International Journal of Heat and Mass Transfer 52 (2009) 4082–4089

Contents lists available at ScienceDirect

International Journal of Heat and Mass Transfer

journal homepage: www.elsevier.com/locate/ijhmt

Immersed boundary computations of shear- and buoyancy-driven flows in complex enclosures

A.F. Shinn *, M.A. Goodwin, S.P. Vanka

Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, 1206 W Green Street, Urbana, IL 61801, USA

ARTICLE INFO

Article history:
Received 5 January 2009
Accepted 23 March 2009
Available online 7 May 2009

Keywords:
Immersed boundary method
Complex geometries

ABSTRACT

Cartesian grids used with the immersed boundary method (IBM) offer an attractive alternative for simulating fluid flows in complex geometries. We present a ghost fluid method for incompressible flows solved with staggered grids. The primary feature is the satisfaction of local mass continuity for ghost pressure cells, rather than extrapolating the pressures from within the flow domain. The method preserves local continuity in each cell and also global continuity. As a result, no explicit mass sources or sinks are needed. We have applied the method to study shear- and buoyancy-driven flows in a number of complex cavities.

1. Introduction

In contrast with curvilinear and unstructured grid methods for numerical computations of flow and heat transfer in complex geometries, Cartesian grid based procedures [1–5] possess the advantages of easy grid generation, energy conserving discretization properties, easy incorporation of higher-order differencing schemes and grid adaptation. As a result, there has been a significant amount of renewed interest in the CFD community in methods based on Cartesian grids. Several variants of Cartesian grid methods have been proposed in the literature. The primitive Cartesian grid method in which cells and cell faces which are located outside the flow domain are blocked is not accurate because of the stair-step representation of the boundary. The inclusion or exclusion of complete cells creates an approximate shape of the boundary on which the discrete (staggered) velocities are held fixed to the boundary values. Although, in the limit of grid refinement, the cell faces can accurately represent the boundary segments, the cell faces must coincide with the grid lines and therefore the precise location as well as the inclination of the boundary is not accounted for locally.

One method that improves upon the primitive Cartesian grid method is the Cartesian cut-cell technique [6–8]. In the Cartesian cut-cell method, the intersections of the boundary segments with the Cartesian grid are determined, and the near-boundary control volumes are re-shaped to include the physical boundary. Thus, the control volumes near the boundary become irregular polygons with areas of cell faces adjusted based on the intersections of the boundary with the cell faces. The cut-cell method has been shown by a number of researchers to improve the accuracy over the simple Cartesian grid method. However, the cut-cell method requires the computation of a number of cell intersections and becomes cumbersome in three dimensions if a complex boundary shape is considered. The problem becomes more difficult if the boundaries are moving.

Another technique to represent a complex boundary on a Cartesian grid is the immersed boundary method (IBM) pioneered by Peskin [9,10]. A considerable amount of literature exists on variants of the immersed boundary method [11–16]. In Peskin's originally proposed method, a combined Eulerian/Lagrangian approach was used. The Eulerian grid is used to compute the flow field and the boundary is prescribed by a set of Lagrangian points. First the forces at the Lagrangian points are computed from a spring stiffness model from knowledge of the displacements induced by the fluid velocities. These forces are then distributed to the discrete grid-based velocity locations through a discrete delta function. The concept of forces at the Lagrangian points has also been used to represent solid inelastic boundaries through boundary elements of very large stiffness [15]. However, when the stiffness becomes large, the time-step required to integrate the equations becomes very small as small displacements can result in large forces at the Lagrangian points. Mohd-Yusof [17] has developed a technique in which the nodes inside the solid are directly forced to imply a displacement consistent with the boundary displacement. In the approach advocated by Mohd-Yusof [17] velocities are computed at the grid locations nearest to the solid boundary as a linear interpolant of the velocity in the interior of the flow domain and the boundary velocity. These velocities are then used in the discrete momentum equations at the points in the solid to compute implied forces to be added.

* Corresponding author. Tel.: +1 217 419 0210; fax: +1 217 244 6534.
E-mail address: afshinn2@illinois.edu (A.F. Shinn).

0017-9310/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijheatmasstransfer.2009.03.044

Nomenclature

A, B, C, D	interpolation points
H	convection and diffusion terms
L	characteristic length
T	absolute temperature
V	characteristic velocity
a	generic coefficient
f	bilinear interpolation coefficient
g	acceleration due to gravity
h	height
n	normal direction
p	non-dimensional static pressure
t	non-dimensional time
u	non-dimensional x-velocity
u_{ref}	arbitrary reference velocity
v	non-dimensional y-velocity
x, y	non-dimensional Cartesian coordinates
Gr	Grashoff number $= g\beta\Delta TL^3/\nu^2$
Pr	Prandtl number $= \nu/\alpha$
Ra	Rayleigh number = Gr Pr
Re	Reynolds number $= VL/\nu$

Greek symbols

α	thermal diffusivity
β	thermal expansion coefficient
Δt	time-step
Δx	grid spacing in x-direction
Δy	grid spacing in y-direction
λ	non-dimensional half-wavelength
ν	kinematic viscosity
Φ	generic scalar
θ	non-dimensional temperature

Subscripts

B	immersed boundary point
G	ghost point
I	interior point
M	image point

Superscripts

$\hat{}$	intermediate value
n	time-level

A third approach, which appears equivalent to the direct forcing approach, is the ghost fluid technique [18–20]. In the ghost fluid model, the velocities for cells adjacent to the solid region are directly fixed by interpolating them from inside the flow domain. A typical ghost fluid method consists of first identifying the ghost points through some form of a search and tag technique. Next, for every ghost point, a normal to the nearest segment is constructed and the location of the ghost point is mirror-reflected into the flow domain. The value of the variable at this reflected location is then interpolated from the surrounding grid based values. The value at the mirror point is subsequently reflected back to the ghost location outside the flow domain. If an explicit integration procedure is used for computing the flow field, the velocity at the ghost point is computed after the completion of the fluid flow computation and is used as a boundary condition for the next time-step. For implicit formulations, one can either perform iterations at a given time-step or implicitly account for the ghost velocities in the momentum equations as boundary conditions. An important issue with ghost cell based staggered grid methods is the conservation of mass in cells which are outside the flow domain. For such cells, some of the cell face velocities may be tagged as ghost velocities and some may be inside the flow domain depending on the boundary orientation and intersections with grid lines. In several previous implementations of the IB method [19–21], the pressure at the ghost cells is extrapolated from inside of the domain by imposing a zero normal derivative at the boundary surface. However, a problem with this method is that if this pressure is subsequently used in the computation of the ghost cell face velocities, the mass continuity for these cells may not necessarily be satisfied. Mass imbalances can occur and can lead to numerical divergence. Kim et al. [22] proposed adding a source/sink term to the continuity equation to compensate for this mass defect. However, this also can cause errors in the velocity field.

In the present paper, we have used a different practice to treat mass continuity at the ghost cells. This procedure is similar to a recent method [23] advocated in the context of a semi-implicit procedure in which the mass continuity at the ghost cells is solved along with the interior equations for pressure. Further, for these ghost cells, the boundary velocity is directly substituted in the continuity equation and the ghost velocities are not used in evaluating the mass residual. This practice along with the ghost cell immersed boundary method is combined with the fractional-step procedure and second-order central differencing scheme of the derivatives in the Navier–Stokes equations. After validating the method in some simple geometries, we applied it to study shear- and buoyancy-driven flows in a variety of complex cavities. The following sections describe the details of the numerical method, and results obtained for several complex enclosures. For brevity, only a limited number of cases are presented in this paper.

2. Problems considered and governing equations

Shear-driven flows in enclosures are rich in complexity, consisting of a hierarchical organization of corner eddies frequently referred to as Moffatt eddies [24] with a nearly precise ratio of vortex strengths and distances between vortex centers. The shear-driven flow in a square cavity [25,26] has been extensively studied as a canonical problem in computational fluid mechanics. In a square cavity, the flow consists of one primary and two corner eddies which in turn generate a hierarchy of counter and co-rotating eddies. A limited number of numerical studies have also been concerned with triangular, trapezoidal, rhombic, and semicircular cavities [27–30]. Besides being of academic interest, shear-driven flows in enclosures (subsonic as well as supersonic) are of importance in aerodynamics, drag reduction, and aero-acoustics. Likewise, natural convection flows in enclosures with differential heating of two side-walls and adiabatic conditions at the other boundaries have been extensively studied. As a model problem, the double-glazing problem has been frequently used for validation and verification of CFD techniques and codes. In the double-glazing problem, the flow varies from a single regular primary vortex at low Rayleigh numbers to a time-dependent complex vortical flow at large Rayleigh numbers. While the square enclosure has enjoyed the most attention [31–35], natural convection in other complex-shaped enclosures has attracted relatively less attention. A few studies have considered rhombic and triangular enclosures as well as trapezoidal enclosures [36–39]. The general features of such flows resemble those in a square enclosure, although the shapes of the vortices are distorted by the shapes of the sloping walls. The equations representing these two classes of flows are the two-dimensional Navier–Stokes equations governing mass, momentum, and energy transport. In the current study,

these equations are solved with constant properties in conjunction with the Boussinesq approximation. The ranges of Reynolds, Prandtl, and Grashoff numbers have been limited to ensure the flows are steady in time. The non-dimensional time-dependent equations, incorporating the Boussinesq approximation can be written as:

$$\frac{\partial u}{\partial t}+\frac{\partial(uu)}{\partial x}+\frac{\partial(uv)}{\partial y}=-\frac{\partial p}{\partial x}+\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+\frac{\mathrm{Gr}_x}{\mathrm{Re}^2}\theta \tag{1}$$

$$\frac{\partial v}{\partial t}+\frac{\partial(uv)}{\partial x}+\frac{\partial(vv)}{\partial y}=-\frac{\partial p}{\partial y}+\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)+\frac{\mathrm{Gr}_y}{\mathrm{Re}^2}\theta \tag{2}$$

$$\frac{\partial \theta}{\partial t}+\frac{\partial(u\theta)}{\partial x}+\frac{\partial(v\theta)}{\partial y}=\frac{1}{\mathrm{RePr}}\left(\frac{\partial^2 \theta}{\partial x^2}+\frac{\partial^2 \theta}{\partial y^2}\right) \tag{3}$$

In the above equations, the velocities are non-dimensionalized by an arbitrary reference velocity u_{ref}. The Grashoff number Gr is defined as

$$\mathrm{Gr}=\frac{g\beta\Delta TL^3}{\nu^2} \tag{4}$$

where ΔT is the characteristic temperature difference $(T_{hot}-T_{cold})$ between the hot and cold side walls, ν is the molecular kinematic viscosity of the fluid, and β is the thermal expansion coefficient. The pressure is non-dimensionalized by the reference dynamic head, and the Reynolds number is defined by the reference velocity and cavity depth.

The above equations are supplemented with boundary conditions for velocities and temperatures. Except for the Boussinesq approximation, the properties are held constant in both the flows. The non-dimensional numbers Re and Ra are varied in a range where the flow is postulated to be steady (this was confirmed by our computations). However, the time-dependent equations were solved until a time asymptotic state was reached.

3. Numerical method

As mentioned above, we have represented the complex boundaries on a Cartesian domain and discretized the equations by a Cartesian grid. To improve accuracy at the boundaries, the immersed boundary method (IBM) using the ghost cell approach has been implemented. As previously mentioned, there are essentially two approaches in implementing the immersed boundary method. In the first approach, discrete forces are imposed in the momentum equations at selected near-boundary locations such that the appropriate velocity components reflect the boundary values at the correct boundary locations. The boundary forces are distributed to nearby locations using a discrete delta function. The discrete force method is well-suited to techniques such as finite-differences, spectral, and finite-element methods. For the finite-volume method, the force method can also be used by modifying directly the ghost cell momentum equations with a force term and iterated, if necessary, to make the ghost values equal to the pre-determined values. Alternately, the ghost cell values can be directly imposed at every time-step based on the variables computed inside the domain.

The present numerical method employs the ghost cell approach. In the ghost cell approach, the values of the variables outside the flow domain or inside the solid obstacle are fixed every time-step using the internal values and the imposed boundary values on the solid boundary. This is done by appropriately mirroring the interior values to the outside of the flow domain (or to the inside of a confined obstacle). The mirroring satisfies the condition

$$\frac{\Phi_G+\Phi_I}{2}=\Phi_B \tag{5}$$

where Φ_G is the value at the ghost node (i.e. outside of the flow domain), Φ_I is the interior value, and Φ_B is the value on the boundary. Φ can be any one of the three velocity components, temperature, or any scalar. Note that the value of pressure on the boundary is not known. Instead, a more appropriate condition is a Neumann boundary condition $\partial p/\partial n=0$.

In previous works [15,16,20,21], the pressure at the ghost nodes was evaluated using the zero normal derivative condition and was used in the computation of the velocities corresponding to such cells. However, this practice, while appropriate, leads to mass sources in the cells with fixed ghost pressures and eventually to oscillatory flow fields or even numerical divergence. In the present work, a different practice is used to evaluate the pressures in the ghost cells.

The first step in the current numerical procedure is the identification of the boundary. We have implemented the current procedure in both two and three dimensions, but we shall describe here in detail only the 2D case. In two dimensions, the geometry can be described by a number of successively connected line segments identified by the vertex numbers and the coordinates of the vertices. From the coordinates, the normals pointing into the solid (or out of the flow domain) are computed. In addition, the boundary values of velocities, temperatures, and type of temperature boundary (Dirichlet or Neumann) are prescribed.

The second step consists of a number of preprocessing calculations to aid in the determination of the value at the mirror image point. This consists of the following sub-steps:

a. *Tagging cells:* Once the Cartesian grid ranging between (x_{min},y_{min}) and (x_{max},y_{max}) is decided, the finite volume cells are tagged based on their physical boundary. Since we are using a staggered grid, separate tagging is needed for the velocity locations and the cell centers. Consider first the cell centers. For each cell center, a boundary segment which is closest (smallest perpendicular distance) is first determined. The segment number and the distance are stored and the cell is identified to be an inside or outside cell based on the sign of the distance (positive if the cell is outside the flow domain). Cells outside are tagged as zero and inside are tagged as 1.

b. *Identifying ghost cells:* Cells closest to the boundary are called the ghost cells. These are cells where a ghost value is computed and used for calculations in the interior. A ghost cell is one for which there is at least one neighbor which is inside the flow domain (to be computed in the usual way). This ghost cell is tagged with a value of 2 and its value is obtained by mirroring the interior value.

c. *Identifying the interpolation neighbors:* For each ghost cell, a mirror image point is determined by reflecting the ghost point across the boundary into the interior flow domain. The coordinates of the image points are

$$(x_M,y_M)=(x_B-[x_G-x_B],y_B-[y_G-y_B]) \tag{6}$$

where (x_B,y_B) are the coordinates of the intersection of the normal from (x_G,y_G) with the line segment. Since this point (x_M,y_M) can be in an arbitrary cell, first the cell containing this point is determined. This is done by hierarchical searching from coarse to fine grid levels. Once the cell containing the mirror point is determined, the four neighbor cells from which the mirrored value is to be determined are fixed. Here there are two methods. In several previous works, the value on the boundary is directly used in the interpolation from a bilinear or quadratic function as

$$\Phi=a_0+a_1x+a_2y+a_3xy \tag{7}$$

The four coefficients are evaluated from the values at A,B,C,D as shown in Fig. 1. This is because two points are

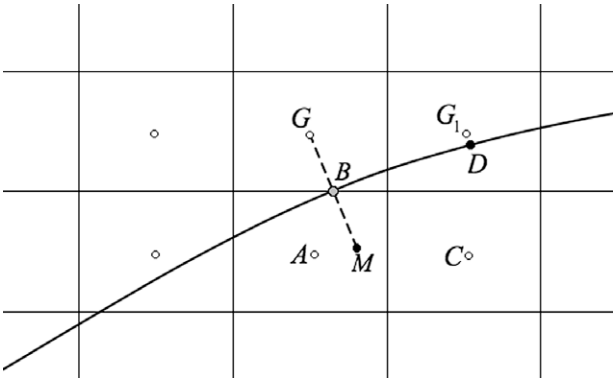

Fig. 1. Immersed boundary (curved line) overlaying grid cells. The dashed line is a perpendicular dropped from the ghost point G to the mirror point M, G_1 is another ghost point, and A, B, C, D are interpolation points. Open circles are cell centers.

on the boundary. An alternative procedure would be to use points A, C, G, G_1. Use of Φ_G to subsequently determine Φ_G from Φ_M has been observed to lead to numerical divergence if G is close to the boundary. Hence, let

$$\Phi_M = f_1\Phi_A + f_2\Phi_C + f_3\Phi_G + f_4\Phi_{G_1} \tag{8}$$

where f_1, f_2, f_3, f_4 are the bilinear interpolation coefficients. Further,

$$\Phi_G = 2\Phi_B - \Phi_M \tag{9}$$

Substituting the value of Φ_G in the previous equation,

$$\Phi_M = f_1\Phi_A + f_2\Phi_C + f_3(2\Phi_B - \Phi_M) + f_4\Phi_{G_1} \tag{10}$$

or

$$\Phi_M(1 + f_3) = f_1\Phi_A + f_2\Phi_C + 2f_3\Phi_B + f_4\Phi_{G_1} \tag{11}$$

This practice is used in the current method.

d. *Computing ghost values:* The above three steps are pre-processing steps and are done once, if the geometry is fixed and not moving. For a given set of ghost points the next step is to determine their values from the interior of the flow domain. This step implements Eq. (9) and Eq. (11) through which four neighboring values of an interior mirror image point are weighted to get Φ_G.
Note that the locations of velocities are different than the cell centers and hence different cell tags and interpolation weights are to be computed. At each time-step, the ghost x- and y-velocities are determined from the interior values.

e. *Continuity satisfaction at ghost cells:* In staggered mesh algorithms, the mass continuity is satisfied directly by the velocities on the cell faces unlike collocated schemes in which the cell face velocities are interpolated from cell centers. For ghost pressures, two practices are possible. In the first practice, as used in [20,21], the ghost pressures are extrapolated from inside by the mirror reflection procedure using a Neumann condition. However, this practice leads to mass fluxes across the solid boundaries and mass errors in the ghost cells. Even if these errors are compensated for by mass sources/sinks, oscillations are observed in the velocity field. Another approach is to directly satisfy the continuity equation for the ghost cells also and determine the pressure the usual way through the Poisson equation. However, the mass errors should not be evaluated using the ghost velocities because they are not solutions of the momentum equations. Instead, the boundary velocities must be directly substituted and the ghost velocities (outside the boundary) must be used only for the momentum equations. This practice preserves global continuity and avoids mass source/sinks in ghost cells. Recently, a similar practice was successfully used in an implicit formulation by Mark and Van Wachem [23].

f. *Solution of interior cells:* The cells that are tagged as zero are those which are outside the flow domain and also are not ghost cells. Hence, it is possible to delete them and "repack" only the cells with tags of one and two. Our current code has indirect addressing of the cells and their neighbors. However, we currently keep in memory the cells tagged zero, but skip their calculation.

For cells that are in the interior (tag = 1) we use a time-marching fractional-step procedure to evaluate velocities and pressure. The steps in this procedure are

$$\frac{\hat{u} - u^n}{\Delta t} = \frac{3}{2}H_u^n - \frac{1}{2}H_u^{n-1} \tag{12}$$

$$\frac{\hat{v} - v^n}{\Delta t} = \frac{3}{2}H_v^n - \frac{1}{2}H_v^{n-1} \tag{13}$$

$$\frac{u^{n+1} - \hat{u}}{\Delta t} = -\left(\frac{\partial p}{\partial x}\right)^{n+1} \tag{14}$$

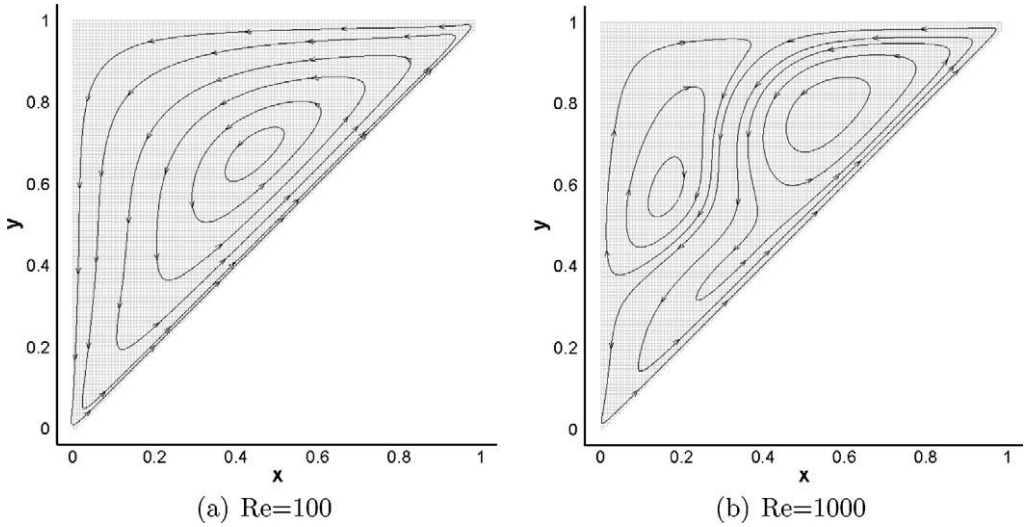

Fig. 2. Triangular cavity with the sloped wall moving for (a) Re = 100 and (b) Re = 1000.

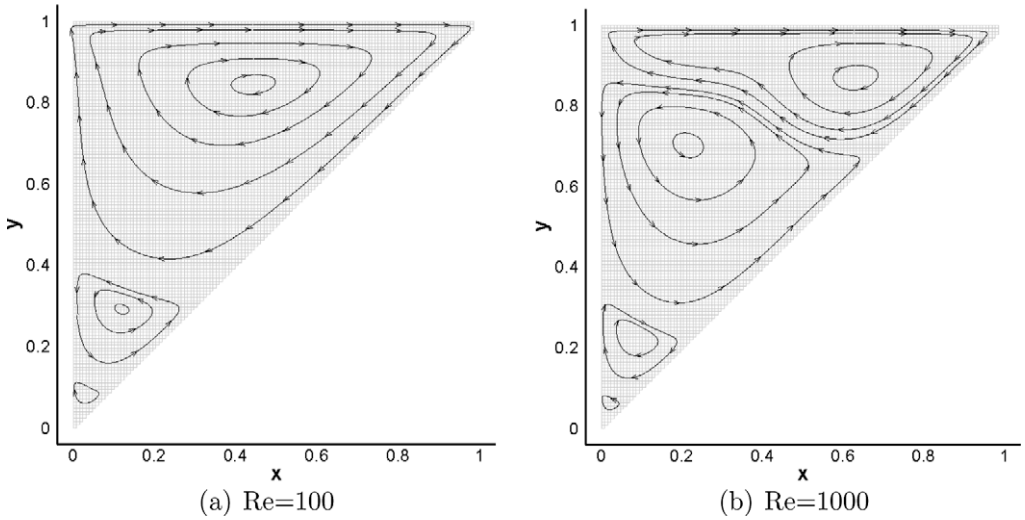

Fig. 3. Triangular cavity with the top wall moving for (a) Re = 100 and (b) Re = 1000.

$$\frac{v^{n+1}-\hat{v}}{\Delta t}=-\left(\frac{\partial p}{\partial y}\right)^{n+1} \tag{15}$$

$$\frac{\partial}{\partial x}\left(\frac{\partial p}{\partial x}\right)^{n+1}+\frac{\partial}{\partial y}\left(\frac{\partial p}{\partial y}\right)^{n+1}=\frac{1}{\Delta t}\left(\frac{\partial \hat{u}}{\partial x}+\frac{\partial \hat{v}}{\partial y}\right) \tag{16}$$

Second-order central differencing is used to discretize the spatial derivatives. The pressure Poisson equation is solved by an algebraic multigrid (AMG) procedure.

4. Results

We now present results of shear- and buoyancy-driven flows in selected complex cavities. The shapes of the geometries have been selected to simulate an interesting variety of enclosure shapes. The computational grids used in these simulations had 128×128 finite volume cells on a unit square domain, giving $\Delta x = \Delta y = 1/128$. This mesh size was selected as a compromise between accuracy and computational effort. Although finer grids could have been considered, the accuracies achieved with the present grid for bench-mark problems (driven square cavity at Re = 1000 and natural convection in a square cavity at Ra = 10^6) were within 1–2% in key quantities (maximum velocities and their locations) and the qualitative patterns did not change with further grid refinement. The discrete equations were solved until a steady flow field was reached. The following sections describe the various cases studied.

4.1. Shear-driven flows

A triangular cavity is considered first, where the two legs of the triangle (top and left walls) are of length unity, and the hypotenuse (sloped wall) is at an angle of 45°. Two cases are considered for the triangular cavity: the sloped wall is moved with a non-dimensional velocity of unity (Fig. 2) and the horizontal top wall is moved to the right at a non-dimensional velocity of unity (Fig. 3). Reynolds numbers ($\mathrm{Re} = Vh/v$) of 100 and 1000 were tested for both cases. The two cases produce considerably different flow patterns. When the sloped wall is moved tangentially at Re = 100, a single vortex is formed in the entire cavity (Fig. 2a). Corner eddies at the 90° corner are not visible at this current resolution. This eddy grows in size and significantly distorts in shape as the Reynolds number is increased to 1000 (Fig. 2b). The primary vortex is elongated in the aft section. It was found that there is no appreciable qualitative

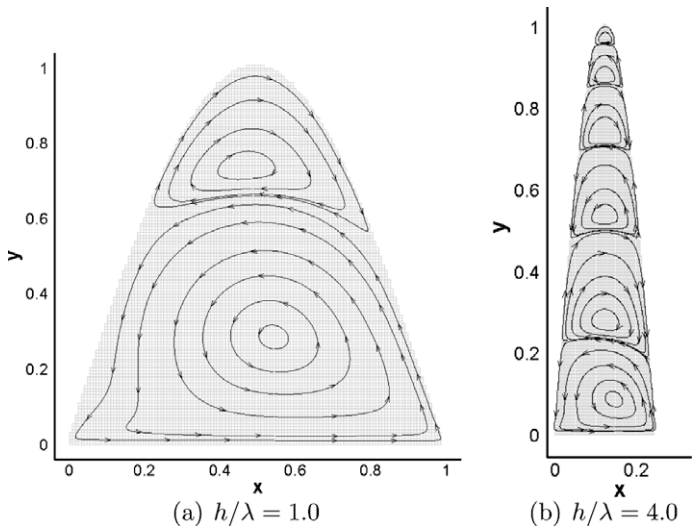

Fig. 4. Sinusoidal cavities with height to half-wavelength ratio of (a) $h/\lambda = 1.0$ and (b) $h/\lambda = 4.0$ with the bottom wall moving for Re = 1000.

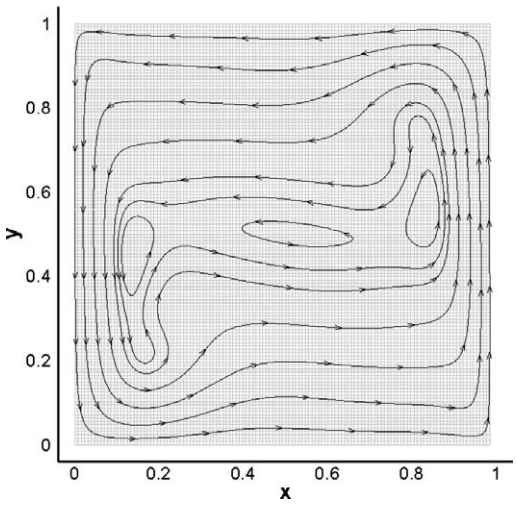

Fig. 5. Natural convection in a square cavity heated from the right side ($\theta = 1$), cooled from the left side ($\theta = 0$), and held as adiabatic along the top and bottom sides ($\mathrm{Ra} = 10^6$).

difference between the patterns at Re = 1000 and 1500 for the moving sloped wall, and thus results for Re = 1500 are not shown. Beyond Re = 1500, the flow may likely become unsteady, and hence was not pursued. For the second case of a moving top wall, a strong primary eddy is produced at Re = 100 (Fig. 3a), followed by a series of counter and co-rotating eddies at progressively diminishing strengths. Further, as the Reynolds number is increased to 1000 (Fig. 3b), the primary eddy becomes elongated with a larger head section. However, the qualitative nature of the flow remains the same from Re = 1000 onwards.

The final example that illustrates the beautiful formation of Moffatt eddies is a sinusoidal cavity. The two parameters characterizing the flow are the height to half-wavelength ratio (h/λ) and the Reynolds number ($\mathrm{Re} = Vh/\nu$). Fig. 4 shows flow patterns for two height to half-wavelength ratios at the same Reynolds number, Re = 1000. The flow patterns for the low aspect ratio show one primary eddy and a secondary eddy (Fig. 4a). However, if the aspect ratio is increased to $h/\lambda = 4.0$, there are six Moffatt eddies, although of very small strength and size (Fig. 4b). It is interesting to see that the Cartesian grid/IBM is able to capture these faithfully.

These results display the formation of the Moffatt eddies in tall enclosures (triangular and sinusoidal) where the nonlinear effects become small. In the present study, we have limited our observations to the qualitative aspects of the flow pattern formation.

4.2. Buoyancy-driven flows

Buoyancy-driven flows are abundantly found in nature and in industry. Many configurations are complex in shape, leading to interesting natural convection flow patterns. In this section, we present results of computations of natural convection flow induced in square, triangular, and circular enclosures. The natural convection is generated by heating one of the side walls and cooling the other. The top and bottom walls for the square configuration and the top wall for the triangular configuration are prescribed to be adiabatic. In addition to the momentum equations, we now solve the energy equation, and add the buoyancy source terms to the momentum equations.

Prior to computing flows in complex enclosures, we have first validated our code for the double-glazing problem [31,32]. Two Rayleigh numbers (10^5 and 10^6) were computed and compared with benchmark data of [35]. The grid used in these computations had 128×128 control volumes versus 640×640 in the reference study. Fig. 5 shows streamlines overlaid on contours of non-dimensional temperature for a Rayleigh number of 10^6. The errors were observed to be small. Also, the accuracy of the immersed boundary method was assessed by tilting the cavity by 45° and orienting the gravity vector parallel to the side walls. The results of this case (not shown here) were the same as the square cavity case after properly resolving the velocity vectors. The predicted velocities and temperatures were in good agreement with benchmark data.

Proceeding now to complex enclosures, we present a triangular cavity where the sloped wall is heated ($\theta = 1$) and the left wall is cooled ($\theta = 0$). The top wall is adiabatic. Fig. 6 shows the streamlines, overlaid on the temperature contours for Rayleigh numbers of 10^5 and 10^6. At the low Rayleigh number, the streamlines are aligned with the walls but as the Rayleigh number is increased, the streamlines become distorted.

A circular enclosure with half of the circumference maintained hot ($\theta = 1$) and the other half kept cold ($\theta = 0$) is treated next (Fig. 7). The differential heating causes a circulatory pattern with flow from the hot wall to the cold wall. In the outer region, the flow is well-aligned with the boundary, but inside vortices are formed.

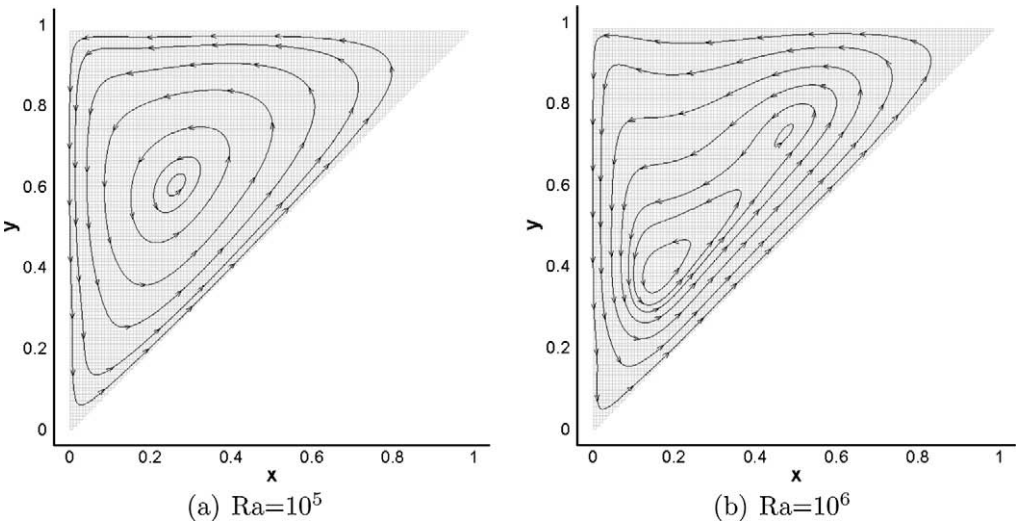

Fig. 6. Natural convection in a triangular cavity with the sloped wall heated ($\theta = 1$), left side cooled ($\theta = 0$), and top side held as adiabatic for (a) $\mathrm{Ra} = 10^5$ and (b) $\mathrm{Ra} = 10^6$.

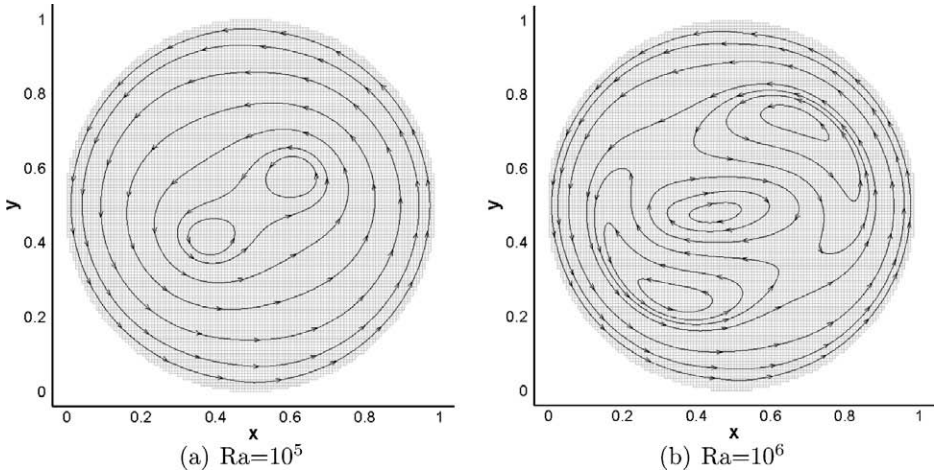

Fig. 7. Natural convection in a circular enclosure with right half heated ($\theta = 1$) and left half cooled ($\theta = 0$) at (a) Ra $= 10^5$ and (b) Ra $= 10^6$.

These vortices deform in shape as the Rayleigh number is increased. At Ra $= 10^6$, the two vortices previously seen at 10^5 increased in size and a new central vortex has formed. In the present computations, the boundary is represented by 72 linear segments spanning 5 degrees each. The outer boundary appears to be non-smooth because of the value blanking in displaying the contours.

In summary, natural convection in enclosures can generate very fascinating and complex flow patterns. Depending on the shape of the bounding surface, multiple vortices are formed at different Rayleigh numbers. There are almost an inexhaustible number of boundary shapes and parameters which can be explored for scientific interest. The present computational tool provides an easy process to represent such problems.

5. Summary and future work

The proposed method is well-suited to the study of several engineering flows in complex geometries. We have already implemented this method in three-dimensions and hope to conduct LES/DNS of turbulent flows in complex geometries. We hope to extend this method with more features such as combustion, sprays, and adaptive mesh refinement. Our eventual goal is to conduct LES of gas turbine combustor flows without extensive mesh generation.

Acknowledgment

This paper is submitted in honour of Prof. D. Brian Spalding, on the occasion of his 85th birthday.

References

[1] M. Kirkpatrick, S. Armfield, J. Kent, A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid, J. Comp. Phys. 184 (1) (2003) 1–36.
[2] J. Frymier, P.D.H. Hassan, M. Salas, Navier–Stokes calculations using Cartesian grids. I. Laminar flows, AIAA J. 26 (10) (1988) 1181–1188.
[3] E. Sanmiguel-Rojas, J. Ortega-Casanova, C. del Pino, R. Fernandez-Feria, A Cartesian grid finite-difference method for 2D incompressible viscous flows in irregular geometries, J. Comp. Phys. 204 (1) (2005) 302–318.
[4] D. Calhoun, A Cartesian grid method for solving the two-dimensional streamfunction–vorticity equations in irregular regions, J. Comp. Phys. 176 (2) (2002) 231–275.
[5] W. Coirier, K. Powell, Cartesian, cell-based approach for adaptively-refined solutions of the Euler and Navier–Stokes equations, Tech. rep., National Aeronautics and Space Administration, Lewis Research Center, Cleveland, OH, 1995.
[6] G. Feng, D. Ingram, D. Causon, C. Mingham, The development of a Cartesian cut cell method for incompressible viscous flows, Inter. J. Num. Meth. Fluids 54 (9) (2007) 1033–1053.
[7] D. Mavriplis, M. Aftosmis, M. Berger, High resolution aerospace applications using the NASA Columbia supercomputer, Int. J. High Perform. Comput. Appl. 21 (1) (2007) 106–126.
[8] M. Popescu, C.-F. Tai, W. Shyy, Cartesian cut-cell method with local grid refinement for wave computations, in: 12th AIAA/CEAS Aeroacoustics Conference, vol.3, 2006, pp. 1558–1576.
[9] C. Peskin, The immersed boundary method, Acta Numerica 11 (2002) 479–517.
[10] L. Ming-Chih, C. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comp. Phys. 160 (2) (2000) 705–719.
[11] C. Haecheon, K. Dokyun, Immersed boundary method for flow around an arbitrary moving body, J. Comp. Phys. 212 (2) (2006) 662–680.
[12] T. Sheu, H. Ting, R. Lin, An immersed boundary method for the incompressible Navier–Stokes equations in complex geometry, Int. J. Num. Meth. Fluids 56 (7) (2008) 877–898.
[13] J. Pacheco, A. Pacheco-Vega, T. Rodic, R. Peck, Numerical simulations of heat transfer and fluid flow problems using an immersed-boundary finite-volume method on nonstaggered grids, Num. Heat Transfer, B: Fundam. 48 (1) (2005) 1–24.
[14] K. Yongsam, C. Peskin, 2-D parachute simulation by the immersed boundary method, SIAM J. Sci. Comput. 28 (6) (2006) 2294–2312.
[15] I. Borazjani, L. Ge, F. Sotiropoulos, Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies, J. Comp. Phys. 227 (16) (2008) 7587–7620.
[16] G. Iaccarino, R. Verzicco, Immersed boundary technique for turbulent flow simulations, Appl. Mech. Rev. 56 (2003) 331–347.
[17] J. Mohd-Yusof, Combined immersed boundary/B-spline methods for simulations of flows in complex geometries, in: Annual Research Briefs, NASA Ames Research Center/Stanford University Center for Turbulence Research, Stanford, CA, 1997, pp. 317–327.
[18] R. Fedkiw, Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method, J. Comp. Phys. 175 (2002) 200–224.
[19] R. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method), J. Comp. Phys. 152 (1999) 457–492.
[20] Y. Tseng, J. Ferziger, A ghost-cell immersed boundary method for flow in complex geometry, J. Comp. Phys. 192 (2) (2003) 593–623.
[21] A. Gilmanov, F. Sotiropoulos, E. Balaras, A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids, J. Comp. Phys. 191 (2003) 660–669.
[22] J. Kim, D. Kim, H. Choi, An immersed-boundary finite volume method for simulations of flow in complex geometries, J. Comp. Phys. 171 (2001) 132–150.
[23] A. Mark, B.V. Wachem, Derivation and validation of a novel implicit second-order accurate immersed boundary method, J. Comp. Phys. 227 (2008) 6660–6680.
[24] H. Moffatt, Viscous and resistive eddies near a sharp corner, J. Fluid Mech. 18 (1964) 1–18.
[25] U. Ghia, K. Ghia, C. Shin, High Reynolds number solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, J. Comp. Phys. 48 (1982) 387–411.
[26] S. Vanka, Block-implicit multigrid solution of Navier–Stokes equations in primative variables, J. Comp. Phys. 65 (1) (1986) 138–158.
[27] R. Jyotsna, S. Vanka, Multigrid calculation of steady, viscous flow in a triangular cavity, J. Comp. Phys. 122 (1995) 107–117.
[28] W. McQuain, C. Ribbens, C. Wang, L. Watson, Steady viscous flow in a trapezoidal cavity, Comput. Fluids 23 (1994) 613–626.
[29] I. Demirdzic, Z. Lilek, M. Peric, Fluid flow and heat transfer test problems for non-orthogonal grids: bench-mark solutions, Int. J. Num. Meth. Fluids 15 (1992) 329–354.

[30] J. Darr, S. Vanka, Separated flow in a driven trapezoidal cavity, Phys. Fluids A 3 (3) (1991) 385–392.
[31] G. deVahlDavis, Natural convection of air in a square cavity: a bench mark numerical solution, Int. J. Num. Meth. Fluids 3 (3) (1983) 249–264.
[32] G. deVahlDavis, I. Jones, Natural convection in square cavity: a comparison exercise, Int. J. Num. Meth. Fluids 3 (3) (1983) 227–248.
[33] N. Markatos, K. Pericleous, Laminar and turbulent natural convection in an enclosed cavity, Int. J. Heat Mass Transfer 27 (5) (1984) 755–772.
[34] P. LeQuere, C. Weisman, H. Paillere, J. Vierendeels, E. Dick, R. Becker, M. Braack, M. Locke, Modeling of natural convection flows with large temperature differences: a benchmark problem for low Mach number solvers. Part 1. Reference solutions, Math. Model. Num. Anal. 39 (3) (2005) 609–616.
[35] M. Hortmann, M. Peric, G. Scheuerer, Finite volume multigrid prediction of laminar natural convection: bench-mark solutions, Int. J. Num. Meth. Fluids 11 (2) (1990) 189–207.
[36] P. Nithiarasu, T. Sundararajan, K. Seetharamu, Finite element analysis of transient natural convection in an odd-shaped enclosure, Int. J. Num. Meth. Heat Fluid Flow 8 (2) (1998) 199–216.
[37] Y. Morsi, S. Das, Numerical investigation of natural convection inside complex enclosures, Heat Transfer Eng. 24 (2) (2003) 30–41.
[38] S. Volker, T. Burton, S. Vanka, Finite-volume multigrid calculation of natural-convection flows on unstructured grids, Num. Heat Transfer, B: Fundam. 30 (1) (1996) 1–22.
[39] G. Iaccarino, S. Moreau, Natural and forced conjugate heat transfer in complex geometries on Cartesian adapted grids, J. Fluids Eng., Transact. ASME 128 (4) (2006) 838–846.