INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 21, 413–432 (1995)

TRANSIENT SOLUTIONS FOR THREE-DIMENSIONAL LID-DRIVEN CAVITY FLOWS BY A LEAST-SQUARES FINITE ELEMENT METHOD

LI Q. TANG,* TIWU CHENG† AND TATE T. H. TSANG‡

Department of Chemical and Materials Engineering, University of Kentucky, Lexington, KY 40506-0046, U.S.A.

SUMMARY

A time-accurate least-squares finite element method is used to simulate three-dimensional flows in a cubic cavity with a uniform moving top. The time- accurate solutions are obtained by the Crank–Nicolson method for time integration and Newton linearization for the convective terms with extensive linearization steps. A matrix-free algorithm of the Jacobi conjugate gradient method is used to solve the symmetric, positive definite linear system of equations. To show that the least-squares finite element method with the Jacobi conjugate gradient technique has promising potential to provide implicit, fully coupled and time-accurate solutions to large-scale three-dimensional fluid flows, we present results for three-dimensional lid-driven flows in a cubic cavity for Reynolds numbers up to 3200.

KEY WORDS: least-squares finite element method; Jacobi conjugate gradient method; three-dimensional flows; time-accurate solutions; lid-driven flows

1. INTRODUCTION

The flow of an incompressible fluid in a square cavity, driven by the moving top at a uniform velocity, has served as a benchmark problem to validate various numerical techniques for two-dimensional fluid flows.[1–4] The interest of the problem is due to its simple geometry and Dirichlet boundary conditions. However, numerical results for three-dimensional driven cavity flows are relatively scarce. Unlike two-dimensional lid-driven cavity flows, flows in a cubic cavity are rich in complicated physical phenomena. Moreover, an increase in the number of unknowns drastically increases the memory requirement for the numerical solution of the resulting linear system of equations by direct methods. It also has, in general, an adverse effect on the rate of convergence of most iterative methods. Therefore difficulties, primarily due to prohibitive computing cost and limitations on computing capabilities, arise when a large number of unknowns are required for accurate numerical solutions of three-dimensional flows.

Earlier simulations of three-dimensional lid-driven flows in a cubic cavity were for low-Reynolds-number flows[5] and results were obtained with inadequate resolutions because of limited computing resources.[6,7] More recently, Guj and Stella[8] used an ADI finite difference method for steady state flows for $400 \leqslant Re \leqslant 2000$ based on the vorticity–velocity formulation of the Navier–Stokes equations with staggered grids varied from $31 \times 31 \times 16$ to $101 \times 101 \times 41$ points. Iwatsu *et al.*[9] used a time-marching MAC method for unsteady state flows at $Re = 2000$ and 4000. The time step was taken as

* Permanent address: Center for Computational Sciences, University of Kentucky, Lexington, KY 40506-0045, U.S.A.
† Permanent address: Department of Basic Sciences, China Textile University, Shanghai, People's Republic of China.
‡ Author to whom correspondence should be addressed.

CCC 0271–2091/95/050413–20

Received 7 September 1994
Revised 4 January 1995

$\Delta t = 10^{-3}$ and the number of grid points was $81 \times 81 \times 81$. The loss of flow symmetry was observed for $Re = 4000$. Direct solutions for unsteady incompressible flows using the velocity–vorticity formulation were obtained by Osswald *et al.*[10] The vorticity transport equation was solved by a modified ADI method. The direct, implicit algorithm was tested for the case of $Re = 100$ with a mesh system of $17 \times 17 \times 17$ grid points.

Ku *et al.*[11] employed a Chebyshev pseudospectral method for unsteady state flows for $100 \leqslant Re \leqslant 1000$ based on primitive variables by using a time-splitting technique with a grid of $31 \times 31 \times 16$ points. Time steps of $\Delta t = 5 \times 10^{-4}$ and 7×10^{-4} were used and it took 7500–25,000 time steps to obtain the steady state solutions.

It is well known that for both steady state and time-dependent problems using an implicit solution approach, most finite element methods lead to a large sparse unsymmetric linear system which is difficult to solve numerically. The steady state lid-driven cavity flows for $100 \leqslant Re \leqslant 400$ based on the vorticity-velocity formulation of the compressible Navier–Stokes equations were simulated by Guevremont *et al.*[12] using a weak Galerkin finite element method followed by Newton linearization. Three grid systems were used: two coarse grids of $8 \times 8 \times 4$ and $14 \times 14 \times 7$ elements by a direct solver and a finer grid of $30 \times 30 \times 15$ elements by an iterative solver based on an Arnoldi method. Kato *et al.*[13] proposed a GSMAC finite element method which uses mass lumping to reduce the memory requirement. The unsteady state flows based on a first-order velocity–vorticity–Bernoulli function formulation were solved for $Re \leqslant 3200$. The grid composed $40 \times 40 \times 20$ elements. Fujima *et al.*[14] reported transient solutions for $Re \leqslant 2000$ by an upwind finite element method based on the choice of upwind and downwind points. The explicit Euler method in time was used for the Navier–Stokes equations and the pressure Poisson equation was solved by an incomplete Cholesky conjugate gradient method. The flow domain was divided into $20 \times 20 \times 10$ non-uniform bricks with six tetrahedra for each brick.

More recently a least-squares finite element method (LSFEM) has been developed to solve steady state incompressible flows in a cubic cavity by Jiang *et al.*[15] at $Re = 100$, 400 and 1000 for a mesh system of $50 \times 52 \times 25$ elements. In addition, Jiang *et al.*[16] presented a detailed theoretical analysis of incompressible Navier– Stokes problems by the LSFEM. They proved that the three-dimensional div–curl system resulting from the LSFEM is properly determined and elliptic and established the existence and uniqueness of the least-squares finite element solution of the system.

Our goal is to develop an accurate, implicit and robust algorithm of the LSFEM to simulate three-dimensional transient flow problems. Particular attention is given to the following approaches.

1. An implicit, fully coupled and time-accurate solution is obtained in terms of a first-order velocity–pressure–vorticity formulation.
2. Temporal discretization of the Navier–Stokes equations is implemented by the Crank–Nicolson method with second-order accuracy in time.
3. The non-linear convective terms are linearized by Newton's method with extensive linearization steps.
4. Spatial discretization of the Navier–Stokes equations is based on the LSFEM to obtain a symmetric, positive definite linear system of equations.
5. The resulting symmetric, positive definite linear system of equations is solved by a matrix-free Jacobi conjugate gradient method (JCG) without forming any global matrix, not even an element matrix.
6. Trilinear hexahedral elements with equal-order interpolation functions are used for all variables.
7. Both uniform and non-uniform meshes are employed up to $60 \times 61 \times 30$ elements.

The content of the paper is organized as follows. In Section 2 we give a brief account of the first-order velocity–pressure–vorticity formulation, compatibility constraint and initial and boundary

conditions. In Section 3 numerical results of three-dimensional lid-driven cavity flows for $Re = 100$–3200 are presented and discussed. The conclusions are given in Section 4.

2. FORMULATION

2.1. Governing equations

Consider the time-dependent three-dimensional incompressible flow in a cubic cavity. The spatial and temporal co-ordinates are denoted by $\mathbf{x} = \{x, y, z\}^{\mathrm{T}}$ and $t \in (0, \infty)$ respectively. The computational domain is $\Omega = \{(x, y, z) \mid 0 \leqslant x \leqslant 1,\ 0 \leqslant y \leqslant 1,\ 0 \leqslant z \leqslant 1\}$. The geometry of the flow configuration is shown in Figure 1. The governing equations in dimensionless form are, in $\Omega \times (0, \infty)$,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \frac{1}{Re} \nabla^2 \mathbf{u} = \mathbf{0}, \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2}$$

where the velocity $\mathbf{u} = \{u, v, w\}^{\mathrm{T}}$, p is the pressure deviation from hydrostatic, $Re = UD/\nu$ is the Reynolds number, U is the characteristic velocity, D is the characteristic length of the domain Ω and $\nu = \mu/\rho$ is the kinematic viscosity.

Compared with the Galerkin finite element method (GFEM), the least-squares approach generally requires more stringent continuity for the trial functions. In order to reduce the continuity requirements so that C^0 elements can be used, Lynne and Arya[17] and Zienkiewicz *et al.*[18] suggested the use of additional variables to reduce the second-order equations to a system of first-order equations. Here we introduce the vorticity $\boldsymbol{\omega} = \nabla \times \mathbf{u} = \{\omega_x, \omega_y, \omega_z\}^{\mathrm{T}}$. Equations (1) and (2) can be transformed into the first-order system

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p + \frac{1}{Re} \nabla \times \boldsymbol{\omega} = \mathbf{0}, \tag{3}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{4}$$

$$\boldsymbol{\omega} - \nabla \times \mathbf{u} = \mathbf{0}. \tag{5}$$

This first-order system is referred to as the velocity–pressure–vorticity formulation and is superior to the velocity–Bernoulli pressure–vorticity formulation because of its properties of faster convergence rate and less sensitivity to initial guesses.[15] Although the total number of unknowns increases in the first-order system, the benefits are that (i) the condition number of the linear system of equations

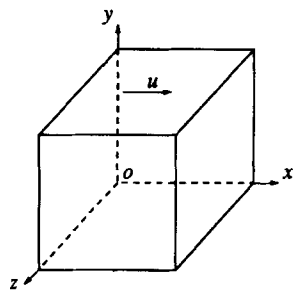

Figure 1. Flow configuration of three-dimensional lid-driven cavity flows

discretized by the LSFEM can be reduced from $O(h^{-4})$[19] to $O(h^{-2})$,[20] (ii) a simple piecewise linear C^0 element can be used, (iii) unlike the GFEM, the choice of approximating spaces is not subject to the LBB condition,[21] so that equal-order interpolation functions can be employed for all variables, and (iv) the resulting matrix system is symmetric, positive definite.

For the three-dimensional problem the first-order system involves an odd number of unknowns (seven) and an odd number of equations and cannot form an elliptic system.[22] Therefore we use a constraint (the solenoidality of vorticity)

$$\nabla \cdot \boldsymbol{\omega} = 0 \quad \text{in } \Omega.$$

The constraint (6) is a very important condition to be enforced in some way in three-dimensional problems when the vorticity is introduced as an unknown. First, it provides an optimal rate of convergence for all variables in the LSFEM.[15,16] Second, it forces $\nabla\cdot\boldsymbol{\omega}(\mathbf{x}, t)$ to approach zero as t goes to infinity if a non-physical initial condition, leading to $\nabla\cdot\boldsymbol{\omega}(\mathbf{x}, t = 0) \neq 0$, is used.[7] On the other hand, if $\nabla\cdot\boldsymbol{\omega} = 0$ initially, it will remain so throughout the computation, since $\partial(\nabla\cdot\boldsymbol{\omega})/\partial t = 0$. Third, the constraint (6) gives an explicit relationship and sets up a physical balance between the vorticity components. Otherwise the vorticity components would implicitly be related by the rotation tensor and the solutions of vorticity may oscillate even at Gaussian points, as will be shown later. For two-dimensional problems the vorticity component exists only in the direction normal to the plane and equation (6) is automatically satisfied. It is worth mentioning that the matrix resulting from the least-squares approximation has a structure like $[\hat{\mathscr{A}}] = [\hat{\mathscr{A}}]^{\mathrm{T}}\,[\hat{\mathscr{A}}]$. The transpose of a non-square matrix $[\mathscr{A}]$ maps itself in $\mathbb{R}N_{\mathrm{eqs}} \times N_{\mathrm{dof}}$ to a matrix $[\mathscr{A}]$ in $\mathbb{R}^{N_{\mathrm{dof}} \times N_{\mathrm{dof}}}$, where N_{eqs} is the number of equations, N_{dof} is the number of degrees of freedom and $N_{\mathrm{eqs}} \geqslant N_{\mathrm{dof}}$. Therefore the additional equation (6) does not increase the total number of unknowns in the system.

2.2. *Time discretization and linearization*

Since we are interested in the time-accurate solutions, equations (3)–(6) are integrated by the Crank–Nicolson scheme, i.e. $\theta = \frac{1}{2}$, $\Delta\tilde{t} = \theta\Delta t$ and $\tilde{\theta} = (1-\theta)/\theta$, and linearized by Newton's method[23] to give

$$\frac{u^{(n+1,m+1)}}{\Delta\tilde{t}} + \mathbf{u}^{(n+1,m)} \cdot \nabla u^{(n+1,m+1)} + \mathbf{u}^{(n+1,m+1)} \cdot \nabla u^{(n+1,m)} + \frac{\partial p^{(n+1,m+1)}}{\partial x} + \frac{1}{Re}\left(\frac{\partial \omega_z}{\partial y} - \frac{\partial \omega_y}{\partial z}\right)^{(n+1,m+1)} = g_u^{(n+1,m)} + f_u^{(n)}, \tag{7}$$

$$\frac{v^{(n+1,M+1)}}{\Delta\tilde{t}} + \mathbf{u}^{(n+1,m)} \cdot \nabla v^{(n+1,m+1)} + \mathbf{u}^{n+1,m+1)} \cdot \nabla v^{(n+1,m)} + \frac{\partial p^{(n+1,m+1)}}{\partial y} + \frac{1}{Re}\left(\frac{\partial \omega_x}{\partial z} - \frac{\partial \omega_z}{\partial x}\right)^{(n+1,m+1)} = g_v^{n+1,m)} + f_v^{(n)}, \tag{8}$$

$$\frac{w^{(n+1,M+1)}}{\Delta\tilde{t}} + \mathbf{u}^{(n+1,m)} \cdot \nabla w^{(n+1,m+1)} + \mathbf{u}^{n+1,m+1)} \cdot \nabla w^{(n+1,m)} + \frac{\partial p^{(n+1,m+1)}}{\partial z} + \frac{1}{Re}\left(\frac{\partial \omega_y}{\partial x} - \frac{\partial \omega_x}{\partial y}\right)^{(n+1,m+1)} = g_w^{(n+1,m)} + f_w^{(n)}, \tag{9}$$

$$\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right)^{(n+1,m+1)} = 0, \tag{10}$$

$$\left(\omega_x + \frac{\partial v}{\partial z} - \frac{\partial w}{\partial y}\right)^{(n+1,m+1)} = 0, \tag{11}$$

$$\left(\omega_y + \frac{\partial w}{\partial x} - \frac{\partial u}{\partial z}\right)^{(n+1,m+1)} = 0, \tag{12}$$

$$\left(\omega_z + \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}\right)^{(n+1,m+1)} = 0, \tag{13}$$

$$\left(\frac{\partial \omega_x}{\partial x} + \frac{\partial \omega_y}{\partial y} + \frac{\partial \omega_z}{\partial z}\right)^{(n+1,m+1)} = 0, \tag{14}$$

where the right-hand-side vectors are

$$g_u^{(n+1,m)} = (\mathbf{u} \cdot \nabla u)^{(n+1,m)}, \tag{15}$$

$$g_v^{(n+1,m)} = (\mathbf{u} \cdot \nabla v)^{(n+1,m)}, \tag{16}$$

$$g_w^{(n+1,m)} = (\mathbf{u} \cdot \nabla w)^{(n+1,m)}, \tag{17}$$

and

$$f_u^{(n)} = \left\{\frac{u}{\Delta \tilde{t}} - \tilde{\theta}\left[\mathbf{u} \cdot \nabla u + \frac{\partial p}{\partial x} + \frac{1}{Re}\left(\frac{\partial \omega_z}{\partial y} - \frac{\partial \omega_y}{\partial z}\right)\right]\right\}^{(n)}, \tag{18}$$

$$f_v^{(n)} = \left\{\frac{v}{\Delta \tilde{t}} - \tilde{\theta}\left[\mathbf{u} \cdot \nabla v + \frac{\partial p}{\partial y} + \frac{1}{Re}\left(\frac{\partial \omega_x}{\partial z} - \frac{\partial \omega_z}{\partial x}\right)\right]\right\}^{(n)}, \tag{19}$$

$$f_w^{(n)} = \left\{\frac{w}{\Delta \tilde{t}} - \tilde{\theta}\left[\mathbf{u} \cdot \nabla w + \frac{\partial p}{\partial z} + \frac{1}{Re}\left(\frac{\partial \omega_y}{\partial x} - \frac{\partial \omega_x}{\partial y}\right)\right]\right\}^{(n)}. \tag{20}$$

Here the superscripts n and m represent the nth time level and the mth Newton linearization step respectively. The structure of the algorithm consists of an outer time loop, Newton linearization and the iterative solver (JCG). For each time step, Newton iterations are carried out. After the solution has converged at the $(m+1)$th Newton iteration, the calculation proceeds to the next time step.

2.3. *Initial and boundary conditions*

Previous results[9,11,13,15] have shown that the flow field is symmetric with respect to the plane $z = 0{\cdot}5$ for Reynolds numbers up to 3200. Thus equations (3)–(6) are solved only for half of the cubic cavity. The planes of the boundary are defined as $\{S_{x-y}\colon\ 0 \leqslant x \leqslant 1,\ 0 \leqslant y \leqslant 1\}$, $\{S_{y-x}\colon\ 0 \leqslant y \leqslant 1,\ 0 \leqslant z \leqslant 0{\cdot}5\}$ and $\{S_{x-z}\colon\ 0 \leqslant x \leqslant 1,\ 0 \leqslant z \leqslant 0{\cdot}5\}$. The boundary conditions are specified as, for $t \geqslant 0$,

$$u = 0,\ v = 0,\ w = 0 \quad \text{on } S_{y-z},\ x = 0 \text{ and } 1, \tag{21}$$

$$u = 1,\ v = 0,\ w = 0 \quad \text{on } S_{x-z},\ y = 1, \tag{22}$$

$$u = 0,\ v = 0,\ w = 0 \quad \text{on } S_{x-z},\ y = 0 \tag{23}$$

$$u = 0,\ v = 0,\ w = 0 \quad \text{on } S_{x-y},\ z = 0 \tag{24}$$

$$\omega_x = 0,\ \omega_y = 0,\ w = 0 \quad \text{on } S_{x-y},\ z = 0{\cdot}5 \tag{25}$$

$$p = 0 \quad \text{at } x = 0{\cdot}5,\ y = 0,\ z = 0{\cdot}5. \tag{26}$$

We assume that the fluid flow starts from a stationary state. The initial conditions are taken as

$$u = v = w = 0 \quad \text{for} \quad \mathbf{x} \in \Omega, \quad t = 0. \tag{27}$$

We should mention that the zero vorticities in boundary condition (25) are not artificial vorticity boundary conditions but are due to symmetry.

The initial–boundary value problem (7)–(14), (21)–(27) can be simply expressed as

$$\{\mathscr{L}\mathbf{v}\}^{(n+1,m+1)} = \{\mathbf{f}\}^{(n)} + \{\mathbf{g}\}^{(n+1,m)} \quad \text{in } \Omega,\ t > 0, \tag{28}$$

$$[\mathscr{B}] \cdot \{\mathbf{v}(\mathbf{x},\ t)\} = \{\mathbf{g}_d(\mathbf{x},\ t)\} \quad \text{on } \Gamma_D,\ t \geq 0, \tag{29}$$

$$\{\mathbf{v}(\mathbf{x},\ 0)\} = \{\mathbf{v}_0(\mathbf{x})\} \quad \text{in } \Omega,\ t = 0, \tag{30}$$

where $\{\mathbf{v}\} = \{u, v, w, p, \omega_x, \omega_y, \omega_z\}^T$, $[\mathscr{B}]$ is the coefficient matrix of boundary conditions and $\{\mathbf{g}_d(\mathbf{x}, t)$ and $\{\mathbf{v}_0(\mathbf{x})\}$ are given functions of boundary and initial conditions. Since the system is first-order, only Dirichlet-type boundary conditions are specified in the LSFEM. If stress boundary conditions have to be specified, we can use shear stresses instead of vorticity to formulate the first-order system of equations, thus leading to Dirichlet-type boundary conditions. In the context of the LSFEM, Jiang[3] discussed different boundary conditions for a variety of fluid flows.

2.4. Least-squares principle

To develop the time-dependent least-squares finite element system, we define the least-squares functional Π of the residual

$$\{\mathscr{R}\} = \{\mathbf{f}\}^{(n)} + \{\mathbf{g}\}^{(n+1,m)} - \{\mathscr{L}\mathbf{v}\}^{(n+1,m+1)} \tag{31}$$

in equation (28) for admissible $\{\mathbf{v}\}^{(n+1,m+1)}$ as

$$\Pi(\{\mathbf{v}\}^{(n+1,m+1)}) = \langle\{\mathscr{R}\},\ \{\mathscr{R}\}\rangle = \int_\Omega \{\mathscr{R}\}^T \cdot \{\mathscr{R}\}\, d\Omega. \tag{32}$$

Let us discretize the domain Ω by a union of finite elements Ω^e as $\Omega = \cup_{e=1}^{N_e} \Omega^e$, where $\Omega^i \cap \Omega^j = 0$ if $i \neq j$ and where N_e is the total number of elements in Ω. The approximate element solution at the $(n + 1, m + 1)$th level is defined as

$$\{\mathbf{v}\}^{(n+1,m+1)} = \{\mathbf{\Phi}(\mathbf{x})\}^T \cdot \{\mathbf{v}(t)\}, \tag{33}$$

where $\{\Phi(\mathbf{x})\}$ is an interpolation function. For the Galerkin finite element method the interpolation function for pressure must be an order lower than that for velocity, whereas the LSFEM allows equal-order interpolation functions for all variables, as indicated by equation (33). Substituting equation (33) into (32), we have

$$\Pi = \int_\Omega \{\mathbf{v}\}^{\mathrm{T}} \cdot [\mathscr{L}\{\mathbf{\Phi}\}]^{\mathrm{T}} \cdot [\mathscr{L}\{\mathbf{\Phi}\}] \cdot \{\mathbf{v}\}\, \mathrm{d}\Omega - 2\int_\Omega \{\mathbf{v}\}^{\mathrm{T}} \cdot [\mathscr{L}\{\mathbf{\Phi}\}]^{\mathrm{T}} \cdot \{\mathbf{b}\}\, \mathrm{d}\mathbf{\Phi}, \tag{34}$$

where $\{\mathbf{b}\} = \{\mathbf{f}\}^{(n)} + \{\mathbf{g}\}^{(n+1,m)}$. This shows that the functional Π is a quadratic with respect to $\{\mathbf{v}\}$. It is desirable to know whether Π is a minimum or a maximum at the point $\delta\Pi = 0$. Obviously equation (34) requires that the sum of the squares of the residual of equation (28) be a minimum at the correct solution.[24] This process is the well-known least-squares method.

Minimization of the functional $\Pi(\{\mathbf{v}\}^{(n+1,m+1)})$ leads to the least-squares weak statement: find

$$\{\mathbf{v}\}^{(n+1,m+1)} \in [\mathscr{H}_0^1(\Omega)]^d \tag{35}$$

such that

$$\delta\Pi(\{\mathbf{v}\}^{(n+1,m+1)}) = \frac{\partial\Pi}{\partial\{\mathbf{v}_1\}}\delta\{\mathbf{v}_1\} + \frac{\partial\Pi}{\partial\{\mathbf{v}_1\}}\delta\{\mathbf{v}_1\} + \cdots + \frac{\partial\Pi}{\partial\{\mathbf{v}_{N_n}\}}\delta\{\mathbf{v}_{N_n n}\} = \mathbf{0}, \tag{36}$$

where $[\mathscr{H}_0^1(\Omega)]^d$ is a subspace of the Sobolev space $[\mathscr{H}^1(\Omega)]^d$, the superscript d denotes the dimension of the space and N_n is the number of nodes in an element. For any variation $\delta\{\mathbf{v}\}$ this yields a set of equations

$$\frac{\partial\Pi}{\partial\{\mathbf{v}\}} = \begin{Bmatrix} \partial\Pi/\partial\{\mathbf{v}_1\} \\ \delta\Pi/\partial\{\mathbf{v}_2\} \\ \vdots \\ \partial\Pi/\partial\{\mathbf{v}_{N_n}\} \end{Bmatrix} = \mathbf{0} \tag{37}$$

from which the solutions $\{\mathbf{v}_i\}$ are found. Therefore equation (37) reduces to a system of linear algebraic equations

$$\sum_e \int_{\Omega^e} [\mathscr{L}\{\mathbf{\Phi}\}]^{\mathrm{T}} \cdot [\mathscr{L}\{\mathbf{\Phi}\}]\, \mathrm{d}\Omega \cdot \{\mathbf{v}\} = \sum_e \int_{\Omega^e} [\mathscr{L}\{\mathbf{\Phi}\}]^{\mathrm{T}} \cdot [\{\mathbf{b}\}]\, \mathrm{d}\Omega \quad \forall\{\mathbf{\Phi}\} \in [\mathscr{H}_0^1(\Omega)]^d, \tag{38}$$

or

$$[\mathscr{K}] \cdot \{\mathscr{V}\} = \{\mathscr{F}\}, \tag{39}$$

where the operator matrix at the ith node of an element is denoted as

$$[\mathscr{L}\{\mathbf{\Phi}\}_i]^T = \begin{pmatrix} \mathrm{d}u_i & \frac{\partial v}{\partial x}\Phi_i & \frac{\partial w}{\partial x}\Phi_i & \frac{\partial \Phi_i}{\partial x} & 0 & -\frac{\partial \Phi_i}{\partial z} & \frac{\partial \Phi_i}{\partial y} & 0 \\ \frac{\partial u}{\partial y}\Phi_i & \mathrm{d}v_i & \frac{\partial w}{\partial y}\Phi_i & \frac{\partial \Phi_i}{\partial y} & \frac{\partial \Phi_i}{\partial z} & 0 & -\frac{\partial \Phi_i}{\partial x} & 0 \\ \frac{\partial u}{\partial z}\Phi_i & \frac{\partial v}{\partial z}\Phi_i & \mathrm{d}w_i & \frac{\partial \Phi_i}{\partial z} & -\frac{\partial \Phi_i}{\partial y} & \frac{\partial \Phi_i}{\partial x} & 0 & 0 \\ \frac{\partial \Phi_i}{\partial x} & \frac{\partial \Phi_i}{\partial y} & \frac{\partial \Phi_i}{\partial z} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{Re}\frac{\partial \Phi_i}{\partial z} & -\frac{1}{Re}\frac{\partial \Phi_i}{\partial y} & 0 & \Phi_i & 0 & 0 & \frac{\partial \Phi_i}{\partial x} \\ -\frac{1}{Re}\frac{\partial \Phi_i}{\partial z} & 0 & \frac{1}{Re}\frac{\partial \Phi_i}{\partial x} & 0 & 0 & \Phi_i & 0 & \frac{\partial \Phi_i}{\partial y} \\ \frac{1}{Re}\frac{\partial \Phi_i}{\partial y} & -\frac{1}{Re}\frac{\partial \Phi_i}{\partial x} & 0 & 0 & 0 & 0 & \Phi_i & \frac{\partial \Phi_i}{\partial z} \end{pmatrix}^{(n+1,m)} \tag{40}$$

and

$$du_i = d\Phi_i + \Phi_i \frac{\partial u}{\partial x}, \qquad dv_i = d\Phi_i + \Phi_i \frac{\partial v}{\partial y}, \qquad dw_i = d\Phi_i + \Phi_i \frac{\partial w}{\partial z},$$
$$d\Phi_i = u\frac{\partial \Phi_i}{\partial x} + v\frac{\partial \Phi_i}{\partial y} + w\frac{\partial \Phi_i}{\partial z} + \frac{\Phi_i}{\Delta \tilde{t}}. \tag{41}$$

2.5. *Jacobi conjugate gradient method*

As mentioned before, the matrix $[\mathscr{K}]$ resulting from the LSFEM is symmetric and positive definite. Assume that a symmetric, positive definite preconditioner $[\mathscr{M}]$ is available. Then we would solve the preconditioned system

$$[\mathscr{M}]^{-1} \cdot [\mathscr{K}] \cdot \{\mathscr{V}\} = [\mathscr{M}]^{-1} \cdot \{\mathscr{F}\}. \tag{42}$$

There are several versions of preconditioned conjugate gradient methods. In this work we use the preconditioned conjugate gradient algorithm of Golub and Loan[25] at the $(n+1)$th time level and the $(m+1)$th Newton linearization step. Matrix–vector multiplication is carried out directly by equation (38) with an element-by-element strategy.

A good preconditioner can have a dramatic effect upon the rate of convergence. Incomplete Cholesky factorization with various levels of fill-in has been commonly used as a preconditioner to the conjugate gradient method. In practice the matrix $[\mathscr{K}]$ and the incomplete Cholesky factorization results need to be stored in an appropriate data structure to facilitate the matrix–vector multiplication. The use of such preconditioners requires a substantial amount of memory for three-dimensional problems. We should mention that although preconditioners reduce the number of iterative steps, their overall performance must be evaluated with the cost of incomplete factorization, the additional memory requirement and the overall CPU time. In this work we choose the preconditioner as

$$[\mathscr{M}] = \operatorname{diag}([\mathscr{K}]) = \int_\Phi \operatorname{diag}([\mathscr{L}\{\boldsymbol{\Phi}\}]^T \cdot [\mathscr{L}\{\boldsymbol{\Phi}\}])\, d\Phi$$
$$= \int_\Phi \begin{pmatrix} du_i^2 + \left[\left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial w}{\partial x}\right)^2\right]\Phi_i^2 + \Sigma_i \\ dv_i^2 + \left[\left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial y}\right)^2\right]\Phi_i^2 + \Sigma_i \\ dw_i^2 + \left[\left(\frac{\partial u}{\partial z}\right)^2 + \left(\frac{\partial v}{\partial z}\right)^2\right]\Phi_i^2 + \Sigma_i \\ \Sigma_i \\ \frac{1}{Re}\left[\left(\frac{\partial \Phi_i}{\partial y}\right)^2 + \left(\frac{\partial \Phi_i}{\partial z}\right)^2\right] + \left(\frac{\partial \Phi_i}{\partial x}\right)^2 + \Phi_i^2 \\ \frac{1}{Re}\left[\left(\frac{\partial \Phi_i}{\partial x}\right)^2 + \left(\frac{\partial \Phi_i}{\partial z}\right)^2\right] + \left(\frac{\partial \Phi_i}{\partial y}\right)^2 + \Phi_i^2 \\ \frac{1}{Re}\left[\left(\frac{\partial \Phi_i}{\partial x}\right)^2 + \left(\frac{\partial \Phi_i}{\partial y}\right)^2\right] + \left(\frac{\partial \Phi_i}{\partial z}\right)^2 + \Phi_i^2 \end{pmatrix} d\Phi, \tag{43}$$

where

$$\Sigma_i = \left(\frac{\partial \Phi_i}{\partial x}\right)^2 + \left(\frac{\partial \Phi_i}{\partial y}\right)^2 + \left(\frac{\partial \Phi_i}{\partial z}\right)^2. \tag{44}$$

Therefore it is a Jacobi conjugate gradient method. Pini and Gambolati[26] found that simple diagonal scaling or the Jacobi conjugate gradient method is an efficient preconditioning scheme for a variety of applications. It is necessary to emphasize that the Jacobi preconditioner is simple and easy to code. It does not require the formation of any global matrix, not even an element matrix. Such a matrix-free approach leads to tremendous savings in computer memory and computing time.[27]

3. RESULTS AND DISCUSSION

A trilinear hexahedral element with eight nodes and one-point Gaussian integration are used in this work. The mesh systems for various Reynolds numbers are given in Table I. The non-uniform mesh system for $Re = 1000$ is shown in Figure 2(a). Figure 2(b) shows the mesh system for $Re = 3200$ with uniform grids in the middle of the cubic cavity and refined grids near the walls for capturing the endwall effects. A layer of thin elements near the top of the cubic cavity is added in order to handle the singularity at the corners when uniform meshes are used for $Re = 1000$ and 2000. The stopping criteria are taken as 10^{-6} for JCG and 10^{-4} for Newton linearization. The time step is chosen as $\Delta t = 0{\cdot}5$ or $1{\cdot}0$. The numerical results show that the pressure and vorticity have some oscillations at nodal points but the solutions at Gaussian points are very smooth. For a trilinear element the nodal values are mapped to the Gaussian point (x_G, y_G, z_G) by the scheme

$$\begin{Bmatrix} p_G \\ \omega_{x_G} \\ \omega_{y_G} \\ \omega_{z_G} \end{Bmatrix} = \frac{1}{8} \sum_{i=1}^{8} \begin{Bmatrix} p_i \\ \omega_{x_i} \\ w_{y_i} \\ \omega_{z_i} \end{Bmatrix}. \tag{45}$$

Equation (45) is an exact arithmetic average.

We first test the flow for $Re = 100$. As mentioned before, the system (3)–(6) has seven unknowns and eight equations. However, it is not overdetermined. Theoretically, a dummy variable ϕ may be added in equation (5) such that

$$\omega - \nabla\phi - \nabla \times \mathbf{u} = \mathbf{0},$$

which satisfies the equation $\nabla^2\phi = 0$. If we artificially enforce the boundary condition $\phi = 0$ on $\partial\Omega$, then the solution of ϕ is uniquely equal to zero throughout the domain Ω. Therefore it is unnecessary

Table I. Mesh systems for various Reynolds numbers

Re	Mesh	Elements	Nodes	Unknowns
100	Uniform	40 × 40 × 20	35301	227580
1000	Uniform	50 × 51 × 25	68952	451803
	Non-uniform	50 × 50 × 25	67626	442975
2000	Uniform	60 × 61 × 30	117242	776463
3200	Non-uniform	60 × 60 × 30	115351	763770

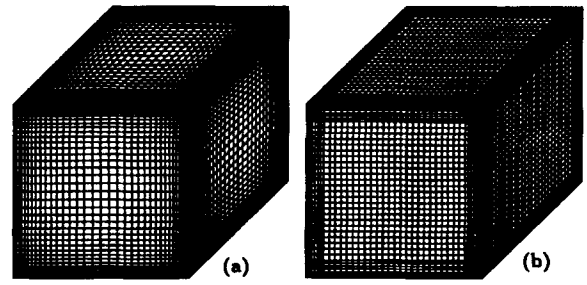

Figure 2. Non-uniform mesh grids for (a) $Re = 1000$ and (b) $Re = 3200$

to add the dummy variable in system (3)–(6). Detailed discussion was given by Jiang *et al.*[15] Our numerical results show that ϕ is not equal to zero but its maximum magnitude is quite small, about $O(10^{-8})$ near the boundary and $O(10^{-10})$ inside the domain when the stopping criterion of JCG is taken to be 10^{-7}. The values of the dummy variable are so small that its existence does not affect the numerical solutions.

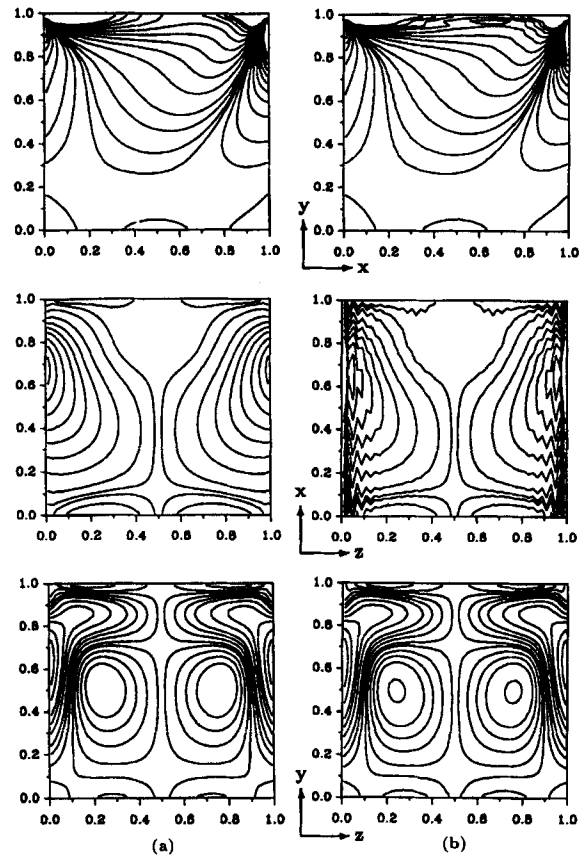

Figure 3. Vorticity contours for $Re = 100$, (a) with constraint (6) and (b) without constraint (6)

Table II. Computer memory requirements and CPU times on an HP-735 workstation

Reynolds number	Number of elements	Stopping time	Time steps	Storage (Mb)	CPU time (h)
100	40 × 40 × 20	8	16	27	2·5
1000	50 × 51 × 25	50	75	54	51·3
2000	60 × 61 × 30	110	160	92	128·6

To show the importance of constraint (6), we simulate the flow for $Re = 100$ for two cases, with and without constraint (6). The vorticity contours on the planes $z = 0{\cdot}5$, $y = 0{\cdot}5$, $x = 0{\cdot}5$ are shown in Figure 3. Both results are smoothed at the Gaussian point of each element by equation (45). The solutions with constraint (6) are very smooth as shown in Figure 3(a). However, without constraint (6) the vorticity components ω_y and ω_z on the planes $y = 0{\cdot}5$, $z = 0{\cdot}5$ oscillate along the walls as shown in Figure 3(b). This indicates that constraint (6) is a very important condition to be enforced in three-dimensional flow problems when vorticity is introduced as an unknown. On the other hand, numerical results also show that the values of velocity and pressure do not differ significantly between the two cases. This implies that constraint (6) hardly disturbs the velocity and pressure fields.

The CPU time for $Re = 100$ is about 2·5 h on an HP-735 workstation (CONVEX MetaSystem) and 8·5 h on an IBM 3090-600J computer. The simulation requires 27 Mb of computer memory. The memory requirements and computing times are given in Table II for various Reynolds numbers.

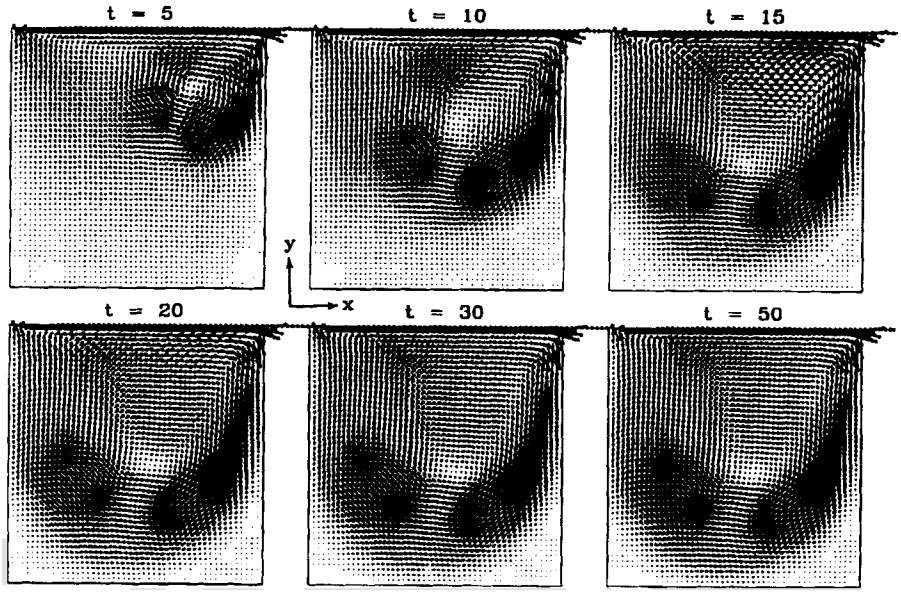

Figure 4. Velocity vectors on plane $z = 0{\cdot}5$ at various time levels for $Re = 1000$

3.1. *Numerical results for Re = 1000*

The transient solutions for $Re = 1000$ are obtained at time levels $t = 5$, 10, 15, 20, 30, 50. The velocity vectors by the uniform mesh on the planes $x = 0{\cdot}5$, $y = 0{\cdot}5$, $z = 0{\cdot}5$ are presented in Figures 4–6. Figure 4 shows that a primary vortex appears near the right corner of the top wall at the beginning and moves to the location at about $x = 0{\cdot}60$, $y = 0{\cdot}45$, similarly to the two-dimensional flow problem.

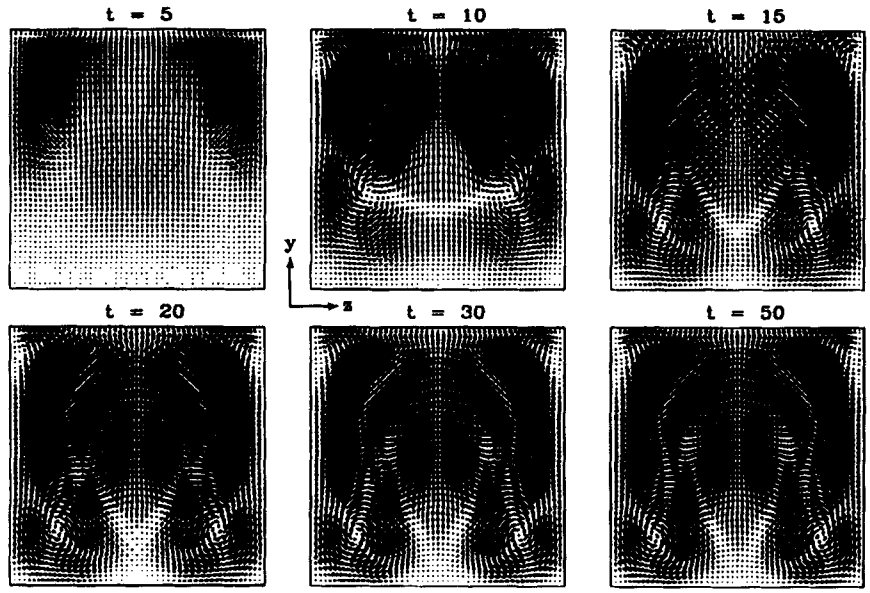

Figure 5. Velocity vectors on plane $x = 0{\cdot}5$ at various time levels for $Re = 1000$

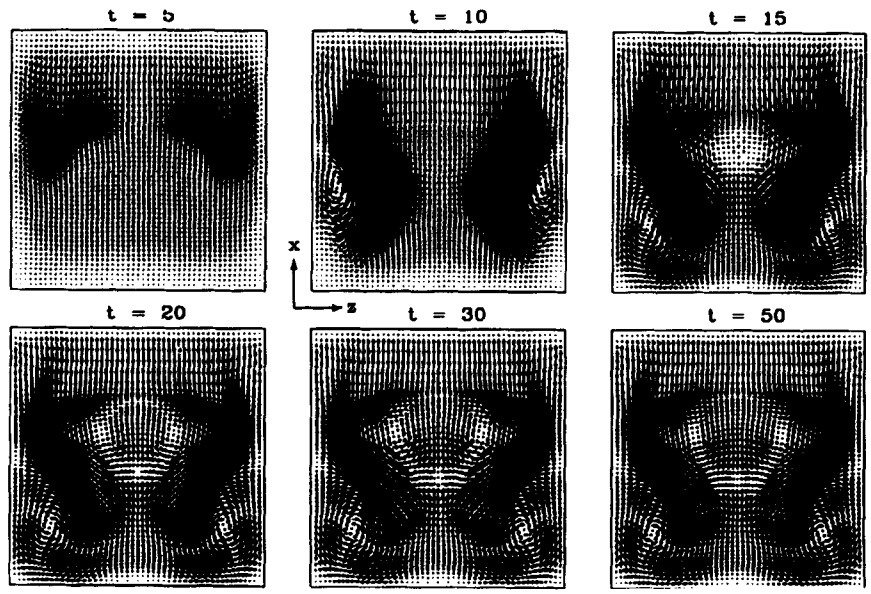

Figure 6. Velocity vectors on plane $y = 0{\cdot}5$ at various time levels for $Re = 1000$

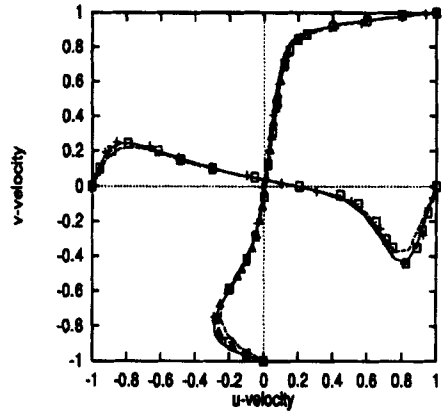

Figure 7. Comparison of velocity profiles u and v along central axes on plane $z = 0{\cdot}5$ for $Re = 1000$:——; Guj and Stella;[8] –+–+–, Ku *et al.*;[11] –△–△–, Jiang *et al.*;[15] ————, present result for non-uniform mesh; – – –, present result for uniform mesh

Figure 5 shows that a pair of vortices forms near the vertical walls on the plane $x = 0{\cdot}5$ and then the vortices move to the lower corners with time. A pair of secondary vortices near the corners of the top wall does not appear on this plane at $t = 50$. As shown in Figure 6, in addition to the pair of vortices at the lower corners, a pair of vortices occurs around the centre of of the cavity on the $y = 0{\cdot}5$ plane at $t = 20$.

The steady state solutions are obtained for both uniform and non-uniform meshes. The velocity profiles along the central lines on the plane $z = 0{\cdot}5$ are presented in Figure 7. Our results for the non-uniform mesh compare very well with the results of Guj and Stella,[8] Ku *et al.*[11] and Jiang *et al.*[15] The absolute values of the velocity components u and v for the uniform mesh are smaller than those for the non-uniform mesh near the vetical and bottom walls. This reveals that the numerical solution is sensitive to the mesh system. This property is also obsesrved by Kato *et al.*[13] and Fujima *et al.*[14] However, the iterative solution with the uniform mesh achieves a faster rate of convergence than that with the non-uniform mesh. The velocity vectors obtained from both uniform and non-uniform meshes are compared in Figure 8. The pair of vortices around the centre of the cavity on the plane $y = 0{\cdot}5$ disappears, which is in agreement with other results.[8,11,13–15] A pair of secondary vortices at the corners of the top wall becomes more evident on the plane $x = 0{\cdot}5$. This is consistent with the results of Fujima *et al.*[14] and Jiang *et al.*[15] We should also mention that our steady state results at the mid-plane of $x = 0{\cdot}5$ in Figure 8 agree with those of Jiang *et al.*[15] in that the vortices at the bottom wall protrude to the centre ($z = 0{\cdot}5$). This vortex behaviour was not reported in the works of Kato *et al.*,[13] Fujima *et al.*[14] and Guj and Stella.[8] It indicates the formation of a rather weak pair of Taylor–Görtler-like (TGL) vortices. The TGL vortices form because of the concave surface generated by the downstream secondary vortex at the bottom of the cavity (see Figure 8 for the lower right corner vortex on the plane $z = 0{\cdot}5$). Pathlines and pressure and vorticity contours are shown in Figures 9 and 10 respectively.

3.2. Numerical results for Re = 2000

With an increase in Reynolds number the flow structure becomes more complicated. A uniform mesh of $60 \times 61 \times 30$ elements is employed for $Re = 2000$. The velocity vectors on the planes $x = 0{\cdot}5, y = 0{\cdot}5, z = 0{\cdot}5$ at time levels $t = 10, 20, 30, 50, 80, 110$ are presented in Figures 11–13. In Figure 11 a primary recirculation forms initially at the right corner of the top wall on the plane $z = 0{\cdot}5$. At $t = 20$ the fluid near the left wall flows against and merges with the primary recirculation. However,

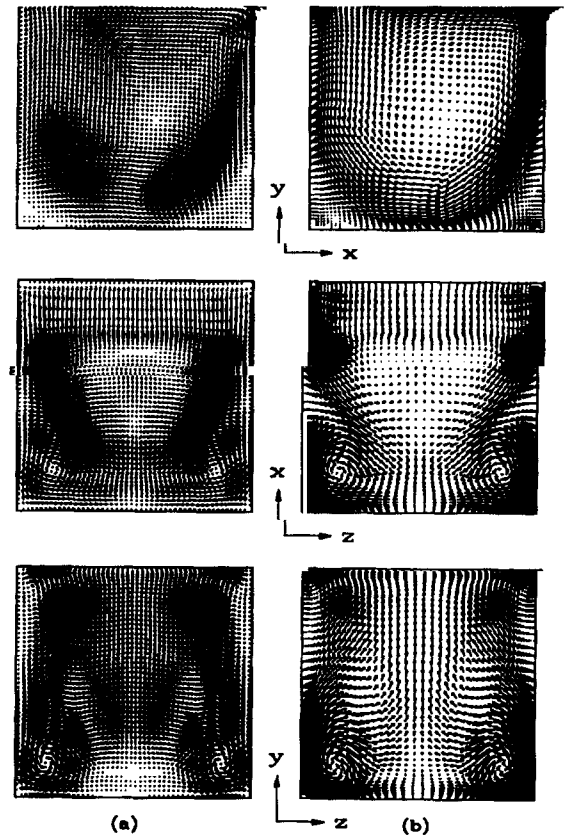

Figure 8. Velocity vectors of steady state solution on planes x, y, $z = 0{\cdot}5$ on (a) uniform and (b) non-uniform meshes for $Re = 1000$

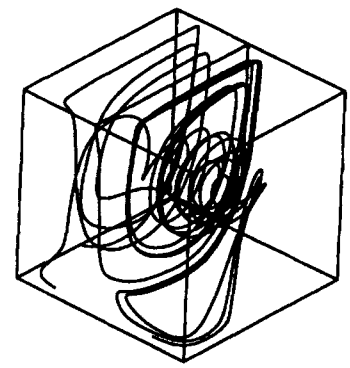

Figure 9. Pathlines for $Re = 1000$

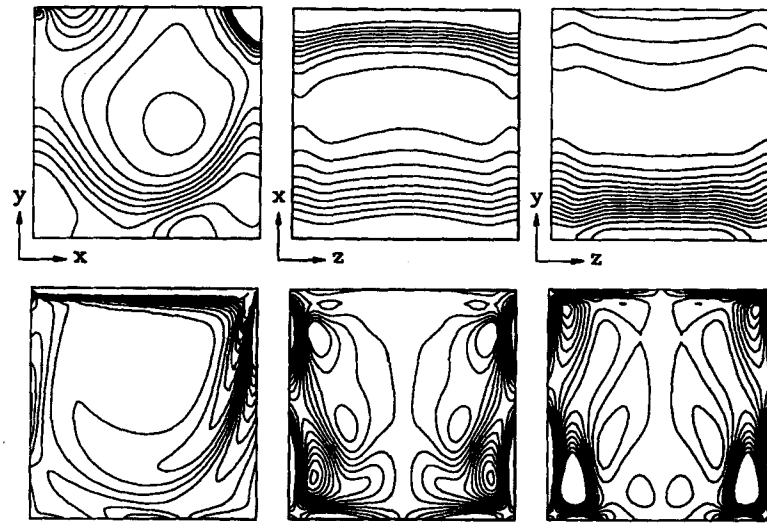

Figure 10. Pressure and vorticity contours of steady state solution on planes $x, y, z = 0{\cdot}5$ on uniform mesh for $Re = 1000$

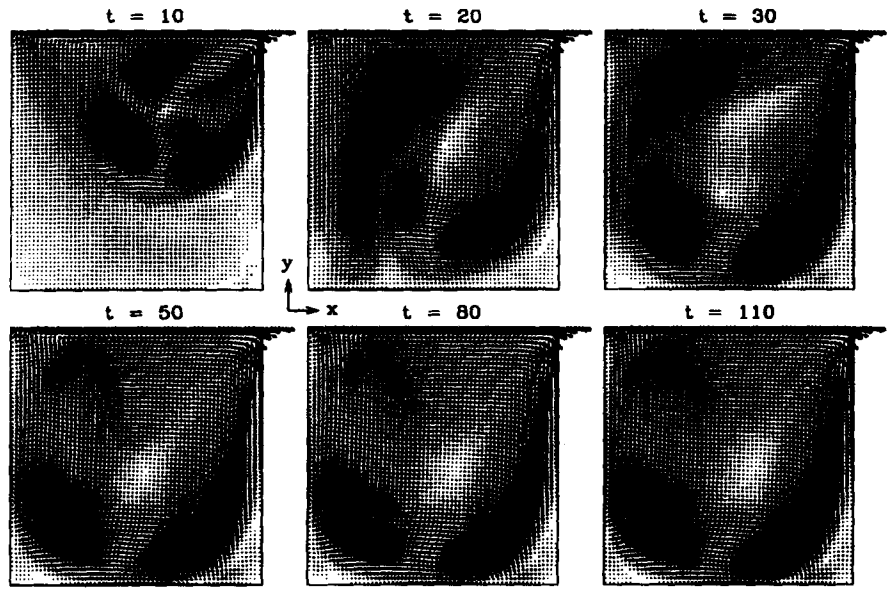

Figure 11. Velocity vectors on plane $z = 0{\cdot}5$ at various time levels for $Re = 2000$

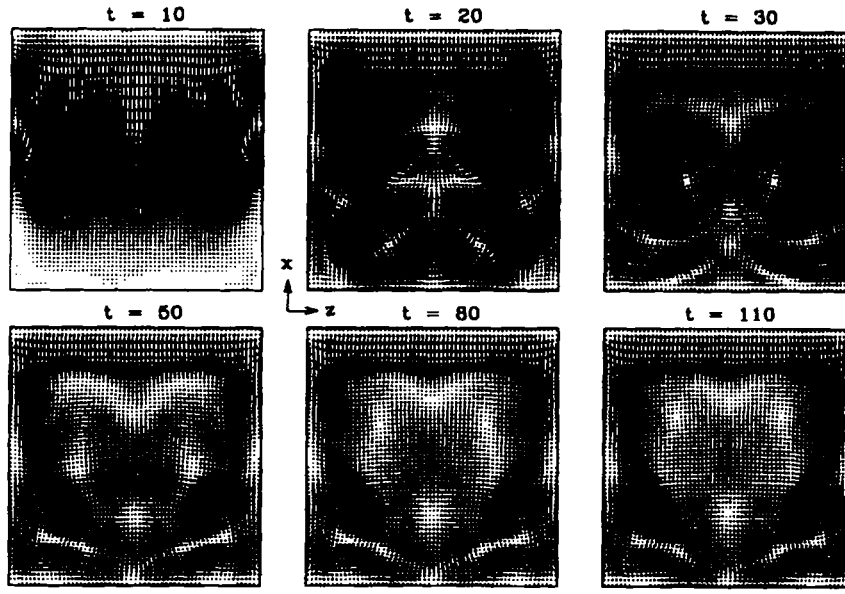

Figure 12. Velocity vectors on plane $y = 0{\cdot}5$ at various time levels for $Re = 2000$

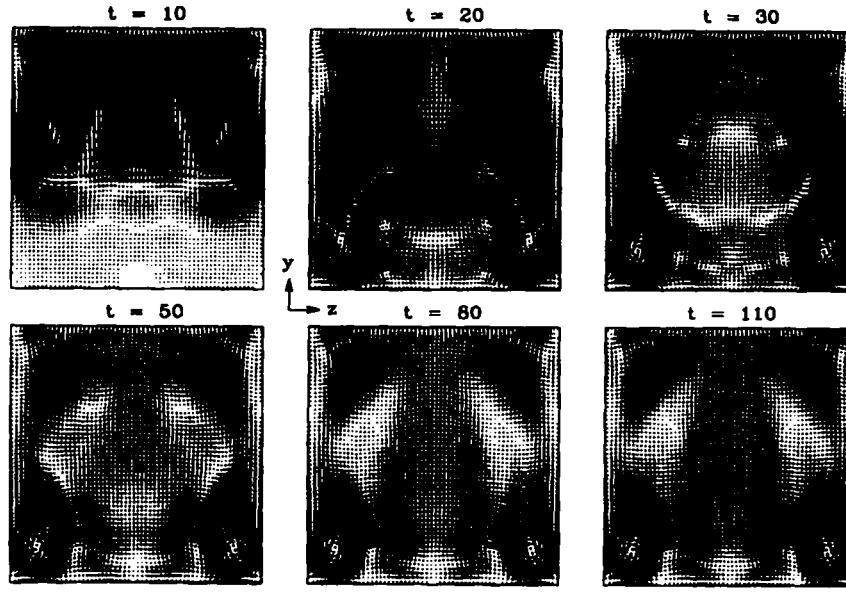

Figure 13. Velocity vectors on plane $x = 0{\cdot}5$ at various time levels for $Re = 2000$

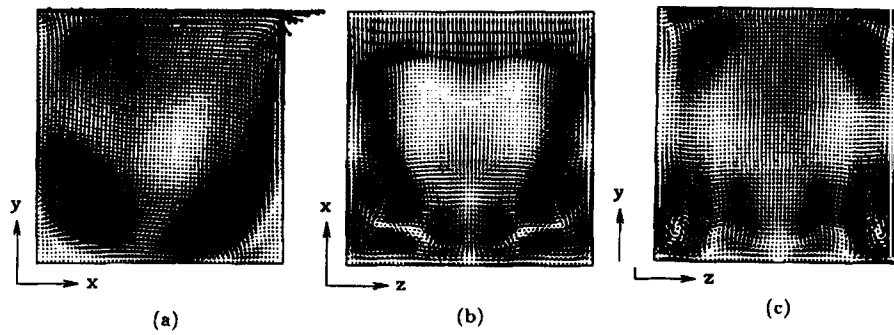

Figure 14. Velocity vectors of steady state solution on planes $x, y, z = 0{\cdot}5$ for $Re = 2000$

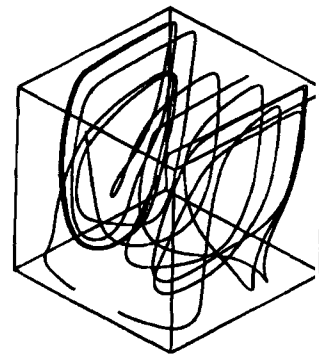

Figure 15. Pathlines for $Re = 2000$

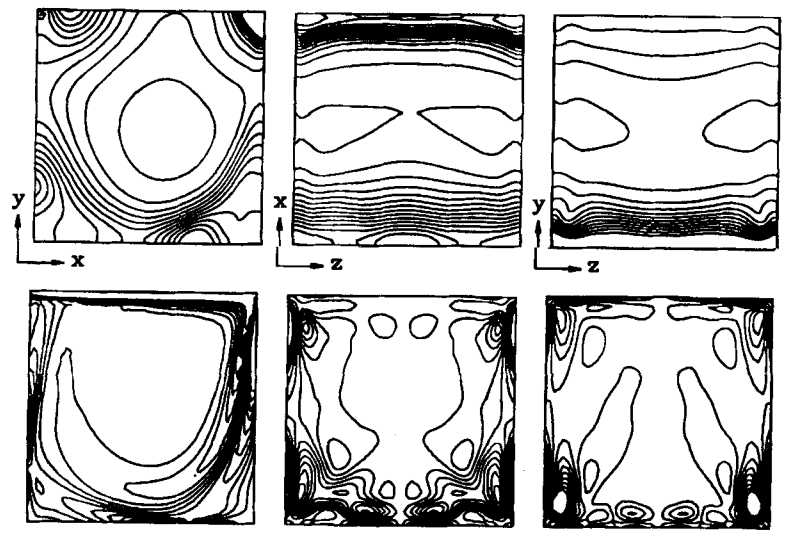

Figure 16. Pressure and vorticity contours of steady state solution on planes $x, y, z = 0{\cdot}5$ on uniform mesh for $Re = 2000$

the same phenomenon is observed for the result of $Re = 3200$ at $t = 90$ by Kato *et al.*[13] The centre of the vortex moves to about $x = 0{\cdot}58$, $y = 0{\cdot}46$ at $t = 110$. In Figure 12 two pairs of vortices appear on the plane $y = 0{\cdot}5$ at about $t = 20$. At $t = 50$ a pair of vortices moves to the location at $x = 0{\cdot}46$ and the other pair of vortices then goes to the location near $x = 0{\cdot}18$ and becomes elongated. In Figure 13, on the plane $x = 0{\cdot}5$, two pairs of vortices appear at about $t = 20$. At $t = 30$ three pairs of vortices appear. The first pair is located at the height of $y = 0{\cdot}58$, the second pair near the centre ($z = 0{\cdot}5$) at the bottom wall and the third pair at the lower corners. At $t = 80$ a pair of secondary vortices starts to form at the corners of the top wall.

The velocity vectors of the steady state solution are presented in Figure 14. The pair of vortices at the centre of the plane $y = 0{\cdot}5$ disappears and only the two elongated vortices exist around the lower corners as shown in Figure 14(b). In Figure 14(c) a pair of secondary vortices at the two corners of the top wall becomes more obvious and larger than the pair for $Re = 1000$. Figure 14(c) also shows the presence of stronger TGL vortices at the bottom of the cavity. Our results compare quite well with those of Guj and Stella[8] and Iwatsu *et al.*[9] Pathlines and pressure and vorticity contours are shown in Figures 15 and 16 respectively.

3.3. *Numerical results for Re = 3200*

When the Reynolds number is 3200, the flow becomes unsteady and difficult to solve numerically. So far, few results for this Reynolds number have been reported. The mesh system used for this problem is shown in Figure 2(b). Sufficient grid points near the walls are needed to extract energy from the driven lid and resolve flow structures. In order to obtain a better rate of convergence for the Jacobi

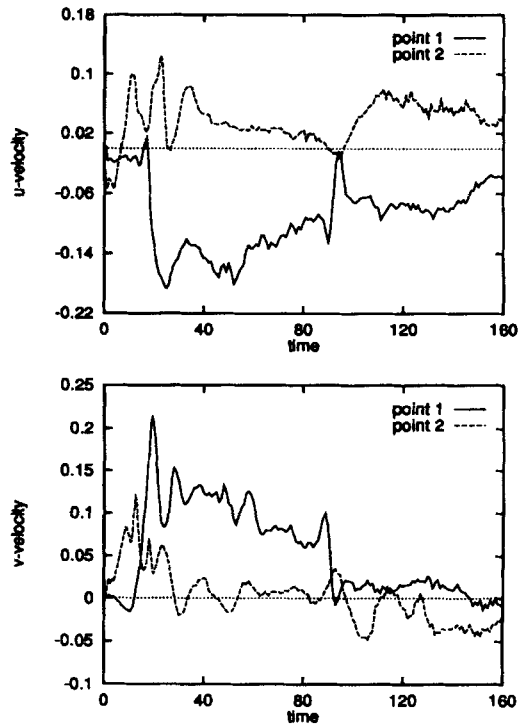

Figure 17. Time histories of velocity components u and v at point 1 ($x = 0{\cdot}5$, $y = 0{\cdot}18$, $z = 0{\cdot}5$) and point 2 ($x = 0{\cdot}5$, $y = 0{\cdot}8$, $z = 0{\cdot}5$) for $Re = 3200$

conjugate gradients, we use uniform grids in the middle of the cubic cavity. The Taylor–Görtler-like vortex pairs are seen at $t = 50$ on the bottom wall. Two points at $x = 0{\cdot}5$, $y = 0{\cdot}18$, $z = 0{\cdot}5$ and $x = 0{\cdot}5$, $y = 0{\cdot}8$, $z = 0{\cdot}5$ are chosen to trace the changes in velocity components u and v with time. The time histories of velocity components u and v are illustrated in Figure 17. They show that the velocity components change with time and no steady state solution appears to be possible at $Re = 3200$. This is consistent with the numerical result of Iwatsu *et al.*[9] The LSFEM is able to simulate the unsteady Taylor–Görtler-like vortices.

It is worth mentioning that even though our simulation represents one of the largest finite element simulations using the implicit and fully coupled equations approach, we observe slight oscillations along the downstream wall ($x = 1{\cdot}0$). The oscillations are confined to the proximity of the downstream wall, but the extension of oscillations changes with time. This implies that extremely fine grids along all the surfaces of the cavity must be used and that it is preferable to use non-uniform grids for such a high-Reynolds-number flow.

4. CONCLUSIONS

A matrix-free algorithm of the LSFEM and JCG with the Crank–Nicolson scheme and Newton linearization is used to simulate three-dimensional time-dependent incompressible flows. Numerical experiments demonstrate that the matrix-free algorithm of the LSFEM with JCG is a robust method which can provide implicit, fully coupled and time-accurate numerical solutions for the Navier–Stokes equations. The three-dimensional transient simulations presented in this work show that the LSFEM with JCG has the potential to simulate realistic, large- scale, time-dependent three-dimensional fluid flow and transport processes.

It appears that literature results for $Re = 1000$ are reliable and in agreement with each other. For flow at $Re = 2000$ we can only comment that literature results are in general agreement with each other, but there is no benchmark solution yet. For flow at $Re = 3200$ it is generally agreed that there is no steady state solution.

ACKNOWLEDGEMENTS

This work was partially supported by the National Science Foundation (Grant ASC-8811171; NSF/KY EPSCoR Program). We would like to thank Dr. B. N. Jiang of NASA Lewis Research Center for useful discussion on the three-dimensional LSFEM.

REFERENCES

1. U. Ghia, K. N. Ghia and C. T. Shin, 'High-*Re* solutions for incompressible flow using the Navier–Stokes equations and multigrid method', *J. Comput. Phys.*, **48**, 387–411 (1982).
2. P. M. Gresho, S. T. Chan, R. L. Lee and C. D. Upson, 'A modified finite element method for solving the time-dependent, incompressible Navier– Stokes equations. Part 2: Applications', *Int. j. numer. methods fluids*, **4**, 619–640 (1984).
3. B. N. Jiang, 'A least-squares finite element method for incompressible Navier–Stokes problems', *Int. j. numer. methods fluids*, **14**, 843–859 (1992).
4. L. Q. Tang and T. T. H. Tsang, 'A least-squares finite element method for time-dependent incompressible flows with thermal convection', *Int. j. numer. methods fluids*, **17**, 271–289 (1993).
5. G. De Vahl Davis and G. D. Mallinson, 'An evaluation of upwind and central difference approximation by a steady recirculating flow', *Comput. Fluids*, **4**, 29–43 (1976).
6. C. J. Freitas, R. L. Street, A. N. Findikakis and J. R. Koseff, 'Numerical simulation of three-dimensional flow in a cavity', *Int. j. numer. methods fluids*, **5**, 561–575 (1985).
7. C. Y. Perng and R. L. Street, 'Three-dimensional unsteady flow simulations: alternative strategies for a volume-averaged calculation', *Int. j. numer. methods fluids*, **9**, 341–362 (1989).
8. G. Guj and F. Stella, 'A vorticity–velocity method for the numerical solution of 3D incompressible flows', *J. Comput. Phys.*, **106**, 286–298 (1993).

9. R. Iwatsu, K. Ishii, T. Kawamura, K. Kuwahara and J. M. Hyun, 'Numerical simulation of three-dimensional flow structure in a driven cavity', *Fluid Dyn. Res.*, **5**, 173–189 (1989).
10. G. A. Osswald, K. N. Ghia and U. Ghia, 'A direct algorithm for solution of incompressible three-dimensional unsteady Navier–Stokes equations', *AIAA Paper 87-1139*, 1987.
11. H. C. Ku, R. S. Hirsh and T. D. Taylor, 'A pseudospectral method for solution of the three-dimensional incompressible Navier–Stokes equations', *J. Comput. Phys.*, **70**, 439–462 (1987).
12. G. Guevremont, W. G. Habash, P. L. Kotiuga and M. M. Hafez, 'Finite element solution of the 3D compressible Navier–Stokes equations by a velocity–vorticity method', *J. Comput. Phys.*, **107**, 176–187 (1993).
13. Y. Kato, H. Kawai and T. Tanahashi, 'Numerical flow analysis in a cubic cavity by the GSMAC finite-element method (in the case where the Reynolds number are 1000 and 3200)', *JSME Int. J., Ser. II*, **33**, 649–658 (1990).
14. S. Fujima, M. Tabata and Y. Fukasawa, 'Extension to three-dimensional problems of the upwind finite element scheme based on the choice of up- and down-wind points', *Comput. Methods Appl. Mech. Eng.*, **112**, 109–131 (1994).
15. B. N. Jiang, T. L. Lin and L. A. Povinelli, 'Large-scale computation of incompressible viscous flow by least-squares finite element method', *Comput. Methods Appl. Mech. Eng.*, **114**, 213–231 (1994).
16. B. N. Jiang, C. Y. Loh and L. A. Povinelli, 'Theoretical study of incompressible Navier–Stokes equations by least-squares method', *Comput. Methods Appl. Mech. Eng.*, submitted.
17. P. P. Lynn and K. Arya, 'Use of the least-square criterion in finite element formulation', *Int. j. numer. methods eng.*, **6**, 75–88 (1973).
18. O. C. Zienkiewicz, D. R. J. Owen and K. N. Lee, 'Least-squares finite element for elastostatic problems—use of reduced integration', *Int. j. numer. methods eng.*, **8**, 341–358 (1974).
19. A. Aziz, B. Kellogg and A. Stephens, 'Least-squares methods for elliptic systems', *Math. Comput.*, **44**, 53–70 (1985).
20. D. C. Jesperson, 'A least-squares decomposition method for solving elliptic systems', *Math. Comput.*, **31**, 873–880 (1977).
21. M. D. Gunzburger, *Finite Element Methods for Viscous Incompressible Flows*, Academic, Boston, MA, 1989.
22. B. N. Jiang and L. A. Povinelli, 'Optimal least-squares finite element method for elliptic problems', *Comput. Methods Appl. Mech. Eng.*, **102**, 199–212 (1993).
23. L. Q. Tang, 'A least-squares finite element method for time-dependent fluid flows and transport phenomena', *Ph.D. Dissertation*, University of Kentucky, Lexington, KY, 1994.
24. O. C. Zienkiewicz and R. L. Taylor, *The Finite Element Method*, Vol. 1, McGraw-Hill, London, 1989.
25. G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd edn, Johns Hopkins University Press, Baltimore, MD, 1989.
26. G. Pini and G. Gambolati, 'Is a simple diagonal scaling the best preconditioner for conjugate gradients on supercomputers?', *Adv. Water Resources*, **13**, 147–153 (1990).
27. L. Q. Tang and T. T. H. Tsang, 'An efficient least-squares finite element method for incompressible flows and transport processes', *Int. J. Comput. Fluid Dyn.*, **4**, 21–39 (1995).