Gas kinetic flux solver based high-order finite-volume method for simulation of two-dimensional compressible flows

L. M. Yang [1,2] C. Shu,[2,*] Z. Chen [2] Y. Y. Liu [2] J. Wu,[1] and X. Shen [3]

[1]*Department of Aerodynamics, College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics, Yudao Street, Nanjing 210016, Jiangsu, China*

[2]*Department of Mechanical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260*

[3]*Department of Mechanical and Construction Engineering, Northumbria University, Newcastle upon Tyne NE1 8ST, United Kingdom*

(Received 11 April 2021; accepted 16 June 2021; published 9 July 2021)

In this work, a high-order gas kinetic flux solver (GKFS) is developed for simulation of two-dimensional (2D) compressible flows. Different from the conventional gas kinetic scheme, which uses the local integral solution to the Boltzmann equation to reconstruct the numerical fluxes of macroscopic governing equations, the GKFS evaluates the numerical fluxes by the local asymptotic solution to the Boltzmann equation. This local asymptotic solution consists of the equilibrium distribution function and its substantial derivative at the cell interface. To achieve high-order accuracy in the simulation, the substantial derivative is discretized by a difference scheme with second-order accuracy in time and fourth-order accuracy in space, which results in a polynomial of the equilibrium distribution function at different locations and time levels. The Taylor series expansion is then introduced to simplify this polynomial. As a result, a simple high-order accurate local asymptotic solution to the Boltzmann equation is obtained and the numerical fluxes of macroscopic governing equations are given explicitly. A series of numerical examples are presented to validate the accuracy and capability of the developed high-order GKFS. Numerical results demonstrate that the high-order GKFS can achieve the desired accuracy on both the quadrilateral mesh and the triangular mesh and it outperforms the second-order counterpart.

DOI: 10.1103/PhysRevE.104.015305

I. INTRODUCTION

Based on kinetic theory, the gas kinetic scheme (GKS) or gas kinetic flux solver (GKFS) has emerged as one of the powerful tools for solving the Euler and/or Navier-Stokes equations (macroscopic governing equations) [1–6]. In this method, the local solution to the Boltzmann equation at the cell interface is utilized to calculate the numerical fluxes of macroscopic governing equations. Since it is reconstructed directly from the physical equation, the GKS enjoys the advantages of robustness, positively preserving and satisfying the entropy condition spontaneously [7,8]. In addition, different from the conventional Navier-Stokes solvers which deal with the inviscid flux and the viscous flux separately, the GKS calculates these fluxes in a uniform way. Due to these unique features, the GKS has been applied to various fields, such as incompressible flow [9–11], compressible flow [12–14], turbulence flow [15,16], multiphase flow [17,18], and chemically reacting flow [19,20].

The commonly used GKS usually reconstructs the numerical fluxes of macroscopic governing equations by the local integral solution to the Boltzmann equation with the Bhatnagar-Gross-Krook (BGK) collision model [2,6–9,12–17]. This local solution consists of the equilibrium distribution function across the cell interface and the initial distribution function at the surrounding points of the cell interface, which, respectively, represent the hydrodynamic scale physics and the kinetic scale physics. The contribution of each part is determined by the ratio of time step size used in solution reconstruction and local collision timescale. Under the continuum assumption, the initial distribution function can be taken as the first-order truncation in terms of the collision timescale to the Chapman-Enskog expansion of the Boltzmann equation. As a result, the numerical fluxes of macroscopic governing equations can be expressed as the function of conservative variables and their derivatives at the left and right sides of the cell interface in principle. Since the physical model used in solution reconstruction is consistent with the macroscopic governing equations, the conventional GKS can produce a stable and crisp shock transition in the discontinuous region while maintaining accurate Navier-Stokes solutions in the smooth region. However, due to its tedious flux reconstruction process, which involves many terms and coefficients associated with the conservative variables and their derivatives, the numerical fluxes of the Navier-Stokes equations are hard to give explicitly for the conventional GKS in the literature [21,22]. Accordingly, it is usually more complicated and less efficient than the conventional Navier-Stokes solvers [23–25]. Recently, the high-order GKS has been developed based on the framework of compact finite-volume discretization and weighted least-square reconstruction [26], two-stage fourth-order temporal discretization and Hermite weighted essentially nonoscillatory (HWENO) reconstruction [27], discontinuous Galerkin [28], etc. To achieve high-order accuracy in both space and time for the local integral solution

*Corresponding author: mpeshuc@nus.edu.sg

to the Boltzmann equation, high-order approximation is needed for calculation of the equilibrium distribution function as well as the initial distribution function [26–30]. Therefore it can be imagined that the full formulations for the high-order GKS would be far more complex than the second-order counterpart.

Different from the above conventional GKS, the GKFS reconstructs the numerical fluxes of macroscopic governing equations by the local asymptotic solution to the Boltzmann equation with the BGK collision model [31–33]. This asymptotic solution is actually the first-order truncation in terms of the collision timescale to the Chapman-Enskog expansion of the Boltzmann equation, which consists of the equilibrium distribution function and its substantial derivative at the cell interface. In the second-order GKFS [31–33], the substantial derivative is discretized by a second-order difference scheme, resulting in a second-order flux solver in both space and time. Since the local asymptotic solution to the Boltzmann equation is simpler than the local integral solution to the Boltzmann equation, the expressions of reconstructed numerical fluxes given by the GKFS are more concise than those of the conventional GKS [2,6–9,12–17], which facilitates implementation and improves computational efficiency accordingly. Given this, the second-order GKFS will be extended to the high-order counterpart in this work. At first, we discretize the substantial derivative of the equilibrium distribution function at the cell interface by a difference scheme with second-order accuracy in time and fourth-order accuracy in space, which results in a polynomial of the equilibrium distribution function at different locations and time levels. Second, we approximate these equilibrium distribution functions by a high-order Taylor series expansion. It can be found that the substantial derivative only involves the first-order derivative of the equilibrium distribution function at the left and right sides of the cell interface, which can be calculated by a high-order interpolation. In this way, a flux solver of the Navier-Stokes equations with second-order accuracy in time and fourth-order accuracy in space is obtained and its formulations can be given explicitly.

Apart from the flux solver, the high-order spatial discretization to the macroscopic governing equations is another important issue for the high-order scheme. The commonly used methods include high-order finite volume (FV) [34–36], high-order finite difference (FD) [37,38], discontinuous Galerkin (DG) [39,40], spectral volume (SV) [41], spectral difference (SD) [42], and the weighted essential nonoscillatory (WENO) scheme [43,44], etc. Among them, the high-order FV method can be extended from its second-order counterpart intuitively, which is widely used in practical engineering. In addition, the high-order FV method is easily applied to the unstructured mesh and inherits the excellently conservative property. Due to these good features, a recently developed high-order FV method, namely, the least-square based finite-difference finite-volume (LSFD FV) method [36,45], is chosen to discretize the macroscopic governing equations in this work. In the method, a Taylor series expansion based on the flow variables and their spatial derivatives at the cell center is introduced to approximate the distribution of the flow variables within each control volume. The spatial derivatives are calculated by the least-square based finite-difference (LSFD) scheme from the flow variables at the centers of the current cell and its neighboring cells. By solving the macroscopic governing equations, the flow variables at each cell center can be updated accordingly. When the high-order GKFS is implemented in this framework, the flow variables and their first-order derivatives at the left and right sides of the cell interface can be calculated directly by the introduced Taylor series expansion. The proposed high-order method is validated by its application to some test examples. Numerical results well demonstrate its high accuracy and efficiency.

The remaining parts of this paper are organized as follows. In Sec. II, the high-order GKFS is presented in detail. In Sec. III, the LSFD FV method is introduced briefly. Section IV validates the performance of the present method by several numerical examples. The last section is the Conclusion.

II. HIGH-ORDER GAS KINETIC FLUX SOLVER

A. Local asymptotic solution to Boltzmann equation

In GKFS, the local asymptotic solution to the Boltzmann equation is used to reconstruct the numerical fluxes of macroscopic governing equations. To do this, the connection between these two systems has to be established in advance. For simulation of compressible flows, the Navier-Stokes equations can be written as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{1}$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u} + p\mathbf{I}) = \nabla \cdot \mathbf{\Pi}, \tag{2}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot [(\rho E + p)\mathbf{u}] = \nabla \cdot (\kappa \nabla T) + \nabla \cdot (\mathbf{u} \cdot \mathbf{\Pi}), \tag{3}$$

where the stress tensor $\mathbf{\Pi}$ is defined by

$$\mathbf{\Pi} = \mu\left[(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \tfrac{2}{3}(\nabla \cdot \mathbf{u})\mathbf{I}\right].$$

Here, ρ, $\mathbf{u}$, p, T, and E are, respectively, the density, the velocity vector, the pressure, the temperature, and the total energy of the mean flow. μ and κ are the dynamic viscosity and the thermal conductivity, respectively. $\mathbf{I}$ is the unit tensor. $\mathbf{u} = (u, v)$ is for the two-dimensional (2D) case considered in this work, where u and v are the velocity components in the x and y directions, respectively. For simplicity, Eqs. (1)–(3) can be cast into a unified form as

$$\frac{\partial \mathbf{W}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{0}, \tag{4}$$

where the conservative variable vector $\mathbf{W}$ and flux vector $\mathbf{F}$ are given by

$$\mathbf{W} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} F_\rho \\ F_{\rho u} \\ F_{\rho v} \\ F_{\rho E} \end{bmatrix}. \tag{5}$$

The Boltzmann equation with the BGK collision model can be written as

$$\frac{\partial f}{\partial t} + \xi \cdot \nabla f = \frac{g - f}{\tau}, \tag{6}$$

where τ is the collision timescale. f and g are the distribution function and its equilibrium state, respectively. For compressible flows, the equilibrium state can be expressed by the Maxwellian distribution function as follows,

$$g = \rho\left(\frac{\lambda}{\pi}\right)^{\frac{2+K}{2}} e^{-\lambda[(\xi_x-u)^2+(\xi_y-v)^2+\sum_{j=1}^{K}\zeta_j^2]}, \tag{7}$$

where $\xi = (\xi_x, \xi_y)$ is the translational particle velocity in 2D space and $\zeta = (\zeta_1, \zeta_2, \ldots, \zeta_K)$ represents the rest of the components of the translational particle velocity in three-dimensional (3D) space and the rotational particle velocity, i.e., $K = 3-2+N$, where N denotes the number of rotational degrees of freedom. $\lambda = 1/(2RT)$ and R is the gas constant. The specific heat ratio γ is determined by

$$\gamma = \frac{b+2}{b} = \frac{K+4}{K+2}, \tag{8}$$

where b is the total number of degrees of freedom of molecules.

According to the Chapman-Enskog analysis [7,46], Eq. (4) can be derived from Eq. (6) by the following moment integration,

$$\frac{\partial\langle\boldsymbol{\varphi}_\alpha f\rangle}{\partial t} + \nabla\cdot\langle\xi\boldsymbol{\varphi}_\alpha f\rangle = \frac{\langle\boldsymbol{\varphi}_\alpha(g-f)\rangle}{\tau}, \tag{9}$$

with the first-order expansion to the distribution function in terms of the collision timescale, namely, the local asymptotic solution to the Boltzmann equation,

$$f = g - \tau Dg + O(\tau^2), \tag{10}$$

where $\langle\phi\rangle = \int_{-\infty}^{+\infty}\phi d\xi d\zeta$ defines the integration of ϕ over the particle velocity space. $Dg = \partial g/\partial t + \xi\cdot\nabla g$ is the substantial derivative of the equilibrium state. $\boldsymbol{\varphi}_\alpha$ is the moment vector given by

$$\boldsymbol{\varphi}_\alpha = \left[1, \xi_x, \xi_y, \frac{1}{2}\left(\xi_x^2+\xi_y^2+\sum_{j=1}^{K}\zeta_j^2\right)\right]^T. \tag{11}$$

According to the compatibility condition, the collision term makes no contribution to the calculation of conservative variables. Thus, the right-hand side of Eq. (9) goes to zero and the conservative variables and numerical fluxes of the Navier-Stokes equations can be expressed as

$$\mathbf{W} = \langle\boldsymbol{\varphi}_\alpha f\rangle = \langle\boldsymbol{\varphi}_\alpha g\rangle, \tag{12}$$

$$\mathbf{F} = \langle\xi\boldsymbol{\varphi}_\alpha f\rangle = \langle\xi\boldsymbol{\varphi}_\alpha(g-\tau Dg)\rangle. \tag{13}$$

In addition, the dynamic viscosity and the thermal conductivity can be calculated by

$$\mu = \tau p, \tag{14}$$

$$\kappa = \tau p(c_V + R) = \mu c_p, \tag{15}$$

where c_V and c_p are specific heat capacities at constant volume and constant pressure, respectively. Equations (12)–(15) establish the connection between the Navier-Stokes equations and the Boltzmann equation. More specifically, the numerical fluxes of the Navier-Stokes equations can be reconstructed from Eq. (13) directly, but note that the numerical fluxes calculated by Eq. (13) correspond to the unit Prandtl number, which should be corrected in the calculation.

For the convenience of application on the unstructured mesh, the local coordinate system defined at the cell interface is introduced. In the local coordinate system, the x_1 axis points to the normal direction and the x_2 axis is aligned with the tangential direction of the cell interface. Under the local coordinate system, the conservative variables and numerical fluxes can be, respectively, expressed as

$$\overline{\mathbf{W}} = \langle\bar{\boldsymbol{\varphi}}_a f\rangle = (\rho, \rho u_1, \rho u_2, \rho E)^T, \tag{16}$$

$$\overline{\mathbf{F}} = \langle\xi_1\bar{\boldsymbol{\varphi}}_a f\rangle = \left(F_\rho, F_{\rho u_1}, F_{\rho u_2}, F_{\rho E}\right)^T, \tag{17}$$

with

$$\bar{\boldsymbol{\varphi}}_a = \left[1, \xi_1, \xi_2, \frac{1}{2}\left(\xi_1^2+\xi_2^2+\sum_{j=1}^{K}\zeta_j^2\right)\right]^T, \tag{18}$$

where (u_1, u_2) and (ξ_1, ξ_2) are the components of the flow velocity and the particle velocity expressed in the local coordinate system, respectively. $\mathbf{F}$ and $\overline{\mathbf{F}}$ can be correlated by the following coordinate transformation,

$$\mathbf{F}\cdot\mathbf{n} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & n_x & -n_y & 0\\ 0 & n_y & n_x & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\overline{\mathbf{F}}, \tag{19}$$

where $\mathbf{n} = (n_x, n_y)$ denotes the unit normal vector of the cell interface in the global Cartesian coordinate system. Equations (17) and (19) indicate that the numerical fluxes of the macroscopic governing equations can be calculated by the local asymptotic solution to the Boltzmann equation in the local coordinate system.

B. High-order approximation to local asymptotic solution

To develop the high-order GKFS, the local asymptotic solution to the Boltzmann equation at the cell interface has to be approximated with high-order accuracy. Suppose that the cell interface is located at $\mathbf{x} = \mathbf{0}$; the discretization form of Eq. (10) by a difference scheme with second-order accuracy in time and fourth-order accuracy in space can be written as

$$f(\mathbf{0}, \xi, t^n + \Delta t_p) = g(\mathbf{0}, \xi, t^n + \Delta t_p) + \frac{\tau}{6\Delta t_p}[2g(-3\xi\Delta t_p, \xi, t^n) - 9g(-2\xi\Delta t_p, \xi, t^n) + 18g(-\xi\Delta t_p, \xi, t^n) - 5g(\mathbf{0}, \xi, t^n) - 6g(\mathbf{0}, \xi, t^n + \Delta t_p)] + O\left(\tau^2, \tau\Delta t_p, \tau|\xi|^4\Delta t_p^3\right), \tag{20}$$

with

$$g(\mathbf{0},\xi,t^n)=H(\xi_1)g_0^L+[1-H(\xi_1)]g_0^R,$$

where t^n denotes the current time level whose flow variables are known. The superscripts "L" and "R" denote the variables at the left and right sides of the cell interface, respectively. $H(\xi_1)$ is the Heaviside function; $H(\xi_1)=1$ for $\xi_1\geqslant 0$ and $H(\xi_1)=0$ for $\xi_1<0$. Δt_p is the virtual time step size used in solution reconstruction, which can be calculated by

$$\Delta t_p=\sigma_p\frac{\min(\Delta l,\Delta r)}{\max(u_1,u_2)+c_s}. \tag{21}$$

Here, Δl and Δr are the shortest edges of the left and right cells around the cell interface, respectively. c_s is the sound speed. σ_p is a constant, which is taken as 0.2 and 0.4 for the triangular mesh and the quadrilateral mesh, respectively. In fact, the truncation error of Eq. (20) can also be obtained by Taylor serial expansion to the terms of $g(-3\xi\Delta t_p,\xi,t^n)$, $g(-2\xi\Delta t_p,\xi,t^n)$, and $g(-\xi\Delta t_p,\xi,t^n)$ in space.

In Eq. (20), the equilibrium distribution function at the surrounding points of the cell interface can be approximated by the function of the equilibrium states and their derivatives at the left and right sides of the cell interface, e.g.,

$$\begin{aligned}g(-\xi\Delta t_p,\xi,t^n)=&H(\xi_1)g_0^L\big[1-a_1^L\xi_1\Delta t_p-a_2^L\xi_2\Delta t_p+\tfrac{1}{2}b_{11}^L(\xi_1\Delta t_p)^2+\tfrac{1}{2}b_{22}^L(\xi_2\Delta t_p)^2\\&+b_{12}^L\xi_1\xi_2\Delta t_p^2-\tfrac{1}{6}c_{111}^L(\xi_1\Delta t_p)^3-\tfrac{1}{6}c_{222}^L(\xi_2\Delta t_p)^3-\tfrac{1}{2}c_{112}^L\xi_1^2\xi_2\Delta t_p^3-\tfrac{1}{2}c_{122}^L\xi_1\xi_2^2\Delta t_p^3\big]\\&+[1-H(\xi_1)]g_0^R\big[1-a_1^R\xi_1\Delta t_p-a_2^R\xi_2\Delta t_p+\tfrac{1}{2}b_{11}^R(\xi_1\Delta t_p)^2+\tfrac{1}{2}b_{22}^R(\xi_2\Delta t_p)^2\\&+b_{12}^R\xi_1\xi_2\Delta t_p^2-\tfrac{1}{6}c_{111}^R(\xi_1\Delta t_p)^3-\tfrac{1}{6}c_{222}^R(\xi_2\Delta t_p)^3-\tfrac{1}{2}c_{112}^R\xi_1^2\xi_2\Delta t_p^3-\tfrac{1}{2}c_{122}^R\xi_1\xi_2^2\Delta t_p^3\big],\end{aligned} \tag{22}$$

where

$$a_1g_0=\frac{\partial g_0}{\partial x_1},\quad a_2g_0=\frac{\partial g_0}{\partial x_2},\quad b_{11}g_0=\frac{\partial^2 g_0}{\partial x_1\partial x_1},\quad b_{22}g_0=\frac{\partial^2 g_0}{\partial x_2\partial x_2},\quad b_{12}g_0=\frac{\partial^2 g_0}{\partial x_1\partial x_2},$$
$$c_{111}g_0=\frac{\partial^3 g_0}{\partial x_1\partial x_1\partial x_1},\quad c_{222}g_0=\frac{\partial^3 g_0}{\partial x_2\partial x_2\partial x_2},\quad c_{112}g_0=\frac{\partial^3 g_0}{\partial x_1\partial x_1\partial x_2},\quad c_{122}g_0=\frac{\partial^3 g_0}{\partial x_1\partial x_2\partial x_2}.$$

Substituting all expansions of $g(-3\xi\Delta t_p,\xi,t^n)$, $g(-2\xi\Delta t_p,\xi,t^n)$, and $g(-\xi\Delta t_p,\xi,t^n)$ into Eq. (20), we have

$$\begin{aligned}f(\mathbf{0},\xi,t^n+\Delta t_p)=&g(\mathbf{0},\xi,t^n+\Delta t_p)+\tau/\Delta t_p\big\{H(\xi_1)g_0^L+[1-H(\xi_1)]g_0^R-g(\mathbf{0},\xi,t^n+\Delta t_p)\big\}\\&-\tau\big\{g_0^L\big(a_1^L\xi_1+a_2^L\xi_2\big)H(\xi_1)+g_0^R\big(a_1^R\xi_1+a_2^R\xi_2\big)[1-H(\xi_1)]\big\}.\end{aligned} \tag{23}$$

It can be seen from Eq. (23) that only the equilibrium states and their first-order derivatives at the left and right sides of the cell interface at time level t^n and the equilibrium state at the cell interface at time level $t^n+\Delta t_p$ are involved in the high-order GKFS, which is more concise than the high-order GKS [26–30].

To obtain the explicit formulation of the numerical fluxes, the expansion coefficients a_1^L, a_2^L, a_1^R, a_2^R, and the equilibrium states should be derived in advance. By taking moment integration to a_1g_0 and a_2g_0, we have

$$\langle a_1\bar{\boldsymbol{\varphi}}_\alpha g_0\rangle=(\partial\rho/\partial x_1,\partial\rho u_1/\partial x_1,\partial\rho u_2/\partial x_1,\partial\rho E/\partial x_1)^T, \tag{24}$$

$$\langle a_2\bar{\boldsymbol{\varphi}}_\alpha g_0\rangle=(\partial\rho/\partial x_2,\partial\rho u_1/\partial x_2,\partial\rho u_2/\partial x_2,\partial\rho E/\partial x_2)^T, \tag{25}$$

with

$$a_1=a_{1,0}+a_{1,1}\xi_1+a_{1,2}\xi_2+\frac{1}{2}a_{1,3}\left(\xi_1^2+\xi_2^2+\sum_{j=1}^K\zeta_j^2\right),\quad a_2=a_{2,0}+a_{2,1}\xi_1+a_{2,2}\xi_2+\frac{1}{2}a_{2,3}\left(\xi_1^2+\xi_2^2+\sum_{j=1}^K\zeta_j^2\right).$$

In the above equations, the superscripts L and R have been omitted for simplicity. It can be seen from Eqs. (24) and (25) that a_1 and a_2 satisfy the same equations. Thus, for any expansion coefficient ϕ, it can be computed by

$$\langle\phi\bar{\boldsymbol{\varphi}}_\alpha g_0\rangle=(h_0,h_1,h_2,h_3)^T, \tag{26}$$

with

$$\phi = \phi_0 + \phi_1\xi_1 + \phi_2\xi_2 + \frac{1}{2}\phi_3\left(\xi_1^2 + \xi_2^2 + \sum_{j=1}^{K}\zeta_j^2\right),$$

where $(h_0, h_1, h_2, h_3)^T$ is the right-hand side of Eqs. (24) or (25). Following the derivations given in Refs. [7,46], the solution of Eq. (26) yields

$$\phi_3 = \frac{4\lambda^2}{K+2}(r_3 - 2u_1r_1 - 2u_2r_2), \tag{27}$$

$$\phi_2 = 2\lambda r_2 - u_2\phi_3, \tag{28}$$

$$\phi_1 = 2\lambda r_1 - u_1\phi_3, \tag{29}$$

$$\phi_0 = h_0/\rho - u_1\phi_1 - u_2\phi_2 - \frac{1}{2}r_0\phi_3, \tag{30}$$

with

$$r_0 = u_1^2 + u_2^2 + \frac{K+2}{2\lambda}, \quad r_1 = (h_1 - u_1h_0)/\rho, \quad r_2 = (h_2 - u_2h_0)/\rho r_3 = (2h_3 - r_0h_0)/\rho.$$

As for the equilibrium states g_0^L and g_0^R, they can be calculated by the flow variables at the left and right sides of the cell interface, respectively. $g(\mathbf{0}, \xi, t^n + \Delta t_p)$ can be determined by the compatibility condition,

$$\overline{\mathbf{W}} = \langle\bar{\boldsymbol{\varphi}}_\alpha g(\mathbf{0}, \xi, t^n + \Delta t_p)\rangle = \left\langle H(\xi_1)\bar{\boldsymbol{\varphi}}_\alpha g_0^L\left[1 - \Delta t_p\left(a_1^L\xi_1 + a_2^L\xi_2\right)\right]\right\rangle + \left\langle[1 - H(\xi_1)]\bar{\boldsymbol{\varphi}}_\alpha g_0^R\left[1 - \Delta t_p\left(a_1^R\xi_1 + a_2^R\xi_2\right)\right]\right\rangle. \tag{31}$$

Specifically, $\overline{\mathbf{W}}$ is calculated from g_0^L and g_0^R by Eq. (31) and $g(\mathbf{0}, \xi, t^n + \Delta t_p)$ is then computed from $\overline{\mathbf{W}}$ by Eq. (7).

C. Numerical fluxes of Navier-Stokes equations at the cell interface

For the convenience of application, the explicit expressions of $\overline{\mathbf{W}}$ and $\overline{\mathbf{F}}$ are given in this section. First, some coefficients of the integral over the full or the half particle velocity space are introduced as follows:

$$\beta_i^L = \frac{1}{\rho^L}\int_{\xi_1>0}\xi_1^i g_0^L d\xi d\zeta, \tag{32}$$

$$\beta_i^R = \frac{1}{\rho^R}\int_{\xi_1<0}\xi_1^i g_0^R d\xi d\zeta, \tag{33}$$

$$\gamma_i = \frac{1}{\rho}\int\xi_1^i g_0 d\xi d\zeta, \tag{34}$$

$$\chi_i = \frac{1}{\rho}\int\xi_2^i g_0 d\xi d\zeta, \tag{35}$$

$$\eta_i = \frac{1}{\rho}\int\sum_{j=1}^{K}\zeta_j^i g_0 d\xi d\zeta. \tag{36}$$

For calculation of $\overline{\mathbf{W}}$ and $\overline{\mathbf{F}}$, the following coefficients will be used:

$$\beta_0^L = \left[\frac{1}{2}\mathrm{erfc}(-\sqrt{\lambda}u_1)\right]^L, \quad \beta_1^L = u_1^L\beta_0^L + \left[\frac{1}{2}\frac{e^{-\lambda u_1^2}}{\sqrt{\pi\lambda}}\right]^L,$$

$$\beta_2^L = u_1^L\beta_1^L + \frac{1}{2\lambda^L}\beta_0^L, \quad \beta_3^L = u_1^L\beta_2^L + \frac{2}{2\lambda^L}\beta_1^L, \quad \beta_4^L = u_1^L\beta_3^L + \frac{3}{2\lambda^L}\beta_2^L,$$

$$\beta_5^L = u_1^L\beta_4^L + \frac{4}{2\lambda^L}\beta_3^L, \quad \beta_6^L = u_1^L\beta_5^L + \frac{5}{2\lambda^L}\beta_4^L.$$

$$\beta_0^R = \left[\frac{1}{2}\mathrm{erfc}(\sqrt{\lambda}u_1)\right]^R, \quad \beta_1^R = u_1^R\beta_0^R - \left[\frac{1}{2}\frac{e^{-\lambda u_1^2}}{\sqrt{\pi\lambda}}\right]^R,$$

$$\beta_2^R = u_1^R\beta_1^R + \frac{1}{2\lambda^R}\beta_0^R, \quad \beta_3^R = u_1^R\beta_2^R + \frac{2}{2\lambda^R}\beta_1^R, \quad \beta_4^R = u_1^R\beta_3^R + \frac{3}{2\lambda^R}\beta_2^R,$$

$$\beta_5^R = u_1^R\beta_4^R + \frac{4}{2\lambda^R}\beta_3^R, \quad \beta_6^R = u_1^R\beta_5^R + \frac{5}{2\lambda^R}\beta_4^R.$$

$$\gamma_1 = u_1, \quad \gamma_2 = u_1^2 + \frac{1}{2\lambda}, \quad \gamma_3 = u_1\gamma_2 + \frac{2}{2\lambda}\gamma_1, \quad \gamma_4 = u_1\gamma_3 + \frac{3}{2\lambda}\gamma_2, \quad \gamma_5 = u_1\gamma_4 + \frac{4}{2\lambda}\gamma_3.$$

$$\chi_1 = u_2, \quad \chi_2 = u_2^2 + \frac{1}{2\lambda}, \quad \chi_3 = u_2\chi_2 + \frac{2}{2\lambda}\chi_1, \quad \chi_4 = u_2\chi_3 + \frac{3}{2\lambda}\chi_2, \quad \chi_5 = u_2\chi_4 + \frac{4}{2\lambda}\chi_3.$$

$$\eta_2 = \frac{K}{2\lambda}, \quad \eta_4 = \frac{3K}{4\lambda^2} + \frac{K(K-1)}{4\lambda^2}.$$

Note that since the coefficients $\gamma_1, \ldots, \gamma_5$, $\chi_1, \ldots, \chi_5$, η_2, and η_4 at the left and right sides of the cell interface have the same expressions, the superscripts L and R are omitted in the above equations.

With the above coefficients, the components of $\overline{\mathbf{W}}$ expressed in the local coordinate system can be written as

$$\overline{\mathbf{W}}(1) = \rho^L\beta_0^L + \rho^R\beta_0^R - \Delta t_p\left(\rho^L\hbar_0^L + \rho^R\hbar_0^R\right), \tag{37}$$

$$\overline{\mathbf{W}}(2) = \rho^L\beta_1^L + \rho^R\beta_1^R - \Delta t_p\left(\rho^L\hbar_1^L + \rho^R\hbar_1^R\right), \tag{38}$$

$$\overline{\mathbf{W}}(3) = \rho^L\beta_0^L\chi_1^L + \rho^R\beta_0^R\chi_1^R - \Delta t_p\left(\rho^L\hbar_2^L + \rho^R\hbar_2^R\right), \tag{39}$$

$$\overline{\mathbf{W}}(4) = \rho^L\left(\beta_2^L + \beta_0^L\chi_2^L + \beta_0^L\eta_2^L\right)/2 + \rho^R\left(\beta_2^R + \beta_0^R\chi_2^R + \beta_0^R\eta_2^R\right)/2 - \Delta t_p\left(\rho^L\hbar_3^L + \rho^R\hbar_3^R\right), \tag{40}$$

The definitions of $\hbar_0$, $\hbar_1$, $\hbar_2$, and $\hbar_3$ are given in Appendix.

Similarly, the four components of $\overline{\mathbf{F}}$ can be computed by

$$\overline{\mathbf{F}}(1) = \rho u_1, \tag{41}$$

$$\overline{\mathbf{F}}(2) = (1 - \tau/\Delta t_p)\rho\gamma_2 + \tau/\Delta t_p\left(\rho^L\beta_2^L + \rho^R\beta_2^R\right) - \tau\left(\rho^L\lambda\!\!\!^{-}_1^L + \rho^R\lambda\!\!\!^{-}_1^R\right), \tag{42}$$

$$\overline{\mathbf{F}}(3) = (1 - \tau/\Delta t_p)\rho\gamma_1\chi_1 + \tau/\Delta t_p\left(\rho^L\beta_1^L\chi_1^L + \rho^R\beta_1^R\chi_1^R\right) - \tau\left(\rho^L\lambda\!\!\!^{-}_2^L + \rho^R\lambda\!\!\!^{-}_2^R\right), \tag{43}$$

$$\overline{\mathbf{F}}(4) = \tfrac{1}{2}(1 - \tau/\Delta t_p)(\gamma_3 + \gamma_1\chi_2 + \gamma_1\eta_2) + \tau/\Delta t_p\left(\rho^L\ell_3^L + \rho^R\ell_3^R\right) - \tau\left(\rho^L\lambda\!\!\!^{-}_3^L + \rho^R\lambda\!\!\!^{-}_3^R\right), \tag{44}$$

The definitions of $\lambda\!\!\!^{-}_1, \lambda\!\!\!^{-}_2, \lambda\!\!\!^{-}_3$, and ℓ_3 are given in Appendix. Note that, in Eqs. (41)–(44), the flow variables without superscripts L and R are determined from the conservative variables at the cell interface $\overline{\mathbf{W}}$.

It can be seen from Eqs. (37)–(44) that the inviscid flux and the viscous flux are calculated simultaneously in the GKFS. This process is different from that in the classic cell-centered second-order FV method to solve Navier-Stokes equations, which deals with the inviscid flux and the viscous flux separately. Since the derivatives of the conservative variables at the cell interface are involved in calculating the viscous flux in the conventional algorithm, the deferred correction is often implemented to reduce the error resulting from nonorthogonality [47–49]. In comparison, the derivatives of the conservative variables at the left and right sides of the cell interface are directly utilized to evaluate the numerical fluxes in the present method; the deferred correction is unnecessary.

As mentioned in Sec. II A, the Prandtl number in the GKFS is fixed at 1. To adjust the Prandtl number to any realistic value, one convenient approach is to modify the energy flux $\overline{\mathbf{F}}(4)$ with a variable Prandtl number Pr [31],

$$\overline{\mathbf{F}}(4)^{\text{correct}} = \overline{\mathbf{F}}(4) + \left(\frac{1}{\text{Pr}} - 1\right)q, \tag{45}$$

where the heat flux q is computed by

$$\begin{aligned} q &= \left\langle \frac{1}{2}(\xi_1 - u_1)\left[(\xi_1 - u_1)^2 + (\xi_2 - u_2)^2 + \sum_{j=1}^{K}\zeta_j^2\right] f(\mathbf{0}, \xi, t^n + \Delta t_p)\right\rangle \\ &= \left\langle \frac{1}{2}(\xi_1 - u_1)\left(\xi_1^2 + \xi_2^2 + \sum_{j=1}^{K}\zeta_j^2 + u_1^2 + u_2^2 - 2\xi_1 u_1 - 2\xi_2 u_2\right) f(\mathbf{0}, \xi, t^n + \Delta t_p)\right\rangle. \end{aligned} \tag{46}$$

Using the definition of $\overline{\mathbf{W}}$ and $\overline{\mathbf{F}}$, Eq. (46) can be reduced to

$$q = \overline{\mathbf{F}}(4) - u_1\overline{\mathbf{F}}(2) - u_2\overline{\mathbf{F}}(3) - \rho u_1\left(E - u_1^2 - u_2^2\right). \tag{47}$$

In Eq. (47), ρ, u_1, u_2, and E are determined from $\overline{\mathbf{W}}$.

Theoretically, the collision timescale should be calculated uniquely by Eq. (14). However, for fluid flow problems with strong shock waves, artificial dissipation is necessary to ensure numerical stability. Thus, the effective viscosity should be a combination

of the physical and numerical viscosities. To achieve this goal, the collision timescale τ is modified as [2]

$$\tau = \frac{\mu}{p} + \frac{|p^L - p^R|}{|p^L + p^R|}\Delta t_p, \tag{48}$$

where p^L and p^R are, respectively, the pressure at the left and right sides of the cell interface. The second part of Eq. (48) corresponds to the numerical viscosity, which takes effect in the presence of shock waves.

III. HIGH-ORDER FINITE-VOLUME DISCRETIZATION OF NAVIER-STOKES EQUATIONS

A. Spatial discretization and derivative approximation

In this work, the fourth-order LSFD FV method is used to discretize the macroscopic governing equations [36,45]. To achieve fourth-order accuracy, the conservative variables in the control volume are approximated by the following Taylor series expansion:

$$\begin{aligned}\mathbf{W}(x,y) = \mathbf{W}_i &+ \frac{\partial \mathbf{W}}{\partial x}\bigg|_i (x-x_i) + \frac{\partial \mathbf{W}}{\partial y}\bigg|_i (y-y_i) + \frac{\partial^2 \mathbf{W}}{\partial x^2}\bigg|_i \frac{(x-x_i)^2}{2} \\ &+ \frac{\partial^2 \mathbf{W}}{\partial y^2}\bigg|_i \frac{(y-y_i)^2}{2} + \frac{\partial^2 \mathbf{W}}{\partial x \partial y}\bigg|_i (x-x_i)(y-y_i) + \frac{\partial^3 \mathbf{W}}{\partial x^3}\bigg|_i \frac{(x-x_i)^3}{6} \\ &+ \frac{\partial^3 \mathbf{W}}{\partial y^3}\bigg|_i \frac{(y-y_i)^3}{6} + \frac{\partial^3 \mathbf{W}}{\partial x^2 \partial y}\bigg|_i \frac{(x-x_i)^2(y-y_i)}{2} + \frac{\partial^3 \mathbf{W}}{\partial y^2 \partial x}\bigg|_i \frac{(y-y_i)^2(x-x_i)}{2}.\end{aligned} \tag{49}$$

Here, the reference point (x_i, y_i) is the centroid of cell i. The derivatives of $\mathbf{W}$ are determined by the LSFD method. Specifically, by applying Eq. (49) to cell i and its neighboring cells, we have

$$\mathbf{S}_i d\mathbf{W}_i = \Delta \mathbf{W}_i, \tag{50}$$

where

$$\mathbf{S}_i = \begin{pmatrix} \Delta x_{i1} & \Delta y_{i1} & \frac{\Delta x_{i1}^2}{2} & \frac{\Delta y_{i1}^2}{2} & \Delta x_{i1}\Delta y_{i1} & \frac{\Delta x_{i1}^3}{6} & \frac{\Delta y_{i1}^3}{6} & \frac{\Delta x_{i1}^2 \Delta y_{i1}}{2} & \frac{\Delta y_{i1}^2 \Delta x_{i1}}{2} \\ \Delta x_{i2} & \Delta y_{i2} & \frac{\Delta x_{i2}^2}{2} & \frac{\Delta y_{i2}^2}{2} & \Delta x_{i2}\Delta y_{i2} & \frac{\Delta x_{i2}^3}{6} & \frac{\Delta y_{i2}^3}{6} & \frac{\Delta x_{i2}^2 \Delta y_{i2}}{2} & \frac{\Delta y_{i2}^2 \Delta x_{i2}}{2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Delta x_{iN} & \Delta y_{iN} & \frac{\Delta x_{iN}^2}{2} & \frac{\Delta y_{iN}^2}{2} & \Delta x_{iN}\Delta y_{iN} & \frac{\Delta x_{iN}^3}{6} & \frac{\Delta y_{iN}^3}{6} & \frac{\Delta x_{iN}^2 \Delta y_{iN}}{2} & \frac{\Delta y_{iN}^2 \Delta x_{iN}}{2} \end{pmatrix}, \tag{51}$$

$$d\mathbf{W}_i^T = \left[\frac{\partial \mathbf{W}}{\partial x}, \frac{\partial \mathbf{W}}{\partial y}, \frac{\partial^2 \mathbf{W}}{\partial x^2}, \frac{\partial^2 \mathbf{W}}{\partial y^2}, \frac{\partial^2 \mathbf{W}}{\partial x \partial y}, \frac{\partial^3 \mathbf{W}}{\partial x^3}, \frac{\partial^3 \mathbf{W}}{\partial y^3}, \frac{\partial^3 \mathbf{W}}{\partial x^2 \partial y}, \frac{\partial^3 \mathbf{W}}{\partial y^2 \partial x}\right]_i, \tag{52}$$

$$\Delta \mathbf{W}_i^T = [\mathbf{W}_{i1} - \mathbf{W}_i, \mathbf{W}_{i2} - \mathbf{W}_i, \ldots, \mathbf{W}_{iN} - \mathbf{W}_i]. \tag{53}$$

Here, $\Delta x_{ij} = x_j - x_i$, $\Delta y_{ij} = y_j - y_i$, $j = 1, \ldots, N$, where N is the number of the neighboring cells of cell i. In this work, N is taken as three layers of the von Neumann neighborhood for the inner cells and 15 for the boundary cells to avoid ill conditioned and singular. The solution of Eq. (50) given by the least-square method yields

$$d\mathbf{W}_i = \mathbf{K}_i \Delta \mathbf{W}_i, \tag{54}$$

where $\mathbf{K}_i$ is a $9\times N$ dimensional weighting coefficient matrix, which is uniquely determined by the centroid of cell i and its neighboring cells.

To update the conservative variables $\mathbf{W}$ at the cell center, Eq. (4) should be marched in time. The discretization form of Eq. (4) by the high-order FV method can be written as

$$\frac{d}{dt}\left(\int_{\Omega_i} \mathbf{W} d\Omega\right) = -\sum_{j=1}^{Nf}\sum_{k=1}^{nGQp} (\mathbf{F}_k \cdot \mathbf{n}A)_j \omega_k, \tag{55}$$

where i is the index of the control volume; Ω_i and Nf represent the volume and the number of faces of the control volume i, respectively. A denotes the area of the interface of the control volume. $nGQp$ is the number of Gaussian quadrature points on the cell interface and ω_k is the corresponding quadrature weight. Substituting Eqs. (49) and (54) into Eq. (55), we have

$$\frac{d}{dt}\left[\left(\Omega_i - \sum_{j=1}^{N}\sum_{k=1}^{9} C_k^i K_{k,j}^i\right)\mathbf{W}_i + \sum_{j=1}^{N}\left(\sum_{k=1}^{9} C_k^i K_{k,j}^i\right)\mathbf{W}_{ij}\right] = \mathbf{R}_i, \tag{56}$$

where C_k^i is the k^{th} component of $\mathbf{C}_i$, which is defined by

$$\mathbf{C}_i^T = \left[\overline{x^1y^0}, \overline{x^0y^1}, \frac{\overline{x^2y^0}}{2}, \frac{\overline{x^0y^2}}{2}, \overline{x^1y^1}, \frac{\overline{x^3y^0}}{6}, \frac{\overline{x^0y^3}}{6}, \frac{\overline{x^2y^1}}{2}, \frac{\overline{x^1y^2}}{2}\right]_i, \tag{57}$$

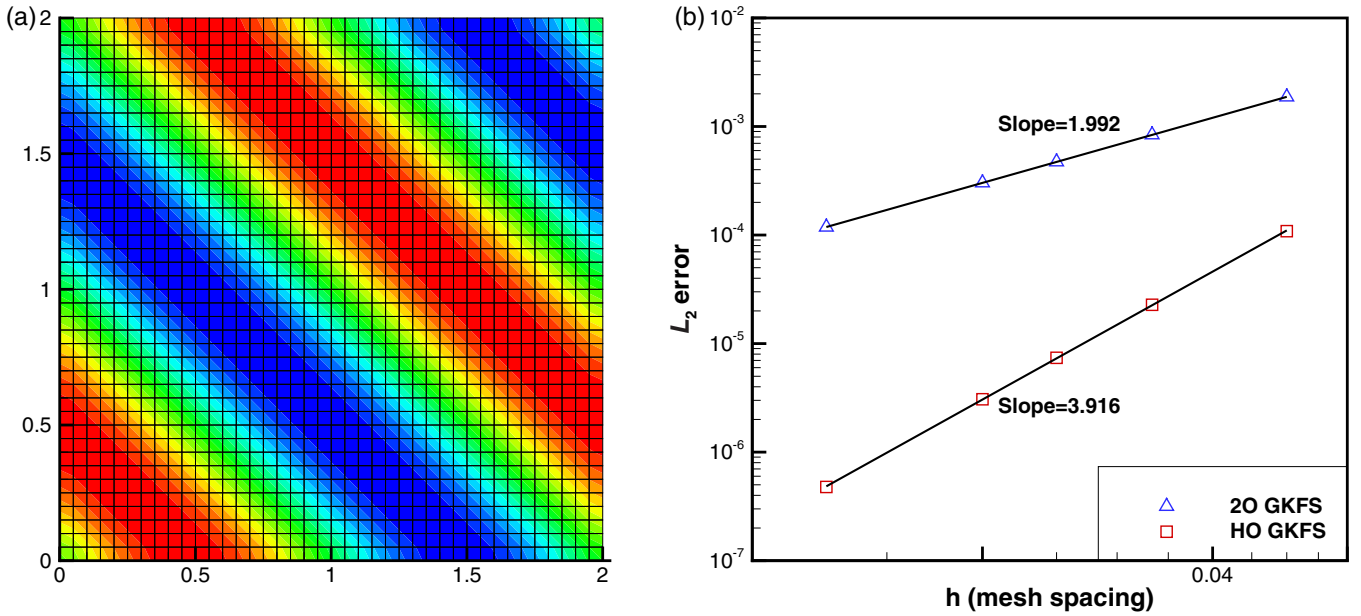

FIG. 1. (a) Density contours and (b) accuracy test of 2O GKFS and HO GKFS for simulating advection of density perturbation on the regular quadrilateral mesh.

where $\overline{x^n y^m}_i = \int_{\Omega_i} (x - x_i)^n (y - y_i)^m d\Omega$ and $\mathbf{R}_i$ is the right-hand side of Eq. (55). By applying Eq. (56) to all control volumes and considering the problem with fixed mesh, the following matrix form can be obtained:

$$\mathbf{M}\frac{d\mathbf{W}}{dt} = \mathbf{R}. \tag{58}$$

Here, $\mathbf{M}$ is composed of the coefficients on the left-hand side of Eq. (56). The solution of Eq. (58) yields $\mathbf{W}$ at the cell center at the next time step.

It should be indicated that the present high-order FV method requires a large stencil to calculate the high-order derivatives of flow variables, which is different from the compact scheme. Such a large stencil causes serious challenges for partition parallel computation, especially for evaluation of the high-order derivatives at the cells in the vicinity of the partition boundary. Specifically, except for the exchanging of information between the direct neighbors of a partition boundary, the information of cells in a large stencil should also be communicated. Even so, some efforts have been made to implement the parallelization for the high-order FV method [50,51]. These strategies of partition parallelization are also expected to be introduced into the present method. In this work, we only use serial computation to do the simulation

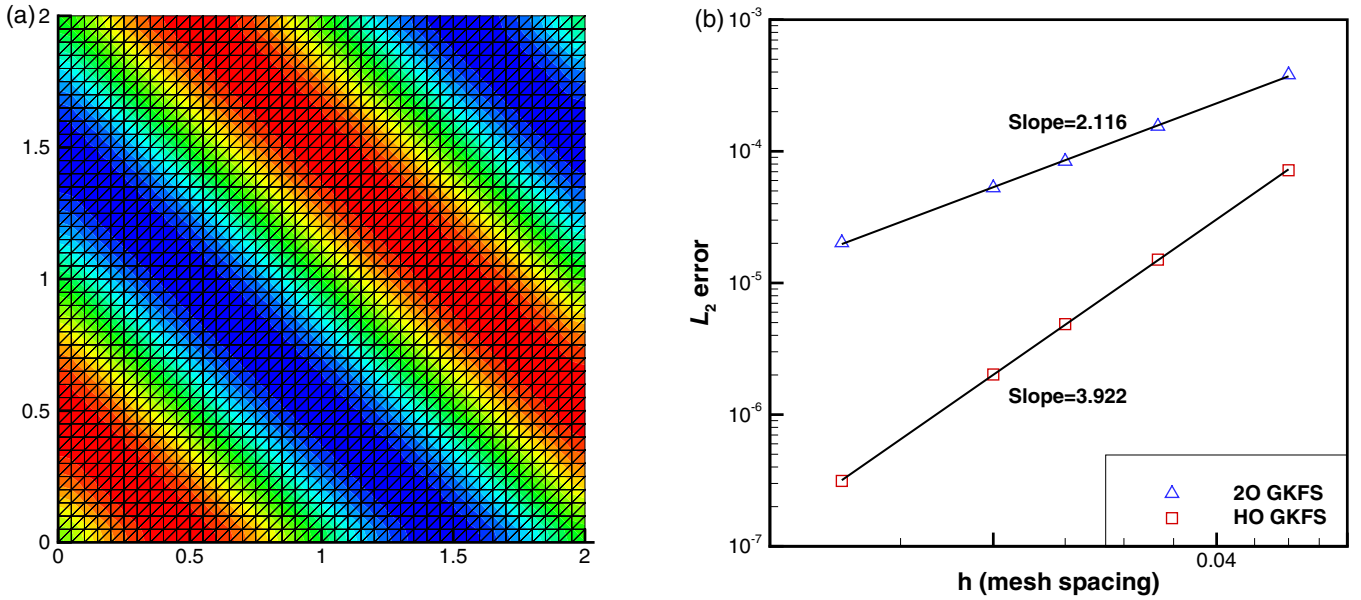

FIG. 2. (a) Density contours and (b) accuracy test of 2O GKFS and HO GKFS for simulating advection of density perturbation on the regular triangular mesh.

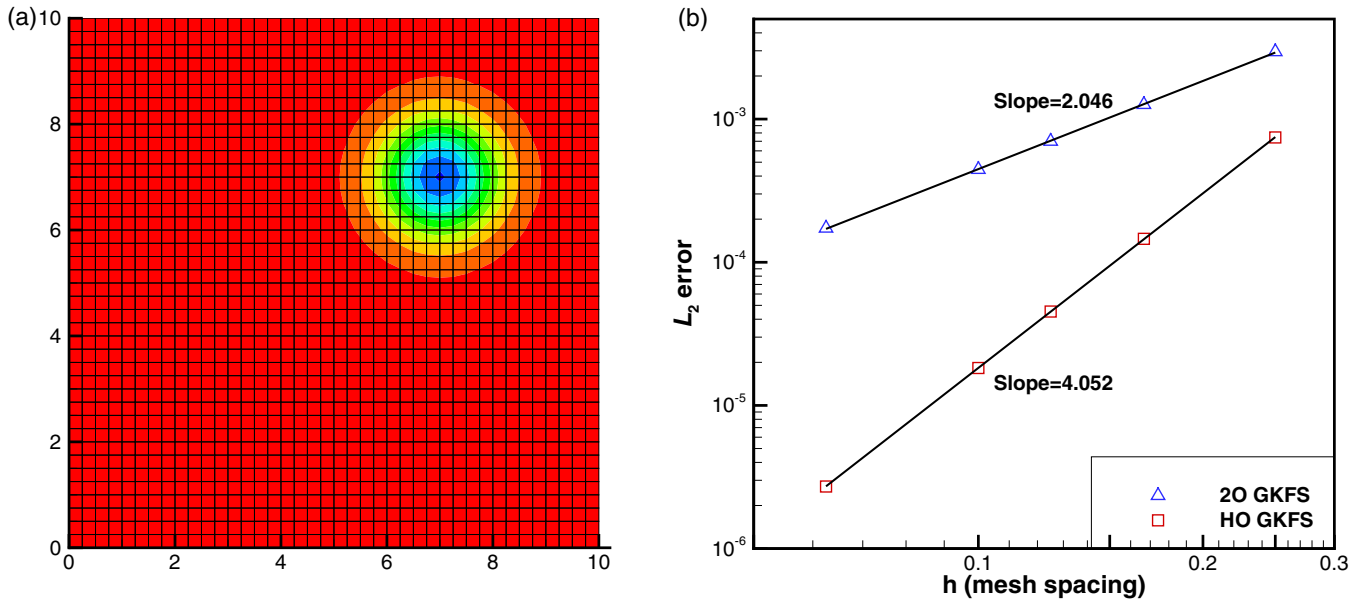

FIG. 3. (a) Density contours and (b) accuracy test of 2O GKFS and HO GKFS for simulating isentropic vortex flow on the regular quadrilateral mesh.

since the total computational time of the 2D case is relatively small.

B. Temporal discretization and computational sequence

To achieve the high-order accuracy in time for unsteady flow problems, the multistage Runge-Kutta scheme is applied as the temporal discretization strategy for Eq. (58) and the point iterative method is used in each stage [45]. For the steady flow problems, the implicit lower-upper symmetric Gauss-Seidel (LU SGS) scheme is adopted. The basic solution procedure of the present solver is summarized as follows:

(1) Calculate the derivatives of conservative variables by Eq. (54).

(2) Reconstruct the conservative variables and their first-order derivatives at the left and right sides of the cell interface by Eq. (49) and its derivatives.

(3) Compute the expansion coefficients a_1^L, a_2^L, a_1^R, a_2^R by Eqs. (27)–(30).

(4) Calculate the virtual time step size Δt_p by Eq. (21) and the collision timescale τ by Eq. (48).

(5) Evaluate the conservative variables at the cell interface by Eqs. (37)–(40).

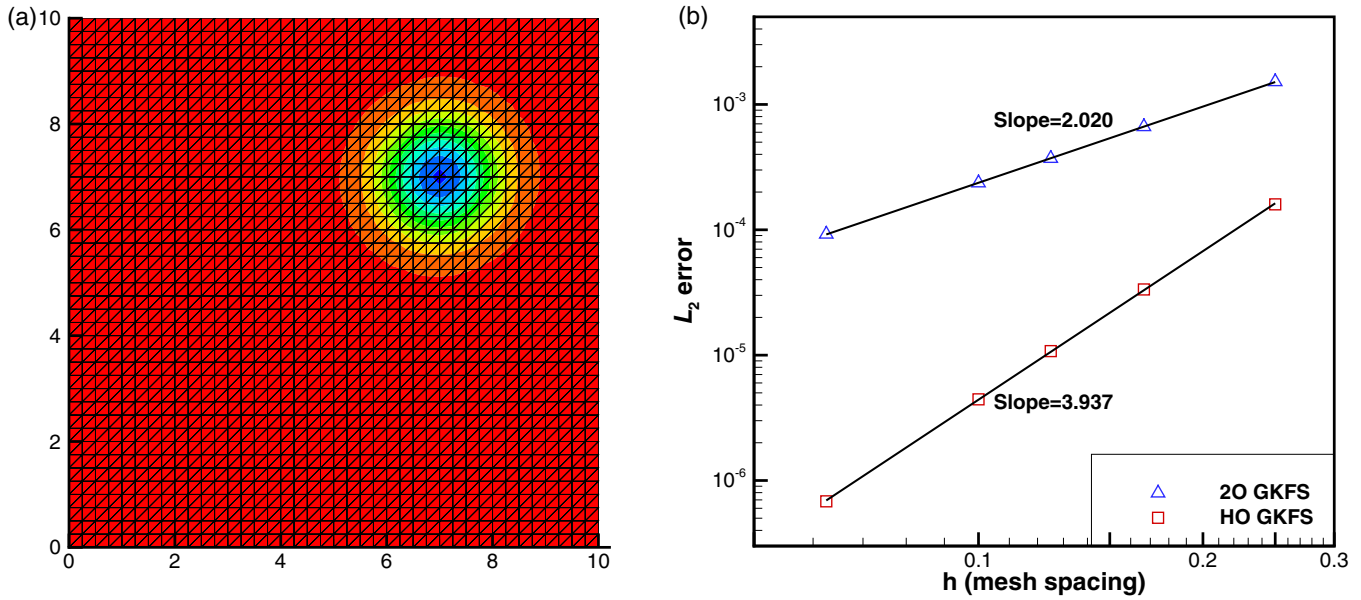

FIG. 4. (a) Density contours and (b) accuracy test of 2O GKFS and HO GKFS for simulating isentropic vortex flow on the regular triangular mesh.

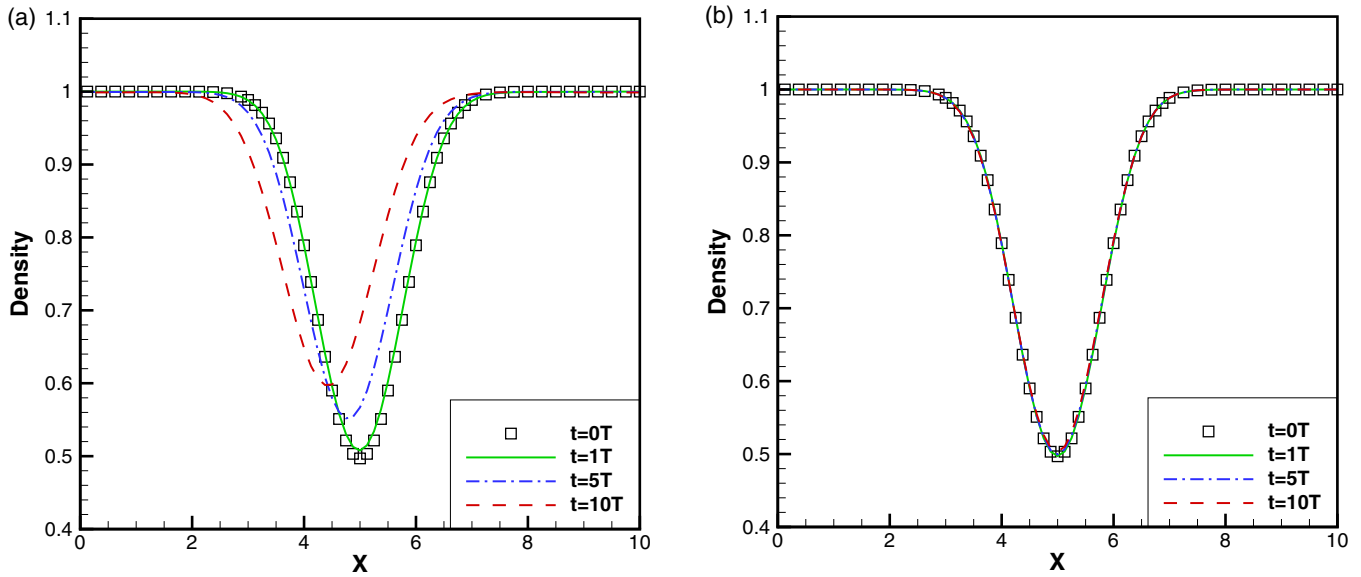

FIG. 5. Density profiles for isentropic vortex flow at different periods obtained by the second-order and high-order GKFS on the regular quadrilateral mesh with 6400 cells: (a) 2O GKFS, (b) HO GKFS.

(6) Compute the numerical fluxes of macroscopic governing equations by Eqs. (41)–(44) and correct the energy flux by Eq. (45).

(7) Update the conservative variables at the cell center by solving Eq. (58).

(8) Repeat steps 1–7 until the computation is finished.

IV. NUMERICAL EXAMPLES

The performance of the present high-order GKFS will be validated and compared with the second-order counterpart in this section. At first, the numerical accuracy of the high-order GKFS is tested by the advection of density perturbation and isentropic vortex flow. Since these two test examples belong to inviscid flows, in Eqs. (23) and (31), τ and Δt_p are set as zero and $\tau/\Delta t_p$ is treated as the weight of the numerical dissipation, which is calculated by a switch function shown in Ref. [52]. Second, the capability of the high-order GKFS on the complex geometry is verified by the transonic flow around the NACA0012 airfoil. Third, the viscous shock tube problem and the shock-boundary layer interaction are simulated to validate the performance of the high-order GKFS for simulation of supersonic flows. For convenience, the second-order GKFS and the high-order GKFS are abbreviated as "2O GKFS" and "HO GKFS" in the following text, respectively.

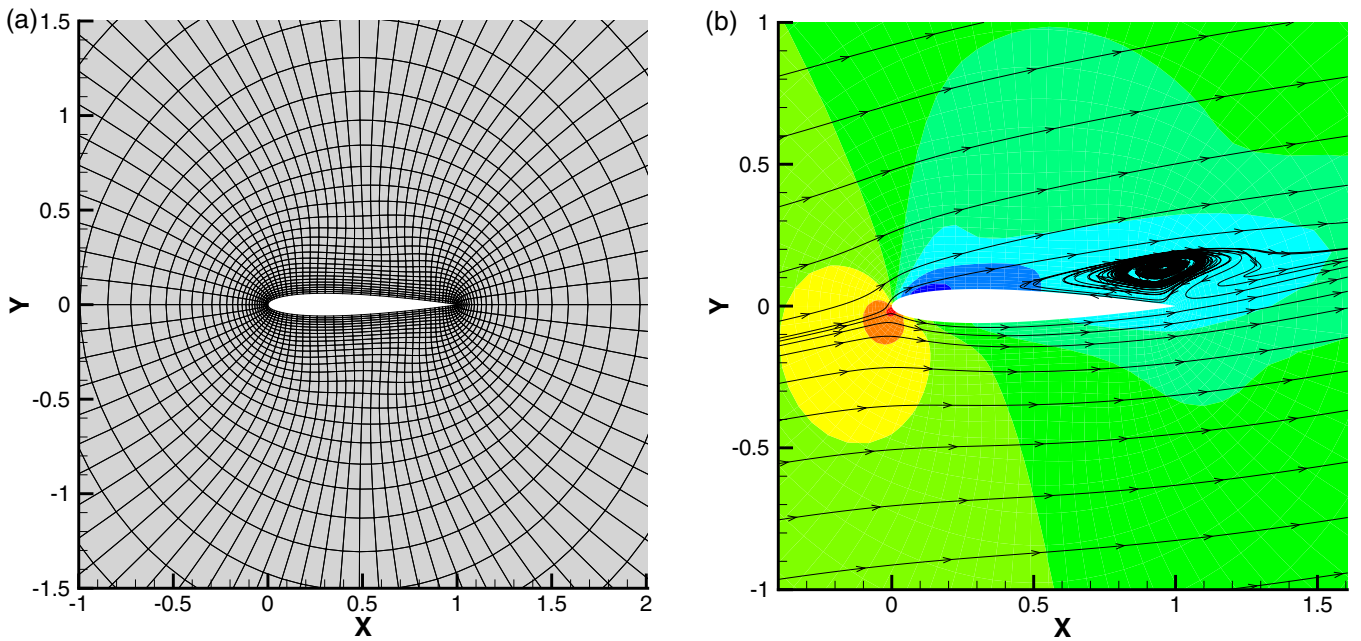

FIG. 6. (a) Computational mesh and (b) streamline pattern for transonic flow around one NACA0012 airfoil.

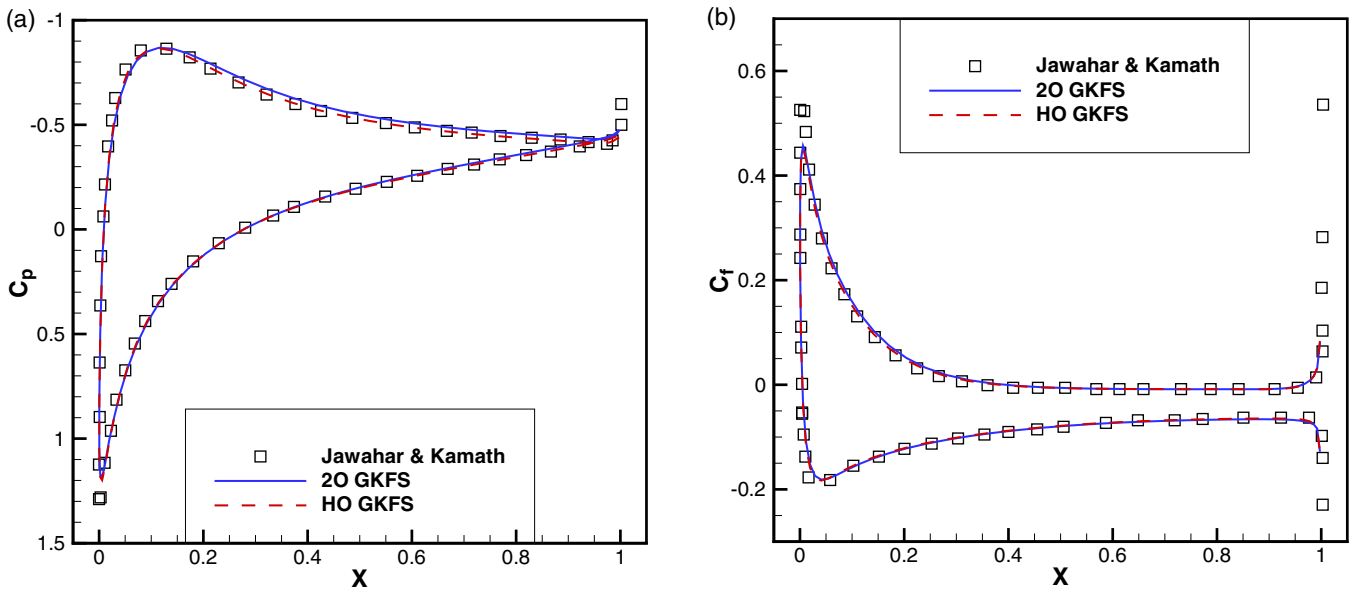

FIG. 7. (a) Pressure coefficient distribution and (b) skin friction coefficient distribution for transonic flow around one NACA0012 airfoil on a mesh with 3600 cells.

A. Case 1: Accuracy test

The first case for the accuracy test is the advection of density perturbation with the following initial condition,

$$\rho(x, y) = 1 + 0.2 \sin[\pi(x + y)],$$
$$u(x, y) = 1, \quad v(x, y) = 1, \quad p(x, y) = 1. \tag{59}$$

The exact solutions under periodic boundary condition are given by

$$\rho(x, y, t) = 1 + 0.2 \sin[\pi(x + y - 2t)],$$
$$u(x, y, t) = 1, \quad v(x, y, t) = 1, \quad p(x, y, t) = 1. \tag{60}$$

Numerical tests are conducted on the computational domain of $[0, 2]\times[0, 2]$, which is discretized by both the regular quadrilateral mesh and the regular triangular mesh with the sizes of $h = 1/20, 1/30, 1/40, 1/50$, and $1/80$. The L_2 norm of the error of density field at $t = 1$ is extracted to validate the convergence order numerically. As shown in Figs. 1 and 2, fourth-order accuracy can be achieved by the HO GKFS for both the quadrilateral mesh and the triangular mesh.

The second case for the accuracy test is the evolution of a 2D isentropic vortex in a free stream with periodic boundary conditions [53]. The free-stream conditions are $(\rho,\ u,\ v,\ p) = (1,\ 1,\ 1,\ 1)$. Initially, an isentropic vortex with no perturbation in entropy is added to the free stream,

$$(\delta u, \delta v) = \frac{\varepsilon}{2\pi} e^{0.5(1-r^2)}(-\bar{y}, \bar{x}),$$
$$\delta T = -\frac{(\gamma - 1)\varepsilon^2}{8\gamma\pi^2} e^{1-r^2},$$
$$\delta\rho = (T + \delta T)^{1/(\gamma-1)} - \rho, \tag{61}$$

where $(\bar{x}, \bar{y}) = (x - 5, y - 5)$, $r^2 = \bar{x}^2 + \bar{y}^2$ and the vortex strength is taken as $\varepsilon = 5$. In the simulation, the computational domain is set as $[0, 10]\times[0, 10]$ and discretized by both the regular quadrilateral mesh and the regular triangular mesh with the sizes of $h = 1/4, 1/6, 1/8, 1/10$, and $1/16$. Figures 3 and 4 show the density contours and the convergence order of the L_2 norm of the error of the density field at $t = 2$. Clearly, the HO GKFS can achieve the desired accuracy on both mesh types. The comparison of density distribution along the line of $y = 0.5$ at different periods ($t = 0$, T, $5T$, and $10T$) is displayed in Fig. 5. It can be seen that the HO GKFS has lower numerical dissipation as compared with the 2O GKFS. As time increases, the density distribution remains almost unchanged for the HO GKFS, while both the peak and the phase deviate significantly from the initial distribution for the 2O GKFS at $t = 10T$.

B. Case 2: Transonic flow around NACA0012 airfoil

In this section, the transonic flows around a NACA0012 airfoil and around a staggered biplane configuration are simulated. In the calculation, the free-stream Mach number is taken as 0.8, the chord length based Reynolds number is chosen as 500, and the angle of attack is set as 10°. For the test case of one NACA0012 airfoil, an O-type grid with 80 points on the airfoil surface and 41 points in the radial direction is utilized, as shown in Fig. 6(a). The far-field boundary is taken 25 times the chord length away from the geometrical center of the airfoil. The streamline pattern around the airfoil

TABLE I. Comparison of lift and drag coefficients for transonic flow around the NACA0012 airfoil.

References	C_d	C_l
GAMM [55]	0.243–0.2868	0.4145–0.517
Jawahar and Kamath [54]	0.27726	0.50231
2O GKFS (3600 cells)	0.29047	0.46283
2O GKFS (7200 cells)	0.28491	0.45776
HO GKFS (3600 cells)	0.28367	0.45060

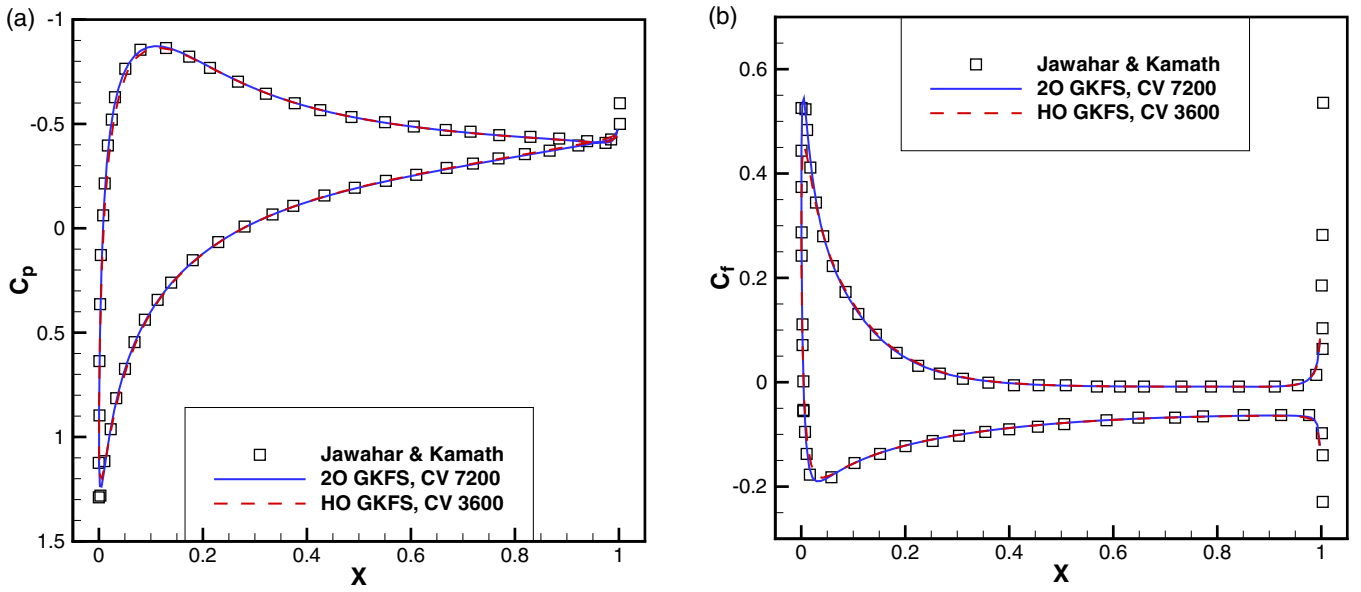

FIG. 8. (a) Pressure coefficient distribution and (b) skin friction coefficient distribution for transonic flow around one NACA0012 airfoil on different meshes.

obtained by the HO GKFS is displayed in Fig. 6(b). A vortex that extends over 50% of the chord on the upper surface of the airfoil is well captured, which is in line with the results reported in Refs. [54,55]. The pressure coefficient and skin friction coefficient distributions on the airfoil surface obtained by the 2O GKFS and the HO GKFS on such a coarse mesh are depicted in Fig. 7. It can be seen that the pressure coefficient distribution computed by the HO GKFS is closer to the reference data of Jawahar and Kamath [54] than the 2O GKFS. The lift and drag coefficients computed by two schemes are tabulated in Table I. As shown in this table, the results of the HO GKFS are within the range reported in the GAMM workshop [55], while the lift coefficient obtained by the 2O GKFS on this mesh is out of the range. To obtain results comparable to those of the HO GKFS, a fine mesh with 120 points on the airfoil surface and 60 points in the radial direction is required for the 2O GKFS, as illustrated in Fig. 8 and Table I. In such a circumstance, the HO GKFS is more efficient than the second-order counterpart as depicted in Fig. 9.

In the test case of biplane configuration, two NACA0012 airfoils are staggered by half a chord length in the pitchwise as well as chordwise directions. As shown in Fig. 10(a), a triangular mesh with 80 points on each airfoil surface and 15 382 cells is utilized to discretize the computational domain. Figure 10(b) displays the streamline pattern obtained by the HO GKFS. It can be found that the separation region on the upper surface of the top airfoil reveals two vortices, which is

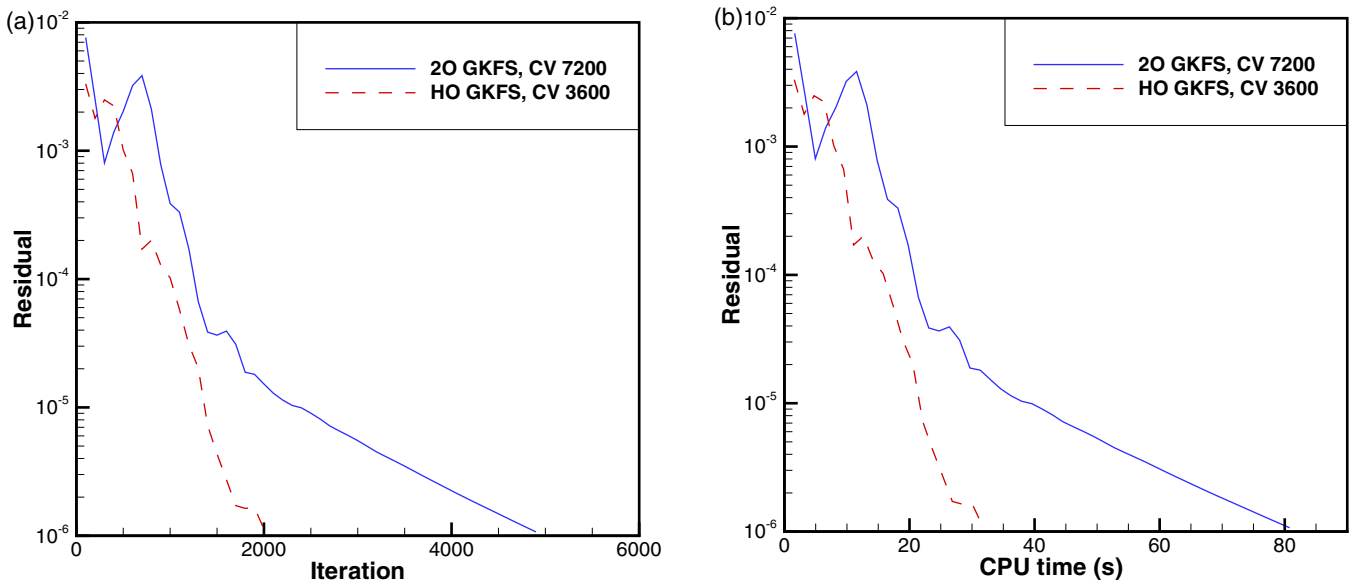

FIG. 9. (a) Convergence history and (b) CPU time for 2O GKFS and HO GKFS when achieving comparable results.

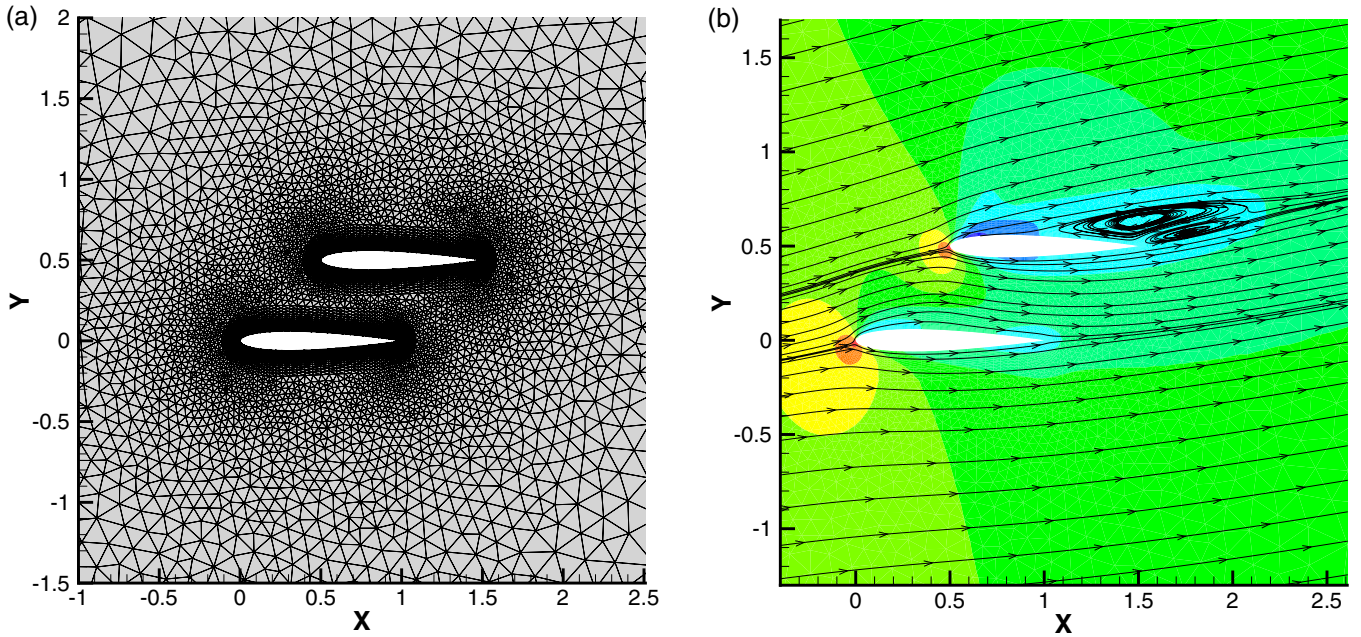

FIG. 10. (a) Computational mesh and (b) streamline pattern for transonic flow around biplane configuration.

in line with the result reported in the work of Jawahar and Kamath [54]. Figure 11 shows the comparison of pressure coefficient and skin friction coefficient distributions on the airfoil surface. There is a better coincidence between the results of the HO GKFS and the reference data [54] as compared with the results of the 2O GKFS. This test case validates the accuracy of the HO GKFS on complex geometry.

C. Case 3: Viscous shock tube problem

A good test case for validity of the HO GKFS in capturing the complicated flow structure and the shock-boundary layer interaction is the viscous shock tube problem [26,56,57]. Initially, the gas with two different states in a 2D box $[0, 1]\times[0, 0.5]$ is separated by a membrane located at $x = 0.5$,

$$(\rho, u, v, P) = \begin{cases} (120, 0, 0, 120/\gamma), & 0 < x < 0.5, \\ (1.2, 0, 0, 1.2/\gamma), & 0.5 \leqslant x \leqslant 1. \end{cases} \tag{62}$$

The symmetrical condition is applied to the upper boundary and the no-slip and adiabatic conditions are imposed on other boundaries. In our simulation, the Prandtl number is taken as $\mathrm{Pr} = 0.73$ and the Reynolds number is set as $\mathrm{Re} = 200$. The viscosity is assumed to be constant in this test example. A shock wave with Mach number of 2.37 comes out when the membrane is removed at time zero. This shock wave will move to the right and reflect on the right wall. After that, it

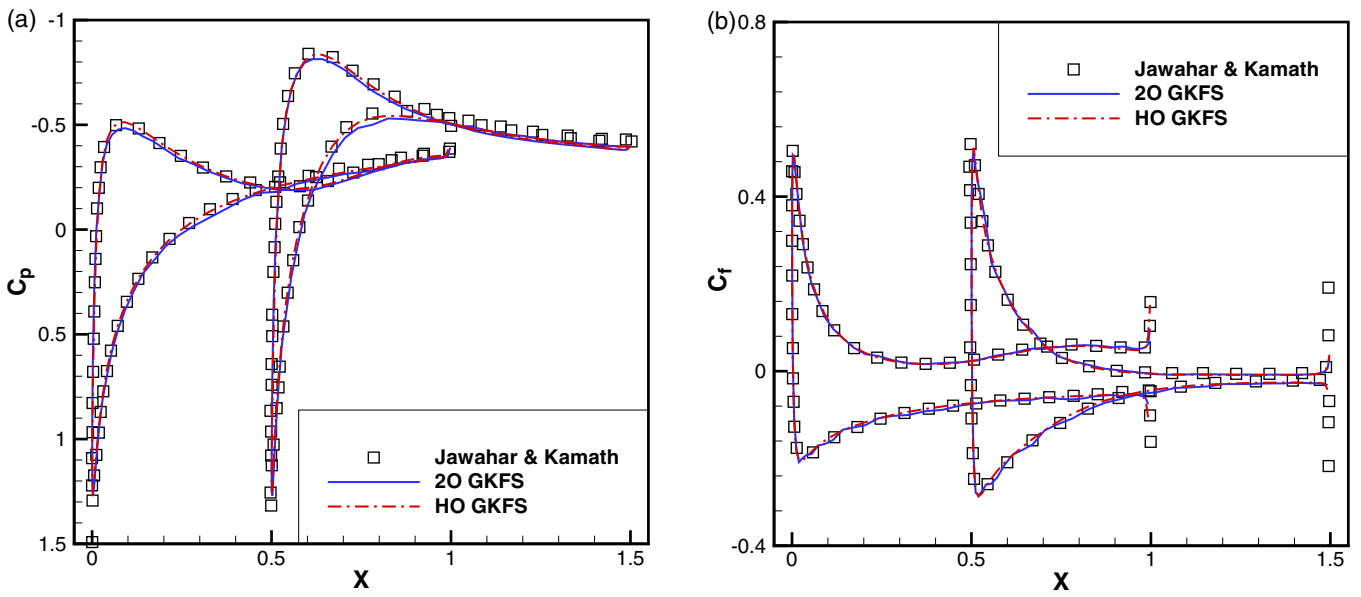

FIG. 11. (a) Pressure coefficient distribution and (b) skin friction coefficient distribution for transonic flow around biplane configuration.

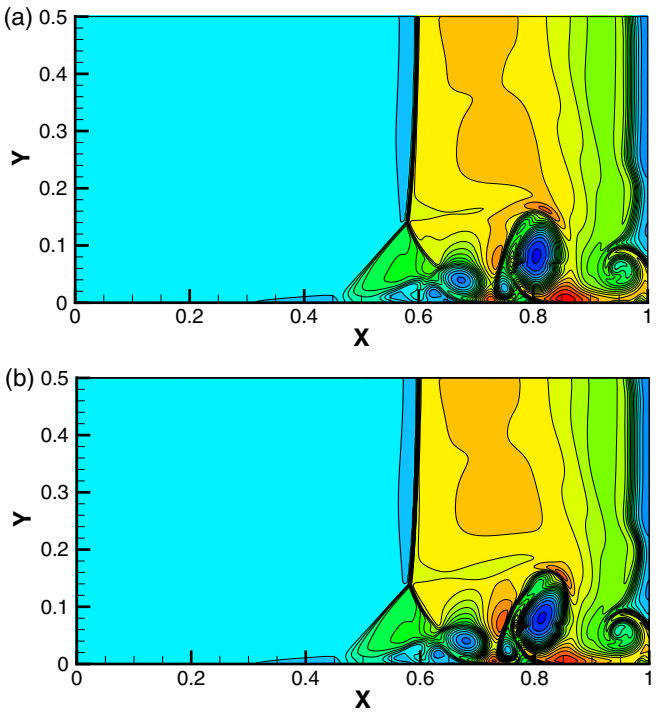

FIG. 12. Density contours for viscous shock tube problem computed by different schemes: (a) 2O GKFS, (b) HO GKFS.

will interact with the contact discontinuity. Finally, a complex 2D shock-shear-boundary layer interaction is developed near the bottom wall.

In the current simulation, both the 2O GKFS and the HO GKFS are applied. To compare with the reference data of other high-order methods [26,56,57], the same mesh size of $h = 1/500$ is used in the present calculation. Figure 12 shows the density contours obtained by two schemes. It can be seen that the HO GKFS can resolve the complex flow structure with higher resolution than the 2O GKFS and it agrees well with the results reported in Refs. [26,56,57]. The quantitative comparisons of the height of the primary vortex and the distributions of density and skin friction along the bottom wall are made in Table II and Fig. 13, respectively. Clearly, the HO GKFS outperforms the second-order counterpart, which validates the capability of the present high-order method for simulation of supersonic flows.

D. Case 4: Shock-boundary layer interaction

The interaction between an oblique shock wave and a laminar boundary layer is considered in the last test case. The shock wave hits the bottom wall at $x = 1$ with the inclined angle of 32.6°. In this problem, the free-stream Mach number is 2, the Reynolds number is 2.96×10^5, and the free-stream temperature is 117 K. The dynamic viscosity is computed by Sutherland's law. In the simulation, the computational domain is taken as a 2D box $[0, 2]\times[0, 1.27]$, which is discretized uniformly in the x direction with 101 points and nonuniformly in the y direction with 81 points or 121 points, and the height of the cell adjacent to the bottom wall is set as 0.001. The no-slip and adiabatic conditions are applied on the bottom wall, the supersonic outflow condition is used on the right boundary, the supersonic inflow condition is utilized on the left boundary with $y < \tan(32.6^\circ)$, and the far-field condition is imposed on the remaining boundaries. For the supersonic outflow condition, the density, velocity, and pressure in the ghost cell are taken to have the same values as those in the inner cell adjacent to the boundary. For the supersonic inflow condition, these flow variables are set as the free-stream state.

Figure 14 compares the pressure contours computed by the 2O GKFS and the HO GKFS on a mesh with 121 points in the y direction. It can be seen that the flow structures captured by the HO GKFS are sharper than the 2O GKFS. The quantitative comparisons of the skin friction coefficient distribution along the bottom wall and the pressure distribution along the line of $x = 1.5$ are shown in Fig. 15. Also displayed in this figure are the results obtained by the high-order FV method combined with the Roe solver (HO ROE) using the same mesh [58]. Clearly, the results of the HO GKFS agree well with those of the HO ROE, while the results of the 2O GKFS deviate from the reference data. Figure 16 depicts the skin friction coefficient distribution and the pressure distribution obtained by the 2O GKFS and the HO GKFS on a coarse mesh with 81 points in the y direction. It is found that the results of the HO GKFS on such a coarse mesh are basically in line with the reference data, while a significant discrepancy is observed for the results of the 2O GKFS.

V. CONCLUSIONS

This work presents a high-order GKFS for simulation of 2D compressible flows under the framework of the high-order FV method. Different from the conventional Navier-Stokes solvers, in which the inviscid and viscous fluxes are treated separately, the GKFS calculates these fluxes simultaneously by the local asymptotic solution to the Boltzmann equation. By approximating the local asymptotic solution with high-order accuracy, the high-order GKFS is then established and the numerical fluxes of the Navier-Stokes equations can be given explicitly. As compared with the high-order GKS, the formulations of the high-order GKFS are more concise.

TABLE II. Comparison of the height of primary vortex for viscous shock tube problem on a mesh of $h = 1/500$.

Scheme	AUSMPW+ [57]	M-AUSMPW+ [57]	WENO-GKS [26]
Height	0.163	0.166	0.165
Scheme	Compact HO GKS [26]	2O GKFS	HO GKFS
Height	0.166	0.161	0.165

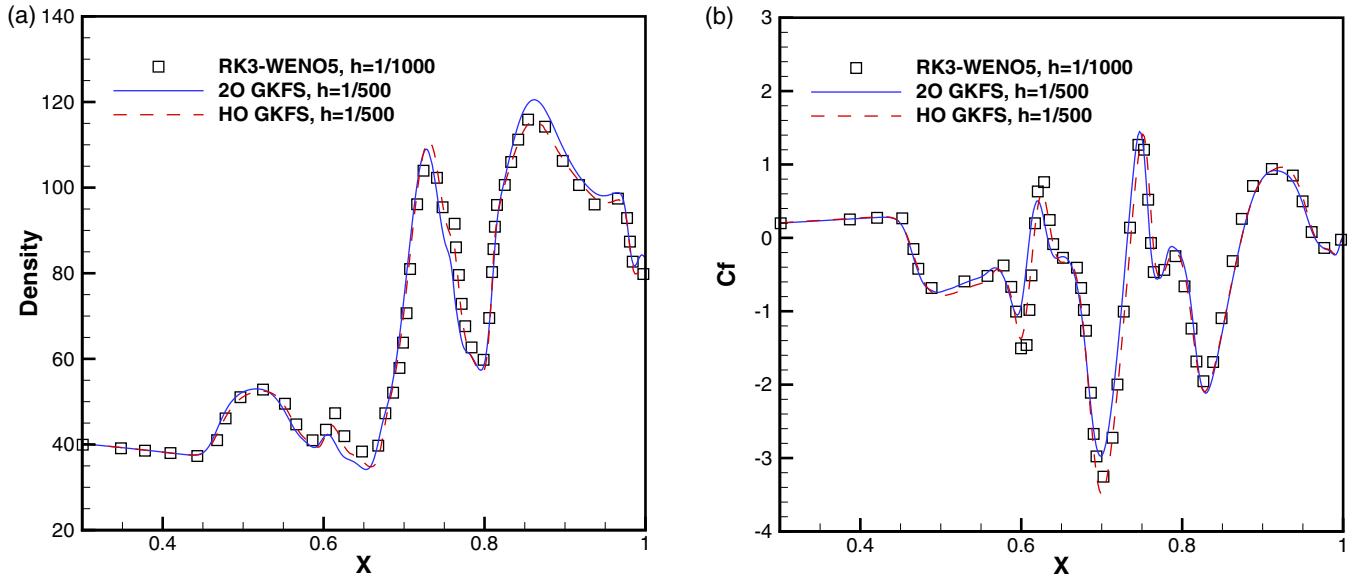

FIG. 13. (a) Density distribution and (b) skin friction distribution along the bottom wall for viscous shock tube problem obtained by the 2O GKFS and the HO GKFS.

Several test cases including the advection of density perturbation, the evolution of a 2D isentropic vortex, the transonic flow around the NACA0012 airfoil, the viscous shock tube problem, and the shock-boundary layer interaction are simulated to validate the performance of the high-order GKFS on both the quadrilateral mesh and the triangular mesh. It is shown that the high-order GKFS can achieve the desired accuracy on both meshes and it enjoys better accuracy in capturing the complex flow structures as compared with the second-order counterpart. Besides, the high-order GKFS is more efficient than the second-order one when achieving comparable accuracy of solution. Moreover, this method can be extended to three-dimensional (3D) simulations straightforwardly by applying the local asymptotic solution to the 3D Boltzmann equation to reconstruct the numerical fluxes of the Navier-Stokes equations, which will be presented in further work.

ACKNOWLEDGMENTS

The research is partially supported by the National Natural Science Foundation of China (Grants No. 11772157 and No. 11832012), the National Numerical Wind Tunnel Project of China (Grant No. NNW2019ZT2-B28), and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

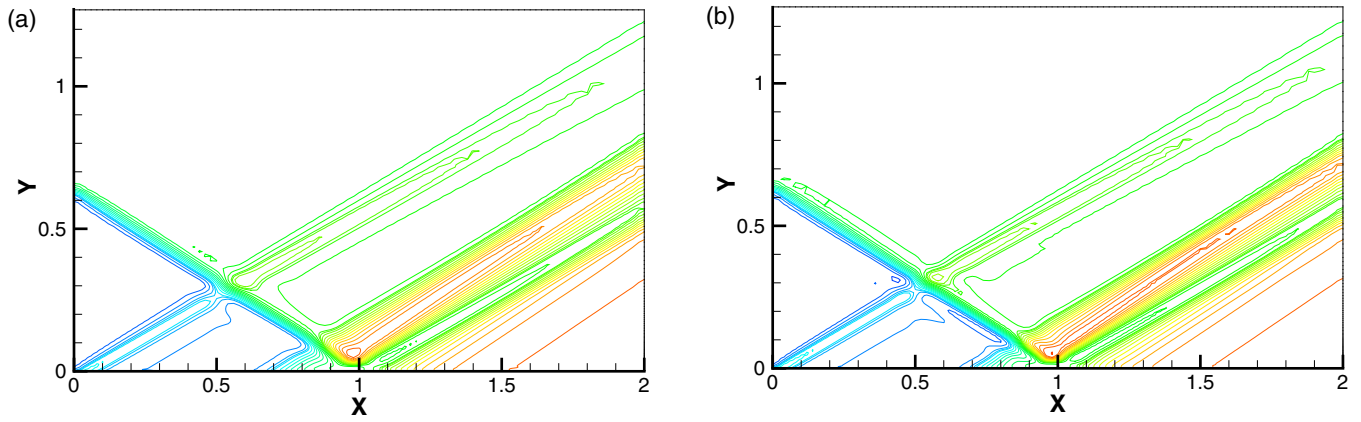

FIG. 14. Pressure contours for shock-boundary layer interaction computed by the second-order and high-order GKFS on a mesh with 100×120 cells: (a) 2O GKFS, (b) HO GKFS.

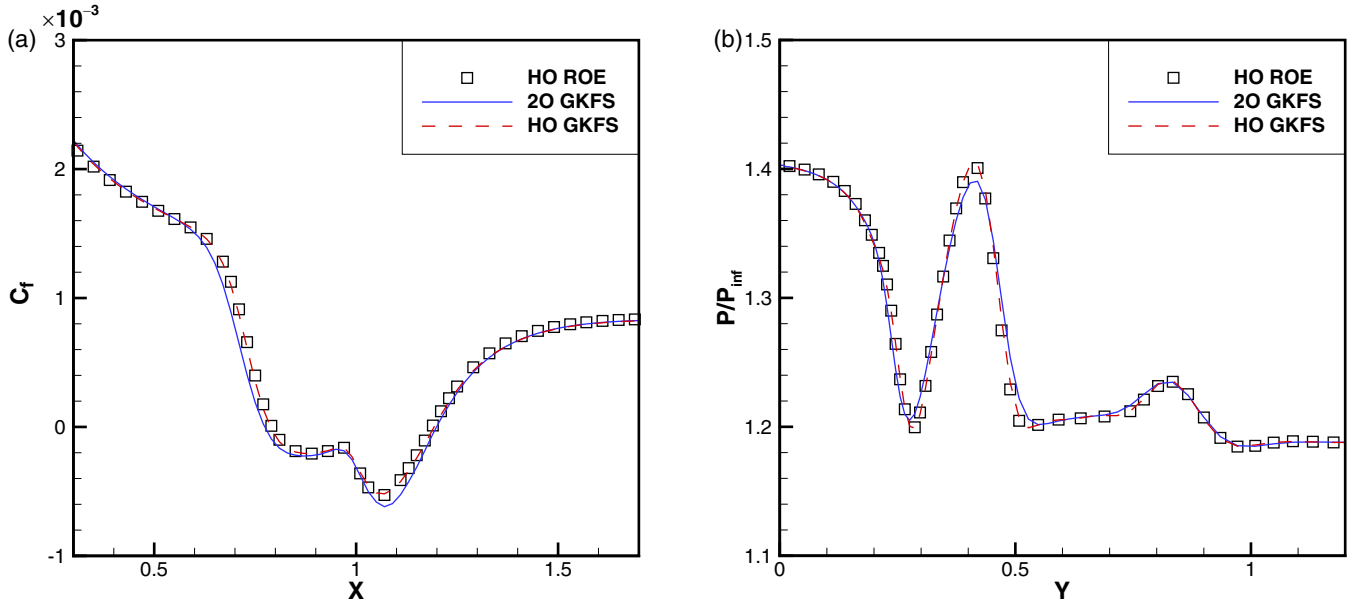

FIG. 15. (a) Skin friction coefficient distribution along the flat plate and (b) pressure distribution along $x = 1.5$ for shock-boundary layer interaction on a mesh with 100×120 cells.

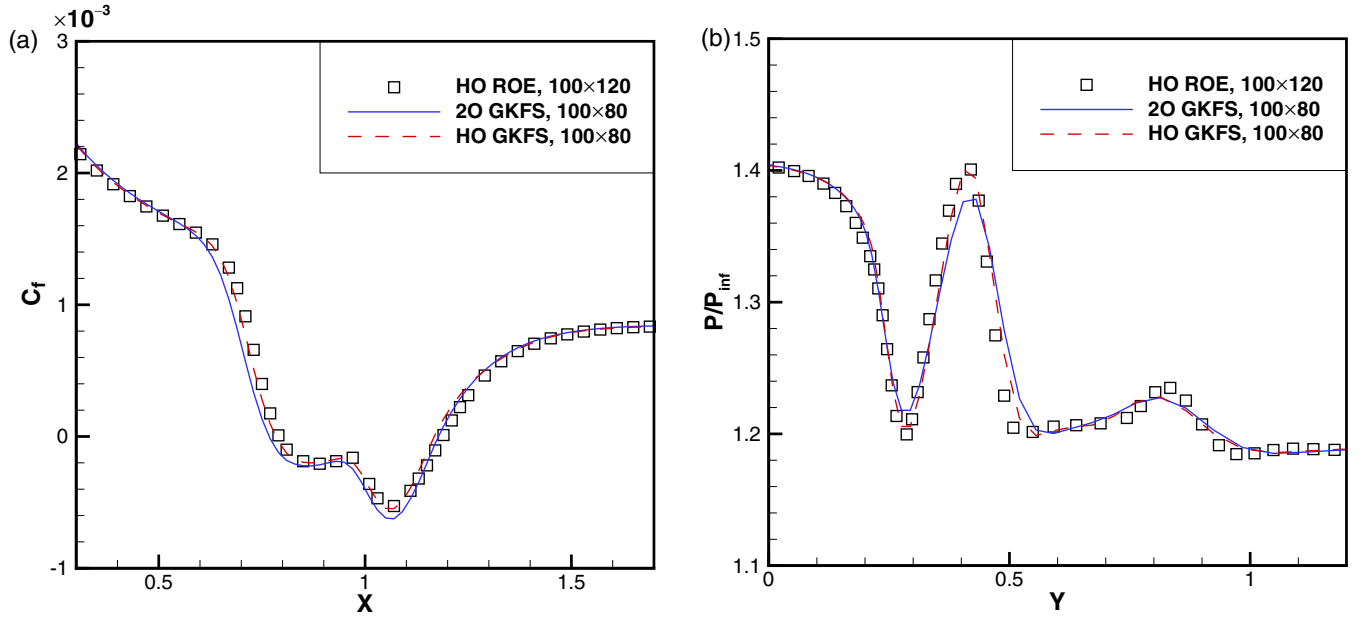

FIG. 16. (a) Skin friction coefficient distribution along the flat plate and (b) pressure distribution along $x = 1.5$ for shock-boundary layer interaction on a mesh with 100×80 cells.

APPENDIX: SOME COEFFICIENTS FOR CALCULATION OF CONSERVATIVE VARIABLES AND FLUXES AT THE CELL INTERFACE

In Eqs. (37)–(40), the definitions of $\hbar_0$, $\hbar_1$, $\hbar_2$, and $\hbar_3$ are as follows:

$$\begin{aligned}\hbar_0 = {} & a_{1,0}\beta_1 + a_{1,1}\beta_2 + a_{1,2}\beta_1\chi_1 + \tfrac{1}{2}a_{1,3}(\beta_3 + \beta_1\chi_2 + \beta_1\eta_2) \\ & + a_{2,0}\beta_0\chi_1 + a_{2,1}\beta_1\chi_1 + a_{2,2}\beta_0\chi_2 + \tfrac{1}{2}a_{2,3}(\beta_2\chi_1 + \beta_0\chi_3 + \beta_0\chi_1\eta_2),\end{aligned} \tag{A1}$$

$$\begin{aligned}\hbar_1 = {} & a_{1,0}\beta_2 + a_{1,1}\beta_3 + a_{1,2}\beta_2\chi_1 + \tfrac{1}{2}a_{1,3}(\beta_4 + \beta_2\chi_2 + \beta_2\eta_2) \\ & + a_{2,0}\beta_1\chi_1 + a_{2,1}\beta_2\chi_1 + a_{2,2}\beta_1\chi_2 + \tfrac{1}{2}a_{2,3}(\beta_3\chi_1 + \beta_1\chi_3 + \beta_1\chi_1\eta_2),\end{aligned} \tag{A2}$$

$$\hbar_2 = a_{1,0}\beta_1\chi_1 + a_{1,1}\beta_2\chi_1 + a_{1,2}\beta_1\chi_2 + \tfrac{1}{2}a_{1,3}(\beta_3\chi_1 + \beta_1\chi_3 + \beta_1\chi_1\eta_2) + a_{2,0}\beta_0\chi_2 + a_{2,1}\beta_1\chi_2 + a_{2,2}\beta_0\chi_3 + \tfrac{1}{2}a_{2,3}(\beta_2\chi_2 + \beta_0\chi_4 + \beta_0\chi_2\eta_2), \tag{A3}$$

$$\begin{aligned}\hbar_3 = {} & \tfrac{1}{2}\left[a_{1,0}\beta_3 + a_{1,1}\beta_4 + a_{1,2}\beta_3\chi_1 + \tfrac{1}{2}a_{1,3}(\beta_5 + \beta_3\chi_2 + \beta_3\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{1,0}\beta_1\chi_2 + a_{1,1}\beta_2\chi_2 + a_{1,2}\beta_1\chi_3 + \tfrac{1}{2}a_{1,3}(\beta_3\chi_2 + \beta_1\chi_4 + \beta_1\chi_2\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{1,0}\beta_1\eta_2 + a_{1,1}\beta_2\eta_2 + a_{1,2}\beta_1\chi_1\eta_2 + \tfrac{1}{2}a_{1,3}(\beta_3\eta_2 + \beta_1\chi_2\eta_2 + \beta_1\eta_4)\right] \\ & + \tfrac{1}{2}\left[a_{2,0}\beta_2\chi_1 + a_{2,1}\beta_3\chi_1 + a_{2,2}\beta_2\chi_2 + \tfrac{1}{2}a_{2,3}(\beta_4\chi_1 + \beta_2\chi_3 + \beta_2\chi_1\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{2,0}\beta_0\chi_3 + a_{2,1}\beta_1\chi_3 + a_{2,2}\beta_0\chi_4 + \tfrac{1}{2}a_{2,3}(\beta_2\chi_3 + \beta_0\chi_5 + \beta_0\chi_3\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{2,0}\beta_0\chi_1\eta_2 + a_{2,1}\beta_1\chi_1\eta_2 + a_{2,2}\beta_0\chi_2\eta_2 + \tfrac{1}{2}a_{2,3}(\beta_2\chi_1\eta_2 + \beta_0\chi_3\eta_2 + \beta_0\chi_1\eta_4)\right].\end{aligned} \tag{A4}$$

In Eqs. (41)–(44), the definitions of λ_1, λ_2, λ_3, and ℓ_3 are as follows:

$$\lambda_1 = a_{1,0}\beta_3 + a_{1,1}\beta_4 + a_{1,2}\beta_3\chi_1 + \tfrac{1}{2}a_{1,3}(\beta_5 + \beta_3\chi_2 + \beta_3\eta_2) + a_{2,0}\beta_2\chi_1 + a_{2,1}\beta_3\chi_1 + a_{2,2}\beta_2\chi_2 + \tfrac{1}{2}a_{2,3}(\beta_4\chi_1 + \beta_2\chi_3 + \beta_2\chi_1\eta_2), \tag{A5}$$

$$\lambda_2 = a_{1,0}\beta_2\chi_1 + a_{1,1}\beta_3\chi_1 + a_{1,2}\beta_2\chi_2 + \tfrac{1}{2}a_{1,3}(\beta_4\chi_1 + \beta_2\chi_3 + \beta_2\chi_1\eta_2) + a_{2,0}\beta_1\chi_2 + a_{2,1}\beta_2\chi_2 + a_{2,2}\beta_1\chi_3 + \tfrac{1}{2}a_{2,3}(\beta_3\chi_2 + \beta_1\chi_4 + \beta_1\chi_2\eta_2), \tag{A6}$$

$$\begin{aligned}\lambda_3 = {} & \tfrac{1}{2}\left[a_{1,0}\beta_4 + a_{1,1}\beta_5 + a_{1,2}\beta_4\chi_1 + \tfrac{1}{2}a_{1,3}(\beta_6 + \beta_4\chi_2 + \beta_4\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{1,0}\beta_2\chi_2 + a_{1,1}\beta_3\chi_2 + a_{1,2}\beta_2\chi_3 + \tfrac{1}{2}a_{1,3}(\beta_4\chi_2 + \beta_2\chi_4 + \beta_2\chi_2\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{1,0}\beta_2\eta_2 + a_{1,1}\beta_3\eta_2 + a_{1,2}\beta_2\chi_1\eta_2 + \tfrac{1}{2}a_{1,3}(\beta_4\eta_2 + \beta_2\chi_2\eta_2 + \beta_2\eta_4)\right] \\ & + \tfrac{1}{2}\left[a_{2,0}\beta_3\chi_1 + a_{2,1}\beta_4\chi_1 + a_{2,2}\beta_3\chi_2 + \tfrac{1}{2}a_{2,3}(\beta_5\chi_1 + \beta_3\chi_3 + \beta_3\chi_1\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{2,0}\beta_1\chi_3 + a_{2,1}\beta_2\chi_3 + a_{2,2}\beta_1\chi_4 + \tfrac{1}{2}a_{2,3}(\beta_3\chi_3 + \beta_1\chi_5 + \beta_1\chi_3\eta_2)\right] \\ & + \tfrac{1}{2}\left[a_{2,0}\beta_1\chi_1\eta_2 + a_{2,1}\beta_2\chi_1\eta_2 + a_{2,2}\beta_1\chi_2\eta_2 + \tfrac{1}{2}a_{2,3}(\beta_3\chi_1\eta_2 + \beta_1\chi_3\eta_2 + \beta_1\chi_1\eta_4)\right],\end{aligned} \tag{A7}$$

$$\ell_3 = \tfrac{1}{2}(\beta_3 + \beta_1\chi_2 + \beta_1\eta_2). \tag{A8}$$

[1] D. Chae, C. Kim, and O. H. Rho, Development of an improved gas-kinetic BGK scheme for inviscid and viscous flows, J. Comput. Phys. **158**, 1 (2000).

[2] K. Xu, A gas-kinetic BGK scheme for the Navier-Stokes equations and its connection with artificial dissipation and Godunov method, J. Comput. Phys. **171**, 289 (2001).

[3] H. Tang, T. Tang, and K. Xu, A gas-kinetic scheme for shallow-water equations with source terms, Z. angew. Math. Phys. **55**, 365 (2004).

[4] C. T. Tian, K. Xu, K. L. Chan, and L. C. Deng, A three-dimensional multidimensional gas-kinetic scheme for the Navier-Stokes equations under gravitational fields, J. Comput. Phys. **226**, 2003 (2007).

[5] L. M. Yang, C. Shu, and J. Wu, A simple distribution function-based gas-kinetic scheme for simulation of viscous incompressible and compressible flows, J. Comput. Phys. **274**, 611 (2014).

[6] J. Li, C. Zhong, Y. Wang, and C. Zhuo, Implementation of dual time-stepping strategy of the gas-kinetic scheme for unsteady flow simulations, Phys. Rev. E **95**, 053307 (2017).

[7] K. Xu, Gas-kinetic schemes for unsteady compressible flow simulations, in *Proceedings of the 29th Computational Fluid Dynamics—VKI LS 1998-03* (von Karman Institute for Fluid Dynamics, Sint-Genesius-Rode, Belgium, 1998).

[8] R. Yuan, C. Zhong, and H. Zhang, An immersed-boundary method based on the gas kinetic BGK scheme for incompressible viscous flow, J. Comput. Phys. **296**, 184 (2015).

[9] K. Xu and S. H. Lui, Rayleigh-Bénard simulation using the gas-kinetic Bhatnagar-Gross-Krook scheme in the incompressible limit, Phys. Rev. E **60**, 464 (1999).

[10] J. Kerimo and S. S. Girimaji, Boltzmann-BGK approach to simulating weakly compressible 3D turbulence: Comparison between lattice Boltzmann and gas kinetic methods, J. Turbul. **8**, N46 (2007).

[11] L. M. Yang, C. Shu, W. M. Yang, Y. Wang, and J. Wu, An immersed boundary-simplified sphere function-based gas kinetic scheme for simulation of 3D incompressible flows, Phys. Fluids **29**, 083605 (2017).

[12] K. Xu, M. Mao, and L. Tang, A multidimensional gas-kinetic BGK scheme for hypersonic viscous flow, J. Comput. Phys. **203**, 405 (2005).

[13] G. May, B. Srinivasan, and A. Jameson, An improved gas-kinetic BGK finite-volume method for three-dimensional transonic flow, J. Comput. Phys. **220**, 856 (2007).

[14] D. Zhou, Z. L. Lu, and T. Q. Guo, A three-dimensional gas-kinetic BGK scheme for simulating flows in rotating machinery, Adv. Appl. Math. Mech. **11**, 168 (2019).

[15] G. Kumar, S. S. Girimaji, and J. Kerimo, WENO-enhanced gas-kinetic scheme for direct simulations of compressible transition and turbulence, J. Comput. Phys. **234**, 499 (2013).

[16] S. Tan, Q. Li, Z. Xiao, and S. Fu, Gas kinetic scheme for turbulence simulation, Aerosp. Sci. Technol. **78**, 214 (2018).

[17] L. Pan, G. Zhao, B. Tian, and S. Wang, A gas kinetic scheme for the Baer-Nunziato two-phase flow model, J. Comput. Phys. **231**, 7518 (2012).

[18] S. Zia, M. Ahmed, and S. Qamar, A gas-kinetic scheme for six-equation two-phase flow model, Appl. Math. **5**, 453 (2014).

[19] Y. S. Lian and K. Xu, A gas-kinetic scheme for multimaterial flows and its application in chemical reactions, J. Comput. Phys. **163**, 349 (2000).

[20] T. P. Yang, J. F. Wang, L. M. Yang, and C. Shu, Development of multicomponent lattice Boltzmann flux solver for simulation of compressible viscous reacting flows, Phys. Rev. E **100**, 033315 (2019).

[21] L. Tang, Progress in gas-kinetic upwind schemes for the solution of Euler/Navier-Stokes equations-I: Overview, Comput. Fluids **56**, 39 (2012).

[22] Z. Guo, H. Liu, L. S. Luo, and K. Xu, A comparative study of the LBE and GKS methods for 2D near incompressible laminar flows, J. Comput. Phys. **227**, 4955 (2008).

[23] M. S. Liou and C. J. Steffen Jr., A new flux splitting scheme, J. Comput. Phys. **107**, 23 (1993).

[24] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* (Springer Science & Business Media, Berlin, 2013).

[25] X. S. Li and X. L. Li, All-speed Roe scheme for the large eddy simulation of homogeneous decaying turbulence, Int. J. Comput. Fluid Dyn. **30**, 69 (2016).

[26] L. Pan and K. Xu, A third-order compact gas-kinetic scheme on unstructured meshes for compressible Navier-Stokes solutions, J. Comput. Phys. **318**, 327 (2016).

[27] X. Ji, L. Pan, W. Shyy, and K. Xu, A compact fourth-order gas-kinetic scheme for the Euler and Navier-Stokes equations, J. Comput. Phys. **372**, 446 (2018).

[28] X. Ren, K. Xu, W. Shyy, and C. Gu, A multi-dimensional high-order discontinuous Galerkin method based on gas kinetic theory for viscous flow computations, J. Comput. Phys. **292**, 176 (2015).

[29] Q. Li, K. Xu, and S. Fu, A high-order gas-kinetic Navier-Stokes flow solver, J. Comput. Phys. **229**, 6715 (2010).

[30] J. Luo and K. Xu, A high-order multidimensional gas-kinetic scheme for hydrodynamic equations, Sci. China: Technol. Sci. **56**, 2370 (2013).

[31] Y. Sun, C. Shu, C. J. Teo, Y. Wang, and L. M. Yang, Explicit formulations of gas-kinetic flux solver for simulation of incompressible and compressible viscous flows, J. Comput. Phys. **300**, 492 (2015).

[32] Y. Sun, C. Shu, Y. Wang, C. J. Teo, and Z. Chen, An immersed boundary-gas kinetic flux solver for simulation of incompressible flows, Comput. Fluids **142**, 45 (2017).

[33] Y. Sun, L. M. Yang, C. Shu, and C. J. Teo, A three-dimensional gas-kinetic flux solver for simulation of viscous flows with explicit formulations of conservative variables and numerical flux, Adv. Aerodyn. **2**, 13 (2020).

[34] T. Barth, Recent developments in high order k-exact reconstruction on unstructured meshes, in *Proceedings of the 31st Aerospace Sciences Meeting and Exhibit* (AIAA, Washington, DC, 1993), p. 668.

[35] G. Hu and N. Yi, An adaptive finite volume solver for steady Euler equations with non-oscillatory k-exact reconstruction, J. Comput. Phys. **312**, 235 (2016).

[36] Y. Y. Liu, C. Shu, H. W. Zhang, and L. M. Yang, A high order least square-based finite difference-finite volume method with lattice Boltzmann flux solver for simulation of incompressible flows on unstructured grids, J. Comput. Phys. **401**, 109019 (2020).

[37] M. Svärd, M. H. Carpenter, and J. Nordström, A stable high-order finite difference scheme for the compressible Navier-Stokes equations, far-field boundary conditions, J. Comput. Phys. **225**, 1020 (2007).

[38] J. M. Duan and H. Z. Tang, High-order accurate entropy stable finite difference schemes for one- and two-dimensional special relativistic hydrodynamics, Adv. Appl. Math. Mech. **12**, 1 (2020).

[39] B. Cockburn and C. W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems, SIAM J. Numer. Anal. **35**, 2440 (1998).

[40] F. Bassi and S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations, in *Discontinuous Galerkin Methods* (Springer, Berlin, 2000), pp. 197–208.

[41] R. Kannan and Z. J. Wang, A study of viscous flux formulations for a p-multigrid spectral volume Navier Stokes solver, J. Sci. Comput. **41**, 165 (2009).

[42] Y. Sun, Z. J. Wang, and Y. Liu, High-order multidomain spectral difference method for the Navier-Stokes equations, in *Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit* (AIAA, Washington, DC, 2006), p. 301.

[43] C. W. Shu, High-order finite difference and finite volume WENO schemes and discontinuous Galerkin methods for CFD, Int. J. Comput. Fluid Dyn. **17**, 107 (2003).

[44] Z. M. Wang, J. Zhu, and N. Zhao, A new robust high-order weighted essentially nonoscillatory scheme for solving well-balanced shallow water equations, Adv. Appl. Math. Mech. **11**, 911 (2019).

[45] Y. Y. Liu, H. W. Zhang, L. M. Yang, and C. Shu, High-order least-square-based finite-difference–finite-volume method for simulation of incompressible thermal flows on arbitrary grids, Phys. Rev. E **100**, 063308 (2019).

[46] L. M. Yang, Y. Wang, Z. Chen, and C. Shu, *Lattice Boltzmann and Gas Kinetic Flux Solvers: Theory and Applications* (World Scientific, Singapore, 2020).

[47] J. H. Ferziger, M. Peric, and R. L. Street, *Computational Methods for Fluid Dynamics* (Springer Nature, Berlin, 2020).

[48] A. Haselbacher and J. Blazek, Accurate and efficient discretization of Navier-Stokes equations on mixed grids, AIAA J. **38**, 2094 (2000).

[49] F. Moukalled, L. Mangani, and M. Darwish, *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with* OPENFOAM® *and* MATLAB (Springer International Publishing, Berlin, 2015).

[50] M. Dumbser, M. Kaser, and J. Puente, Arbitrary high-order finite volume schemes for seismic wave propagation on unstructured meshes in 2D and 3D, Geophys. J. Int. **171**, 665 (2007).

[51] M. Dumbser, M. Kaser, V. A. Titarev, and E. F. Toro, Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems, J. Comput. Phys. **226**, 204 (2007).

[52] L. M. Yang, C. Shu, and J. Wu, A hybrid lattice Boltzmann flux solver for simulation of viscous compressible flows, Adv. Appl. Math. Mech. **8**, 887 (2016).

[53] C. Hu and C. W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, J. Comput. Phys. **150**, 97 (1999).

[54] P. Jawahar and H. Kamath, A high-resolution procedure for Euler and Navier-Stokes computations on unstructured grids, J. Comput. Phys. **164**, 165 (2000).

[55] M. O. Bristeau, *Numerical Simulation of Compressible Navier-Stokes Flows: A GAMM Workshop* (Springer Science & Business Media, Berlin, 2013).

[56] V. Daru and C. Tenaud, Numerical simulation of the viscous shock tube problem by using a high resolution monotonicity-preserving scheme, Comput. Fluids **38**, 664 (2009).

[57] K. H. Kim and C. Kim, Accurate, efficient and monotonic numerical methods for multi-dimensional compressible flows: Part I: Spatial discretization, J. Comput. Phys. **208**, 527 (2005).

[58] S. S. Kim, C. Kim, O. H. Rho, and S. Hong, Cure for shock instability-development of an improved Roe scheme, in *Proceedings of the 40th AIAA Aerospace Sciences Meeting and Exhibit* (AIAA, Washington, DC, 2002), p. 548.