

Runge-Kutta Finite Element Method Based on the Characteristic for the Incompressible Navier-Stokes Equations

Shaokai Liao^{1,3}, Yan Zhang^{1,2,*} and Da Chen^{2,*}

¹ College of Mechanics and Materials, Hohai University, Nanjing 210089, Jiangsu, China

² College of Harbour, Coastal and Offshore Engineering, Hohai University, Nanjing 210089, Jiangsu, China

³ College of Engineering & Architecture, Jiaying University, Jiaying 314001, Zhejiang, China

Received 25 June 2018; Accepted (in revised version) 28 April 2019

Abstract. In this paper, a finite element method based on the characteristic for the incompressible Navier-Stokes equations is proposed by introducing Runge-Kutta method. At first, coordinate transformation operation is performed to obtain the alternative Navier-Stokes equations without convection term. Then, instead of the classical characteristic-based split (CBS) method, we use the third-order Runge-Kutta method along the characteristic to carry out time discretization in order to improve calculation accuracy, and segregate the calculation of the pressure from that of the velocity based on the momentum-pressure Poisson equation method. Finally, some classical benchmark problems are used to validate the effectiveness of the present method. Compared with the classical method, the present method has lower dissipation, larger permissible time step, and higher time accuracy. The code can be downloaded at DOI: 10.13140/RG.2.2.36336.56329.

AMS subject classifications: 65M12, 76D05

Key words: Finite element method, characteristic, Navier-Stokes equations, Runge-Kutta method, accuracy.

1 Introduction

In the field of fluid dynamics, The incompressible Navier–Stokes (N-S) equations are used to model a number of important physical phenomena. However, as is well known,

*Corresponding author.

Emails: yan.zhang@hhu.edu.cn (Y. Zhang), chenda@hhu.edu.cn (D. Chen)

in FEM, as well as in FDM and FVM, the numerical solution of incompressible N-S equations with the standard Galerkin method leads to spurious oscillatory at high Reynolds number. So significant emphasis has been placed in the literature on developing stabilized and high-accuracy algorithm enough to simulate complex flows [1,2].

The numerical instability is due to the following three aspects. Firstly, the form of the convection term is non-self-adjoint, but the standard Galerkin method adopt a central-difference type approximation to the convection term. Secondly, in convection dominated flows, for which layers appears where the velocity solution and its gradient exhibit rapid variation, the standard Galerkin method leads to numerical oscillations in these layer regions which can spread quickly and pollute the entire solution domain [3]. Lastly, the use of inappropriate combinations of interpolation functions for the velocity and pressure yields unstable schemes [4].

Based on this, the stream upwind Petrov-Galerkin method (SUPG) [5], the Galerkin least square method (GLS) [6,7], the finite increment method (FIC) [8], the penalty function method [9,10] and the Taylor-Galerkin method [11,12] were proposed, which were recognized as the effective stabilization method of the pressure field. The idea of all these methods is based on adding a viscous dissipation term to the original Galerkin formulation of the governing equations, in order to suppress spurious oscillations efficiently. In these methods, the SUPG method [13,14] uses the asymmetric weight function to increase the weight of the inflow direction and reduce the weight of the outflow direction, and then obtain artificial viscous dissipation term. However, the optimal upwind coefficient is usually difficult to determine, and the element matrix needs to be updated at each iteration step, which reduces the computational efficiency. The characteristic Galerkin (CG) [11,15] method discretizes the particle time derivatives along the characteristic instead of the spatial time derivative, so the convection term disappears and the problem is that of simple diffusion, for which the standard Galerkin approximation is optimal.

In order to avoid the complex programming and time consuming of the CG method, Zienkiewicz and Codina [16,17] proposed the characteristic-based time splitting Galerkin method (CBS) and made foundation research. The CBS method takes advantage of local Taylor expansion along the characteristic to eliminate the convection term, and introduces an auxiliary variable velocity to decouple the pressure. Finally, based on the standard Galerkin method, the velocity and pressure field are obtained. The CBS method is well documented and easy to implement, More importantly, it can also use the same interpolation for both velocity and pressure to circumvent the LBB condition [18], so it has been applied widely for the solution of fluid and solid dynamic problems in recent years [19–21], including shallow water flow [22], flow past bluff-bodies [23,24] and porous medium flow [25].

It is worth noting that the CBS method has only first order accuracy and needs to introduce an auxiliary variable velocity, so many researchers develop and improve it from different aspects. By retaining the pressure gradient in the split step and artificial compressibility (AC) parameter, Nithiarasu [26] and He [27] improved the CBS method to second-order accuracy, but the AC parameter needs to be determined empirically.

Bao [28] and Enjilela [29] proposed the two-step method and the pressure projection method in order to improve calculation accuracy and stabilize pressure field. Based on semi-implicit CBS procedures, Chen [30] derived a cell-based smoothed finite element method, which has merits on better robustness against distorted mesh with only slight more computation time. Based on the idea of characteristic-based splitting, Wang and Shui [31, 32] split the N-S equations into the diffusive part and the convective part, and then proposed the characteristic-based operator-splitting finite element method. At present, the research on calculation accuracy and dissipation is still not deep enough. As is well known, the classic Runge-Kutta method [33] is widely used to solve the ordinary differential equation in order to efficiently improve the accuracy and time step of numerical solution, but it is not yet found in the characteristic-based numerical algorithm of fluid dynamics.

In this paper, a characteristic-based finite element method for the incompressible Navier-Stokes (N-S) equations is proposed by introducing the third-order Runge-Kutta method. Firstly, the framework of governing equations and traditional methods are briefly presented. Then, the present method is derived in detail in Section 3. Finally, based on the numerical simulations of some benchmark problems, we validate and discuss the present method with the traditional method in dissipation, accuracy, convergence and computational cost.

2 Governing equations and traditional methods

2.1 Governing equations

The motion of the unsteady incompressible viscous fluid can be expressed by N-S equations including mass conservation equation. without the external force, the non-dimensionalized expressions are

$$\begin{cases} \frac{\partial u_i}{\partial t} = -u_j \frac{\partial u_i}{\partial x_j} - \frac{\partial p}{\partial x_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i}{\partial x_j \partial x_j} \\ \frac{\partial u_i}{\partial x_j} = 0 \end{cases} \quad \Omega \times (0, T), \quad (2.1)$$

where $\Omega \times (0, T)$ are the spatial and temporal domains, t is the time, u_i is the i -component velocity, p is the pressure, and Re is the Reynolds number.

The primitive unknown variables u_i and p are spatially approximated by the shape functions:

$$u_i = \varphi^T \tilde{u}_i, \quad p = \psi^T \tilde{p}, \quad (2.2)$$

where \tilde{u}_i and \tilde{p} are the node value column vectors of the corresponding velocity and pressure, and φ and ψ are the node shape function column vectors of the corresponding velocity and pressure.

2.2 The Galerkin method

The weak weighted integral of the N-S equations and the mass conservation equation is respectively obtained by using the velocity and pressure shape functions, the derivation can be referred to classical references of Zienkiewicz [34] and Bi [35]. The implicit Galerkin method is expressed by

$$\begin{cases} (M + \Delta t(N + D))\tilde{u}_i^{n+1} + \Delta t G_i \tilde{p}^{n+1} = M\tilde{u}_i^n - S, \\ B_i \tilde{u}_i^{n+1} = 0, \end{cases} \quad (2.3)$$

where

$$\begin{aligned} M &= \int_{\Omega^e} \varphi \varphi^T d\Omega, & N &= \int_{\Omega^e} \varphi u_i^n \frac{\partial \varphi^T}{\partial x} d\Omega, & D &= \frac{1}{\text{Re}} \int_{\Omega^e} \frac{\partial \varphi}{\partial x_i} \frac{\partial \varphi^T}{\partial x_i} d\Omega, \\ G_i &= \int_{\Omega^e} \varphi \frac{\partial \psi^T}{\partial x_i} d\Omega, & S &= \frac{1}{\text{Re}} \int_{\Gamma^e} \varphi \frac{\partial u_i^{n+1}}{\partial x_j} n_j d\Gamma, & B_i &= \int_{\Omega^e} \psi \frac{\partial \varphi^T}{\partial x_i} d\Omega, \end{aligned}$$

Δt is the time step.

The velocity and pressure can be obtained by solving simultaneous equations (2.3). It is worth noting that the Galerkin method does not take into account the effects of the convection term, which can cause error or even divergence at high Reynolds number. In the subsequent section, we will further illustrate this by some examples.

2.3 The CBS method

In order to consider the effect of the convection term, Zienkiewicz et al. [17] proposed the CBS method. The derivation can be referred to the classical reference of Zienkiewicz et al.. Now we only give the solving procedure of the CBS method as follows:

(1) Obtain \tilde{u}_i^* by solving the intermediate momentum equation (2.4)

$$M\tilde{u}_i^* = M\tilde{u}_i^n - \Delta t(N + D)\tilde{u}_i^n - \frac{\Delta t^2}{2}C\tilde{u}_i^n + S. \quad (2.4)$$

(2) Obtain \tilde{p}^{n+1} by solving the Poisson equation (2.5)

$$H\tilde{p}^{n+1} = \frac{1}{\Delta t}Q_i\tilde{u}_i^* - L. \quad (2.5)$$

(3) Obtain \tilde{u}_i^{n+1} by solving the velocity correction equation (2.6)

$$M(\tilde{u}_i^{n+1} - \tilde{u}_i^n) = -\Delta t((N + D)\tilde{u}_i^n + G_i\tilde{p}^{n+1}) - \frac{\Delta t^2}{2}(C\tilde{u}_i^n + F_i\tilde{p}^n) + S, \quad (2.6)$$

where M, N, D, S and G_i refer to the Galerkin method, the others are as follows:

$$C = \int_{\Omega^e} \frac{\partial \varphi}{\partial x_j} u_j^n l_k^n \frac{\partial \varphi^T}{\partial x_k} d\Omega, \quad H = \int_{\Omega^e} \frac{\partial \psi}{\partial x_i} \frac{\partial \psi^T}{\partial x_i} d\Omega, \quad Q_i = \int_{\Omega^e} \frac{\partial \psi}{\partial x_i} \varphi^T d\Omega,$$

$$L = \int_{\Gamma^e} \psi u_i^{n+1} n_i d\Gamma, \quad F_i = \int_{\Omega^e} \frac{\partial \varphi}{\partial x_j} u_j^n \frac{\partial \psi}{\partial x_i} d\Omega.$$

Comparing Eq. (2.6) with Eq. (2.3), we can observe that the CBS method considers the influence of convection by introducing the viscous dissipation term $\frac{\Delta t^2}{2} (C\tilde{u}_i^n + F_i\tilde{p}^n)$. Compared with the SUPG method, the CBS method has the uniform viscous dissipation term format and is beneficial to be applied and extended.

3 Characteristic-based Runge-Kutta finite element method

3.1 Math ground

In this subsection, through the coordinate transform, we will derive the two-dimensional N-S equations without the convection term. At a moment, assuming a particle is at $P_1(x_1, y_1)$ in the Euler coordinate system, s_1 is the corresponding arc coordinate in the natural coordinate system, the angle of the streamline direction (s -direction) and the x -direction is α , as shown in Fig. 1, so the direction derivation along the s -direction is shown in Eq. (3.1), and the N-S equations is shown in Eq. (3.2a). After the time interval Δt , the particle reaches $P_2(x_2, y_2)$ along the s -direction. Now a Euler coordinate system and a natural coordinate system are translated by $\overrightarrow{P_1P_2}$ to obtain the new Euler coordinate system ($x'o'y'$) and the new natural coordinates ($o's'$), as shown in Fig. 2. The new coordinate systems are the moving coordinate systems, which change with the time interval Δt . The relationship between the old and new coordinate systems is shown in Eq. (3.2b)

$$\left. \frac{\partial u_i}{\partial s} \right|_{(x_i, y_i)} = \frac{u_i}{U} \left. \frac{\partial u_i}{\partial x_i} \right|_{(x_i, y_i)}, \tag{3.1}$$

where U is the resultant velocity

$$\frac{\partial u_i}{\partial t} = -U \frac{\partial u_i}{\partial s} - \frac{\partial p}{\partial x_i} + \frac{1}{Re} \frac{\partial^2 u_i}{\partial x_j \partial x_j}, \tag{3.2a}$$

$$s' = s - U\Delta t. \tag{3.2b}$$

Based on Eq. (3.2b), the main behavior patterns of Eq. (3.2a) can be determined by a change of the independent variable s to s' . Noting that $u_i = u_i(s', t)$, we have

$$\left. \frac{\partial u_i}{\partial t} \right|_s = \frac{\partial u_i}{\partial s'} \frac{\partial s'}{\partial t} + \left. \frac{\partial u_i}{\partial t} \right|_{s'} = -U \frac{\partial u_i}{\partial s'} + \left. \frac{\partial u_i}{\partial t} \right|_{s'}. \tag{3.3}$$

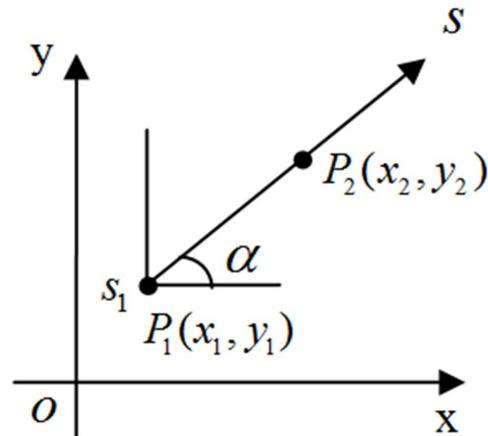


Figure 1: Static coordinate systems.

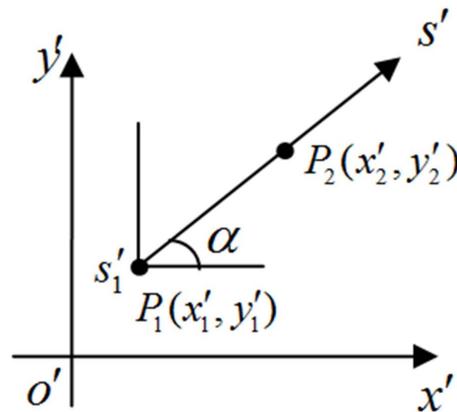


Figure 2: Moving coordinate systems.

So the alternative N-S equation (3.2a) in the moving coordinate system now becomes simply

$$\frac{\partial u_i}{\partial t} \Big|_{s'} = -\frac{\partial p}{\partial x'_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i}{\partial x'_j \partial x'_j}. \quad (3.4)$$

As we hoped, the convection term disappears on the moving coordinate system. Now, Eq. (3.4) is self-adjointness, so it is optimal to use the Galerkin spatial discretization. In the subsequent subsection, firstly, we will carry out the time discretization for Eq. (3.4) based on the third-order Runge-Kutta method, and carry out the spatial discretization based on the Galerkin method. And then we can obtain the finite element equations to solve the velocity and pressure.

3.2 Time discretization

The CBS method has only first-order accuracy for the time discretization of Eq. (3.4). In order to improve time accuracy, we carry out the time discretization of Eq. (3.4) along the streamline characteristic based on the third-order Runge-Kutta method. Noting the moving nature of the coordinates, as shown in Fig. 3, the corresponding time discretization scheme of Runge-Kutta method is as follows:

$$u_i^{n+1} = u_i|_{s_1-ds_1}^n + \frac{\Delta t}{6} (K_1|_{s_1-ds_1}^n + K_2|_{s_1-ds_1/2}^{n+1/2} + K_3^{n+1}), \tag{3.5a}$$

$$K_1|_{s_1-ds_1}^n = \left(-\frac{\partial p}{\partial x'_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i}{\partial x_j'^2} \right) \Big|_{s_1-ds_1}^n, \tag{3.5b}$$

$$K_2|_{s_1-ds_1/2}^{n+1/2} = \left(-\frac{\partial p}{\partial x'_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i}{\partial x_j'^2} \right) \Big|_{s_1-ds_1/2}^{n+1/2}, \tag{3.5c}$$

$$K_3^{n+1} = -\frac{\partial p^{n+1}}{\partial x'_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i^{*n+1}}{\partial x_j' \partial x_j'}, \tag{3.5d}$$

where $u_i|_{s_1-ds_1}^n$ is the velocity, $K_1|_{s_1-ds_1}^n$ and $K_2|_{s_1-ds_1/2}^{n+1/2}$ are the slopes. Their right superscript indicates the time, their right subscript indicates the location, which demonstrates all of the above are expressed in the moving coordinate system. K_3^{n+1} is the slope of time t^{n+1} at the location of s_1 , which is determined in the static coordinate system. In order to facilitate expression, we default the right subscript in the static coordinate system. u_i^{*n+1} is the predicted velocity of time t^{n+1} , which can be expressed as follows:

$$u_i^{*n+1} = u_i|_{s_1-ds_1}^n + 2\Delta t K_2|_{s_1-ds_1/2}^{n+1/2} - \Delta t K_1|_{s_1-ds_1}^n. \tag{3.6}$$

In order to circumvent the difficulty of mesh updating due to the moving coordinate system, which arises in the original characteristic-based method, we introduce two characteristics, as shown in Fig. 3. The characteristic 1 indicates that the particle moves the

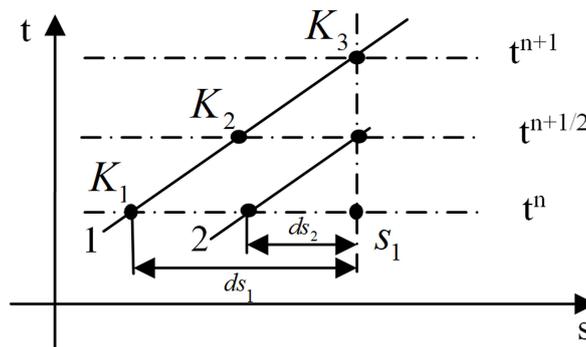


Figure 3: Time discretization along the characteristics.

distance ds_1 along the streamline direction at an average velocity U_1 from time t^n to time t^{n+1} . The characteristic 2 indicates that the particle moves the distance ds_2 along the streamline direction at an average velocity U_2 from time t^n to time $t^{n+1/2}$. So the two characteristics can be expressed:

$$ds_1 = U_1 \Delta t, \quad (3.7a)$$

$$ds_2 = U_2 \frac{\Delta t}{2}, \quad (3.7b)$$

where the average velocity along the characteristic can be approximated:

$$U_1 = U|_{s_1-ds_1/2}^{n+1/2}, \quad (3.8a)$$

$$U_2 = U|_{s_1-ds_2}^n. \quad (3.8b)$$

Using the Taylor expansion, we have:

$$U_1 = U^{n+1/2} - \frac{U^{n+1/2}}{2} \Delta t \frac{\partial U^{n+1/2}}{\partial s} + o(\Delta t^2), \quad (3.9a)$$

$$U_2 = U^n - \frac{U^n}{2} \Delta t \frac{\partial U^n}{\partial s} + o(\Delta t^2). \quad (3.9b)$$

Therefore, the Taylor expansion of an arbitrary amount " f " along the characteristic 1 and 2 can be expressed as follows:

$$f|_{s_1-ds_1}^n = f^n - ds_1 \frac{\partial f^n}{\partial s} + \frac{ds_1^2}{2} \frac{\partial^2 f^n}{\partial s^2} + o(\Delta t^3), \quad (3.10a)$$

$$f|_{s_1-ds_1/2}^{n+1/2} = f^{n+1/2} - \frac{ds_1}{2} \frac{\partial f^{n+1/2}}{\partial s} + \frac{ds_1^2}{8} \frac{\partial^2 f^{n+1/2}}{\partial s^2} + o(\Delta t^3), \quad (3.10b)$$

$$f|_{s_1-ds_2}^n = f^n - ds_2 \frac{\partial f^n}{\partial s} + \frac{ds_2^2}{2} \frac{\partial^2 f^n}{\partial s^2} + o(\Delta t^3). \quad (3.10c)$$

Based on Eqs. (3.1) and (3.7a)-(3.10c), and ignoring high order items, Eqs. (3.5a)-(3.5c) can be expressed as follows:

$$u_i|_{s_1-ds_1}^n = u_i^n - \Delta t u_j^{n+1/2} \frac{\partial u_i^n}{\partial x_j} + \frac{\Delta t^2}{2} u_k^{n+1/2} \frac{\partial}{\partial x_k} \left(u_j^{n+1/2} \frac{\partial u_i^n}{\partial x_j} \right), \quad (3.11a)$$

$$K_1|_{s_1-ds_1}^n = -\frac{\partial p^{n+\theta_1}}{\partial x_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i^n}{\partial x_j \partial x_j} - \Delta t u_k^{n+1/2} \frac{\partial}{\partial x_k} \left(-\frac{\partial p^{n+\theta_1}}{\partial x_i} \right), \quad (3.11b)$$

$$K_2|_{s_1-ds_1/2}^{n+1/2} = -\frac{\partial p^{n+\theta_1}}{\partial x_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i^{n+1/2}}{\partial x_j \partial x_j} - \frac{\Delta t}{2} u_k^{n+1/2} \frac{\partial}{\partial x_k} \left(-\frac{\partial p^{n+\theta_1}}{\partial x_i} \right), \quad (3.11c)$$

where $u_i^{n+1/2}$ can be solved using the Euler method:

$$u_i^{n+1/2} = u_i^n \Big|_{s_1-ds_2} + \frac{\Delta t}{2} \left(-\frac{\partial p}{\partial x_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i}{\partial x_j \partial x_j} \right) \Big|_{s_1-ds_2}$$

$$u_i^{n+1/2} = u_i^n - \frac{\Delta t}{2} \left(u_j^n \frac{\partial u_i^n}{\partial x_j} + \frac{\partial p^{n+\theta_1}}{\partial x_i} - \frac{1}{\text{Re}} \frac{\partial^2 u_i^n}{\partial x_j \partial x_j} \right) + \frac{\Delta t^2}{8} u_k^n \frac{\partial}{\partial x_k} \left(u_j^n \frac{\partial u_i^n}{\partial x_j} + 2 \frac{\partial p^{n+\theta_1}}{\partial x_i} \right). \quad (3.12)$$

Substituting Eqs. (3.11a)-(3.11c), into Eq. (3.6), we have the following:

$$u_i^{*n+1} = u_i^n - \Delta t \left(u_j^{n+1/2} \frac{\partial u_i^n}{\partial x_j} + \frac{\partial p^{n+\theta_1}}{\partial x_i} - \frac{1}{\text{Re}} \frac{\partial^2 (2u_i^{n+1/2} - u_i^n)}{\partial x_j \partial x_j} \right)$$

$$+ \frac{\Delta t^2}{2} u_k^{n+1/2} \frac{\partial}{\partial x_k} \left(u_j^{n+1/2} \frac{\partial u_i^n}{\partial x_j} \right). \quad (3.13)$$

Substituting Eqs. (3.11a)-(3.13) into Eqs. (3.5a), we can obtain the velocity of time t^{n+1} :

$$u_i^{n+1} = u_i^n - \Delta t \left(u_j^{n+1/2} \frac{\partial u_i^n}{\partial x_j} + \frac{\partial p^{n+\theta_1}}{\partial x_i} - \frac{1}{6\text{Re}} \frac{\partial^2 (u_i^n + 4u_i^{n+1/2} + u_i^{*n+1})}{\partial x_j \partial x_j} \right)$$

$$+ \frac{\Delta t^2}{2} u_k^{n+1/2} \frac{\partial}{\partial x_k} \left(u_j^{n+1/2} \frac{\partial u_i^n}{\partial x_j} + \frac{\partial p^{n+\theta_1}}{\partial x_i} \right), \quad (3.14)$$

where $\theta_1 \in [0, 1]$; when $\theta_1 = 0$, it is an explicit method, when $0 < \theta_1 < 1$, it is a semi-implicit method, when $\theta_1 = 1$, the method is a implicit method. In this paper, $\theta_1 = 0$ in Eqs. (3.12), (3.13), $\theta_1 = 1$ in Eq. (3.14).

Obviously, before solving Eq. (3.14), we must solve the pressure of time t^{n+1} . Contrary to the CBS method, we directly take the divergence of Eq. (3.14) without splitting it. Considering the incompressible condition of time t^{n+1} and ignoring high order stability term, the pressure Poisson equation can be obtained :

$$\frac{\partial p^{n+1}}{\partial x_i \partial x_i} = \frac{1}{\Delta t} \frac{\partial u_i^n}{\partial x_i} - \frac{\partial}{\partial x_i} \left(u_j^{n+1/2} \frac{\partial u_i^n}{\partial x_j} - \frac{1}{6\text{Re}} \frac{\partial^2 (u_i^n + 4u_i^{n+1/2} + u_i^{*n+1})}{\partial x_j \partial x_j} \right). \quad (3.15)$$

3.3 Spatial discretization

Based on the standard Galerkin finite element method, we carry out the spatial discretization for Eqs. (3.12)-(3.15), respectively. The weight function is derived from the shape function of Eq. (2.2). Firstly, multiplying the pressure weight function by Eq. (3.15), and spatially integrating in the computational domain, we can obtain the weak integral form of Eq. (3.15). Substituting Eq. (3.14) into the weak integral form and ignoring its stability term, we have

$$H\tilde{p}^{n+1} = \frac{1}{\Delta t} Q_i \tilde{u}_i^n - R\tilde{u}_i^n - \frac{1}{\Delta t} L, \quad (3.16)$$

where

$$\begin{aligned}
 H &= \int_{\Omega^e} \frac{\partial \psi}{\partial x_i} \frac{\partial \psi^T}{\partial x_i} d\Omega, & Q_i &= \int_{\Omega^e} \frac{\partial \psi}{\partial x_i} \varphi^T d\Omega, \\
 R_i &= \int_{\Omega^e} \frac{\partial \psi}{\partial x_i} u_j^{n+1/2} \frac{\partial \varphi^T}{\partial x_j} d\Omega, & L &= \int_{\Gamma^e} \psi u_i^{n+1} n_i d\Gamma.
 \end{aligned}$$

Then, multiplying the velocity weight function by Eqs. (3.12)-(3.14), respectively, and spatially integrating in the computational domain, we can obtain the following discrete momentum equations:

$$M(\tilde{u}_i^{n+1/2} - \tilde{u}_i^n) = -\frac{\Delta t}{2}((N+D)\tilde{u}_i^n + G_i \tilde{p}^n) - \frac{\Delta t^2}{2}(C^n \tilde{u}_i^n + 2F_i \tilde{p}^n) + S^n, \tag{3.17a}$$

$$\begin{aligned}
 M(\tilde{u}_i^{*n+1} - \tilde{u}_i^n) &= -\Delta t(N^{n+1/2} \tilde{u}_i^n + D(2\tilde{u}_i^{n+1/2} - \tilde{u}_i^n) + G_i \tilde{p}^n) \\
 &\quad - \frac{\Delta t^2}{2} C^{n+1/2} \tilde{u}_i^n + 2S^{n+1/2} - S^n,
 \end{aligned} \tag{3.17b}$$

$$\begin{aligned}
 M(\tilde{u}_i^{*n+1} - \tilde{u}_i^n) &= -\Delta t(N^{n+1/2} \tilde{u}_i^n + \frac{D}{6}(\tilde{u}_i^n + 4\tilde{u}_i^{n+1/2} + \tilde{u}_i^{*n+1}) + G_i \tilde{p}^{n+1}) \\
 &\quad - \frac{\Delta t^2}{2}(C^{n+1/2} \tilde{u}_i^n + F_i^{n+\theta_2} \tilde{p}^{n+1}) + 2S^{n+1/2} \\
 &\quad - \frac{1}{6}(S^n + 4S^{n+1/2} + S^{n+1}),
 \end{aligned} \tag{3.17c}$$

where

$$\begin{aligned}
 M &= \int_{\Omega^e} \varphi \varphi^T d\Omega, \quad N^{n+\theta_2} = \int_{\Omega^e} \varphi u_i^{n+\theta_2} \frac{\partial \varphi^T}{\partial x_i} d\Omega, \quad D = \frac{1}{\text{Re}} \int_{\Omega^e} \frac{\partial \varphi}{\partial x_i} \frac{\partial \varphi^T}{\partial x_i} d\Omega, \\
 G_i &= \int_{\Omega^e} \varphi \frac{\partial \psi^T}{\partial x_i} d\Omega, \quad F_i^{n+\theta_2} = \int_{\Omega^e} \frac{\partial \varphi}{\partial x_j} u_j^{n+\theta_2} \frac{\partial \psi}{\partial x_i} d\Omega, \quad C^{n+\theta_2} = \int_{\Omega^e} \frac{\partial \varphi}{\partial x_j} u_j^{n+\theta_2} u_k^{n+\theta_2} \frac{\partial \varphi^T}{\partial x_k} d\Omega, \\
 S^{n+\theta_2} &= \frac{1}{\text{Re}} \int_{\Gamma^e} \varphi \frac{\partial u_i^{n+\theta_2}}{\partial x_j} n_j d\Gamma,
 \end{aligned}$$

θ_2 is 0, 1/2 and 1.

Comparing Eq. (3.17c) with Eq. (2.6), we can observe that the diffusion term of the present method is approximated through the third-order Runge-Kutta method, and its viscous dissipation term is slightly different from the CBS method.

3.4 Process of the present method

Based on Runge-Kutta finite element method along the characteristic, we have obtained all the finite element equations. Its solution steps are summarized:

- (1) Obtain the velocity $u_i^{n+1/2}$ of time $t^{n+1/2}$ by solving the momentum equation (3.17a),

- (2) Obtain the pressure p^{n+1} of time t^{n+1} by solving the pressure Poisson equation (3.16),
- (3) Obtain the predicted velocity u_i^{*n+1} of time t^{n+1} by solving Eq. (3.17b) based on the Runge-Kutta method,
- (4) Obtain the velocity u_i^{n+1} of the time t^{n+1} by solving Eq. (3.17c) based on the Runge-Kutta method.

The solution of the all above equations is realized by compiling Matlab program. The code can be downloaded at DOI: 10.13140/RG.2.2.36336.56329.

4 Numerical validation and comparison

In this section, we apply the above three methods to numerically simulate incompressible flow in a lid-driven cavity. The problem has been studied by many scholars but probably the most detailed investigation was that of Erturk and Ghia [36,37], in which they quote many solutions for different Reynolds numbers based on high precision mesh. We will use those results for comparison, in order to reveal viscous dissipation, accuracy and convergence of the three methods. In the following simulations, we have used equal-order velocity-pressure interpolation function, and adopted the bilinear quadrilateral element with structured mesh.

4.1 Boundary conditions and mesh

The classic model of flow in a cavity is shown in Fig. 4. The dimensionless cavity sizes are 1×1 . Dimensionless velocities $u_x = 1$ and $v_y = 0$ are imposed on the top wall, and the other three sides are solid walls where nonslip boundary conditions $u_x = 0$ and $v_y = 0$ are imposed. The coordinate origin is at the lower left corner where the relative pressure $p = 0$. Two kinds of mesh are employed to simulate flow in cavity at different Reynolds numbers (Re), as shown in Table 1. The mesh 80×80 is nonuniform structured mesh, as shown in Fig. 5, and the others are uniform structured mesh.

Table 1: Mesh for different Re.

	Re = 1000	Re = 5000	Re = 7500
Coarse mesh	$20 \times 20, 40 \times 40$	$40 \times 40, 50 \times 50$	40×40
Fine mesh	50×50	80×80	100×100

4.2 Viscous dissipation

When Re is 1000, the uniform mesh 20×20 is used, and the time step size is 0.01. we adopt the above three methods to carry out numerical simulation. The comparisons of

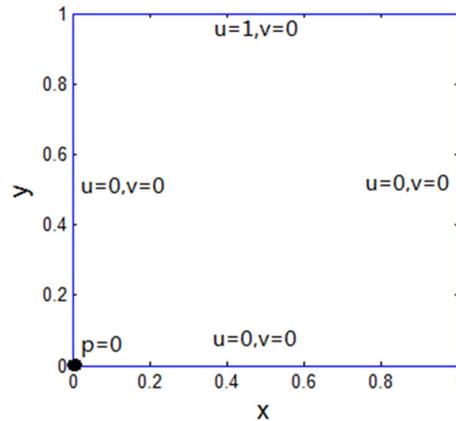
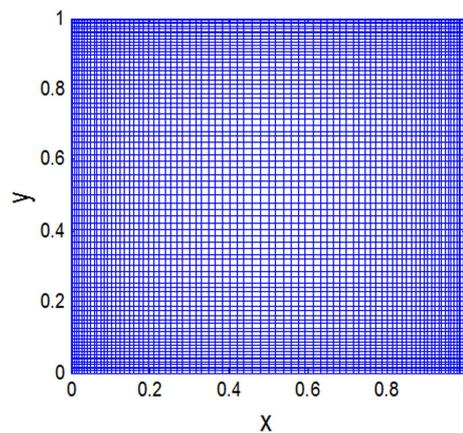
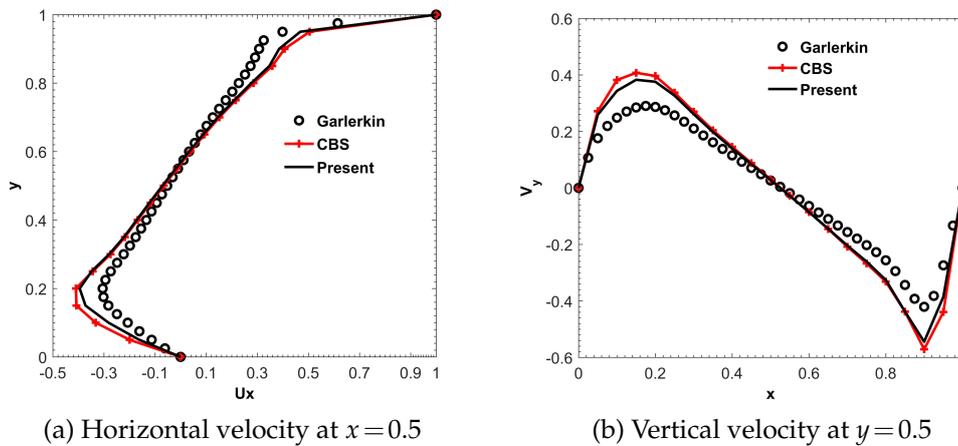
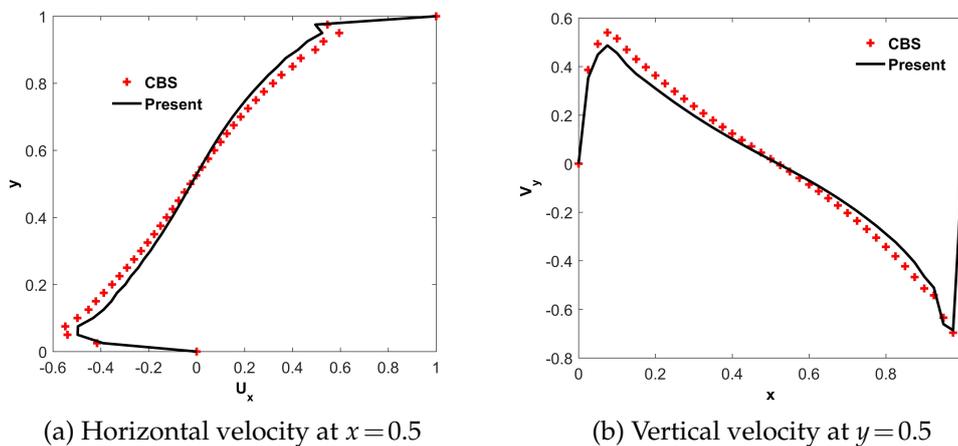


Figure 4: Flow in a cavity.

Figure 5: Non-uniform mesh 80×80 .

horizontal velocity profiles along the cavity vertical section $x=0.5$ are shown in Fig. 6(a), and the comparisons of vertical velocity profiles along the cavity horizontal section $y=0.5$ are shown in Fig. 6(b). Because the Galerkin method does not introduce viscous dissipation terms to consider the influence of convection term, the calculation results are mainly controlled by diffusion term, so the obtained velocity field is more uniform. On the contrary, the CBS method and the present method introduce viscous dissipation terms to increase the influence of convection term, and make the velocity field change violently. It can be seen from Fig. 6 that the extreme and slope of velocity obtained by the Galerkin method are the smallest, the CBS method is the largest, and the present method is between the Galerkin method and the CBS method. In other words, the Galerkin method is zero dissipation, and the present method has lower dissipation than the CBS method.

Now, we continue to increase Re to 7500, and use the CBS method and the present

Figure 6: Velocity profiles at $Re=1000$ when using coarse mesh 20×20 .Figure 7: Velocity profiles at $Re=7500$ when using coarse mesh 40×40 .

method to simulate. The uniform mesh 40×40 is used, and the time step size is 0.02. We obtain the velocity profiles along the two symmetric axes, as shown in Fig. 7. Firstly, due to the increase of Re , the convection effect is more obvious, and the velocity field change more violently. Secondly, the velocity profiles obtained by the CBS method are steeper than the present method, which further illustrate the present method has lower dissipation.

4.3 Calculation accuracy

The fine meshes in Table 1 are chosen to simulate flow in a cavity of different Re . To determine the accuracy of the three methods, we compare the numerical results with

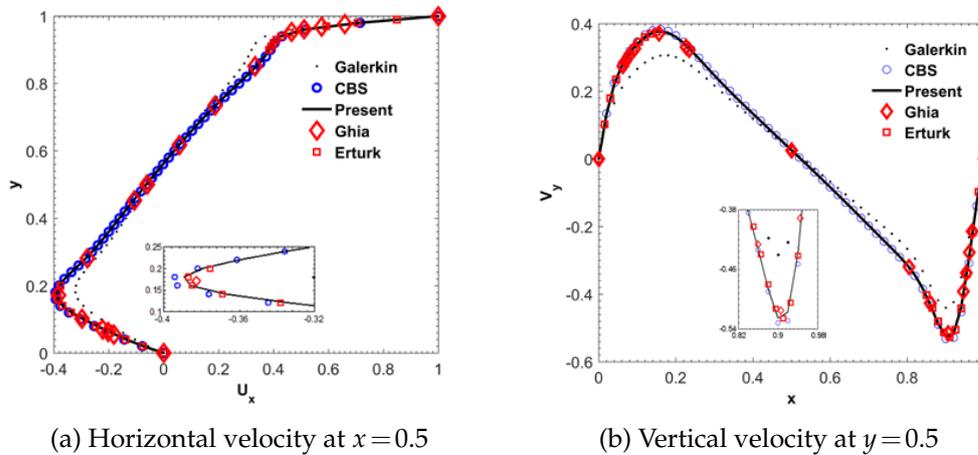


Figure 8: Velocity profiles at $Re = 1000$ when using fine mesh 50×50 .

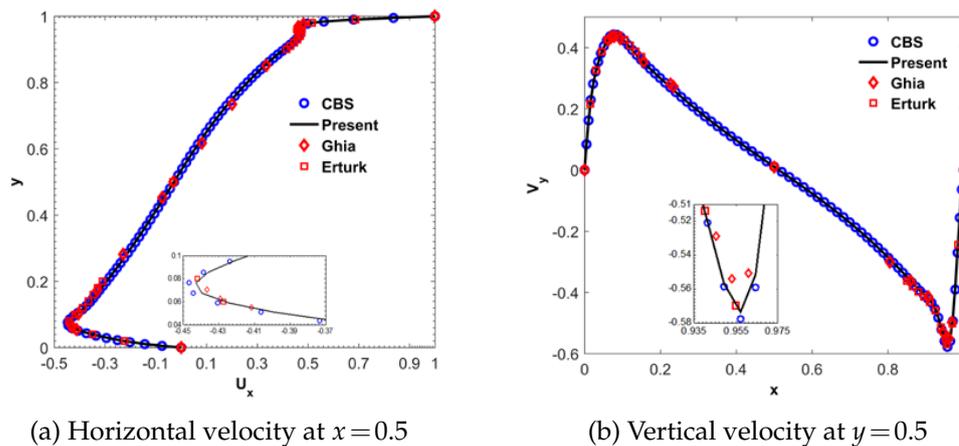


Figure 9: Velocity profiles at $Re = 5000$ when using fine mesh 80×80 .

the reference data that are obtained by Erturk and Ghia [36, 37] with a finer uniform grid mesh of 601×601 . When Re is 1000, the comparisons of horizontal velocity profiles along the cavity vertical section $x = 0.5$ are shown in Fig. 8(a), the comparisons of vertical velocity profiles along the cavity horizontal section $y = 0.5$ are shown in Fig. 8(b), and the minimum horizontal and vertical velocity is given in Table 2. It is observed that the Galerkin method result in bad accuracy due to the lack of viscous dissipation term, the velocity profiles obtained by the present method are closer to the reference data obtained by Erturk than those obtained by the CBS method.

In order to better show the accuracy difference of the CBS method and the present method, we simulate flow in a cavity of higher Re . Fig. 9 show the velocity profiles at $Re = 5000$, Fig. 10 show the velocity profiles at $Re = 7500$, and Table 2 gives the quantitative

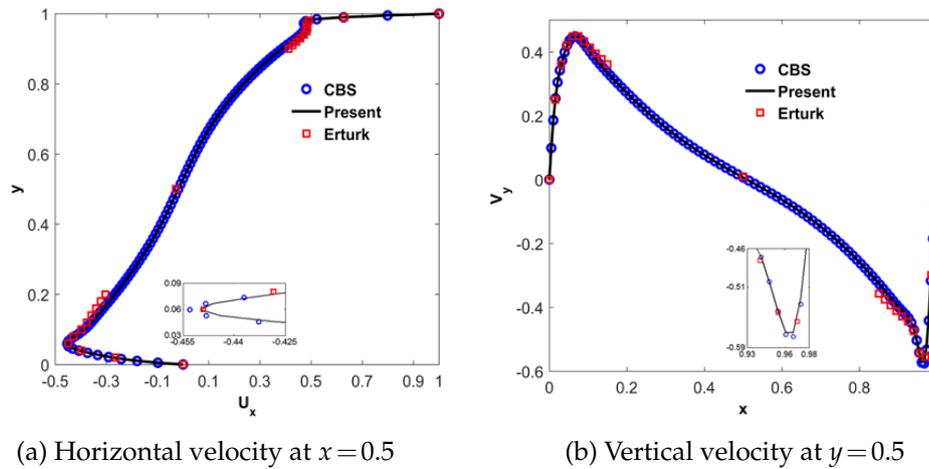


Figure 10: Velocity profiles at $Re=7500$ when using fine mesh 100×100 .

Table 2: Comparison of minimum velocity at different Re .

method	minimum horizontal velocity					minimum vertical velocity		
	fine mesh			coarse mesh		fine mesh		
	1000	5000	7500	5000	7500	1000	5000	7500
Galerkin	-0.321	–	–	–	–	-0.4402	–	–
CBS	-0.3941	-0.4465	-0.4530	-0.5043	-0.5483	-0.5321	-0.5779	-0.5753
Present	-0.3896	-0.4427	-0.4499	-0.4838	-0.4957	-0.5259	-0.5735	-0.5698
Erturk [29]	-0.3869	-0.4419	-0.4491	-0.4419	-0.4491	-0.5263	-0.5700	-0.5550

comparison of minimum velocity at different Re . The results further illustrate the present method has higher accuracy than the CBS method. There are two reasons to explain the improvement of accuracy. Firstly, the diffusion term of the CBS method is based on the first-order Euler method, and that of the present method is based on the third-order Runge-Kutta method. Secondly, the present method has lower dissipation than the CBS method.

In order to determine the dependence of accuracy on mesh, we choose the coarse mesh in Table 1 to simulate. We obtain the horizontal velocity profiles along the section $x=0.5$ at different Re , as shown in Fig. 11. The corresponding quantitative comparison is shown in Table 2. The present method still obtains good results under the coarse mesh, which are closer to the reference data obtained by Erturk than those of the CBS method. Thus, comparing the present method, the CBS method has a higher requirement for mesh number.

Based on the present method, we can obtain the pressure contour and streamline contour for $Re = 1000, 5000$ and 7500 , as shown in Figs. 12 and 13. As seen from Fig. 12, the results are smooth and free of oscillations. As seen from Fig. 13, secondary vortices on the both bottom corners are predicted excellently by the present method. Particularly

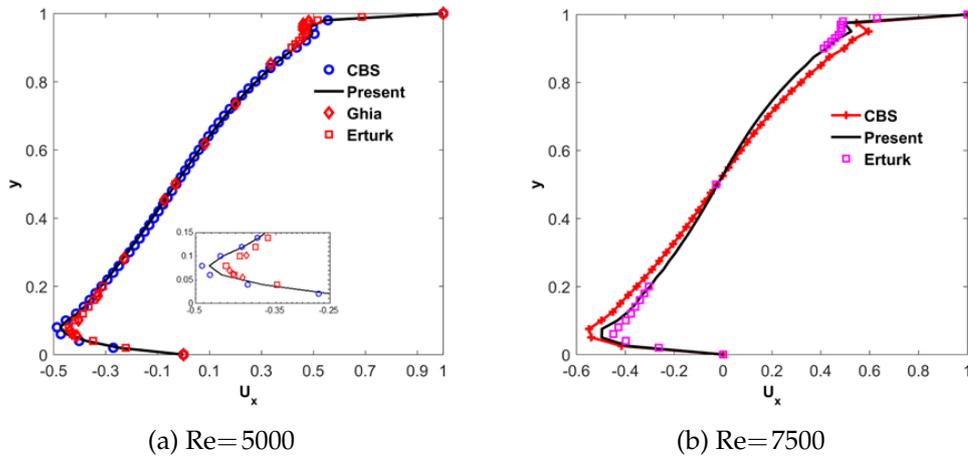


Figure 11: Horizontal velocity profiles along the section $x=0.5$ at different Re.

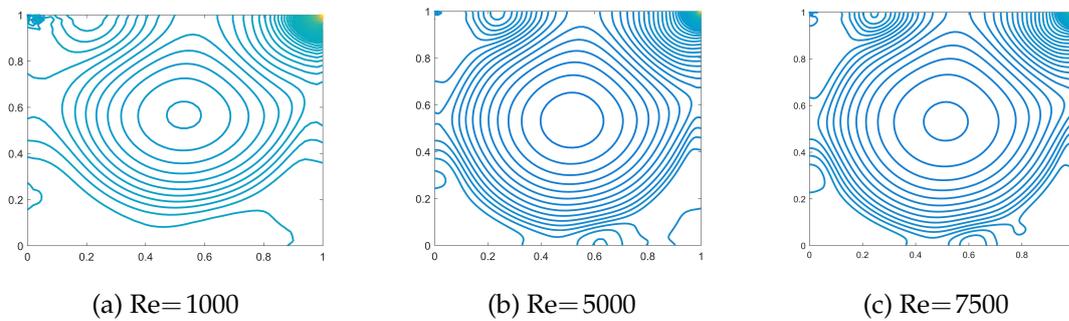


Figure 12: Streamline contours for flow in a cavity at different Re when using fine mesh and the present method.

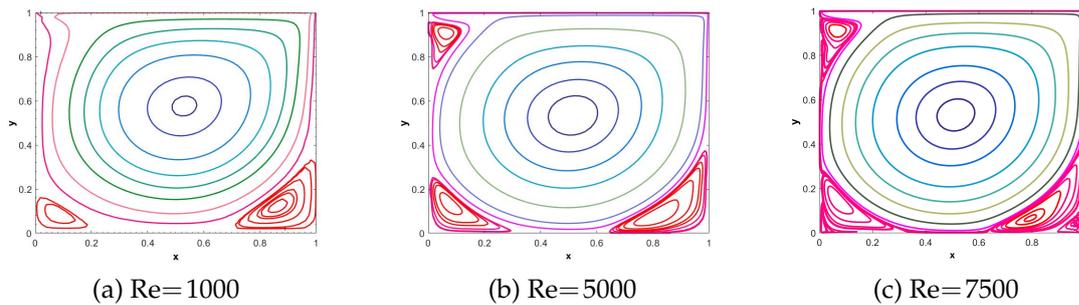


Figure 13: Pressure contours for flow in a cavity at different Re when using fine mesh and the present method.

when Re is 7500, a small corner vortex at the bottom right corner has been predicted by the present method, which normally requires a very high mesh resolution.

4.4 Convergence and computational cost

The above three methods are explicit form and are conditionally stable. The permissible time step is governed by the critical step as follows:

$$\Delta t \leq \min \left(\frac{h}{|u|}, \frac{h^2}{2v} \right), \tag{4.1}$$

where v is the kinematic viscosity.

In fact, the permissible time step of many algorithms is difficult to reach the critical time step. Now, based on the uniform coarse mesh in Table 1, we use the large time step to test the convergence of the three methods for different Re. The convergence is defined by L_2 norm of velocity field as follows:

$$\sqrt{\sum_{i=1}^{No.of\ nodes} (|u_i^{n+1} - |u_i^n|^2)} / \sqrt{\sum_{i=1}^{No.of\ nodes} (|u_i^{n+1}|)^2} \leq 10^{-5}. \tag{4.2}$$

The chosen time step and convergence situation are shown in Table 3. Here, "C" represents convergence, and "D" represents divergence. Based on the results of mesh 20×20 , we can obtain that the permissible time step of the Galerkin method is the smallest. We continue to discuss the convergence of the present method and the CBS method, Fig. 14 shows the change of L_2 norm with the increase of time steps, which further illustrate that the convergence of the present method is better than the CBS method. It also means that the present method have larger critical time step size and can save calculation time.

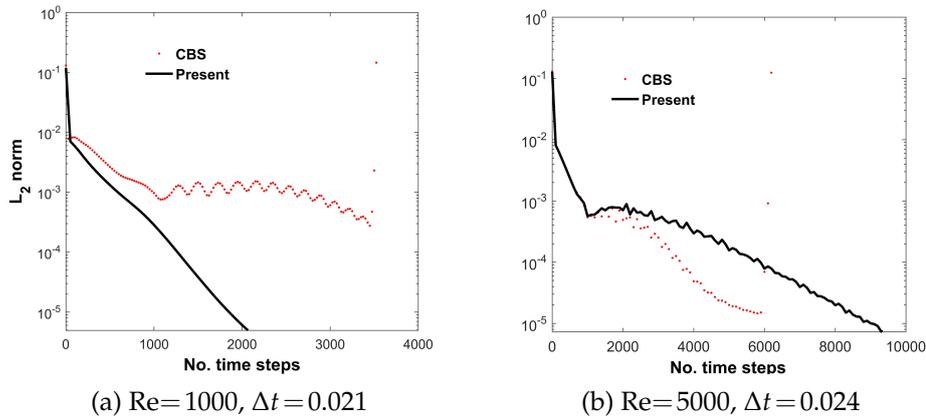
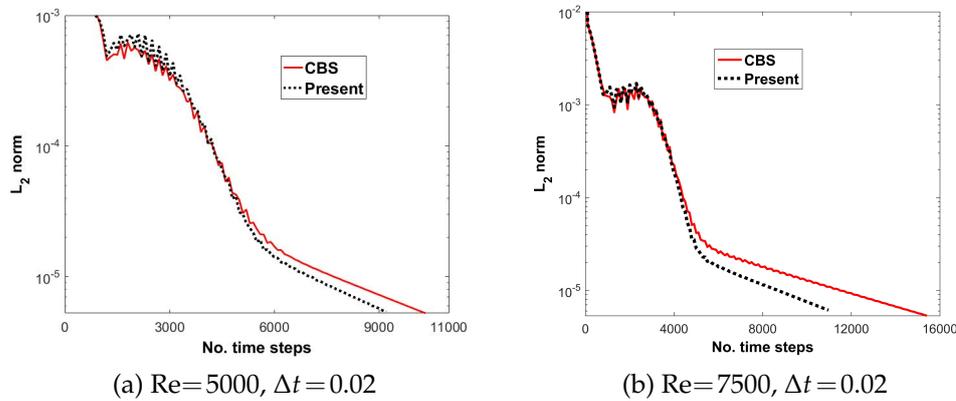
Now, we discuss the computational cost of the CBS method and the present method. The total computational cost is determined by the product of the running time of single

Table 3: Convergence situation of different time step.

method	Time step							
	Mesh 20×20		Mesh 40×40				Mesh 50×50	
	Re=1000		Re=1000		Re=5000		Re=5000	
	0.04	0.01	0.019	0.021	0.02	0.024	0.01	0.015
Galerkin	D	C	-	-	-	-	-	-
CBS	C	C	D	D	C	D	C	D
present	C	C	C	C	C	C	C	C

Table 4: Comparison of the computational cost.

Re	method	total iteration	running times (single step)		Total computation
		step number	updating matrix	solving equation	(s)
5000	CBS	10339	1	5	62304
	present	9207	2	7	82863
7500	CBS	15412	1	5	92472
	present	10969	2	7	98721

Figure 14: Convergence and divergence histories at different Re when using mesh 40×40 .Figure 15: Convergence histories when using the same mesh 40×40 and time step.

step and total iterative step number. The running times of single step of the present method is 1.5 times that of the CBS method, because the CBS method needs to update element matrix once and solve equations five times, and the present method needs to update element matrix twice and solve equations seven times in each iteration step. It is worth noting that the CBS method is based on the single-step Euler method for time discretization, but the present method adopt the three-step Runge-Kutta method, which contributes to improve the convergence of the algorithm. In order to illustrate that point, based on the same mesh 40×40 and time step, we obtain the convergence histories and the total iteration step number of the two methods, as shown in Fig. 15 and Table 4. It is observed that the present method can reach the tolerance faster than the CBS method, which is beneficial to reduce the computational cost. In general, considering the product of the running time of single step and total iterative step number, the total computational cost of the present method is slightly higher than that of the CBS method, but within

acceptable limit.

5 Conclusions

In this paper, the Galerkin method and the CBS method for solving fluid mechanics equations are given. On this basis, using coordinate transformation along the characteristic, we derive two dimensional N-S equations without the convection term under the moving coordinate. Based on Runge-Kutta method along the characteristic, a third-order time discretization scheme of the alternative N-S equations is proposed. In order to avoid numerical solving difficulty from mesh updating, the local Taylor expansion along the characteristic is used to express the velocities and pressures of different time. Meanwhile, in order to avoid splitting the momentum equation, the divergence theorem is applied to the momentum equation, and then the pressure Poisson equation can be obtained. Finally, applying the Galerkin spatial approximation, we obtain the characteristic-based Runge-Kutta finite element method.

We carry out numerical calculations of flow in a cavity, in order to validate the effectiveness of the present method. Based on the coarse mesh and large time step, Comparing with the traditional Galerkin method and the CBS method, the present method has lower dissipation and more convergence due to the introduce of the third order Runge-Kutta method. Based on the fine mesh, we further confirm that the present method has higher calculation accuracy than the other methods. Because the present method needs to update the total element matrix twice and solve equations seven times in each iteration step, its running times of single step is obviously higher than that of the CBS method. However, the present method has bigger critical time step and reach the tolerance faster than the CBS method, so the total computational cost of the present method is slightly higher than that of the CBS method. In the future work, we will further focus on how to reduce the running times of single step to save the computational cost.

Acknowledgements

This work was supported by the National Nature Science Foundation of China (Project Nos. 51579088, 51509081 and 51779087), the Natural Science Foundation of Jiangsu province (Nos. BK20161507, BK20150037 and BK20150811), the Fundamental Research Funds for the Central Universities (No. 2018B17114) and Science and Technology Project of Jiaxing (No. 2016AY13009).

References

- [1] G. F. PANG, W. CHEN AND K. Y. SZE, *A comparative study of finite element and finite difference methods for two-dimensional space-fractional advection-dispersion equation*, Adv. Appl. Math. Mech., 8(1) (2016) pp. 166–186.

- [2] Z. D. LUO, *A new finite volume element formulation for the non-stationary Navier-Stokes equations*, Adv. Appl. Math. Mech., 6(5) (2014) pp. 615–636.
- [3] E. HACHEM, B. RIVAUS AND T. KLOCZKO ET AL., *Stabilized finite element method for incompressible flows with high Reynolds number*, J. Comput. Phys., 229 (2010), pp. 8643–8665.
- [4] L. FRANCA AND A. NESLITURK, *On a two-level finite element method for the incompressible Navier-Stokes equations*, Int. J. Numer. Meth. Eng., 52 (2001), pp. 433–453.
- [5] A. N. BROOKS AND T. J. R. HUGHES, *Streamline upwind/Petrov-Galerkin formulation for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equation*, Comput. Method. Appl. Mech. Eng., 32 (1982), pp. 199–259.
- [6] T. J. R. HUGHES, L. P. FRANCA AND G. M. HULBEERT, *A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations*, Comput. Method. Appl. Mech. Eng., 73 (1989), pp. 173–189.
- [7] N. KONDO AND S. YAMADA, *Third-order upwind finite element computation of the incompressible Navier-Stokes equations Part I. Computation of flow around rectangular cylinders*, Comput. Method. Appl. Mech. Eng., 127 (1995), pp. 87–97.
- [8] E. ONATE, *A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation*, Comput. Method. Appl. Mech. Eng., 182 (2000), pp. 355–370.
- [9] J. C. HEINRICH, R. S. MARSHALL AND O. C. ZIENKIEWICZ, *Penalty function solution of coupled convective and conductive heat transfer*, Numerical Methods in Laminar and Turbulent Flow, (1979), pp. 935–946.
- [10] T. J. R. HUGHES, W. K. LIU AND A. N. BROOKS, *Finite element analysis of incompressible viscous flows by the penalty function formulation*, J. Comput. Phys., 30 (1979), pp. 1–60.
- [11] J. DOUGLAS AND T. F. RUSSELL, *Numerical methods for convection dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures*, SIAM J. Numer. Anal., 19 (1982), pp. 871–885.
- [12] J. DONERA, *A Taylor-Galerkin method for convection transport problems*, Int. J. Numer. Meth. Eng., 20 (1984), pp. 101–119.
- [13] M. TABATA AND S. FUJIMA, *An upwind finite element scheme for high Reynolds number flows*, Int. J. Numer. Methods Fluids, 12 (1991), pp. 305–322.
- [14] H. M. ZHOU, B. YAN AND Y. LI, *Glass pressing simulation based on the PG method*, Int. J. Comput. Fluid D., 22(3) (2008), pp. 201–207.
- [15] R. L. T. BEVAN, R. VANLOON AND P. NITHIARASU, *The locally conservative Galerkin (LCG) method—a discontinuous methodology applied to a continuous framework*, Adv. Appl. Math. Mech., 1(3) (2009), pp. 319–340.
- [16] O. C. ZIENKIEWICZ AND R. CODINA, *A general algorithm for compressible and incompressible flow-part I: The split, characteristic-based scheme*, Int. J. Numer. Methods Fluids, 20 (1995), pp. 869–885.
- [17] O. C. ZIENKIEWICZ, *The Finite Element Method for Fluid Mechanics*, Word Book Inc., (2015).
- [18] J. CHEN, Z. Q. ZHANG AND H. XU ET AL., *A cell-based smoothed finite element method with semi-implicit CBS procedures for incompressible laminar viscous flows*, Int. J. Numer. Methods Fluids, 86 (2018), pp. 20–45.
- [19] P. NITHIARASU AND O. C. ZIENKIEWICZ, *On stabilization of the CBS algorithm, Internal and external time steps*, Int. J. Numer. Meth. Eng., 48 (2000), pp. 875–880.
- [20] C. G. THOMAS AND P. NITHIARASU, *Effect of variable smoothing and streamline direction on the viscous compressible flow calculations*, Int. J. Numer. Method. H., 15 (2005), pp. 420–428.
- [21] T. ZHANG, Z. Y. SI AND Y. N. HE, *A stabilized characteristic finite element method for transient*

- Navier-Stokes equations*, Int. J. Comput. Fluid D., 24(9) (2010), pp. 369–381.
- [22] M. MORANDI-CECCHI AND M. VENTURIN, *Characteristic-based split (CBS) algorithm finite element modelling for shallow waters in the Venice lagoon*, Int. J. Numer. Meth. Eng., 66(10) (2010), pp. 1641–1657.
- [23] C. B. LIU AND P. NITHIARASU, *The characteristic-based split (CBS) scheme for viscous elastic flow past a circular cylinder*, Int. J. Numer. Methods Fluids, 57 (2008), pp. 157–176.
- [24] J. TU, W. SUN AND D. ZHOU, *Flow characteristics and dynamic responses of a rear circular cylinder behind the square cylinder with different side lengths*, J. Vibroeng., 19 (2017), pp. 2956–2975.
- [25] G. CORTELLESA AND F. ARPINO, *Two-phase explicit CBS procedure for compressible viscous flow transport in porous materials*, Int. J. Numer. Method. H., 28 (2018), pp. 336–360.
- [26] P. NITHIARASU AND C. B. LIU, *An artificial compressibility based characteristic based split (CBS) scheme for steady and unsteady turbulent incompressible flows*, Comput. Method. Appl. M., 195 (2006), pp. 2961–2982.
- [27] T. HE, K. ZHANG AND T. WANG, *AC-CBS-based partitioned semi-implicit coupling algorithm for fluid-structure interaction using stabilized second-order pressure scheme*, Commun. Comput. Phys., 21 (2017), pp. 1449–1474.
- [28] Y. BAO AND D. ZHOU, *A two-step Taylor-characteristic-based Galerkin method for incompressible flows and its application to flow*, Int. J. Numer. Methods Fluids, 62 (2010), pp. 1181–1208.
- [29] V. ENJILELA AND A. AREFMANESH, *A two-step Taylor-characteristic MLPG method for fluid flow and heat transfer applications*, Eng. Anal. Bound. Elem., 51 (2015), pp. 174–190.
- [30] J. CHEN, Z. Q. ZHANG AND H. XU ET AL., *A cell-based smoothed finite element method with semi-implicit CBS procedures for incompressible laminar viscous flows*, Int. J. Numer. Methods Fluids, 86 (2018), pp. 20–45.
- [31] D. G. WANG, H. J. WANG AND J. H. XIONG, *A general algorithm for compressible and incompressible-flow, I: The split characteristic-based scheme*, Sci. China, 54(8) (2011), pp. 2157–2166.
- [32] Q. X. SHUI, D. G. WANG AND L. S. SHEN, *Improved characteristic-based operator-splitting finite element for Navier-Stokes equations*, J. S. Mech., 20(4) (2016), pp. 381–392.
- [33] K. KANAGARAJAN AND S. INDRAKUMAR, *Numerical solution of Nth-order fuzzy differential equation by Runge-Kutta method of order five*, Int. J. Math. Anal., 6 (2012), pp. 2885–2896.
- [34] O. C. ZIENKIEWICZ, *The Finite Element Method for Solid Mechanics*, Word Book Inc., (2009).
- [35] C. BI, *Finite Element Method of Computational Fluid Dynamics and Its Programming*, China Machine Press, (2016).
- [36] E. ERTURK, T. C. CORKE AND C. GOKCAL, *Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers*, Int. J. Numer. Methods Fluids, 48 (2005), pp. 747–774.
- [37] U. GHIA, K. N. GHIA AND C. T. SHIN, *High-Re solutions for incompressible flow using the Navier-Stokes equations and a multi grid method*, J. Comput. Phys., 48 (1982), pp. 387–411.