An adaptive inner iterative pressure-based algorithm for steady and unsteady incompressible flows

Jin-Ping Wang, Jian-Fei Zhang, Zhi-Guo Qu and Wen-Quan Tao
Key Laboratory of Thermo-Fluid Science and Engineering, Ministry of Education, School of Energy and Power Engineering, Xi'an Jiaotong University, Xi'an, China

Received 11 September 2018
Revised 14 December 2018
Accepted 25 December 2018

Abstract

Purpose – Pressure-based methods have been demonstrated to be powerful for solving many practical problems in engineering. In many pressure-based methods, inner iterative processes are proposed to get efficient solutions. However, the number of inner iterations is set empirically and kept fixed during the whole computation for different problems, which is overestimated in some computations but underestimated in other computations. This paper aims to develop an algorithm with adaptive inner iteration processes for steady and unsteady incompressible flows.

Design/methodology/approach – In this work, with the use of two different criteria in two inner iterative processes, a mechanism is proposed to control inner iteration processes to make the number of inner iterations vary during computing according to different problems. By doing so, adaptive inner iteration processes can be achieved.

Findings – The adaptive inner iterative algorithm is verified to be valid by solving classic steady and unsteady incompressible problems. Results show that the adaptive inner iteration algorithm works more efficient than the fixed inner iteration one.

Originality/value – The algorithm with adaptive inner iteration processes is first proposed in this paper. As the mechanism for controlling inner iteration processes is based on physical meaning and the feature of iterative calculations, it can be used in any methods where there exist inner iteration processes. It is not limited for incompressible flows. The performance of the adaptive inner iteration processes in compressible flows is conducted in a further study.

Keywords Efficiency, Adaptive inner iterative processes, Incompressible flows

Paper type Research paper

Nomenclature

a = Coefficient in discretized momentum equation;
A = Coefficient in discretized pressure equation;
b = Source term;
E = Time step;
P = Pressure;
S = Cross-sectional area;
u, v = Velocity components in the x- and y-directions; and
x, y = Cartesian coordinates.

This work is supported by National Natural Science Foundation of China (No. 51576155, No. 51206129) and the Foundation for Innovative Research Groups of National Natural Science Foundation of China (No. 51721004).

International Journal of Numerical Methods for Heat & Fluid Flow
Vol. 30 No. 4, 2020
pp. 2003-2024

0961-5539
DOI 10.1108/HFF-09-2018-0483

Greek symbols

ρ = Density; and

Δt = Time step.

Subscripts

e, w, ... = East, West face of a control volume;

E, W, ... = East, West neighbor of the main grid point;

nb = Neighbors of the P grid point; and

P = Grid point P.

1. Introduction

A pressure-based method was originally devised for incompressible flows and can be easily extended for compressible flows, showing a great potential to be developed into a unified method for flows at all speeds (Wang *et al.*, 2014; Issa, 1986). In the method, pressure equation or pressure correction equation is derived from the continuity and momentum equations such that the velocity field, corrected by the pressure (pressure correction), satisfies the continuity equation. As the governing equations are nonlinear, the solution process involves iterations wherein the governing equations are solved repeatedly until the solution converges.

SIMPLE algorithm is a famous pressure-based method proposed by Patankar and Spalding. There are two assumptions in the algorithm:

(1) The initial pressure field and the initial velocity field are assumed independently.

(2) The effects of the pressure corrections of the neighboring nodes are arbitrarily dropped to simplify the solution procedure (Patankar and Splading, 1972).

The first assumption results in a poor interconnection between pressure and velocity. The second assumption does not affect the accuracy, though it will slow down the convergence rate. After SIMPLE was proposed, many researchers worked on eliminating these two assumptions. In 1981, SIMPLER algorithm was put forward to remove the first assumption, improving the inherent interconnection between pressure and velocity (Patankar, 1981). In 2004, CLEAR algorithm, which serves to remove the second assumption of SIMPLE algorithm, was developed to improve the efficiency of computations (Tao *et al.*, 2004). IDEAL algorithm was brought forward for incompressible flows to improve the performance of CLEAR algorithm (Sun *et al.*, 2008a). It has been proved that IDEAL algorithm is more stable and robust than CLEAR algorithm.

The success of IDEAL method lies in the inner doubly iterative processes for pressure equation. The first inner iteration process is to get pressure field and the second inner iteration process is to get velocity field.

In Sun's study, it is said that the convergence and stability of computations can be controlled by the number of inner iterations (Sun *et al.*, 2008a). However, number of the steps in the two inner iteration processes is set empirically according to different computations in his study.

In this paper, by combining physical meaning and the feature of iterative calculation, a mechanism is proposed first to achieve adaptive inner iteration. To the best of the author's knowledge, this mechanism to control inner iterative processes is unique in pressure-based methods where inner iterative processes exist. Based on the adaptive inner iteration, an algorithm with adaptive inner iteration processes is developed to overcome the defect in fixed inner iteration algorithms. It is to be expected that the number of inner iterations in adaptive inner iterative algorithm varies with the computational progress according to different problems, making the algorithm efficient and robust.

Steady and unsteady incompressible flows, including a steady Navier–Stokes (NS) solution, lid-driven cavity flow, flow over a step, an exact unsteady flow, Couette flow with a transient flow field and oscillating lid-driven square cavity are calculated to check the performance of the adaptive inner iteration algorithm.

In Section 2, governing equations of incompressible flows and the discretized equations are presented. Implementation of adaptive inner iterations is shown in Section 3. The validity of the adaptive inner iteration algorithm is verified in Section 4. Conclusions are stated in Section 5.

2. Governing equations and discretized equations

The governing equations of two-dimensional incompressible flows is written on Cartesian coordinate:

Momentum equation:

$$\frac{\partial(\rho u_i)}{\partial t}+\frac{\partial(\rho u_j u_i)}{\partial x_j}=-\frac{\partial p}{\partial x_i}+\frac{\partial}{\partial x_j}\eta\left(\frac{\partial u_i}{\partial x_j}\right)+S_i \tag{1}$$

Continuity equation:

$$\frac{\partial(\rho u_i)}{\partial x_i}=0 \tag{2}$$

The governing equations above are discretized in finite-volume form on staggered grid system.

As stated in book (Tao, 2001), the discretized momentum equations can be written as follows:

$$a_e u_e=\sum a_{nb}u_{nb}+b+(p_P-p_E)S_e \tag{3}$$

$$a_n v_n=\sum a_{nb}v_{nb}+b+(p_P-p_N)S_n \tag{4}$$

Discretized continuity equation is:

$$\rho_e u_e S_e-\rho_w u_w S_w+\rho_n v_n S_n-\rho_s v_s S_s=0 \tag{5}$$

where S is the area which the fluid flows through.

As there is no independent equation for pressure, the pressure equation is derived from the discretized continuity and momentum equations.

Equation (3) is rewritten as:

$$u_e=\frac{\sum a_{nb}u_{nb}+b+S_e(p_P-p_E)}{a_e} \tag{6}$$

Equation (4) is rewritten as:

$$v_n=\frac{\sum a_{nb}v_{nb}+b+S_n(p_P-p_N)}{a_n} \tag{7}$$

Similarly, expressions for u_w and v_s are written as:

$$u_w=\frac{\sum a_{nb}u_{nb}+b+S_w(p_W-p_P)}{a_w} \tag{8}$$

$$v_s = \frac{\sum a_{nb} v_{nb} + b + S_s(p_S - p_P)}{a_s} \tag{9}$$

Substituting equations (6)-(9) in equation (5) results in a discretized pressure equation:

$$A_P P_p = A_E P_E + A_W P_W + A_N P_N + A_S P_S + b \tag{10}$$

where $A_E = \rho_e d_e S_e, A_W = \rho_w d_w S_w, A_N = \rho_n d_n S_n$ and $A_S = \rho_s d_s S_s$;

$$d_e = \frac{S_e}{a_e},\ d_w = \frac{S_w}{a_w},\ d_n = \frac{S_n}{a_n},\ d_s = \frac{S_s}{a_s},\ A_P = A_E + A_W + A_N + A_S;$$

$$b = \rho_w d_w \left(\sum a_{nb} u_{nb} + b\right) - \rho_e d_e \left(\sum a_{nb} u_{nb} + b\right) + \rho_s d_s \left(\sum a_{nb} v_{nb} + b\right) - \rho_n d_n \left(\sum a_{nb} v_{nb} + b\right)$$

3. Implementation of adaptive inner iterative processes

In IDEAL method, the first inner iterative process is to improve the coupling between velocity and pressure. Pressure equation is solved in the first iterative process to update pressure field, which is treated as one of the source terms in the momentum equation. After the first inner iterative process, momentum equation is solved to get intermediate velocity, which generally cannot satisfy the continuity equation. In the second inner iteration process, pressure equation is solved to improve velocity fields to make mass conservation well guaranteed. The numbers of inner iterations are controlled by N1 (first inner iteration) and N2 (second inner iteration). In IDEAL algorithm, the numbers are usually set as 4 (Sun *et al.*, 2008a). It has been proved that four times inner iterations can work well in most cases (Sun *et al.*, 2008b). However, in terms of fine-mesh or high-Re cases in lid-driven flows, the author set N1 and N2 as 5 or 10 (Sun *et al.*, 2008b). Obviously, the number of inner iterations should be adjusted according to computational problems. To do so, one mechanism to control inner iterative processes should be developed.

With the combination of physical meaning and iterative calculation, one mechanism is proposed in this work to get the number of inner iterations adjusted according to computational problems. Details are presented below.

The first inner iterative process: In incompressible flows, pressure and velocity are coupled with each other. Inner iterative calculation can improve the coupling between pressure and velocity to achieve accurate solutions. Pressure can be treated as the source term of the momentum equations (3) and (4). To get a reasonable pressure field before solving momentum equations, inner iterative calculations for pressure equation should be set. The pressure equation solutions are monitored by using residual norm of pressure equation.

The residual norm of pressure equation is written (take one-dimensional form as an example) as:

$$R_p^{(n)} = \left\{ \sum \left[\left(A_E P_E + A_W P_W + b - A_P P_P \right)^{(n)} \right]^2 \right\}^{\frac{1}{2}} \tag{11}$$

where superscript n stands for the nth calculation in the inner iteration.

The criterion $R_P{}^{(n)}/R_p{}^{(1)} < 1$ is used to end the first inner iteration process. By doing so, at least two inner iterative steps will be conducted during computing, which keeps the feature of iteration. At the same time, with the surveillance of the pressure equation error, the whole computational program is running stably.

The second inner iterative process: The velocity field obtained from solving momentum equations could not satisfy the continuity equation in general. Pressure equation derived from continuity equation is solved to revise velocity equation to obtain mass conservation. To maximum satisfy mass conservation, set an inner iterative process for solving pressure equation. Global mass residual is adopted to check how much mass conservation is obtained.

The global mass residual is written as:

$$R_m{}^{(n)} = \sum (\rho_e u_e S_e - \rho_w u_w S_w)^{(n)} \tag{12}$$

where superscript n stands for the nth calculation in the second inner iteration.

The second inner iterative process ends when the criterion $R_m{}^{(n)}/R_m{}^{(1)} < 1$ is satisfied. With this constraint, minimum two inner iterations are conducted and the error of continuity equation is getting controlled, making the continuity equation well guaranteed.

Criteria proposed above can be used in any method where inner iterations exist. Like in PISO, more than one corrector steps are introduced to obtain a velocity field that satisfies the zero-divergence condition (Issa, 1985), and a minimum of two corrector steps was recommended by the author. In Karki and Patankar (1989), the author suggested that the pressure correction equation should be solved two or three times to satisfy continuity equation. However, if equation (12) is used in these methods, the number for the inner iteration will be decided automatically and adjusted according to computational problems without manual setup.

4. Results and discussion

In this part, the validity of the adaptive inner iteration algorithm is verified. To know the performance of the adaptive inner iteration, solutions are compared with those of fixed inner iteration algorithm (IDEAL algorithm).

4.1 Steady incompressible flows

Three classical computational models are calculated in this part. They are a steady NV solution, lid-driven cavity flow and flow over a backward-facing step.

A steady NS solution: Consider a steady NS solution proposed by Kovasznay (Kovasznay, 1948) The governing equations (1) and (2) are solved with density $\rho = 1$ and viscosity $\eta = 1/40$ in a domain defined by $-1/2 \ll \mathrm{x} \ll 1$, $-1/2 \ll \mathrm{y} \ll 3/2$. Boundary conditions are set as follows: on the top and bottom of the domain, periodic boundary condition is used. On the left side and right side, Dirichlet boundary conditions are set to be consistent with the exact solution:

$$u(\mathrm{x},\mathrm{y}) = 1 - e^{\lambda x}\cos 2\pi y \tag{13}$$

$$v(\mathrm{x},\mathrm{y}) = \lambda e^{\lambda x}\sin 2\pi y/2\pi \tag{14}$$

$$P(\mathrm{x},\mathrm{y}) = (1 - e^{2\lambda x})/2 \tag{15}$$

where $\lambda = 1/2\eta - (1/4\eta^2 + 4\pi^2)^{1/2}$

The streamline pattern in the adaptive inner iteration solution is depicted in Figure 1, which is consistent with that given by Maday *et al.* (1990).

To know how the number of inner iterations changes with the calculation proceeding in adaptive inner iterative solutions, Figure 2 shows the variation of inner iterative times on grid system 102 × 102. Here, "inner 1" in the picture means the first inner iteration; similarly, "inner 2" is the second inner iteration. We can see that the number of the first inner iteration varies from two to six as calculation proceeds. The number of second inner iteration is changing from two to four during computing.

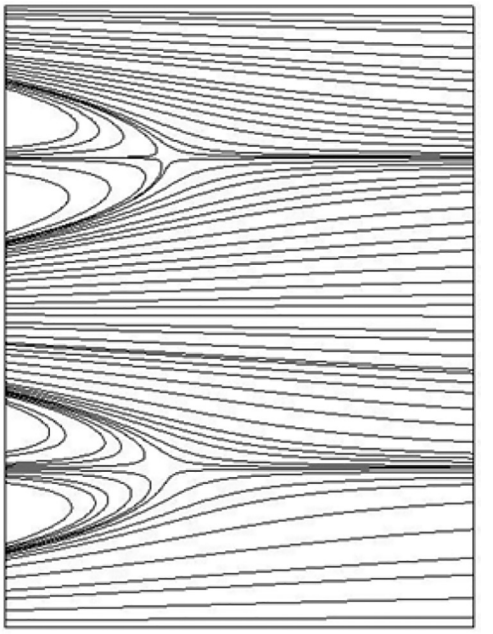


Figure 1.
Streamline pattern in steady Navier–Stokes solution

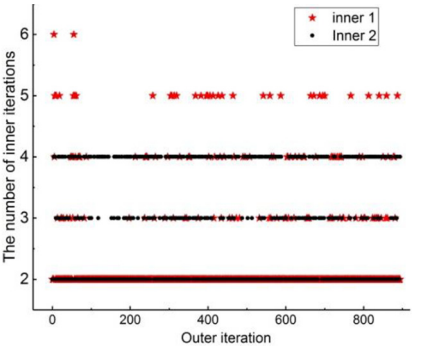


Figure 2.
Variation of inner iteration number with computational progress on 102 × 102 grid system

Solutions of fixed inner iterations (four for each inner iteration) and adaptive inner iterations are compared with the exact solution on different grid systems. The relative errors and computing time are shown in Table I.

From Table I, it is seen that the fixed inner iterative solutions and the adaptive inner iterative solutions have the same level accuracy. With the increase in the grid number, the accuracy increases. In addition, solutions of adaptive inner iterations cost less time than those of fixed ones. Corresponding to grid systems 52×52, 102×102, 152×152 and 202×202, computing time is saved by 18.9, 52.4, 52.4 and 48.5 per cent, respectively, by using adaptive inner iterations.

Lid-driven cavity flow: The laminar incompressible flow in a cavity whose top wall moves with a constant velocity has served over and over again as a model problem for evaluating numerical technique (Ghia *et al.*, 1982). Computational domain is defined by $0 \ll x \ll 1, 0 \ll y \ll 1$. Boundary conditions are set as follows: zero-slip condition at the left, right and bottom walls and $U = u = 1$ at top wall.

Results of the lid-driven cavity problem were confirmed by many other studies and the solution obtained at Re = 1,000 is quite close from one author to another (Bruneau and Saad, 2006). Among the results, Ghia *et al.* (Issa, 1985) adopted the vorticity-stream function method incorporating multi-grid techniques to obtain solutions for high Reynolds numbers and provided solutions in detail for Re = 100-10,000. Their results have long been regarded as the benchmark solutions in computational fluid dynamics-numerical heat transfer communities (Sun *et al.*, 2008b; Musavi and Ashrafizaadeh, 2016; Romanò and Kuhlmann, 2016).

The predicted streamline contours for different Reynolds numbers (Re = 1,000 and Re = 5,000) are shown in Figure 3. Figure 3(a) exhibits a large primary vortex with two secondary vortices in the two bottom corners in the solutions of Re = 1,000. In addition to the two secondary vortices in the bottom, a third vortex is present in the upper left corner in the solutions of Re = 5,000, seen in Figure 3(b). This is consistent with the results in literature (Ghia *et al.*, 1982) [Figure 3(c) and (d)] and (Bruneau and Saad, 2006) [Figure 3(e) and (f)].

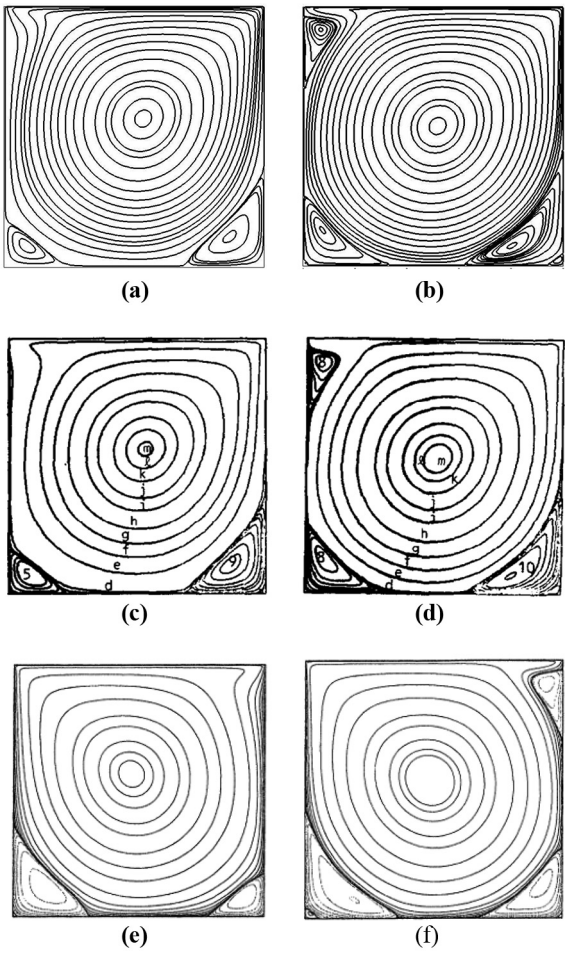
Figure 4(a) and (b) presents the u-velocity profiles along vertical lines and v-velocity profiles along horizontal lines passing through the geometric center of the cavity for Re = 1,000 and Re = 5,000. From the four pictures, we can see that adaptive inner iteration solutions agree with Ghia's results (Ghia *et al.*, 1982) very well, illustrating the accuracy of the adaptive inner iteration.

Figure 5 shows the variation of the inner iterative times at Re = 1,000. We can see that the number of the first inner iteration varies from two to four and stays as two most of the time. The number of the second inner iteration is uniformly distributed as two, three and four.

Figure 6 shows the comparison of computing time between the adaptive inner iteration solution and fixed inner iteration solution [four times inner iteration when Re < 1,000 and five times inner iteration for Re ≥ 1,000 according to Sun *et al.* (2008b)] for Re = 100-5,000.

Grid size	Fixed inner iterative solution		Adaptive inner iterative solution	
	Error (%)	Time (s)	Error (%)	Time (s)
52×52	0.19	3.71	0.19	3.12
102×102	0.05	67.42	0.05	44.23
152×152	0.03	316.87	0.03	207.95
202×202	0.02	906.93	0.02	610.73

Table I. Relative errors of u-velocity and computing time in steady NS solution

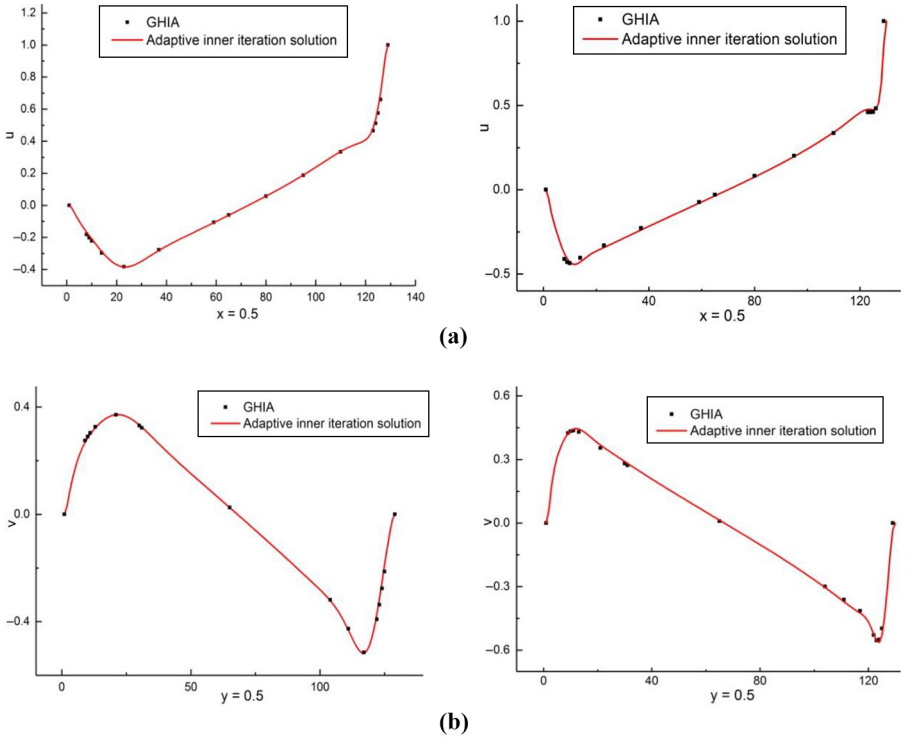


Notes: (a) Streamline patterns in present solutions; (d) streamline patterns in present solutions; (c) streamline patterns in literature (Ghia *et al.*, 1982) [U = (1,0)]; (d) streamline patterns in literature (Ghia *et al.*, 1982) [U = (1,0)]; (e) streamline patterns in literature (Bruneau and Saad, 2006) [U = (−1,0)]; (f) streamline patterns in literature (Bruneau and Saad, 2006) [U = (−1,0)]

Figure 3.
Streamline patterns (left: Re = 1,000; right: Re = 5,000)

The efficiency of the adaptive inner iteration solutions is higher than that of fixed ones in the whole region of Reynolds numbers. It is noted that at Re = 100, the computing time is saved by 1.36 times using adaptive inner iteration.

Laminar fluid flow over a backward-facing step: Flow over a backward-facing step is regarded as having the simplest geometry while retaining rich flow physics (Erturk, 2008; Benarda *et al.*, 2016).



Notes: (a) u-velocity along vertical line through geometric center (left: Re = 1,000; right: Re = 5,000); (b) v-velocity along horizontal lines through geometric center (left: Re = 1,000; right: Re = 5,000)

Figure 4.
Comparison of velocity profiles with Ghia's solutions

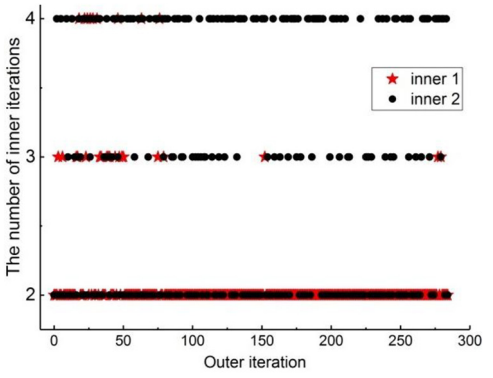


Figure 5.
Variation of inner iteration number with computational progress (Re = 1,000)

Cruchaga (1998) solved the backward-facing step flow both with and without using an inlet channel and obtained slightly different solutions. Based on Cruchaga's study, a computational model without inlet channel upstream is established. The case of expansion ratio H/h of 1.9423 is solved in the present study, which has

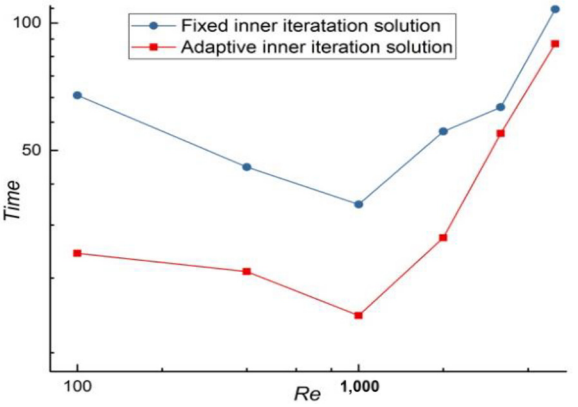


Figure 6.
Comparison of computing time for Re = 100∼5,000

been studied by many researchers (Erturk, 2008; Armaly *et al.*, 1983; Biswas *et al.*, 2004).

At the inlet boundary, a fully developed Plane Poiseuille flow is imposed between parallel plates such that the inlet velocity profile is parabolic.

The predicted length of the primary recirculation length x1/h is 2.83, as shown in Figure 7. The relative error is 4.35 per cent compared with that in Erturk (2008) and is 4.70 per cent compared with that in Biswas *et al.* (2004).

It was found that at Reynolds numbers smaller than 400, the flow maintained its two-dimensionality (Armaly *et al.*, 1983). In the second case of flow over a backward-facing step, Re = 389 is calculated to have a comparison with the experimental results.

Figures 8(a) shows velocity profiles for Re = 389 at different x/s locations in adaptive inner iteration solutions. The corresponding experimental results in Armaly *et al.* (1983) are presented in Figure 8(b). Good agreement is seen between the adaptive inner iteration solution and the experimental data.

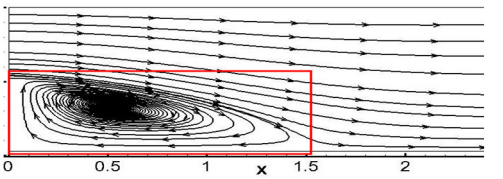
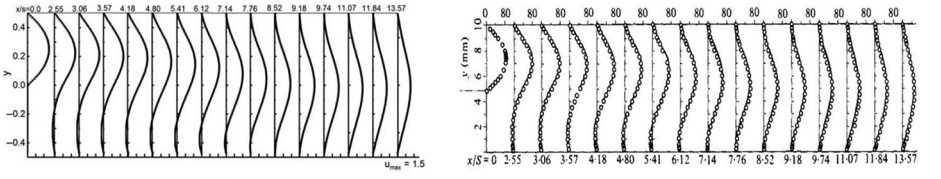


Figure 7.
Stream function contours at Re = 100



(a) (b)

Figure 8.
Velocity profiles for Re = 389 at different x/s locations

Notes: (a) Adaptive inner iteration solution; (b) solution in Armaly *et al.* (1983)

Figure 9 shows the length of the main recirculation region with respect to the Reynolds numbers (Re = 50-389), which is consistent with those in the experimental work (Armaly *et al.*, 1983) and numerical work (Biswas *et al.*, 2004).

Variation of the number of inner iterations with the computational progress (Re = 100) is shown in Figure 10. It is seen that the numbers of first inner iteration and second inner iteration change from two to four and stay at two most of the time during computing.

Computing time of the adaptive inner iterations and fixed inner iterations solutions (four for each inner iteration, according to Sun *et al.* (2008b) is presented in Figure 11. It is seen that the efficiency is highly improved by using the adaptive inner iteration.

4.2 Unsteady incompressible flows

After verification in steady flows, the adaptive inner iteration processes are extended for unsteady flows. An exact unsteady flow, Couette flow with a transient flow field, and oscillating lid-driven cavity are calculated in this part.

4.2.1 An unsteady flow with exact solutions. Consider a simple model proposed by Kim and Moin (1985). With $\rho = 1$ and $\eta = 1/2\pi^2$, governing equations (1) and (2) are solved in domain $-1 \leq x \leq 1, -1 \leq y \leq 1$. The initial conditions are given:

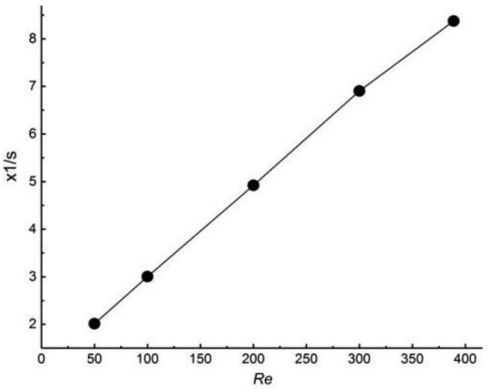


Figure 9.
Length of the primary recirculation region behind the backward-facing step

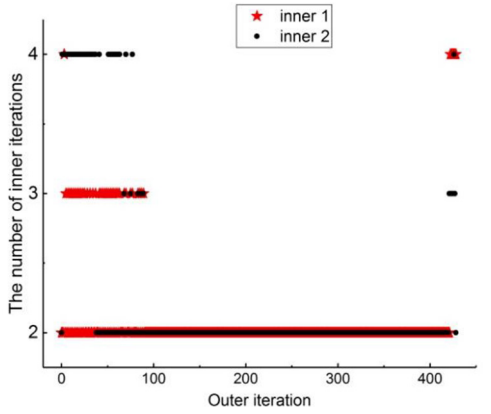


Figure 10.
Variation of inner iteration number with computational progress (Re = 100)

$$u_0(x, y) = -\cos \pi x \sin \pi y \tag{16}$$

$$v_0(x, y) = \sin \pi x \cos \pi y \tag{17}$$

for which the exact solution is obtained by:

$$u(x, y) = (-\cos \pi x \sin \pi y)e^{-t} \tag{18}$$

$$v(x, y) = (\sin \pi x \cos \pi y)e^{-t} \tag{19}$$

$$P(x, y) = -(\cos 2\pi x + \cos 2\pi y)e^{-2t}/4 \tag{20}$$

The following solutions are results at time $t = 1$ with time step $\Delta t = 0.01$. Figure 12 shows u-velocity fields from exact solution [Figure 12(a)], adaptive inner iteration solution [Figure 12(b)] and IDEAL algorithm solution [Figure 12(c)]. We can see the numerical results are consistent with the exact solution qualitatively.

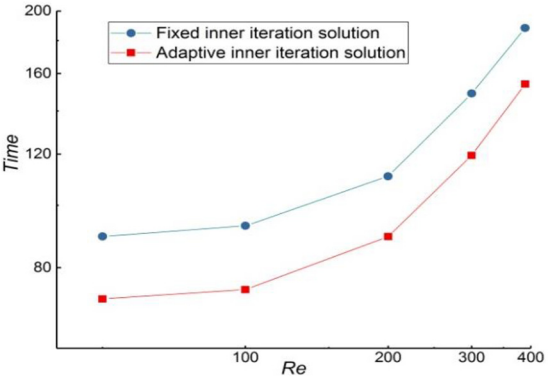


Figure 11.
Comparison of computing time with respect to Reynolds numbers

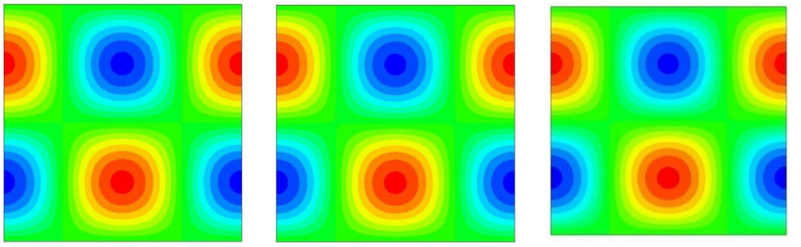


Figure 12.
u-velocity fields in different solutions

Notes: (a) Exact solution; (b) adaptive inner iteration solution; (c) fixed inner iteration solution

Adaptive inner iteration solutions and fixed inner iteration solutions are compared with exact solutions quantitatively. Table II shows the relative error of u-velocity and computing time on different grid systems. It is observed that these two solutions have the same level accuracy, and the accuracy increases with the increase in grid number. However, computing time in adaptive inner iteration solutions is less than that in fixed inner iteration solutions, which illustrates the higher efficiency of adaptive inner iteration. Corresponding to gird systems 102 × 102, 152 × 152, 202 × 202 and 252 × 252, the efficiency is improved by 9.9, 12.6, 25.7 and 12.7 per cent, respectively, by using adaptive inner iteration algorithm.

Inner iteration numbers in adaptive inner iteration solutions at three different moments on 252 × 252 grid system are shown in Figure 13. At the beginning of the calculation, number of the first inner iteration varies from 2 to 30. In the calculation at $t = 0.01$, number of the first inner iteration changes from 2 to 25. In the calculation at $t = 0.02$, number of the first inner iteration changes from 2 to 20. At different time steps, the number of the second inner iteration stays as two most of the time.

4.2.2 Couette flow with a transient flow field. Consider a viscous incompressible Couette flow (Figure 14). The upper plate is moving to right at a constant velocity. The lower plate is stationary. The two plates are parallel and infinitely long. Flow is steady under this condition.

Imagine, for a moment, the space between the upper and lower plates is filled with a flow field $u = 2y$ (Figure 15). Such a flow exists at some instant during the start-up process just after the upper plate is set into motion. This would be a transient flow field. After enough time elapses, the flow will approach a steady state, and this steady state will be the Couette flow (Figure 14).In addition, u-velocity fields at different time steps are tracked using adaptive inner iteration processes. The results are shown in Figure 16. The flow field has been a steady state at $t = 60$ s. Also, u-velocity profiles of different time at right boundary are presented in Figure 17, from which the state of flow field can be clearly observed.

In this solution, computing time of adaptive inner iteration is 78.11 s, and time used by fixed inner iteration (IDEAL algorithm) is 97.97 s. Efficiency is enhanced by 25.4 per cent using adaptive inner iteration algorithm.

Variation of the inner iteration numbers in the adaptive inner iteration solutions at three different moments is shown in Figure 18. From the three pictures, we can see at different time, numbers of the first inner iterations vary from two to four and numbers of the second inner iterations stay at two most of time, which explains that the number of inner iterations (four for each inner iteration) in IDEAL algorithm is overestimated in some computations.

4.2.3 Oscillating lid-driven square cavity. In the previous steady computation, lid-driven cavity problem in which the top lid moves at a constant velocity is solved. To further illustrate the accuracy of adaptive inner iteration algorithm in this steady flow, predicted maximum and minimum values of velocity are compared with those of Ghia *et al.* (1982),

Table II. Relative errors of u-velocity and computing time in unsteady exact solutions

Grid size	Fixed inner iterative solution		Adaptive inner iterative solution	
	Error (%)	Time (s)	Error (%)	Time (s)
102 × 102	1.23	63.52	1.23	57.81
152 × 152	0.90	223.35	0.90	198.39
202 × 202	0.75	674.44	0.75	536.69
252 × 252	0.67	1,330.30	0.67	1,180.80

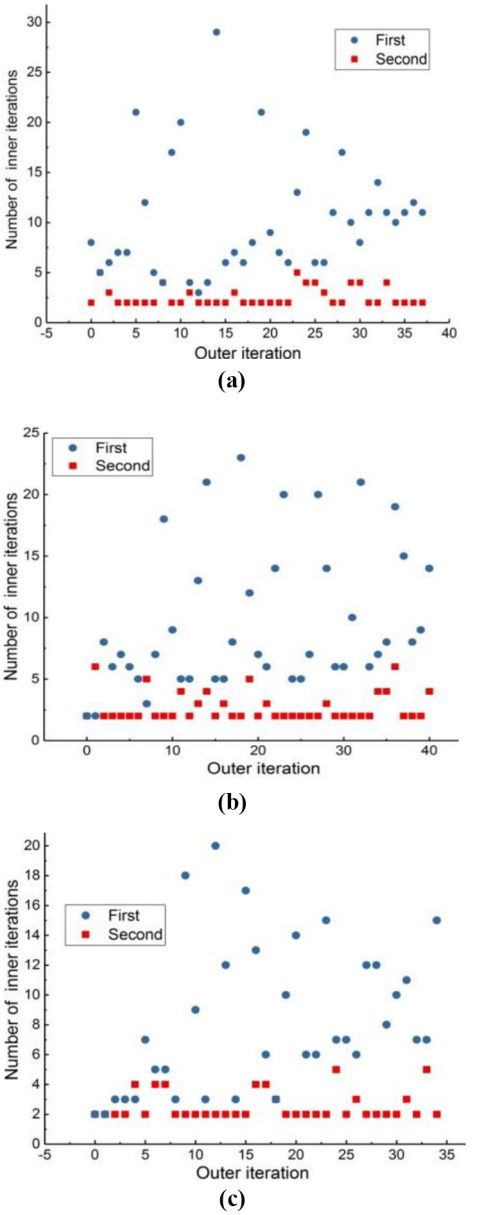


Figure 13.
Variation of inner iteration number with computational progress at different time steps on 252 × 252 grid system

Notes: (a) $t = 0$; (b) $t = 0.01$; (c) $t = 0.02$

seen in Table III. It is noted that the computed maximum relative error is 0.86 per cent, indicating the accuracy of the adaptive inner iteration.

After adaptive inner iteration is verified to be accurate in the cavity flow driven by constant motion of the top lid, it is extended for oscillating lid-driven flow. Figure 19 shows the computational model, in which $\omega = 2\pi/6$. Use small time step $\Delta t = 0.01$ to

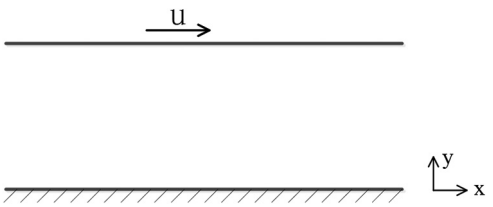


Figure 14.
Computational model of Couette flow

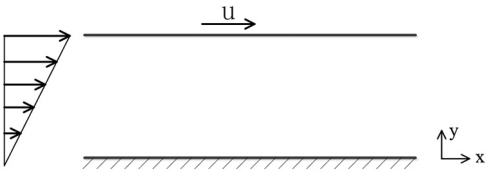


Figure 15.
A flow field exerted in Couette flow

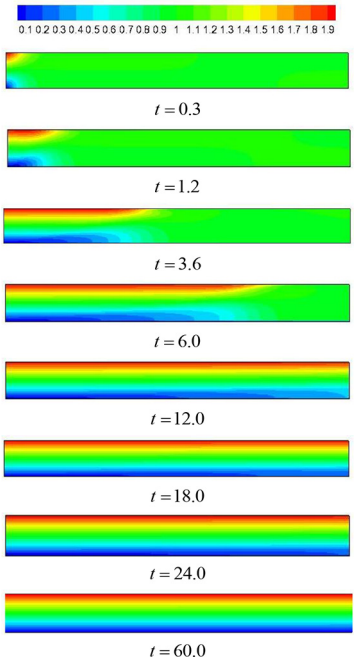


Figure 16.
u-velocity fields at different time steps

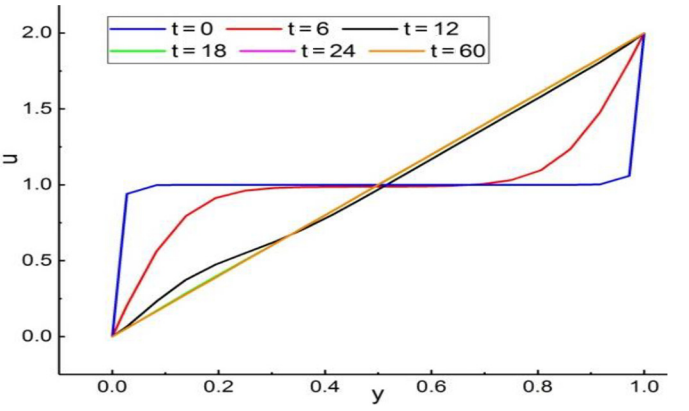


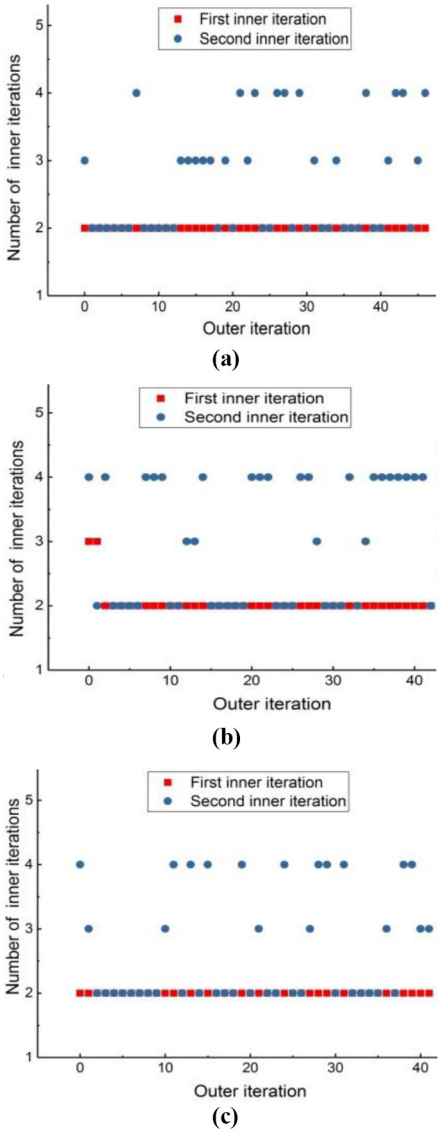
Figure 17.
u-velocity profiles at right boundary

ensure the accuracy in unsteady solutions. With the oscillating lid, the flow field becomes periodic with a frequency identical to that of the lid. In this calculation, one cycle is divided into 12 intervals, whereby the first half of the cycle contains the first six intervals. To reduce computing time, program runs end at $t = 3$ when the first half of the cycle is completed.

Velocity profiles at Re = 100 are shown in Figure 20. From the pictures, we can see the magnitude and direction of the velocity undergo a number of changes with time. Take u-velocity profile at $t = 1.8$ (seen in Figure 21) and v-velocity at $t = 1.2$ as an explanation (seen in Figure 22). At $t = 1.8$, The u-velocity changes from local maximum negative value −0.31 to 0.0, increases to local maximum positive value 0.14 and then decreases again to local maximum negative value −0.04. Thereafter, the fluid velocity gradually decreases to 0 as it approaches the bottom wall. At $t = 1.2$, v-velocity increases from 0 at the left wall to reach a local maximum positive value 0.05, then decreases to 0. Thereafter, the velocity continuously decreases to a local maximum negative value −0.07 and then increases to 0 at the right wall. These phenomena match with those in paper (Mendu and Das, 2013).

Variation of inner iterative numbers in the adaptive inner iteration solutions at three different moments for Re = 100 are shown in Figure 23. At the beginning of the calculation, number of the first inner iteration stays as two and the number of second inner iteration varies from two to four. In the calculation at $t = 0.01$, number of the first inner iteration changes from 2 to 12 and stays as 2 most of the time. The number of second inner iteration varies from two to four. In the calculation at $t = 0.02$, number of the first inner iteration changes from 2 to 12 and holds as 2 most of the time. Number of the second inner iteration varies from two to six.

In this case, computing time cost by adaptive inner iteration is 138.51 s, and computing time cost by fixed inner iteration is 163.79 s. The efficiency is improved by 18.3 per cent by using adaptive inner iteration algorithm.Also, u-velocity profiles for Re = 400 and Re = 1,000 in adaptive inner iteration solutions are presented in Figure 24. Compared with the u-velocity profile at Re = 100, we can see that as Re increases, the movement of the oscillating lid can only set in motion a limited fluid mass just below it. The rest of the fluid body stays as zero for the entire period of the cycle.



Notes: (a) $t = 0$; (b) $t = 0.01$; (c) $t = 0.02$

Figure 18.
Variation of inner iteration number with computational progress at different time steps

Variation of the inner iteration numbers for different Reynolds numbers at $t = 0.02$ is shown in Figure. 25. From Figure 25(a), we can see for Re = 400, number of the first inner iteration changes from 2 to 12 and keeps as 2 most of the time, and number of the second inner iteration varies from 2 to 7. Computing time in the adaptive inner iteration

solution is 1,157.33 s and in the fixed inner iteration solution is 1181.82 s. Computing efficiency is improved by using adaptive inner iteration. At Re = 1,000, number of the first inner iteration changes from two to four and keeps as two most of the time. Number of the second inner iteration varies from two to eight, as seen in Figure 25(b). Computing time cost by adaptive inner iteration is 1,330.97 s and cost by fixed inner iteration algorithm is 1,645.95 s. Thus, 23.67 per cent computing time is saved by using adaptive inner iteration algorithm.

5. Conclusions

In this paper, a mechanism is developed to achieve adaptive inner iteration algorithm. With the use of two different criteria in two inner iterative processes, the adaptive inner iteration algorithm is proved to be valid. The efficiency of all the solutions is improved using adaptive inner iteration without deteriorating the accuracy. Compared with fixed inner iteration solutions, in steady lid-driven cavity flow, the efficiency is improved by 1.36 times at Re = 100, and in unsteady Couette flow, the efficiency is enhanced by 25.4 per cent using adaptive inner iteration algorithm.

In addition, the mechanism proposed in this paper for controlling inner iterative processes can be used in any method where there exists inner iteration processes to decide the number of inner iterations. It is not limited for incompressible flows. The performance of the adaptive inner iteration processes in compressible flows is conducted in our further study.

Table III. Comparison of minimum and maximum values of velocity between the present solution and Ghia's results (Ghia *et al.*, 1982) in oscillating lid-driven flows

	Present	Ghia *et al.* (1982	Error
Re = 100			
$(u/U)_{min}$	−0.2097	−0.2109	0.57
$(v/U)_{min}$	−0.24384	−0.24533	0.61
$(v/U)_{max}$	0.17377	0.17527	0.86
Re = 400			
$(u/U)_{min}$	−0.32634	−0.32726	0.28
$(v/U)_{min}$	−0.45032	−0.44993	0.09
$(v/U)_{max}$	0.30062	0.30203	0.47
Re = 1,000			
$(u/U)_{min}$	−0.38303	−0.38289	0.04
$(v/U)_{min}$	−0.51852	−0.5155	0.59
$(v/U)_{max}$	0.37111	0.37095	0.04

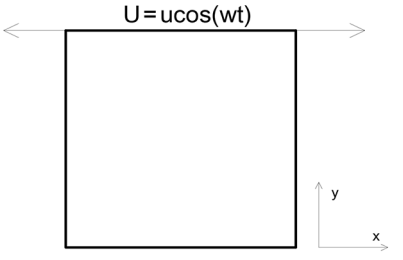


Figure 19. Computational model of oscillating lid-driven flow

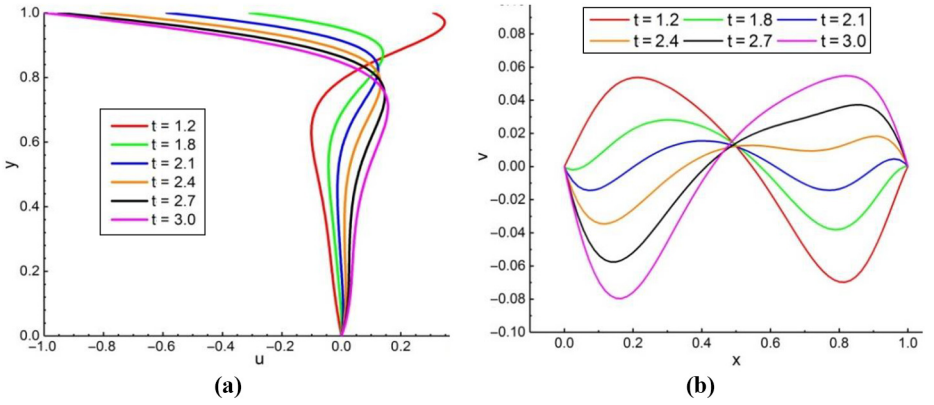


Figure 20. Velocity profiles at Re = 100 in oscillating lid-driven flow

Notes: (a) u-velocity profile along the vertical centerline; (b) v-velocity profile along the horizontal centerline

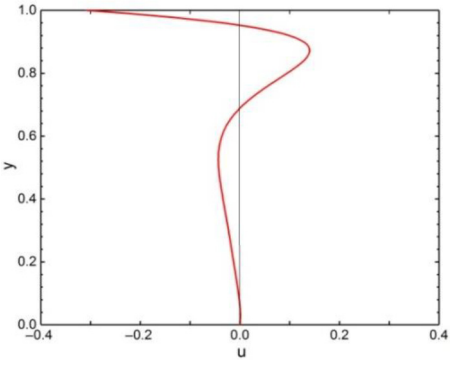


Figure 21. u-velocity profile along the vertical centerline of the cavity at $t = 1.8$ for Re = 100

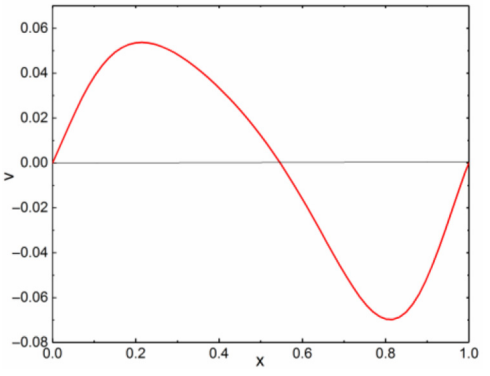


Figure 22. v-velocity profile along the horizontal centerline of the cavity at $t = 1.2$ for Re = 100

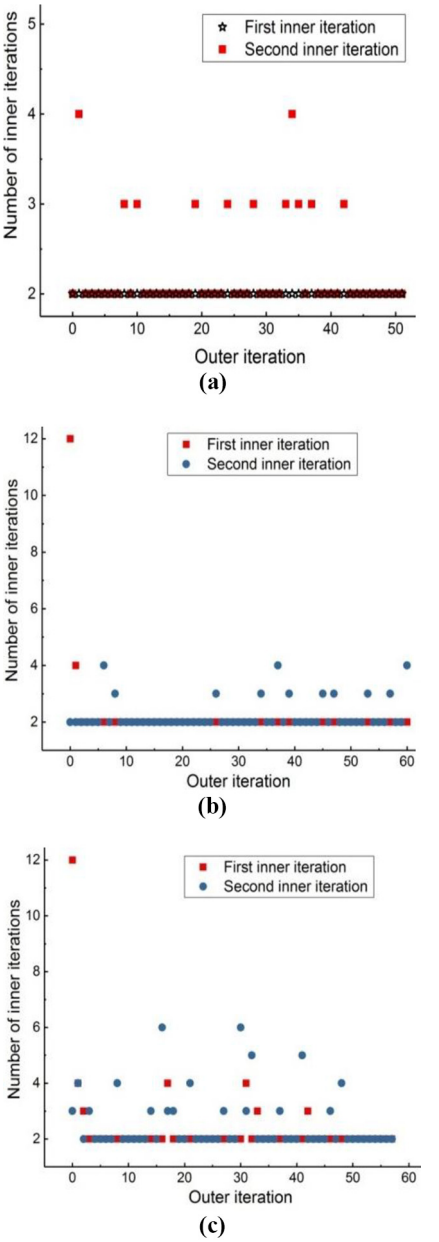


Figure 23.
Variation of inner iteration number at different time steps for Re = 100

Notes: (a) $t = 0$; (b) $t = 0.01$; (c) $t = 0.02$

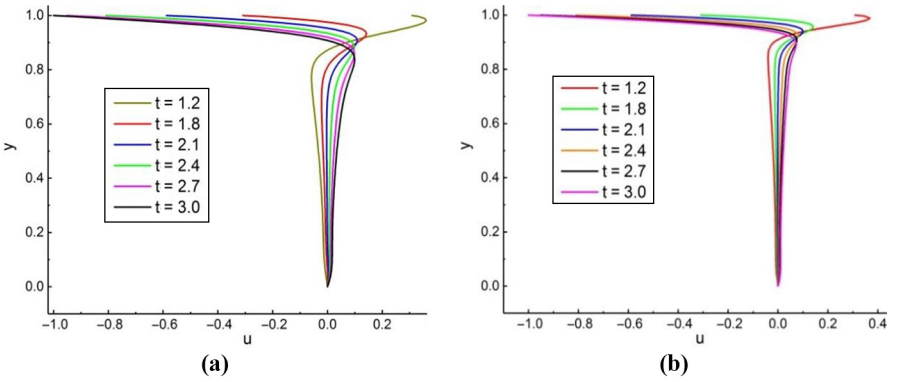


Figure 24.
u-velocity profiles along the vertical centerline of the cavity for different Reynolds numbers

Notes: (a) Re = 400; (b) Re = 1,000

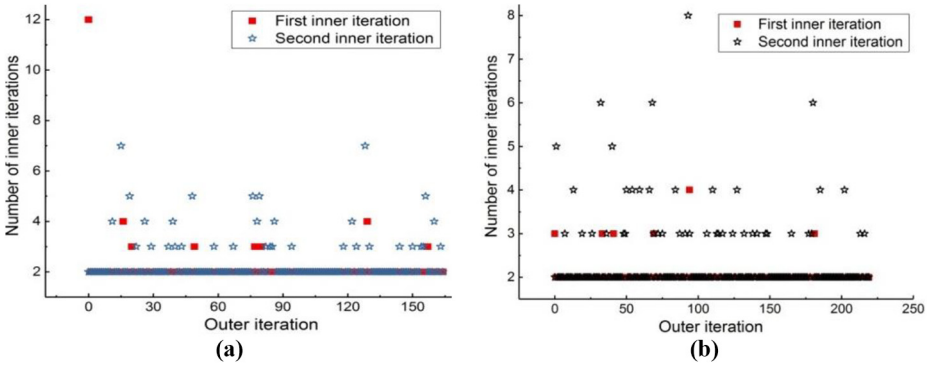


Figure 25.
Variation of inner iteration number for different Reynolds numbers at $t = 0.02$

Notes: (a) Re = 400; (b) Re = 1,000

References

Armaly, B.F., Durst, F., Pereira, J.C.F. and Schonung, B. (1983), "Experimental and theoretical investigation of backward-facing step flow", *Journal of Fluid Mechanics*, Vol. 127 No. 1, pp. 473-496.

Benarda, N., Patricia, S.G., Bonnet, J.P. and Moreaua, E. (2016), "Control of the coherent structure dynamics downstream of a backward facing step by DBD plasma actuator", Vol. 61, pp. 158-173.

Biswas, G., Breuer, M. and Durst, F. (2004), "Backward-Facing step flows for various expansion ratios at low and moderate reynolds numbers", *Journal of Fluids Engineering*, Vol. 126 No. 3, pp. 362-374.

Bruneau, C.H. and Saad, M. (2006), "The 2D lid-driven cavity problem revisited", *Computers and Fluids*, Vol. 35, pp. 326-348.

Cruchaga, M.A. (1998), "A study of the backward-facing step problem using a generalized streamline formulation", *Communications in Numerical Methods in Engineering*, Vol. 14 No. 8, pp. 697-708.

Erturk, E. (2008), "Numerical solutions of 2-D steady incompressible flow over a backward-facing step, part I: High reynolds number solutions", *Computer and Fluids*, Vol. 37, pp. 633-655.

Ghia, U., Ghia, K.N. and Shin, C.T. (1982), "High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method", *Journal of Computational Physics*, Vol. 48 No. 3, pp. 387-411.

Issa, R.I. (1985), "Solution of the implicitly discretised fluid flow equations by Operator-Splitting", *Journal of Computational Physics*, Vol. 62 No. 1, pp. 40-65.

Issa, R.I. (1986), "The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme", *Journal of Computational Physics*, Vol. 62 No. 1, pp. 66-82.

Karki, K.C. and Patankar, S.V. (1989), "Pressure based calculation procedure for viscous flows at all speeds in arbitrary configurations", *AIAA Journal*, Vol. 27 No. 9, pp. 1167-1174.

Kim, J. and Moin, P. (1985), "Application of a fractional-step method to incompressible Navier-Stokes equations", *Journal of Computational Physics*, Vol. 59 No. 2, pp. 308-323.

Kovasznay, L.I.G. (1948), "Laminar flow behind a two-dimensional grid", *Mathematical Proceedings of the Cambridge Philosophical Society*, Vol. 44 No. 1, pp. 58-62.

Maday, Y., Patera, A.T. and Ronquist, E.M. (1990), "An operator-integration-factor splitting method for time-dependent problems: application to incompressible fluid flow", *Journal of Scientific Computing*, Vol. 5 No. 4, pp. 263-292.

Mendu, S.S. and Das, P.K. (2013), "Fluid flow in a cavity driven by an oscillating lid – a simulation by lattice Boltzmann method", *European Journal of Mechanics B/Fluids*, Vol. 39, pp. 59-70.

Musavi, S.H. and Ashrafizaadeh, M. (2016), "A mesh-free lattice Boltzmann solver for flows in complex geometries", *International Journal of Heat and Fluid Flow*, Vol. 59, pp. 10-19.

Patankar, S.V. and Spalding, D.B. (1972), "A calculation procedure for heat mass and momentum transfer in three dimensional parabolic flows", *International Journal of Heat and Mass Transfer*, Vol. 15 No. 10, pp. 1787-1806.

Patankar, S.V. (1981), "A calculation procedure for two-dimensional elliptic situations", *Numer Heat Transfer*, Vol. 4 No. 4, pp. 409-425.

Romanò, F. and Kuhlmann, H.C. (2016), "Numerical investigation of the interaction of a finite-size particle with a tangentially moving boundary", *International Journal of Heat and Fluid Flow*, Vol. 62, pp. 75-82.

Sun, D.L., Qu, Z.G., He, Y.L. and Tao, W.Q. (2008a), "An efficient segregated algorithm for incompressible fluid flow and heat transfer Problems-IDEAL (inner doubly iterative efficient algorithm for linked equations) part II: Application examples", *Numerical Heat Transfer*, Vol. 53 No. 1, pp. 18-38.

Sun, D.L., Qu, Z.G., He, Y.L. and Tao, W.Q. (2008b), "An efficient segregated algorithm for incompressible fluid flow and heat transfer problems-IDEAL(inner doubly iterative efficient algorithm for linked equations) part I: mathematical formulation and solution procedure", *Numerical Heat Transfer*, Vol. 53 No. 1, pp. 1-17.

Tao, W.Q. (2001), *Numerical Heat Transfer*, 2th ed., Xi'an Jiaotong University Press, Xi'an.

Tao, W.Q., Qu, Z.G. and He, Y.L. (2004), "A novel segregated algorithm for incompressible fluid flow and heat transfer problems – CLEAR (coupled and linked equations algorithm revised) part I: Mathematical formulation and solution procedure", *Numerical Heat Transfer, Part B*, Vol. 45, pp. 1-17.

Wang, J.P., Zhang, J.F., Qu, Z.G., He, Y.L. and Tao, W.Q. (2014), "Comparison of robustness and efficiency for SIMPLE and CLEAR algorithm with 13 High-Resolution convection schemes in compressible flows", *Numerical Heat Transfer, Part B: Fundamentals*, Vol. 66 No. 2, pp. 133-161.

Corresponding authors
Jian-Fei Zhang can be contacted at: zhangjf@mail.xjtu.edu.cn and Zhi-Guo Qu can be contacted at: zgqu@mail.xjtu.edu.cn