 | 4 | 1.665(−4) | 1.346(−3) | 8.975(−6) |
| 2 | 0 | 3.081(−2) | 2.517(−2) |  |
| 2 | 1 | 3.279(−1) | 5.265(−2) | −1.220(−2) |
| 2 | 2 | −6.046(−3) | −7.217(−3) | −1.853(−2) |
| 2 | 3 | 9.453(−4) | 3.565(−3) | 1.619(−4) |
| 3 | 0 | 4.168(−1) | 4.577(−2) |  |
| 3 | 1 | −2.698(−2) | −7.412(−3) | −6.105(−2) |
| 3 | 2 | 4.376(−3) | 5.783(−3) | 7.187(−4) |
| 4 | 0 | 2.649(−2) | 1.739(−2) |  |
| 4 | 1 | 2.844(−2) | 1.748(−2) | −6.123(−4) |
| 4 | 2 | −1.542(−3) | −3.162(−3) | −2.217(−3) |
| 5 | 0 | 3.626(−2) | 2.343(−2) |  |
| 5 | 1 | −4.102(−3) | −4.545(−3) | −1.108(−4) |
| 5 | 2 | −4.523(−4) | −3.259(−4) | 1.402(−4) |
| 6 | 0 | 5.896(−3) | 1.048(−2) |  |
| 6 | 1 | 2.314(−3) | 5.534(−3) | 1.275(−4) |
| 7 | 0 | 4.465(−3) | 9.847(−3) |  |
| 7 | 1 | −6.179(−4) | −1.878(−3) | 6.198(−4) |
| 8 | 0 | 1.100(−3) | 4.999(−3) |  |
| 8 | 1 | 3.840(−4) | 2.603(−3) | 4.431(−5) |
| 9 | 0 | 7.305(−4) | 3.692(−3) |  |

maximum norm of the residual. The residual converges to zero in all cases, but somewhat slowly at small $K$ and in an oscillatory manner for the $\psi$ equation. (The notation 4.82(+3), for example, means $4.82 \times 10^3$.) The Nusselt number converges at an "exponential" rate with increasing $K$, which is expected for smooth solutions. The computed solutions are sufficiently accurate (for most purposes) with $K=9$, even though this is not evident from an examination of the residual. The last column of Table III shows the required computational time, in units of s/iteration on a Cray XMP. The CPU time is closer to $O(M^3)$ than to the expected $O(M^2)$. However, most of the terms in (9) contain $\psi$, and it is easily verified that the CPU time is proportional to the square of the number of terms in the $\psi$ approximation. The total CPU time is obtained by multiplying by the number of iterations needed. This number depends strongly on the iteration tolerance and the quality of the initial guess, but is usually between 50 and 200.

Shown in Table IV is the convergence in Fourier space of the modal functions with $K=9$. Within each grouping ($i=$ constant) the convergence of the Fourier series in the $y$ direction is evident. Similarly, convergence in the $x$ direction is seen when $j=$ constant. Due to the symmetry of the Rayleigh–Bénard problem, the modal functions $\phi_{ij}$ and $\theta_{ij}$ are even and $\psi_{ij}$ is odd when $i+j$ is odd. The other modes have the opposite parity. (The parity of a function is defined with respect to $z=\frac{1}{2}$.) In the table the even functions are evaluated at $z=\frac{1}{2}$ and the odd functions at $z=\frac{1}{4}$. The resulting values are usually close to the maxima of the modal functions, and this procedure facilitates comparisons with other results as they become available.

## 5. Numerical Examples

This section is limited to a short discussion of one effect of the Prandtl number and some visualizations of the solution since the physics of the problem is (and will be) the main focus of other papers [22–24].

One of the most significant differences between two- and three-dimensional convection is the possibility of vertical vorticity in the latter, which is given by $-\Delta_2\psi$. From (4b) it is clear that for large Prandtl numbers, the toroidal field $\psi$ is approximately proportional to $1/P$. In particular, $\psi$ is identically zero for infinite $P$. On the other hand, the vertical velocity does not depend directly on $\psi$. Therefore, the heat transfer is expected to decrease with decreasing $P$ because energy must be removed from the poloidal field $\phi$ in proportion to $1/P$ in order to drive the toroidal field. This is shown in Table V, where the Nusselt number $N$ is given for several values of $1/P$ with $R=30{,}000$, $a=2.4$, and $b=6$. For this moderate value of the Rayleigh number, $N$ decreases linearly at first and then levels off. For 2D [illegible] of $P$ when $P > 7$, which is to be expected since there is no vertical vorticity in 2D convection for any Prandtl number.

Figure 1 and the values of the modal functions shown in Table V help to explain

TABLE V

The Nusselt Number and Four Modal Functions at $z=\frac{1}{2}$ as Functions of the Inverse of the Prandtl Number for $R=30{,}000$

| $1/P$ | $N$ | $\phi_{10}$ | $\phi_{30}$ | $\phi_{01}$ | $\psi_{11}$ |
|---|---|---|---|---|---|
| 0 | 3.512 | 7.67 | 0.31 | 0.80 | 0.0 |
| 0.02 | 3.454 | 7.84 | 0.36 | 0.65 | −0.19 |
| 0.04 | 3.396 | 7.93 | 0.40 | 0.52 | −0.29 |
| 0.06 | 3.335 | 7.98 | 0.43 | 0.39 | −0.32 |
| 0.08 | 3.270 | 8.01 | 0.46 | 0.25 | −0.26 |
| 0.10 | 3.244 | 8.03 | 0.47 | ~0 | ~0 |

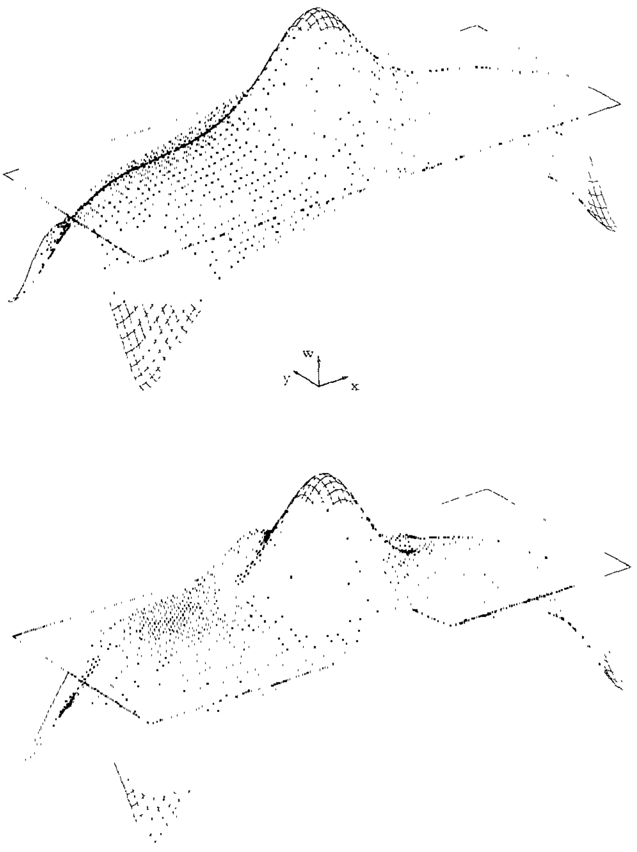

FIG. 1. Vertical velocity at the midlayer for $P=\infty$ (top) and $P=20$ (bottom), with $R=40{,}000$, $a=2.5$, and $b=6$.

the "leveling off" of the Nusselt number for 3D convection. Shown in the figure are surface plots of the vertical velocity ($w$) at the midplane of one cell (centered horizontally at the origin) for $R=40{,}000$, $a=2.5$, $b=6$, and two Prandtl numbers. The lines $w=0$ at the cell boundaries are given for reference. The figure and table both reveal that the decrease in $N$ corresponds to a decrease in the magnitude of the largest mode in the $y$ direction, which is given by $\phi_{01}(z)\cos by$. From (4c) we see that this mode multiplied by the first mode in the $x$ direction, $\phi_{10}(z)\cos ax$, drives the first toroidal mode, $\psi_{11}(z)\sin ax \sin by$. However, $\psi_{11}$ tends to decrease the magnitude of $\phi_{01}$ through (4a). Thus for $P$ decreasing from infinity, $\phi_{01}$ decreases monotonically while the magnitude of $\psi_{11}$ first increases from zero, reaches a maximum, and then decreases back to zero. For sufficiently small $P$, all $y$-direction modes approach zero and the flow becomes 2D. This cannot be observed experimentally since 2D flow is known to be unstable for this Rayleigh number. Another feature evident in Fig. 1 and Table V is that as the magnitudes of the $y$-direction modes decrease with decreasing $P$, the magnitudes of the $x$-direction harmonics ($\phi_{n0}$, especially $n>1$) increase. This leads to the surprising result that fully 3D convection at high $R$ requires fewer $x$-direction modes than 2D flow at (certain) lower $R$. (The total number of modes, of course, is considerably greater.)

Contours of $\psi$ in two horizontal planes corresponding to the bottom plot in Fig. 1 are plotted in Fig. 2. These show that each cell contains four vertical vortices

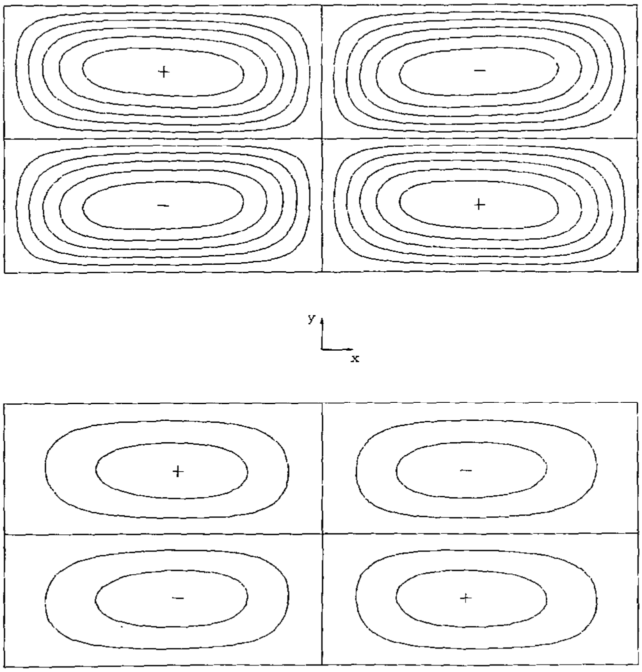

FIG. 2. Contours of $\psi$ in the horizontal planes $z=\frac{1}{2}$ (top) and $z=\frac{1}{4}$ (bottom) corresponding to the bottom plot of Fig. 1. Difference between contour levels is 0.1.

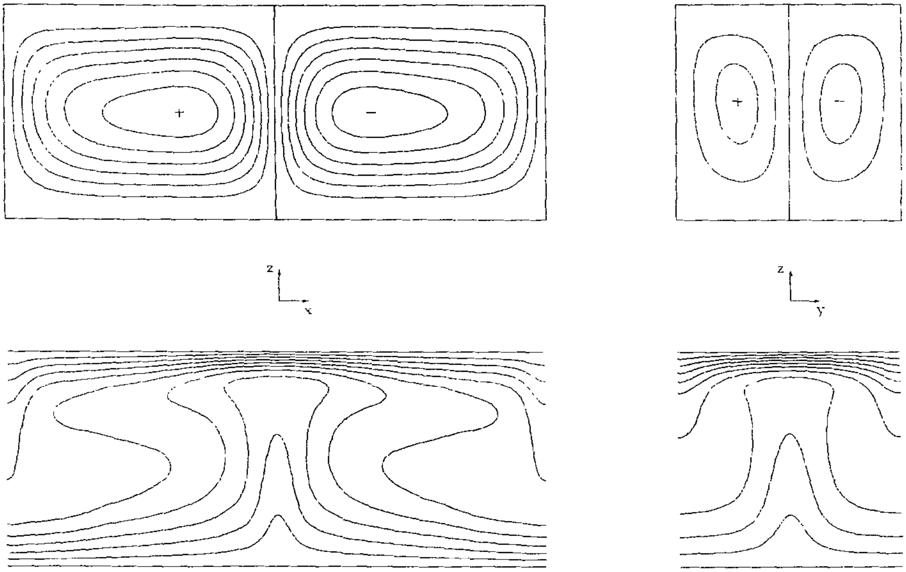

FIG. 3. Contours of $\phi_x$ (top left) and $\phi_y$ (top right) in the vertical planes $y=0$ and $x=0$, respectively, with the isotherms shown below, corresponding to the bottom plot of Fig. 1. Difference in contour levels is 3.0 (top) and 0.1 (bottom).

that are strongest in the midlayer and die off towards the boundaries (because of the no-slip condition). The vortices turn so that there is flow from the peak (in Fig. 1) towards the long edge, out along this edge to the short edge, and then back up the middle to the peak. Because of the higher horizontal harmonics, the "tilt" of the vortices is also noticeable.

One final visualization of 3D convection (with the same parameters as above) is given in Fig. 3. Shown there are contours of $\phi_x$ and $\phi_y$ in the planes $y=0$ and $x=0$, respectively, and the corresponding isotherms. (Both planes are vertical and go through the center of the bottom plot in Fig. 1.) We see that in both planes the fluid rises in a strong, narrow plume and descends over a larger area. In the vertical planes along the edges of the cell, the opposite is true because of the symmetry of the solution.

## 6. CONCLUSIONS

The mixed finite difference/Galerkin method has been shown here to be effective for solving the PDEs modeling steady 3D Rayleigh–Bénard convection, while requiring only slightly more array storage than that needed for the solution and its vertical derivatives. The method allows the use of any finite difference scheme in the vertical direction; however, OCI differencing is considerably more efficient than standard centered differencing, as well as more efficient and convenient than Richardson extrapolation. One disadvantage of the method is that on the order of 100 iterations are needed because of the decoupling of the horizontal modes. Coupling these together through the use of a full Newton's method would reduce the number of iterations to about five or ten, but this is not feasible because of the large amount of storage and linear algebra associated with it, and because the OCI methods used here have been developed only for scalar ODEs, not for systems of ODEs. For problems which require a very large number of modes, the Galerkin method becomes inefficient relative to transform methods.

## Appendix A

Below we list the interaction coefficients that appear in (9).

$$
\begin{aligned}
s^{(0)}_{ijklmn} &= a_i a_k I^{(1)}_{ikm} I^{(2)}_{jln} + b_j b_l I^{(2)}_{ikm} I^{(1)}_{jln},\\
s^{(1)}_{ijklmn} &= c^2_{ij} s^{(0)}_{ijklmn} - (a^2_i - b^2_j)(a^2_k - b^2_l)\, I^{(2)}_{ikm} I^{(2)}_{jln} - 4 a_i b_j a_k b_l I^{(1)}_{ikm} I^{(1)}_{jln},\\
s^{(2)}_{ijklmn} &= c^2_{ij}(c^2_{kl} I^{(2)}_{ikm} I^{(2)}_{jln} - s^{(0)}_{ijklmn}),\\
s^{(3)}_{ijklmn} &= -c^2_{ij} s^{(1)}_{ijklmn} - c^2_{kl} s^{(2)}_{ijklmn},\\
s^{(4)}_{ijklmn} &= 2[a_k b_l (a^2_i - b^2_j)\, I^{(2)}_{ikm} I^{(2)}_{jln} - a_i b_j (a^2_k - b^2_l)\, I^{(1)}_{ikm} I^{(1)}_{jln}],\\
s^{(5)}_{ijklmn} &= c^2_{ij}(a_i b_l I^{(1)}_{ikm} I^{(2)}_{jln} + b_j a_k I^{(2)}_{ikm} I^{(1)}_{jln}),\\
s^{(6)}_{ijklmn} &= s^{(4)}_{ijklmn} - s^{(5)}_{ijklmn},\\
s^{(7)}_{ijklmn} &= c^2_{kl} s^{(5)}_{ijklmn} - c^2_{ij} s^{(6)}_{ijklmn},\\
s^{(8)}_{ijklmn} &= 2(a^2_i b^2_l + b^2_j a^2_k)\, I^{(1)}_{ikm} I^{(1)}_{jln} - 4 a_i b_j a_k b_l I^{(2)}_{ikm} I^{(2)}_{jln},\\
s^{(9)}_{ijklmn} &= c^2_{ij}(a_i b_l I^{(1)}_{imk} I^{(1)}_{lnj} - b_j a_k I^{(1)}_{kmi} I^{(1)}_{jln}),\\
s^{(10)}_{ijklmn} &= c^2_{ij}(a_i a_k I^{(1)}_{imk} I^{(1)}_{lnj} + b_j b_l I^{(1)}_{kmi} I^{(1)}_{jnl} + c^2_{kl} I^{(1)}_{kmi} I^{(1)}_{lnj}),\\
s^{(11)}_{ijklmn} &= -c^2_{kl}(a_i a_k I^{(1)}_{imk} I^{(1)}_{lnj} + b_j b_l I^{(1)}_{kmi} I^{(1)}_{jnl} + c^2_{ij} I^{(1)}_{kmi} I^{(1)}_{lnj}),\\
s^{(12)}_{ijklmn} &= c^2_{ij}(a_i b_l I^{(1)}_{kmi} I^{(1)}_{jnl} - b_j a_k I^{(1)}_{imk} I^{(1)}_{lnj}),\\
s^{(13)}_{ijklmn} &= c^2_{ij} I^{(2)}_{ikm} I^{(2)}_{jln},\\
s^{(14)}_{ijklmn} &= a_i b_l I^{(2)}_{ikm} I^{(1)}_{jln} - b_j a_k I^{(1)}_{ikm} I^{(2)}_{jln}.
\end{aligned}
$$

The two triple-product integrals of trigonometric functions are given by

$$I^{(1)}_{ijk} = (2a/\pi) \int_0^{2\pi/a} \sin a_i x \sin a_j x \cos a_k x \, dx,$$

$$I^{(2)}_{ijk} = (2a/\pi) \int_0^{2\pi/a} \cos a_i x \cos a_j x \cos a_k x \, dx.$$

Evaluating these integrals yields

$$I^{(1)}_{ijk} = \begin{cases} 2, & k = 0 \text{ and } i = j, \\ 1, & k = i - j \text{ or } k = j - i,\ k \neq 0, \\ -1, & k = i + j, \\ 0. & i = 0 \text{ or } j = 0 \text{ or otherwise,} \end{cases}$$

$$I^{(2)}_{ijk} = \begin{cases} 4, & i = j = k = 0, \\ 2, & i = 0 \text{ and } j = k \neq 0, \text{ or } j = 0 \text{ and } i = k \neq 0, \text{ or } k = 0 \text{ and } i = j \neq 0, \\ 1, & k = i + j \text{ or } k = i - j \text{ or } k = j - i, \text{ and } i \neq 0 \text{ and } j \neq 0 \text{ and } k \neq 0, \\ 0, & \text{otherwise.} \end{cases}$$

The corresponding $y$-direction integrals are obtained by replacing $x$ and $a$ with $y$ and $b$, respectively.

## Appendix B

We present here an OCI finite difference approximation to a general linear second-order ODE with Dirichlet boundary conditions.

$$\frac{d^2u}{dz^2} + \alpha(z)\frac{du}{dz} + \beta(z)u = \gamma(z), \tag{B1}$$

$$u(0) = \delta_0, \qquad u(1) = \delta_1. \tag{B2}$$

Stepleman [18] derived an OCI approximation for nonlinear equations. For (B1) this approximation can be written out explicitly in matrix form,

$$q_{i,i-1}u_{i-1}q_{i,i}u_i + q_{i,i+1}u_{i+1} = r_i, \tag{B3}$$

where $u_i$ is the approximation to the true solution $u(z_i)$. The tridiagonal matrix elements $q_{ij}$ and the RHS vector elements $r_i$ are given by

$$q_{i,i-1} = 1 - \frac{h}{2}\alpha_i + \left[\frac{h}{24}(-3\alpha_{i-1} + 2\alpha_i + \alpha_{i+1}) + \frac{h^2}{12}\beta_{i-1} + \zeta_i(3\alpha_{i-1} + \alpha_{i+1} - 2h\beta_{i-1})\right], \tag{B4a}$$

$$q_{i,i} = -2 + h^2\beta_i + \left[\frac{h}{6}(\alpha_{i-1} - \alpha_{i+1}) - \frac{h^2}{6}\beta_i - 4\zeta_i(\alpha_{i-1} + \alpha_{i+1})\right], \tag{B4b}$$

$$q_{i,i+1} = 1 + \frac{h}{2}\alpha_i + \left[\frac{h}{24}(-\alpha_{i-1} - 2\alpha_i + 3\alpha_{i+1}) + \frac{h^2}{12}\beta_{i+1} + \zeta_i(\alpha_{i-1} + 3\alpha_{i+1} + 2h\beta_{i+1})\right], \tag{B4c}$$

$$r_i = h^2\gamma_i + \left[\frac{h^2}{12}(\gamma_{i-1} - 2\gamma_i + \gamma_{i+1}) - 2h\zeta_i(\gamma_{i-1} - \gamma_{i+1})\right], \tag{B4d}$$

where

$$\zeta_i \equiv -\frac{h^2}{144}(\alpha_{i-1} - 5\alpha_i + \alpha_{i+1}), 2 \leqslant i \leqslant L-1, \qquad \alpha_i \equiv \alpha(z_i), \text{ etc.}$$

We note that centered differencing is recovered if everything within the square brackets is deleted. The matrix elements corresponding to the boundary conditions (B2) are

$$\begin{aligned} q_{1,1} &= 1, \qquad r_1 = \delta_0, \\ q_{L,L} &= 1, \qquad r_L = \delta_1. \end{aligned} \tag{B5}$$

## Acknowledgment

This research was partly supported by NSF Grant MEA 81-05542 while the author was at UCLA.

## References

1. D. Gottlieb and S. A. Orszag, *Numerical Analysis of Spectral Methods: Theory and Applications* (NSF-CBMS Monograph No. 26, SIAM, Philadelphia, 1977).
2. S. A. Orszag, *J. Comp. Phys.* **37**, 70 (1980).
3. C. A. J. Fletcher, *Computational Galerkin Methods* (Springer-Verlag, Berlin, 1984).
4. G. Veronis, *J. Fluid Mech.* **26**, 49 (1966).
5. F. H. Busse, *J. Math. Phys.* **46**, 140 (1967).
6. H. Frick, F. H. Busse, and R. M. Clever, *J. Fluid Mech.* **127**, 141 (1983).
7. J. W. Deardoff, *J. Atmos. Sci.* **21**, 419 (1964).
8. J. E. Fromm, *Phys. Fluids* **8**, 1757 (1965).
9. F. B. Lipps and R. C. J. Somerville, *Phys. Fluids* **14**, 759 (1971).
10. A. J. Chorin, *J. Comp. Phys.* **2**, 12 (1967).
11. E. H. Rogers and D. W. Beard, *J. Comp. Phys.* **4**, 1 (1969).
12. R. Meyer-Spasche and H. B. Keller, Applied Mathematics Report, California Institute of Technology, Pasadena, 1978. (unpublished)
13. R. Meyer-Spasche and H. B. Keller, *J. Comp. Phys.* **35**, 100 (1980).
14. J. M. McDonough, Dissertation, School of Engineering and Applied Science, University of California, Los Angeles, 1980 (unpublished).
15. W. Bourke, *Mon. Weather Rev.* **102**, 688 (1974).
16. J. M. McDonough and I. Catton, *Int. J. Heat Mass Transfer* **25**, 1137 (1982).
17. D. D. Schnack, D. C. Baxter, and E. J. Caramana, *J. Comp. Phys.* **55**, 485 (1984).
18. R. S. Stepleman, *Math. Comput.* **30**, 92 (1976).
19. J. C. Buell, *SIAM J. Sci. Statist. Comput.* **7**, 1232 (1986).
20. F. H. Busse and A. C. Or, *J. Appl. Math. Phys. (ZAMP)* **37**, 608 (1986).
21. R. W. Hornbeck, *Numerical Methods* (Quantum, New York, 1975), p. 150.
22. J. C. Buell and I. Catton, *Phys. Fluids* **29**, 23 (1986).
23. J. C. Buell and I. Catton, *J. Fluid Mech.* **171**, 477 (1986).
24. J. C. Buell and I. Catton, *Phys. Fluids* **30**, 318 (1987).