Journal of Scientific Computing, Vol. 18, No. 2, April 2003 (© 2003)

A Fourth Order Scheme for Incompressible Boussinesq Equations

Jian-Guo Liu,[1] Cheng Wang,[2] and Hans Johnston[3]

Received February 15, 2001; accepted (in revised form) April 4, 2002

A fourth order finite difference method is presented for the 2D unsteady viscous incompressible Boussinesq equations in vorticity-stream function formulation. The method is especially suitable for moderate to large Reynolds number flows. The momentum equation is discretized by a compact fourth order scheme with the no-slip boundary condition enforced using a local vorticity boundary condition. Fourth order long-stencil discretizations are used for the temperature transport equation with one-sided extrapolation applied near the boundary. The time stepping scheme for both equations is classical fourth order Runge–Kutta. The method is highly efficient. The main computation consists of the solution of two Poisson-like equations at each Runge–Kutta time stage for which standard FFT based fast Poisson solvers are used. An example of Lorenz flow is presented, in which the full fourth order accuracy is checked. The numerical simulation of a strong shear flow induced by a temperature jump, is resolved by two perfectly matching resolutions. Additionally, we present benchmark quality simulations of a differentially-heated cavity problem. This flow was the focus of a special session at the first MIT conference on *Computational Fluid and Solid Mechanics* in June 2001.

KEY WORDS: Boussinesq equations; incompressible flow; compact scheme; long-stencil approximation; one-sided extrapolation; vorticity boundary condition; Kelvin–Helmholtz instability.

[1] Institute for Physical Science and Technology and Department of Mathematics, University of Maryland, College Park, Maryland 20742.

[2] Institute for Scientific Computing and Applied Mathematics and Department of Mathematics, Indiana University, Bloomington, Indiana 47405. E-mail: cwang@indiana.edu

[3] Department of Mathematics, University of Michigan, Ann Arbor, Michigan 48109.

0885-7474/03/0400-0253/0

1. INTRODUCTION

The dimensionless form of the Boussinesq approximation for the 2D incompressible Navier–Stokes equations on a domain Ω is given by

$$\begin{cases} \partial_t \theta + (\mathbf{u}\cdot\nabla)\,\theta = \dfrac{1}{Re\cdot Pr}\,\Delta\theta, \\ \partial_t \mathbf{u} + (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \nabla p = \dfrac{1}{Re}\,\Delta\mathbf{u} + Ri\cdot\theta\cdot\begin{pmatrix}0\\1\end{pmatrix}, \\ \nabla\cdot\mathbf{u} = 0, \end{cases} \tag{1.1}$$

where $\mathbf{u} = (u, v)^t$ is the velocity, p the pressure, θ the temperature, and Re the Reynolds number. Pr is the Prandtl number, the ratio of the kinematic viscosity to the heat conductivity. The Richardson number Ri accounts for the gravitational force and the thermal expansion of the fluid. One may introduce other physically relevant dimensionless quantities, such as the Rayleigh number $Ra = Ri\cdot Re^2\cdot Pr$, and the Grashof number $Gr = Ra/Pr = Ri\cdot Re^2$. For brevity of presentation we denote $\nu = 1/Re$ and $\kappa = 1/(Re\cdot Pr)$.

We consider the simplest boundary conditions for $\mathbf{u}$, the no-penetration, no-slip conditions $\mathbf{u}|_\Gamma = 0$. For the temperature θ we can impose either the Dirichlet boundary condition

$$\theta|_\Gamma = \theta_b, \tag{1.2}$$

where θ_b is a given temperature distribution on the boundary Γ of Ω, or a Neumann boundary condition

$$\left.\frac{\partial\theta}{\partial\mathbf{n}}\right|_\Gamma = \theta_f, \tag{1.3}$$

where θ_f is a given heat flux. For example, $\theta_f = 0$ corresponds to an adiabatic boundary condition.

In 2D the vorticity formulation of (1.1) is computationally advantageous for it eliminates the pressure variable and automatically enforces incompressibility. Introducing the vorticity $\omega = \nabla\times\mathbf{u}$ and the stream function ψ, (1.1) is equivalent to

$$\begin{cases} \partial_t\theta + (\mathbf{u}\cdot\nabla)\,\theta = \kappa\,\Delta\theta, \\ \partial_t\omega + (\mathbf{u}\cdot\nabla)\,\omega = \nu\,\Delta\omega + Ri\,\partial_x\theta, \\ \Delta\psi = \omega, \\ u = -\partial_y\psi, \qquad v = \partial_x\psi. \end{cases} \tag{1.4}$$

For a simply-connected domain, the case considered here, the no penetration, no-slip boundary conditions $\mathbf{u}|_\Gamma = 0$ are recast in terms of ψ as

$$\psi = 0, \qquad \frac{\partial \psi}{\partial \mathbf{n}} = 0. \tag{1.5}$$

The boundary conditions (1.5) for the stream function have been extensively studied in the context of numerical solutions of the Navier–Stokes equations, cf. [9, 14, 8, 15, 10]. Most noteworthy of the previous work in the numerical simulation of incompressible Boussinesq equations (1.4) can be found in [1, 10, 6].

In this paper a fourth order numerical method based on the vorticity formulation (1.2)–(1.5) is presented. A compact discretization, proposed by E and Liu [5] for the 2D Navier–Stokes equations (NSE), is used to solve the momentum equation in (1.4), with the gravity term treated explicitly in time. A compact approach provides high order accuracy while avoiding extrapolation in order to prescribe "ghost" computational grid points for the vorticity along the boundary. This is important for large Reynolds number flows where, in general, the viscous boundary layer is highly singular. Moreover, a detailed stability and convergence analysis of the fourth order compact approach [21, 22] shows it to introduce less numerical dissipation than standard high order centered difference schemes. In contrast, a compact approach is not indicated for the temperature transport equation. Indeed, the prescribed boundary condition for temperature, (1.2) or (1.3), allows the solution of the temperature equation to fourth order using long-stencil approximations. This avoids the computational cost of solving a Poisson-like equation involving an auxiliary temperature variable required of a compact approach. This necessitates temperature data at "ghost" points outside of the computational domain, which is prescribed using one-sided extrapolation. The number of interior points in these formulas is reduced by applying information obtained from the temperature equation at the boundary. Similar ideas can be found in [9]. We note that this approach for the temperature transport equation can be applied to any type of passive scalar equation. The stability of the high order long-stencil approximation with one-sided extrapolation is demonstrated numerically in Secs. 3 and 4, and theoretically in a forthcoming article [23].

The time discretization scheme for both the momentum and temperature equations is classical fourth order Runge–Kutta (RK4). In conjunction with the fourth order spatial discretization, the resulting multi-stage explicit time stepping procedure is simple to implement and highly efficient. The main computation is the solution of two Poisson-like equations per stage, which are solved using FFT based schemes. Additionally, the choice of

RK4 avoids stability restrictions in the form of cell-Reynolds number constraints. The numerical scheme, including the implementation of the time-stepping procedure, is described in Sec. 2.

The paper is organized as follows. In Sec. 3 a well known model of Rayleigh–Bénard type convection is used to verify the accuracy of our numerical scheme. The stream function and temperature in the 2D flow are represented by three parameters with the help of single and double mode analyses. The evolution of these parameters is described by a nonlinear system of ODEs, the Lorenz system. An accuracy check is carried out for our computational method applied to the Boussinesq equations (with a single source term) based on the Lorenz system, demonstrating fourth order accuracy of the method.

In Sec. 4.1 we demonstrate the robustness of our numerical method by simulating an example of strong shear flow induced by a temperature jump, with ratio 1.5:1, in an insulated box. (Similar experimental work on stratified shear flow has been carried out by Thorpe [18, 19]). The resulting roll-up structure, triggered by a Kelvin–Helmholtz instability of the vortex sheet, is completely resolved. The accuracy of our computation is verified by the excellent agreement between the two grid resolutions, 2049×257 and 4097×513.

As further evidence of the accuracy and efficiency of our method we present in Sec. 4.2 benchmark quality simulations for a differentially-heated cavity problem. The computation of this flow was the focus of a special session at the first MIT conference on *Computational Fluid and Solid Mechanics* in June 2001. Submissions to the session included simulations computed using finite difference, finite element, finite volume, and spectral methods. The reference benchmark simulation was computed using a spectral code [20], which was used to rank the submissions to the special session. In all there were six composite metrics on which submissions were judged. The simulation computed by our method received *three* first place rankings and *one* second place ranking. In particular, with respect to numerical accuracy and efficiency our method performed extremely well.

2. DESCRIPTION OF THE SCHEME

For simplicity of presentation the computational domain is taken as $\Omega = [0, 1] \times [0, 1]$ with grid size $\Delta x = \Delta y = h$. The boundary Γ is composed of Γ_x: $\{y = 0, 1\}$ and Γ_y: $\{x = 0, 1\}$. The numerical grid is denoted by $\Omega_h = \{x_i = i/N,\ y_j = j/N,\ i, j = 0, 1, \ldots, N\}$. The spatial discretizations for (1.4) are first outlined, with the complete time stepping of the system (1.4) described in Sec. 2.5.

2.1. Temperature Transport Equation

The temperature transport equation is treated as a standard convection-diffusion equation. It is discretized using fourth order long-stencil difference operators, which requires temperature values to be defined at "ghost" grid points outside of the computational domain. They are prescribed using one-sided extrapolation. To reduce the number of interior points required in the extrapolation, for both computational convenience and better stability, we apply information obtained from the temperature equation on the boundary. Similar ideas can be found in [9].

To begin, standard fourth order centered long-stencil approximations of ∂_x and ∂_y are given by, respectively,

$$\partial_x = \tilde{D}_x \left(1 - \frac{h^2}{6} D_x^2\right) + O(h^4), \qquad \partial_y = \tilde{D}_y \left(1 - \frac{h^2}{6} D_y^2\right) + O(h^4), \tag{2.1}$$

where $\tilde{D}_x$ and D_x^2 are standard second order centered difference approximations to ∂_x and ∂_x^2, respectively. To approximate Δ in the diffusion term, note that

$$\Delta = \Delta_h - \frac{h^2}{12} (D_x^4 + D_y^4) + O(h^4), \tag{2.2}$$

where $\Delta_h = D_x^2 + D_y^2$. The temperature equation is then discretized using

$$\partial_t \theta + u \tilde{D}_x \left(1 - \frac{h^2}{6} D_x^2\right) \theta + v \tilde{D}_y \left(1 - \frac{h^2}{6} D_y^2\right) \theta = \kappa \left(\Delta_h - \frac{h^2}{12} (D_x^4 + D_y^4)\right) \theta. \tag{2.3}$$

2.2. Temperature at Ghost Point(s)

Determination of θ at "ghost" points is needed at each boundary to implement (2.3), one point in the case of Dirichlet boundary conditions (1.2) and two points in the case of the Neumann boundary condition (1.3). Improved stability generally results by using as few interior points, for a given order approximation, as possible in the one-sided stencils. For brevity of the presentation we concentrate on the boundary section Γ_x where $j = 0$.

2.2.1. Dirichlet Boundary Condition for Temperature

In the case of a Dirichlet boundary condition (1.2) for the temperature, $\theta_{i,0}$ is given on the boundary by $\theta_b(x_i, 0)$. The temperature in (2.3) is then updated at interior points (x_i, y_j), $1 \leqslant i,\ j \leqslant N-1$, requiring the

prescription of the "ghost" value $\theta_{i,-1}$ due to the stencil of the discretization (2.3). Local Taylor expansion near the boundary gives

$$\theta_{i,-1} = 2\theta_{i,0} - \theta_{i,1} + h^2\, \partial_y^2 \theta_{i,0} + O(h^4). \tag{2.4}$$

To apply (2.4) the term $\partial_y^2 \theta$ for $j = 0$ is prescribed by considering the temperature transport equation at the boundary:

$$\partial_t \theta|_{\Gamma_x} = \kappa\, \Delta\theta|_{\Gamma_x} = \kappa(\partial_x^2 + \partial_y^2)\, \theta|_{\Gamma_x} = \kappa(\partial_x^2 \theta_b + \partial_y^2 \theta|_{\Gamma_x}). \tag{2.5}$$

The first equality follows from the convection term being identically zero along Γ due to the boundary condition $\mathbf{u}|_\Gamma = 0$, and we have

$$\partial_y^2 \theta|_{\Gamma_x} = \frac{1}{\kappa}\, \partial_t \theta_b - \partial_x^2 \theta_b, \tag{2.6}$$

where the right hand side is a known since $\theta = \theta_b$ is given on Γ. Substituting (2.6) into (2.4), we have

$$\theta_{i,-1} = 2\theta_{i,0} - \theta_{i,1} + h^2 \left(\frac{1}{\kappa}\, \partial_t \theta_b - \partial_x^2 \theta_b \right) + O(h^4), \tag{2.7}$$

Analogous derivations apply on the other three boundary sections. It can be shown that this formula gives full fourth order accuracy. See the results in Table I.

Remark 2.1. In (2.4) we used a fourth order one-sided approximation for the temperature near the boundary. In fact, a fifth order Taylor expansion near the boundary can also be used, which gives

$$\theta_{i,-1} = \frac{20}{11}\, \theta_{i,0} - \frac{6}{11}\, \theta_{i,1} - \frac{4}{11}\, \theta_{i,2} + \frac{1}{11}\, \theta_{i,3} + \frac{12}{11}\, h^2\, \partial_y^2 \theta_{i,0} + O(h^5). \tag{2.8}$$

The derivation of $\partial_y^2 \theta_{i,0}$ on Γ_x, (2.5) and (2.6), is unchanged. The combination of (2.8) and (2.6) results in

$$\theta_{i,-1} = \frac{20}{11}\, \theta_{i,0} - \frac{6}{11}\, \theta_{i,1} - \frac{4}{11}\, \theta_{i,2} + \frac{1}{11}\, \theta_{i,3} + \frac{12}{11}\, h^2 \left(\frac{1}{\kappa}\, \partial_t \theta_b - \partial_x^2 \theta_b \right) + O(h^5). \tag{2.9}$$

which is a $O(h^5)$ formula analogous to (2.7). Our computation shows that both (2.7) and (2.9) provide stability and full accuracy, as explained in Sec. 3. Since the formula (2.7) requires only one interior point, we suggest

using (2.7) in practical computation for convenience. However, for technical considerations [23] in the stability analysis of the overall scheme, the fifth order approximation is preferred.

2.2.2. *Neumann Boundary Condition for Temperature*

In the case of the Neumann boundary condition (1.3) the temperature on the boundary is not known explicitly, only its normal derivative. Thus, (2.3) is applied at every computational point (x_i, y_j), $0 \leqslant i, j \leqslant N$ requiring us to determine two "ghost" point values, $\theta_{i,-1}$ and $\theta_{i,-2}$, to carry out (2.3). As in the Dirichlet case above we begin by deriving one-sided approximations. Local Taylor expansion near the boundary gives

$$\theta_{i,-1} = \theta_{i,1} - 2h\,\partial_y\theta_{i,0} - \frac{h^3}{3}\,\partial_y^3\theta_{i,0} + O(h^5), \tag{2.10}$$

and

$$\theta_{i,-2} = \theta_{i,2} - 4h\,\partial_y\theta_{i,0} - \frac{8h^3}{3}\,\partial_y^3\theta_{i,0} + O(h^5). \tag{2.11}$$

The term $\partial_y\theta_{i,0}$ in (2.10) and (2.11) is known from the flux boundary condition (1.3). It remains to determine $\partial_y^3\theta_{i,0}$, for which we use information from the PDE and its derivatives. Applying ∂_y to the temperature equation along Γ_x gives

$$\theta_{yt} + u_y\theta_x + u\theta_{xy} + v_y\theta_y + v\theta_{yy} = \kappa(\theta_{yxx} + \partial_y^3\theta), \qquad \text{on } \Gamma_x. \tag{2.12}$$

The first term on the left-hand side as well as the first term on the right-hand side of (2.12) are known functions, θ_{ft} and θ_{fxx}, respectively. The third and fifth terms on the left-hand side are zero since $\mathbf{u}|_\Gamma = 0$. The fourth term on the left-hand side is also zero because of the no-slip boundary condition and incompressibility, i.e., $v_y = -u_x = 0$ on Γ_x. It remains to determine the second term on the left-hand side. Since $v_x = 0$ along Γ_x it follows that $u_y = -(v_x - u_y) = -\omega$ along Γ_x. Moreover, since (2.3) is updated at all grid points, including the boundary points, θ_x on Γ_x can be calculated by the standard fourth order long-stencil formula (2.1). Combining all these arguments and substituting them back into (2.12), $\partial_y^3\theta$ is approximated along Γ_x by

$$\partial_y^3\theta_{i,0} = \frac{1}{\kappa}\left(\theta_{ft} - \omega_{i,0}\tilde{D}_x\left(1 - \frac{h^2}{6}D_x^2\right)\theta_{i,0}\right) - \theta_{fxx}. \tag{2.13}$$

Plugging (2.13) back into (2.10) and (2.11), we have

$$\theta_{i,-1} = \theta_{i,1} - 2h\theta_f - \frac{h^3}{3}\left(\frac{1}{\kappa}\theta_{ft} - \frac{1}{\kappa}\omega_{i,0}\tilde{D}_x\left(1 - \frac{h^2}{6}D_x^2\right)\theta_{i,0} - \theta_{fxx}\right), \tag{2.14}$$

$$\theta_{i,-2} = \theta_{i,2} - 4h\theta_f - \frac{8h^3}{3}\left(\frac{1}{\kappa}\theta_{ft} - \frac{1}{\kappa}\omega_{i,0}\tilde{D}_x\left(1 - \frac{h^2}{6}D_x^2\right)\theta_{i,0} - \theta_{fxx}\right). \tag{2.15}$$

In the no-flux (or fixed-flux) case we have $\theta_{ft} = \theta_{fxx} = 0$, and the formulae reduce to

$$\theta_{i,-1} = \theta_{i,1} + \frac{h^3}{3}\frac{\omega_{i,0}}{\kappa}\tilde{D}_x\left(1 - \frac{h^2}{6}D_x^2\right)\theta_{i,0}, \tag{2.16}$$

$$\theta_{i,-2} = \theta_{i,2} + \frac{8h^3}{3}\frac{\omega_{i,0}}{\kappa}\tilde{D}_x\left(1 - \frac{h^2}{6}D_x^2\right)\theta_{i,0}. \tag{2.17}$$

Analogous formulas follow for the remaining three boundaries.

The above one-sided approximations of the temperature near the boundary is can be shown to be stable, the analysis of which will appear in a forthcoming article [23].

2.3. Momentum Equation

The momentum equation in (1.4) is discretized using the Essentially Compact Fourth Order Scheme (EC4) approach proposed by E and Liu in [5]. By the introduction of an auxiliary vorticity variable the momentum equation is updated by a compact scheme. The kinematic equation between stream function and vorticity is also discretized by fourth-order compact differencing, giving rise to a discrete Poisson-like equation for the stream function. This is solved using the Dirichlet no-penetration boundary condition $\psi|_\Gamma = 0$. The no-slip boundary condition $\frac{\partial\psi}{\partial\mathbf{n}}|_\Gamma = 0$, along with a one-sided approximation of $\omega = \Delta\psi$, is converted into a local vorticity boundary condition such as Briley's formula or a new fourth order vorticity boundary formula proposed in [22]. The vorticity field is then recovered by solving a Poisson-like equation. Moreover, the main difference between the momentum equation in Boussinesq equations and the usual fluid equations is the addition of the gravity term, which is also treated with a fourth order discretization and explicitly in the time stepping procedure.

The starting point of the scheme is a fourth order compact discretization of the Laplacian operator Δ given by

$$\Delta = \frac{\Delta_h + \frac{h^2}{6}D_x^2 D_y^2}{1 + \frac{h^2}{12}\Delta_h} + O(h^4). \tag{2.18}$$

Substituting (2.18) into the momentum equation, and then multiplying the result by the denominator of the operator in (2.18), namely $1+\frac{h^2}{12}\Delta_h$, gives

$$\left(1+\frac{h^2}{12}\Delta_h\right)\partial_t\omega+\left(1+\frac{h^2}{12}\Delta_h\right)\nabla\cdot(\mathbf{u}\omega)-Ri\left(1+\frac{h^2}{12}\Delta_h\right)\partial_x\theta$$

$$=\nu\left(\Delta_h+\frac{h^2}{6}D_x^2D_y^2\right)\omega+O(h^4). \tag{2.19}$$

Applying the same procedure to the kinematic equation leads to

$$\left(\Delta_h+\frac{h^2}{6}D_x^2D_y^2\right)\psi=\left(1+\frac{h^2}{12}\Delta_h\right)\omega+O(h^4). \tag{2.20}$$

Incompressibility implies that $\nabla\cdot(\mathbf{u}\omega)=(\mathbf{u}\cdot\nabla\omega)$, and the nonlinear convection is estimated as

$$\left(1+\frac{h^2}{12}\Delta_h\right)(\mathbf{u}\cdot\nabla\omega)=\tilde{D}_x\left(1+\frac{h^2}{6}D_y^2\right)(u\omega)+\tilde{D}_y\left(1+\frac{h^2}{6}D_x^2\right)(v\omega)$$

$$-\frac{h^2}{12}\Delta_h(u\tilde{D}_x\omega+v\tilde{D}_y\omega)+O(h^4). \tag{2.21}$$

The first and the second terms in (2.21) are compact. The third term is not, but this does not cause any difficulties computationally since the boundary condition $\mathbf{u}|_\Gamma=0$ implies that $u\tilde{D}_x\omega+v\tilde{D}_y\omega$ can be taken as 0 on Γ. The gravity term is dealt with similarly, and a formal Taylor expansion gives

$$\left(1+\frac{h^2}{12}\Delta_h\right)\partial_x=\tilde{D}_x\left(1+\frac{h^2}{12}D_y^2-\frac{h^2}{12}D_x^2\right)+O(h^4)$$

$$=\tilde{D}_x+\frac{h^2}{12}\tilde{D}_xD_y^2-\frac{h^2}{12}\tilde{D}_xD_x^2+O(h^4). \tag{2.22}$$

Note that the third term requires "ghost" point values for θ, whose prescription was discussed in the last subsection. Then introducing the auxiliary variable $\bar{\omega}$ defined by

$$\bar{\omega}=\left(1+\frac{h^2}{12}\Delta_h\right)\omega, \tag{2.23}$$

and combining (2.19)–(2.22), the momentum equation is discretized by

$$\partial_t \bar{\omega} + \tilde{D}_x \left(1 + \frac{h^2}{6} D_y^2\right)(u\omega) + \tilde{D}_y \left(1 + \frac{h^2}{6} D_x^2\right)(v\omega) - \frac{h^2}{12} \Delta_h (u\tilde{D}_x \omega + v\tilde{D}_y \omega) - Ri\, \tilde{D}_x \left(1 + \frac{h^2}{12}(D_y^2 - D_x^2)\right)\theta = \nu \left(\Delta_h + \frac{h^2}{6} D_x^2 D_y^2\right)\omega. \tag{2.24}$$

The stream function is recovered to fourth order given $\bar{\omega}$ by solving (2.20) with the Dirichlet boundary condition $\psi|_\Gamma = 0$. The velocity $\mathbf{u} = \nabla^\perp \psi = (-\partial_y \psi, \partial_x \psi)$ is recovered using (2.1), long-stencil approximations to ∂_x and ∂_y, respectively,

$$u = -\tilde{D}_y \left(1 - \frac{h^2}{6} D_y^2\right)\psi, \qquad v = \tilde{D}_x \left(1 - \frac{h^2}{6} D_x^2\right)\psi, \tag{2.25}$$

along with the enforcement of $\mathbf{u}|_\Gamma = 0$. However, note that (2.25) requires value of ψ at one "ghost" point to compute a derivative normal to a given boundary at the first interior point, e.g., u with $j = 1$ along Γ_x. This prescription is discussed in the next subsection in conjunction with the derivation of a local vorticity boundary condition for ω required to determined ω from $\bar{\omega}$ via (2.23).

Local Vorticity Boundary Conditions

The subject of the vorticity boundary condition in the context of finite difference schemes has a long history, going back at least to the 1930s when Thom's formula was derived [17]. Physically the vorticity boundary condition enforces the no-slip boundary condition for the velocity, which in turn determines the structure of the viscous boundary layer. Thus, it is essential that the boundary layer is resolved for it eventually separates and the vortical structures generated from the separation drastically influence the overall flow.

Local vorticity boundary conditions are derived from discretizations of the the kinematic relation $\omega = \Delta\psi$ at the boundary, requiring numerical values of ψ at "ghost" points outside of the computational domain, which are derived from discretizations of $\frac{\partial \psi}{\partial \mathbf{n}} = 0$. It is in this way that the boundary condition for vorticity enforces the no-slip boundary condition for the velocity. A detailed discussion of the derivation and use of local vorticity boundary conditions can be found in [5].

On the boundary section Γ_x, where $j = 0$, Briley's formula reads

$$\omega_{i,0} = \frac{1}{18h^2}(108\psi_{i,1} - 27\psi_{i,2} + 4\psi_{i,3}). \tag{2.26}$$

The corresponding "ghost" point values for stream function are given by

$$\psi_{i,-1} = 6\psi_{i,1} - 2\psi_{i,2} + \frac{1}{3}\psi_{i,3} - 4h\left(\frac{\partial\psi}{\partial y}\right)_{i,0} + O(h^5), \tag{2.27}$$

$$\psi_{i,-2} = 40\psi_{i,1} - 15\psi_{i,2} + \frac{8}{3}\psi_{i,3} - 12h\left(\frac{\partial\psi}{\partial y}\right)_{i,0} + O(h^5). \tag{2.28}$$

One-sided approximation and high order Taylor expansion for stream function around the boundary was used in the derivation of these local formulae. Briley's formula was initially proposed in [2] and its use in the EC4 scheme was analyzed in [5, 21].

A new fourth order formula for the vorticity on the boundary, which was proposed in [22], gives

$$\omega_{i,0} = \frac{1}{h^2}\left(8\psi_{i,1} - 3\psi_{i,2} + \frac{8}{9}\psi_{i,3} - \frac{1}{8}\psi_{i,4}\right), \tag{2.29}$$

along with the corresponding one-sided approximation for stream function at "ghost" points

$$\psi_{i,-1} = 10\psi_{i,1} - 5\psi_{i,2} + \frac{5}{3}\psi_{i,3} - \frac{1}{4}\psi_{i,4} - 5h\left(\frac{\partial\psi}{\partial y}\right)_{i,0} + O(h^6), \tag{2.30}$$

$$\psi_{i,-2} = 80\psi_{i,1} - 45\psi_{i,2} + 16\psi_{i,3} - \frac{5}{2}\psi_{i,4} - 30h\left(\frac{\partial\psi}{\partial y}\right)_{i,0} + O(h^6). \tag{2.31}$$

It was shown in [21, 22] that the above one-sided vorticity boundary condition preserves stability and is consistent with the centered difference applied at interior points. Both formulae give us fourth order accuracy for 2-D Navier–Stokes equations. For computational convenience, we suggest using Briley's formula along with (2.27), (2.28) in the calculation.

2.4. Motivation for Using Different Solvers for the Momentum and Temperature Equations

Let us review our scheme for the Boussinesq equations. Due to the singular behavior of vorticity near the boundary at high Reynolds numbers a compact approach is necessary for the momentum equation in order to achieve fourth order accuracy while at the same time avoiding possible stability issues arising from the use of long-stencil approximations near the boundary. The price paid is the introduction of an auxiliary vorticity variable. However, it only requires updating in time at the interior grid points. Accordingly, the stream function can be solved by the Poisson-like equation (2.20) using Dirichlet boundary condition. Then given the stream function we can calculate the vorticity on the boundary by a local formula, either (2.26) or (2.29), which enforces the no-slip boundary condition. Consequently, the vorticity field is obtained by solving the linear system (2.23), which is also a Poisson-like equation. The velocity field is determined by the long-stencil formula (2.25). Finally, we update the auxiliary vorticity at the next time stage, using finite differencing in (2.24).

In contrast, a compact approach is not indicated for the computation of the temperature transport equation. Alternatively, long-stencil centered fourth-order approximation of the spatial derivatives can be used due the fact that temperature variable, a passive scalar, is much smoother than the vorticity variable in the boundary layer. The well-defined boundary condition for temperature, either (1.2) or (1.3), along with the technique of using information from the original PDE on the boundary, helps us to formulate a stable one-sided approximation. This approach does not require any linear system solvers, unlike the two Poisson solvers for the momentum equation. Therefore, it simplifies the computation without any loss of accuracy in the temperature.

Physically, the vorticity profile near the boundary for large Reynolds number flow is generally structurally complicated, with a thickness of $\delta = O(1/\sqrt{Re})$ and amplitude $O(\sqrt{Re})$ in vorticity, by formal asymptotic expansion, with possibly a non-monotone profile. In contrast, the magnitude of the temperature near the boundary is $O(1)$. Thus, the primary numerical challenge lies in capturing the detailed structures of the vorticity in the boundary layer. The vorticity boundary condition and the one-sided extrapolation for the temperature in our scheme, both of which are local formulae near the boundary, result in a methodology capable of resolving such complex flows. Moreover, the stability and accuracy of the formulae make them very robust in practical calculations, especially at high Reynolds numbers.

2.5. Time Discretization

The classical fourth order Runge–Kutta method is used for the time discretization of both the momentum equation and the temperature equation. The multi-stage explicit time stepping procedure makes the fourth order spatial discretization very easy to implement.

The convection, diffusion terms and the gravity term appearing in the Boussinesq equations, together with the fourth order spatial discretizations discussed above, are updated explicitly. Such explicit treatment avoids any stability concern caused by the cell-Reynolds number constraint if a high order Runge–Kutta method, such as classical RK3 or RK4, is applied. This observation was first made in [4] and its extension to EC4 scheme was documented in [5]. For the sake of fourth order accuracy, RK4 is used in our method to update both the momentum and temperature equations. The multi-stage explicit scheme circumvents the long-standing difficulty of the coupling among the momentum equation, kinematic equation, and the vorticity boundary condition.

Only two standard Poisson-like equations, in Steps 3 and 6 described later, are required to be solved at each Runge–Kutta time stage, for which fast FFT-based methods are used. Our numerical experiments show that approximately 90 percent of the CPU is spent in the two Poisson solvers. That makes the method extremely efficient.

For simplicity we only present the forward Euler time-discretization. The extension to Runge–Kutta method is straightforward.

Initialization. Given $\{\omega_{ij}^0\}$, compute

$$\left(1+\frac{h^2}{12}\,\Delta_h\right)\omega^0=\bar{\omega}^0. \tag{2.32}$$

Time-Stepping. Given the vorticity ω^n and the temperature θ^n at time t^n, we compute all the profiles at the time step t^{n+1} via the following steps.

Step 1. Update $\{\bar{\omega}_{i,j}^{n+1}\}$, at interior points (x_i, y_j), for $1\leqslant i, j\leqslant N-1$, using

$$\begin{aligned}\frac{\bar{\omega}^{n+1}-\bar{\omega}^n}{\Delta t}&+\tilde{D}_x\left(1+\frac{h^2}{6}D_y^2\right)(u^n\omega^n)+\tilde{D}_y\left(1+\frac{h^2}{6}D_x^2\right)(v^n\omega^n)\\&-\frac{h^2}{12}\Delta_h(u^n\tilde{D}_x\omega^n+v^n\tilde{D}_y\omega^n)-Ri\;\tilde{D}_x\left(1+\frac{h^2}{12}(D_y^2-D_x^2)\right)\theta^n\\&=\nu\left(\Delta_h+\frac{h^2}{6}D_x^2D_y^2\right)\omega^n.\end{aligned} \tag{2.33}$$

Step 2. Obtain $\theta_{i,j}^{n+1}$ using

$$\frac{\theta^{n+1}-\theta^n}{\Delta t}+u^n\tilde{D}_x\left(1-\frac{h^2}{6}D_x^2\right)\theta^n+v^n\tilde{D}_y\left(1-\frac{h^2}{6}D_y^2\right)\theta^n$$
$$=\kappa\left(\Delta_h-\frac{h^2}{12}(D_x^4+D_y^4)\right)\theta^n. \tag{2.34}$$

If the Dirichlet boundary condition is imposed for the temperature, (2.34) is updated at interior points (x_i, y_j), $1\leqslant i, j\leqslant N-1$, and the boundary value of θ^{n+1} is given by (1.2); if the Neumann boundary condition is imposed for the temperature, (2.34) is updated at all computational points (x_i, y_j), $0\leqslant i, j\leqslant N$.

Step 3. Solve for $\{\psi_{i,j}^{n+1}\}_{1\leqslant i,j\leqslant N-1}$ using

$$\begin{cases}\left(\Delta_h+\dfrac{h^2}{6}D_x^2D_y^2\right)\psi^{n+1}=\bar{\omega}^{n+1},\\ \psi^{n+1}|_\Gamma=0,\end{cases} \tag{2.35}$$

where only Sine transformations are needed. Compute ψ^{n+1} at the "ghost" points using (2.27), (2.28) (together with Briley's vorticity boundary condition (2.26)), or using (2.30), (2.31) (together with the new vorticity boundary condition (2.29)). We note that solving (2.35) only requires $\bar{\omega}^{n+1}$ at interior points (x_i, y_j), $1\leqslant i, j\leqslant N-1$, which has been computed in Step 1.

Step 4. If the Dirichlet boundary condition is imposed for the temperature, calculate "ghost" point value $\theta_{i,-1}$ by the formula (2.7) or (2.9); if the Neumann boundary condition is imposed for the temperature, use (2.14), (2.15) to calculate θ at "ghost" points.

Step 5. Since ψ^{n+1} (including the "ghost" point value) has been computed in Step 3, we are able to obtain the boundary value for ω^{n+1} by either Briley's formula (2.26) or the new fourth order formula (2.29).

Step 6. Now we use the boundary values for ω^{n+1} updated in Step 5 to solve for $\{\omega_{i,j}^{n+1}\}_{i\geqslant 1, j\geqslant 1}$ using

$$\left(1+\frac{h^2}{12}\Delta_h\right)\omega^{n+1}=\bar{\omega}^{n+1}. \tag{2.35}$$

Step 7. Update the velocity $u_{i,j}^{n+1}$, $v_{i,j}^{n+1}$ using the fourth order difference scheme

$$u^{n+1}=-\tilde{D}_y\left(1-\frac{h^2}{6}D_y^2\right)\psi^{n+1},\qquad v^{n+1}=\tilde{D}_x\left(1-\frac{h^2}{6}D_x^2\right)\psi^{n+1}, \tag{2.36}$$

for $i, j\geqslant 1$, and $u^{n+1}|_\Gamma=0$, $v^{n+1}|_\Gamma=0$.

As for the time step constraint, the overall scheme is stable as long as Δt satisfies

$$\frac{\|\mathbf{u}\|_\infty \Delta t}{h} = \mathrm{CFL} \leqslant 1.0 \qquad \text{and} \qquad \frac{\alpha \Delta t}{h^2} \leqslant \left(\frac{1}{4}\right), \tag{2.37}$$

where $h = \min\{\Delta x, \Delta y\}$, and $\alpha = \max\{\nu, \kappa\}$. See [5, 21] for a further discussion of issues concerning the choice of the time stepping scheme and stability conditions.

3. ACCURACY CHECK USING THE LORENZ SYSTEM

We consider a well-known model dealing with Rayleigh–Bénard convection, which was proposed by Lorenz (1963). He expanded the equations describing two-dimensional nonlinear convection on a uniformly heated plane with stress-free boundaries in a double Fourier series. The resulting system of equations was then truncated radically, so that only three ODEs remained. These are the so-called Lorenz system

$$\begin{cases} \dfrac{dX}{dT} = -\sigma X + \sigma Y, \\ \dfrac{dY}{dT} = rX - Y - ZX, \\ \dfrac{dZ}{dT} = -bZ + XY, \end{cases} \tag{3.1}$$

in which X is proportional to the amplitude of the convection motions, Y is proportional to the temperature difference between the ascending and descending motions (i.e., the horizontal temperature difference across a roll), and Z is proportional to the deviation of the vertical temperature profile from the linear profile. The parameter σ stands for Prandtl number; while r is the ratio of the Rayleigh number to the critical Rayleigh number, i.e., $r = \frac{Ra}{R_c}$, and b is a parameter related to the wavenumber which will be shown later.

Now we fit the Lorenz system to the Boussinesq equations. A single-mode stream function can be chosen as

$$\psi_e(\mathbf{x}, t) = P(t) \sin(kx) \sin(y), \tag{3.2}$$

and the temperature can be chosen as

$$\theta_e(\mathbf{x}, t) = A(t) \cos(kx) \sin(y) + B(t) \sin(2y) + (\pi - y), \tag{3.3}$$

where k is the wavenumber. It shall be noted that two different modes were used in the temperature profile. The interaction between these two modes reveal a rich nonlinear dynamics phenomenon. The term $\pi - y$ in the temperature stands for the linear profile. Accordingly, the corresponding velocity $\mathbf{u}_e = (-\partial_y \psi_e, \partial_x \psi_e)$ and the vorticity $\omega_e = \Delta \psi_e$ are computed to be

$$\begin{aligned} u_e(\mathbf{x}, t) &= -P(t) \sin(kx) \cos(y), \\ v_e(\mathbf{x}, t) &= k\, P(t) \cos(kx) \sin(y), \\ \omega_e(\mathbf{x}, t) &= \lambda_k P(t) \sin(kx) \sin(y), \end{aligned} \tag{3.4}$$

where $\lambda_k = -(k^2+1)$. Plugging (3.2)–(3.4) into the momentum equation,

$$\partial_t \omega_e + (\mathbf{u}_e \cdot \nabla)\, \omega_e - Ri\, \partial_x \theta_e = \nu\, \Delta \omega_e, \tag{3.5}$$

we obtain the following nonlinear ODE:

$$\frac{dP}{dt} = \nu \lambda_k P - Ri\, \frac{k}{\lambda_k} A. \tag{3.6}$$

However, these profiles do not satisfy the temperature transport equation exactly. In the original derivation of the Lorenz equation, the high order production term was truncated, which leads to the Lorenz system (3.1). This can be reformulated by adding a force term, which represents the truncated term, to the heat transport equation, so that the profiles satisfy

$$\partial_t \theta_e + \mathbf{u}_e \cdot \nabla \theta_e = \kappa\, \Delta \theta_e + \mathbf{f}, \tag{3.7}$$

where

$$\mathbf{f} = 2PB \cos(kx) \sin(y)(\cos(2y) - 1). \tag{3.8}$$

The combination of (3.7) and (3.8) result in the following ODEs:

$$\begin{cases} \dfrac{dA}{dt} = k\, P + \kappa \lambda_k A - 2kBP, \\ \dfrac{dB}{dt} = -4\kappa B - \dfrac{k}{2} PA. \end{cases} \tag{3.9}$$

We can see that (3.6), (3.9) form a closed system of ODEs for the parameters $P(t)$, $A(t)$, $B(t)$. The equivalence between them and the parameters X, Y, Z appearing in the Lorenz system (3.1) can be derived by the following scaling transformations: denoting $X = \alpha P$, $Y = \beta A$, $Z = \gamma B$, $t = -\frac{T}{\kappa \lambda_k}$ and substituting into (3.1), we get

$$\sigma = \frac{\nu}{\kappa}, \qquad r = -\frac{Ri \cdot k^2}{\nu \kappa \lambda_k^3}, \qquad b = -\frac{4}{\lambda_k}, \tag{3.10}$$

and

$$\alpha = \frac{k}{\kappa\lambda_k}, \qquad \beta = \frac{Ri \cdot k^2}{\nu\kappa\lambda_k^3}, \qquad \gamma = -\frac{2\ Ri\ k^2}{\nu\kappa\lambda_k^3}, \tag{3.11}$$

from which it can be seen that σ is the Prandtl number, b is one parameter related to the wavenumber, and r is the ratio of the Rayleigh number to the critical Rayleigh number.

Accuracy Check

We use our fourth order method proposed in Sec. 2 to solve the Boussinesq flow with force term (3.5), (3.7), (3.8). The initial data are taken as the profiles (3.2)–(3.4) when $t = 0$, and $A(0)$, $B(0)$, $C(0)$ are chosen to be 1. The vorticity boundary condition is taken to be the new fourth order formula (2.29). The application of Briley's formula (2.26) leads to a similar accuracy result. Both the Dirichlet and Neumann boundary condition for the temperature can be imposed in this example. In the case of the Dirichlet boundary condition, the fifth order formula (2.9) is used as our extrapolation for the temperature near the boundary, while the fourth order formula (2.7) leads to almost the same result in our computation. In the case of the Neumann boundary condition, the two formulae (2.14) and (2.15) are used as our extrapolation for the temperature near the boundary, while the force

Table I. Errors and Orders of Accuracy for Boussinesq Equation at $t = 2$ when the Fourth Order Method Is Used and the Dirichlet Boundary Condition for the Temperature Is Imposed. CFL $= \frac{1}{2}$, Where CFL $= \frac{\Delta t}{\Delta x}$

	N	L^∞ error	L^∞ order	L^1 error	L^1 order	L^2 error	L^2 order
θ	32	1.25e−04		4.82e−04		2.00e−04	
	64	7.85e−06	3.99	3.11e−05	3.96	1.27e−05	3.98
	128	4.89e−07	4.00	1.96e−06	3.99	7.95e−07	4.00
	256	3.07e−08	4.00	1.23e−07	4.00	4.99e−08	4.00
ψ	32	4.97e−05		1.46e−04		6.50e−05	
	64	3.21e−06	3.95	9.45e−06	3.95	4.21e−06	3.95
	128	2.01e−07	4.00	5.94e−07	3.99	2.64e−07	4.00
	256	1.27e−08	3.99	3.74e−08	3.99	1.66e−08	3.99
ω	32	6.38e−04		8.50e−04		3.93e−04	
	64	4.46e−05	3.84	4.88e−05	4.12	2.33e−05	4.07
	128	4.31e−06	3.37	3.07e−06	3.99	1.57e−06	3.90
	256	4.69e−07	3.20	2.07e−07	3.89	1.15e−07	3.77

Table II. Errors and Orders of Accuracy for Boussinesq Equation at $t = 2$ when the Fourth Order Method Is Used and the Neumann Boundary Condition for the Temperature Is Imposed. CFL $= \frac{1}{2}$, Where CFL $= \frac{\Delta t}{\Delta x}$

	N	L^∞ error	L^∞ order	L^1 error	L^1 order	L^2 error	L^2 order
θ	32	1.23e−04		4.92e−04		2.01e−04	
	64	7.78e−06	3.98	3.16e−05	3.96	1.28e−05	3.97
	128	4.86e−07	4.00	1.99e−06	3.99	8.01e−07	4.00
	256	3.05e−08	4.00	1.25e−07	3.99	5.03e−08	3.99
ψ	32	5.04e−05		1.49e−04		6.63e−05	
	64	3.23e−06	3.96	9.57e−06	3.96	4.26e−06	3.96
	128	2.02e−07	4.00	6.00e−07	4.00	2.67e−07	4.00
	256	1.27e−08	3.99	3.78e−08	3.99	1.68e−08	3.99
ω	32	4.01e−04		6.69e−04		2.79e−04	
	64	3.44e−05	3.54	4.07e−05	4.04	1.72e−05	4.02
	128	3.55e−06	3.28	2.69e−06	3.92	1.27e−06	3.76
	256	2.85e−07	3.64	1.83e−07	3.88	9.33e−08	3.77

term has to be added and the slip velocity on the boundary, which can be seen from (3.4), will be taken into consideration when we derive $\partial_y^3 \theta$ on Γ_x.

We choose the wave number $k = 1$. The final time is taken to be $t = 2.0$. The other physical parameters are chosen as: $Ri = 1$, $\nu = \kappa = 0.001$. Accordingly, $r = 1.25 \times 10^5$, $\sigma = 1$ and $b = 2$. The computational domain is chosen as $[0, \pi]^2$ with uniform spatial grids $\Delta x = \Delta y = h$, and the time step $\Delta t = \frac{1}{2} \Delta x$.

The exact solutions of stream function and temperature are given by (3.2), (3.3), where the coefficients $P(t)$, $A(t)$ and $B(t)$ are computed by the fourth order Runge–Kutta method applied to the system (3.6), (3.9).

The computations are performed on a sequence of grids: 32^2, 64^2, 128^2, 256^2. The absolute errors of the stream function, vorticity and temperature are listed in Tables I and II, with Dirichlet and Neumann boundary conditions for the temperature imposed, respectively. We see that the temperature and the stream function achieve fourth order accuracy in L^1, L^2 and L^∞ norms. The vorticity achieves almost fourth order accuracy in L^1, L^2 norms and a little less than fourth order accuracy in L^∞ norms.

4. COMPUTATIONAL EXAMPLES

4.1. Marsigli Flow

To illustrate the performance of the fourth order method, we compute an example of the Marsigli flow which has been known since the 17th century.

This example can be found in the work of Marsigli (1681). The detailed story is described in Gill's book "Atmosphere-Ocean Dynamics" [7]:

> It seems that when Marsigli went to Constantinople in 1679 he was told about a well-known undercurrent in the Bosphorous: "... for the fisherman of the towns on the Bosphorous say that the whole stream does not flow in the direction of Byzantium, but while the upper current which we can see plainly does flow in this direction, the deep water of the abyss, as it is called, moves in a direction exactly opposite to that of the upper current and so flows continuously against the current which is seen." That is, the undercurrent water flows toward the Black Sea from the Mediterranean. Marsigli reasoned that the effect was due to density differences: water from the Black Sea is lighter than water from the Mediterranean. The lower density of the Black Sea can be attributed to lower salinity resulting from river runoff. He then performed a laboratory experiment: A container is initially divided in two by a partition. The left side contained water taken from the undercurrent in the Bosphorous, while the right side contained dyed water having the density of surface water in the Black Sea. The experiment was to put two holes in the partition to observe the resulting flow. The flow through the lower hole was in the direction of the undercurrent in the Bosphorous, while the flow through the upper hole was in the direction of the surface flow.

We simulated the above physical process in a simple setup: Boussinesq flow with two initially piecewise constant temperatures in an insulated box $\Omega = [0, 8] \times [0, 1]$. The partition was located at $x = 4$. The temperature was chosen to be 1.5 at the left half, which indicated the lower density, 1 at the right half, which indicated the higher density. (By Boussinesq assumption, the density difference can be converted into temperature difference with the reverse ratio.) The whole flow was at rest at $t = 0$. A no-slip boundary condition was imposed for the velocity and adiabatic boundary condition was imposed for the temperature. The computational method was based on the fourth order scheme discussed above, coupled with fourth order Runge–Kutta time stepping, as described in Sec. 2.5. Briley's formula (2.26) was used as the boundary condition for the vorticity. The adiabatic boundary condition imposed for the temperature indicated the use of (2.16), (2.17) to evaluate the temperature at "ghost" points. In our computation, the Reynolds number was chosen to be $Re = 5000$, the Prandtl number was chosen to be 1, and the Richardson number Ri, which corresponds to the gravity effect, was chosen to be 4. We repeated the computations using two resolutions: 2049×257, 4097×513.

Figures 1 and 2 show the computation results on the resolution of 2049×257 of temperature and vorticity at a sequence of times: $t_1 = 2$, $t_2 = 4$, $t_3 = 6$, $t_4 = 8$, respectively. To save space we only plot the vorticity on the left-half domain $[0, 4] \times [0, 1]$. The vorticity on the right-half domain $[4, 8] \times [0, 1]$ is axis-symmetric to that of the left-half domain. Once the partition is removed, the flow is driven by the gravity force. The results indicate clearly the appearance of an upper current

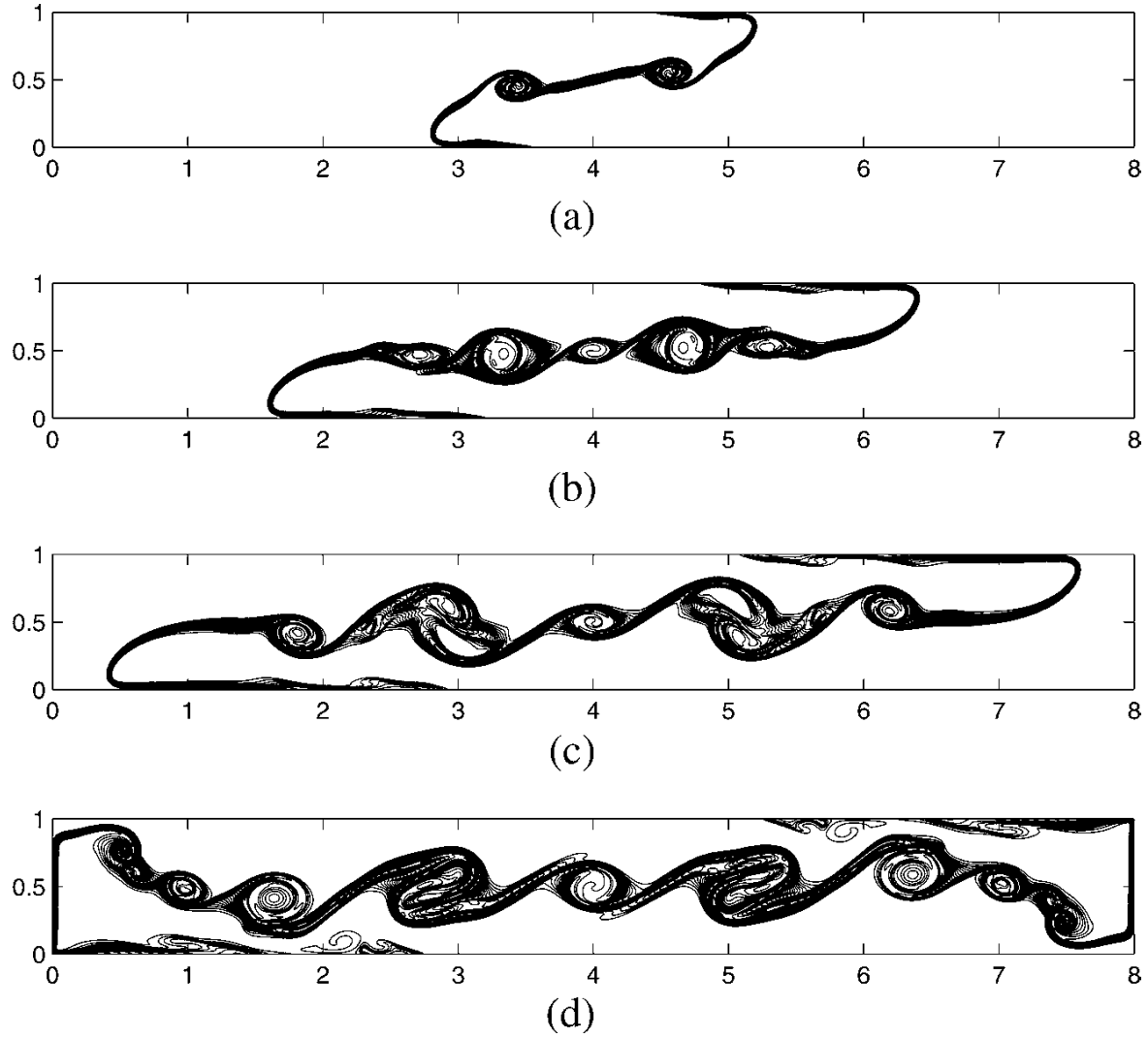

Fig. 1. Temperature plots (a)–(d) of the interaction between two flows with different densities $\frac{3}{2}$:1 in an insulated box $\Omega = [0, 8] \times [0, 1]$, at a sequence of times: $t_1 = 2$, $t_2 = 4$, $t_3 = 6$, $t_4 = 8$. Initially, the two flows are partitioned at $x = 4$. Other physical parameters: $Re = 5000$, $Pr = 1$, $Ri = 4$. The computation is based on the fourth order method with 2049×257 resolution.

flow, which moves from the left side to the right side, and an undercurrent flow, which moves in the opposite direction. It coincides with the phenomenon observed by Marsigli. Consequently, a sharp interface is formed between the two currents. In other words, two currents with different moving directions are separated by an interface. A strong shear flow and vortex sheet form along the interface. This vortex sheet exhibits the Kelvin–Helmholtz instability. As a result, at $t_1 = 2$, two symmetric vortices and the rolling up structures are formed. As the time goes on, more and more rolling-up structures are generated and swell. To see the details, we plotted the temperature and vorticity in a zooming region of $[2.5, 3.5] \times [0, 1]$ at $t_3 = 6$, on the resolution of 4097×513, in Figs. 3 and 4, respectively.

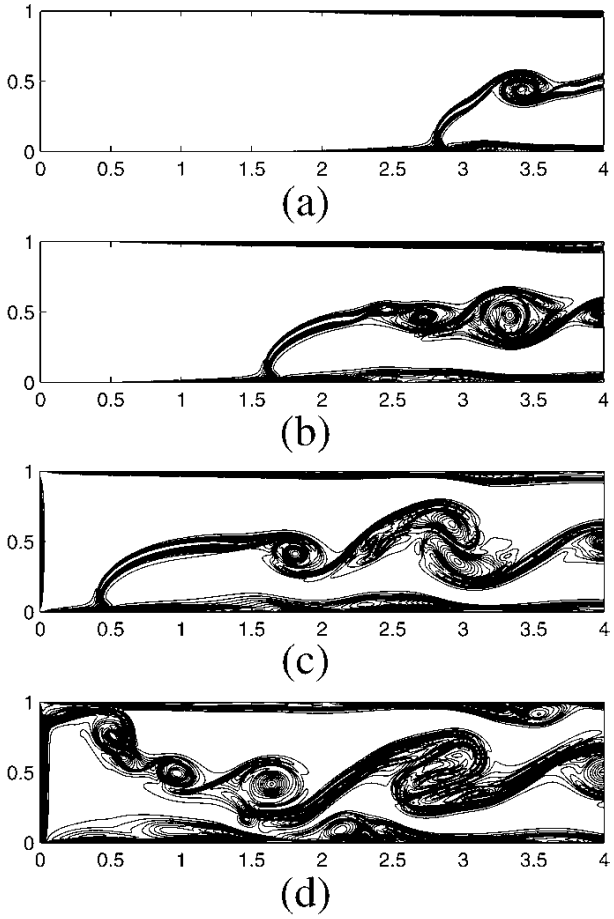

Fig. 2. Vorticity plots (a)–(d) of the left half domain $[0, 4] \times [0, 1]$, at the same times, physical parameters and resolution as in Fig. 1. Forty equally spaced contours from -21 to 31 are used. We omit the vorticity plots on the right half domain $[4, 8] \times [0, 1]$, which is axis-symmetric to the left half domain.

The numerical simulation for this type of Kelvin–Helmholtz instability is quite challenging. To verify the accuracy of our method, we compared the temperature and the vorticity at time $t = 6$ on a $y = \frac{1}{2}$ cut between two resolutions: 2049×257, 4097×513 in Figs. 5 and 6, respectively. It is evident that even though there are many vortical structures and sharp transitional areas for temperature and vorticity profiles, the results using two resolutions match perfectly well. The detailed structures of both kinematic and thermal boundary layers are clearly shown in the plots as well.

4.2. Differentially Heated Cavity: A Benchmark Problem

In this section we present benchmark quality simulations for a differentially-heated cavity problem as further evidence of the accuracy and efficiency of our method. The computation of this flow was the focus of a

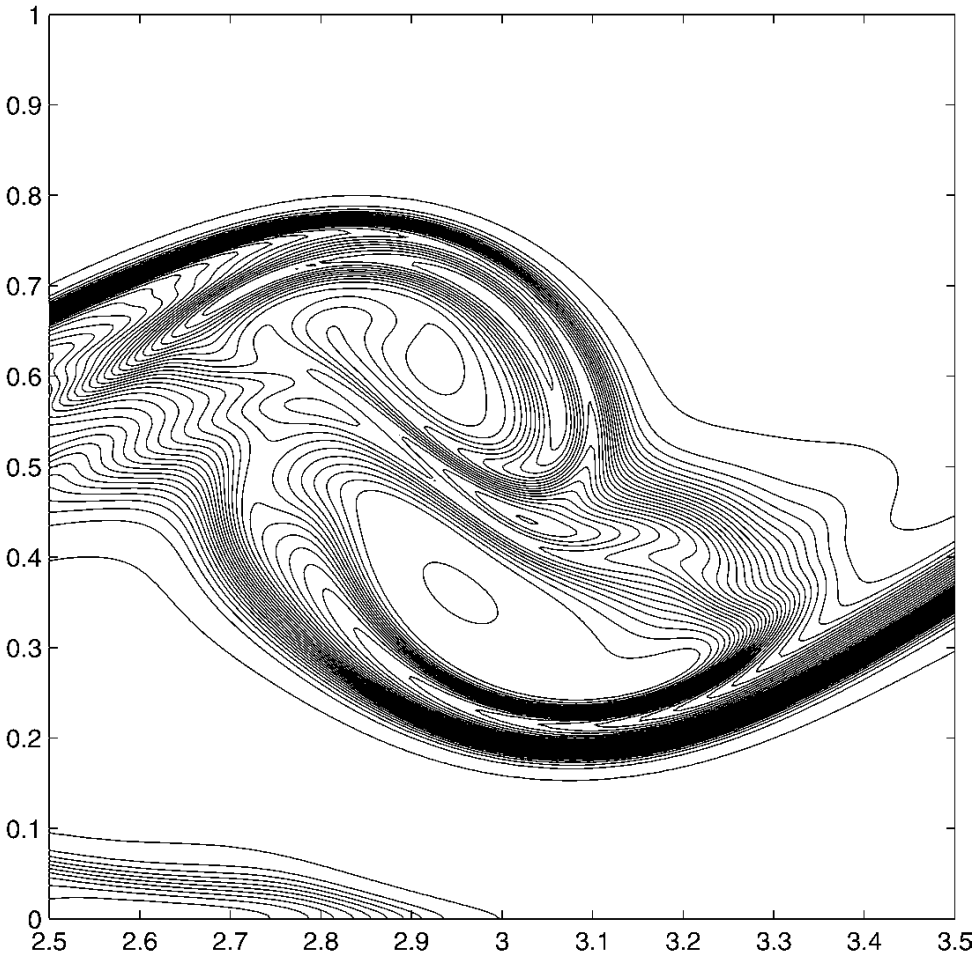

Fig. 3. Zoom plot of temperature in Fig. 1(c) of region $[2.5, 3.5] \times [0, 1]$ with 40 equally spaced contours from 1.001 to 1.499.

special session at the first MIT conference on *Computational Fluid and Solid Mechanics* in June 2001. Submissions to the session included simulations computed using finite difference, finite element, finite volume, and spectral methods. A detailed description of the problem setup, as well as a summary of the overall results can be found at [3]. Additionally, an upcoming special issue of the *International Journal for Numerical Methods in Fluids* will be devoted to the results of the session [12].

The setup of the problem is a buoyancy driven flow enclosed in a differentially-heated tall rectangular cavity of height H and width W, and aspect ratio $8 = H/W$. The flow is driven by an imposed constant temperature difference along the two vertical walls. This in turn defines the nondimensional temperature in terms of the wall temperature difference and a reference temperature as

$$\theta = \frac{T - T_r}{T_h - T_c},$$

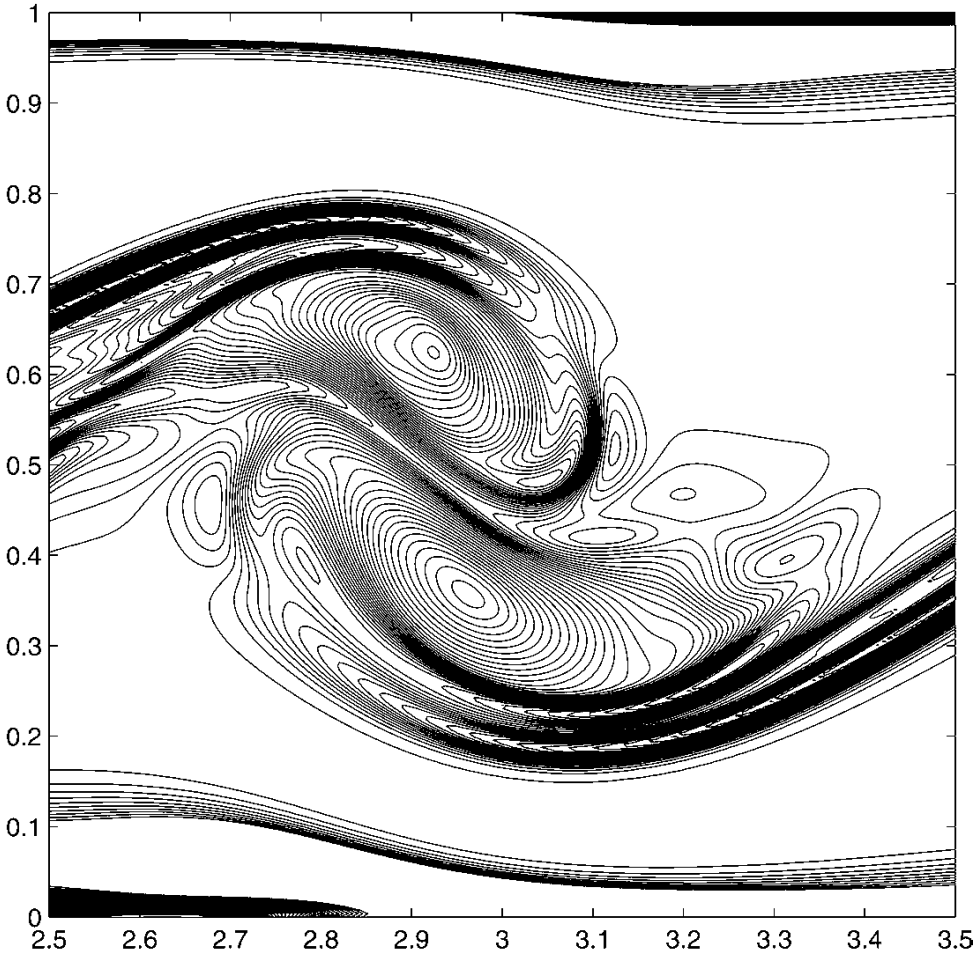

Fig. 4. Zoom plot of vorticity in Fig. 2(c) of region $[2.5, 3.5] \times [0, 1]$ with 40 equally spaced contours from -16.6 to 5.4.

where

$$T_r = \frac{T_h + T_c}{2},$$

and T_h and T_c denote the temperature of the hot and cold wall, respectively. The non-dimensionalized temperature at the left wall ($x = 0$) is then fixed at $\theta = 0.5$, and the right wall ($x = W$) at $\theta = -0.5$. The top and bottom of the cavity ($y = 0, H$) are perfectly insulated, i.e., $\partial\theta/\partial y|_{y=0,H} = 0$.

For our simulation a Boussinesq fluid with Prandtl number $Pr = 0.71$ is initially at rest in a $[0, 1] \times [0, 8]$ enclosure. The Rayleigh number for the benchmark problem was chosen to be a supercritical value of $Ra = 3.4 \times 10^5$. The Richardson number is $Ri = 1$, giving a Reynolds number of $Re = \sqrt{Pr/Ra} \approx 692$. For this parameter set and geometry, after an initial transient period, the flow exhibits time-periodic behavior. We found a final time of $T = 1{,}000$ sufficient for the flow settle into the periodic regime.

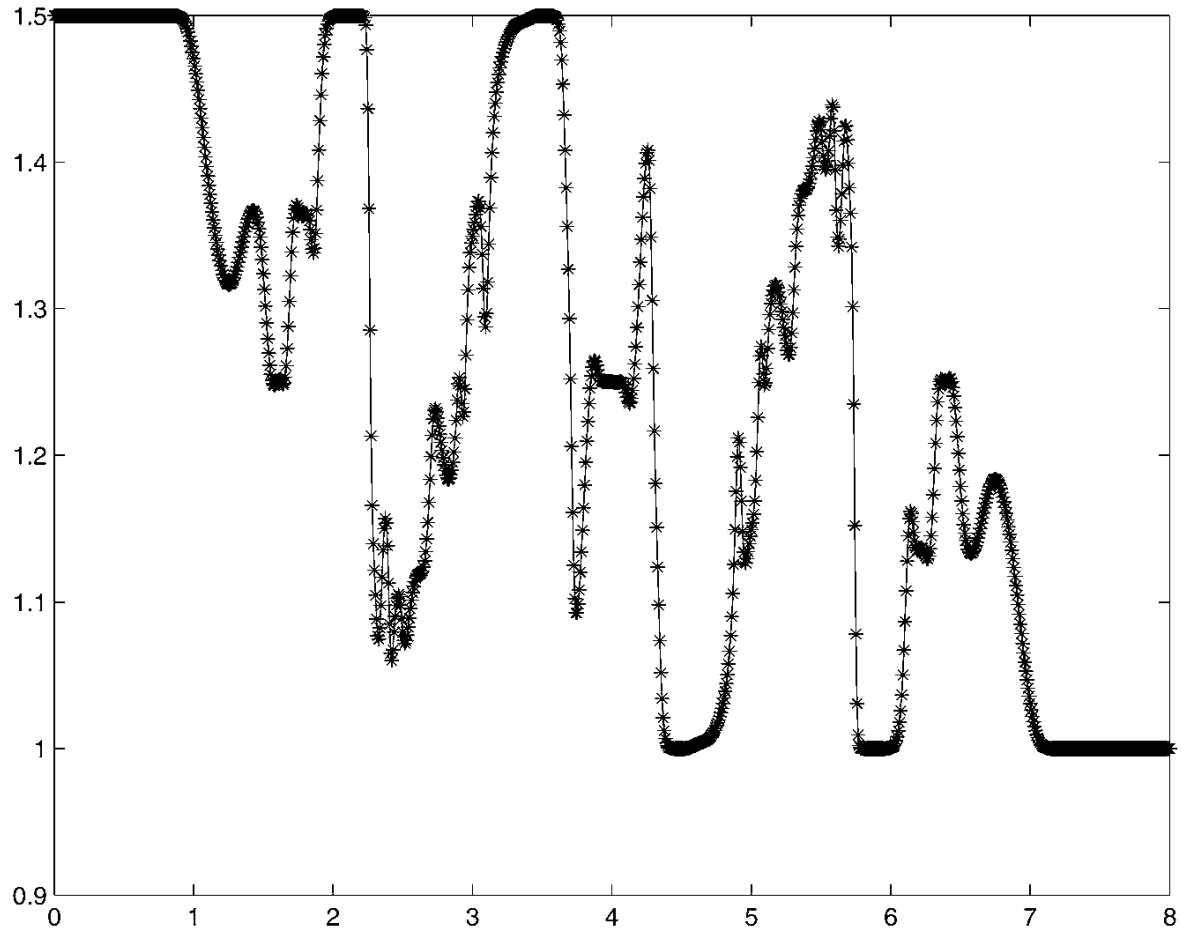

Fig. 5. Comparison of the temperature profile cut along $y = 1/2$ at time $t = 6$ for two grid resolutions. The solid line is from a 4097×513 resolution computation and the star line 2049×257. For clarity only the even points in the star line are plotted, i.e., the graph of the star line only shows 1025 points.

The accuracy of each submission was determined by comparing submitted results with a benchmark simulation computed using a spectral method [20]. In Table III is shown the five points (dimensional) in the flow domain at which the time history of the physical data was collected. In our discretization these points did not correspond to grid points, thus

Table III. Dimensional Coordinates of Time-History Points for the Differentially-Heated Cavity Problem

point	x	y
1	0.181	7.370
2	0.819	0.630
3	0.181	0.630
4	0.819	7.370
5	0.181	4.000

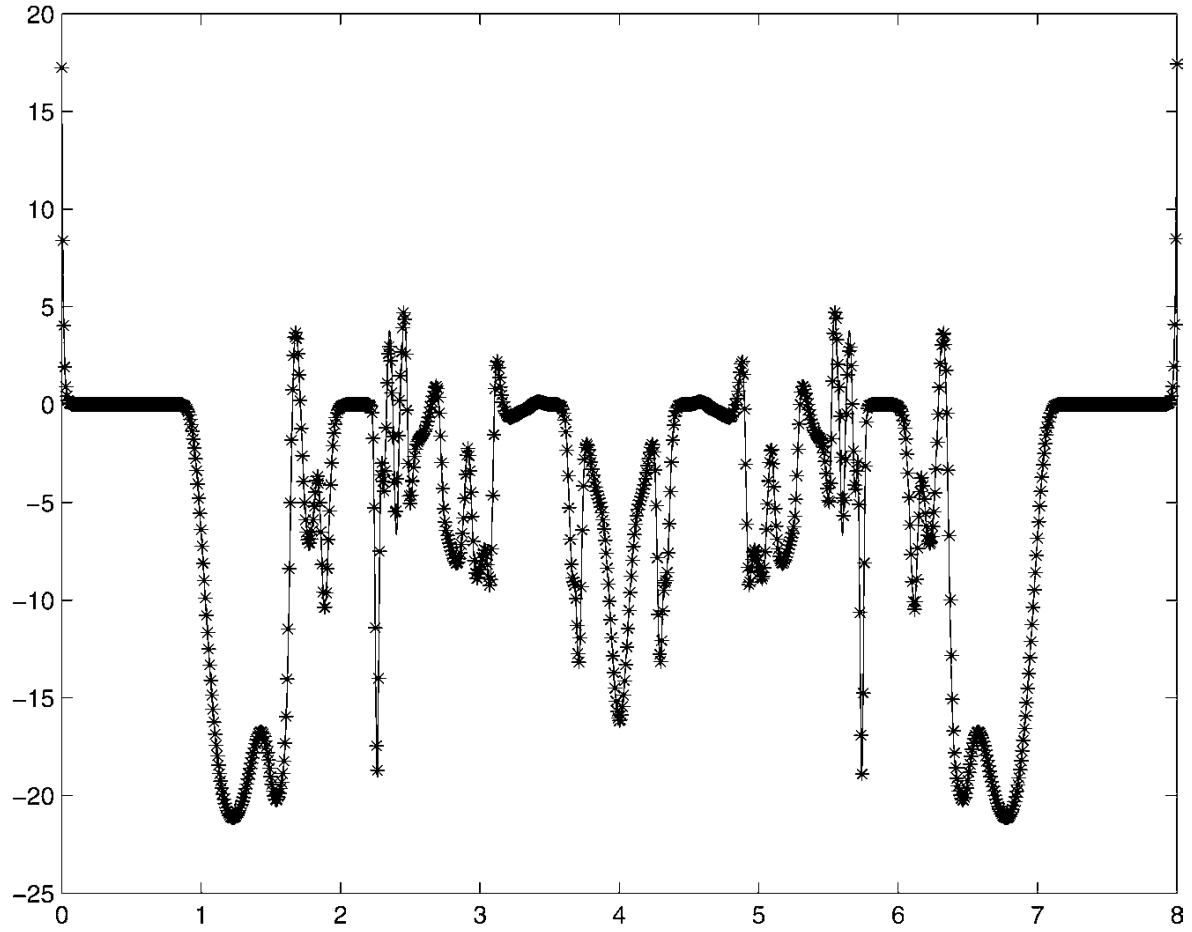

Fig. 6. Comparison of the vorticity profile along $y = 1/2$ at time $t = 6$ with the same two resolutions as in Fig. 5.

our measurements of the flow variables at the data points were obtained using cubic interpolation. The derivatives required in the interpolation procedure were approximated by fourth order finite differences. Time history of the following data was collected: the velocity **u**, vorticity ω, stream function ψ, temperature θ, pressure p, and the skewness ε_{ij} and $\Delta \mathrm{P}_{ij}$ defined by, respectively,

$$\varepsilon_{ij} = \theta_i - \theta_j, \qquad \Delta \mathrm{P}_{ij} = p_i - p_j.$$

The subscripts denote the data point number. Additionally, wall and global measurements were also collected and tabulated. The wall metric consisted of the Nusselt number, computed along each vertical wall via

$$Nu(t) = \frac{1}{H} \int_0^H \left. \frac{\partial \theta}{\partial x} \right|_{x=0,W} dy,$$

where $H = 8$ and $W = 1$. Finally, two global metrics were collected in the form of the square-root of the kinetic energy and the entropy,

$$\hat{u}(t) = \sqrt{\frac{1}{2A}\int_A \mathbf{u}\cdot\mathbf{u}\, dA}, \qquad \hat{\omega}(t) = \sqrt{\frac{1}{2A}\int_A \omega^2\, dA},$$

where $A = 8$, the area of the enclosure.

The period of each measurement was computed by finding the zero crossings of mean adjusted data taken over 10 periods starting at approximately $T = 950$. Simpson's rule was used to approximate all integrals, and the derivative in the evaluation of Nu was approximated to fourth order via finite differences. The stencil, e.g., along $x = 0$, included the points $\{\theta_{i,j}: i = -1,\ldots, 4\}$ with the $\theta_{-1,j}$ ghost point defined as discussed in Sec. 2.2.1.

Our numerical scheme is based on the (ω, ψ) formulation, thus for the pressure measurements we solve the derived pressure Poisson equation

$$\triangle p = (\nabla\cdot\mathbf{u})^2 - (\nabla\mathbf{u}) : (\nabla\mathbf{u})^T + Ri\,\theta_y, \tag{4.1}$$

using a second order finite difference scheme for non-staggered grids based on local pressure boundary conditions. The scheme was proposed in [13], and here we briefly explain the key aspect of the method. Note that (4.1) follows by taking the divergence of the momentum equation in (1.1) and using $\nabla\cdot\mathbf{u} = 0$ to simplify the result. One can then view (4.1) as a replacement for the incompressibility condition, but with the additional requirement that

$$\nabla\cdot\mathbf{u}|_\Gamma = 0. \tag{4.2}$$

A natural boundary condition for (4.1) is

$$\frac{\partial p}{\partial n} = (\nu\triangle\mathbf{u} + Ri\,\hat{\jmath}\theta)\cdot\mathbf{n}, \tag{4.3}$$

a Neumann boundary condition derived by dotting the momentum equation in (1.1) with the unit normal $\mathbf{n}$ at the boundary. Here $\hat{\jmath}$ is the unit vector in the y direction. The difficulty numerically in applying (4.3) is that it requires an approximation of the Laplacian at the boundary. Specifically, the stencil of the second order centered approximation of $\triangle$ along the boundary Γ requires an undefined "ghost point" lying outside of the computational domain. We remedy the situation as follows: Consider the

horizontal boundary segment Γ_x along $y = 0$. A second order centered approximation of $(\nu \triangle \mathbf{u}) \cdot \mathbf{n}$ at the $(i, 0)$ grid point is given by

$$\begin{aligned}(\nu \triangle \mathbf{u}) \cdot \mathbf{n} &= \nu(v_{xx} + v_{yy}) \\ &= \nu(v_{i+1,0} + v_{i-1,0} + v_{i,1} + v_{i,-1} - 4v_{i,0})/h^2 + O(h^2) \\ &= \nu(v_{i,1} + v_{i,-1})/h^2 + O(h^2), \end{aligned} \tag{4.4}$$

where we have used $\mathbf{u}|_\Gamma = 0$, which requires the value $v_{i,-1}$. Discretizing the boundary condition $\nabla \cdot \mathbf{u} = 0$ in (4.2) as

$$0 = \nabla \cdot \mathbf{u} = u_x + v_y = 0 + v_y = \frac{(v_{i,1} - v_{i,-1})}{2h} + O(h^2),$$

where again we have used $\mathbf{u}|_\Gamma = 0$, giving $v_{i,-1} = v_{i,1} + O(h^3)$, implying in (4.4) that one should take $v_{i,-1} = v_{i,1}$. The result is the following approximation for the Neumann boundary condition (4.3):

$$\left.\frac{\partial p}{\partial y}\right|_{(x_i, 0)} = \frac{2\nu}{h^2} v_{i,1} + Ri\, \theta_{i,0}.$$

We note the similarly in philosophy between the derivation of local pressure boundary conditions and local vorticity boundary conditions. With the discrete Neumann boundary condition a second order centered discretization of (4.1) is efficiently solved using FFT methods, resulting in a globally second order accurate solution; see [13].

Two grid resolutions $(N_x + 1) \times (N_y + 1)$, equally-spaced in each coordinate direction, were computed: 97×769, and 129×1025. In each case $\Delta x = \Delta y$. The simulations were run until a final non-dimensional time of $T = 1000$. A fixed Δt was used for each simulation as determined by (2.37) with CFL $= 0.75$ and $\|\mathbf{u}\|_\infty = 1.0$. The CFL number was chosen to be less than 1 in order to avoid instability due to the time step during the initial transient where the velocity exceeds $\|\mathbf{u}\|_\infty = 1.0$.

In Tables IV and V are shown tabulated results at the two grid resolutions. Point metrics without subscripts are for the data point 1, $(x, y) = (0.181, 7.370)$. This data was used for comparison of submissions to the special session. Overall the results at the two resolutions are in good agreement. The largest discrepancies are seen in the ΔP pressure measurements. This is attributable to the fact that the pressure is only recovered to second order accuracy. However, we emphasize that the pressure computation does not affect the overall accuracy of the numerical scheme since it is performed only to collect the pressure point data.

Table IV. Point and Global Metrics for the Differentially-Heated Cavity Problem Using a 97×769 Grid

	Grid resolution: 97×769 Duration: [950, 984.11], steps per period: 438		
Quantity	Average	Amplitude	Period
Point Metrics			
U	5.6395438e−02	2.7623066e−02	3.4111534e+00
V	4.6183149e−01	3.8842483e−02	3.4111601e+00
θ	2.6548480e−01	2.1525621e−02	3.4111579e+00
ε_{12}	0.0000000e+00	0.0000000e+00	
ψ	−7.3688018e−02	3.5338156e−03	3.4111682e+00
ω	−2.3687941e+00	5.4377394e−01	3.4111443e+00
ΔP_{14}	−1.3921419e−03	1.0191256e−02	3.4111650e+00
ΔP_{51}	−5.3076112e−01	1.1234816e−02	3.4111632e+00
ΔP_{35}	5.3215326e−01	5.0527695e−03	3.4111605e+00
Global Metrics			
Nu: $x = 0$	−4.5791427e+00	3.5675550e−03	3.4111610e+00
Nu: $x = W$	−4.5791427e+00	3.5675550e−03	3.4111610e+00
$\hat{\mathbf{u}}$	2.3951401e−01	1.6869000e−05	3.4112994e+00
$\hat{\omega}$	3.0170593e+00	1.6070450e−03	3.4111571e+00

Table V. Point and Global Metrics for the Differentially-Heated Cavity Problem Using a 129×1025 Grid

	Grid resolution: 129×1025 Duration: [950, 984.11], steps per period: 582		
Quantity	Average	Amplitude	Period
Point Metrics			
U	5.6362400e−02	2.7466010e−02	3.4114069e+00
V	4.6186152e−01	3.8658700e−02	3.4114072e+00
θ	2.6547916e−01	2.1406913e−02	3.4114065e+00
ε_{12}	0.0000000e+00	0.0000000e+00	
ψ	−7.3704567e−02	3.5142367e−03	3.4113987e+00
ω	−2.3709233e+00	5.4053272e−01	3.4114080e+00
ΔP_{14}	−1.5092478e−03	1.0162467e−02	3.4113997e+00
ΔP_{51}	−5.3185368e−01	1.1202529e−02	3.4114100e+00
ΔP_{35}	5.3336293e−01	5.0273980e−03	3.4114043e+00
Global Metrics			
Nu: $x = 0$	−4.5792112e+00	3.5527550e−03	3.4114053e+00
Nu: $x = W$	−4.5792112e+00	3.5527550e−03	3.4114053e+00
$\hat{\mathbf{u}}$	2.3951080e−01	1.6805000e−05	3.4115615e+00
$\hat{\omega}$	3.0170787e+00	1.6003350e−03	3.4114035e+00

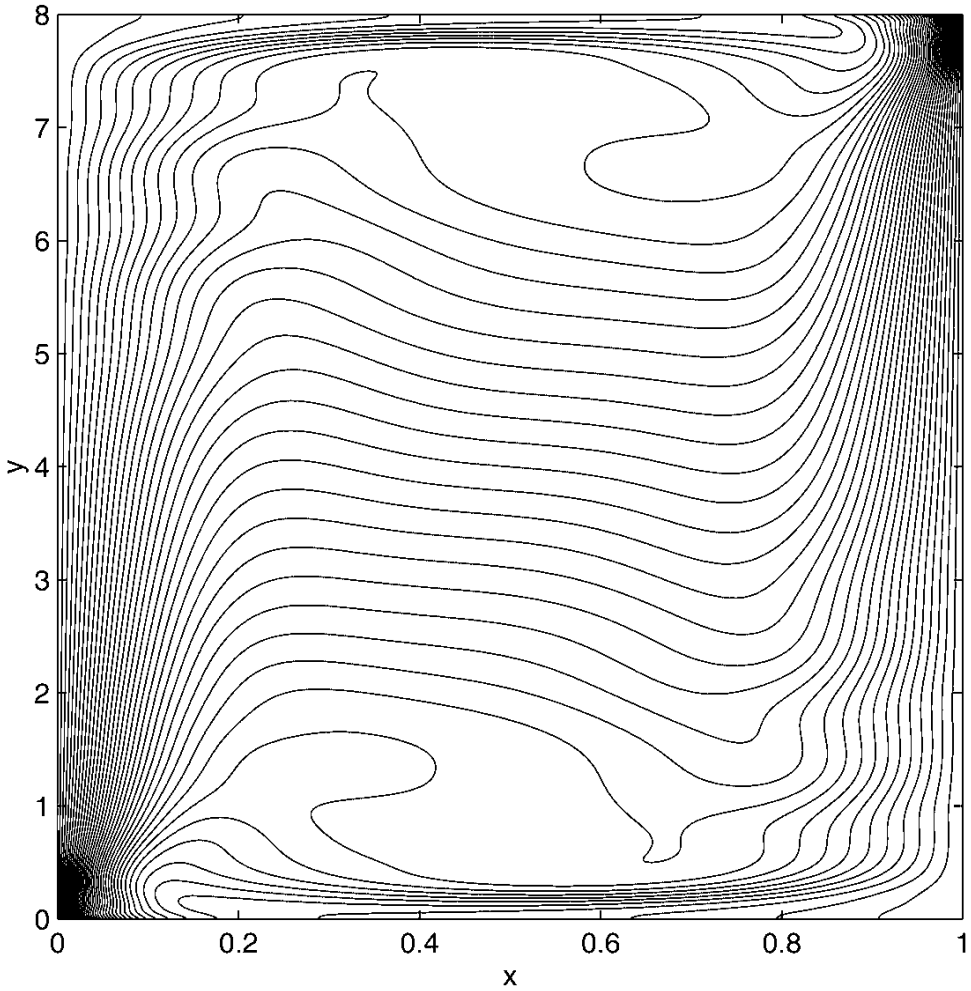

Fig. 7. Contour plot of θ at time $T = 1{,}000$ for the differentially-heated cavity problem for the 129×1025 grid. The levels are (0.5:0.025:−0.5).

Note that in each computation the skewness ε_{12} was identically zero, indicating that the skew-symmetry of the physical flow is captured in our simulation. This feature is evident in the Fig. 7, a contour plot of the temperature at $T = 1000$ for the 129×1025 grid. In Fig. 8 is shown a time history of the temperature θ_1 at $(x, y) = (0.181, 7.370)$ for the 129×1025 grid. Note the initial transient, which settles into a time-periodic behavior as evidenced in the zoom plot.

Submissions were also judge on their computational efficiency. The computations presented here were carried out on a single processor DEC Alpha 500au Personal Workstation with 1028 MB of memory and a 2MB cache. The CPU is an Alpha 21164 500 MHz chip with a SPECfp95 rating of 19.5 (18.0 base). All computations were performed in double precision (64 bits) and the Fortran code was compiled using the *-fast* optimization option. Performance data and timings for each grid based on a final simulation time of $T = 1{,}000$ are shown in Table VII.

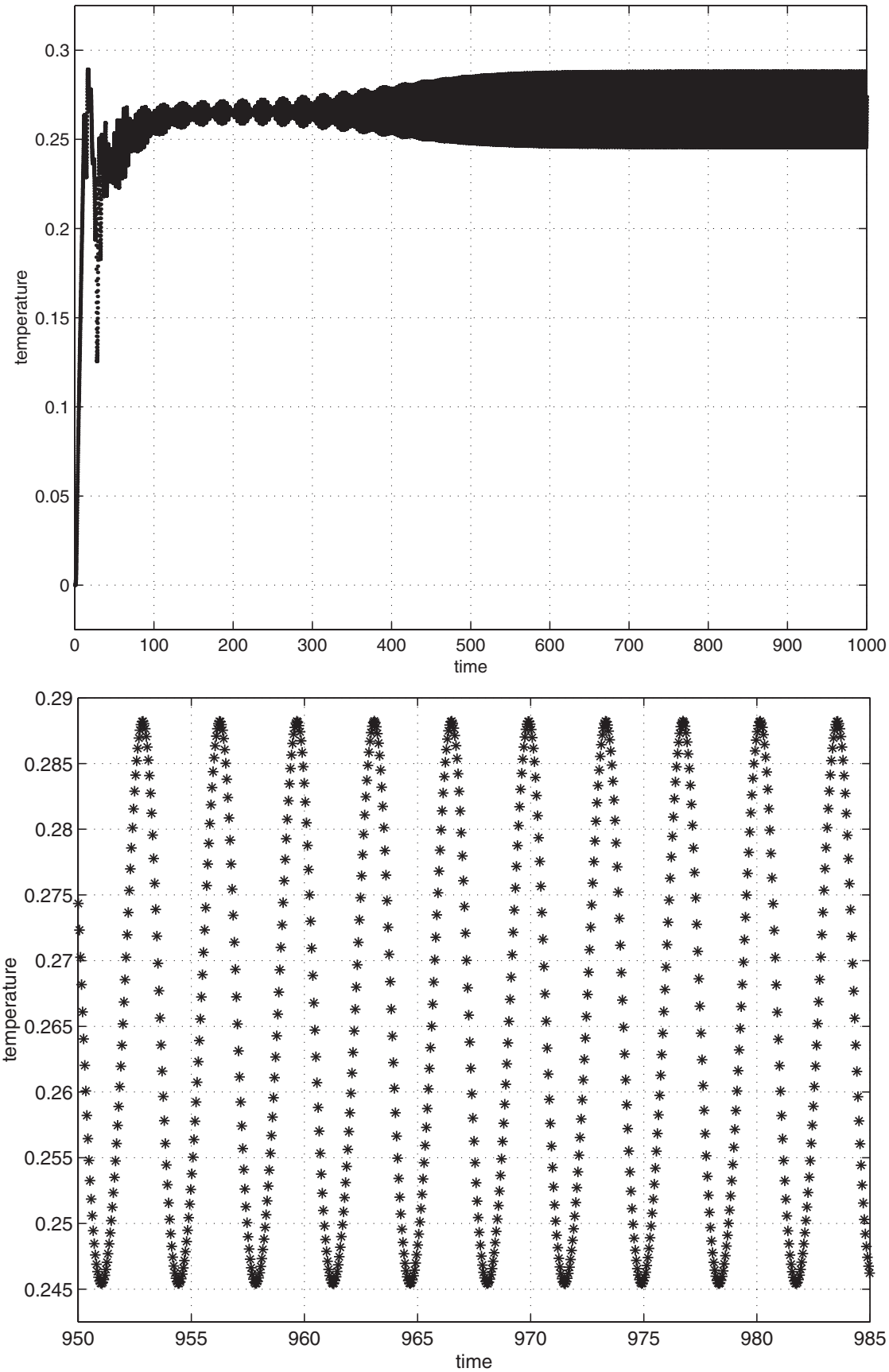

Fig. 8. Time history of θ at $(x, y) = (0.181, 7.370)$ for the differentially-heated cavity problem for the 129×1025 grid.

Table VI. Reference Benchmark Data for the Differentially-Heated Cavity [20]

Quantity	Average	Amplitude
Point Metrics		
U	5.63560e−02	2.7414e−02
θ	2.65480e−01	2.1370e−02
ΔP_{14}	−1.85000e−03	1.0190e−02
Global Metric		
Nu	−4.57946e+00	3.5500e−03

The results of our simulation on the 97×769 grid was submitted to the special session at the first MIT conference on *Computational Fluid and Solid Mechanics*. As noted above, the methodology used to compute the reference benchmark was a spectral code [20, 3, 16] with 48×180 modes. In Table VI is shown flow measurements from the reference benchmark, with the period of each reported as 3.41150 [20]. Information concerning all submissions including the reference benchmark simulation, in particular timings and memory usage, can be found at [20, 3]. Additionally, all results will appear in an upcoming special issue of the *International Journal for Numerical Methods in Fluids*.

The reference benchmark results were used to rank the session submissions. In all there were fifteen basic metrics, with six composite metrics used for the overall rankings. Our simulation with the 97×769 grid received *five* first place rankings and *one* second place ranking among the fifteen basic metrics. Moreover, our results received *three* first place rankings and *one* second place ranking of the six composite metrics. In particular, with respect to numerical accuracy and efficiency our method performed extremely well.

Finally, recall that our simulations were performed using a fixed Δt. This was done in order to collect flow data at a precise fixed time interval. This allowed FFT techniques to be used to compute the metrics. These results are presented in the conference proceedings [11]. The results presented here compute the metrics by determining the zero crossings of mean

Table VII. Timings and Performance Data for Computation of the Differentially-Heated Cavity

Grid size	Δt	Steps	CPU (sec)	Memory (MB)	CPU (μsec)/point/step
(97, 769)	7.812500e−03	128,000	121,161	11.41	12.69
(129, 1025)	5.859375e−03	170,600	335,917	18.62	14.89

adjusted data as describe above, which we found to be more accurate. Thus, a fixed Δt is not necessary and in fact adds significantly to the total computational time. This was determined by performing a simulation and allowing for a variable time step with the 97×769 grid and CFL $= 1.0$. Allowing for a variable time step by monitoring $\|\mathbf{u}\|_\infty$ during the run to determine Δt resulted in a one-third reduction in the total runtime.

ACKNOWLEDGMENTS

The work of J.-G. Liu was supported by NSF Grant DMS-0107218.

REFERENCES

1. Bell, J. B., and Marcus, D. L. (1992). A Second-order projection method for variable-density flows. *J. Comput. Phys.* **101**, 334–348.
2. Briley, W. R. (1971). A numerical study of laminar separation bubbles using the Navier–Stokes equations. *J. Fluid Mech.* **47**, 713–736.
3. Christon, M. (2001). Results Summary: Special session on computational predictability of natural convection flows in enclosures, http://wotan.me.unm.edu/~christon/mit_convection/summary/
4. E, Weinan, and Liu, J.-G. (1996). Vorticity boundary condition for finite difference schemes. *J. Comput. Phys.* **124**, 368–382.
5. E, Weinan, and Liu, J.-G. (1996). Essentially compact schemes for unsteady viscous incompressible flows. *J. Comput. Phys.* **126**, 122–138.
6. E, Weinan, and Shu, Chi-Wang (1994). Small-scale structures in Boussinesq convection. *Phys. Fluids* **6** (1), 49–58.
7. Gill, A. E. (1982). *Atmosphere-Ocean Dynamics*, Academic Press.
8. Glowinski, R., and Pironneau, O. (1979). Numerical methods for the first biharmonic equation and for the two-dimensional Stokes problem. *SIAM Rev.* **21**, 167–212.
9. Henshaw, W. D., Kreiss, H. O., and Reyna, L. G. M. (1994). A fourth-order accurate difference approximation for the incompressible Navier–Stokes equations. *Comput. and Fluids* **23**, 575–593.
10. Hou, T. Y., and Wetton, B., Stable fourth order stream-function methods for incompressible flows with boundaries, unpublished.
11. Johnston, H., and Krasny, R. (2001). Computational predictability of natural convection flows in enclosures: A benchmark problem. In Bathe, K. J. (ed.), *Computational Fluids and Solid Mechanics (Conference Proceedings)*, Elsevier Science.
12. Johnston, H., and Krasny, R. (2002). Fourth order finite difference simulation of a differentially-heated cavity. To appear in *Int. J. Num. Meth. Fluids*.
13. Johnston, H., and Liu J.-G. (2002). Finite difference schemes for incompressible flow based on local pressure boundary conditions. *J. Comput. Phys.* **180**, 120–154.
14. Orszag, S. A., and Israeli, M. (1974). Numerical simulation of viscous incompressible flows. *Ann. Rev. Fluid Mech.* **6**, 281–318.
15. Quartapelle, L. (1983). *Numerical Solution of the Incompressible Navier–Stokes Equations*, Birkhauser, Berlin.
16. Quéré, P. L., and Behnia, M. (1998). From onset of unsteadiness to chaos in a differentially heated cavity. *J. Fluid Mech.* **359**, 81–107.

17. Thom, A. (1933). The flow past circular cylinders at low speeds, *Proc. Roy. Soc. A* **141**, 651–669.
18. Thorpe, S. A. (1968). On standing internal gravity waves of finite amplitude. *J. Fluid Mech.* **32**, 489–528.
19. Thorpe, S. A. (1969). Experiments on the instability of stratified shear flows: Immiscible fluids. *J. Fluid Mech.* **39**, 25–48.
20. Xin, S., and Le Quéré, P. L. (2001). In Bathe, K. J. (ed.), *Computational Fluids and Solid Mechanics (Conference Proceedings)*, Elsevier Science.
21. Wang, C., and Liu, J.-G. (2002). Analysis of finite difference schemes for unsteady Navier–Stokes equations in vorticity formulation. *Numer. Math.* **91**, 543–576.
22. Wang, C., and Liu, J.-G., Fourth order convergence of compact finite difference solvers for 2D incompressible flow. Accepted for publication in *Comm. in Appl. Anal.*
23. Wang, C., Liu, J.-G., and Johnston, H. Analysis of a fourth order finite difference method for incompressible Boussinesq equations. Submitted to *Numer. Math.*