Advances in Aerodynamics

RESEARCH Open Access

A three-dimensional gas-kinetic flux solver for simulation of viscous flows with explicit formulations of conservative variables and numerical flux

Y. Sun[1], L. M. Yang[2*], C. Shu[2] and C. J. Teo[2]

* Correspondence: yangliming_2011@126.com
[2]Department of Mechanical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260, Singapore
Full list of author information is available at the end of the article

Abstract

A truly three-dimensional (3D) gas-kinetic flux solver for simulation of incompressible and compressible viscous flows is presented in this work. By local reconstruction of continuous Boltzmann equation, the inviscid and viscous fluxes across the cell interface are evaluated simultaneously in the solver. Different from conventional gas-kinetic scheme, in the present work, the distribution function at cell interface is computed in a straightforward way. As an extension of our previous work (Sun et al., Journal of Computational Physics, 300 (2015) 492–519), the non-equilibrium distribution function is calculated by the difference of equilibrium distribution functions between the cell interface and its surrounding points. As a result, the distribution function at cell interface can be simply calculated and the formulations for computing the conservative flow variables and fluxes can be given explicitly. To validate the proposed flux solver, several incompressible and compressible viscous flows are simulated. Numerical results show that the current scheme can provide accurate numerical results for three-dimensional incompressible and compressible viscous flows.

Keywords: 3D flux solver, Gas-kinetic scheme, Viscous flow, Navier-Stokes equations

1 Introduction

In the last few decades, the gas-kinetic scheme has been developed in both continuum [1–8] and rarefied flow regimes [9–12]. Unlike the traditional Riemann solver [13–15], the gas-kinetic scheme reconstructs the solution for the continuous Boltzmann equation at local cell interface. As the continuum assumption is avoided in the continuous Boltzmann equation, the gas-kinetic scheme can be applied in both continuum and rarefied flow problems, which is one of the advantages as compared with traditional Riemann solver. Another advantage is that the gas-kinetic scheme can compute the inviscid and viscous fluxes simultaneously from the solution of Boltzmann equation. In contrast, the traditional Riemann solver can only evaluate the inviscid flux and an additional step is required to compute the viscous flux by smooth function approximation.

Currently, there are two common types of gas-kinetic schemes: the kinetic flux vector scheme (KFVS) and gas-kinetic Bhatnagar-Gross-Krook (BGK) scheme. In KFVS, the Boltzmann equation without collision term, which is also called the collisionless Boltzmann equation, is solved. Basically, there are two stages in KFVS: free transport and collision. In the free transport stage, the collisionless Boltzmann equation is solved to calculate the flux at the interface. In the collision stage, the artificial collisions are added in the calculation of initial Maxwellian distribution at the beginning of next time step. The KFVS has been demonstrated to have good positivity property for simulation of flows with strong shock waves [2]. However, owing to the fact that the numerical dissipation in KFVS is proportional to the mesh size, the KFVS usually gives more diffusive results than the Godunov or flux difference splitting (FDS) scheme [16], and is not able to give accurate Navier-Stokes solutions except for cases in which the physical viscosity is much larger than the numerical viscosity. Some of representative researches on KFVS include Pullin [17], Deshpande [18], Perthame [19], Mandal and Deshpande [20], and Chou and Baganoff [21].

One of the significant developments in gas-kinetic schemes is the gas-kinetic BGK scheme, which was firstly proposed by Prendergast and Xu [22], and further developed by Chae et al. [23], Xu [24] and other researchers. In this method, the BGK collision model is adopted in the solution process to obtain the numerical fluxes across the interface. As a consequence, the dissipation in the transport can be controlled by a real collision time, which is proportional to the dynamic viscosity. The gas-kinetic BGK scheme enjoys some intrinsic advantages. Firstly, it has been shown that the gas-kinetic BGK scheme is able to generate a stable and crisp shock transition in the discontinuous region with a delicate dissipative mechanism [24]. At the same time, an accurate Navier-Stokes solution can be obtained in the smooth region. What is more, it is demonstrated that the entropy condition is always satisfied in the gas-kinetic BGK scheme and the "carbuncle phenomenon" is avoided for hypersonic flow simulations [25]. However, the gas-kinetic BGK scheme is not completely free from drawbacks. It is usually more complicated and inefficient than conventional computational fluid dynamics (CFD) schemes. This is because in the gas-kinetic BGK scheme, a number of coefficients related to the physical space should be calculated to evaluate the distribution function at each interface and each time step. Moreover, to the best of our knowledge, there is still no work of the gas-kinetic BGK scheme which can give explicit formulations for evaluating the conservative variables and numerical fluxes.

Recently, a straightforward way to evaluate the distribution function was proposed by Sun et al. [1], which is named as gas-kinetic flux solver (GKFS). Different from the gas-kinetic BGK scheme [24], the non-equilibrium distribution function at cell interface is approximated by the difference of equilibrium distribution functions between the cell interface and its surrounding points in GKFS. To be specific, the equilibrium distribution functions at the surrounding points of the cell interface are firstly given by interpolation from the conservative variables at cell centers. Then, the equilibrium distribution function at cell interface can be evaluated by a streaming process from the surrounding points. After the above steps, the non-equilibrium distribution function at cell interface can be simply calculated and the explicit formulations for computing the conservative flow variables and fluxes can be derived. It has been proven that GKFS can give the same results and only requires about 60% of the computational time as compared with the conventional gas-kinetic BGK scheme [1]. Inspired by the previous work of GKFS [1], a 3D GKFS is developed in this work. In the scheme, the 3D Navier-Stokes equations are discretized by the finite volume method and the numerical flux across the interface is evaluated by the local solution of 3D Boltzmann equation. Therefore, the present scheme can be viewed as a truly 3D flux solver. At the same time, a coordinate transformation is made at the local cell interface to transform the velocities in the Cartesian coordinate system to the normal and tangential directions of interface. In this way, all the cell interfaces can be treated using the same way for evaluation of conservative variables and numerical fluxes. Like the two-dimensional (2D) case, the non-equilibrium distribution function is approximated by the difference of the equilibrium distribution functions between the cell interface and its surrounding points. It is indicated that the present work is the first time to give explicit formulations for evaluating the conservative variables and numerical fluxes for the 3D viscous flow problems. Like other gas-kinetic schemes, the present scheme can be applied to both incompressible and compressible viscous flow problems without any modification. To validate the developed scheme, both incompressible and compressible viscous test cases are solved, including 3D driven cavity flow, incompressible flow past a stationary sphere, flow around ONERA M6 wing and DLR-F6 wing-body configuration. Numerical results show that the present solver can provide accurate results for both incompressible and compressible flows.

2 Boltzmann equation, Maxwellian distribution function and Navier-Stokes equations

2.1 Boltzmann equation and conservative forms of moments for Maxwellian function

With Bhatnagar-Gross-Krook (BGK) collision model [26], the continuous Boltzmann equation in the three-dimensional Cartesian coordinate system can be written as

$$\frac{\partial f}{\partial t}+u\frac{\partial f}{\partial x}+v\frac{\partial f}{\partial y}+w\frac{\partial f}{\partial z}=\frac{g-f}{\tau}, \tag{1}$$

where f is the real particle distribution function and g is the equilibrium particle distribution function. τ is the collision time, which is determined by the dynamic viscosity and the pressure. The right-hand side of the equation is the collision term which alters the distribution function from f to g within a collision time τ. Both f and g are functions of space (x, y, z), time (t) and particle velocity (u, v, w, ξ). The internal degree of freedom K in ξ is determined by the space dimension and the ratio of specific heat with the relation $K + D = 2/(\gamma - 1)$, where D is the abbreviation of the dimension ($D = 3$ in three dimension) and γ is the specific heat ratio. The equilibrium state g of the Maxwellian distribution is

$$g=\rho\left(\frac{\lambda}{\pi}\right)^{\frac{K+3}{2}}e^{-\lambda\left((u-U)^2+(v-V)^2+(w-W)^2+\xi^2\right)}, \tag{2}$$

where ρ is the density of the mean flow; $\mathbf{U} = (U, V, W)$ is the macroscopic velocity vector expressed in the x-, y- and z- directions; $\lambda = m/(2kT) = 1/(2RT)$, where m is the molecular mass, k is the Boltzmann constant, R is the gas constant and T is the temperature. In the equilibrium state, ξ^2 is the abbreviation of $\xi^2=\xi_1^2+\xi_2^2+\cdots+\xi_K^2$.

With the Maxwellian distribution function in Eq. (2), the following 7 conservative forms of moments will be satisfied, which are used to recover Navier-Stokes equations by Eq. (1) through Chapman-Enskog expansion analysis:

$$\int g d\Xi = \rho, \tag{3}$$

$$\int g u_\alpha d\Xi = \rho U_\alpha, \tag{4}$$

$$\int g\left(u_\alpha u_\alpha + \xi^2\right) d\Xi = \rho(U_\alpha U_\alpha + bRT), \tag{5}$$

$$\int g u_\alpha u_\beta d\Xi = \rho U_\alpha U_\beta + p\delta_{\alpha\beta}, \tag{6}$$

$$\int g\left(u_\alpha u_\alpha + \xi^2\right) u_\beta d\Xi = \rho[U_\alpha U_\alpha + (b+2)RT]U_\beta, \tag{7}$$

$$\int g u_\alpha u_\beta u_\chi d\Xi = p\left(U_\alpha \delta_{\beta\chi} + U_\beta \delta_{\chi\alpha} + U_\chi \delta_{\alpha\beta}\right) + \rho U_\alpha U_\beta U_\chi, \tag{8}$$

$$\begin{aligned}&\int g\left(u_\alpha u_\alpha + \xi^2\right) u_\beta u_\chi d\Xi \\ &= \rho\left\{U_\alpha U_\alpha U_\beta U_\chi + \left[(b+4)U_\beta U_\chi + U_\alpha U_\alpha \delta_{\beta\chi}\right]RT + (b+2)R^2T^2\delta_{\beta\chi}\right\},\end{aligned} \tag{9}$$

where u_α, u_β, u_χ and U_α, U_β, U_χ are the particle velocities and the macroscopic flow velocities in the α-, β- and χ- directions. p is the pressure and $b = K + D$ represents the total degree of freedoms of molecules. $d\Xi = du_\alpha du_\beta du_\chi d\xi_1 d\xi_2 \cdots d\xi_K$ is the volume element in the particle velocity space. The integral domain for u_α, u_β, u_χ, ξ_1, ξ_2, …, ξ_K is from $-\infty$ to $+\infty$. Eqs. (3)–(5) are applied to recover the fluid density, momentum and energy, respectively. Eqs. (6) and (7) are used to recover convective fluxes of the momentum equation and the energy equation. Eqs. (8) and (9) are to recover diffusive fluxes of the momentum equation and the energy equation.

2.2 Macroscopic governing equations discretized by finite volume method

In this work, the 3D Navier-Stokes equations are solved using the finite volume discretization with the conservative variables defined at cell centers, which can be written as

$$\frac{d\mathbf{W}}{dt} + \frac{1}{\Omega}\sum_{i=1}^{N} \mathbf{F}_i S_i = 0, \tag{10}$$

where $\mathbf{W}$ is the vector of conservative variables, Ω and N are the volume and number of interfaces of the control volume, respectively, $\mathbf{F}_i$ and S_i are the flux vector and the area of interface i. It should be noted that the numerical fluxes $\mathbf{F}_i$ are reconstructed locally at cell interface from the conservative variables $\mathbf{W}$ at cell centers. In the gas-kinetic scheme, the connection between the distribution function f and the conservative variables is

$$\mathbf{W} = (\rho,\ \rho U,\ \rho V,\ \rho W,\ \rho E)^T = \int \boldsymbol{\varphi}_\alpha f d\Xi, \tag{11}$$

where $E = \frac{1}{2}\left(U^2 + V^2 + W^2 + bRT\right)$. $\boldsymbol{\varphi}_\alpha$ is the moment given by

$$\boldsymbol{\phi}_\alpha = \left(1,\ \ u,\ \ v,\ \ w,\ \ \frac{1}{2}\left(u^2+v^2+w^2+\xi^2\right)\right)^T. \tag{12}$$

With the compatibility condition,

$$\int \boldsymbol{\phi}_\alpha \frac{g-f}{\tau} d\Xi = 0, \tag{13}$$

Eq. (11) is equivalent to

$$\mathbf{W} = (\rho,\ \ \rho U,\ \ \rho V,\ \ \rho W,\ \ \rho E)^T = \int \boldsymbol{\phi}_\alpha g d\Xi. \tag{14}$$

The above equation shows that the non-equilibrium distribution function has no contribution to the calculation of conservative variables.

After evaluation of conservative variables, the flux vector **F** can also be obtained from the distribution function

$$\mathbf{F} = \int u \boldsymbol{\phi}_\alpha f d\Xi. \tag{15}$$

It should be noted that Eq. (15) is the flux vector of *x*-direction in the Cartesian coordinate system. In the practical application such as curved boundary problems, we need to calculate the numerical flux in the normal direction of interface $\mathbf{F}_n$

$$\mathbf{F}_n = (F_1,\ \ F_2,\ \ F_3,\ \ F_4,\ \ F_5)^T = \int u^{'} \boldsymbol{\phi}_\alpha f d\Xi, \tag{16}$$

where $u^{'}$ is the particle velocity in the normal direction of interface. Suppose that $\mathbf{n}_1 = (n_{1x},\ \ n_{1y},\ \ n_{1z})$ is the unit vector in the normal direction of interface and $\mathbf{n}_2 = (n_{2x},\ \ n_{2y},\ \ n_{2z})$, $\mathbf{n}_3 = (n_{3x},\ \ n_{3y},\ \ n_{3z})$ are the unit vectors in the tangential directions. Then, the relationship between the particle velocities in the normal and tangential directions $(u^{'},\ \ v^{'}, w^{'})$ and the particle velocities in the Cartesian coordinate system $(u,\ \ v,\ \ w)$ are

$$\begin{aligned} u^{'} &= un_{1x}+vn_{1y}+wn_{1z}, \qquad v^{'} = un_{2x}+vn_{2y}+wn_{2z}, \qquad w^{'} \\ &= un_{3x}+vn_{3y}+wn_{3z}, \end{aligned} \tag{17}$$

and similarly

$$\begin{aligned} u &= u^{'}n_{1x}+v^{'}n_{2x}+w^{'}n_{3x}, \qquad v = u^{'}n_{1y}+v^{'}n_{2y}+w^{'}n_{3y}, \qquad w \\ &= u^{'}n_{1z}+v^{'}n_{2z}+w^{'}n_{3z}. \end{aligned} \tag{18}$$

Substituting Eq. (18) into Eq. (12), we have

$$\boldsymbol{\phi}_\alpha = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & n_{1x} & n_{2x} & n_{3x} & 0 \\ 0 & n_{1y} & n_{2y} & n_{3y} & 0 \\ 0 & n_{1z} & n_{2z} & n_{3z} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \left(1,\ \ u^{'},\ \ v^{'},\ \ w^{'},\ \ \frac{1}{2}\left(u^{'2}+v^{'2}+w^{'2}+\xi^2\right)\right)^T. \tag{19}$$

With the definition of a new moment

$$\boldsymbol{\varphi}_{\alpha}^{*}=\left(1,\ u^{'},\ v^{'},\ w^{'},\ \frac{1}{2}\left(u^{'2}+v^{'2}+w^{'2}+\xi^{2}\right)\right)^{T}, \tag{20}$$

and its corresponding flux vector

$$\mathbf{F}_{n}^{*}=\left(F_{1}^{*},\ F_{2}^{*},\ F_{3}^{*},\ F_{4}^{*},\ F_{5}^{*}\right)^{T}=\int u^{'}\boldsymbol{\varphi}_{\alpha}^{*}fd\Xi, \tag{21}$$

the real flux vector $\mathbf{F}_n$ can be obtained by substituting Eq. (19) into Eq. (16) and using Eq. (21)

$$\mathbf{F}_{n}=\int u^{'}\boldsymbol{\varphi}_{\alpha}fd\Xi=\begin{pmatrix}1 & 0 & 0 & 0 & 0\\ 0 & n_{1x} & n_{2x} & n_{3x} & 0\\ 0 & n_{1y} & n_{2y} & n_{3y} & 0\\ 0 & n_{1z} & n_{2z} & n_{3z} & 0\\ 0 & 0 & 0 & 0 & 1\end{pmatrix}\mathbf{F}_{n}^{*}. \tag{22}$$

The above Eq. (22) shows that the calculation of $\mathbf{F}_n$ is equivalent to the evaluation of $\mathbf{F}_n^*$ and the key issue is to obtain the gas distribution function f. In the next subsection, a 3D GKFS will be introduced to evaluate the gas distribution function f at cell interface.

3 Three-dimensional gas-kinetic flux solver

As the flux vector $\mathbf{F}_n^*$ is evaluated at the local interface, a local coordinate system is applied in the derivation of distribution function f. It is known that the distribution function f can be separated into two parts, the equilibrium part f^{eq} and the non-equilibrium part f^{neq} with the relationship of

$$f=f^{eq}+f^{neq}. \tag{23}$$

Here, the equilibrium part f^{eq} equals to

$$f^{eq}=g. \tag{24}$$

With the Chapman-Enskog expansion analysis, the non-equilibrium distribution function can be approximated as

$$f^{neq}=-\tau\left(\frac{\partial}{\partial t}+\mathbf{u}\cdot\nabla\right)f^{eq}=-\tau\left(\frac{\partial}{\partial t}+\mathbf{u}\cdot\nabla\right)g. \tag{25}$$

Therefore, the gas distribution function truncated to the Navier-Stokes level becomes

$$f=f^{eq}+f^{neq}=g-\tau\left(\frac{\partial g}{\partial t}+u^{'}\frac{\partial g}{\partial n_{1}}+v^{'}\frac{\partial g}{\partial n_{2}}+w^{'}\frac{\partial g}{\partial n_{3}}\right). \tag{26}$$

By applying the Taylor series expansion in time and physical space, the above equation can be simplified to

$$\begin{aligned}&f(0,0,0,t+\delta t)\\&=g(0,0,0,t+\delta t)-\frac{\tau}{\delta t}\left[g(0,0,0,t+\delta t)-g\left(-u^{'}\delta t,-v^{'}\delta t,-w^{'}\delta t,t\right)\right],\end{aligned} \tag{27}$$

where $f(0,0,0,t+\delta t)$ is the gas distribution function at local interface; $g(0,0,0,t+\delta t)$ and $g(-u^{'}\delta t,-v^{'}\delta t,-w^{'}\delta t,t)$ are the equilibrium distribution functions at local interface and its surrounding points, respectively. δt is the streaming time step. From Eq. (27), it can be seen that the non-equilibrium distribution f^{neq} is calculated by the difference of equilibrium distribution functions between the interface and its surrounding points, which makes current GKFS be much more straightforward.

In the present work, the conservative variables in Eq. (10) are defined at cell centers. In order to solve Eq. (10) by marching in time, the numerical flux in the normal direction of each cell interface $\mathbf{F}_n^*$ should be evaluated first. Suppose that the conservative variables at cell centers and their first order derivatives are already known, the conservative variables at the left and the right sides of an interface can be easily given by interpolation. Then, the equilibrium distribution functions at these two sides of interface can be given via Eq. (2). After that, the second order approximation of $g(-u^{'}\delta t, -v^{'}\delta t, -w^{'}\delta t, t)$ at the time level t can be written as

$$g\left(-u^{'}\delta t, -v^{'}\delta t, -w^{'}\delta t, t\right) = \begin{cases} g_l - \dfrac{\partial g_l}{\partial n_1}u^{'}\delta t - \dfrac{\partial g_l}{\partial n_2}v^{'}\delta t - \dfrac{\partial g_l}{\partial n_3}w^{'}\delta t, & u^{'} \geq 0, \\ g_r - \dfrac{\partial g_r}{\partial n_1}u^{'}\delta t - \dfrac{\partial g_r}{\partial n_2}v^{'}\delta t - \dfrac{\partial g_r}{\partial n_3}w^{'}\delta t, & u^{'} < 0. \end{cases} \tag{28}$$

Where g_l and g_r are the equilibrium distribution functions at the left and the right sides of interface, respectively. Note that in Eq. (28), the equilibrium distribution functions at two sides of interface are not necessarily the same, which means that a possible discontinuity has been taken into account in the form. By substituting Eq. (28) into Eq. (27), we have

$$\begin{aligned} f(0,0,0,t+\delta t) &= g(0,0,0,t+\delta t) \\ &- \frac{\tau}{\delta t}\left[g(0,0,0,t+\delta t) - g_l H\left(u^{'}\right) - g_r\left(1-H\left(u^{'}\right)\right)\right] \\ &- \tau\left[\left(\frac{\partial u^{'}g_l}{\partial n_1} + \frac{\partial v^{'}g_l}{\partial n_2} + \frac{\partial w^{'}g_l}{\partial n_3}\right)H\left(u^{'}\right) + \left(\frac{\partial u^{'}g_r}{\partial n_1} + \frac{\partial v^{'}g_r}{\partial n_2} + \frac{\partial w^{'}g_r}{\partial n_3}\right)\left(1-H\left(u^{'}\right)\right)\right], \end{aligned} \tag{29}$$

where $H(u^{'})$ is the Heaviside function defined as

$$H\left(u^{'}\right) = \begin{cases} 0, & u^{'} < 0, \\ 1, & u^{'} \geq 0. \end{cases}$$

Equation (29) shows that the full information of distribution function at the interface can be decided once we have the equilibrium distribution function at cell interface and its surrounding points.

3.1 Evaluation of conservative variables $\mathbf{W}^*$ at cell interface

It is known that the non-equilibrium distribution has no influence on the computation of conservative variables, and thus Eq. (14) can be adopted to calculate the conservative variables $\mathbf{W}^*$ at local interface

$$\mathbf{W}^* = \left(\rho,\ \rho U^{'},\ \rho V^{'},\ \rho W^{'},\ \rho E\right)^T = \int g\boldsymbol{\Phi}_\alpha^* d\Xi. \tag{30}$$

According to the compatibility condition (see Eq. (13)), by substituting Eq. (27) and Eq. (28) into Eq. (30), we have

$$
\begin{aligned}
\mathbf{W}^{*} &= \int \boldsymbol{\Phi}_{\alpha}^{*} g(0,0,0,t+\delta t)d\Xi = \int \boldsymbol{\Phi}_{\alpha}^{*} g\left(-u^{'}\delta t, -v^{'}\delta t, -w^{'}\delta t, t\right)d\Xi \\
&= \int\int_{u^{'}>0} \boldsymbol{\Phi}_{\alpha}^{*}\left(g_{l}-\frac{\partial g_{l}}{\partial n_{1}}u^{'}\delta t-\frac{\partial g_{l}}{\partial n_{2}}v^{'}\delta t-\frac{\partial g_{l}}{\partial n_{3}}w^{'}\delta t\right)d\Xi \\
&+\int\int_{u^{'}<0} \boldsymbol{\Phi}_{\alpha}^{*}\left(g_{r}-\frac{\partial g_{r}}{\partial n_{1}}u^{'}\delta t-\frac{\partial g_{r}}{\partial n_{2}}v^{'}\delta t-\frac{\partial g_{r}}{\partial n_{3}}w^{'}\delta t\right)d\Xi.
\end{aligned}
\tag{31}
$$

The above equation shows that the conservative variables at cell interface can be obtained by equilibrium distribution function of the surrounding points. By taking the limit $\delta t \rightarrow 0$ [24], the conservative variables at cell interface can be calculated by

$$
\mathbf{W}^{*} = \int\int_{u^{'}>0} \boldsymbol{\Phi}_{\alpha}^{*} g_{l} d\Xi + \int\int_{u^{'}<0} \boldsymbol{\Phi}_{\alpha}^{*} g_{r} d\Xi. \tag{32}
$$

The above equation means that the conservative variables at cell interface are simply computed by the reconstructed variables of left and right sides. With parameters defined in the Appendix, the conservative variables $\mathbf{W}^{*}$ at cell interface are given by

$$
\rho = \left(\rho_{l} a_{l} + \rho_{r} a_{r}\right), \tag{33}
$$

$$
\rho U^{'} = \left(\rho_{l} b_{l} + \rho_{r} b_{r}\right), \tag{34}
$$

$$
\rho V^{'} = \left(\rho_{l} V_{l}^{'} a_{l} + \rho_{r} V_{r}^{'} a_{r}\right), \tag{35}
$$

$$
\rho W^{'} = \left(\rho_{l} W_{l}^{'} a_{l} + \rho_{r} W_{r}^{'} a_{r}\right), \tag{36}
$$

$$
\begin{aligned}
\rho E = &\frac{1}{2}\rho_{l}\left[c_{l} + \left(V_{l}^{'2} + W_{l}^{'2} + (b-1)\ RT_{l}\right) a_{l}\right] \\
&+\frac{1}{2}\rho_{r}\left[c_{r} + \left(V_{r}^{'2} + W_{r}^{'2} + (b-1)\ RT_{r}\right) a_{r}\right],
\end{aligned}
\tag{37}
$$

where "$\cdot_{l}$" and "$\cdot_{r}$" ("$\cdot$" stands for any variable) denote the variables at the left and the right sides of interface, respectively.

3.2 Evaluation of numerical flux $\mathbf{F}_{n}^{*}$ at cell interface

As soon as the conservative variables at local interface $\mathbf{W}^{*}$ are obtained, the equilibrium distribution function $g(0,0,0,t+\delta t)$ can be known by Eq. (2). Then the numerical flux across the cell interface can be calculated via Eq. (29)

$$
\begin{aligned}
\mathbf{F}_{n}^{*} &= \int u^{'} \boldsymbol{\Phi}_{\alpha}^{*} f(0,0,0,t+\delta t)d\Xi \\
&= \int u^{'} \boldsymbol{\Phi}_{\alpha}^{*} g(0,0,0,t+\delta t)d\Xi - \frac{\tau}{\delta t}\left[\int u^{'} \boldsymbol{\Phi}_{\alpha}^{*} g(0,0,0,t+\delta t)d\Xi\right. \\
&- \int\int_{u^{'}>0} u^{'} \boldsymbol{\Phi}_{\alpha}^{*} g_{l} d\Xi - \left.\int\int_{u^{'}<0} u^{'} \boldsymbol{\Phi}_{\alpha}^{*} g_{r} d\Xi\right] \\
&-\tau\left[\frac{\partial}{\partial n_{1}}\int\left(\int_{u^{'}>0} u^{'2} \boldsymbol{\Phi}_{\alpha}^{*} g_{l} + \int_{u^{'}<0} u^{'2} \boldsymbol{\Phi}_{\alpha}^{*} g_{r}\right)d\Xi\right. \\
&+\frac{\partial}{\partial n_{2}}\int\left(\int_{u^{'}>0} u^{'} v^{'} \boldsymbol{\Phi}_{\alpha}^{*} g_{l} + \int_{u<0} u^{'} v^{'} \boldsymbol{\Phi}_{\alpha}^{*} g_{r}\right)d\Xi \\
&+\left.\frac{\partial}{\partial n_{3}}\int\left(\int_{u^{'}>0} u^{'} w^{'} \boldsymbol{\Phi}_{\alpha}^{*} g_{l} + \int_{u<0} u^{'} w^{'} \boldsymbol{\Phi}_{\alpha}^{*} g_{r}\right)d\Xi\right].
\end{aligned}
\tag{38}
$$

Note that $g(0,0,0,t+\delta t)$ is the equilibrium distribution function at the interface and time level $t+\delta t$, and g_l, g_r are the distribution functions at the left and the right sides of interface and the time level t. By taking the limit $\delta t \to 0$, we have

$$\begin{aligned}
&-\frac{\tau}{\delta t}\left[\int u^{'}\boldsymbol{\Phi}_{\alpha}^{*}g(0,0,0,t+\delta t)d\Xi - \int\int_{u^{'}>0} u^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_{l}d\Xi - \int\int_{u^{'}<0} u^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_{r}d\Xi\right] \\
&= -\tau\int u^{'}\boldsymbol{\Phi}_{\alpha}^{*}\frac{\partial g(0,0,0,t)}{\partial t}d\Xi.
\end{aligned} \tag{39}$$

According to the work of Xu [24], $\partial g/\partial t$ can be expanded by

$$\frac{\partial g(0,0,0,t)}{\partial t} = g(0,0,0,t)\left(A_1 + A_2u^{'} + A_3v^{'} + A_4w^{'} + A_5\varepsilon\right), \tag{40}$$

where A_1, A_2, A_3, A_4 and A_5 are the derivatives of macroscopic variables with respect to physical space, which will be determined from the compatibility condition, $\varepsilon = \frac{1}{2}(u^{'2} + v^{'2} + w^{'2} + \xi^2)$. Thus, the flux expression in Eq. (38) can be written as

$$\begin{aligned}
\mathbf{F}_n^{*} = &\int u^{'}\boldsymbol{\Phi}_{\alpha}^{*}g(0,0,0,t)d\Xi \\
&-\tau\int u^{'}\boldsymbol{\Phi}_{\alpha}^{*}g(0,0,0,t)\left(A_1 + A_2u^{'} + A_3v^{'} + A_4w^{'} + A_5\varepsilon\right)d\Xi \\
&-\tau\left[\frac{\partial}{\partial n_1}\int\left(\int_{u^{'}>0} u^{'2}\boldsymbol{\Phi}_{\alpha}^{*}g_l + \int_{u^{'}<0} u^{'2}\boldsymbol{\Phi}_{\alpha}^{*}g_r\right)d\Xi\right. \\
&\quad + \frac{\partial}{\partial n_2}\int\left(\int_{u^{'}>0} u^{'}v^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_l + \int_{u<0} u^{'}v^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_r\right)d\Xi \\
&\quad \left. + \frac{\partial}{\partial n_3}\int\left(\int_{u^{'}>0} u^{'}w^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_l + \int_{u<0} u^{'}w^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_r\right)d\Xi\right].
\end{aligned} \tag{41}$$

Note that $\delta t \to 0$ has been applied in Eq. (41). In the above equation, the only undetermined variables are the coefficients A_1, A_2, A_3, A_4 and A_5.

Substituting Eq. (29) into Eq. (11) and adopting the compatibility condition, we have

$$\begin{aligned}
&\frac{1}{\delta t}\left[\int \boldsymbol{\Phi}_{\alpha}^{*}g(0,0,0,t+\delta t)d\Xi - \int\int_{u^{'}>0}\boldsymbol{\Phi}_{\alpha}^{*}g_l d\Xi - \int\int_{u^{'}<0}\boldsymbol{\Phi}_{\alpha}^{*}g_r d\Xi\right] \\
&= -\left[\int\int_{u^{'}>0}\boldsymbol{\Phi}_{\alpha}^{*}\left(\frac{\partial g_l}{\partial n_1}u^{'} + \frac{\partial g_l}{\partial n_2}v^{'} + \frac{\partial g_l}{\partial n_3}w^{'}\right)d\Xi\right. \\
&\quad \left. + \int\int_{u^{'}<0}\boldsymbol{\Phi}_{\alpha}^{*}\left(\frac{\partial g_r}{\partial n_1}u^{'} + \frac{\partial g_r}{\partial n_2}v^{'} + \frac{\partial g_r}{\partial n_3}w^{'}\right)d\Xi\right].
\end{aligned} \tag{42}$$

Using Eqs. (39)–(40), the above equation can be written as

$$\begin{aligned}
&\int \boldsymbol{\Phi}_{\alpha}^{*}g(0,0,0,t)\left(A_1 + A_2u^{'} + A_3v^{'} + A_4w^{'} + A_5\varepsilon\right)d\Xi \\
&= -\left[\frac{\partial}{\partial n_1}\int\left(\int_{u^{'}>0} u^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_l + \int_{u^{'}<0} u^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_r\right)d\Xi + \frac{\partial}{\partial n_2}\int\left(\int_{u^{'}>0} v^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_l + \int_{u^{'}<0} v^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_r\right)d\Xi\right. \\
&\quad \left. + \frac{\partial}{\partial n_3}\int\left(\int_{u^{'}>0} w^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_l + \int_{u^{'}<0} w^{'}\boldsymbol{\Phi}_{\alpha}^{*}g_r\right)d\Xi\right].
\end{aligned} \tag{43}$$

Defining

$$\begin{aligned}&\frac{\partial}{\partial n_1}\int\left(\int_{u'>0}u'\boldsymbol{\phi}_\alpha^* g_l+\int_{u'<0}u'\boldsymbol{\phi}_\alpha^* g_r\right)d\Xi+\frac{\partial}{\partial n_2}\int\left(\int_{u'>0}v'\boldsymbol{\phi}_\alpha^* g_l+\int_{u'<0}v'\boldsymbol{\phi}_\alpha^* g_r\right)d\Xi\\&\qquad+\frac{\partial}{\partial n_3}\int\left(\int_{u'>0}w'\boldsymbol{\phi}_\alpha^* g_l+\int_{u'<0}w'\boldsymbol{\phi}_\alpha^* g_r\right)d\Xi\\&=\begin{pmatrix}G_1\\G_2\\G_3\\G_4\\G_5\end{pmatrix}.\end{aligned}\tag{44}$$

The explicit formulations of G_1 to G_5 are given in the Appendix. After a similar derivation process to the work of Xu [24], the coefficients A_1, A_2, A_3, A_4 and A_5 can be determined by

$$A_5=-\frac{8\lambda^2}{(K+3)\rho}\left[G_5-U'G_2-V'G_3-W'G_4-\left(\mathfrak{R}_1-U'^2-V'^2-W'^2\right)G_1\right],\tag{45}$$

$$A_4=-\frac{2\lambda}{\rho}\left(G_4-W'G_1\right)-W'A_5,\tag{46}$$

$$A_3=-\frac{2\lambda}{\rho}\left(G_3-V'G_1\right)-V'A_5,\tag{47}$$

$$A_2=-\frac{2\lambda}{\rho}\left(G_2-U'G_1\right)-U'A_5,\tag{48}$$

$$A_1=-\frac{1}{\rho}G_1-U'A_2-V'A_3-W'A_4-\mathfrak{R}_1A_5,\tag{49}$$

where

$$\mathfrak{R}_1=\frac{1}{2}\left(U'^2+V'^2+W'^2+\frac{K+3}{2\lambda}\right).\tag{50}$$

Once the above coefficients are obtained, the numerical flux $\mathbf{F}_n^*$ across the interface can be calculated via Eq. (41). Similar to the conservative variables $\mathbf{W}^*$, the explicit expressions for numerical flux $\mathbf{F}_n^*$ can also be given as

$$F_1^*=\rho U',\tag{51}$$

$$\begin{aligned}F_2^*=&\left(\rho U'^2+p\right)-\tau\rho\Big[A_1\left\langle u'^2\right\rangle+A_2\left\langle u'^3\right\rangle+A_3\left\langle u'^2\right\rangle\left\langle v'^1\right\rangle+A_4\left\langle u'^2\right\rangle\left\langle w'^1\right\rangle\\&+\frac{1}{2}A_5\left(\left\langle u'^4\right\rangle+\left\langle u'^2\right\rangle\left\langle v'^2\right\rangle+\left\langle u'^2\right\rangle\left\langle w'^2\right\rangle+\left\langle u'^2\right\rangle\left\langle\xi^2\right\rangle\right)\Big]\\&-\tau\left[\frac{\partial\left(\rho_l d_l+\rho_r d_r\right)}{\partial n_1}+\frac{\partial\left(\rho_l V'_l c_l+\rho_r V'_r c_r\right)}{\partial n_2}+\frac{\partial\left(\rho_l W'_l c_l+\rho_r W'_r c_r\right)}{\partial n_3}\right],\end{aligned}\tag{52}$$

$$\begin{aligned}F_3^*=&\rho U'V'-\tau\rho\Big[A_1\left\langle u'^1\right\rangle\left\langle v'^1\right\rangle+A_2\left\langle u'^2\right\rangle\left\langle v'^1\right\rangle+A_3\left\langle u'^1\right\rangle\left\langle v'^2\right\rangle+A_4\left\langle u'^1\right\rangle\left\langle v'^1\right\rangle\left\langle w'^1\right\rangle\\&+\frac{1}{2}A_5\left(\left\langle u'^3\right\rangle\left\langle v'^1\right\rangle+\left\langle u'^1\right\rangle\left\langle v'^3\right\rangle+\left\langle u'^1\right\rangle\left\langle v'^1\right\rangle\left\langle w'^2\right\rangle+\left\langle u'^1\right\rangle\left\langle v'^1\right\rangle\left\langle\xi^2\right\rangle\right)\Big]\\&-\tau\Bigg[\frac{\partial\left(\rho_l V'_l c_l+\rho_r V'_r c_r\right)}{\partial n_1}+\frac{\partial\left[\left(\rho_l V'^2_l+p_l\right)b_l+\left(\rho_r V'^2_r+p_r\right)b_r\right]}{\partial n_2}\\&\qquad+\frac{\partial\left(\rho_l V_l{}'W_l{}'b_l+\rho_r V_r{}'W_r{}'b_r\right)}{\partial n_3}\Bigg],\end{aligned}\tag{53}$$

$$
\begin{aligned}
F_4^* = {} & \rho U^{'} W^{'} - \tau\rho\Big[A_1\Big\langle u^{'1}\Big\rangle\Big\langle w^{'1}\Big\rangle + A_2\Big\langle u^{'2}\Big\rangle\Big\langle w^{'1}\Big\rangle + A_3\Big\langle u^{'1}\Big\rangle\Big\langle v^{'1}\Big\rangle\Big\langle w^{'1}\Big\rangle + A_4\Big\langle u^{'1}\Big\rangle\Big\langle w^{'2}\Big\rangle \\
& + \frac{1}{2}A_5\Big(\Big\langle u^{'3}\Big\rangle\Big\langle w^{'1}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle v^{'2}\Big\rangle\Big\langle w^{'1}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle w^{'3}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle w^{'1}\Big\rangle\langle \xi^2\rangle\Big)\Big] \\
& - \tau\left[\frac{\partial\left(\rho_l W_l^{'} c_l + \rho_r W_r^{'} c_r\right)}{\partial n_1} + \frac{\partial\left(\rho_l V_l^{'} W_l^{'} b_l + \rho_r V_r^{'} W_r^{'} b_r\right)}{\partial n_2}\right. \\
& \left. + \frac{\partial\left[\left(\rho_l W_l^{'2} + p_l\right)b_l + \left(\rho_r W_r^{'2} + p_r\right)b_r\right]}{\partial n_3}\right],
\end{aligned}
\tag{54}
$$

$$
\begin{aligned}
F_5^* = {} & (\rho E + p)U^{'} - \frac{1}{2}\tau\rho\Big\{A_1\Big[\Big\langle u^{'3}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle v^{'2}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle w^{'2}\Big\rangle + \Big\langle u^{'1}\Big\rangle\langle \xi^2\rangle\Big] \\
& + A_2\Big[\Big\langle u^{'4}\Big\rangle + \Big\langle u^{'2}\Big\rangle\Big\langle v^{'2}\Big\rangle + \Big\langle u^{'2}\Big\rangle\Big\langle w^{'2}\Big\rangle + \Big\langle u^{'2}\Big\rangle\langle \xi^2\rangle\Big] \\
& + A_3\Big[\Big\langle u^{'3}\Big\rangle\Big\langle v^{'1}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle v^{'3}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle v^{'1}\Big\rangle\Big\langle w^{'2}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle v^{'1}\Big\rangle\langle \xi^2\rangle\Big] \\
& + A_4\Big[\Big\langle u^{'3}\Big\rangle\Big\langle w^{'1}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle v^{'2}\Big\rangle\Big\langle w^{'1}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle w^{'3}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle w^{'1}\Big\rangle\langle \xi^2\rangle\Big] \\
& + \frac{1}{2}A_5\Big[\Big\langle u^{'5}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle v^{'4}\Big\rangle + \Big\langle u^{'1}\Big\rangle\Big\langle w^{'4}\Big\rangle + \Big\langle u^{'1}\Big\rangle\langle \xi^4\rangle + 2\Big\langle u^{'3}\Big\rangle\Big\langle v^{'2}\Big\rangle + 2\Big\langle u^{'3}\Big\rangle\Big\langle w^{'2}\Big\rangle \\
& + 2\Big\langle u^{'3}\Big\rangle\langle \xi^2\rangle + 2\Big\langle u^{'1}\Big\rangle\Big\langle v^{'2}\Big\rangle\Big\langle w^{'2}\Big\rangle + 2\Big\langle u^{'1}\Big\rangle\Big\langle v^{'2}\Big\rangle\langle \xi^2\rangle + 2\Big\langle u^{'1}\Big\rangle\Big\langle w^{'2}\Big\rangle\langle \xi^2\rangle\Big]\Big\} \\
& - \frac{1}{2}\tau\Big\{\frac{\partial}{\partial n_1}\Big\{\rho_l\Big[e_l + \Big(V_l^{'2} + W_l^{'2} + (b–1)RT_l\Big)c_l\Big] \\
& + \rho_r\Big[e_r + \Big(V_r^{'2} + W_r^{'2} + (b–1)RT_r\Big)c_r\Big]\Big\} \\
& + \frac{\partial}{\partial n_2}\Big\{\rho_l V_l^{'}\Big[d_l + \Big(V_l^{'2} + W_l^{'2} + (b+1)RT_l\Big)b_l\Big] \\
& + \rho_r V_r^{'}\Big[d_r + \Big(V_r^{'2} + W_r^{'2} + (b+1)RT_r\Big)b_r\Big]\Big\} \\
& + \frac{\partial}{\partial n_3}\Big\{\rho_l W_l^{'}\Big[d_l + \Big(V_l^{'2} + W_l^{'2} + (b+1)RT_l\Big)b_l\Big] \\
& + \rho_r W_r^{'}\Big[d_r + \Big(V_r^{'2} + W_r^{'2} + (b+1)RT_r\Big)b_r\Big]\Big\}\Big\}.
\end{aligned}
\tag{55}
$$

In Eqs. (51)–(55), the definitions of $\langle\cdot\rangle$ and all coefficients can be found in the Appendix. They are expressed explicitly as the functions of conservative variables and their derivatives. In addition, since the structured mesh is used in this work, the spatial derivatives in Eqs. (51)–(55) can be approximated directly by the finite difference scheme.

From the above derivations, it can be seen that there are two major differences between the present solver and the gas-kinetic BGK scheme [24]. The first difference is that the local asymptotic solution to the Boltzmann equation (see Eq. (26)) is used to calculate the distribution function on the cell interface for the GKFS, while the local integral solution to the Boltzmann equation is utilized for the gas-kinetic BGK scheme. Another difference is that the non-equilibrium distribution function is approximated by the difference of equilibrium distribution functions on the cell interface and its surrounding streaming nodes in GKFS (see Eq. (27)), while in the gas-kinetic BGK scheme, the non-equilibrium distribution function is included in the initial distribution function around the cell interface. These differences lead to the numerical flux reconstructed by the GKFS being time-independent (see Eq. (41)), while that of the gas-kinetic BGK scheme is time-dependent. Since $\delta t \to 0$ is adopted, the GKFS actually reconstructs the numerical flux at the time level t, as shown in Eq. (41). In the gas-kinetic BGK scheme, the numerical flux can be viewed as the integral average in the time interval $[t, t+\Delta t]$. From this point of view, the temporal accuracy of the flux in GKFS is $O(\Delta t)$, while that of the gas-kinetic BGK scheme is $O(\Delta t^2)$. But in fact, most of the conventional CFD schemes, such as the Roe scheme, HLL scheme and AUSM, also calculate the numerical flux at the time level t, which indicates that the temporal accuracy of the flux may not be important for solving the Euler/Navier-Stokes equations for general cases. In terms of simplicity, fewer coefficients are involved in GKFS than the gas-kinetic BGK scheme.

3.3 Determination of collision time τ

Theoretically, the collision time τ in Eq. (1) is proportional to the physical viscosity

$$\tau = \mu/p, \tag{56}$$

where μ is the dynamic viscosity and p is the pressure. However, the numerical dissipation in Eq. (56) might not be sufficient to get a stable solution in cases such as strong shock wave. Therefore, the effective viscosity should be a combination of both physical and numerical ones. Xu [24] proposed a simple and effective treatment to incorporate the numerical viscosity into the gas-kinetic BGK scheme, which is also adopted in the present work:

$$\tau = \frac{\mu}{p} + \frac{|p_l - p_r|}{p_l + p_r}\Delta t, \tag{57}$$

where Δt is the time step in the solution of Navier-Stokes equations, p_l and p_r are the pressure at the left and the right sides of interface, respectively. The additional term of the above equation corresponds to numerical viscosity, which is applied to take the pressure jump with a thickness in the order of cell size into account.

3.4 Prandtl number fix

It is well known that the Prandtl number in the gas-kinetic BGK scheme corresponds to unity [24]. Several approaches are available to make the Prandtl number be consistent with the real problem. BGK-Shakhov model [27] is one of these attempts, which adjusts the heat flux in the relaxation term. In the Shakhov model, the Shakhov equilibrium distribution function is given by

$$g^s = g\left[1 + (1 - \mathrm{Pr})\mathbf{c}\cdot\mathbf{q}\left(\frac{c^2}{RT} - 5\right)/(5pRT)\right], \tag{58}$$

where g is the Maxwellian distribution function in Eq. (2), Pr is the Prandtl number, $\mathbf{c} = \mathbf{u} - \mathbf{U}$ is the peculiar velocity and $\mathbf{q}$ is the heat flux

$$\mathbf{q} = \frac{1}{2}\int(\mathbf{u} - \mathbf{U})\left((u - U)^2 + (v - V)^2 + (w - W)^2 + \xi^2\right) f d\Xi. \tag{59}$$

It can be seen from Eq. (58) that the Prandtl number can be changed to any realistic value easily. However, considerable work has to be devoted to extend the current GKFS to the above Shakhov model.

Another alternative approach is to make correction for heat flux, which has been presented in [24].

$$F_5^{new} = F_5 + \left(\frac{1}{\mathrm{Pr}} - 1\right)\mathbf{q}\cdot\mathbf{n}_1, \tag{60}$$

where F_5 is the energy flux and $\mathbf{q}$ is the heat flux defined in Eq. (59). Since all momentums in Eq. (60) have been obtained in the evaluation of energy flux F_5, there will not be much additional work in the above Prandtl number fix. Therefore, Eq. (60) is employed to adjust the Prandtl number in the present work.

3.5 Computational sequence

In this section, the basic solution procedure of the current 3D GKFS is summarized as follows:

(1) Firstly, we need to calculate the derivatives of conservative variables and reconstruct the initial conservative variables at two sides of cell interface.
(2) Compute the unit vector in the normal direction $\mathbf{n}_1$ and in the tangential directions $\mathbf{n}_2$ and $\mathbf{n}_3$ of cell interface. Convert the velocities in the Cartesian coordinate system into the local coordinate system via Eq. (17).
(3) Calculate the conservative variables at cell interface $\mathbf{W}^*$ by using Eqs. (33)–(37).
(4) Calculate the vector $(G_1,\ G_2,\ G_3,\ G_4,\ G_5)^T$ by using Eqs. (A.19)-(A.23) and further compute coefficients $A_1,\ A_2,\ A_3,\ A_4,\ A_5$ by Eqs. (45)–(49).
(5) Calculate the numerical flux $\mathbf{F}_n^*$ by Eqs. (51)–(55).
(6) Compute the heat flux $\mathbf{q}$ via Eq. (59), and make correction for energy flux by using Eq. (60).
(7) Convert the numerical flux in the local coordinate system $\mathbf{F}_n^*$ to the global Cartesian coordinate system $\mathbf{F}_n$ by using Eq. (22).
(8) Once the fluxes at all cell interfaces are obtained, solve ordinary differential equation (Eq. (10)) by using time marching method. This step gives the conservative variables at cell centers at new time step.
(9) Repeat steps (1) to (8) until convergence criterion is satisfied.

4 Numerical results and discussion

To validate the proposed 3D GKFS for simulation of incompressible and compressible viscous flows, the 3D lid-driven cavity flow, incompressible flow past a stationary sphere, flow around ONERA M6 wing and DLR-F6 wing-body configuration are considered. For temporal discretization to Eq. (10), four-stages Runge-Kutta method is applied in cases of 3D lid-driven cavity flow and flow past a stationary sphere. In compressible cases, the Lower-upper symmetric-Gauss-Seidel (LU-SGS) scheme [28] is adopted to accelerate the convergence and the Venkatakrishnan's limiter [29] is used to calculate the conservative variables at two sides of interface $\mathbf{W}^L$ and $\mathbf{W}^R$ in the reconstruction process. Specifically, $\mathbf{W}^L$ and $\mathbf{W}^R$ are computed by

$$\begin{aligned}\mathbf{W}^L &= \mathbf{W}_c^L + \Psi_c^L\left(\mathbf{x}_b-\mathbf{x}_c^L\right)\cdot\nabla\mathbf{W}_c^L,\\ \mathbf{W}^R &= \mathbf{W}_c^R + \Psi_c^R\left(\mathbf{x}_b-\mathbf{x}_c^R\right)\cdot\nabla\mathbf{W}_c^R,\end{aligned} \tag{61}$$

where $\mathbf{W}_c^L$ and $\mathbf{W}_c^R$ are the conservative flow variables at centers of the left and the right cells, respectively; $\nabla\mathbf{W}_c^L$ and $\nabla\mathbf{W}_c^R$ are their corresponding first-order derivatives. $\mathbf{x}_c^L$, $\mathbf{x}_c^R$ and $\mathbf{x}_b$ are the coordinates of the left cell center, the right cell center and the midpoint of cell interface, respectively. Ψ_c^L and Ψ_c^R are the limiter functions utilized in the left and the right cells, respectively. In addition, all the simulations were done on a PC with 3.10GHz CPU.

Before applying the GKFS to various fluid flow problems, its accuracy is first validated by the advection of density perturbation for three-dimensional flows [30]. The initial condition of this problem is set as

$$\begin{aligned}&\rho(x,y,z) = 1 + 0.2\ \sin(\pi(x+y+z)),\\&u(x,y,z) = 1, v(x,y,z) = 1, w(x,y,z) = 1, p(x,y,z) = 1.\end{aligned} \tag{62}$$

The exact solutions under periodic boundary condition are

$$\begin{aligned}&\rho(x,y,z,t) = 1 + 0.2\ \sin(\pi(x+y+z-3t)),\\&u(x,y,z,t) = 1, v(x,y,z,t) = 1, w(x,y,z,t) = 1, p(x,y,z,t) = 1.\end{aligned} \tag{63}$$

Since this test case belongs to the inviscid flow, the collision time τ takes

$$\tau = \varepsilon\Delta t + \frac{|p_l - p_r|}{p_l + p_r}\Delta t, \tag{64}$$

where $\varepsilon = 0.01$ is used. Numerical tests are conducted on the computational domain of [0, 2] × [0, 2] × [0, 2]. The uniform meshes with $\Delta x = \Delta y = \Delta z = 2/N$ and $N = 20, 40, 60, 80$ are used. The L^1 error of the density field at $t = 2$ is extracted and shown in Fig. 1. It can be seen that the GKFS is about the second order of accuracy in space.

4.1 Case 1: 3D lid-driven cavity flow

The 3D lid-driven cavity flows in a cube are simulated to test the capability of the proposed explicit GKFS for simulating 3D incompressible viscous flows. The non-uniform mesh of 81 × 81 × 81 is used for the cases of Re = 100 and 400. The mesh point in the x-direction is generated by

$$\begin{aligned}&x_i = 0.5\left(1-\eta\ \tan^{-1}((1-\kappa_i)\cdot\ \tan(1/\eta))\right), && i \le (i\ \max + 1)/2,\\&x_i = 1.0 - x_{i\ \max+1-i}, && else.\end{aligned} \tag{65}$$

Where $\kappa_i = (i-1)/((i\max - 1)/2)$, i and $i$max are the mesh point index and total number of mesh points in the x direction; η is the parameter to control the mesh

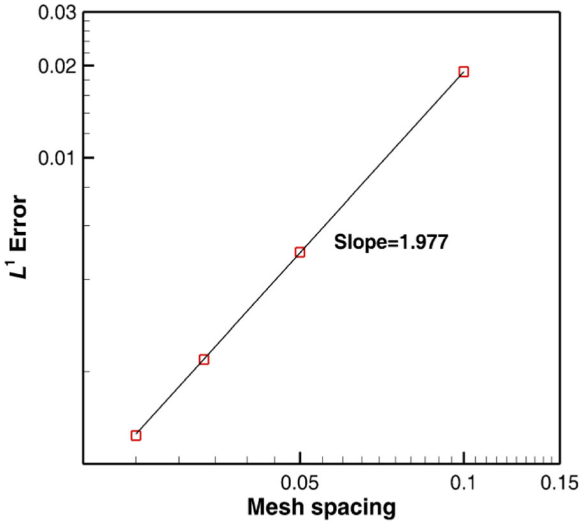


Fig. 1 L^1 error of the density field for the advection of density perturbation

stretching and is selected as 1.1 in this study. Similarly, the mesh point in the y- and z-directions is generated in the same way.

In the current simulation, the fluid density is taken as $\rho = 1.0$ and the lid velocity is chosen as $U_\infty = 0.1$. Initially, the density inside the cavity is constant and the flow is static. The lid on the top boundary moves along the x-direction. The no-slip wall condition is imposed at all boundaries. To quantitatively examine the performance of 3D GKFS, the velocity profiles of x-direction component u along the vertical centerline and y-direction component v along the horizontal centerline for $\mathrm{Re} = 100$ and 400 are plotted in Fig. 2. For comparison, the results of Shu et al. [31] and Wu and Shu [32] are also included in the figure. It can be found that all the velocity profiles by current 3D GKFS agree very well with those of Shu et al. [31] and Wu and Shu [32], which demonstrates the capability of present solver for the simulation of 3D incompressible flows on non-uniform grids. To further show the flow patterns of 3D lid-driven cavity flow, the streamlines for $\mathrm{Re} = 100$ and 400 at three orthogonal mid-planes located at $x = 0.5$, $y = 0.5$ and $z = 0.5$ are displayed in Fig. 3. The flow patterns along the mid-plane of $z = 0.5$ in Fig. 3 demonstrate that the primary vortices gradually shift toward the center position and the second vortices gradually move to the lower bottom wall when the Reynolds number is increased. In this process, the strength of these vortices is also enhanced, which can also be proven by the flow patterns along other two mid-planes. All these observations match well with those in Shu et al. [31].

4.2 Case 2: incompressible flow past a stationary sphere

In this section, the 3D GKFS is applied to a benchmark case of incompressible flow past a stationary sphere. In this case, the flow is characterized by the Reynolds number defined by $\mathrm{Re} = \rho U_\infty D / \mu$, where ρ and μ are the fluid density and dynamic viscosity, respectively. U_∞ is the free stream velocity and D is the sphere diameter. To simulate this test case with a simple Cartesian mesh, the implicit boundary condition-enforced immersed boundary method [33, 34] is coupled with the present 3D GKFS. The computational domain is selected as a rectangular box of $30D \times 20D \times 20D$ in the x-, y- and z- directions. The sphere is initially placed at $(10D,\ \ 10D,\ \ 10D)$, which is discretized by triangular elements with 1195 vertices. As shown in Fig. 4, a non-uniform Cartesian mesh with mesh size of $137 \times 122 \times 122$ is used, in which a uniform mesh spacing of $0.02D$ is applied around the sphere. The no-slip condition on the curved boundary is imposed by correcting the velocity on the Cartesian mesh through the immersed boundary method [33, 34]. Here, laminar flows at low Reynolds numbers of 50, 100, 150, 200 and 250 are considered.

At first, the drag coefficients at $\mathrm{Re} = 100$, 200 and 250 are computed and compared quantitatively in Table 1 to verify the accuracy of the present solver. The numerical results of Johnson and Patel [35], Wu and Shu [32], Kim et al. [36] and Wang et al. [37] are also included in the table for comparison. It can be clearly observed that the present results match well with those in the literature.

Then, for the steady axisymmetric flow, the streamlines of flow past a sphere at $\mathrm{Re} \leq 200$ are depicted in Fig. 5. Since the flow is axisymmetric, only the streamlines on the x- y plane of symmetry are given. From the figure, a recirculation region appears behind the sphere and its length L_s increases with Reynolds number. Quantitative

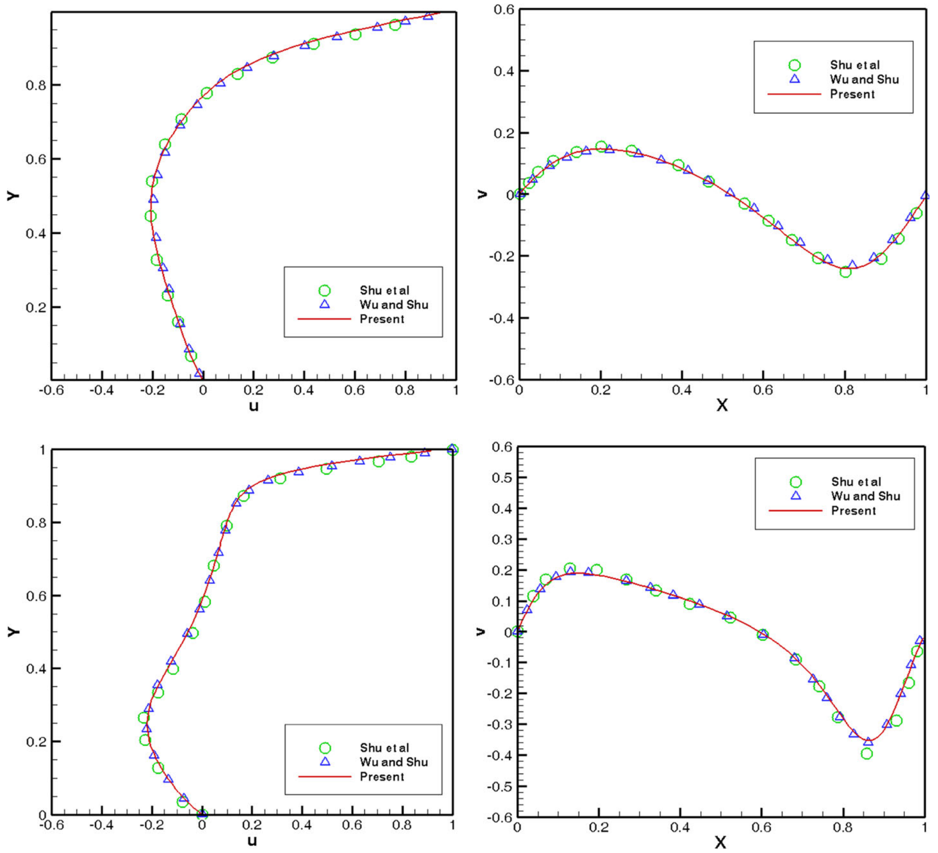
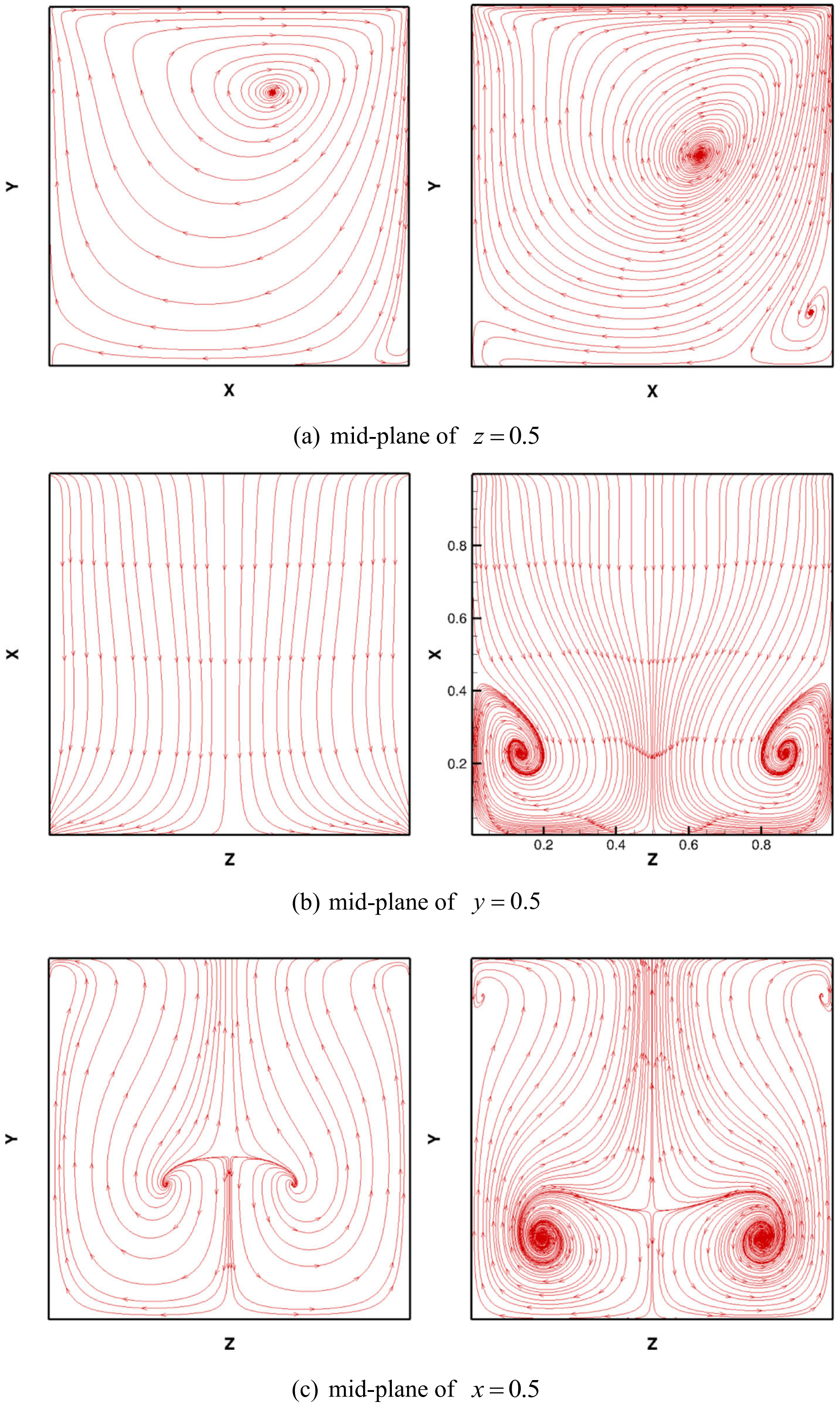


Fig. 2 Comparison of velocity profiles on the plane of $z = 0.5$ for 3D lid-driven cavity flow. Upper: Re = 100; Lower: Re = 400

comparison between the present results of L_s and those of Johnson and Patel [35] and Gilmanov et al. [38] is made in Fig. 6. Good agreement can be found in the figure. When the Reynolds number is increased to 250, the phenomenon of steady non-axisymmetric pattern shows up, which can be seen in Fig. 7. In the figure, the streamlines on the x- z plane remain symmetric. However, there are two asymmetric vortices on the x- y plane, which implies that the symmetry is lost in this plane. These results are in good agreement with previous investigations [35, 38].

4.3 Case 3: flow around ONERA M6 wing

The ONERA M6 test case is chosen to validate the present solver for the simulation of compressible viscous flows with complex geometry. For numerical simulation, the free-stream Mach number is taken as $M_\infty = 0.8395$, the mean-chord based Reynolds number is chosen as $Re = 11.72 \times 10^6$ and the angle of attack is $\alpha = 3.06^\circ$. The computational mesh in the NASA website [39] is adopted in this work, which has 4 blocks and 316, 932 grid points. The mesh spacing of the first mesh point adjacent to the wing surface is 4.5×10^{-5}. To take turbulent effect into consideration, the Spalart-Allmaras turbulent model [40] is applied. Figure 8 shows the pressure contours at the wing surface obtained from the present solver, in which the "λ" shape shock wave on the upper surface is clearly presented. The above phenomenon matches well with the result from sphere



(a) mid-plane of $z = 0.5$

(b) mid-plane of $y = 0.5$

(c) mid-plane of $x = 0.5$

Fig. 3 Streamlines on three mid-planes for Re = 100 (left) and Re = 400 (right). **a** mid-plane of $z = 0.5$. **b** mid-plane of $y = 0.5$. **c** mid-plane of $x = 0.5$

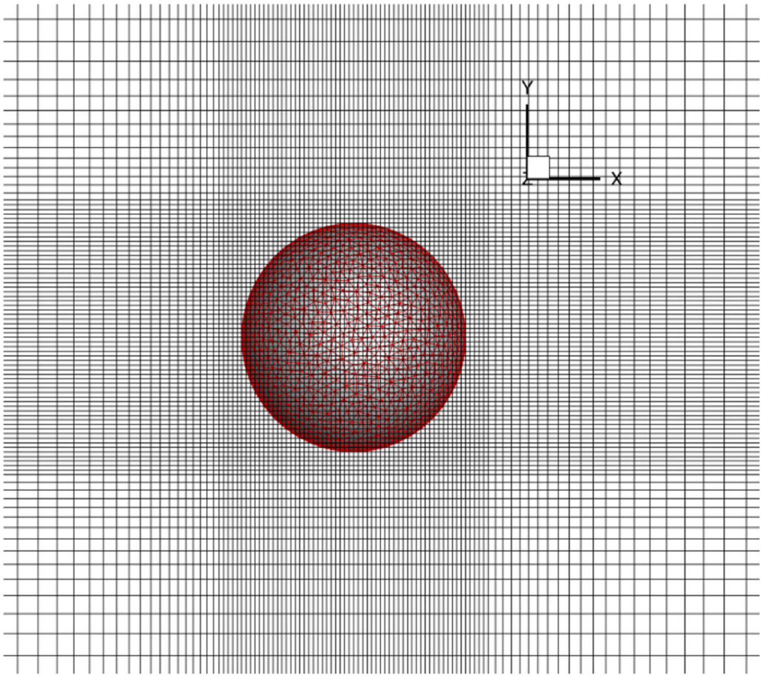


Fig. 4 Partial view of computational mesh for flow past a sphere

function-based gas-kinetic scheme [41]. To further validate the present results, the pressure coefficient distributions at selected span-wise locations obtained from the present solver are displayed in Fig. 9. The numerical results of WIND scheme [39] and the experimental results [42] are also included for comparison. As can be seen from the figure, the present results are close to those of WIND scheme [39] and compare well with the experimental data [42]. What is more, the pressure coefficient distributions at 65% and 80% spans show that the present results are much closer to the experimental results [42] as compared with the results from WIND scheme [39]. It demonstrates that the present solver captures the shock wave more precisely and controls the numerical dissipation well.

To further investigate the performance of GKFS for simulation of high speed flows, in this test, we change the Mach number to $M_\infty = 5$ while keep other parameters the same as the above case. Figure 10 shows the pressure contours and the pressure

Table 1 Comparison of drag coefficient for flow past a stationary sphere

Re	References	C_d
100	Johnson and Patel [35]	1.112
	Wu and Shu [32]	1.128
	Present	1.116
200	Johnson and Patel [35]	0.79
	Wu and Shu [32]	0.8
	Present	0.791
250	Kim et al. [36]	0.706
	Wang et al. [37]	0.746
	Present	0.720

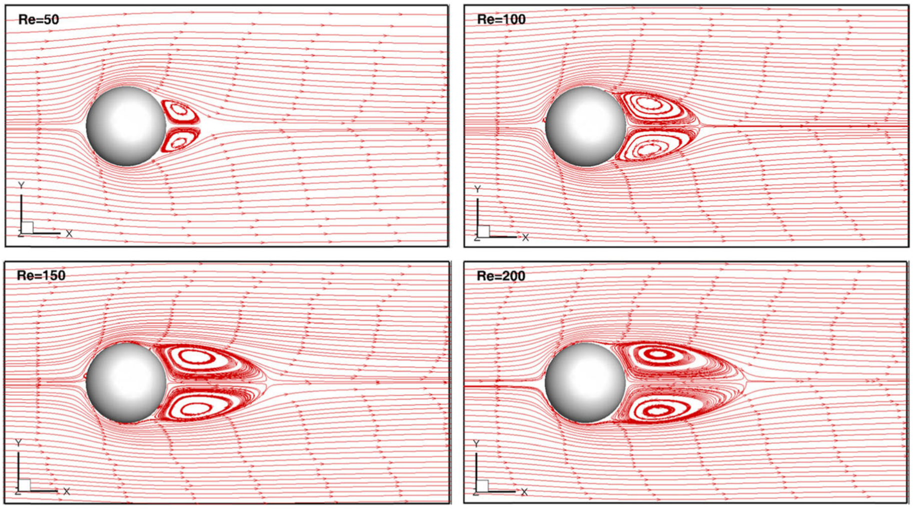


Fig. 5 Streamlines at four different Reynolds numbers of 50, 100, 150 and 200 in the steady axisymmetric regime

coefficient distribution at 65% span. It can be seen that the GKFS captures strong shock wave without any oscillation and the pressure coefficient distribution agrees well with the AUSM scheme [43].

4.4 Case 4: DLR-F6 wing-body configuration

The DLR-F6 wing-body configuration is a generic transport aircraft model from the 3rd AIAA CFD drag prediction workshop (DPW III) [44]. At first, numerical simulations are conducted at a free-stream Mach number of $M_\infty = 0.75$, a mean-chord based

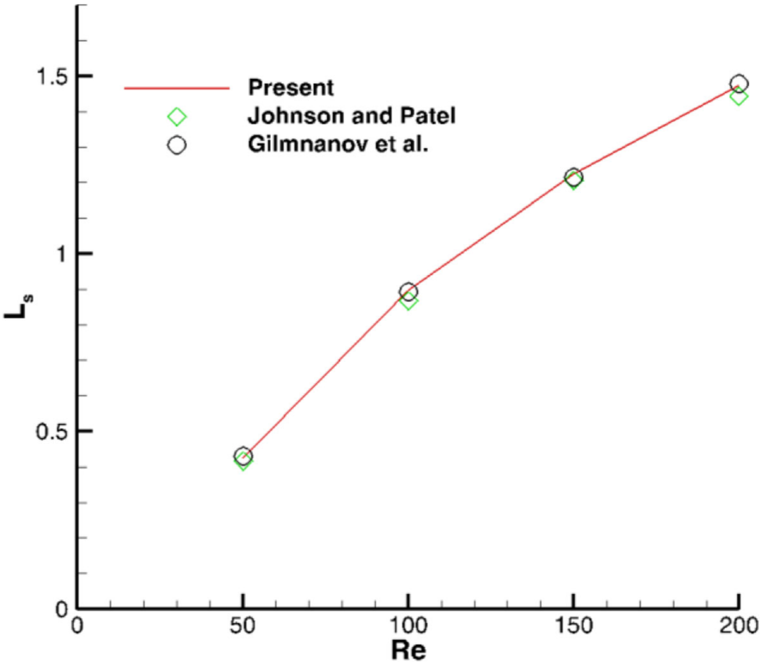


Fig. 6 Comparison of recirculation length L_s

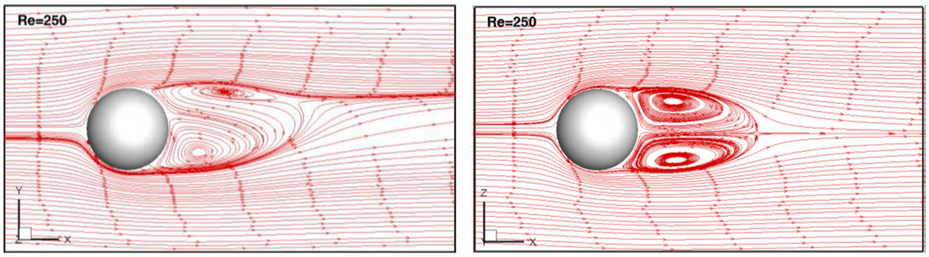


Fig. 7 Streamlines for flow past a stationary sphere at Re = 250 in the steady non-axisymmetric regime

Reynolds number of $\mathrm{Re} = 3 \times 10^6$ and an angle of attack $\alpha = 0.49^\circ$. The geometry and computational mesh from the NASA website [45] are utilized in the current work. Owing to the limitation of the computer's memory, only the coarse mesh with 26 blocks and 2,298,880 cells is used. Figure 11 is the pressure contour of DLR-F6 wing-body obtained by present GKFS. The separation bubble at the intersection of wing and body is clearly recognized in Fig. 12, which is in line with the observations of Vassberg et al. [46]. To make a quantitative comparison, the pressure coefficient distributions at selected span-wise locations obtained by present 3D GKFS are compared with the experimental results [47] and numerical results of Vassberg et al. [46] and Yang et al. [48] in Fig. 13. It can be observed that the current results are close to those of Vassberg et al.

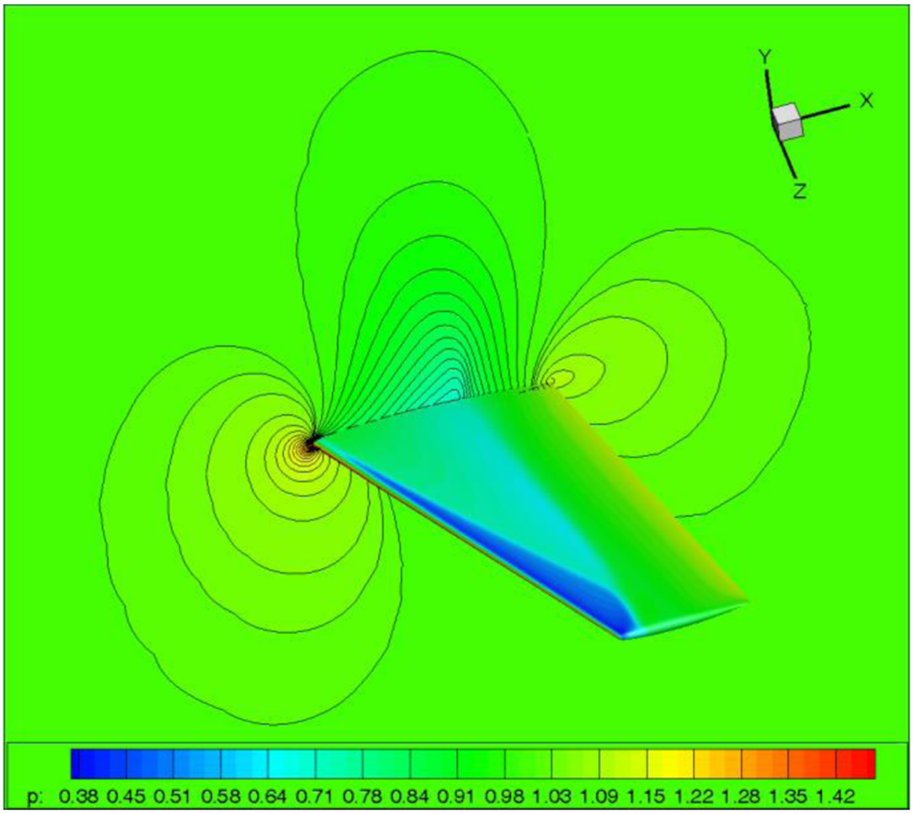


Fig. 8 Pressure contours of flow around ONERA M6 wing

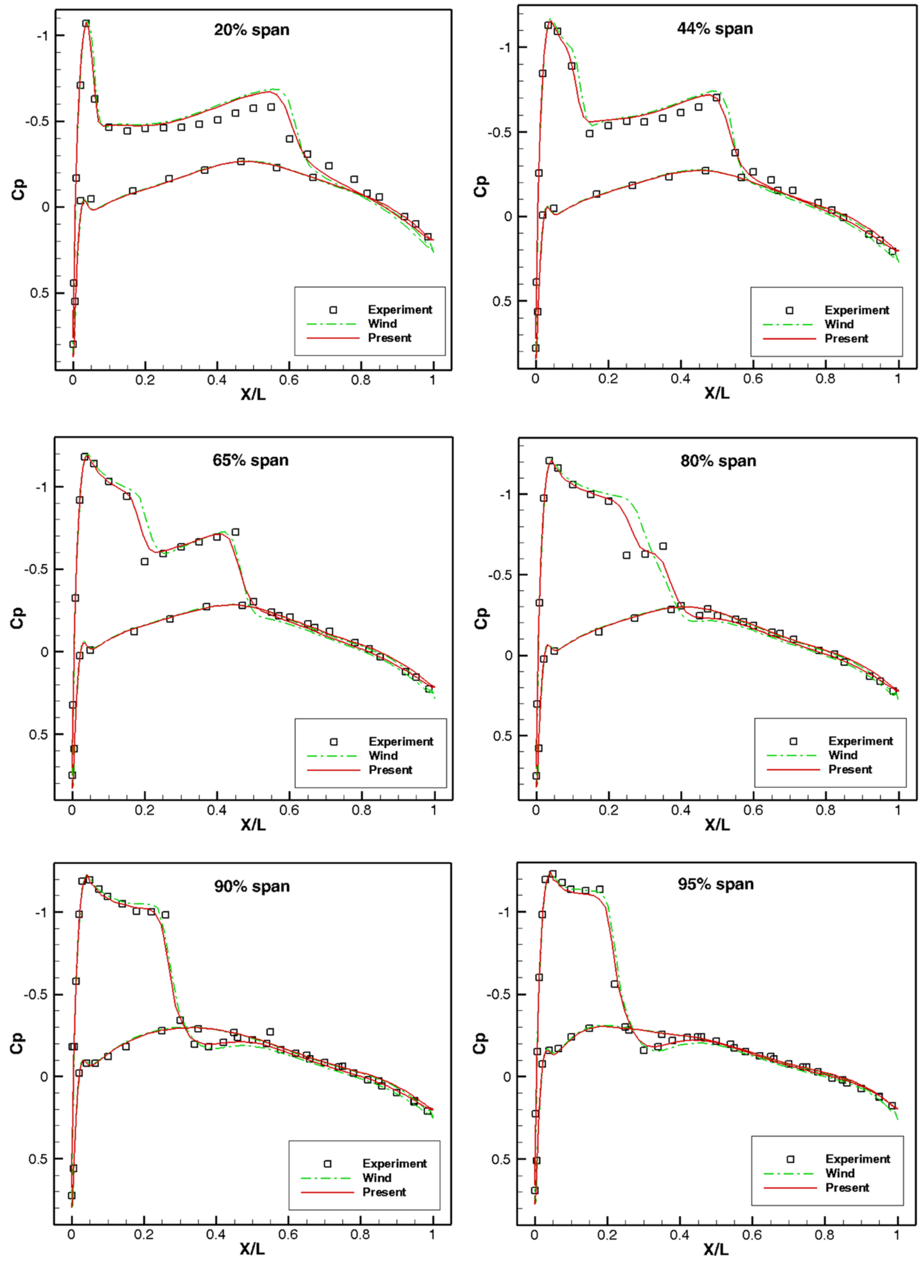


Fig. 9 Comparison of pressure coefficient distributions at selected positions for ONERA M6 Wing

[46] and Yang et al. [48] and all of them basically agree well with the experimental measurement [47].

To further verify the force coefficients of current solver for the DLR-F6 wing-body, another test case is simulated with the free stream condition of Mach number $M_\infty = 0.75$, Reynolds number $Re = 5 \times 10^6$ and angle of attach $\alpha = 0^\circ$. Table 2 shows the present results of force coefficients, including lift coefficient C_l, pressure drag coefficient $C_{d,\ p}$, friction drag coefficient $C_{d,\ f}$, total drag coefficient C_d and moment coefficient C_M. The results of present solver are close to the results of LBFS [48] and can essentially match well with the reference data of Vassberg et al. [46].

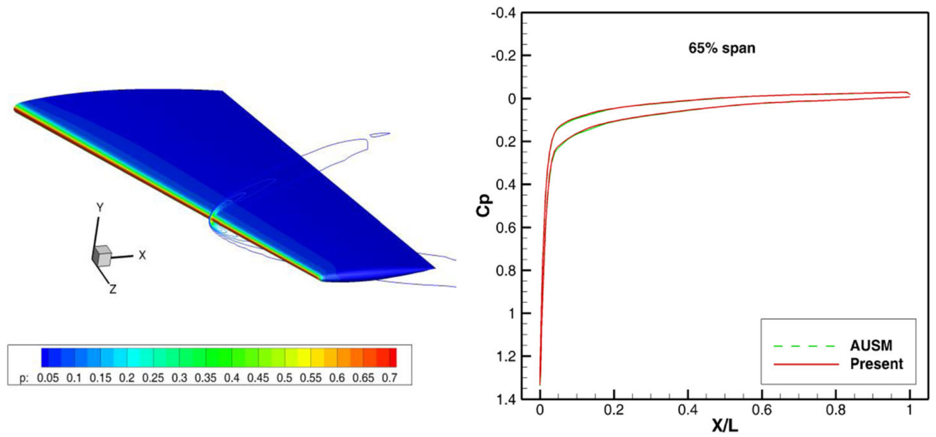


Fig. 10 Pressure contours and pressure coefficient distribution at 65% span for ONERA M6 Wing at $M_\infty = 5$

5 Conclusions

This paper presents a three-dimensional GKFS for simulation of incompressible and compressible viscous flows. The present work is the extension of our previous work [1], where a new gas-kinetic scheme is presented to simulate two-dimensional viscous flows. In this work, the non-equilibrium distribution function is evaluated by the difference of equilibrium distribution functions at cell interface and its surrounding points. As a result, the distribution function at the interface can be simply derived and the formulations of the conservative variables and fluxes at cell interface can be explicitly given. Since the solution of 3D continuous Boltzmann equation is reconstructed locally at cell interface, the present scheme can be viewed as a truly 3D flux solver for viscous flows. To consider general 3D cases, a local coordinate transformation is made to transform the velocities in the global Cartesian coordinate system to the local normal and tangential directions at each cell interface. In this way, all the interfaces can be treated using the same way. Several numerical experiments are conducted to validate the proposed scheme, including 3D lid-driven cavity flow, incompressible flow past a stationary sphere, compressible flow around ONERA M6 wing and DLR-F6 wing-body configuration. Numerical results showed that the proposed flux solver can provide accurate numerical results for three-dimensional incompressible and compressible viscous flows.

6 Appendix

6.1 Moments of Maxwellian Distribution Function

In the paper, some notations are taken to simplify the formulations. In this appendix, the notations for the moments of Maxwellian distribution function are introduced. At first, the Maxwellian distribution function for 3D flows is given as (Eq. (2))

$$g = \rho\left(\frac{\lambda}{\pi}\right)^{\frac{K+3}{2}} e^{-\lambda\left((u-U)^2+(v-V)^2+(w-W)^2+\xi^2\right)}.$$

Following the idea of [2], the notation for the moments of g is defined as

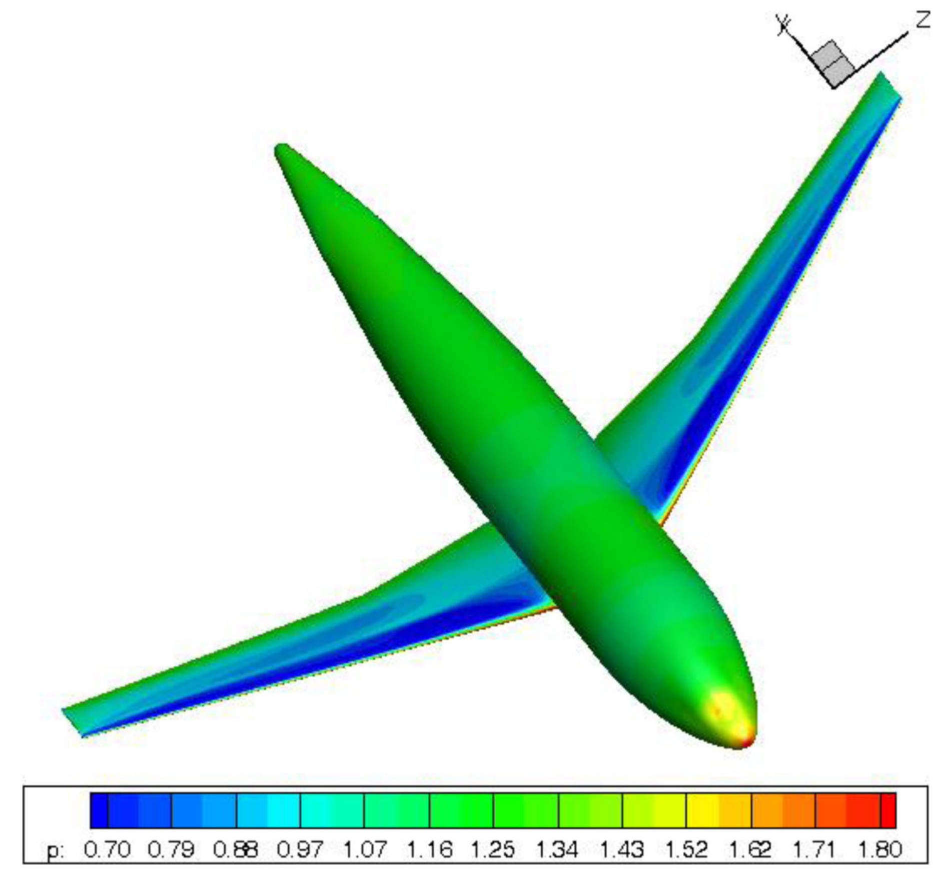

Y
Z
p: 0.70 0.79 0.88 0.97 1.07 1.16 1.25 1.34 1.43 1.52 1.62 1.71 1.80

Fig. 11 Pressure contours of DLR-F6 wing/body

$$\rho\langle\cdot\rangle = \int(\cdot)gdudvdwd\xi. \tag{A.1}$$

Then the general moment formualtion becomes

$$\left\langle u^n v^m w^l \xi^p \right\rangle = \langle u^n\rangle\langle v^m\rangle\left\langle w^l\right\rangle\langle\xi^p\rangle. \tag{A.2}$$

It should be noted that in Section 3, the conservative variables and numerical fluxes are derived at local interface and the transformation of velocities is made at each cell

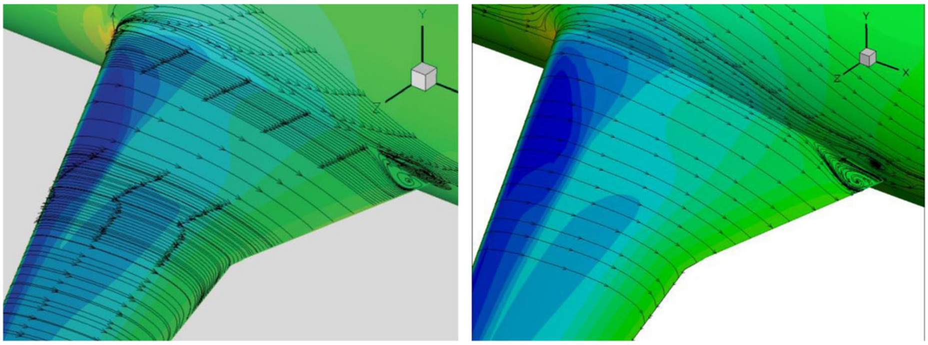


Fig. 12 Separation bubble on the intersection of wing and body (left: Vassberg et al. [46]; right: present)

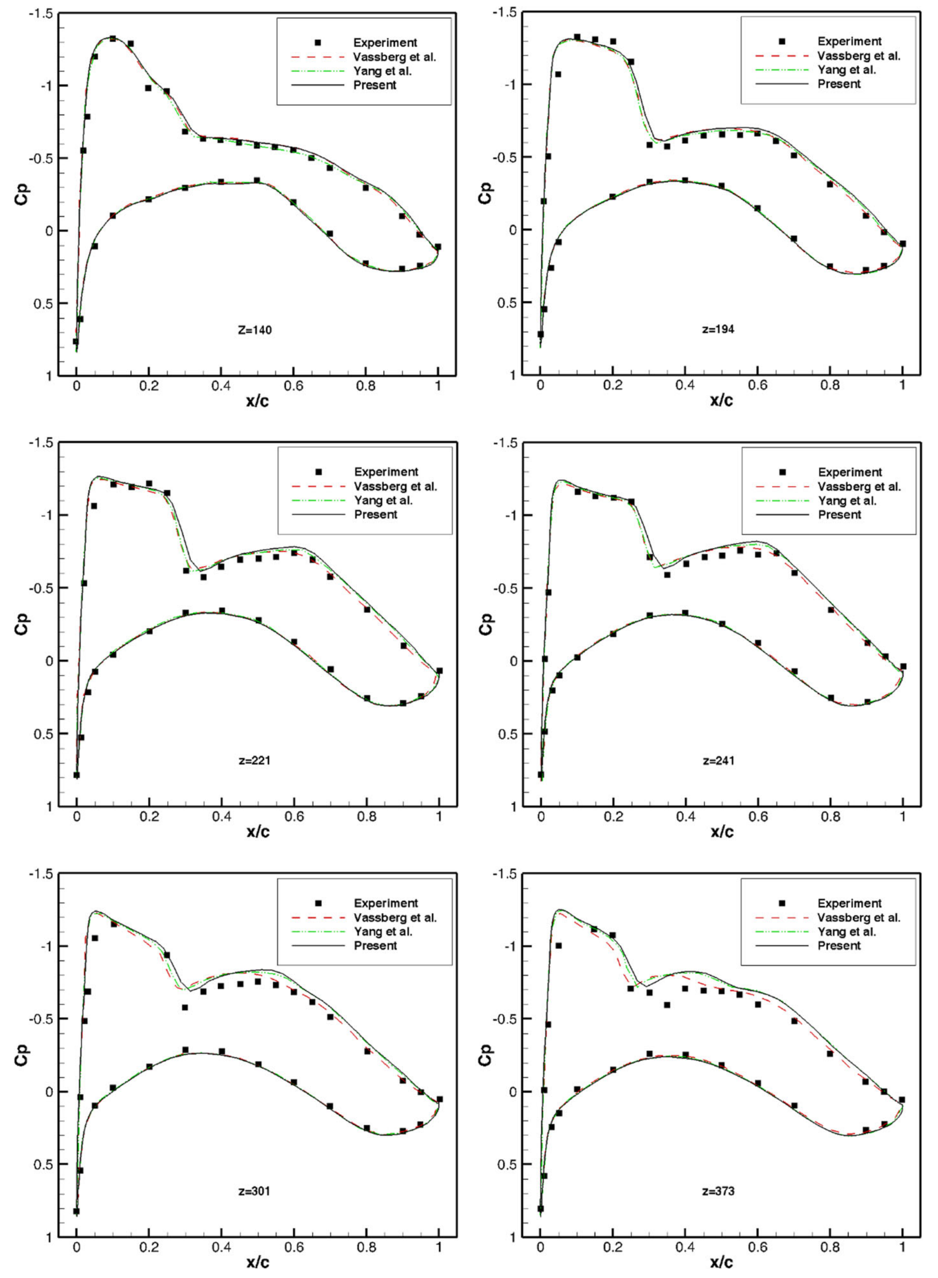


Fig. 13 Comparison of pressure coefficient distributions of DLR-F6 wing/body at different locations

Table 2 Comparison of force coefficients for DLR-F6 wing-body configuration

References	C_l	$C_{d,\ p}$	$C_{d,\ f}$	C_d	C_M
Vassberg et al. [46]	0.51600	0.01502	0.01229	0.02731	−0.15280
Yang et al. [48]	0.52312	0.01554	0.00979	0.02533	−0.14988
Present	0.52470	0.01549	0.00947	0.02496	−0.16230

interface. Therefore, the moments of normal and tangential velocities ($u^{'}$, $v^{'}$, $w^{'}$) are presented to keep pace with the formulations in the paper. As the integration of three components of the particle velocity ($u^{'}, v^{'}, w^{'}$) are similar to each other, only the moments of $u^{'}$ are presented here. When the integral of velocity is from −∞ to +∞, the moments of $u^{'n}$ and ξ^n are

$$\left\langle u^{'0} \right\rangle = 1, \tag{A.3}$$

$$\left\langle u^{'1} \right\rangle = U^{'}, \tag{A.4}$$

$$\left\langle u^{'(n)} \right\rangle = U^{'} \left\langle u^{'(n-1)} \right\rangle + \frac{n-1}{2\lambda} \left\langle u^{'(n-2)} \right\rangle, \tag{A.5}$$

and

$$\langle \xi^0 \rangle = 1, \tag{A.6}$$

$$\langle \xi^2 \rangle = \frac{K}{2\lambda}, \tag{A.7}$$

$$\langle \xi^4 \rangle = \frac{K^2 + 2K}{4\lambda^2}. \tag{A.8}$$

When the moments for $u^{'n}$ are calculated in the half space, the exponential function and the complementary error function appear in the formulation. Take notation of the integral from 0 to +∞ as $\langle \cdot \rangle_{>0}$ and integral from −∞ to 0 as $\langle \cdot \rangle_{<0}$, the moments become

$$a_l = \left\langle u^{'0} \right\rangle_{>0} = \frac{1}{2}\ erfc\left(-\sqrt{\lambda_l} U_l^{'} \right), \tag{A.9}$$

$$b_l = \left\langle u^{'1} \right\rangle_{>0} = U_l^{'} a_l + \frac{1}{2} \frac{e^{-\lambda_l U_l^{'2}}}{\sqrt{\pi \lambda_l}}, \tag{A.10}$$

$$c_l = \left\langle u^{'2} \right\rangle_{>0} = U_l^{'} b_l + \frac{1}{2\lambda_l} a_l, \tag{A.11}$$

$$d_l = \left\langle u^{'3} \right\rangle_{>0} = U_l^{'} c_l + \frac{1}{\lambda_l} b_l, \tag{A.12}$$

$$e_l = \left\langle u^{'4} \right\rangle_{>0} = U_l^{'} d_l + \frac{3}{2\lambda_l} c_l, \tag{A.13}$$

and

$$a_r = \left\langle u^{'0} \right\rangle_{<0} = \frac{1}{2}\ erfc\left(\sqrt{\lambda_r} U_r^{'} \right), \tag{A.14}$$

$$b_r = \left\langle u^{'1} \right\rangle_{<0} = U_r^{'} a_r - \frac{1}{2} \frac{e^{-\lambda_r U_r^{'2}}}{\sqrt{\pi \lambda_r}}, \tag{A.15}$$

$$c_r = \left\langle u^{'2} \right\rangle_{<0} = U_r^{'} b_r + \frac{1}{2\lambda_r} a_r, \tag{A.16}$$

$$d_r = \left\langle u^{'3} \right\rangle_{<0} = U_r^{'} c_r + \frac{1}{\lambda_r} b_r, \tag{A.17}$$

$$e_r = \left\langle u'^4 \right\rangle_{<0} = U'_r d_r + \frac{3}{2\lambda_r} c_r. \tag{A.18}$$

In order to compute the coefficients A_1 to A_5 via Eqs. (45)–(49), the values of G_1 to G_5 in Eq. (44) should be calculated in advance. The explicit formulations of G_1 to G_5 can be given as

$$G_1 = \frac{\partial\left(\rho_l b_l + \rho_r b_r\right)}{\partial n_1} + \frac{\partial\left(\rho_l V'_l a_l + \rho_r V'_r a_r\right)}{\partial n_2} + \frac{\partial\left(\rho_l W'_l a_l + \rho_r W'_r a_r\right)}{\partial n_3}, \tag{A.19}$$

$$G_2 = \frac{\partial\left(\rho_l c_l + \rho_r c_r\right)}{\partial n_1} + \frac{\partial\left(\rho_l V'_l b_l + \rho_r V'_r b_r\right)}{\partial n_2} + \frac{\partial\left(\rho_l W'_l b_l + \rho_r W'_r b_r\right)}{\partial n_3}, \tag{A.20}$$

$$\begin{aligned} G_3 = {} & \frac{\partial(\rho_l V_l{}' b_l + \rho_r V_r{}' b_r)}{\partial n_1} + \frac{\partial[(\rho_l V'^2_l + p_l)a_l + (\rho_r V'^2_r + p_r)a_r]}{\partial n_2} \\ & + \frac{\partial(\rho_l V_l{}' W_l{}' a_l + \rho_r V_r{}' W_r{}' a_r)}{\partial n_3}, \end{aligned} \tag{A.21}$$

$$\begin{aligned} G_4 = {} & \frac{\partial(\rho_l W_l{}' b_l + \rho_r W_r{}' b_r)}{\partial n_1} + \frac{\partial(\rho_l V_l{}' W_l{}' a_l + \rho_r V_r{}' W_r{}' a_r)}{\partial n_2} \\ & + \frac{\partial[(\rho_l W'^2_l + p_l)a_l + (\rho_r W'^2_r + p_r)a_r]}{\partial n_3}, \end{aligned} \tag{A.22}$$

$$\begin{aligned} G_5 = {} & \frac{\partial}{\partial n_1}\{\rho_l[d_l + (V'^2_l + W'^2_l + (b-1)RT_l)b_l] \\ & +\rho_r[d_r + (V'^2_r + W'^2_r + (b-1)RT_r)b_r]\} \\ & + \frac{\partial}{\partial n_2}\{\rho_l V'_l[c_l + (V'^2_l + W'^2_l + (b+1)RT_l)a_l] \\ & +\rho_r V'_r[c_r + (V'^2_r + W'^2_r + (b+1)RT_r)a_r]\} \\ & + \frac{\partial}{\partial n_3}\{\rho_l W'_l[c_l + (V'^2_l + W'^2_l + (b+1)RT_l)a_l] \\ & +\rho_r W'_r[c_r + (V'^2_r + W'^2_r + (b+1)RT_r)a_r]\}. \end{aligned} \tag{A.23}$$

Abbreviations

2D: Two-dimensional; 3D: Three-dimensional; KFVS: Kinetic flux vector scheme; BGK: Bhatnagar-Gross-Krook; FDS: Flux difference splitting; CFD: Computational fluid dynamics; GKFS: Gas-kinetic flux solver; LU-SGS: Lower-upper symmetric-Gauss-Seidel

Acknowledgments

This work is supported by National Natural Science Foundation of China (Grant Nos. 11772157 and 11832012).

Authors' contributions

The contribution of the authors to this work is equivalent. All authors read and approved the final manuscript.

Funding

National Natural Science Foundation of China (Grant Nos. 11772157 and 11832012).

Availability of data and materials

All data generated or analyzed during this study are included in this published article.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Huawei Technologies Co., Ltd, Bantian, Longgang District, Shenzhen 518129, China. [2]Department of Mechanical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260, Singapore.

Received: 7 March 2020 Accepted: 1 May 2020
Published online: 26 May 2020

References

1. Sun Y, Shu C, Teo CJ et al (2015) Explicit formulations of gas-kinetic flux solver for simulation of incompressible and compressible viscous flows. J Comput Phys 300:492–519
2. Xu K (1998) Gas-kinetic schemes for unsteady compressible flow simulations. VKI for Fluid Dynamics Lecture Series
3. Su M, Xu K, Ghidaoui MS (1999) Low-speed flow simulation by the gas-kinetic scheme. J Comput Phys 150(1):17–39
4. Xu K, Mao M, Tang L (2005) A multidimensional gas-kinetic BGK scheme for hypersonic viscous flow. J Comput Phys 203(2):405–421
5. Tian CT, Xu K, Chan KL et al (2007) A three-dimensional multidimensional gas-kinetic scheme for the Navier-Stokes equations under gravitational fields. J Comput Phys 226(2):2003–2027
6. Jiang J, Qian Y (2012) Implicit gas-kinetic BGK scheme with multigrid for 3D stationary transonic high-Reynolds number flows. Comput Fluids 66:21–28
7. Yang LM, Shu C, Wu J (2014) A simple distribution function-based gas-kinetic scheme for simulation of viscous incompressible and compressible flows. J Comput Phys 274:611–632
8. Li W, Kaneda M, Suga K (2014) An implicit gas kinetic BGK scheme for high temperature equilibrium gas flows on unstructured meshes. Comput Fluids 93:100–106
9. Li ZH, Zhang HX (2004) Study on gas kinetic unified algorithm for flows from rarefied transition to continuum. J Comput Phys 193(2):708–738
10. Xu K, Huang JC (2010) A unified gas-kinetic scheme for continuum and rarefied flows. J Comput Phys 229(20):7747–7764
11. Guo Z, Xu K, Wang R (2013) Discrete unified gas kinetic scheme for all Knudsen number flows: low-speed isothermal case. Phys Rev E 88(3):033305
12. Liu S, Yu P, Xu K et al (2014) Unified gas-kinetic scheme for diatomic molecular simulations in all flow regimes. J Comput Phys 259:96–113
13. Roe PL (1981) Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 43(2):357–372
14. Engquist B, Osher S (1981) One-sided difference approximations for nonlinear conservation laws. Math Comput 36(154): 321–351
15. Van Leer B (1997) Flux-vector splitting for the Euler equation. In: Hussaini MY, van Leer B, Van Rosendale J (eds) Upwind and high-resolution schemes. Springer, Heidelberg
16. Osher S, Chakravarthy S (1983) Upwind schemes and boundary conditions with applications to Euler equations in general geometries. J Comput Phys 50(3):447–481
17. Pullin DI (1980) Direct simulation methods for compressible inviscid ideal-gas flow. J Comput Phys 34(2):231–244
18. Deshpande SM (1986) A second-order accurate kinetic-theory-based method for inviscid compressible flows. NASA STI/Recon Technical Report N 87
19. Perthame B (1992) Second-order Boltzmann schemes for compressible Euler equations in one and two space dimensions. SIAM J Numer Anal 29(1):1–19
20. Mandal JC, Deshpande SM (1994) Kinetic flux vector splitting for Euler equations. Comput Fluids 23(2):447–478
21. Chou SY, Baganoff D (1997) Kinetic flux-vector splitting for the Navier-Stokes equations. J Comput Phys 130(2):217–230
22. Prendergast KH, Xu K (1993) Numerical hydrodynamics from gas-kinetic theory. J Comput Phys 109(1):53–66
23. Chae D, Kim C, Rho OH (2000) Development of an improved gas-kinetic BGK scheme for inviscid and viscous flows. J Comput Phys 158(1):1–27
24. Xu K (2001) A gas-kinetic BGK scheme for the Navier-Stokes equations and its connection with artificial dissipation and Godunov method. J Comput Phys 171(1):289–335
25. Xu K, Sun Q, Yu P (2010) Valid physical processes from numerical discontinuities in computational fluid dynamics. Int J Hypersonics 1(3):157–172
26. Bhatnagar PL, Gross EP, Krook M (1954) A model for collision processes in gases. I Small amplitude processes in charged and neutral one-component systems. Phys Rev 94(3):511–525
27. Shakhov EM (1968) Generalization of the Krook kinetic relaxation equation. Fluid Dyn 3(5):95–96
28. Yoon S, Jameson A (1988) Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations. AIAA J 26(9):1025–1026
29. Venkatakrishnan V (1995) Convergence to steady state solutions of the Euler equations on unstructured grids with limiters. J Comput Phys 118(1):120–130
30. Pan L, Xu K (2020) High-order gas-kinetic scheme with three-dimensional WENO reconstruction for the Euler and Navier-Stokes solutions. Comput Fluids 198:104401
31. Shu C, Niu XD, Chew YT (2003) Taylor series expansion and least squares-based lattice Boltzmann method: three-dimensional formulation and its applications. Int J Mod Phys C 14(07):925–944
32. Wu J, Shu C (2010) An improved immersed boundary-lattice Boltzmann method for simulating three-dimensional incompressible flows. J Comput Phys 229(13):5022–5042
33. Wu J, Shu C (2009) Implicit velocity correction-based immersed boundary-lattice Boltzmann method and its applications. J Comput Phys 228(6):1963–1979
34. Wang Y, Shu C, Teo CJ et al (2015) An immersed boundary-lattice Boltzmann flux solver and its applications to fluid-structure interaction problems. J Fluids Struct 54:440–465
35. Johnson TA, Patel VC (1999) Flow past a sphere up to a Reynolds number of 300. J Fluid Mech 378:19–70
36. Kim J, Kim D, Choi H (2001) An immersed-boundary finite-volume method for simulations of flow in complex geometries. J Comput Phys 171(1):132–150
37. Wang XY, Yeo KS, Chew CS et al (2008) A SVD-GFD scheme for computing 3D incompressible viscous fluid flows. Comput Fluids 37(6):733–746
38. Gilmanov A, Sotiropoulos F, Balaras E (2003) A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids. J Comput Phys 191(2):660–669

39. Slater JW (2002) ONERA M6 Wing. http://www.grc.nasa.gov/WWW/wind/valid/m6wing/m6wing01/m6wing01.html. Accessed 30 Aug 2002
40. Spalart P, Allmaras S (1992) A one-equation turbulence model for aerodynamic flows. AIAA paper 92-0439
41. Yang LM, Shu C, Wu J (2015) A three-dimensional explicit sphere function-based gas-kinetic flux solver for simulation of inviscid compressible flows. J Comput Phys 295:322–339
42. Schmitt V, Charpin F (1979) Pressure distributions on the ONERA-M6-wing at transonic Mach numbers, experimental data base for computer program assessment. Report of the Fluid Dynamics Panel Working Group 04, AGARD AR 138, May 1979
43. Liou MS (2006) A sequel to AUSM, part II: AUSM+-up for all speeds. J Comput Phys 214:137–170
44. Frink NT (2006) 3rd AIAA CFD Drag Prediction Workshop. https://aiaa-dpw.larc.nasa.gov/Workshop3/workshop3.html. Accessed 29 Nov 2006
45. Morrison J (2006) https://dpw.larc.nasa.gov/DPW3/multiblock_Boeing_CGNS/F6wb_struc_boeing. Accessed 18 Jan 2006
46. Vassberg J, Tinoco E, Mani M et al (2007) Summary of the third AIAA CFD drag prediction workshop. Paper presented at the 45th AIAA aerospace sciences meeting and exhibit, 1 Jan 2007
47. Rumsey CL, Rivers SM, Morrison JH (2005) Study of CFD variation on transport configurations from the second drag-prediction workshop. Comput Fluids 34(7):785–816
48. Yang LM, Shu C, Wu J (2014) A hybrid lattice Boltzmann flux solver for simulation of 3D compressible viscous flows. Paper presented at the eighth international conference on computational fluid dynamics, Chengdu, China, 24-29 July 2014

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.