J Math Chem (2015) 53:844–867
DOI 10.1007/s10910-014-0461-7

ORIGINAL PAPER

Sensitivity analysis of a mesh refinement method using the numerical solutions of 2D lid-driven cavity flow

Rajnesh Lal · Zhenquan Li

Received: 17 July 2014 / Accepted: 15 December 2014 / Published online: 24 December 2014

Abstract Lid-driven cavity flows have been widely investigated and accurate results have been achieved as benchmarks for testing the accuracy of computational methods. This paper investigates sensitivity of a mesh refinement method against the accuracy of numerical solutions of the 2-D steady incompressible lid-driven flow from a collocated finite volume method. The sensitivity analysis is shown by comparing the coordinates of centres of primary and secondary vortices located by the mesh refinement method with the corresponding benchmark results. The accuracy of the numerical solutions is shown by comparing the profiles of horizontal and vertical components of velocity fields with the corresponding benchmarks and the streamlines. The sensitivity analysis shows that the mesh refinement method provides accurate coordinates of primary and secondary vortices depending on the accuracy of the numerical solutions. The adaptive mesh refinement method considered can be applied to incompressible fluid or steady state fluid flows or mass and heat transfer.

Keywords Mesh refinement · Mass conservation · Lid-driven cavity flow · Collocated finite volume method

1 Introduction

Meshing is the process of breaking up a physical domain into smaller sub-domains (elements or cells) in order to evaluate the numerical solution of differential equations.

R. Lal (✉)
School of Mathematical and Computing sciences, Fiji National University, Suva, Fiji Islands
e-mail: rajnesh.lal@fnu.ac.fj

Z. Li
School of Computing and Mathematics, Charles Sturt University, Albury, NSW 2640, Australia
e-mail: jali@csu.edu.au

Adaptive mesh refinement is a computational technique to improve the accuracy of numerical solutions of differential equations by starting the calculations on a coarse basic mesh (initial mesh) and then refining this mesh only where less accuracy may occur locally.

There are a large number of publications on mesh adaptive refinements and their applications. Some refinement methods use a refinement criterion which is based on local truncation errors (e.g. [1,3–5]). Other common methods include h-refinement (e.g. [20,23]), p-refinement (e.g. [3,25]) or r-refinement (e.g. [21,22]), with different combinations of these also possible (e.g. [6,7]). The overall aim of these adaptive algorithms is to obtain a balance between accuracy and computational efficiency. The h-refinement is a method where meshes are refined and/or coarsened to achieve a prescribed accuracy and efficiency. The p-refinement is a method where the accuracy orders are assigned to elements to achieve exponential convergence rates and r-refinement is a method where elements are moved and redistributed to track evolving non-uniformities.

We introduced adaptive mesh refinement methods from a different point of view for two-dimensional velocity fields [19] and for three-dimensional fields [18] based on a theorem in qualitative theory of differential equations (Theorem 1.14, page 18, [24]). The theorem indicates that a 2D divergence free vector field has no limit cycles or one sided limit cycles, that is, the trajectories (or streamlines) of divergence free vector fields are closed curves in bounded domains. Identification of accurate locations of singular points (singular points are streamlines) and asymptotic lines (planes), and drawing closed streamlines are some of the accuracy measures for computational methods. We consider these accuracy measures in the mesh refinement methods we proposed. The adaptive mesh refinement methods adaptively refine cells where the linear interpolation of the evaluated numerical velocity fields is not a divergence free vector field. Using numerical velocity fields obtained by taking the vectors of the analytical velocity fields at nodes of meshes, examples showing the accuracy of the methods include: locating the singular points and asymptotic lines for two-dimensions [19]; the singular points and asymptotic plane for three-dimensions [18]; and drawing closed streamlines (Li [16,17]) using the refined meshes with a pre-specified number of refinements of the initial meshes. The examples also showed that the Lebesgue measure of the set of cells on which the linear interpolation of the evaluated numerical velocity fields is not divergence free becomes smaller with the increase of the number of mesh refinements. However, it is impossible to achieve such numerical velocity fields in practice. The sensitivity analysis for the adaptive mesh refinement methods for the numerical velocity fields obtained by solving mathematical models numerically is necessary before applying methods in practice.

This paper considers the sensitivity analysis of the 2D adaptive mesh refinement method using the numerical velocity fields of 2D lid-driven cavity flows obtained from solving Navier–Stokes equations numerically with the boundary conditions. We solve the Navier–Stokes equations with the boundary conditions numerically using a second order collocated finite volume method (FVM) with a splitting method for time discretization [9] and then apply the refinement method to the numerical solutions once and then compare the results with the corresponding benchmarks. The comparisons show that the adaptive mesh refinement method provides the similar results to those

for the analytical velocity fields [19]. The accuracy of the coordinates of vortices mentioned above fully depends on the accuracy of the numerical solutions. This provides the reliability of the mesh refinement method when it is applied to solve problems. In the following discussion, we consider incompressible fluid only but the discussion is the same for steady compressible flows by replacing the linearly interpolated velocity field $\mathbf{V}_l$ with linearly interpolated momentum vector $\rho\mathbf{V}_l$ (where ρ is the density).

2 Algorithm of mesh refinement

This section briefly summarizes the 2D adaptive mesh refinement method [19] and gives detailed algorithms.

Assume that $\mathbf{V}_l = \mathbf{AX} + \mathbf{B}$ is a vector field obtained by linearly interpolating the vectors at the three vertexes of a triangle, where

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \quad \text{and} \quad \mathbf{B} = \begin{pmatrix} b_1' \\ b_2' \end{pmatrix}$$

are constant matrices and vertical vector respectively, and $\mathbf{X} = (x_1, x_2)^T$. The vector $\mathbf{V}_l$ is unique if the area of the triangle is not zero [14]. Mass conservation for a steady flow or an incompressible fluid requires that

$$\nabla \cdot \mathbf{V}_l = \text{trace}(\mathbf{A}) = 0. \tag{1}$$

Let f be a scalar function depending only on spatial variables. We assume that $f\mathbf{V}_l$ satisfies Eq. (1) and then calculate the expressions of f. The expressions of f were derived for the four different Jacobian forms of coefficient matrix $\mathbf{A}$ as shown in Table 1 [19]. Variables y_1 and y_2 in Table 1 are the components of $(y_1, y_2)^T = \mathbf{V}^{-1}\mathbf{X}$ where $\mathbf{V}$ satisfies $\mathbf{AV} = \mathbf{VJ}$ and $\mathbf{J}$ is one of the Jacobian matrices in Table 1. The introduction of scalar function f reduces the number of refined cells but $f\mathbf{V}_l$ and $\mathbf{V}_l$ produce the same streamlines if $f \neq 0$ or ∞ [15]. The introduction of the adaptive mesh refinement is to achieve refined meshes on which $\mathbf{V}_l$ or $f\mathbf{V}_l$ are not divergence free in a set of cells with controllable small Lebesgue measure.

Table 1 Jacobian matrices and corresponding expressions of f ($C \neq 0$)

Case	Jacobian	f
1	$\begin{pmatrix} r_1 & 0 \\ 0 & r_2 \end{pmatrix}$ $(0 \neq r_1 \neq r_2 \neq 0)$	$\frac{C}{\left(y_1+\frac{b_1}{r_1}\right)\left(y_2+\frac{b_2}{r_2}\right)}$
2	$\begin{pmatrix} r_1 & 0 \\ 0 & 0 \end{pmatrix}$ $(r_1 \neq 0)$	$\frac{C}{y_1+\frac{b_1}{r_1}}$
3	$\begin{pmatrix} r_1 & 0 \\ 0 & r_1 \end{pmatrix}$ $(r_1 \neq 0)$	$\frac{C}{\left(y_1+\frac{b_1}{r_1}\right)^2}$
4	$\begin{pmatrix} \mu & \lambda \\ -\lambda & \mu \end{pmatrix}$ $(\mu \neq 0, \lambda \neq 0)$	$\frac{C}{\left(y_1+\frac{\mu b_1-\lambda b_2}{\mu^2+\lambda^2}\right)^2+\left(y_2+\frac{\lambda b_1+\mu b_2}{\mu^2+\lambda^2}\right)^2}$

The conditions (MC)(MC is the abbreviation of mass conservation) are the functions f in Table 1 not equalling zero or infinity at any point on the triangular domains when $f\mathbf{V}_l$ is divergence free on these triangular domains.

We describe the algorithm of adaptive mesh refinement for quadrilateral mesh in this paper. The algorithm is also applicable to triangular meshes. The following cell refinement algorithm describes how to use the conditions (MC) to refine a quadrilateral cell in a given mesh. To avoid an infinite refinement of the mesh, we choose a pre-specified threshold number of refinements T based on the accuracy requirements. The algorithm of cell refinement is:

Step 1 Subdivide a quadrilateral cell into two triangles and check if $\mathbf{V}_l$ satisfies Eq. (1) on both triangles. If yes, no refinement for the cell is required. If no, go to Step 2.

Step 2 Apply the conditions (MC) to both of the triangles. If the conditions (MC) are satisfied on both triangles, there is no need to subdivide the cell. Otherwise, we subdivide the cell into a number of small cells such that the lengths of all sides of the small cells are truly reduced.

The algorithm of adaptive mesh refinement is:

Step 1 Evaluate the numerical velocity field for a given initial mesh;

Step 2 Refine the cells of the mesh one by one using the above cell refinement algorithm;

Step 3 Take the refined mesh as initial mesh and go to Step 1 until a satisfactory numerical velocity field is obtained or the threshold number T is reached.

In this paper, we subdivide a quadrilateral cell by connecting the mid-points of the two opposite sides of a quadrilateral and the threshold number $T = 1$, that is, we subdivide a cell once only for testing the sensitivity of the adaptive mesh refinement method. Quadrilateral and triangular meshes are the commonly used 2D meshes but the application of the conditions (MC) is harder to quadrilateral than to triangular. We show the harder case. The abbreviations BR, BL and TL refer to bottom right, bottom left and top left corners of the cavity, respectively. The number following these abbreviations refer to the vortices that appear in the flow, which are numbered according to size (for example, BR1 refers to bottom right secondary vortex).

3 Collocated finite volume scheme with a splitting method for the time discretization

In this section, we briefly review the colocated finite volume scheme used for evaluating the numerical solutions of 2D lid-driven cavity flows [9].

3.1 Navier–Stokes equations for incompressible fluids

For given volume force $\mathbf{f} = (f_u, f_v)$, we look for the velocity field $\mathbf{u}$ and the pressure p that satisfy

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\,\mathbf{u} + \nabla p = \mathbf{f}, \tag{2}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{3}$$

in $\Omega \times [0, T]$, where $\nu > 0$ is the kinematic viscosity, $\mathbf{u} = (u(x, y, t), v(x, y, t))$, and $t \geq 0$. On the boundary $\partial\Omega$ of Ω, a Dirichlet no-slip boundary condition is used

$$\mathbf{u}|_{\partial\Omega} = \mathbf{g}.$$

3.2 Time discretization

The time discretization for (2) and (3) is

$$\frac{3\mathbf{u}^{n+1} - 4\mathbf{u}^{n} + \mathbf{u}^{n-1}}{2\Delta t} - \nu \Delta \mathbf{u}^{n+1} + \hbar\left(\mathbf{u}^{n}, \mathbf{u}^{n-1}\right) + 2\nabla p^{n} - \nabla p^{n-1} = \mathbf{f}^{n+1}, \tag{4}$$

where $\hbar\left(\mathbf{u}^{n}, \mathbf{u}^{n-1}\right) = 2\left(\mathbf{u}^{n} \cdot \nabla \mathbf{u}^{n}\right) - \left(\mathbf{u}^{n-1} \cdot \nabla \mathbf{u}^{n-1}\right)$.

We calculate $\mathbf{u}^{n+1}$ using (4) with the boundary condition $\mathbf{u}|_{\partial\Omega} = \mathbf{g}$ and compute the pressure p^{n+1} from

$$\begin{cases} \Delta \psi^{n+1} = \nabla \cdot \left(\frac{3\mathbf{u}^{n+1} - 4\mathbf{u}^{n} + \mathbf{u}^{n-1}}{2\Delta t}\right) \\ \frac{\partial \psi^{n+1}}{\partial \mathbf{n}} = 0 \end{cases}$$

$$p^{n+1} = \psi^{n+1} + 2p^{n} - p^{n-1} - \nu \nabla \cdot \mathbf{u}^{n+1}.$$

3.3 Finite volume discretization

We summarize the steps of the algorithm.

3.3.1 Compute the new velocity field $\mathbf{u}^{n+1} = \left(u^{n+1}, v^{n+1}\right)$

$$\begin{aligned} &\Delta x \Delta y \frac{3\mathbf{u}_{ij}^{n+1} - 4\mathbf{u}_{ij}^{n} + \mathbf{u}_{ij}^{n-1}}{2\Delta t} - \nu \left[\Delta y \frac{\mathbf{u}_{i+1j}^{n+1} - \mathbf{u}_{ij}^{n+1}}{\Delta x} + \Delta y \frac{\mathbf{u}_{i-1j}^{n+1} - \mathbf{u}_{ij}^{n+1}}{\Delta x} \right. \\ &\left. + \Delta x \frac{\mathbf{u}_{ij+1}^{n+1} - \mathbf{u}_{ij}^{n+1}}{\Delta y} + \Delta x \frac{\mathbf{u}_{ij-1}^{n+1} - \mathbf{u}_{ij}^{n+1}}{\Delta y} \right] \\ &+ \aleph\left(p^{n}, p^{n-1}\right) + \hbar\left(\mathbf{u}^{n}, \mathbf{u}^{n-1}\right)_{ij} = \Delta x \Delta y \mathbf{f}_{ij}^{n+1} \end{aligned}$$

where

$$\aleph\left(p^{n}, p^{n-1}\right) = 2 \begin{bmatrix} \frac{\Delta y}{2}\left(p_{i+1j}^{n} - p_{i-1j}^{n}\right) \\ \frac{\Delta x}{2}\left(p_{ij+1}^{n} - p_{ij-1}^{n}\right) \end{bmatrix} - \begin{bmatrix} \frac{\Delta y}{2}\left(p_{i+1j}^{n-1} - p_{i-1j}^{n-1}\right) \\ \frac{\Delta x}{2}\left(p_{ij+1}^{n-1} - p_{ij-1}^{n-1}\right) \end{bmatrix}$$

and

$$\begin{aligned}\hbar\left(\mathbf{u}^n,\mathbf{u}^{n-1}\right)=&\Delta y\left(2F^n_{ui+1/2j}-F^{n-1}_{ui+1/2j}\right)\left(2\frac{\mathbf{u}^n_{i+1j}+\mathbf{u}^n_{ij}}{2}-\frac{\mathbf{u}^{n-1}_{i+1j}+\mathbf{u}^{n-1}_{ij}}{2}\right)\\&-\Delta y\left(2F^n_{ui-1/2j}-F^{n-1}_{ui-1/2j}\right)\left(2\frac{\mathbf{u}^n_{i-1j}+\mathbf{u}^n_{ij}}{2}-\frac{\mathbf{u}^{n-1}_{i-1j}+\mathbf{u}^{n-1}_{ij}}{2}\right)\\&+\Delta x\left(2F^n_{vij+1/2}-F^{n-1}_{vij+1/2}\right)\left(2\frac{\mathbf{u}^n_{ij+1}+\mathbf{u}^n_{ij}}{2}-\frac{\mathbf{u}^{n-1}_{ij+1}+\mathbf{u}^{n-1}_{ij}}{2}\right)\\&-\Delta x\left(2F^n_{vij-1/2}-F^{n-1}_{vij-1/2}\right)\left(2\frac{\mathbf{u}^n_{ij-1}+\mathbf{u}^n_{ij}}{2}-\frac{\mathbf{u}^{n-1}_{ij-1}+\mathbf{u}^{n-1}_{ij}}{2}\right)\end{aligned}$$

3.3.2 Compute the pressure p^{n+1}

$$\begin{aligned}p^{n+1}_{ij}=\psi^{n+1}_{ij}+2p^n_{ij}-p_{ij}-\frac{\nu}{\Delta x\Delta y}\Big[&\Delta y\left(F^{n+1}_{ui+1/2j}-F^{n+1}_{ui-1/2j}\right)\\&+\Delta x\left(F^{n+1}_{vij+1/2}-F^{n+1}_{vij-1/2}\right)\Big]\end{aligned}$$

where ψ^{n+1} is computed from

$$\begin{aligned}&\Delta y\frac{\psi^{n+1}_{i+1j}-\psi^{n+1}_{ij}}{\Delta x}+\Delta y\frac{\psi^{n+1}_{i-1j}-\psi^{n+1}_{ij}}{\Delta x}+\Delta x\frac{\psi^{n+1}_{ij+1}-\psi^{n+1}_{ij}}{\Delta y}\\&+\Delta x\frac{\psi^{n+1}_{ij-1}-\psi^{n+1}_{ij}}{\Delta y}\\&=\frac{1}{2\Delta t}\Big\{\Delta y\Big[\left(3F^{n+1}_{ui+1/2j}-4F^n_{ui+1/2j}+F^{n-1}_{ui+1/2j}\right)\\&\quad-\left(3F^{n+1}_{ui-1/2j}-4F^n_{ui-1/2j}+F^{n-1}_{ui-1/2j}\right)\Big]\\&\quad+\Delta x\Big[\left(3F^{n+1}_{vij+1/2}-4F^n_{vij+1/2}+F^{n-1}_{vij+1/2}\right)\\&\quad-\left(3F^{n+1}_{vij-1/2}-4F^n_{vij-1/2}+F^{n-1}_{vij-1/2}\right)\Big]\Big\}\end{aligned}$$

With Neumann boundary condition

$$\psi^{n+1}_{M+1j}=\psi^{n+1}_{Mj},\quad \psi^{n+1}_{0j}=\psi^{n+1}_{1j},\quad \psi^{n+1}_{iN+1}=\psi^{n+1}_{iN},\quad \psi^{n+1}_{i0}=\psi^{n+1}_{i1},$$

where M and N are the number of lines inserted in x interval and y interval of the domain equally.

3.3.3 Compute the flux F^{n+1}

$$F^{n+1}_{ui+1/2j} = \frac{u^{n+1}_{i+1j} + u^{n+1}_{ij}}{2} + \theta\frac{\Delta y}{4a}\left(p^n_{i+2j} - 2p^n_{i+1j} + p^n_{ij}\right) - \theta\frac{\Delta y}{4a}\left(p^n_{i+1j} - 2p^n_{ij} + p^n_{i-1j}\right)$$

$$F^{n+1}_{vij+1/2} = \frac{v^{n+1}_{ij+1} + v^{n+1}_{ij}}{2} + \theta\frac{\Delta x}{4a}\left(p^n_{ij+2} - 2p^n_{ij+1} + p^n_{ij}\right) - \theta\frac{\Delta x}{4a}\left(p^n_{ij+1} - 2p^n_{ij} + p^n_{ij-1}\right)$$

where $\theta = 1/4$ as a relaxation coefficient and

$$a = \left(\frac{\Delta x\Delta y}{\Delta t} + \nu\frac{\Delta y}{\Delta x} + \nu\frac{\Delta x}{\Delta y}\right)$$

The stop criterion for the calculations in this paper is $\|\mathbf{u}^{n+1} - \mathbf{u}^n\| < 10^{-6}$.

4 Control volume face centred mesh

In this section, we present the initial mesh used to evaluate the numerical solutions of 2D lid-driven cavity flows. Since the mesh refinement method is for the numerical velocity fields given at the nodes of the initial mesh, we must construct a new mesh for the colocated finite volume method such that the nodes (stars at the intersection of the solid lines) of the initial mesh locate in the midway between faces of control volumes of the new mesh (dashed lines inside of the unit square in Fig. 1) [10]. To

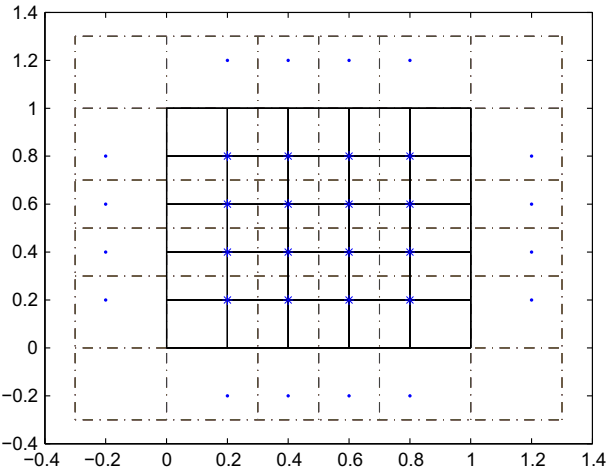

Fig. 1 Control volume face centred mesh

facilitate the boundary conditions accurately, we extend the initial mesh and add more nodes (the dots) in the outside region of the unit square as shown in Fig. 1. We apply the boundary conditions to the average of the velocity field at the nodes outside of unit square and those inside of the square symmetrical to them. This arrangement provides second order accuracy of the boundary conditions in the evaluations of the numerical solutions. The mesh refinement algorithm applies to the numerical velocity fields at the nodes inside of unit square and on the boundaries evaluated by the given boundary conditions.

5 Sensitivity analysis

In the analysis of the 2D adaptive mesh refinement for numerical velocity fields obtained from analytical velocity fields, we have previously shown that the singular points (centres of vortices) of the velocity fields are contained inside of refined cells and asymptotic lines are located inside of blocked refined cells [19]. A cell is said to be a refined cell if a cross is drawn inside. In this section, we provide the same information for numerical velocity fields achieved from solving Navier–Stokes equations.

We consider the refined meshes for two-dimensional lid-driven cavity flows for different mesh sizes and Reynolds number Re $=$ 100, 1,000, and 2,500, respectively. We show horizontal and vertical velocity profiles, refined meshes, and refined meshes with streamlines. The streamlines are generated by Matlab built-in function *streamline*. In each subsection, we consider two initial meshes for one Reynolds number: one coarser and one finer. We take the coordinates of intersections of crosses in isolated refined cells in refined meshes as the estimates for centres of vortices. The figures of horizontal velocity fields at $x = 0.5$ and vertical velocity profiles at $y = 0.5$ with their corresponding benchmarks show the partial accuracy of numerical velocity fields. Matlab built-in function *streamline* generate streamlines from vector data. Therefore, the figures of streamlines show the global accuracy of numerical velocity fields given by verifying whether the streamlines are closed. The sensitivity of mesh refinement are indicated in the figures of refined meshes and Tables 2 and 3.

Table 2 Primary and second vortex centre locations for coarser mesh sizes

Vortex type	Reynolds numbers		
	Re = 100 (41 × 41)	Re = 1,000 (69 × 69)	Re = 2,500 (91 × 91)
Primary vortex	(0.6463,0.7683)	(0.5435,0.5870)	(0.5355,0.5550)
	(0.6172,0.7344)	*(0.5300,0.5650)*	*(0.5200,0.5433)*
BR1	(0.9634,0.0610)	(0.8624,0.1232)	(0.8297,0.0934)
	(0.9453,0.0625)	*(0.8633,0.1117)*	*(0.8350,0.0917)*
BL1	–	(0.0797,0.0797)	(0.08242,0.1044)
	(0.0313,0.0391)	*(0.0833,0.0783)*	*(0.0850,0.1100)*
TL1	–	–	(0.03847,0.8956)
	–	–	*(0.0433,0.8900)*

Table 3 Primary and second vortex centre locations for mesh size 121 × 121

Vortex type	Reynolds numbers		
	$Re = 100$	$Re = 1{,}000$	$Re = 2{,}500$
Primary vortex	(0.6240,0.7479)	(0.5331,0.5744)	(0.5248,0.5496)
	(0.6172,0.7344)	*(0.5300,0.5650)*	*(0.5200,0.5433)*
BR1	(0.9463,0.0620)	(0.8636,0.1198)	(0.8306,0.0950)
	(0.9453,0.0625)	*(0.8633,0.1117)*	*(0.8350,0.0917)*
BL1	(0.0372,0.0372)	(0.0868,0.0785)	(0.0868,0.1033)
	(0.0313,0.0391)	*(0.0833,0.0783)*	*(0.0850,0.1100)*
TL1	–	–	(0.0455,0.8967)
	–	–	*(0.0433,0.8900)*

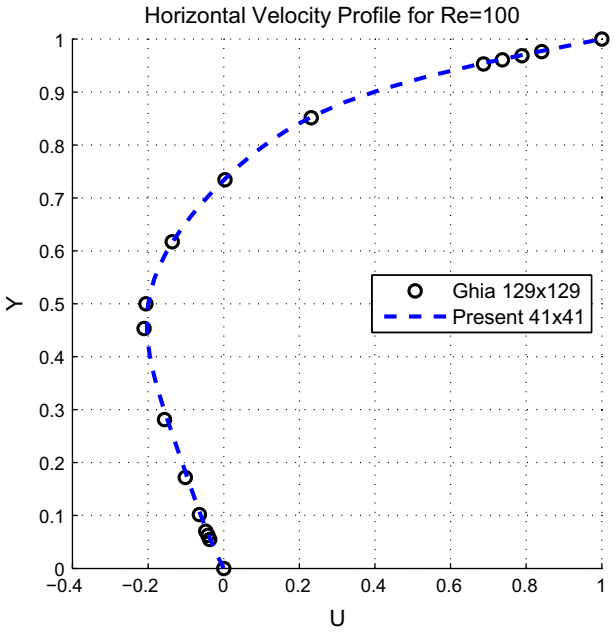

Fig. 2 Horizontal profile of velocity field for mesh size 41 × 41

5.1 Re = 100

Figure 2 shows the horizontal velocity profile and Fig. 3 shows the vertical velocity profile for mesh size 41 × 41 together with the corresponding benchmark results [11]. Even though both figures show right shapes of the two components, there are errors especially for vertical component which may not be acceptable if the required accuracy of the computations is small. From Figs. 4 and 5, we find two centres of vortices: primary and secondary bottom right (BR1) without difficulty. The streamlines shown in Fig. 5 are not closed and this indicates that the linearly interpolated

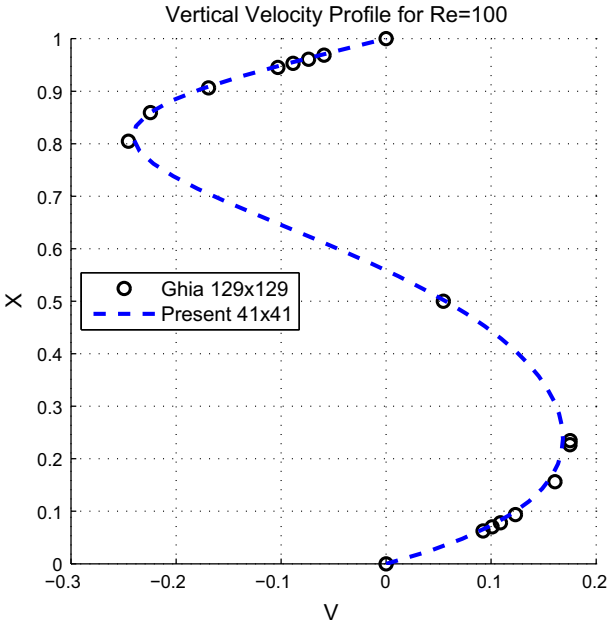

Fig. 3 Vertical profile of velocity field for mesh size 41 × 41

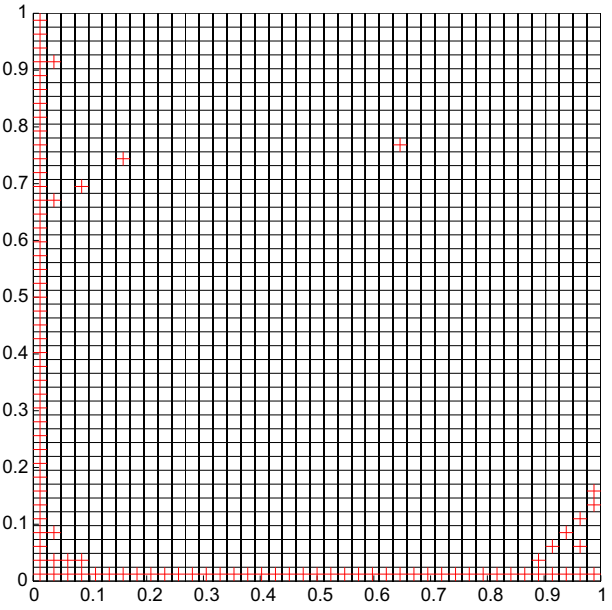

Fig. 4 Refined mesh for mesh size 41 × 41

numerical velocity field $\mathbf{V}_l$ or $f\mathbf{V}_l$ do not satisfy Eq. (1) [13]. Figure 4 shows the refined cells (containing crosses) where $f\mathbf{V}_l$ or $\mathbf{V}_l$ do not satisfy Eq. (1). If the interpolated velocity field $\mathbf{V}_l$ or $f\mathbf{V}_l$ satisfy Eq. (1) on all cells, all streamlines

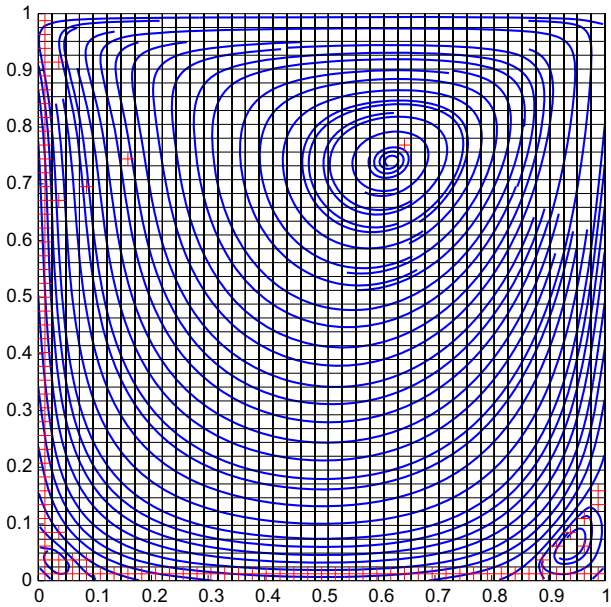

Fig. 5 Refined mesh with streamlines for mesh size 41 × 41

are closed. The estimates of centre of vortices for mesh size 41 × 41 are shown in Table 2.

Figures 6 and 7 show the horizontal and vertical velocity profiles, respectively, for mesh size 121 × 121 together with the corresponding benchmark results [11]. The accuracy of the computations has been greatly improved from these two figures. From Figs. 8 and 9, we find three centres of vortices: primary, secondary bottom left (BL1) and right (BR1). Even though the streamlines shown in both Figs. 5 and 9 are not closed, the streamlines in Fig. 9 are spiral much closer. Figure 8 is the same as Fig. 4 showing the refined cells where $\mathbf{V}_l$ or $f\mathbf{V}_l$ do not satisfy Eq. (1). The estimates of centre of vortices for mesh size 121 × 121 are shown in Table 3. The centre of BL1 is not identified by mesh size 41 × 41 but it is identified by mesh size 121 × 121.

5.2 Re = 1,000

Since Reynolds number Re increases, finer mesh is required to produce reliable results [11]. We replace mesh size 41 × 41 by 69 × 69. Figures 10 and 11 show horizontal and vertical velocity profiles respectively for mesh size 69 × 69 together with the corresponding benchmark results [8]. It is the same as Figs. 2 and 3 that even though both figures show right shapes of the two components, there are errors especially for vertical component which may not be acceptable. Based on the numerical velocity fields with the errors shown in Figs. 10 and 11, three centres of vortices: primary, secondary bottom right (BR1) and left (BL1) are found and shown in Figs. 12 and 13.

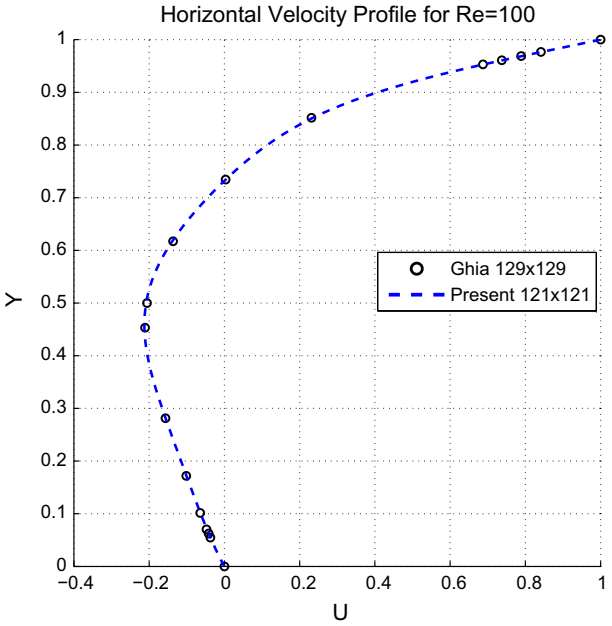

Fig. 6 Horizontal profile of velocity field for mesh size 121 × 121

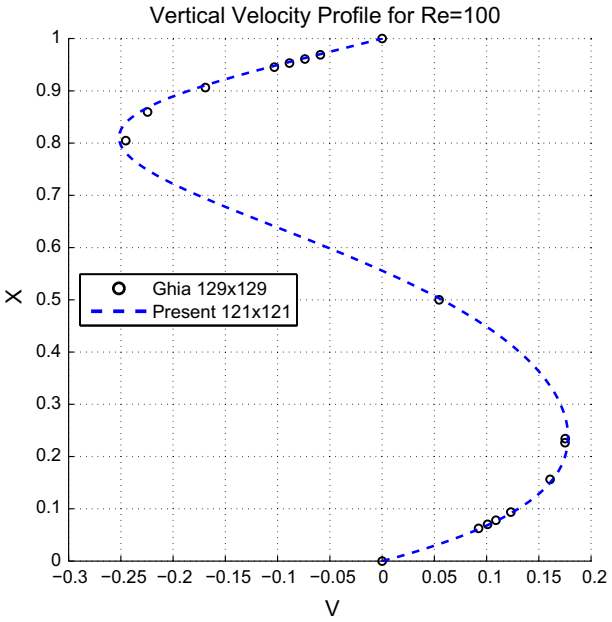

Fig. 7 Vertical profile of velocity field for mesh size 121 × 121

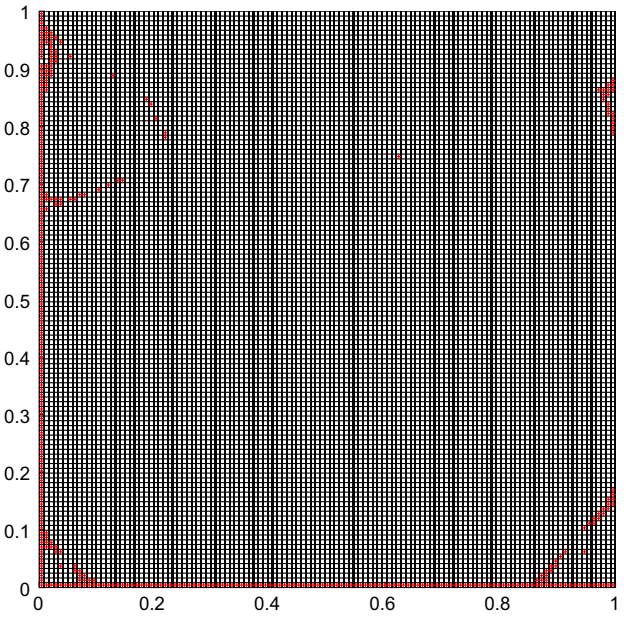

Fig. 8 Refined mesh for mesh size 121 × 121

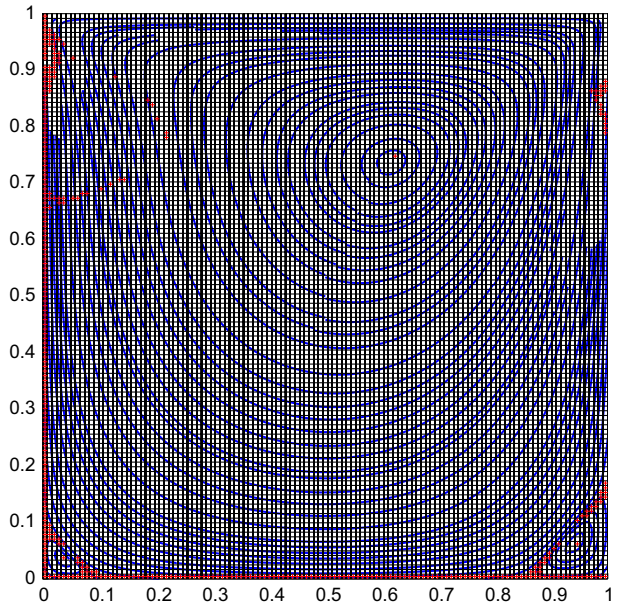

Fig. 9 Refined mesh with streamlines for mesh size 121 × 121

The streamlines shown in Fig. 13 are not closed even though they are better than those in Fig. 5 and this indicates that the interpolated numerical velocity field $\mathbf{V}_l$ or $f\mathbf{V}_l$ do not satisfy Eq. (1) [13]. Figure 12 shows the refined mesh which contains the refined

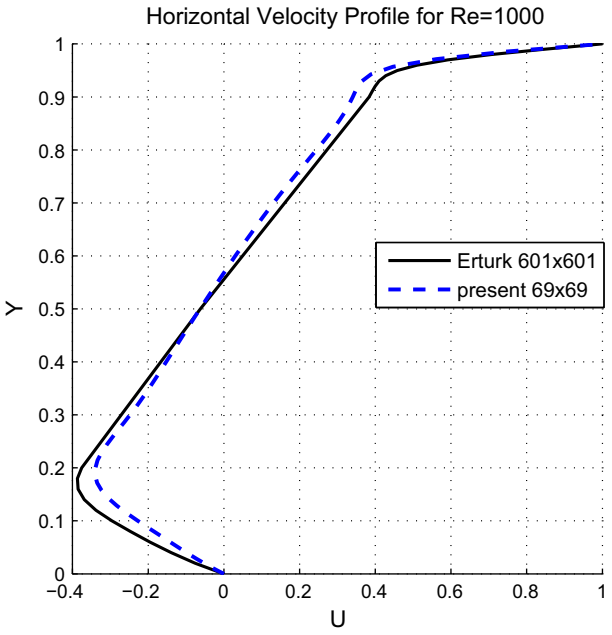

Fig. 10 Horizontal profile of velocity field for mesh size 69 × 69

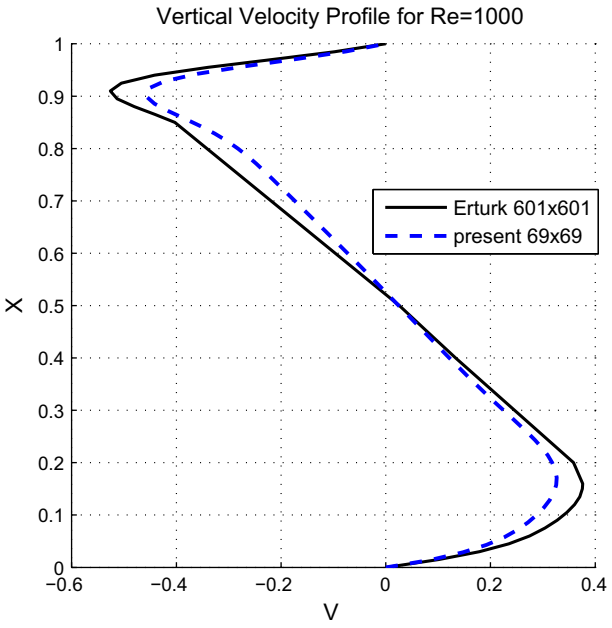

Fig. 11 Vertical profile of velocity field for mesh size 69 × 69

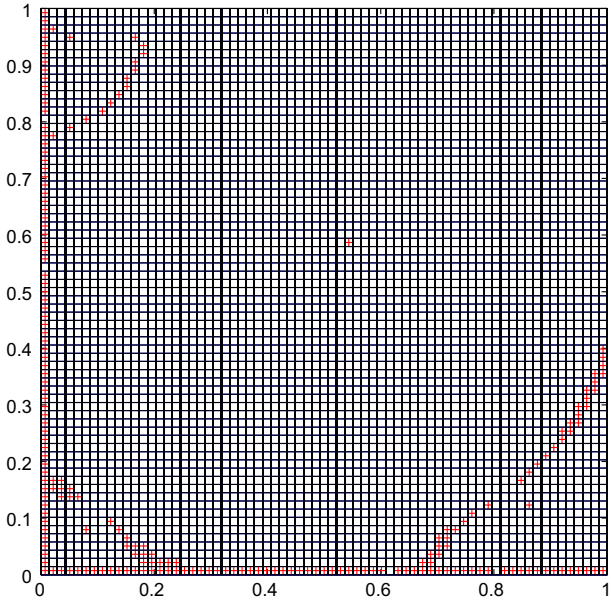

Fig. 12 Refined mesh for mesh size 69 × 69

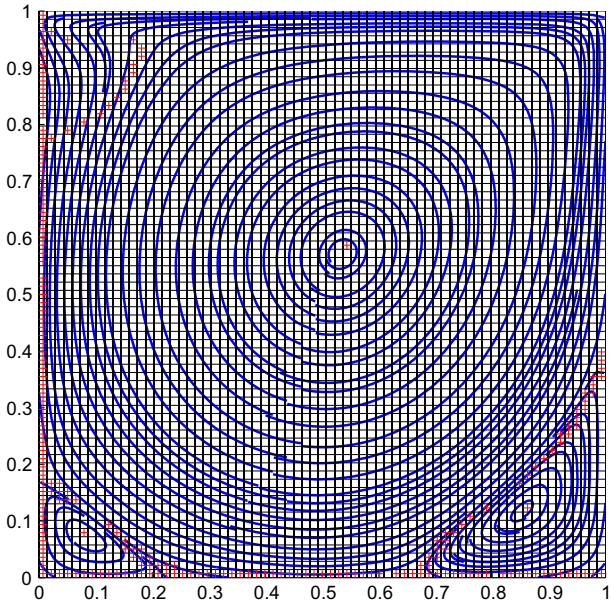

Fig. 13 Refined mesh with streamlines for mesh size 69 × 69

cells where $\mathbf{V}_l$ or $f\mathbf{V}_l$ do not satisfy Eq. (1). The estimates of centre of vortices for mesh size 69 × 69 are shown in Table 2.

Figures 14 and 15 show the horizontal and vertical velocity profiles, respectively, for mesh size 121 × 121 together with the corresponding benchmark results [8]. The

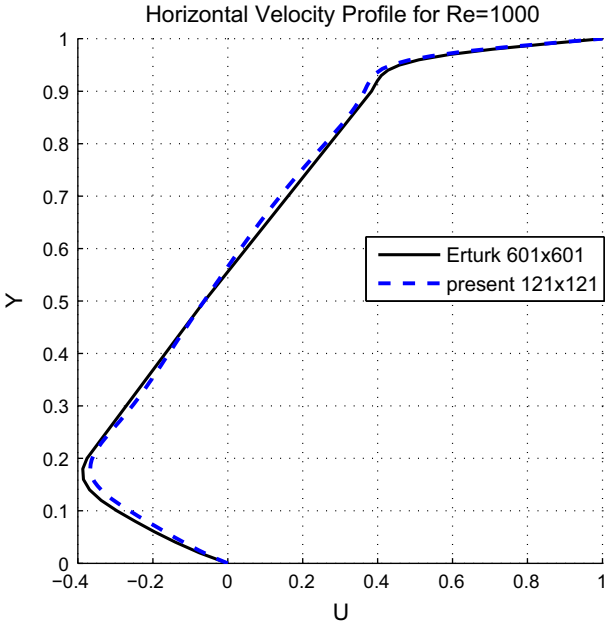

Fig. 14 Horizontal profile of velocity field for mesh size 121 × 121

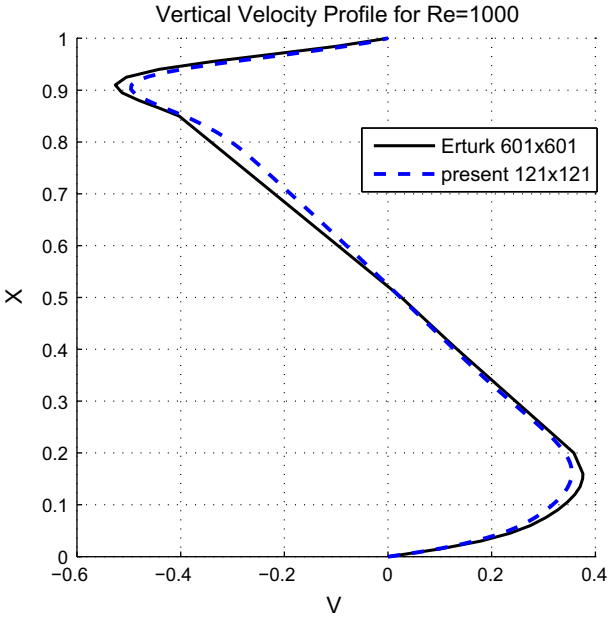

Fig. 15 Vertical profile of velocity field for mesh size 121 × 121

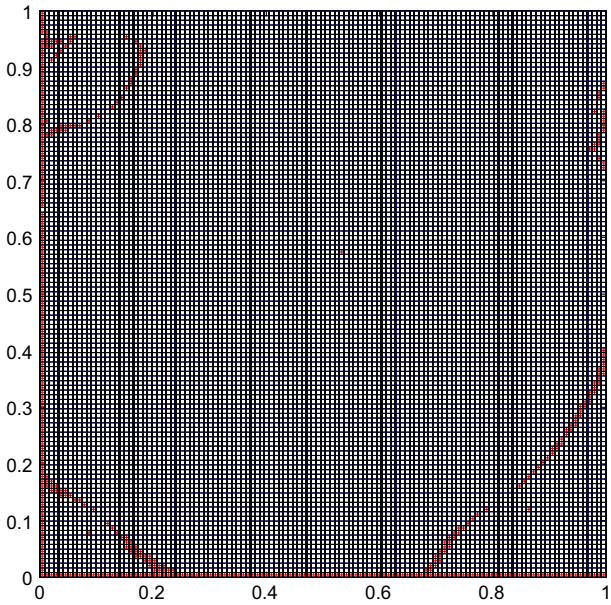

Fig. 16 Refined mesh for mesh size 121 × 121

accuracy of the computations has been greatly improved from these two figures but from Figs. 16 and 17, we find same three centres of vortices: primary, secondary bottom left (BL1) and right (BR1). The streamlines in Fig. 17 are also improved. Figures 12 and 16 do not show obvious differences between the two different mesh sizes. However, the accuracies of the estimates for the coordinates of centres of the three vortices for the two different mesh sizes are different as shown in Tables 2 and 3.

5.3 Re = 2,500

We use mesh size 91 × 91 as a coarser mesh. Figures 18 and 19 show horizontal and vertical velocity profiles respectively for the mesh size together with the corresponding benchmark results [8]. Figures 18 and 19 show similar errors for both components of the numerical velocity field. Based on the numerical velocity fields with the errors shown in Figs. 18 and 19, four centres of vortices: primary, secondary bottom right (BR1) and left (BL1), and top left (TL1) are identified and shown in Figs. 20 and 21. The streamlines shown in Fig. 21 are not closed so this fact indicates that the interpolated numerical velocity field $\mathbf{v}_l$ or $f\mathbf{v}_l$ do not satisfy Eq. (1) [13]. Figure 20 shows the refined mesh which presents information where $\mathbf{v}_l$ or $f\mathbf{v}_l$ do not satisfy Eq. (1). The estimates of centre of vortices for mesh size 91 × 91 are shown in Table 2.

Same finer mesh size 121 × 121 is used for presenting the change of accuracy of numerical solutions for lid-driven cavity flows with the change of Reynolds number in the following consideration. Figures 22 and 23 show the horizontal and vertical velocity profiles, respectively, for mesh size 121 × 121 together with the corresponding benchmark results [8]. The accuracies of the horizontal and vertical velocity profiles

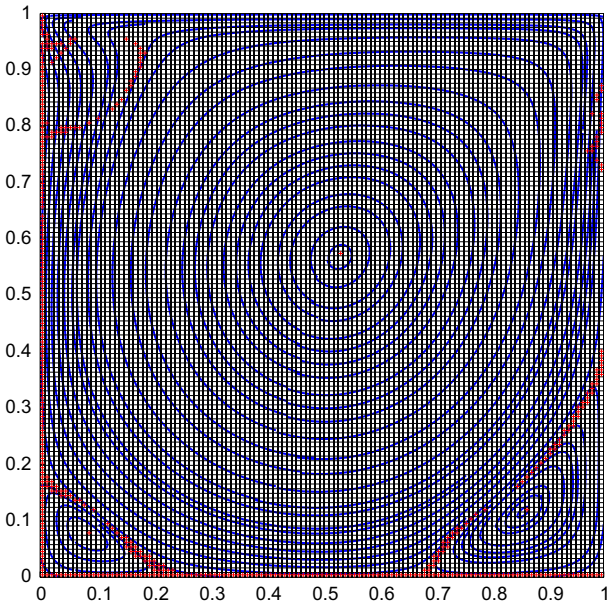

Fig. 17 Refined mesh with streamlines for mesh size 121 × 121

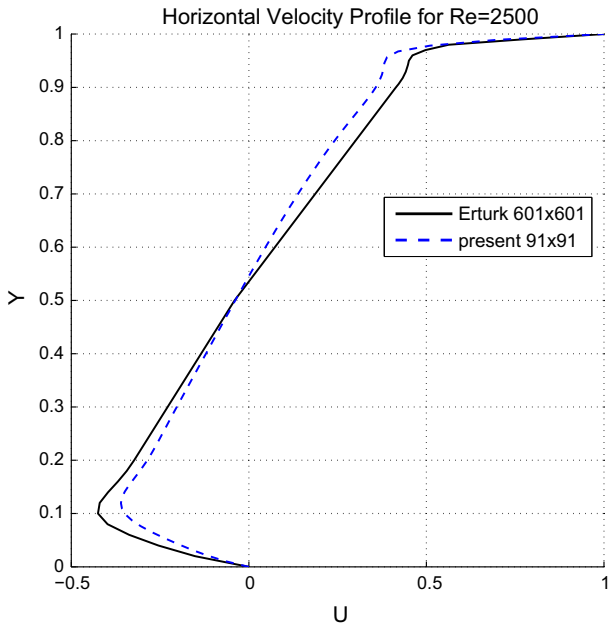

Fig. 18 Horizontal profile of velocity field for mesh size 91 × 91

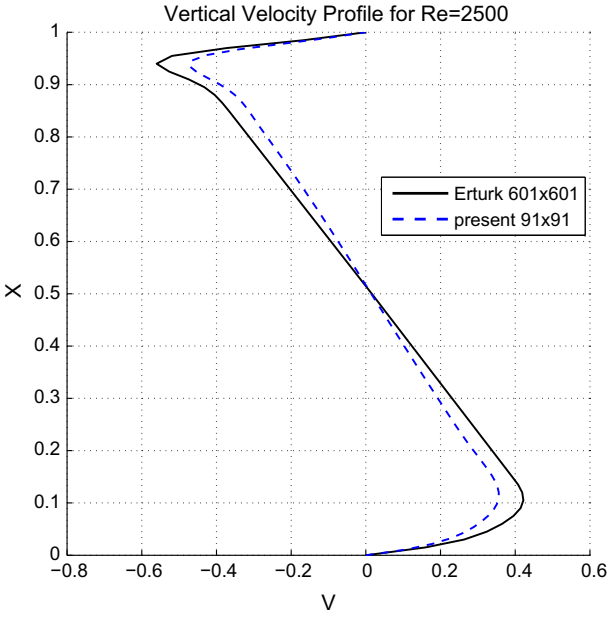

Fig. 19 Vertical profile of velocity field for mesh size 91 × 91

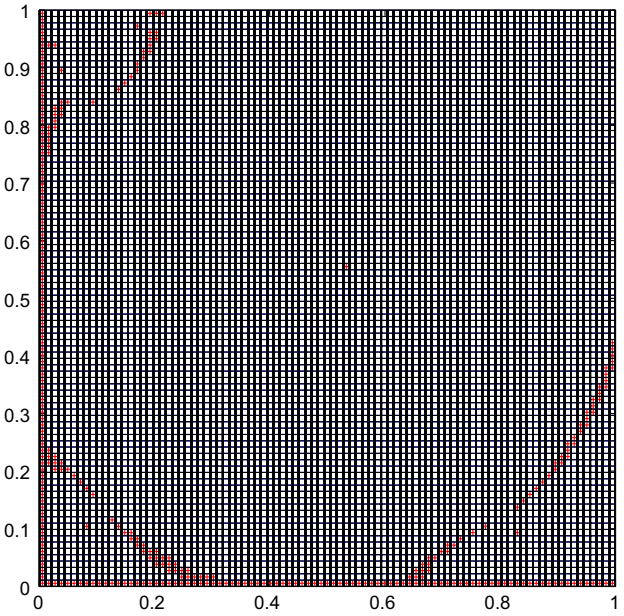

Fig. 20 Refined mesh for mesh size 91 × 91

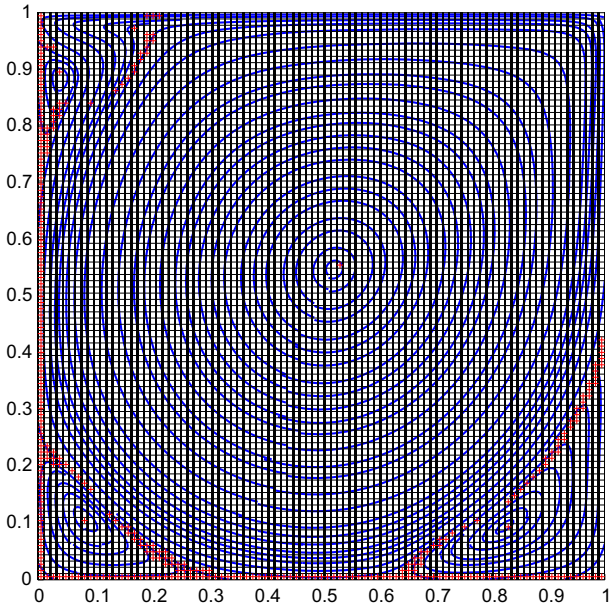

Fig. 21 Refined mesh with streamlines for mesh size 91 × 91

for Re = 2,500 using the same finer mesh sizes are worse than those for Re = 1,000 and Re = 100. The accuracy of the computations has been greatly improved by comparing Figs. 22 and 23 with Figs. 18 and 19. However, we find the same four centres of vortices: primary, secondary bottom left (BL1) and right (BR1), and top left (TL1) in Figs. 24 and 25. It is difficult to find the differences between the streamlines in Figs. 21 and 25. However, the accuracies of the estimates for the coordinates of centres of the four vortices for the two different mesh sizes are different as shown in Tables 2 and 3.

5.4 Vortex centre locations

The coordinates of centres of primary, secondary and tertiary vortices reported from [11] are evaluated using a mesh with 129 × 129 uniform cells for Re = 100 and [8] using a mesh with 601 × 601 uniform cells for Re = 1,000, and 2,500. Tables 2 and 3 present that the accuracy of the estimates for centres of vortices is improved when the mesh sizes increases. For Re = 100, there is a vortex in the bottom left corner from the information in the refined mesh but we cannot find which cell contains the centre of the vortex. The coordinates in italic in Tables 2 and 3 are the benchmark results [11] for $Re = 100$, and the other italic coordinates are the benchmark results [8].

6 Discussion

There are some cases in which no isolated refined cells are found in refined meshes. However, we can calculate the coordinates of the centres from the interpolated velocity

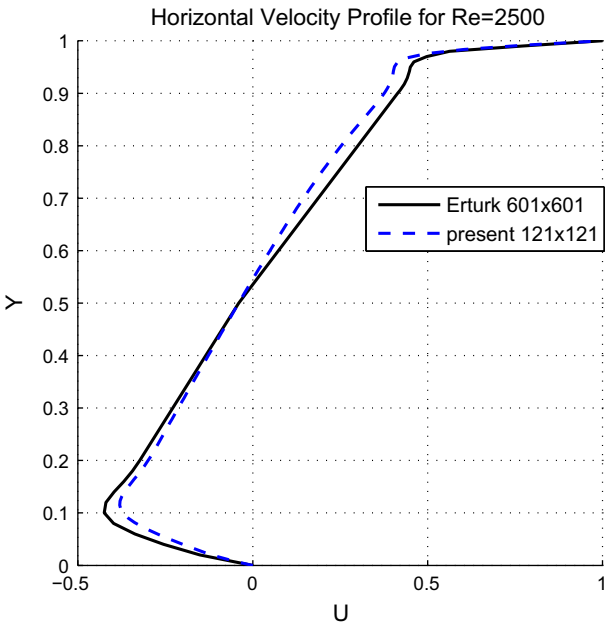

Fig. 22 Horizontal profile of velocity field for mesh size 121 × 121

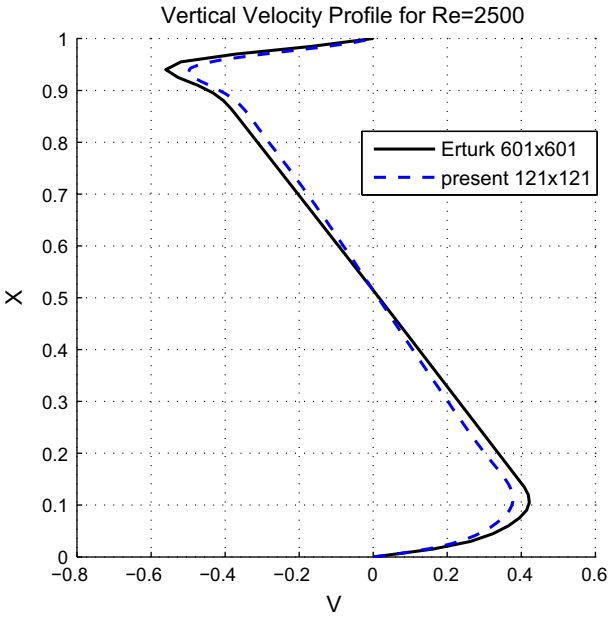

Fig. 23 Vertical profile of velocity field for mesh size 121 × 121

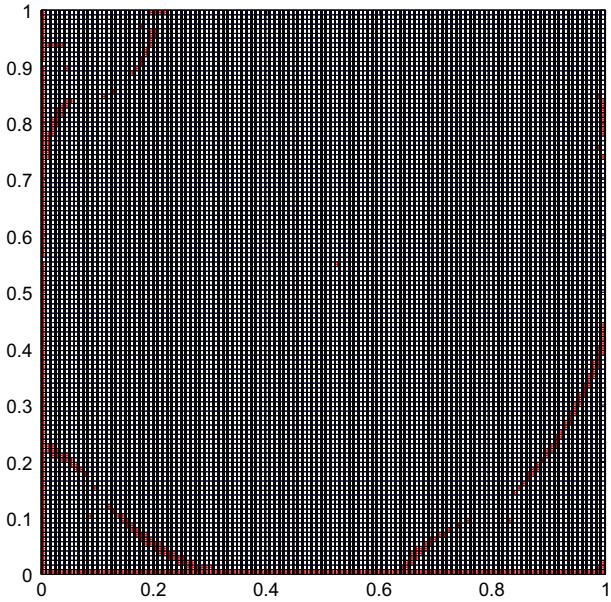

Fig. 24 Refined mesh for mesh size 121 × 121

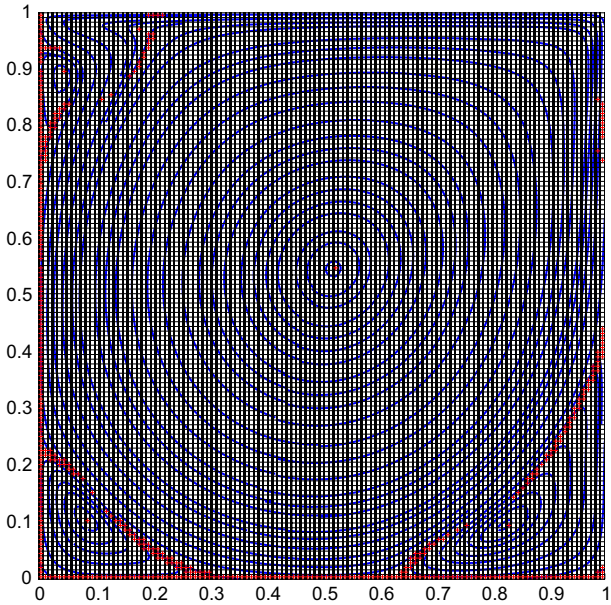

Fig. 25 Refined mesh with streamlines for mesh size 121 × 121

fields with both components of the velocity fields are zero. We can take these coordinates as the estimates for the centres of vortices. If there is no singular point in a region, all streamlines inside the region are closed [14]. The refined meshes also show

other refined elements and some of which form separation curves for multiple regions of separated flow [2]. However, we do not have the data for the separation curves. Therefore, we are not able to test the sensitivity of the mesh refinement method using separation curves.

The results shown in this paper demonstrate that the proposed mesh refinement method for 2D velocity fields is not very sensitive to the accuracy of numerical velocity fields. We conclude that the refinement method provides accurate and reliable refined meshes from which accurate numerical solutions can be achieved by adaptively solving mathematical models which contain continuity equation.

References

1. A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations. J. Comput. Phys. **142**, 1–46 (1998)
2. B.F. Armaly, F. Durst, J.C.F. Pereira, B. Schonung, Experimental and theoretical investigation of backward-facing step flow. J. Fluid Mech. **127**, 473–496 (1983)
3. J. Bell, M. Berger, J. Saltzman, M. Welcome, Three-dimensional adaptive mesh refinement for hyperbolic conservation laws. SIAM J. Sci. Comput. **15**, 127–138 (1994)
4. M.J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics. J. Comput. Phys. **82**, 64–84 (1989)
5. M.J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations. J. Comput. Phys. **53**, 484–512 (1984)
6. P.J. Capon, P.K. Jimack, An adaptive finite element method for the compressible Navier-Stokes equations, in *Numerical Methods for Fluid Dynamics*, vol. 5, eds. by M.J. Baines, K.W. Morton (OUP, 1995)
7. L. Demkowicz, J.T. Oden, W. Rachwicz, O. Hardy, An h-p Taylor–Galerkin finite element method for the compressible Euler equations. Comput. Methods Appl. Mech. Eng. **88**, 363–396 (1991)
8. E. Erturk, T.C. Corke, C. Gökcöl, Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers. Int. J. Numer. Meth. Fluids **48**, 747–774 (2005)
9. S. Faure, J. Laminie, R. Temam, Colocated finite volume schemes for fluid flows. Commun. Comput. Phys. **4**, 1–25 (2008)
10. J.H. Ferziger, M. Peric, *Computational Methods for Fluid Dynamics*, 3rd edn. (Springer, Berlin, 2002)
11. U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. J. Comput. Phys. **48**, 387–411 (1982)
12. R.D. Henderson, Adaptive spectral element methods for turbulence and transition, in *High-Order Methods for Computational Physics*, ed. by T.J. Barth, H. Deconinck (Springer, Berlin, 1999)
13. Z. Li, G. Mallinson, Mass conservative fluid flow visualisation for CFD velocity fields. KSME Int. J. **15**, 1794–1800 (2001)
14. Z. Li, A mass conservative streamline tracking method for two dimensional CFD velocity fields. J. Flow Vis. Image Process. **9**, 75–87 (2002)
15. Z. Li, G. Mallinson, Simplifications of an existing mass conservative streamline tracking method for 2D CFD velocity fields, in *GIS and Remote Sensing in Hydrology. Water Resources and Environment*, vol. 289, ed. by Y. Chen, K. Takara, I.D. Cluckies, F.H. DeSmedt (IAHS Press, Wallingford, 2004), pp. 269–275
16. Z. Li, An adaptive streamline tracking method for two-dimensional CFD velocity fields based on the law of mass conservation. J. Flow Vis. Image Process. **13**, 1–14 (2006)
17. Z. Li, An adaptive streamline tracking method for three-dimensional CFD velocity fields based on the law of mass conservation. J. Flow Vis. Image Process. **13**, 359–376 (2006)
18. Z. Li, An adaptive three-dimensional mesh refinement method based on the law of mass conservation. J. Flow Vis. Image Process. **14**, 375–395 (2007)
19. Z. Li, An adaptive two-dimensional mesh refinement method based on the law of mass conservation. J. Flow Vis. Image Process. **15**, 17–33 (2008)

20. R. Lohner, An adaptive finite element scheme for transient problems in CFD. Comput. Methods Appl. Mech. Eng. **61**, 323–338 (1987)
21. K. Miller, R. Miller, Moving finite elements. Part I. SIAM J. Numer. Anal. **18**, 1019–1032 (1981)
22. M.C. Mosher, A variable node finite element method. J. Comput. Phys. **57**, 157–187 (1985)
23. W. Speares, M. Berzins, A 3-D unstructured mesh adaptation algorithm for time-dependent shock dominated problems. Int. J. Numer. Methods Fluids **25**, 81–104 (1997)
24. Y. Ye et al., *Theory of Limit Cycles* (American Mathematical Society Press, Providence, 1986)
25. O.C. Zienkiewicz, D.W. Kelly, J.P. Gago, The hierarchical concept in finite element analysis. Comput. Struct. **16**, 53–65 (1983)