High-Order Discontinuous Galerkin Solution of Compressible Flows with a Hybrid Lattice Boltzmann Flux

Sun Yongcheng * , *Cai Junwei* , *Qin Wanglong*

The 28th Research Institute of China Electronics Technology Group Corporation, Nanjing 210007, P. R. China

(Received 14 November 2017; revised 18 December 2017; accepted 22 December 2017)

Abstract: A discontinuous Galerkin (DG)-based lattice Boltzmann method is employed to solve the Euler and Navier-Stokes equations. Instead of adopting the widely used local Lax-Friedrichs flux and Roe Flux etc. , a hybrid lattice Boltzmann flux solver (LBFS) is employed to evaluate the inviscid flux across the cell interfaces. The main advantage of the hybrid LBFS is its flexibility for capturing both strong shocks and thin boundary layers through introducing a function which varies from zero to one to control the artificial viscosity. Numerical results indicate that the hybrid lattice Boltzmann flux solver behaves very well combining with the high-order DG method when simulating both inviscid and viscous flows.

Key words: hybrid lattice Boltzmann flux solver; discontinuous Galerkin (DG) method; Euler equations; Navier-Stokes equations

CLC number: O241. 6 **Document code**: A **Article ID**: 1005-1120(2018)03-0413-10

0 Introduction

During the past two decades, high-order discontinuous Galerkin (DG) methods[1] have been receiving growing interests due to its advantages such as high accuracy, great geometry flexibility, straightforward implementation of h/P adaption and parallel computing. Bassi et al. used a high-order DG method to solve the Euler equations in 1997[2] and then developed BR1 scheme (the first Bassi-Rebay scheme) and BR2 scheme (the second Bassi-Rebay scheme) for solving the Navier-Stokes (N-S) equations[3, 4]. Almost at the same time, Oden et al. introduced a high-order DG scheme for the N-S equations without using any auxiliary variables[5]. In recent years, the LDG and CDG schemes were also developed for solving the N-S equations in Refs. [6, 7], respectively. Since the DG method is quite similar with the finite volume (FV) method, the numerical fluxes employed in FV method are also widely used in DG method, such as local Lax-Friedrichs (LLF), Roe etc.

As an alternative, the Boltzmann equation-based flux solvers are becoming more and more attractive, such as the kinetic flux vector splitting (KFVS) scheme[8] and the gas-kinetic Bhatnagar-Gross-Krook (BGK) scheme[9, 10]. The KFVS scheme usually cannot produce numerical results as accurate as those obtained by Roe[11] or AUSM[12] since the collision process is controlled by numerical time step. The gas-kinetic Bhatnagar-Gross-Krook (BGK) scheme handles the particle collisions using the BGK model and controls the dissipation in the streaming process by the collision time, which gives accurate solutions for both inviscid and viscous flows. However, the use of the Maxwellian function in KFVS and the gas-kinetic Bhatnagar-Gross-Krook (BGK) scheme increases the complexity and reduces the efficiency[8-10].

Recently, some efforts have been made to develop efficient lattice Boltzmann flux solver (LBFS) for compressible flows[13-16], where the inviscid flux at the cell interface is reconstructed

* Corresponding author, E-mail address: yongchengsun@126. com.

How to cite this article: Sun Yongcheng, Cai Junwei, Qin Wanglong. High-order discontinuous Galerkin solution of compressible flows with a hybrid lattice Boltzmann flux[J]. Trans. Nanjing Univ. Aero. Astro. , 2018,35(3):413-422.
http://dx. doi. org/10. 16356/j. 1005-1120. 2018. 03. 413

using the local solution of one-dimensional compressible lattice Boltzmann model. It is noted that the non-equilibrium part of the distribution function at cell interfaces introduces numerical dissipation, which is helpful for capturing shocks, but not expected in smooth regions, such as boundary layers.

A high-order DG method is employed here to solve both inviscid and viscous flows. Instead of using the conventional numerical inviscid fluxes, a hybrid LBFS is adopted, in which a switch function is introduced to control the numerical dissipation. Numerical results indicate that the hybrid LBFS combining with the high-order DG method can give accurate solutions for both inviscid and viscous flows.

1 DG Discretization of N-S Equations

The N-S equations in the conservation form can be written as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}_c(\boldsymbol{U}) + \nabla \cdot \boldsymbol{F}_v(\boldsymbol{U}, \nabla \boldsymbol{U}) = \boldsymbol{0} \tag{1}$$

where $\boldsymbol{U}$ are the conservation variables and $\boldsymbol{F}_c$, $\boldsymbol{F}_v$ are the inviscid and viscous flux functions, respectively.

After multiplying a test function V, integrating over the computational domain and performing an integration by parts, the following weak form is obtained

$$\frac{\partial}{\partial t}\int_{\Omega} V\boldsymbol{U}\mathrm{d}\Omega + \int_{\partial\Omega} V\boldsymbol{F}(\boldsymbol{U}, \nabla\boldsymbol{U}) \cdot \boldsymbol{n}\mathrm{d}\delta - \int_{\Omega} \nabla V \cdot \boldsymbol{F}(\boldsymbol{U}, \nabla\boldsymbol{U})\mathrm{d}\Omega = 0 \qquad \forall V \tag{2}$$

where $\boldsymbol{F}(\boldsymbol{U}, \nabla\boldsymbol{U}) = \boldsymbol{F}_c(\boldsymbol{U}) + \boldsymbol{F}_v(\boldsymbol{U}, \nabla\boldsymbol{U})$. By subdividing the computational domain Ω into the non-overlapping elements Ω_e, the semi-discrete system is written as

$$\frac{\partial}{\partial t}\int_{\Omega_e} V_\mathrm{h}\boldsymbol{U}_\mathrm{h}\mathrm{d}\Omega_e + \int_{\partial\Omega_e} V_\mathrm{h}\boldsymbol{F}(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h}) \cdot \boldsymbol{n}\mathrm{d}\delta - \int_{\Omega_e} \nabla V_\mathrm{h} \cdot \boldsymbol{F}(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h})\mathrm{d}\Omega_e = 0 \quad \text{for } V_\mathrm{h} \tag{3}$$

where $\boldsymbol{U}_\mathrm{h}$ and V_h are the high-order approximations of $\boldsymbol{U}$ and V

$$\boldsymbol{U}_\mathrm{h}(x, y, t) = \sum_{j=1}^{N(p)} \boldsymbol{u}_j(t)\phi_j(x, y) \tag{4}$$

$$V_\mathrm{h}(x, y, t) = \sum_{j=1}^{N(p)} v_j(t)\phi_j(x, y) \tag{5}$$

where $\phi_j(x, y)$ are the basis functions of degree p. Eq. (3) then becomes

$$\frac{\partial}{\partial t}\int_{\Omega_e} \phi_i\boldsymbol{U}_\mathrm{h}\mathrm{d}\Omega_e + \int_{\partial\Omega_e} \phi_i\boldsymbol{F}(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h}) \cdot \boldsymbol{n}\mathrm{d}\delta - \int_{\Omega_e} \nabla \phi_i \cdot \boldsymbol{F}(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h})\mathrm{d}\Omega_e = 0 \qquad 1 \leqslant i \leqslant N(p) \tag{6}$$

Here, the BR2 scheme[4] is employed, which introduces the following "face" contributions defined as

$$\int_{\Omega_e} \phi_i\boldsymbol{r}_{\partial\Omega_e}\mathrm{d}\Omega_e = \int_{\partial\Omega_e} \phi_i(\boldsymbol{U}_0 - \boldsymbol{U})\boldsymbol{n}\mathrm{d}\delta \tag{7}$$

where $\boldsymbol{U}_0 = \frac{1}{2}(\boldsymbol{U} + \boldsymbol{U}^+)$ and $\boldsymbol{U}^+$ is the state of the neighboring element. Replacing the flux function $\boldsymbol{F}(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h}) \cdot \boldsymbol{n}$ with a numerical flux function, Eq. (6) becomes

$$\frac{\partial}{\partial t}\int_{\Omega_e} \phi_i\boldsymbol{U}_\mathrm{h}\mathrm{d}\Omega_e + \int_{\partial\Omega_e} \phi_i\boldsymbol{H}(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h} + \boldsymbol{r}_{\partial\Omega_e}, \boldsymbol{U}_\mathrm{h}^+, \nabla\boldsymbol{U}_\mathrm{h}^+ + \boldsymbol{r}_{\partial\Omega_e}^+, \boldsymbol{n})\mathrm{d}\delta - \int_{\Omega_e} \nabla \phi_i \cdot \boldsymbol{F}(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h} + \boldsymbol{r}_{\partial\Omega_e})\mathrm{d}\Omega_e = 0 \qquad 1 \leqslant i \leqslant N(p) \tag{8}$$

There are volume and surface integrations in Eq. (8). In order to efficiently solve this equation, Gauss numerical integration method is adopted in this work. The numerical flux $\boldsymbol{H}$ includes the inviscid part $\boldsymbol{H}_c(\boldsymbol{U}_\mathrm{h}, \boldsymbol{U}_\mathrm{h}^+, \boldsymbol{n})$ and the viscous part $\boldsymbol{H}_v(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h} + \boldsymbol{r}_{\partial\Omega_e}, \boldsymbol{U}_\mathrm{h}^+, \nabla\boldsymbol{U}_\mathrm{h}^+ + \boldsymbol{r}_{\partial\Omega_e}^+, \boldsymbol{n})$. The viscous part can be evaluated by averaging the left and right side of the interface

$$\boldsymbol{H}_v(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h} + \boldsymbol{r}_{\partial\Omega_e}, \boldsymbol{U}_\mathrm{h}^+, \nabla\boldsymbol{U}_\mathrm{h}^+ + \boldsymbol{r}_{\partial\Omega_e}^+, \boldsymbol{n}) = \frac{1}{2}[\boldsymbol{F}_v(\boldsymbol{U}_\mathrm{h}, \nabla\boldsymbol{U}_\mathrm{h} + \boldsymbol{r}_{\partial\Omega_e}) + \boldsymbol{F}_v(\boldsymbol{U}_\mathrm{h}^+, \nabla\boldsymbol{U}_\mathrm{h}^+ + \boldsymbol{r}_{\partial\Omega_e}^+)] \cdot \boldsymbol{n} \tag{9}$$

As to the inviscid numerical flux, instead of using the widely used LLF flux, Roe flux, AUSM flux, etc., an LBFS is employed in this work.

2 Hybrid Lattice Boltzmann Flux Solver

In the hybrid lattice Boltzmann flux solver, the inviscid flux is evaluated by local reconstruction of solution using one-dimensional lattice Boltzmann model, while the viscous flux is still approximated by conventional smooth function approximation. Thus, in this work, we only consider the evaluation of inviscid flux at the cell interface. For the sake of derivation, the detailed expression of inviscid flux is given first. For two-dimensional case, the inviscid flux in Eq. (7) can be written as

$$\boldsymbol{H}_c=\begin{bmatrix}\rho\boldsymbol{U}_{\mathrm{h}}\\(\rho U_nU_n+p)n_x-\rho U_nU_\tau n_y\\(\rho U_nU_n+p)n_y-\rho U_nU_\tau n_x\\(\rho E+p)U_n\end{bmatrix}\tag{10}$$

where $\boldsymbol{n}=(n_x,n_y)$ denotes the unit normal vector on the control surface in the Cartesian coordinate system. ρ, U_n, U_τ and p present the density, normal velocity, tangential velocity and the pressure of the mean flow, respectively. E is the total energy defined as

$$E=e+\frac{1}{2}(U_n^2+U_\tau^2)\tag{11}$$

where $e=p/[(\gamma-1)\rho]$ is the potential energy of the mean flow.

In the present work, the inviscid flux $\boldsymbol{H}_c$ is computed by LBFS. It is known that the conservative variables and fluxes at the cell interfaces can be expressed as the summation of lattice velocity, moments and distribution function according to the conservation forms of moments[15]. Usually, the distribution function at the cell interface consists of equilibrium part and non-equilibrium part. From the Chapman-Enskog analysis and applying Taylor series expansion in time and physical space, the non-equilibrium part of the distribution function can be approximated by the difference of equilibrium distribution functions at the cell interface and its surrounding point[17, 18]. Finally, the inviscid flux at the cell interface can be written as

$$\begin{gathered}\boldsymbol{H}_c=\sum_{i=1}^{4}\xi_i\boldsymbol{\varphi}_a g_i(0,t)+\\ \tau_0\Big[\sum_{i=1}^{4}\xi_i\boldsymbol{\varphi}_a g_i(-\xi_i\delta t,t-\delta t)-\sum_{i=1}^{4}\xi_i\boldsymbol{\varphi}_a g_i(0,t)\Big]=\\ \boldsymbol{H}_c^{\mathrm{I}}+\tau_0[\boldsymbol{H}_c^{\mathrm{II}}-\boldsymbol{H}_c^{\mathrm{I}}]\end{gathered}\tag{12}$$

where τ_0 is the dimensionless collision time, δt the streaming time step, ξ_i the particle velocity in i-direction, and $g_i(0,t)$, and $g_i(-\xi_i\delta t,t-\delta t)$ are the equilibrium distribution functions at the cell interface and its surrounding points. A non-free parameter D1Q4 model, proposed by Yang et al.[14, 15], is adopted in Eq. (14). $\boldsymbol{\varphi}_a$ stands for the moments, which can be written as

$$\boldsymbol{\varphi}_a=\left(1,\xi_i,\frac{1}{2}\xi^2+e_p\right)^{\mathrm{T}}\tag{13}$$

where $e_p=\left[1-\dfrac{D}{2}(\gamma-1)\right]e$ is the potential energy of particles and D the abbreviation of dimensional lattice Boltzmann model ($D=1$ for D1Q4 model). From the Chapman-Enskog analysis[19], we know that the first term on the R. H. S. of Eq. (14) contributes to the inviscid flux while the second term on the R. H. S. of Eq. (14) is the numerical dissipation. τ_0 can be viewed as the weight of the numerical dissipation in the present work. By introducing a switch function to control the value of τ_0, we can achieve the goal of setting the numerical dissipation to be zero in smooth region while the maximum value in the vicinity of strong shock wave. The details of the evaluation for $\boldsymbol{H}_c^{\mathrm{I}}$, $\boldsymbol{H}_c^{\mathrm{II}}$ and τ_0 in Eq. (14) will be shown below.

2.1 Evaluation of $\boldsymbol{H}_c^{\mathrm{I}}$

$\boldsymbol{H}_c^{\mathrm{I}}$ is the flux attributed to the equilibrium distribution function at the cell interface $g_i(0,t)$. To obtain $g_i(0,t)$, the conservative variables at the cell interface should be computed in advance. According to the compatibility condition and the conservation forms of moments, the density, momentum in the normal direction and energy attributed to normal velocity can be computed by

$$\boldsymbol{U}_h^*=\left[\rho,\rho h_0U_n,\frac{1}{2}\rho U_n^2+\rho e\right]^{\mathrm{T}}=\sum_{i=1}^{4}\xi_i\boldsymbol{\varphi}_a g_i(-\xi_i\delta t,t-\delta t)\tag{14}$$

In Eq. (16), the key issue is to evaluate the equilibrium distribution function $g_i(-\xi_i\delta t,t-\delta t)$. Like the conventional upwind schemes, it is assumed that a local Riemann problem is formed at the cell interface. Thus, the equilibrium distribution function $g_i(-\xi_i\delta t,t-\delta t)$ can be evaluated according to the location of $-\xi_i\delta t$. The non-free

D1Q4 model is applied in this paper and $g_i(-\xi_i\delta t, t-\delta t)$ can be specified as

$$g_i(-\xi_i\delta t, t-\delta t)=\begin{cases} g_i^{\mathrm{L}} & i=1,3 \\ g_i^{\mathrm{R}} & i=2,4 \end{cases} \tag{15}$$

where g_i^{L} and g_i^{R} are the equilibrium distribution functions at the left and right sides of the cell interface, which can be calculated from the corresponding conservative variables $\boldsymbol{U}_{\mathrm{h}}$ and $\boldsymbol{U}_{\mathrm{h}}^{+}$. The process is illustrated in Fig. 1 and the details can be found in Ref. [15]. Substituting Eq. (17) into Eq. (16), we can obtain

$$\boldsymbol{U}_{\mathrm{h}}^{*}=\sum_{i=1,3}\varphi_a g_i^{\mathrm{L}}+\sum_{i=2,4}\varphi_a g_i^{\mathrm{R}} \tag{16}$$

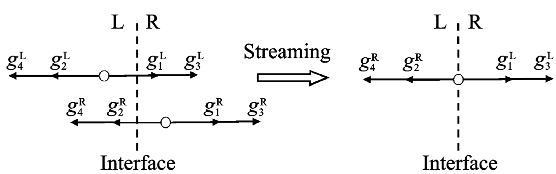


Fig. 1 Streaming process of non-free parameter D1Q4 model at the cell interface

In Eq. (18), only the conservative variables, which attributed to the normal velocity, are obtained. To evaluate the tangential velocity at the cell interface, one of the feasible ways can be expressed as

$$(\rho U_\tau)^{*}=\sum_{i=1}^{4} g_i\cdot U_\tau^{*}=\sum_{i=1,3} g_i^{\mathrm{L}}\cdot U_\tau^{\mathrm{L}}+\sum_{i=2,4} g_i^{\mathrm{R}}\cdot U_\tau^{\mathrm{R}} \tag{17}$$

where U_τ, U_τ^{L} and U_τ^{R} are the tangential velocities at the cell interface, the left and right side of cell interface, respectively. With Eqs. (18), (19), the primitive variables ρ^{*}, U_n^{*}, U_τ^{*}, p^{*} can be obtained in a straightforward way. By substituting the above primitive variables directly into Eq. (4), the inviscid flux at the cell interface $\boldsymbol{H}_c^{\mathrm{I}}$ can be expressed as

$$\boldsymbol{H}_c=\begin{bmatrix} \rho U_n \\ (\rho U_n U_n+p)n_x-\rho U_n U_\tau n_y \\ (\rho U_n U_n+p)n_y-\rho U_n U_\tau n_x \\ (\rho E+p)U_n \end{bmatrix} \tag{18}$$

2.2 Evaluation of $\boldsymbol{H}_c^{\mathrm{II}}$

$\boldsymbol{H}_c^{\mathrm{II}}$ is the flux attributed to the equilibrium distribution function at the surrounding points of the cell interface $g_i(-\xi_i\delta t, t-\delta t)$. Since $g_i(-\xi_i\delta t, t-\delta t)$ has been determined by Eq. (17), $\boldsymbol{H}_c^{\mathrm{II}}$ can be computed directly for this case. The mass flux, the momentum flux in the normal direction and the energy flux attributed to the normal velocity across the interface can be written as

$$\boldsymbol{H}_c^{\mathrm{II}}=\left[\rho U_n, \rho U_n^2+p\left(\rho\left(\frac{1}{2}U_n^2+e\right)+p\right)U_n\right]^{\mathrm{T}}=\sum_{i=1,3}\xi_i\varphi_a g_i^{\mathrm{L}}+\sum_{i=2,4}\xi_i\varphi_a g_i^{\mathrm{R}} \tag{19}$$

In addition, the contribution of the tangential velocity to the momentum flux in the tangential direction and energy flux can be approximated by

$$(\rho U_n U_\tau)^{*}=\sum_{i=1}^{4}\xi_i g_i(-\xi_i\delta t, t-\delta t)\cdot U_\tau^{*}=\sum_{i=1,3}\xi_i g_i^{\mathrm{L}}\cdot U_\tau^{\mathrm{L}}+\sum_{i=2,4}\xi_i g_i^{\mathrm{R}}\cdot U_\tau^{\mathrm{R}} \tag{20}$$

$$(\rho U_n U_\tau^2)^{*}=\sum_{i=1}^{4}\xi_i g_i(-\xi_i\delta t, t-\delta t)\cdot (U_\tau^{*})^2=\sum_{i=1,3}\xi_i g_i^{\mathrm{L}}\cdot (U_\tau^{\mathrm{L}})^2+\sum_{i=2,4}\xi_i g_i^{\mathrm{R}}\cdot (U_\tau^{\mathrm{R}})^2 \tag{21}$$

By collecting Eqs. (21)—(23), the full expression of $\boldsymbol{H}_c^{\mathrm{II}}$ can be obtained

$$\boldsymbol{H}_c^{\mathrm{II}}=\begin{bmatrix} \sum_{i=1}^{4}\xi_i g_i \\ \sum_{i=1}^{4}\xi_i^2 g_i n_x-\sum_{i=1}^{4}\xi_i g_i\cdot U_\tau^{*} n_y \\ \sum_{i=1}^{4}\xi_i^2 g_i n_y+\sum_{i=1}^{4}\xi_i g_i\cdot U_\tau^{*} n_x \\ \sum_{i=1}^{4}\xi_i\left(\frac{1}{2}\xi_i^2+e_p\right)g_i+\frac{1}{2}\sum_{i=1}^{4}\xi_i g_i (U_\tau^{*})^2 \end{bmatrix} \tag{22}$$

where $g_i=g_i(-\xi_i\delta t, t-\delta t)$.

2.3 Evaluation of τ_0

In the hybrid LBFS, τ_0 is controlled by an introduced switch function. In the smooth region such as in boundary layer, the switch function takes a value close to zero. However, around the shock wave, it tends to be one. That is, in hybrid LBFS, the numerical dissipation is almost zero in smooth region to simulate accurately the thin boundary layer, and a relatively large numerical dissipation near the strong shock wave is contained to suppress the instability of shock wave. The switch function is defined as

$$\tau_0=\tanh\left(C\frac{|\rho_{\mathrm{L}}-\rho_{\mathrm{R}}|}{\rho_{\mathrm{L}}+\rho_{\mathrm{R}}}\right) \tag{23}$$

where tanh(x) is the hyperbolic tangent function, and ρ_L, ρ_R the densities at the left and right sides of the cell interface. $|\rho_L - \rho_R|$ is the local discontinuity over the cell interface and C is the amplification factor, which is set as $C=10$ in this work.

3 Numerical Results

A few benchmark problems are tested with the hybrid LBFS introduced above, including the Sod shock tube, the inviscid flow around a cylinder, the inviscid flow with shocks around the NACA0012 airfoil and the laminar flow around the NACA0012 airfoil.

3.1 Sod shock tube problem

The shock tube problem is a particularly interesting and difficult test case, since it presents an exact solution to the full system of one-dimensional Euler equations containing simultaneously a shock wave, a contact discontinuity and expansion fan. This test case is chosen to demonstrate the flexibility of the LBFS combining with DG method in capturing the shock. The initial value of this problem is set as

$$\begin{cases} (\rho_L, u_L, p_L) = (1, 0, 1) & -0.5 < x < 0 \\ (\rho_R, u_R, p_R) = (0.125, 0, 0.1) & 0 < x < 0.5 \end{cases} \tag{24}$$

A mesh size of 250 is used for simulation and a third order ($p=2$) DG method is adopted. Fig. 2 shows the computed density profile obtained by the LBFS scheme (red solid line) at $t=0.22$. Exact solution (black solid line), the results of Lax-Friedrichs (LF) scheme (green dotted line), HLLC scheme (blue dashed line), AUSMDV scheme (cyan dotted line) are also displayed in Fig. 2. It is observed that the performance of the schemes in capturing the shock is basically the same. However, in the vicinity of the expansion wave, the result of LBFS is the same with that of the AUSMDV scheme, which is steeper than the results of both LF scheme and HLLC scheme, demonstrating a more accurate scheme in dealing with complex flows. Fig. 3 gives the computed profiles of pressure and velocity with different schemes. The results of LBFS match well with those of other schemes and the exact solutions, which demonstrate the capacity of the hybrid lattice Boltzmann flux solver.

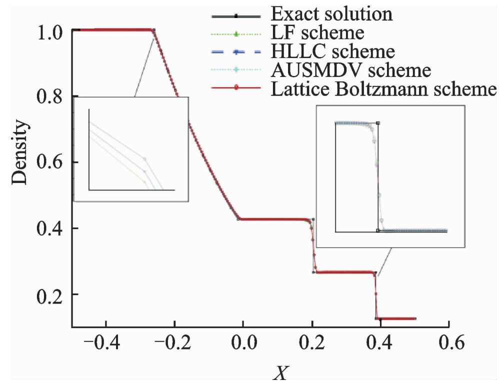


Fig. 2 Density profile for the 1D Sod shock tube

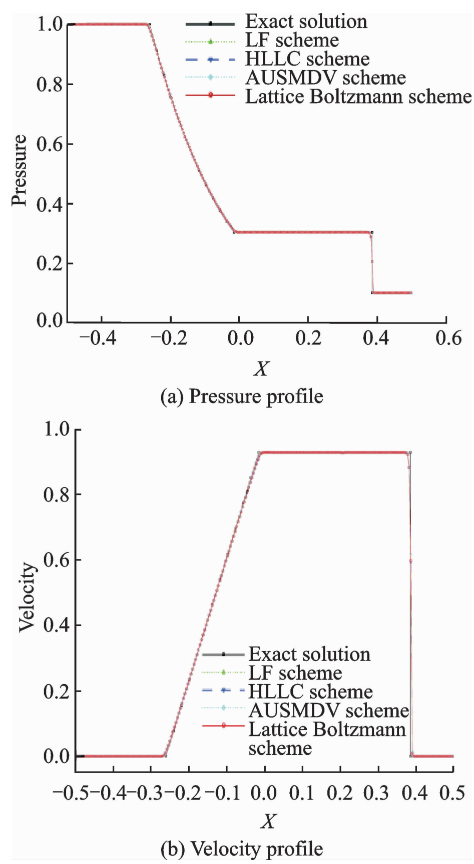
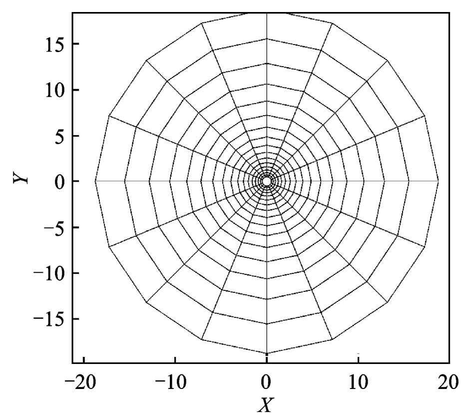


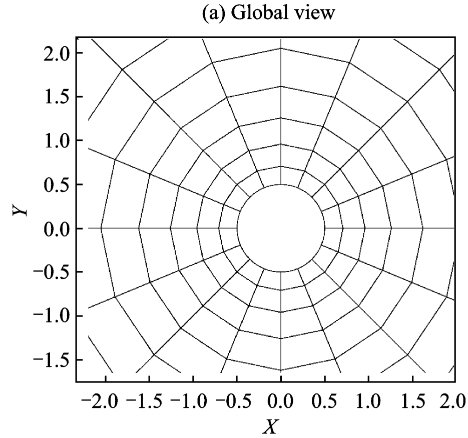
Fig. 3 Pressure and velocity profiles for 1D Sod shock tube

3.2 Inviscid flow around a cylinder

The case of the inviscid flow around a cylinder is used to validate the accuracy of the DG-LBFS scheme. A structured C-grid (16×16) is used in the entire computational domain (Fig. 4), which is very coarse. The obtained Mach number isolines with various orders ranging from 1 to 4 are given in Fig. 5, where the Mach number isolines become smoother with increasing order. The numerical solutions are very close to the analytical solution when $p \geqslant 3$. Fig. 6 displays the distributions of the pressure coefficient C_p. It can be seen that the numerical solution becomes more and more accurate with increasing order and no significant difference can be observed when the order $p \geqslant 2$, which demonstrates the great capacity of the DG-LBFS scheme.

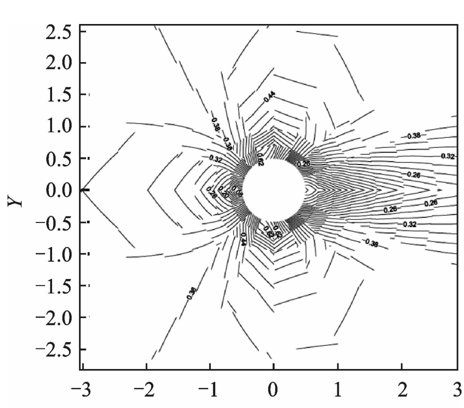


(a) Global view

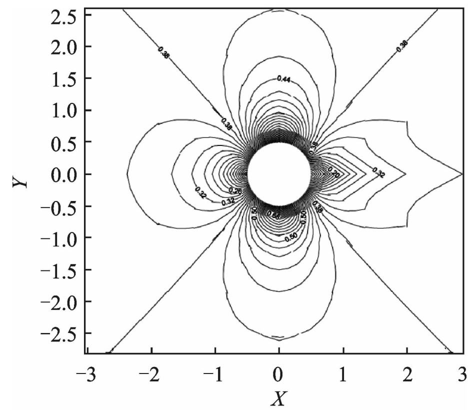


(b) Local view

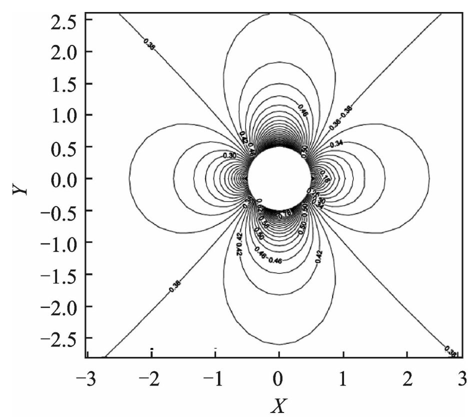
Fig. 4 Mesh for inviscid flow around a 2D cylinder



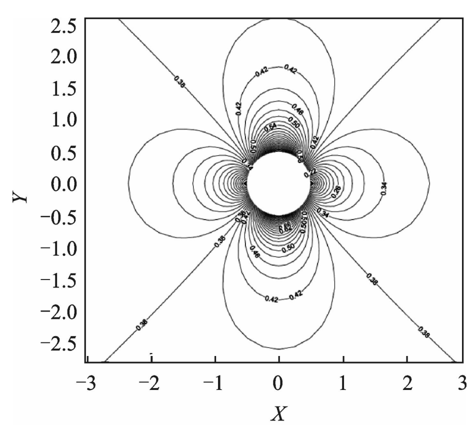
(a) p=1



(b) p=2



(c) p=3



(d) p=4

Fig. 5 Mach number contours obtained using different orders

3.3 Transonic flow past NACA0012 airfoil

A transonic flow around the NACA0012 airfoil case is used to test the ability of the DG-LBFS method in capturing the shock. The Mach number of the flow condition is 0.85 and the attack angle is 1.25°.

Fig. 7 gives the global view and the local view of the computational mesh, which contains 900 elements in total. Fig. 8 demonstrates the Mach

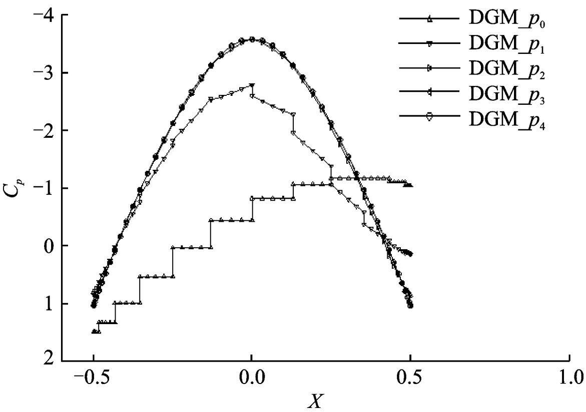


Fig. 6 The distribution of C_p

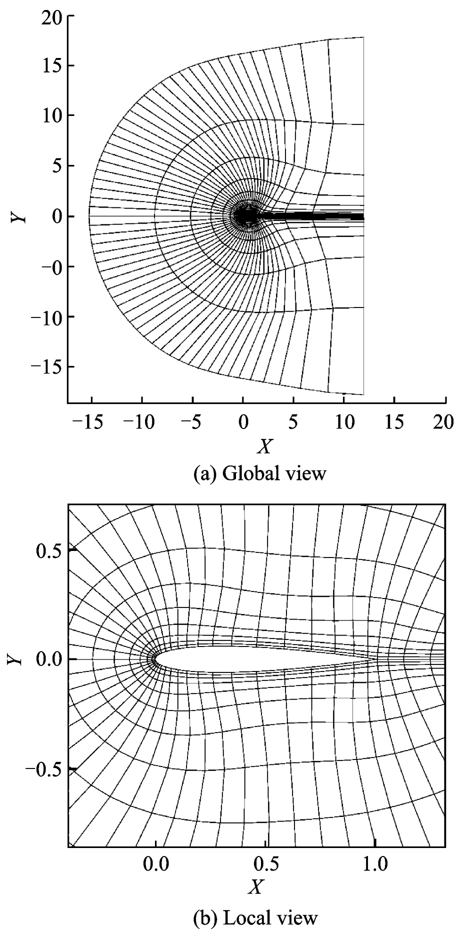


Fig. 7 Computational mesh for transonic flow past NACA0012 airfoil

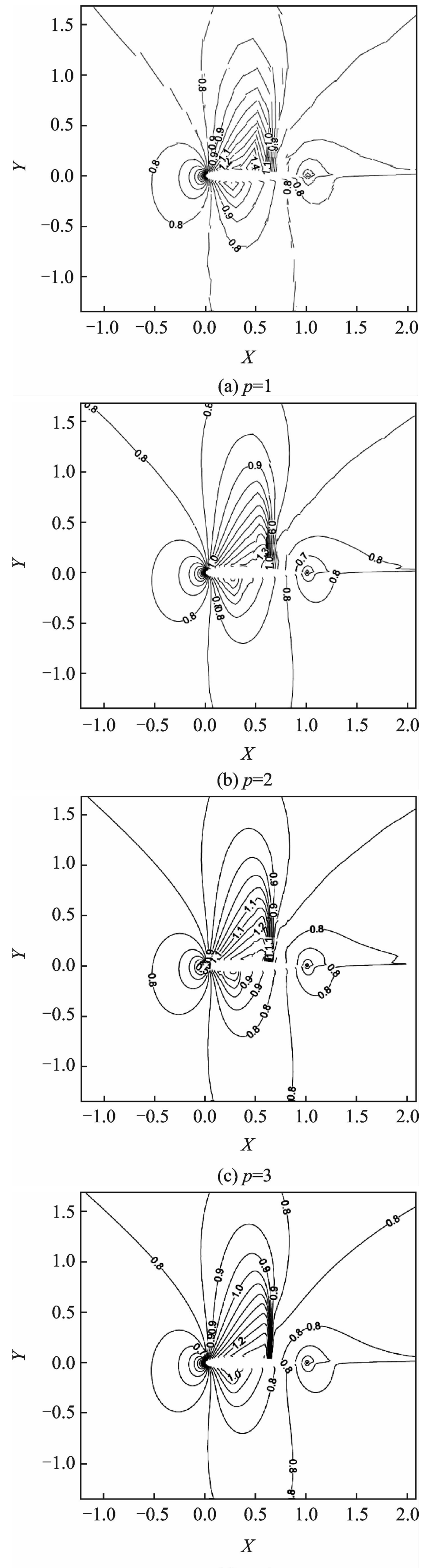


Fig. 8 Mach number isolines obtained using different orders

number isolines obtained with different orders. It can be observed that the accuracy becomes higher with increasing orders. The computed pressure coefficient C_p ($p=4$) is shown in Fig. 9, which depicts a good match compared with the result of Ref. [20]. Fig. 10 illustrates the logarithmic density residual versus time step using an implicit method. Converged results can be obtained within

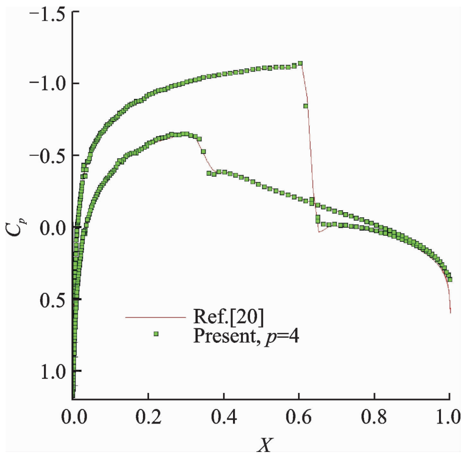


Fig. 9 C_p distribution for transonic flow past NACA012 airfoil ($p=4$)

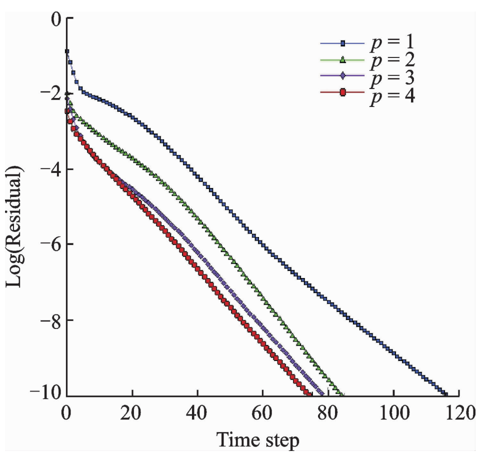


Fig. 10 Logarithmic density residual versus time step ($p=1$—4) for the transonic flow past NACA012 airfoil

several time steps for different orders. Table 1 gives the computed lift coefficient C_l and drag coefficient C_d using DG-LBFS method with different orders. It can be seen that although a coarse grid is used in this case, the results of the DG-LBFS method agree well with that of the finite volume method[20].

Table 1 Results of C_l and C_d

Method	C_l	C_d
Ref. [20]	0.354 95	0.022 55
DGM_p_1	0.325 9	0.025 4
DGM_p_2	0.348 0	0.023 6
DGM_p_3	0.348 5	0.023 0
DGM_p_4	0.347 3	0.022 6

3.4 Laminar flow past NACA0012 airfoil

A laminar flow around the NACA0012 airfoil is chosen to test the ability of the DG-LBFS method in capturing the thin boundary layer. The free-stream flow condition is given as $Ma_\infty = 0.5$ and $Re_\infty = 5\ 000$ with an angle of attack $\alpha = 0°$. A coarse mesh with 476 quadrilateral elements is used for this test case as in Fig. 11. The Mach number isolines of order $p = 4$ is displayed in Fig. 12, in which a small recirculation bubble caused by the separation of the flow can be seen in the near-wake region of the airfoil. The results of the pressure coefficient C_p and the friction coefficient C_f are given in Fig. 13, which concur with the results of the Godunov flux function[3]. The computed lift coefficient C_l and drag coefficient C_d of different orders using DG-LBFS method are listed in Table 2, which agree well with the results of Refs. [3,21], showing the capacity of the DG-LBFS method in solving the viscous flow problems.

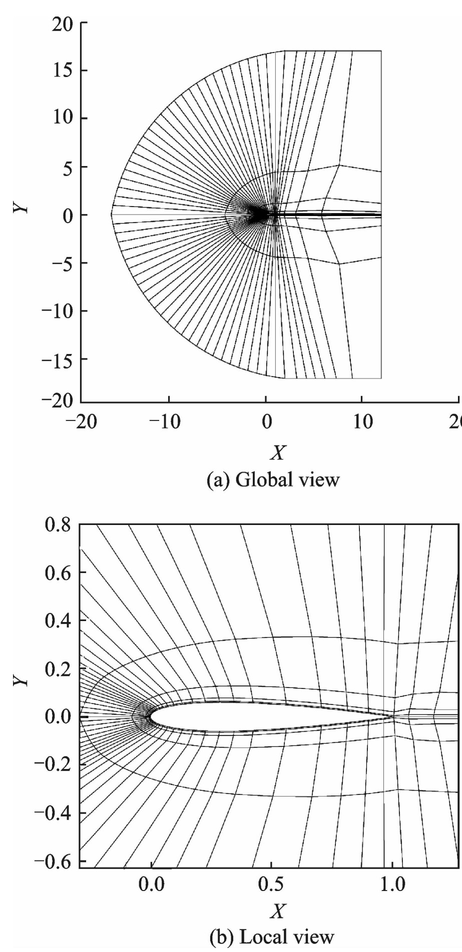


Fig. 11 Computing mesh used for simulation of laminar flow past NACA0012 airfoil

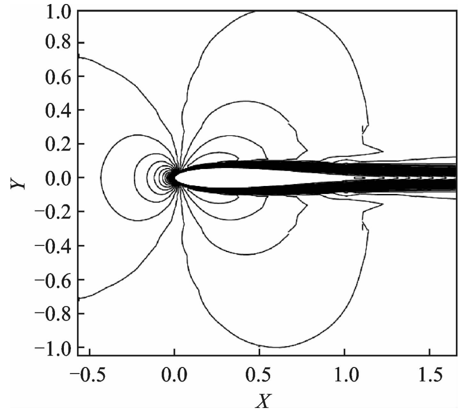


Fig. 12 Computed Mach number isolines of laminar flow past NACA0012 airfoil ($p=4$)

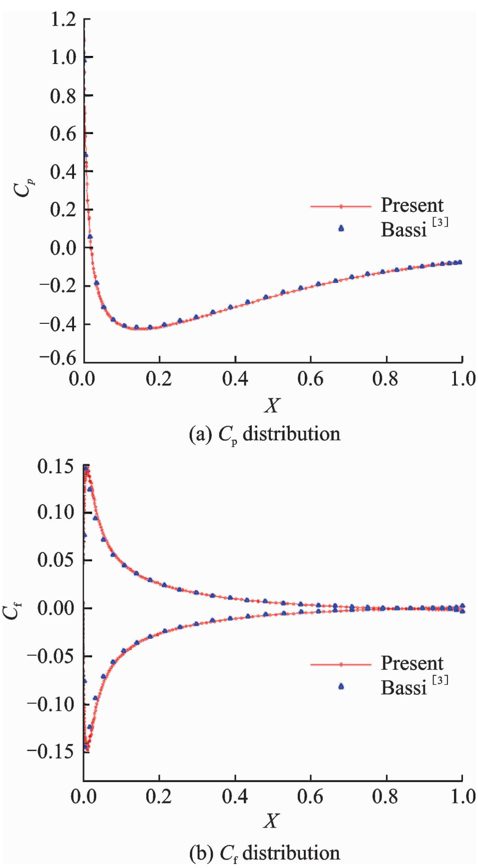


Fig. 13 Computed C_p and C_f for laminar flow past NACA0012 airfoil

Table 2 Comparison of C_l and C_d

Method	$C_{l,total}$	$C_{d,total}$
Ref. [20]	0.0	0.055 57
Ref. [3]	0.0	0.055 11
DGM_p_1	0.0	0.066 20
DGM_p_2	0.0	0.054 90
DGM_p_3	0.0	0.053 50
DGM_p_4	0.0	0.053 30

4 Conclusions

The high-order DG method combining with a hybrid lattice Boltzmann flux solver is employed to solve the compressible Euler equations and Navier-Stokes equations. A switch function ranging from 0 to 1 is introduced to control the numerical dissipation when solving the inviscid numerical flux, making the scheme accurate for simulating both smooth flows and the flows with shocks. Numerical results of some benchmark problems indicate that accurate solutions can be obtained even on very coarse grids using the introduced numerical solver.

References:

[1] COCKBURN B, KARNIADAKIS G E, SHU C W. Discontinuous galerkin methods: Theory, computation and applications[M]. Berlin:Springer, 1999.

[2] BASSI F, REBAY S. High-order accurate discontinuous finite element solution of the 2D Euler equations [J]. Journal of Computational Physics, 1997, 138: 251-285.

[3] BASSI F, REBAY S R. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations [J]. Journal of Computational Physics, 1997, 131: 267-279.

[4] BASSI F, CRIVELLINI A, REBAY S. Discontinuous Galerkin solution of the Reynolds-averaged Navier-Stokes and k-ω, turbulence model equations[J]. Computers & Fluids, 2005, 34(2): 507-540.

[5] BAUMANN C E, ODEN J T. A discontinuous hp finite element method for the Euler and Navier-Stokes equations[J]. International Journal for Numerical Methods in Fluids, 1999, 31: 79-95.

[6] COCKBURN B, SHU C W. The local discontinuous Galerkin method for time-dependent convection-diffusion system[J]. SIAM Journal on Numerical Analysis, 1998, 35(6):2440-2463.

[7] PERAIRE J, PERSSON P O. The compact discontinuous Galerkin (CDG) method for elliptic problems [J]. SIAM Journal on Scientific Computing, 2008, 30(4): 1806-1824.

[8] CHOU S Y, BAGANOFF D. Kinetic flux-vector splitting for the Navier-Stokes equations[J]. Journal of Computational Physics, 1997, 130: 217-230.

[9] CHAE D, KIM C, RHO O H. Development of an improved gas-kinetic BGK scheme for inviscid and

viscous flows[J]. Journal of Computational Physics, 2000, 158: 1-27.

[10] XU K. A gas-kinetic BGK scheme for the Navier-Stokes equations and its connection with artificial dissipation and Godunov method[J]. Journal of Computational Physics, 2001, 171: 289-335.

[11] ROE P L. Approximate Riemann solvers, parameter vectors, and difference schemes[J]. Journal of Computational Physics, 1981, 43: 357-372.

[12] LIOU M S, STE_EN C J. A new flux splitting scheme[J]. Journal of Computational Physics, 1993, 107: 23-39.

[13] JI C Z, SHU C, ZHAO N. A Lattice Boltzmann method-based flux solver and its application to solve shock tube problem[J]. Mod Phys Lett B, 2009, 23: 313-316.

[14] YANG L M, SHU C, WU J. Development and comparative studies of three non-free parameter lattice Boltzmann models for simulation of compressible flows[J]. Adv Appl Math Mech, 2012, 4: 454-472.

[15] YANG L M, SHU C, WU J. A moment conservation-based non-free parameter compressible lattice Boltzmann model and its application for flux evaluation at cell interface[J]. Computers & Fluids, 2013, 79: 190-199.

[16] SHU C, YANG Y, YANG L M, et al. Lattice Boltzmann solver: An efficient approach for numerical simulation of fluid flows[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2014, 31: 1-15.

[17] WANG Y, SHU C, TEO C J. Thermal lattice Boltzmann flux solver and its application for simulation of incompressible thermal flows[J]. Computers & Fluids, 2014, 94: 98-111.

[18] WANG L M, SHU C, WU J. A simple distribution function-based gas-kinetic scheme for simulation of viscous incompressible and compressible flows[J]. Journal of Computational Physics, 2014, 274: 611-632.

[19] GUO Z L, SHU C. Lattice Boltzmann method and its applictions in engineering[M]. World Scientific Publishing, 2013.

[20] ECONOMON T D, PALACIOS F, COPELAND S R, et al. SU2: An open-source suite for multiphysics simulation and design[J]. AIAA Journal, 2016, 54(3): 828-846.

[21] JAWAHAR P, KAMATH H. A high-resolution procedure for Euler and Navier-Stokes computations on unstructured grids[J]. Journal of Computational Physics, 2000, 164: 165-203.

Dr. **Sun Yongcheng** received his Ph. D. degree in control science and engineering from Nanjing University of Science and Technology, Nanjing, China, in 2005. From then on, he has been working in Department of Airforce, the 28th Research Institute of China Electronics Technology Group Corporation. His current research interests include numerical simulations and C4ISR system.

Dr. **Cai Junwei** received his Ph. D. degree in automation engineering from Nanjing University of Aeronautics and Astronautics in 2009. His current research interests include numerical simulations, robust control methods and C4ISR system.

Dr. **Qin Wanglong** received his B. S. and Ph. D degrees in aerospace engineering and fluid dynamics from Nanjing University of Aeronautics and Astronautics in 2011 and 2016, respectively. In April 2017, he joined the Department of Airforce, the 28th Research Institute of China Electronics Technology Group Corporation.

(Production Editor: Zhang Tong)