JOURNAL OF COMPUTATIONAL PHYSICS 120, 260–265 (1995)

Incompressibility and No-Slip Boundaries in the Chebyshev–Tau Approximation: Correction to Kleiser and Schumann's Influence-Matrix Solution

J. WERNE

Advanced Study Program, National Center for Atmospheric Research, Boulder, Colorado 80307

Received July 21, 1994; revised January 23, 1995

I present a minor, but important correction to Kleiser and Schumann's influence-matrix solution for numerical simulation of 3D time-dependent incompressible flow between no-slip plane boundaries (L. Kleiser, and U. Schumann, in *Proc. 3rd GAMM Conf. Numerical Methods in Fluid Mechanics*, edited by E. H. Hirschel, Vieweg, Braunschweig, 1980, p. 165). The solution technique I present is, for the most part, identical to Kleiser and Schumann's, the only noteworthy difference being my treatment of the truncation (or "tau") errors. Though this improvement appears to be but a slight modification to their original algorithm, the error encountered when attempting to satisfy the incompressibility constraint is reduced dramatically (by a factor of 10^{11}), producing velocity fields which are divergence-free to within machine precision.

1. INTRODUCTION

In 1980 Kleiser and Schumann presented a numerical spectral solution technique for the Navier–Stokes equations in a cartesian geometry bounded by two parallel, no-slip planes [1]. Their method incorporates Fourier expansions in directions parallel to the no-slip boundaries and a Chebyshev-"tau" discretization [2, 3] in the perpendicular direction. In order to satisfy the no-slip condition at the bounding planes, Kleiser and Schumann use the "influence matrix" [4] (also referred to as the "capacitance matrix") and are careful to correct the truncation-errors (also called "tau"-errors) which contaminate *all of the mode-equations for the Chebyshev coefficients.*

A review and generalization of Kleiser and Schumann's influence matrix method to non-periodic geometries already exists [5] and I refer interested readers to that excellent work. My intent here is neither to review nor to generalize the influence matrix method further, but rather only to point out a subtle error present in Kleiser and Schumann's original formulation so that readers of Kleiser and Schumann's work can avoid introducing this easily overlooked mistake into their own computer programs

I begin by briefly outlining the important steps in the influence-matrix technique applied to a coupled system of 2 one-dimensional Helmholtz equations (as the system appears in the numerical solution to the Navier–Stokes equations). Next, I present the form of the truncation-errors in the Chebyshev-"tau" approximation to this system. I then detail the correction required to eliminate the truncation errors, carefully pointing out the subtle mistake made by Kleiser and Schumann. Although much of the content of this brief review already appears in a slightly different form in [1], the re-examination here is required to build the necessary framework with which to discuss the proper treatment of the truncation errors. I end with example-calculations of 3D Rayleigh–Bénard convection wherein the "tau" corrections are implemented: (1) with the original Kleiser–Schumann solution and (2) according to the present formulation. The magnitude of the normalized divergence in the resulting velocity fields is 7×10^{-3} and 7×10^{-14} respectively.

2. COUPLED HELMHOLTZ EQUATIONS

When numerically integrating the incompressible Navier–Stokes equations between no-slip boundaries, the standard primitive-variable solution-technique requires solving a coupled system of 1D Helmholtz equations of the form

$$(D^2 - \lambda)W = DP + r_1 \tag{1.a}$$

$$(D^2 - k^2)P = r_2 \tag{1.b}$$

$$W(\pm 1) = 0 \tag{1.c}$$

$$DW(\pm 1) = 0. \tag{1.d}$$

The solution variables W and P and the right-hand-sides r_1 and r_2 are functions of z; D represents the derivative $\partial/\partial z$; λ and k^2 are constants. (See the appendix for a derivation of this system.) A straightforward solution to this system might be to solve (1.b) to obtain P, then compute DP in order to solve (1.a) for W. Note, however, that this solution can proceed only if one knows the correct boundary conditions for P which will result in (1.c) and (1.d). Because the correct boundary conditions for P are unknown at the outset, one is faced with the following three options: (1) employ an iterative method which

0021-9991/95 $12.00

requires an initial guess at P's boundary conditions; (2) eliminate P from (1.a) and (1.b) and solve the resulting fourth-order equation for W; or (3) make use of the linearity of the system to directly solve (1.a) and (1.b) simultaneously, *i.e.*, employ the influence-matrix technique. Method 3 is the one chosen by Kleiser and Schumann for this problem and the only one I discuss here. The method begins by solving the related problems:

$$\begin{aligned} &(D^2 - \lambda)\bar{\bar{W}} = D\bar{\bar{P}} + r_1, \quad \bar{\bar{W}}(\pm 1) = 0, \\ &(D^2 - k^2)\bar{\bar{P}} = r_2, \qquad \bar{\bar{P}}(\pm 1) = 0; \end{aligned} \tag{2}$$

$$\begin{aligned} &(D^2 - \lambda)W_+ = DP_+, \qquad W_+(\pm 1) = 0, \\ &(D^2 - k^2)P_+ = 0, \qquad P_+(+1) = 1 \quad P_+(-1) = 0; \end{aligned} \tag{3}$$

$$\begin{aligned} &(D^2 - \lambda)W_- = DP_-, \qquad W_-(\pm 1) = 0, \\ &(D^2 - k^2)P_- = 0, \qquad P_-(+1) = 0 \quad P_-(-1) = 1. \end{aligned} \tag{4}$$

The final solution (W, P) is then constructed from a linear combination of these solutions:

$$\begin{pmatrix} W \\ P \end{pmatrix} = \begin{pmatrix} \bar{\bar{W}} \\ \bar{\bar{P}} \end{pmatrix} + a \begin{pmatrix} W_+ \\ P_+ \end{pmatrix} + b \begin{pmatrix} W_- \\ P_- \end{pmatrix}. \tag{5}$$

The constants a and b are chosen such that (1.d) is satisfied ((1.c) holds for *any* choice of a and b); this involves solving the 2×2 "influence" matrix problem

$$\begin{pmatrix} DW_+(+1) & DW_-(+1) \\ DW_+(-1) & DW_-(-1) \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = - \begin{pmatrix} D\bar{\bar{W}}(+1) \\ D\bar{\bar{W}}(-1) \end{pmatrix}, \tag{6}$$

hence, the method's name.

Note that (3) and (4) are symmetric counterparts,

$$W_-(z) = -W_+(-z), \quad P_-(z) = +P_+(-z); \tag{7}$$

therefore, once (3) is solved, one need not solve (4). Also note that it is equally valid to employ the alternate pressure boundary conditions

$$D\bar{\bar{P}}(+1) = 0, \quad D\bar{\bar{P}}(-1) = 0, \tag{8.a}$$

$$DP_+(+1) = 1, \quad DP_+(-1) = 0, \tag{8.b}$$

$$DP_-(+1) = 0, \quad DP_-(-1) = 1. \tag{8.c}$$

Experience shows that boundary conditions (8) produce slightly more accurate numerical solutions than the pressure boundary conditions in (2)–(4). Therefore, all of the solutions presented here are actually computed using (8).

3. CHEBYSHEV-"TAU" APPROXIMATION

When solving Eq. (1) on a digital computer, the smooth (*i.e.*, C_∞) nature of the equations is potentially lost as a result of the finite memory of the machine. In the Chebyshev-"tau" approximation, this is manifested in truncation of the Chebyshev representations for the variables W and P, *e.g.*,

$$W(z) = \sum_{j=0}^{N} W_j T_j(z),$$

where $T_j(z)$ is the jth-order Chebyshev polynomial [6] and N is finite. Naively, one might consider the "discrete" Chebyshev-"tau" representation of Eq. (1) to be

$$\left.\begin{aligned} W_j^{(2)} - \lambda W_j &= P_j^{(1)} + r_{1j} \\ P_j^{(2)} - k^2 P_j &= r_{2j} \end{aligned}\right\} \quad j \leq (N-2) \tag{9.a, 9.b}$$

$$\sum_{j=0}^{N} (\pm 1)^j W_j = 0 \tag{9.c}$$

$$\sum_{j=0}^{N} (\pm 1)^{j+1} W_j j^2 = 0, \tag{9.d}$$

where

$$D^n u(z) = \sum_{j=0}^{N} u_j^{(n)} T_j(z).$$

Equations (9.a) and (9.b) are obtained by taking the inner products of (1.a) and (1.b) with $T_j(z)$ with respect to the weight function $(1 - z^2)^{-1/2}$. Equations (9.c) and (9.d) are derived from (1.c) and (1.d) using the Chebyshev boundary values

$$T_j(\pm 1) = (\pm 1)^j, \quad DT_j(\pm 1) = (\pm 1)^{j+1} j^2.$$

Note that the two highest-mode equations ($j = N - 1, N$) are (arbitrarily) disregarded in lieu of equations for the boundary conditions (9.c) and (9.d). This replacement of the highest-mode equations with boundary conditions is precisely the "tau" approximation [2, 3].

To understand why (9) is only a "naive" attempt at discretizing the continuous system Eq. (1), one must appreciate the nature of the coupling between W and P and the repercussions this coupling has on the propagation of truncation errors from the (neglected) two highest-mode equations for W ((9.a) with $j = N - 1, N$) to all of the mode-equations for P (9.b).

To illustrate, let us denote the error incurred by neglecting (9.a) for the two highest modes $\sigma_j/2j$ ($j = N - 1, N$); *i.e.*,

$$W_j^{(2)} - \lambda W_j = P_j^{(1)} + r_{1j} - \frac{\sigma_j}{2j}, \quad j \leq N, \tag{9.a'}$$

where

$$\left(\frac{\sigma_j}{2j}\right) = \left(0, 0, 0, \cdots, 0, 0, 0, 0, \frac{\sigma_{N-1}}{2(N-1)}, \frac{\sigma_N}{2N}\right). \quad (10)$$

The $2j$-denominator is included to simplify subsequent equations. Using the following relations for truncated Chebyshev expansions

$$c_k u_k^{(2)} = \sum_{\substack{j=k+2 \\ j+k \text{ even}}}^{N} u_j j(j^2 - k^2) \rightarrow u_N^{(2)} = 0, \quad u_{N-1}^{(2)} = 0$$

$$c_k u_k^{(1)} = \sum_{\substack{j=k+1 \\ j+k \text{ odd}}}^{N} u_j 2j \qquad \rightarrow u_N^{(1)} = 0, \quad u_{N-1}^{(1)} = 2N u_N$$

($c_k = 1 + \delta_{0,k}$), we can evaluate these errors:

$$\sigma_{N-1} = 2(N-1)(r_{1N-1} + \lambda W_{N-1} + 2NP_N) \quad (11.a)$$

$$\sigma_N = 2N(r_{1N} + \lambda W_N). \quad (11.b)$$

Furthermore, because (1.a) must be differentiated to derive (1.b) (see the Appendix), we must likewise compute the discrete derivative of (9.a′) to obtain the *correct* Chebyshev-"tau" discretization of (1.b):

$$P_j^{(2)} - k^2 P_j = r_{2j} + \left(\frac{\sigma_j}{2j}\right)^{(1)}, \quad j \leq (N-2), \quad (9.b')$$

where

$$\left(\frac{\sigma_j}{2j}\right)^{(1)} = \left(\frac{\sigma_{N-1}}{2}, \sigma_N, \sigma_{N-1}, \cdots, \sigma_{N-1}, \sigma_N, \sigma_{N-1}, \sigma_N, 0\right). \quad (12)$$

Note that the errors in the two highest modes of (9.a′) propagate to *all* of the modes of (9.b′) through $(\sigma_j/2j)^{(1)}$. To obtain (12), the following Chebyshev recursion relation is useful:

$$c_j u_j^{(1)} = u_{j+2}^{(1)} + 2(j+1)u_{j+1}.$$

Therefore, the *correct* Chebyshev-"tau" discretization of Eq. (1) includes (9.a′), (9.b′), (9.c), and (9.d). I discuss the solution to this system in the next section.

4. CORRECTING THE "TAU" ERRORS

We seek a solution to the discrete system, (9.a′) and (9.b′), subject to boundary conditions (9.c) and (9.d). We employ the linearity of the system and consider separately the following two sets of equations:

$$\left.\begin{aligned} \overline{W}_j^{(2)} - \lambda \overline{W}_j &= \overline{P}_j^{(1)} + r_{1j} \quad (13.a) \\ \overline{P}_j^{(2)} - k^2 \overline{P}_j &= r_{2j} \quad (13.b) \end{aligned}\right\} j \leq (N-2)$$

$$\overline{W}(\pm 1) = D\overline{W}(\pm 1) = 0;$$

$$\left.\begin{aligned} \tilde{W}_j^{(2)} - \lambda \tilde{W}_j &= \tilde{P}_j^{(1)} \quad (14.a) \\ \tilde{P}_j^{(2)} - k^2 \tilde{P}_j &= 1/c_j \quad (14.b) \end{aligned}\right\} j \leq (N-2)$$

$$\tilde{W}(\pm 1) = D\tilde{W}(\pm 1) = 0.$$

Equation (13) is identical to the "naive" discretization (9); Eq. (14) embodies the "tau-correction" required to remove the propagated truncation errors. Each of these sets of equations is solved by the influence-matrix technique outlined in Section 2. The final solution is

$$W = \overline{W} + \beta \tilde{W}, \quad P = \overline{P} + \beta \tilde{P},$$

where β is a constant ($\ll 1$) whose magnitude is determined by the size of the correction needed to remove the "tau" errors. To compute β, we must evaluate the error in the two highest modes of (9.a′). From (9.a′), (13.a), and (14.a), these errors must satisfy

$$\sigma_j = \overline{\sigma}_j + \beta \tilde{\sigma}_j, \quad j = (N-1), N. \quad (15)$$

Also, from (12), (9.b′), (13.b), and (14.b) we have

$$\left(\frac{\sigma_j}{2j}\right)^{(1)} = \frac{\beta}{c_j}. \quad (16)$$

Combining (12), (15), and (16) we obtain β,

$$\beta = \sigma_j = \frac{\overline{\sigma}_j}{1 - \tilde{\sigma}_j},$$

which, as we see, is different for the even and odd modes. Our final solution is

$$W_j = \overline{W}_j + \begin{cases} \dfrac{\overline{\sigma}_N}{1 - \tilde{\sigma}_N} \tilde{W}_j, & j \text{ even} \\ \dfrac{\overline{\sigma}_{N-1}}{1 - \tilde{\sigma}_{N-1}} \tilde{W}_j, & j \text{ odd} \end{cases}$$

$$P_j = \overline{P}_j + \begin{cases} \dfrac{\overline{\sigma}_{N-1}}{1 - \tilde{\sigma}_{N-1}} \tilde{P}_j, & j \text{ even} \\ \dfrac{\overline{\sigma}_N}{1 - \tilde{\sigma}_N} \tilde{P}_j, & j \text{ odd,} \end{cases}$$

where $\overline{\sigma}_j$ and $\tilde{\sigma}_j$ are evaluated from (13.a) and (14.a) according to (11).

Note that this solution is a bit misleading because we initially assumed β to be constant, while we concluded with different values of β for the even and odd modes. This turns out to be consistent only because the solution to (14) decouples even and odd modes. In other words, if we re-examine the analysis, considering even- and odd-mode corrections separately, the result will be the same as above.

This completes our review of Kleiser and Schumann's numerical solution to Eq. (1). The careful reader of Kleiser and Schumann will note, however, that the two solutions (Kleiser and Schumann's and the one presented here) are *not* identical. Specifically, Kleiser and Schumann do not employ $D\tilde{W}(\pm 1) = 0$ as boundary conditions for the "tau" correction. Instead, they use $\tilde{P}(\pm 1) = 0$. The explanation for their choosing these boundary conditions lies in the fact that the influence-matrix method, as outlined in Section 2, *provides* the boundary conditions required for the pressure, namely $P(+1) = a$ and $P(-1) = b$ (using the notation of Section 2). Therefore, (the argument continues) once (13.a) and (13.b) have been solved, the correct boundary conditions for P are known and one should not alter these boundary conditions further when computing the solution $\tilde{P}$. Hence, one should choose $\tilde{P}(\pm 1) = 0$. The flaw in this reasoning is that the boundary conditions for $\overline{P}$ (obtained when (13.a) and (13.b) are solved by the influence-matrix technique) are those required to enforce $D\overline{W}(\pm 1) = 0$, not $DW(\pm 1) = 0$. Therefore, the final solution W (obtained using $\tilde{P}(\pm 1) = 0$), will not have $DW(\pm 1) = 0$ and, as a result, will have a non-zero divergence. This explains why Kleiser and Schumann's solutions satisfy $\nabla \cdot \mathbf{v} = 0$ to only a few significant digits [1] while solutions obtained using $D\tilde{W}(\pm 1) = 0$ are *exactly* divergence-free (see below).

I should point out that imposing the correct boundary condition $D\tilde{W}(\pm 1) = 0$ requires *no more computational effort* than the incorrect boundary condition $\tilde{P}(\pm 1) = 0$. The reason for this is that the solution (W_+, P_+) (see Section 2), once computed to solve (13), can be *reused* for (14).

5. RAYLEIGH–BÉNARD EXAMPLE

To demonstrate the solution outlined in this paper, Rayleigh–Bénard convection between horizontal plates held at fixed temperatures has been simulated. The domain-size is $4 \times 4 \times 1$ and the flow is periodic in the horizontal directions. The Rayleigh number is 6.4×10^5 and the Prandtl number is 1. See Table I for more details concerning the parameters used to perform the calculation. Discussion of the physical processes revealed by the simulations will appear elsewhere [7]; here I concentrate only on the magnitude of $|\nabla \cdot \mathbf{v}|$.

Figure 1 shows the time-evolution for the maximum value of $|\nabla \cdot \mathbf{v}|$, normalized by the maximum value of $|\partial_z W|$. Two different treatments of the truncation errors are used: (1) the Kleiser–Schumann solution using $\tilde{P}(\pm 1) = 0$; (2) the present formulation using $D\tilde{W}(\pm 1) = 0$, where $\mathbf{v}$ is the fluid velocity, W is the vertical component of $\mathbf{v}$, and z is the vertical coordinate. As is evident from the figure, the error in $\nabla \cdot \mathbf{v} = 0$ for the Kleiser–Schumann solution is $\approx 7 \times 10^{-3}$ (or roughly 1%) while that for the present implementation is only $\approx 7 \times 10^{-14}$. Therefore, the correction presented in this paper improves the Kleiser–Schumann solution by 11 orders of magnitude and produces velocity fields which are divergence-free to within machine round-off. (Calculations were performed using CRAY Y-MP 64-bit arithmetic.)

TABLE I

Parameters Used to Compute the Numerical Solutions

Ra	σ	Geometry	$N_x \times N_y \times N_z$	$\frac{\delta x}{l_K}$	$\frac{\delta t\, U}{\delta x}$
6.4×10^5	1	$4 \times 4 \times 1$	$96 \times 96 \times 33$	1.7	0.25

Note. Ra and σ are the Rayleigh and Prandtl numbers used for the simulations. *Geometry* refers to the dimensions of the physical domain in units of L, the depth of the fluid layer. $N_x \times N_y \times N_z$ is the number of spectral modes used in the $x \times y \times z$-directions; Fourier modes are used to represent the horizontal directions (x, y) while Chebyshev polynomials form the basis set for the vertical direction z; $\delta x/l_K$ represents the maximum value of the ratio of the grid spacing to the dissipative (or Kolmogorov) length-scale for the velocity fluctuations; values of order 1 demonstrate adequate spatial resolution of the small-scale fluctuations; $\delta t\, U/\delta x$ is the Courant–Friedrichs–Lewy (CFL) number for each *substep* used for the Runge–Kutta scheme; U is the maximum velocity at any given time.

6. CONCLUSION AND SUMMARY

A subtle error in Kleiser and Schumann's influence-matrix solution to incompressible flow between plane no-slip bound-

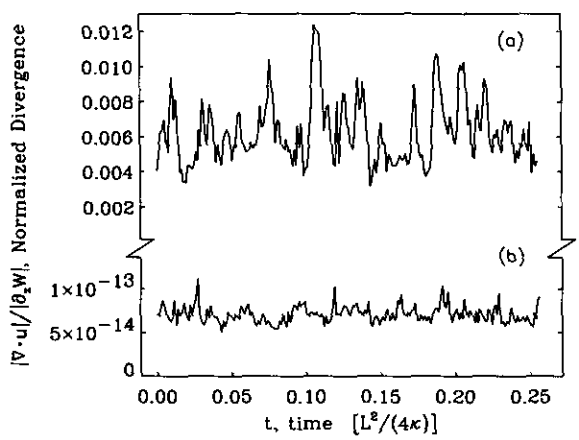

FIG. 1. Normalized divergence versus time. Curve (a) depicts the time-evolution of the maximum value of $|\nabla \cdot \mathbf{v}|$ divided by the maximum value of $|\partial_z W|$ for Boussinesq convection computed using the Kleiser–Schumann solution. Curve (b) shows the same quantity computed using the present solution. Because a Boussinesq fluid obeys $\nabla \cdot \mathbf{v} = 0$, the two curves represent the numerical errors encountered in attempting to enforce incompressibility. Note that the vertical axis is greatly expanded near the origin so that the small (round-off) errors in the present solution may be viewed.

aries is identified. A minor modification to their treatment of the truncation errors is demonstrated (via simulations of 3D Rayleigh–Bénard convection) which reduces the error in satisfying the incompressibility constraint ($\nabla \cdot \mathbf{v} = 0$) by a factor of 10^{11}, *i.e.*, to within machine precision. The modified solution requires no more computation than Kleiser and Schumann's original implementation.

APPENDIX

The Navier–Stokes equations for a Boussinesq (*i.e.*, incompressible) fluid [8] are

$$\partial_t \mathbf{v} + \boldsymbol{\omega} \times \mathbf{v} = \sigma \nabla^2 \mathbf{v} - \nabla P + \sigma \frac{\mathrm{Ra}}{16} T \hat{z} \tag{17.a}$$

$$\partial_t T + \mathbf{v} \cdot \nabla T = \nabla^2 T \tag{17.b}$$

$$\nabla \cdot \mathbf{v} = 0, \tag{17.c}$$

where $\mathbf{v}$, $\boldsymbol{\omega}$, T, and P are the nondimensional fluid velocity, vorticity, temperature, and pressure head. $\sigma = \nu/\kappa$ is the Prandtl number, where ν is the kinematic viscosity and κ is the thermal diffusivity. We seek solutions to Eq. (17) between horizontal rigid planes positioned a distance L apart. $\mathrm{Ra} = g\alpha\Delta L^3/(\nu\kappa)$ is the Rayleigh number, where Δ is the temperature drop imposed between the planes, g is the local acceleration due to gravity, and α is the fluid's thermal expansion coefficient. For no-slip, fixed-temperature boundary conditions at $z = \pm 1$ (z is normalized by $L/2$), we have

$$T(\pm 1) = \mp 1 \tag{18.a}$$

$$\mathbf{v}(\pm 1) = 0 \tag{18.b}$$

$$DW(\pm 1) = 0. \tag{18c}$$

The last condition is a consequence of (18.b) and (17.c). T is normalized by $\Delta/2$.

When numerically integrating (17) forward in time, let us follow [1] and treat the non-linear terms explicitly (*i.e.*, use values of $\boldsymbol{\omega} \times \mathbf{v}$ and $\mathbf{v} \cdot \nabla T$ at previous timesteps to extrapolate to a future timestep) and the remaining linear terms implicitly (*i.e.*, include values of $\mathbf{v}$, T, and P at the future timestep in the algorithm to advance the field variables). The particular time-stepping algorithm I use, although of this general ilk, is different from that employed by Kleiser and Schumann. I use the mixed implicit/explicit third-order Runge–Kutta scheme developed by Spalart, Rogers, and Moser [9]. This scheme, although more stable and more accurate than Kleiser and Schumann's second-order Crank–Nicolson/Adams–Bashforth scheme, requires no more storage or computational effort. In addition, the basic structure of the two schemes is identical.

The form of the implicit time-stepping algorithm for the linear terms is obtained by replacing each term in (17) with the time-discrete analogue of the latest time-level contribution for that term. Denoting time-levels by $n-2$, $n-1$, n, $n+1$, etc., we have

$$\frac{\gamma}{\delta t} \mathbf{v}^{n+1} + \mathbf{f}^n = \sigma \nabla^2 \mathbf{v}^{n+1} - \nabla P^{n+1} + \sigma \frac{\mathrm{Ra}}{16} T^{n+1} \hat{z} \tag{19.a}$$

$$\frac{\gamma}{\delta t} T^{n+1} + s^n = \nabla^2 T^{n+1} \tag{19.b}$$

$$\nabla \cdot \mathbf{v}^{n+1} = 0, \tag{19.c}$$

where γ is a constant which depends on the specific time-stepping algorithm used and δt is the timestep; $\mathbf{f}^n$ and s^n include all other terms in (19.a) and (19.b) which involve time-levels n and earlier. Rewriting (19.a) and (19.b) and taking the divergence of (19.a) (*while making use of (19.c)*), we have

$$\left(\nabla^2 - \frac{\gamma}{\sigma \delta t}\right) \mathbf{v}^{n+1} = \nabla \frac{P^{n+1}}{\sigma} - \frac{\mathrm{Ra}}{16} T^{n+1} \hat{z} + \frac{\mathbf{f}^n}{\sigma} \tag{19.a'}$$

$$\left(\nabla^2 - \frac{\gamma}{\delta t}\right) T^{n+1} = s^n \tag{19.b'}$$

$$\nabla^2 \frac{P^{n+1}}{\sigma} = -\frac{\nabla \cdot \mathbf{f}^n}{\sigma} + \frac{\mathrm{Ra}}{16} D T^{n+1}. \tag{19.c'}$$

Furthermore, if the horizontal representation is Fourier, *e.g.*,

$$\mathbf{v} = \sum_{\mathbf{k}} \mathbf{v}_{\mathbf{k}}(z) e^{-i\mathbf{k}\cdot\mathbf{x}},$$

where $\mathbf{k} = (k_x, k_y, 0)$ is the horizontal wave vector and $\mathbf{x} = (x, y, z)$ is the position vector, (19′) takes the form

$$(D^2 - \eta) T_{\mathbf{k}}^{n+1} = s_{\mathbf{k}}^n \tag{20.a}$$

$$(D^2 - \lambda) U_{\mathbf{k}}^{n+1} = -ik_x \frac{P_{\mathbf{k}}^{n+1}}{\sigma} + \frac{f_{1\mathbf{k}}^n}{\sigma} \tag{20.b}$$

$$(D^2 - \lambda) V_{\mathbf{k}}^{n+1} = -ik_y \frac{P_{\mathbf{k}}^{n+1}}{\sigma} + \frac{f_{2\mathbf{k}}^n}{\sigma} \tag{20.c}$$

$$(D^2 - \lambda) W_{\mathbf{k}}^{n+1} = D \frac{P_{\mathbf{k}}^{n+1}}{\sigma} + \frac{f_{3\mathbf{k}}^n}{\sigma} - \frac{\mathrm{Ra}}{16} T_{\mathbf{k}}^{n+1} \tag{20.d}$$

$$(D^2 - k^2) \frac{P_{\mathbf{k}}^{n+1}}{\sigma} = -\left(\frac{\nabla \cdot \mathbf{f}^n}{\sigma}\right)_{\mathbf{k}} + \frac{\mathrm{Ra}}{16} D T_{\mathbf{k}}^{n+1}, \tag{20.e}$$

$k^2 = \mathbf{k} \cdot \mathbf{k}$, $\eta = k^2 + \gamma/\delta t$, and $\lambda = k^2 + \gamma/(\sigma \delta t)$. The technique required to solve this system begins by solving the linear Helmholtz equation for $T_{\mathbf{k}}^{n+1}$ (20.a) subject to boundary conditions (18.a). Then, with the temperature at the future time known, the "right-hand sides" to (20.d) and (20.e) can be computed. This results in a subsystem of the form:

$$(D^2 - \lambda)W_k^{n+1} = \frac{DP_k^{n+1}}{\sigma} + r_{1k}$$

$$(D^2 - k^2)\frac{P_k^{n+1}}{\sigma} = r_{2k}$$

$$W_k^{n+1}(\pm 1) = DW_k^{n+1}(\pm 1) = 0.$$

The solution to this subsystem can be obtained using the influence-matrix technique, which is the subject of the body of this paper. Once this subsystem is solved, the right-hand sides to (20.b) and (20.c) can be evaluated and those equations solved for U_k^{n+1} and V_k^{n+1}. The solution at later timesteps is obtained by repeating the procedure.

ACKNOWLEDGMENTS

I gladly acknowledge useful discussions with Peter Sullivan and Laurette Tuckerman and I thank Nic Brummell and Keith Julien for carefully reading the manuscript. I also thank Fausto Cattaneo and Ed DeLuca for teaching me the influence-matrix technique. NCAR is sponsored by the National Science Foundation.

REFERENCES

1. L. Kleiser and U. Schumann, in *Proceedings, 3rd GAMM Conf. on Numerical Methods in Fluid Mechanics*, edited by E. H. Hirschel (Vieweg, Braunschweig, 1980), p. 165.
2. D. Gottlieb and S. A. Orszag, *Numerical Analysis of Spectral Methods for Partial Differential Equations* (SIAM, Philadelphia, 1977).
3. C. Canuto, M. Y. Hussaini, A. Quateroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics* (Springer-Verlag, Berlin, 1987).
4. B. L. Buzbee, F. W. Dorr, J. A. George, and G. H. Golub, *SIAM J. Numer. Anal.* **8,** 722 (1971).
5. L. S. Tuckerman, *J. Comput. Phys.* **80,** 403 (1989).
6. M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions* (Dover, New York, 1972).
7. K. Julien, S. Legg, J. McWilliams, and J. Werne, in preparation.
8. S. Chandrasekhar, *Hydrodynamic and Hydromagnetic Stability* (Oxford Univ. Press, Oxford, 1961).
9. P. R. Spalart, R. D. Moser, and M. M. Rogers, *J. Comput. Phys.* **96,** 297 (1991).