A parallel Runge–Kutta discontinuous Galerkin solver for rarefied gas flows based on 2D Boltzmann kinetic equations

CrossMark

Wei Su [a,b], Alina A. Alexeenko [b,*], Guobiao Cai [a]

[a] School of Astronautics, Beihang University, Beijing 100191, China
[b] School of Aeronautics & Astronautics, Purdue University, West Lafayette, IN 47907, USA

A R T I C L E   I N F O

A B S T R A C T

The high-order Runge–Kutta Discontinuous Galerkin (RKDG) method is applied to solve the 2D Boltzmann kinetic equations. A conservative DG type discretization of the non-linear collision relaxation term is formulated for both the Bhatnagar–Gross–Krook and the ellipsoidal statistical kinetic models. Verification is carried out for a steady and an unsteady oscillatory 1D Couette flows, a 2D conduction problem as well as for a 2D long microchannel flow by comparison with the DSMC and analytical solutions. The computational performance of the RKDG method is compared with a widely used second-order finite volume method. The RKDG method has up to 3rd-order spatial accuracy and up to 4th-order time accuracy and is more efficient than the finite volume approach. The parallelization by domain decomposition in physical space is implemented and parallel performance is evaluated. It is shown that 2nd order RKDG is over 15 times faster than the 2nd-order FVM method for the Couette flow test case. The high-order RKDG method is especially attractive for solution of low-speed and unsteady rarefied flows.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

The non-equilibrium rarefied gas flows are often encountered in high-altitude aerodynamics, vacuum technology and micro-electro-mechanical systems (MEMS). Accurate physical models and efficient numerical methods for solution of Boltzmann equation are required for predicting the non-equilibrium flow phenomena encountered in such rarefied flows. The degree of rarefaction is characterized by the Knudsen number, the ratio of the mean free path λ to the flow characteristic dimension L [1]. For Knudsen numbers much smaller than one, the Navier–Stokes equations are valid and the continuum computational fluid dynamics (CFD) methods could be employed for numerical simulations. At $Kn > 0.01$ the no-slip boundary condition breaks down. For larger Knudsen numbers, the shear stress and heat flux in the hydrodynamic models cannot be simply expressed in terms of the lower-order macroscopic quantities and models of kinetic theory based on the Boltzmann equation for the velocity distribution function of molecules are required.

In gas kinetic theory, the motion of molecules in dilute gas is mathematically described by the integro-differential Boltzmann equation which is valid for the entire range of Knudsen number.

However, due to the high dimensionality of the phase space and the complexity of the nonlinear collision integral term, the full Boltzmann equation is amenable to analytical solutions only for a few special cases. The direct simulation Monte Carlo (DSMC) method is the most widely used numerical method for solution of the Boltzmann equation [1]. It is a particle-based method which evaluates the molecular behavior stochastically according to a prescribed binary interaction model. The DSMC method is absolutely stable and amenable to various high-fidelity physical and chemical models. Since the early 1960s, it has been successfully applied to a wide range of problems in rarefied gas dynamics, especially for the hypersonic flows. However, the major features of the stochastic method, including the statistical noise, the explicit and low-order time integration and the strict requirements for resolution of the mean free path and collision time scales by appropriate cell sizes and time steps, make it very inefficient in simulating low-speed flows, unsteady flows and near continuum flows [2–4]. Many improved particle-based approaches have been proposed to deal with computational cell size and time step limitations of DSMC, such as implicit Monte Carlo method [5], equilibrium particle method [6] and low diffusion particle method [7], and IP-DSMC method [8]. Recently, the variance reduction methods are developed to significantly reduce statistical uncertainty of the DSMC method for the simulations of flows in the small Mach number limit [9,10], which has been proven to solve the Boltzmann equations [11]. However, the particle nature of these stochastic

methods hinders coupling to continuum CFD and deterministic structural, thermal and electrostatic modeling [12,13].

The deterministic numerical simulation (DNS) approaches relaying on the discrete ordinate method [14,15] are attractive alternatives to overcome the DSMC limitation. These methods adopt a numerical quadrature to approximate the integration with respect to molecular velocity on a discrete set of velocities. Then, the distribution functions which are continuous in physical space and time but discrete in velocity space are solved by CFD methods involving direct numerical solution procedures for the governing equations through physical-space discretization. The DNS methods are free of statistical noise and admit high-order formulations in phase space as well as implicit time marching. Other advantages of the deterministic approaches include the simplified application of subsonic boundary conditions and procedures to couple with other deterministic solvers for fluid and solid domains. Due to the multi-dimensionality of the velocity distribution function in the phase space which spans both spatial and velocity coordinates, high-order discretization is critically important for direct solvers to reduce memory and computational time. The high-order finite difference method (FDM) and finite volume method (FVM) have been developed for the Boltzmann based equations [16–22].

Recently, another class of high-order numerical method, the discontinuous Galerkin (DG) finite element method has become popular. The first DG method was introduced to solve the steady neutron transport equation in 1973 by Reed and Hill [23]. Then the so-called Runge–Kutta DG (RKDG) method was well developed for nonlinear time-dependent equations by Cockburn and Shu [24–27]. One of the most attractive features of this method is its ability to naturally obtain fluxes at the boundaries with the same high-order accuracy as in the interior of the domain [23,28]. This is especially important for applications of non-equilibrium flow analysis where stress and heat flux distributions on the solid boundaries are the main quantities of interest. Comparing to other high-order methods such as FVM and FDM, it also provides advantages including easy formulation on arbitrary meshes, straightforward implementation of boundary conditions, as well as efficient parallel computation due to compactness of the scheme [29]. Although this method is more efficient in discretization than the FDM or FVM with same order of accuracy [28], the RKDG method is seldom used to solve the rarefied gaseous flows. Liu and Xu developed a DG solver for 2D moment equations without collision term on rectangle meshes [30]. Gobbert solved 2D/2V and 3D3V linear Boltzmann equation using DG method in transient regime [31]. Baker and Hadjiconstantinou formulated a DG discretization for Boltzmann equation and used variance reduction approach to evaluate the weak form of the collision integral, which resolves the state with small deviations from the equilibrium. The test problems were performed in zero and one spatial dimensions [32]. Alekseenko proposed a velocity discretization based on DG method for 1D BGK model kinetic equation [33].

As an extension of our previous work [34], we present a conservative RKDG formulation and parallel solver for 2D/2V and 2D/3V kinetic equations and study its performance. The strictly conservative formulation of the collision term is obtained on arbitrary triangular meshes for the first time. By introducing a set of discrete velocities, the integro-differential equation is replaced by a system of differential equations, which are discretized in physical space using a DG method. The obtained ordinary-differential system is discretized in time by a class of explicit Runge–Kutta method [35]. A conservative DG type discretization of the non-linear collision relaxation term is formulated for both the Bhatnagar–Gross–Krook and the ellipsoidal statistical models. Computational performance is investigated by comparing with a finite volume method. The remainder of the paper is organized as follows. In Section 2 the numerical scheme is described including the velocity

discretization, DG method, time marching, numerical flux and boundary conditions. The approach for domain decomposition for the parallel implementation is discussed in Section 3. Four different verification problems are presented in Section 4 together with the analysis of computational performance. Conclusions are summarized in Section 5.

2. Numerical method

2.1. The governing equations

In the kinetic model, the original Boltzmann equation is reduced to a relatively simpler partial differential equation by introducing the collision relaxation term. Two frequently used forms are the Bhatnagar–Gross–Krook (BGK) [36] and the ellipsoidal statistical BGK (ES-BGK) [37] models. The kinetic model equations are written as

$$\frac{\partial f}{\partial t} + \mathbf{c} \cdot \frac{\partial f}{\partial \mathbf{r}} = \nu(f_E - f) \tag{1}$$

where f is the distribution function of molecular velocities, which is defined so that $f(t,\mathbf{r},\mathbf{c})\mathrm{d}\mathbf{r}\mathrm{d}\mathbf{c}$ is the number of molecules at time instant t, with velocity components lying within the limits $\mathbf{c}$ and $\mathbf{c} + \mathrm{d}\mathbf{c}$, and spatial coordinates lying within the limits $\mathbf{r}$ and $\mathbf{r} + \mathrm{d}\mathbf{r}$. Here f_E denotes a suitable distribution under equilibrium. In the standard BGK model which corresponds to the gas Prandtl number $Pr = 1$, f_E is the local Maxwell distribution, while in ES-BGK model f_E is a local anisotropic Gaussian and ν is the collision frequency given as

$$\nu = Pr\frac{P}{\mu} \tag{2}$$

where P is the local gas pressure and μ is the viscosity coefficient. An arbitrary viscosity law could be specified and a frequently-used power-law $\mu = \mu_{ref}(T/T_{ref})^\omega$ is applied here with the reference properties μ_{ref}, T_{ref} and ω given by Bird [1]. The collision frequency ν involves the Prandtl number Pr, which is a free parameter allowing the ES-BGK collision model to reproduce both the viscosity and thermal conductivity corresponding to an arbitrary Prandtl number [38]. The Prandtl number is equal to 2/3 for monatomic gases, while the BGK model is a special case of the ES-BGK for $Pr = 1$. Note that in general collision frequency models depending on velocity could also be implemented within kinetic models for high non-equilibrium flows [20].

For two-dimensional flows, the dimensionality of phase space could be reduced by integrating the distribution functions over the velocity space with respect to the component c_z with weighting factors 1 and c_z^2

$$f_1(t,x,y,c_x,c_y) = \int_{-\infty}^{\infty} f(t,x,y,\mathbf{c})\mathrm{d}c_z \tag{3}$$

$$f_2(t,x,y,c_x,c_y) = \int_{-\infty}^{\infty} c_z^2 f(t,x,y,\mathbf{c})\mathrm{d}c_z \tag{4}$$

The reduced system has only five independent variables, i.e. two spatial and two velocity components plus the time variable. For the sake of generality for the three-dimensional simulations, we still keep a 2V/3V system and denote the full-distribution function f as f_0. For the numerical solution of the system, it is convenient to non-dimensionalize all variables and functions. Let L, n_0 and T_0 be reference dimensional values of the length, number density and temperature. Then the reference velocity is $u_0 = (2RT_0)^{1/2}$, the reference time is L/u_0, the reference pressure is $P_0 = n_0kT_0$, and the reference viscosity is n_0kT_0L/u_0, where R is the gas constant and k is the Boltzmann constant. Finally, if we non-dimensionalize the full velocity-distribution function f_0 by $n_0/(2RT_0)^3$, the reduced

distribution function f_1 and f_2 respectively by $n_0/(2RT_0)$ and n_0, the non-dimensionalized system is obtained as

$$\frac{\partial \tilde{f}_p}{\partial \tilde{t}} + \tilde{c}_x \frac{\partial \tilde{f}_p}{\partial \tilde{x}} + \tilde{c}_y \frac{\partial \tilde{f}_p}{\partial \tilde{y}} = \tilde{v}(\tilde{f}_E - \tilde{f}), \quad p = 0, 1 \text{ or } 2 \tag{5}$$

The non-dimensional macroscopic gas flow parameters are obtained by integrating the distribution functions over the velocity phase:

$$\tilde{n} = \int_{-\infty}^{\infty} \tilde{f}_{0,1} d\tilde{\boldsymbol{c}} \tag{6}$$

$$\tilde{\boldsymbol{u}} = \frac{1}{\tilde{n}} \int_{-\infty}^{\infty} \tilde{\boldsymbol{c}} \tilde{f}_{0,1} d\tilde{\boldsymbol{c}} \tag{7}$$

$$\tilde{T} = \frac{2}{3\tilde{n}} \int_{-\infty}^{\infty} (\tilde{\boldsymbol{c}} - \tilde{\boldsymbol{u}})^2 \tilde{f}_0 d\tilde{\boldsymbol{c}}, \ \tilde{T} = \frac{2}{3\tilde{n}} \int_{-\infty}^{\infty} [(\tilde{\boldsymbol{c}} - \tilde{\boldsymbol{u}})^2 \tilde{f}_1 + \tilde{f}_2] d\tilde{\boldsymbol{c}} \tag{8}$$

In the remainder of the paper, we will omit the tilde symbols denoting non-dimensional variables.

2.2. Discrete velocity model

The velocity space discretization is implemented using both Cartesian and spherical meshes [39]. The Cartesian type consists of discretization of the velocity component c_x with total N_1 uniform velocity abscissas $c_x(j_1) = c_{x,\min} + (j_1 - 1)\Delta c_x + 1/2\Delta c_x$, $j_1 = 1,\ldots,N_1$, where $\Delta c_x = (c_{x,\max} - c_{x,\min})/N_1$. The components c_y and c_z are discretized in a similar way. The limiting values $\boldsymbol{c}_{\max}$ and $\boldsymbol{c}_{\min}$ must be chosen carefully to ensure that the effects of the tails of the distribution function are negligible. The spherical mesh involves high order discretization in the velocity magnitude and has no bound limitation. We apply here Gaussian–Laguerre quadrature up to 16th order in velocity magnitude and both 3/8th Simpson rule and uniform abscissas in velocity angles. The velocity nodes are stored in an array, where $\boldsymbol{c}^j(c_x^{j1}, c_y^{j2}, c_z^{j3})$ is the jth element of the array. If we denote $f_p^j(t,x,y) = f_p(t,x,y,\boldsymbol{c}^j)$, the governing equation is then transformed into the system of $N = N_1 \times N_2 \times N_3$ or $N = 2 \times N_1 \times N_2$ equations

$$\frac{\partial f_p^j}{\partial t} + c_x^j \frac{\partial f_p^j}{\partial x} + c_y^j \frac{\partial f_p^j}{\partial y} = v(f_E^j - f_p^j), \quad p = 0, 1 \text{ or } 2 \tag{9}$$

The macroscopic parameters such as number density n, velocity $\boldsymbol{u}$ and temperature T are evaluated through numerical quadratures as follows:

$$n = \langle f_{0,1}^j \rangle_d \tag{10}$$

$$\boldsymbol{u} = \frac{1}{n} \langle \boldsymbol{c} f_{0,1}^j \rangle_d \tag{11}$$

$$T = \frac{2}{3n} \langle (\boldsymbol{c} - \boldsymbol{u})^2 f_0^j \rangle_d, \ T = \frac{2}{3n} \langle (\boldsymbol{c} - \boldsymbol{u})^2 f_1^j + f_2^j \rangle_d \tag{12}$$

where the operator $\langle \cdot \rangle_d$ denotes the integration over the velocity space with the chosen numerical quadrature rule.

2.3. Discontinuous Galerkin formulation and time discretizations

We use the discontinuous Galerkin method to discretize the system in physical phase. The 2D computational domain Ω_S is partitioned into M triangular elements K_i (i is the index of the triangle elements) to handle arbitrary geometries. Then approximate solutions of f_p^j are sought in the finite element space of piecewise polynomials within each triangle K_i

$$f_p^j(t,x,y) = \sum_{l=1}^{k} F_{p,i}^{j,l}(t) \varphi_i^l(x,y) \tag{13}$$

where $\varphi_i^l(x,y)$ is the basis function supported in K_i and k is the total number of the basis functions, while $F_{p,i}^{j,l}(t)$ is the respective degree

of freedom. In this work, we present the piecewise linear and piecewise quadratic approximations with 2nd-order and 3rd-order spatial accuracy, respectively. For the 2nd-order case, the three basis functions are linear polynomials which yield the value 1 at one of the midpoints of the edges of K_i and the value 0 at the midpoints of the other two edges. For the 3rd-order case, the six basis functions are quadratic polynomials which yield the value 1 at one of the six points (the three midpoints of edges and three vertices) in K_i and the value 0 at the remaining five points [27].

In order to determine the degrees of freedom, standard techniques of finite element formulations are applied to obtain the weak formulations of the governing system, which is expressed as

$$\sum_{l=1}^{k} M_{ml} \frac{d}{dt} F_{p,i}^{j,l}(t) + \sum_{e \in \partial K_i} \int_e h_{e,K_i}(t,x,y) \varphi_i^m(x,y) d\Gamma$$

$$- c_x^j \sum_{l=1}^{k} F_{p,i}^{j,l}(t) Q_{ml}^x - c_y^j \sum_{l=1}^{k} F_{p,i}^{j,l}(t) Q_{ml}^y$$

$$= \int_{K_i} v \left(f_{E,p}^{i,j} - \sum_{l=1}^{k} F_{p,i}^{j,l}(t) \varphi_i^l(x,y) \right) \varphi_i^m(x,y) dx dy, \ m = 1,\ldots k \tag{14}$$

where $h_{e,Ki}(t,x,y)$ is the numerical flux at the edge e of the triangle K_i, with matrices M_{ml} and Q_{ml} are defined as

$$M_{ml} = \int_{K_i} \varphi_i^m(x,y) \varphi_i^l(x,y) dx dy \tag{15}$$

$$Q_{ml}^x = \int_{K_i} \varphi_i^l(x,y) \frac{\partial}{\partial x} \varphi_i^m(x,y) dx dy,$$

$$Q_{ml}^y = \int_{K_i} \varphi_i^l(x,y) \frac{\partial}{\partial y} \varphi_i^m(x,y) dx dy \tag{16}$$

Finally, the resulting system of ordinary differential equations (ODEs) is required to be discretized in time with a method that is at least 2nd and 3rd-order accurate for 2nd and 3rd-order DG schemes, respectively [27]. This is done by a special class of explicit total variation diminishing (TVD) Runge–Kutta methods [35]. As any explicit method, the RKDG scheme has stability restrictions required by the CFL condition:

$$|\boldsymbol{c}^j|_{\max} \frac{\Delta t}{h} \leqslant Const \tag{17}$$

where h is the element size and $Const$ is a constant depends on the order of spatial and time discretized schemes. In one spatial dimension, the p-th order DG space discretization combined with a p-th order RK time marching was proven to be stable with $Const = 1/[2(p - 1) + 1]$ [40]. However, for the cases with higher dimension, the CFL condition is highly dependent on the pattern of the spatial element and cannot be summarized in a simple inequality [29,41]. In this work, h is set as the minimum triangle height and $Const$ is set as 0.3 for 2nd-order scheme and 0.2 for 3rd-order one. This is a safe estimation. Finally, the integrals that appear in the above system are numerically estimated using quadrature rules [42].

2.4. Numerical flux and boundary conditions

At the edge e of the triangles, the discontinuity of the values of f_p^j allows the use of the numerical fluxes, which are based on exact or approximate Riemann solvers, to replace the real ones. Two-point first order monotonic fluxes are widely used due to their simplicity [43]. At any point $\boldsymbol{r}_e$ which belongs to the edge e, these numerical fluxes $h_{e,Ki}(t,\boldsymbol{r}_e)$ are dependent on both the approximate solutions obtained from the interior of the element K_i, i.e. $f_p^j(t, \boldsymbol{r}_{e,\text{int}(Ki)})$, and the ones obtained from the exterior of the element, i.e. $f_p^j(t, \boldsymbol{r}_{e,\text{ext}(Ki)})$ [27]. Here the simple upwind flux is used:

$$h_{e,K_i}(t, \mathbf{r}_e) = \begin{cases} \mathbf{c}^j \cdot \mathbf{n}_{e,K_i} f_p^j(t, \mathbf{r}_{e,\text{int}(K_i)}), & \mathbf{c}^j \cdot \mathbf{n}_{e,K_i} \geqslant 0 \\ \mathbf{c}^j \cdot \mathbf{n}_{e,K_i} f_p^j(t, \mathbf{r}_{e,\text{ext}(K_i)}), & \mathbf{c}^j \cdot \mathbf{n}_{e,K_i} < 0 \end{cases} \qquad (18)$$

where $\mathbf{n}_{e,Ki}$ is the outward unit normal to the edge e.

The boundary values $f_p^j(t, \mathbf{r}_{e,\text{ext}(Ki)})$ should be specified at the boundary edges. Five different types of boundary conditions are described below, namely, the symmetry boundary, specular-diffuse moving wall with given accommodation coefficient, periodic boundaries, far-pressure inlet/outlet boundaries, and supersonic inlet/outlet boundaries.

2.4.1. Symmetry boundary

For each incident discrete velocity $\mathbf{c}^j$, the reflected velocity $\mathbf{c}^{jr}$ is found as

$$\mathbf{c}^{jr} = \mathbf{c}^j - 2(\mathbf{c}^j \cdot \mathbf{n}_{e,K_i})\mathbf{n}_{e,K_i} \qquad (19)$$

The index jr associated with the reflected velocity is sought from the minimum of the product $[(\mathbf{c}^{jr} - \mathbf{c}^j)\cdot(\mathbf{c}^{jr} - \mathbf{c}^j)]$ over all $\mathbf{c}^j$. Finally, the boundary values are calculated as

$$f_p^j(t, \mathbf{r}_{e,\text{ext}(K_i)}) = f_p^{jr}(t, \mathbf{r}_{e,\text{int}(K_i)}) \qquad (20)$$

2.4.2. Wall boundary

A widely used Maxwell specular-diffuse wall model is implemented as follows. Given a wall at temperature T_w, velocity u_w and a specified gas–solid accommodation coefficient α, the boundary values are computed as

$$f_p^j(t, \mathbf{r}_{e,\text{ext}(K_i)}) = \alpha f_{p,w}^j + (1 - \alpha) f_{p,\text{symmetry}}^j \qquad (21)$$

where $f_{p,\text{symmetry}}^j$ is the symmetric value given as Eq. (20), and $f_{p,w}^j$ are the Maxwellian distribution functions at the wall conditions

$$f_{0,w}^j = \frac{n_w}{(\pi T_w)^{3/2}} \exp\left[-\frac{1}{T_w}(\mathbf{c}^j - \mathbf{u}_w)^2\right] \qquad (22)$$

The wall number density n_w is calculated from the conservation of mass:

$$\sum_{(\mathbf{c}^j - \mathbf{u}_w)\cdot\mathbf{n}_{e,K_i} \geqslant 0} (\mathbf{c}^j - \mathbf{u}_w) \cdot \mathbf{n}_{e,K_i} f_0^j(t, \mathbf{r}_{e,\text{int}(K_i)})$$
$$+ \sum_{(\mathbf{c}^j - \mathbf{u}_w)\cdot\mathbf{n}_{e,K_i} < 0} (\mathbf{c}^j - \mathbf{u}_w) \cdot \mathbf{n}_{e,K_i} f_{0,w}^j = 0 \qquad (23)$$

2.4.3. Periodic/periodic shadow boundary

The boundary values at the periodic/periodic shadow edge are equal to the approximate solutions evaluated at the relative edge in the periodic edge pair

$$f_p^j(t, \mathbf{r}_{e,\text{ext}(K_i)}) = f_p^j(t, \mathbf{r}_{ep,\text{int}(K_p)}) \qquad (24)$$

where $\mathbf{r}_{ep}$ and K_p are the relative point and triangle associated with the point $\mathbf{r}_e$ and triangle K_i respectively through the periodic connection.

2.4.4. Far-pressure inlet/outlet boundary

The boundary values for velocity directions incoming to the domain are set to equilibrium values at given density n_0, temperature T_0 and zero velocity:

$$f_0^j(t, \mathbf{r}_{e,\text{ext}(K_i)}, t) = \frac{n_0}{(\pi T_0)^{3/2}} \exp\left[-\frac{1}{T_0}\mathbf{c}^{j2}\right] \qquad (25)$$

2.4.5. Supersonic inlet/outlet boundary

If the flow is in the supersonic regime, the one-dimensional characteristic theory indicates that there will be three incoming

characteristic lines along which information propagates at the inlet and no incoming characteristic line at the outlet. Therefore, density n_0, temperature T_0 and velocity $\mathbf{u}_0$ should be freely specified at the inlet and the boundary values of the velocity distribution functions are given as Maxwellian

$$f_0^j(t, \mathbf{r}_{e,\text{ext}(K_i)}, t) = \frac{n_0}{(\pi T_0)^{3/2}} \exp\left[-\frac{1}{T_0}(\mathbf{c}^j - \mathbf{u}_0)^2\right] \qquad (26)$$

The boundary values at the outlet are obtained from the interior:

$$f_p^j(t, \mathbf{r}_{e,\text{ext}(K_i)}) = f_p^j(t, \mathbf{r}_{e,\text{int}(K_i)}) \qquad (27)$$

Note that non-Maxwellian distribution functions could also be used corresponding to the Navier–Stokes solution with stresses and heat flux computed by the CFD solver for coupled continuum-rarefied calculations.

2.5. Conservative discretization of the collision term

In order to complete the discontinuous Galerkin formulation of the kinetic model equations, we need to specify the discrete equilibrium distribution functions $f_{E,p}^{i,j}$ at each intermediate step of the Runge–Kutta process such that the mass, momentum and energy conservations are enforced in the collision relaxation term. The preservation of these collision invariants is essential for accurate numerical solution [44]. The collision term has the form

$$R_{m,p}^{i,j} = \int_{K_i} \nu\left(f_{E,p}^{i,j} - \sum_{l=1}^{k} F_{p,i}^{j,l}(t)\varphi_i^l(x,y)\right)\varphi_i^m(x,y)\mathrm{d}x\mathrm{d}y \qquad (28)$$

The local equilibrium distribution functions are defined in each triangle element. The function for the BGK type equilibrium equation is chosen in the form

$$f_{E,0}^{i,j} = \exp[a_1 - a_2(\mathbf{c}^j - \mathbf{u})^2 + a_3(c_x^j - u) + a_4(c_y^j - v)] \qquad (29)$$

where $\mathbf{u}(u,v)$ is the macroscopic flow velocity. The conservation is achieved by specifying the condition that the equilibrium functions satisfy the discretized version of the following conservation equations: $\int S d\mathbf{c} = 0$, $\int c_x S d\mathbf{c} = 0$, $\int c_y S d\mathbf{c} = 0$ and $\int (\mathbf{c} - \mathbf{u})^2 S d\mathbf{c} = 0$, where $S = \nu(f_{E,p} - f_p)$ is the collision term. The discrete set of the conservation equations for the full equilibrium distribution is shown below

$$\langle f_{E,0}^{i,j}\rangle_d - n = 0 \qquad (30)$$

$$\langle \mathbf{c}^j f_{E,0}^{i,j}\rangle_d - n\mathbf{u} = 0 \qquad (31)$$

$$\langle \mathbf{c}^{j2} f_{E,0}^{i,j}\rangle_d - n\left(\mathbf{u}^2 + \frac{3}{2}T\right) = 0 \qquad (32)$$

Similarly, the ES-BGK type equilibrium equation is of the form

$$f_{E,0}^{i,j} = \exp\left[a_1 - a_2(c_x^j - u)^2 + a_3(c_x^j - u) - a_4(c_y^j - v)^2 \right.$$
$$\left. + a_5(c_y^j - v) + a_6 c_x^j c_y^j - a_7 c_z^{j2}\right] \qquad (33)$$

Besides the conservation of mass and x, y-momentum, the following energy conservation equations are also included

$$\langle c_y^{j2} f_{E,0}^{i,j}\rangle_d - n(v^2 + T_{yy}) = 0 \qquad (34)$$

$$\langle c_x^{j2} f_{E,0}^{i,j}\rangle_d - n(u^2 + T_{xx}) = 0 \qquad (35)$$

$$\langle c_z^{j2} f_{E,0}^{i,j}\rangle_d - nT_{zz} = 0 \qquad (36)$$

$$\langle c_x^j c_y^j f_{E,0}^{i,j}\rangle_d - n(uv + T_{xy}) = 0 \qquad (37)$$

where the normal components are $T_{xx} = \langle(c_x^j - u)^2[(1 - 1/Pr)f_{E,0}^j + 1/Pr f_{E,0,\text{BGK}}^j]\rangle_d/n$ and the cross-terms are $T_{xy} = \langle(c_x^j - u)(c_y^j - v)[(1 - 1/Pr)f_{E,0}^j + 1/Pr f_{E,0,\text{BGK}}^j]\rangle_d/n$. Pr is the Prandtl number and $f_{E,0,\text{BGK}}^j$ the BGK equilibrium distribution function.

In order to be consistent with the weak formulation of the DG method and to retain high order accuracy, the unknown coefficients a_s are sought in each triangle element with the form

$$a_s(x,y) = \sum_{l=1}^{k} A_s^l \varphi_i^l(x,y) \tag{38}$$

The principal difference between this form of collisional relaxation term and the one used in the FVM method [39] is that the collision frequency and other macroscale properties can vary inside the spatial elements. Note that this relaxes the requirement on the cell size to be smaller than the mean free path of molecules. Now the $4k$ or $7k$ unknown coefficients A_s^l can be found from the weak formulations of the conservations in collision relaxation term, formed

$$\int_{K_i} v(x,y) C \varphi_i^m(x,y) dx dy = 0, \quad m = 1, \ldots, k \tag{39}$$

where C is the left-hand side of equations. This yields a system of $4k$ or $7k$ non-linear equations in each triangle element, which are solved iteratively using the Newton's method. A tolerance limit of 10^{-10} and a maximal iteration of 100 have been used. The Jacobian matrices of the systems are inversed by the Gaussian with scaled column pivoting algorithm. In this way, the discrete collision term does not give rise to any source or sink of mass, momentum or energy.

3. Spatial discretization and parallel implementation

In the present approach, the discrete velocity method is employed to discretize the velocity space and to cast the governing equation into a system of equations for the discrete velocity distributions. Then, the physical space is split up into a set of triangles and the finite element method is developed to solve the distributions at each velocity points. Within the triangle elements, the velocity distribution functions are sought in the finite element space of piecewise functions. Therefore, the complexity of the current computational problem is proportional to the number of velocity nodes N, the number of spatial elements M, and to the order of basis functions k. Due to the multi-dimensionality of the problem, parallel computations are highly desirable.

The parallel domain decomposition may be applied for the discrete velocity space Ω_V, for the physical space Ω_S, or both. In order to obtain high parallel efficiency and speed-up, the communication time and data passing among processors should be reduced. It can be indicated from the computational algorithm that, after the initial procedures, the time evolution of the velocity distribution functions involves two major parts: (1) computation of flux and (2) evaluation of the collision term. The flux computation is only over the physical space and the computational stencil involves only the nearest-neighbor communication, independent of the order. In the collision term calculation, the summations over the whole velocity space are required. Therefore, the present RKDG method is well suited for decomposition in physical space Ω_S for efficient parallelization. At the initialization step, the domain Ω_S is partitioned into several sub-domains with nearly equal number of triangles with each processor allocated one sub-domain. Triangular element pairs at the sub-domain boundaries will share edges, and the processors on which they reside communicate with each other at every intermediate time step to pass information on fluxes across their shared element edges.

4. Results and discussion

For verification we consider one unsteady and three steady-state flow problems. The steady-state is obtained by iterating in time until the convergence is reached. The time convergence

criterion is that the global relative residual of the velocity distribution functions between successive iterations decreases by a factor of 10^5 or 10^6. The residual is defined as

$$\sqrt{\frac{\sum_j \int_{\Omega_S} (f_{new}^j - f_{old}^j)^2 dx dy}{\sum_j \int_{\Omega_S} (f_{old}^j)^2 dx dy}} \tag{40}$$

where f_{new}^j are the distributions obtained at the current time step and f_{old}^j are the ones obtained at the previous time step. The RKDG solvers of second and third order are applied to solve the Boltzmann-BGK and ES-BGK equations for planar Couette flow, 2D conduction problem and long microchannel flow. The results are compared with the analytical results, DSMC solutions and solutions from a second-order finite volume solver [45], which is widely used currently. For convenience, the notations of the numerical method are illustrated in Table 1. These entire tests were done in double precision on a queue of the CARTER parallel cluster in Purdue University. Specially, the queue has 4 nodes with two 8-Core Intel Xeon-E5 processors and 32 GB RAM per nodes.

4.1. Steady Couette flow

The first test case considered is the planar Couette flow. Specifically, this case is used to verify the implementation of the periodic boundary condition and the wall boundary condition with a variable accommodation coefficient α. Both the RKDG and FVM methods were used to solve the 2D/3V BGK and ES-BGK models. The parameters used are the same as in the work by Mieussens [20] with argon gas between two plates $H = 1$ m apart maintained at a temperature of $T_w = T_0 = 273$ K. The bottom wall is at rest while the top one is moving with a velocity $u_w = 300$ m/s. Initially the gas has a density $\rho_0 = 9.28 \times 10^{-6}$ kg/m³, corresponding to the Knudsen number from the variable hard sphere models as 0.00925. For the cases $Kn = 0.0925$ and $Kn = 0.925$, all the conditions are same except that the gas density are now $\rho_0 = 9.28 \times 10^{-7}$ kg/m³ and $\rho_0 = 9.28 \times 10^{-8}$ kg/m³, respectively.

The problem was simulated on two-dimensional spatial mesh shown in Fig. 1. The triangular mesh used in RKDG simulations and the rectangle type mesh in FVM. The top and bottom boundaries are solid walls with given accommodation coefficients. The west and east boundaries were, respectively, the periodic condition in RKDG simulations and the zero-gradient condition in FVM cases. For such a one-dimensional flow, the two boundary conditions are equivalent. In both methods, the residual for the convergence criterion used was evaluated as

$$\sqrt{\frac{\sum_i \sum_j (f_{new}^{i,j} - f_{old}^{i,j})^2}{\sum_i \sum_j f_{old}^{i,j2}}} \tag{41}$$

The values of the distribution functions in spatial element i were calculated as the cell average value of f in RKDG method. All the

Table 1
Notations of the numerical methods used in this work.

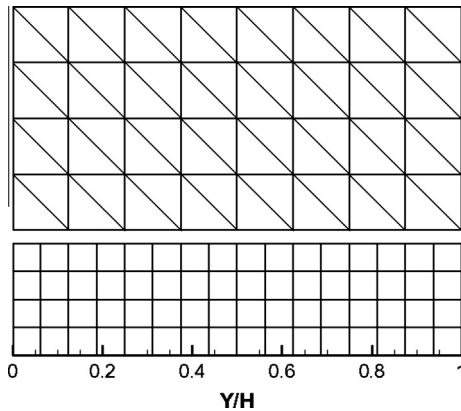| Notation	Numerical method
FVM-2	FVM with 2nd order minmod flux combined with 2nd order TVD Runge–Kutta time scheme
RKDG-2	2nd order DG method combined with 2nd order TVD Runge–Kutta time scheme
RKDG-3	3rd order DG method combined with 3rd order TVD Runge–Kutta time scheme

Fig. 1. Schematics of the two-dimensional spatial meshes. Top: triangle mesh used in RKDG simulation; bottom: rectangle mesh used in FVM simulation.

cases were run on a single processor. The DSMC results are used as a reference, which were obtained by Bird's 1D code [1]. The VHS model was applied in DSMC calculation. Therefore the exponent of the viscosity law in kinetic models used in DG and FVM solvers was set as 0.81 for Argon. By choosing the same viscosity, the

comparisons were expected to make sense. However, due to the intrinsic difference of the collision implementations between the stochastic and deterministic methods, the error measurements may inevitably contain modeling error.

First of all, to show the convergences of the different methods, the $Kn = 0.00925$ case with accommodation coefficient $\alpha = 1.0$ was run on spatial meshes with different number of elements. The Cartesian velocity grid of $10 \times 10 \times 10$ discrete velocities with bounds $[-4 \times (T_w/2)^{1/2}, u_w + 4 \times (T_w/2)^{1/2}]$ in c_x direction and $[-4 \times (T_w/2)^{1/2}, 4 \times (T_w/2)^{1/2}]$ in the other two directions was used. With this velocity grid, a further increase of the number of grid points improves the results by a magnitude no more than 0.05%. Solutions of different methods for the BGK model are plotted in Fig. 2a–c. The calculated bulk velocities are extracted along a vertical plane at the center of the computational domain and are compared with the DSMC result. N_y in the figures denotes the number of spatial elements along y direction. It is shown that as the numbers of elements increase, all the solutions converge to the DSMC result within a few percent. The FVM-2 results for bulk velocity converge to DSMC solution for $N_y = 128$ and larger, while the RKDG-2 results are converging for $N_y = 8$ and larger. The convergent results of the calculated temperature for different models are plotted in Fig. 3. The ES-BGK model obtains a correct result due to the fact that it gives a correct Pr number.
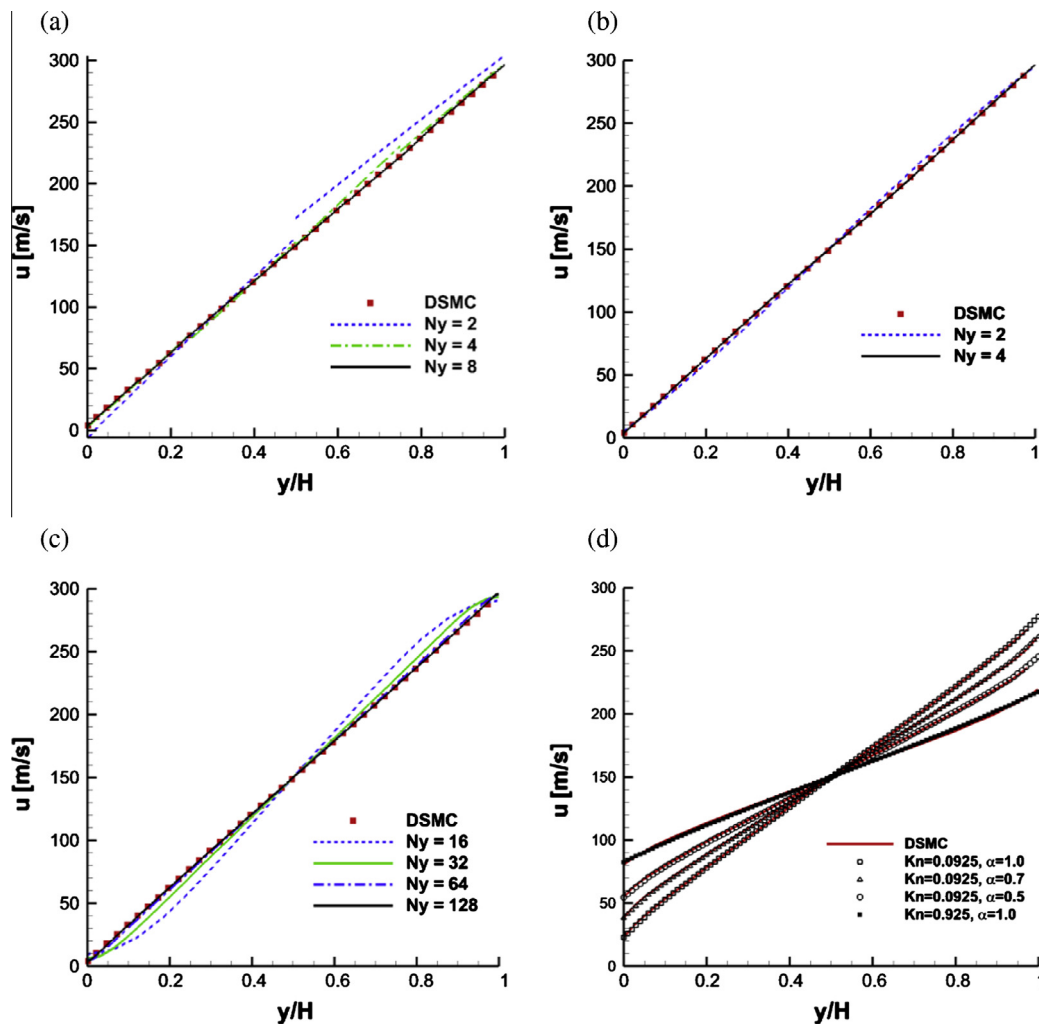

Fig. 2. Calculated bulk velocities for BGK solutions of Couette flow with different numerical method: (a) BGK RKDG-2 solutions with $Kn = 0.00925$, $\alpha = 1.0$; (b) BGK RKDG-3 solutions with $Kn = 0.00925$, $\alpha = 1.0$; (c) BGK FVM-2 solutions with $Kn = 0.00925$, $\alpha = 1.0$; (d) BGK RKDG-2 solutions with variable Kn and α. N_y denotes the number of spatial elements along y direction.
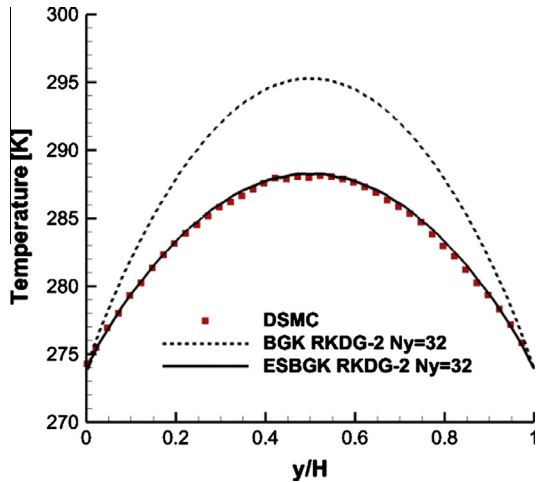
Fig. 3. Calculated temperature for two models with $Kn = 0.00925$, $\alpha = 1.0$.

For comparison of the computational efficiency of the methods, the time step, the number of iterations and total CPU times for the different methods are listed in Table 2. The time intervals for direct solvers are dependent on the cell sizes. The CFL numbers of about 0.3, 0.25 and 0.6 were applied to RKDG-2, RKDG-3 and FVM-2 respectively. It is shown that the RKDG method in general is much more CPU intensive than the second-order finite volume method with equivalent spatial elements. The most computationally intensive part of the RKDG solution of the kinetic models is the calculation of the equilibrium distribution functions in collision relaxation term due to the fact that the non-linear system of equations has to be solved iteratively. For the same reason, the solutions of the ES-BGK model take approximately thrice the CPU time of BGK solutions under the same conditions. Non-conservative schemes that retain the order of accuracy of space and time discretization can be constructed, and they are much faster but should be used with caution [34].

The comparison of the relative L_1 and L_2 errors of the RKDG and FVM solutions evaluated based on the DSMC result demonstrates that the RKDG-2 solution with $N_y = 8$ is at least as good as FVM-2 solution with $N_y = 128$. The CPU time required to obtain RKDG-2 solution with $N_y = 8$ is about 15 times smaller than that of the FVM-2 solution with $N_y = 128$. The required memory is about 5

times smaller. In general, based on this comparison one can conclude that the second-order Runge–Kutta discontinuous Galerkin solution for BGK model requires significantly less memory and CPU time than a second-order finite-volume method with the same accuracy. In addition, the RKDG-3 solution with $N_y = 4$ has the same accuracy as the FVM-2 solution with $N_y = 128$, which indicates that the RKDG-3 method is even more efficient in the discretization of physical space. However, the CPU time required in RKDG-3 scheme is about 3 times larger than that of RKDG-2 due to the fact that smaller time steps are required for the stability of the higher order RKDG method. Another probable reason is that, the flux scheme used in the present algorithm has only first-order accuracy with oscillatory convergence and thus requires a large number of iterations. High-order flux schemes to be investigated in the future offer an opportunity for further acceleration of the RDKG method for Boltzmann kinetic equations.

Finally, the Couette flow simulations are applied to verify the wall boundary condition with a given accommodation coefficient and to investigate the ability of the RKDG solver to reproduce rarefied flows in different flow regions. The RKDG-2 was used to solve the BGK model for flows with variable Knudsen number Kn and accommodation coefficients α. More velocity nodes are required to get convergent results for high-Knudsen number flows. Cartesian velocity grid of $14 \times 14 \times 14$ and $16 \times 16 \times 16$ were applied to the flows of $Kn = 0.0925$ and $Kn = 0.925$ respectively. The calculated bulk velocities are compared with the DSMC results in Fig. 2d. Very good agreements are shown and the relative L_2 norm errors of the cases are within 1.5%.

4.2. Oscillatory Couette flow

In the second case, in order to test the RKDG method for capturing the unsteady solutions, an oscillatory Couette flow was calculated by solving the 2D/2V BGK kinetic model equations. Unlike the steady Couette flow, the top wall moves with a harmonic velocity $u = u_0 \sin(\omega t)$, where u_0 is the velocity amplitude and ω is the oscillation frequency. In addition to the Knudsen number, this kind of flow is characterized by an additional dimensionless parameters, characterizing oscillation frequency: the ratio of oscillation frequency and collision frequencies $\theta = \omega/\nu$ [46]. In this case, the argon gas lays between two plates with initial parameters: $\rho_0 = 1.256 \times 10^{-3}$ kg/m^3, $T_0 = 300$ K. The moving wall has a temperature of $T_w = 300$ K, velocity amplitude of $u_0 = 10$ m/s and

Table 2
Computational parameters of different methods and models for Couette flow.

| Solution	Mesh	Δt (s)	# of iterations	CPU time (h)	RL_1 error (%)	RL_2 error (%)
BGK RKDG-2	4×2	5×10^{-5}	1257	0.02	12.14	7.20
	4×4	3×10^{-5}	14,502	0.44	2.51	1.92
	4×8	2×10^{-5}	17,722	1.12	0.72	0.41
	4×16	1×10^{-5}	35,247	4.16	0.50	0.23
	4×32	8×10^{-6}	50,459	9.46	0.49	0.22
ES-BGK RKDG-2	4×2	5×10^{-5}	1,263	0.07	12.34	7.20
	4×4	4×10^{-5}	14,730	1.36	2.47	1.90
	4×8	2×10^{-5}	17,908	3.43	0.70	0.42
	4×16	1×10^{-5}	35,639	12.99	0.42	0.19
	4×32	8×10^{-6}	44,646	32.96	0.33	0.15
BGK RKDG-3	4×2	3×10^{-5}	15,877	2.62	3.06	1.87
	4×4	2×10^{-5}	17,755	4.55	0.65	0.37
	4×8	1×10^{-5}	35,291	20.87	0.50	0.25
	4×16	8×10^{-6}	44,167	45.36	0.49	0.22
BGK FVM-2	4×16	2×10^{-5}	9,925	0.27	13.74	7.93
	4×32	1×10^{-5}	28,233	1.49	7.66	3.66
	4×64	8×10^{-6}	41,096	4.36	2.96	1.22
	4×128	4×10^{-6}	86,500	18.12	1.04	0.38
	4×256	2×10^{-6}	175,721	71.62	0.82	0.45
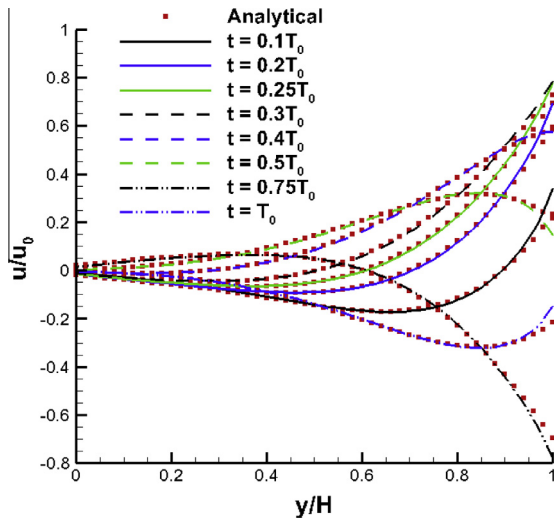
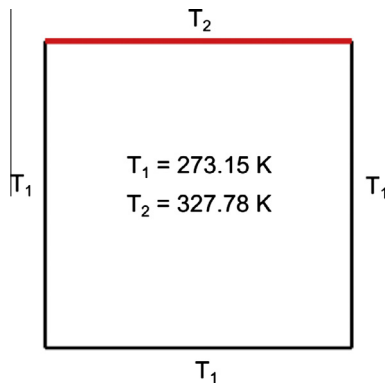**Fig. 4.** The velocity profiles of the oscillatory Couette flow during a period.

Fig. 5. Sketch of the two dimensional conduction problems.

a frequency of $\omega = 3.535 \times 10^5$. The HS model was employed to calculate the viscosity. This gives a Knudsen number of 0.09 and a frequency ratio of 0.1.

The problem was also simulated on two-dimensional spatial mesh with uniform triangles. The boundary conditions were the same as the ones in the steady case. Through a convergence study, 8×46 discrete velocities with an 8th-order Guass–Laguerre quadrature in magnitude and 3/8th Simpson integral in angle were adopted. Initially, the gas was at rest and reached the periodic steady state after 4 periods. The velocity profiles during a period are plotted in Fig. 4 by compared with analytical solutions. The numerical results were obtained from RKDG-2 scheme on spatial mesh with $N_y = 64$ triangles and a time interval of $\Delta t = 0.0017$ (in dimensionless form). The analytical solutions are from Navier–Stokes equation with the second-order slip velocity boundary condition [47]. The RKDG results agree very well with the analytical solutions confirming time-accurate discretization. Only a small discrepancy appears near the moving wall. The error is mainly due to the fact that the second-order slip boundary could not capture the highly non-equilibrium kinetic layer at this Knudsen number. The same difference is also observed between the DSMC and analytical solutions in Ref. [47].

RKDG-3 scheme was also employed to obtain solution of this flow. It has been shown that both implementations are time-accurate and converge to periodic solution at the same time interval. The main difference is that RKDG-3 scheme uses less spatial elements due to its higher order discretization in space.

4.3. Two-dimensional conduction problem

The two-dimensional verification is carried out by comparison to the analytical solution of a steady heat conduction problem in a square cavity. The three boundaries are maintained at $T_1 = 273.15$ K, while the forth side is at $T_2 = 327.78$ K, as shown in Fig. 5. The Argon gas is initialed with temperature $T_0 = 273.15$ K and density $\rho_0 = 4.77 \times 10^{-4}$ kg/m^3, corresponding to a Knudsen number from VHS model of 0.0018. This case is specially used to verify the symmetrical boundary condition. The obtained solutions of the 2D/2V BGK model are compared with the analytical distribution, which is the solution of the general heat equation under the assumption of two-dimensional steady state [48] given as

$$\theta(x, y) = \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{(-1)^{n+1} + 1}{n} \sin(n\pi x/L) \frac{\sinh(n\pi y/L)}{\sinh(n\pi W/L)} \quad (42)$$

where θ is the normalized temperature defined as $\theta = (T - T_1)/(T_2 - T_1)$. L and W are respectively the length and width of the plate, which are both set as 0.1 m in this test. For computationality, a truncation of $n = 100$ is used. The solutions were sought on the half domain of 0.05×0.1 m with six consecutively refined spatial meshes. The symmetric boundary was used on the vertical plane crossing $x = 0.05$ m. The spherical velocity discretization of an 8th order Gauss–Laguerre quadrature in velocity magnitude and constant integral in angels was used with totally 8×12 nodes. The cases were run on parallel processors and the converged results were reached when the residual of distribution functions was less than the factor of 5×10^{-5}. The calculated temperatures on different special meshes are plotted in Figs. 6 and 7, while the time intervals Δt, number of iterations, number of processors, cumulative CPU time and relative L_2 errors are listed in Table 3. The relative L_2 errors are obtained through the analytical results.

The first three meshes are structured triangle meshes with element numbers from 4 to 64. As the number of elements increases, both the RKDG-2 and RKDG-3 solutions approach to analytical consistent with the order of accuracy of the methods. As expected, the RKDG-2 solutions are piecewise linear, while the RKDG-3 solutions are piecewise quadratic. Therefore, the results of RKDG-3 are much better than that of RKDG-2 with same spatial mesh. Further observations of the solutions on 16 cells, i.e. Fig. 6c and d, show that the simulated temperature contours on the bottom half of the plate are very close to the analytical results and the large error mainly comes from the region with high gradients. This indicates that using more cells in the large-gradient region may provide faster convergence. Therefore, 6 more cells were artificially put on the top of the plate as shown in Fig. 7a and b. Compared the results of 16 elements and 22 elements, about 1.5 times more accurate solutions were reached with only 6 additional cells. This points to an automatic mesh-adaptation algorithm which can refine the spatial mesh according to the local gradients, based on the natural discontinuities in cell edge values.

Another way to improve the solutions with relatively small cost is adopting non-uniform unstructured mesh with more triangles in large gradient areas. As illustrated in the last four figures from Fig. 7c–f, the coarse mesh has 68 triangles and the refined one has 312 elements in total. We can see that the RKDG-2 solution on 68 unstructured triangles is about 1.4 times more accurate than the one on 64 uniform triangles. Besides, the unstructured mesh can easy handle complex geometries. The solutions converged with an error of about 0.14%, which mainly comes from the fact that the continuum heat conduction equation is only a limit equation of the ES-BGK model when Kn tends to 0. The converged RKDG-2 solution was obtained on mesh with 312 triangles and cost 9.04 h cumulative CPU time (actual CPU time multiplied by the number of processors), while the same accuracy solution of RKDG-3 method
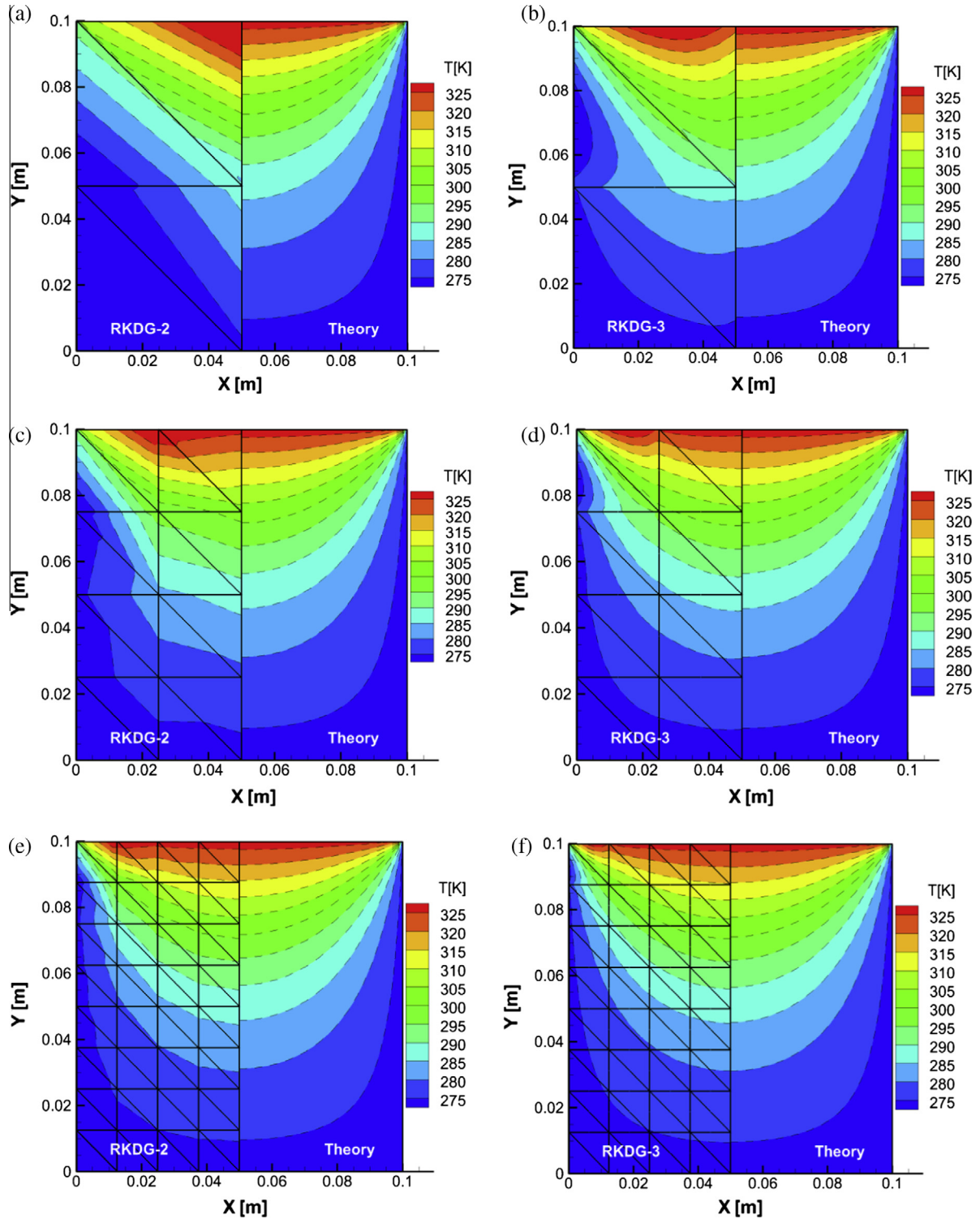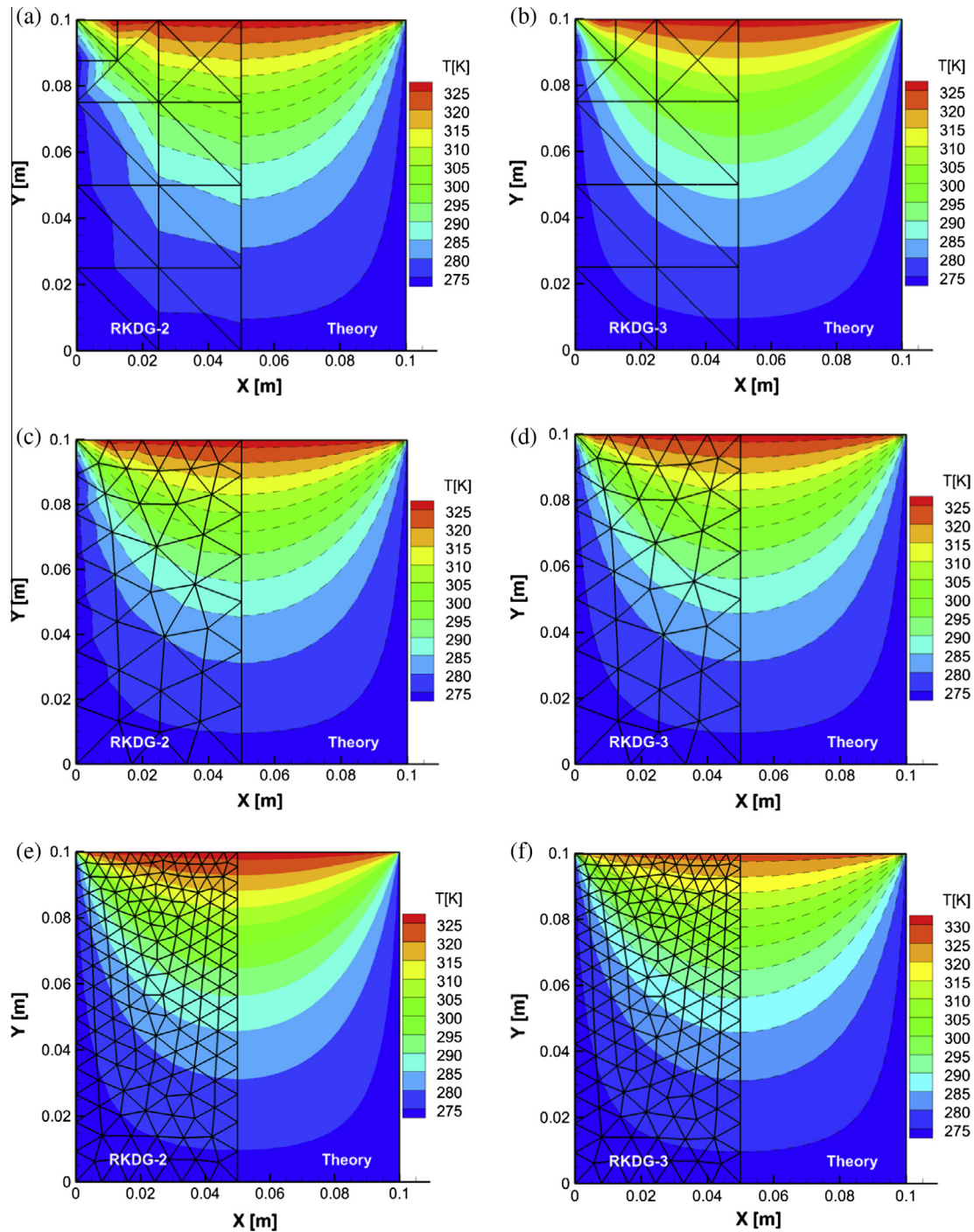
Fig. 6. Temperature distributions of the 2D conduction problem on structured spatial mesh: (a) RKDG-2 solution on 4 elements; (b) RKDG-3 solution on 4 elements; (c) RKDG-2 solution on 16 elements; (d) RKDG-3 solution on 16 elements; (e) RKDG-2 solution on 64 elements; (f) RKDG-3 solution on 64 elements.

was obtained on mesh with 22 triangles which cost 6.32 h cumulative CPU time. The CPU memories required for the two solutions are about 7.9 MB and 0.8 MB respectively. The RKDG-3 method is faster here due to the fact that the non-uniform unstructured mesh was used in the physical space. As we know, in regions with large gradients, a highly refined mesh is usually required in order to obtain accurate solutions, while in smooth areas, a coarse mesh should suffice. Compared to DG2, DG3 implements a more efficient

discretization but results in a larger number of degrees of freedom, which has an uncertain (positive or negative) contribution to computation time depending upon the mesh types. In the Couette flow case with uniform structured meshes, the DG3, although having adopted smaller grid cells, was more computationally expensive than the DG2. While in the conduction case utilizing unstructured meshes, much less cells were needed in DG3 and its advantages are realized.

Fig. 7. Temperature distributions of the 2D conduction problem on unstructured spatial mesh: (a) RKDG-2 solution on 22 elements; (b) RKDG-3 solution on 22 elements; (c) RKDG-2 solution on 68 elements; (d) RKDG-3 solution on 68 elements; (e) RKDG-2 solution on 312 elements; (f) RKDG-3 solution on 312 elements.

The RKDG method has a different cell size requirement from that of DSMC method. This is mainly due to their intrinsic differences in the approach to solving the Boltzmann equation. In DSMC method, the two-step process of particle moving and colliding consists of simulating advection and collision terms in Boltzmann equation, which is directly discretized by deterministic solvers. The different ways of molecular collision model implementation have direct influence on the resultant transport, e.g. viscosity and thermal conductivity. In the DSMC method, the contribution of particle collision to these transport processes is affected by the cell size and significant errors occur when cell dimensions are larger

than mean free path [49]. On the other hand, the deterministic solvers based on the ES-BGK model kinetic equations resolve the transport coefficients by setting the collision frequency as Eq. (2). In this way, the expression of the transport coefficients would be the same as given in the full collision integral, when Chapman–Enskog method is employed to solve the model equations [50]. Hence, the viscosity and thermal conductivity correlate with the pressure and temperature obtained by discrete distribution functions which are represented by polynomials. Therefore, similar criterion as in a RKDG solver for macroscopic equations such as Euler or Navier–Stokes is used to determine grid refinement.

Table 3
Computational parameters of different solutions on variable meshes for 2D conduction problem.

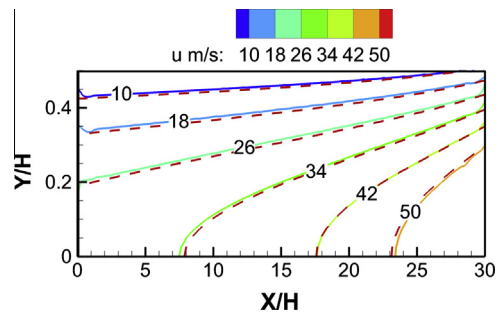| Solution	Element number	Δt (s)	# of iterations	# of processors	Cumulative CPU time (h)	RL_2 error (%)
RKDG-2	4	1×10^{-6}	21,559	1	0.03	1.64
	16	1×10^{-6}	25,746	2	0.18	0.79
	22	7×10^{-7}	37,394	2	0.36	0.49
	64	7×10^{-7}	39,100	4	0.84	0.38
	68	7×10^{-7}	38,852	4	1.00	0.27
	312	4×10^{-7}	68,691	8	9.04	0.14
RKDG-3	4	7×10^{-7}	38,988	1	0.59	0.41
	16	7×10^{-7}	39,626	2	2.54	0.22
	22	5×10^{-7}	55,064	2	6.32	0.14
	64	5×10^{-7}	55,107	4	15.36	0.14
	68	5×10^{-7}	55,076	4	21.36	0.13
	312	2×10^{-7}	13,7689	8	194.80	0.13
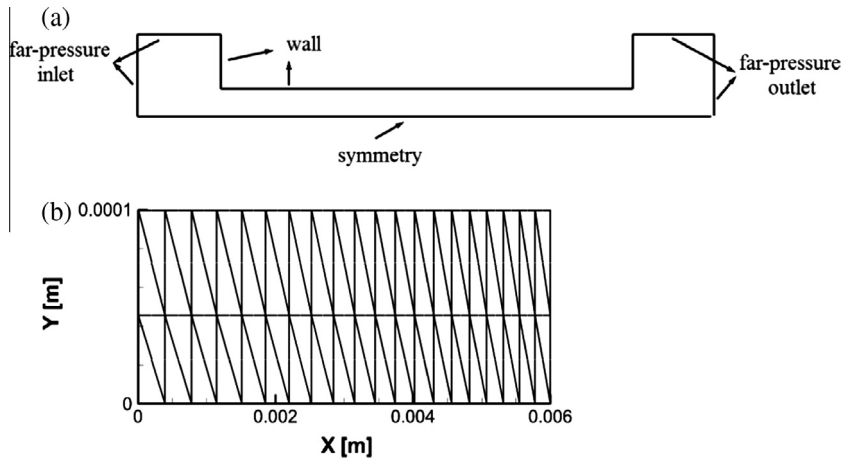
Fig. 8. Schematics of the computation domain, boundary conditions and spatial meshes for microchannel flow: (a) computation domain and boundary conditions; (b) spatial meshes in channel.
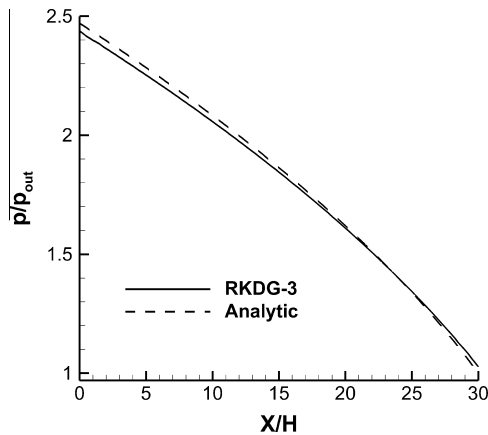
Fig. 9. Pressure profile for long microchannel flow, Kn = 0.05, L/H = 30. Dash line is the analytical solution, solid line is the RKDG-3 result.

Fig. 10. x-component velocity contours for long microchannel flow, Kn = 0.05, L/H = 30. Read dash line is the analytical solution, color solid line is the RKDG-3 result.

4.4. Long microchannel flow

The last test is a benchmark problem of long microchannel flow at a low Knudsen number. To validate the implementation of the far-pressure inlet/outlet boundary conditions, the nitrogen flow in a straight channel with aspect ratio L/H of 30 was calculated based on the ES-BGK model and compared with the analytic approximate solution from 2D Navier–Stokes equations with first-order slip-velocity boundary conditions [51]. The height of the channel was set as 0.2 mm and Knudsen number based on

the outlet condition and with respect to the channel height is 0.05. The pressure ratio between the inlet and outlet is 2.47. The inlet temperature of the gas is 300 K which corresponds to a value of the viscosity coefficient of 1.776×10^{-5} kg/m s based on the VHS model with a viscosity-temperature exponent of 0.74. Initial density and temperature of the gas are 6.15×10^{-3} kg/m^3 and 300 K respectively.

The governing equation, i.e. Eq. (1), for the present system is accurate for monatomic gas. If the gas is not simply a monatomic gas but has internal structure, the equation must be modified and the problem will become rather difficult. One simplistic approximation is to assume that all internal molecular energy modes are in equilibrium, which is suitable when the condition
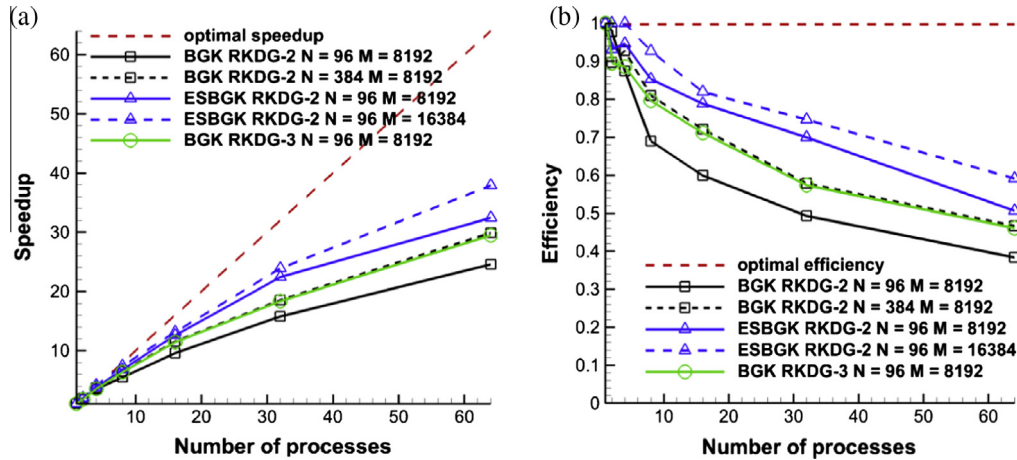
Fig. 11. Speed up and efficiency for the RKDG solver: (a) speed-up ratio; (b) efficiency.

deviate a bit from the equilibrium. Therefore, we just modified the Pr as 0.72 for nitrogen without any other changes. Fig. 8 shows the schematics of the computation domain, boundary conditions and spatial meshes in channel. It was found in previous works that the inlet and outlet reservoirs should be included to well model the gas flow in finite length channel. The chamber-to-channel length and width-to-channel height ratio were suggested larger than one-tenth and three, respectively [52]. The far-pressure inlet/out let boundary conditions were applied for the inlet and outlet chambers and a constant 300 K wall temperature conditions were used. The RKDG-3 method was employed to resolve the problem on 4 processors. The steady state solution was assumed to have been reached when the residual of distribution functions was less than 1×10^{-6}. The RKDG numerical solution of this application generally depends on three parameters: the chamber size, the number of triangles and the number of discrete velocities. In order to ensure that the results of the calculations presented here are accurate to within a few percent, a convergence study was performed by running the tests on different combinations of these parameters. Finally, chamber size of 2.0×1.2 mm, 20×4 triangles for channel and 7×16 triangles for chambers, and 8×25 discrete velocities with an 8th order Gauss–Laguerre quadrature in magnitude and 3/8th Simpson integral in angle were employed. Further enlargement of the chamber and increase of the numbers of triangles and velocity points improved the results by a magnitude no more than 1% to 2%. A time-interval of 6.1×10^{-9} s was used, which took a total of 30,886 iterations within a cumulative CPU time of about 164 h. Figs. 9 and 10 show the comparison of the numerical and analytical solutions with slip boundary conditions for pressure and x-component velocities. Excellent agreement is observed and the relative differences between the two solutions are within 1% and 2.5% respectively for pressure and velocity distributions. The errors are mainly due to the finite length of chambers and the analytical model error, since the precision of the analytical solution has the order of $O(H/L)$. It is believed the larger the chamber size and the height–length ratio, the better the agreements. It is worth mentioning the same test was run using DSMC method with a total cost of 200,000 cells, 12.8 million simulated particles and over 1,000,000 time steps to achieve similar accuracy that the difference in pressure of the numerical and analytical solutions does not exceed 1% [53]. The DSMC calculations required well over 1000 h. Therefore, the deterministic approach is another very efficient way for low Knudsen number and low-speed microscopic flow simulations.

4.5. Parallel performance studies

In this section, we present results of the parallel performance studies for the present RKDG codes on a queue of the CARTER parallel cluster in Purdue University. Specially, the queue has 4 nodes with two 8-Core Intel Xeon-E5 processors and 32 GB RAM per nodes. The Message-Passing Interface (MPI) standard implemented here is MVAPICH2.

Several processors working on a fixed-size problem should be able to solve the problem faster than a single processor. The Parallel speedup is defined as $S = T_{serial}/T_{parallel}$, where T_{serial} is the CPU time for a computational on 1 processor and $T_{parallel}$ is the CPU time for a computation using more than 1 processors. Ideally, the speedup is equal to N_p when N_p processors are used. However, using more processors needs additional communication operations. Therefore, using more processors simultaneously causes decreasing calculation time and increasing communication time. The speedup curve is approaching to a constant value first as the number of processors increases. When further increasing N_p, other overhead time will finally drive the curve downward. As mentioned in Section 3, the complexity of the problems is proportional to the numbers of $N \times M \times k$ and also dependent on the type of model.

We choose the two-dimensional conduction problem to investigate the parallel performance, as this problem has a simple geometry. Different model equations, numbers of velocity nodes, spatial elements and basis functions were chosen to study their affect on the parallel performance. All the cases were run with the same time steps. Fig. 11 contains speedup and efficiency plots and the optimal values are shown as the read dashed lines. The speedup is good, and the efficiencies are within 60% up to 16 processors in all cases. From the RKDG-2 cases for BGK model, we can see that using more velocity nodes results in a better speedup. Although using more velocity nodes can both increase the computation time in serial simulation and communication time in parallel simulation, the effect of the additional computation complexity is dominated and gives better speedup. Due to the same reason, with same velocity mesh and spatial mesh, RKDG-3 solver can also obtain a better speedup than RKDG-2 solver. Comparing the simulations using BGK and ES-BGK models, it is found that speedup for the ES-BGK model is much better than BGK models. This is due to the fact that ES-BGK model only brings in more computational operations. Finally, the more spatial elements are used, the better speedup is obtained. The test cases were simulated using 2D/2V

code. However the results for the 2D3V simulations are expected the same.

5. Conclusions

The high-order Runge–Kutta discontinuous Galerkin method has been extended to 2D/2V and 2D/3V Boltzmann BGK and ES-BGK kinetic model equation solution on arbitrary unstructured spatial mesh. With this approach, the velocity space is first discretized using either Cartesian or spherical type discrete velocity methods. Then, the discrete partial differential equations are discretized on spatial triangle-type meshes using the discontinuous Galerkin (DG) method. The linear and quadratic functions are chosen as the basis functions respectively for the second-order and third-order DG method. The system of ordinary differential equations, which is obtained from the spatial discretizations, is finally discretized in time using a special class of explicit Runge–Kutta time discretization methods. At each intermediate step of the RK process, the equilibrium velocity distribution function in the model equations is estimated using a discontinuous conservative discretization method, which enforces a weak conservation of mass, momentum and energy for the collision relaxation term. The first-order upwind flux scheme is employed to evaluate the numerical fluxes and five different boundary conditions including the symmetry boundary, specular-diffuse moving wall with given accommodation coefficient, periodic boundaries, for pressure inlet/outlet boundaries, and supersonic inlet/outlet boundaries are incorporated.

Verification of the formulation and solvers has been performed by comparison with DSMC and analytical solutions for rarefied compressible Couette flow, an oscillatory Couette flow in the slip regime, near-continuum 2D thermal conduction problem and long microchannel flow. Based on the Couette flow solutions obtained on the 2D meshes with uniform elements, the second-order RKDG solution requires significantly less memory and CPU time than that of a widely used second-order FVM solver with the same accuracy. Both structured and unstructured spatial meshes have been applied in the solutions of the 2D conduction problem. Results show that, the third-order RKDG method is more efficient than the second-order discretization in physical space. It is also observed that when an adaptive mesh with finer cells in large gradient regions is used, the third-order RKDG solution takes less memory and CPU time than that of the second-order one. The unsteady test confirms the time-accurate discretization of the RKDG method.

A physical space decomposition strategy is employed for parallel computation. Because of the compact nature of the discontinuous Galerkin discretization, communications are required only in flux computations. Investigation of the parallel performance has been conducted.

Acknowledgments

References

[1] Bird GA. Molecular gas dynamics and the direct simulation. Oxford: Clarendon; 1994.
[2] Cai C, Boyd ID, Fan J, Candler GV. Direct simulation methods for low-speed microchannel flows. J Thermophys Heat Transf 2000;14(3):368–78.
[3] Alexeenko AA, Levin DA, Gimelshein SF, Collins RJ, Reed BD. Numerical modeling of axisymmetric and three-dimensional flows in microelectromechanical systems nozzles. AIAA J 2002;40(5):897–904.
[4] Laux M. Local time stepping with automatic adaptation for the DSMC method. In: 7th AIAA/ASME joint thermophysics and hear transfer conference. Albuquerque, New Mexico (USA); 1998.
[5] Pareschi L, Caflish RE. An implicit Monte Carlo method for rarefied gas dynamics: I. the space homogeneous gas. J Comput Phys 1999;154(1):90–116.
[6] Macrossan MN, Metchnik MV, Pinto PA. Hypersonic flow over a wedge with a particle flux method. In: Proceeding of the 24th international symposium on rarefied gas dynamics. AIP; 2005. p. 650.
[7] Burt JM, Boyd ID. A low diffusion particle method for simulating compressible inviscid flows. J Comput Phys 2008;227(9):4653–70.
[8] Fan J, Shen C. Statistical simulation of low-speed rarefied gas flows. J Comput Phys 2001;167(2):393–412.
[9] Baker LL, Hadjiconstantinou NG. Variance reduction for Monte Carlo solutions of the Boltzmann equation. Phys Fluids 2005;17:051703.
[10] Radtke GA, Hadjiconstantinou NG, Wagner W. Low-noise Monte Carlo simulation of the variable hard sphere gas. Phys Fluids 2011;23:030606.
[11] Wagner W. Deviational particle Monte Carlo for the Boltzmann equation. Monte Carlo Meth Appl 2008;14(3):191–268.
[12] Burt JM, Boyd ID. A hybrid particle approach for continuum and rarefied flow simulation. J Comput Phys 2009;228(2):460–75.
[13] Kolobov VI, Bayyuk SA, Arslanbekov RR, Aristov VV, Frolova AA, Zabelok SA. Construction of a unified continuum/kinetic solver for aerodynamic problems. J Spacecr Rocket 2005;42(4):598–606.
[14] Huang AB, Giddens DP. The discrete ordinate method for the linearized boundary value problem in kinetic theory of gases. In: Brundin CL, editor. 5th international symposium on rarefied gas dynamics. New York: Academic Press; 1967. p. 481–504.
[15] Aristov VV. Direct methods for solving the Boltzmann equation and study of non-equilibrium flows. Norwell: Kluwer Academic Publishers; 2001.
[16] Yang JY, Huang JC. Rarefied flow computations using nonlinear model Boltzmann equations. J Comput Phys 1995;120(2):323–39.
[17] Huang JC. A conservative discrete ordinate method for model Boltzmann equations. Comput Fluids 2011;45:261–7.
[18] Li ZH, Zhang HX. Gas kinetic algorithm using Boltzmann model equation. Comput Fluids 2004;33:967–91.
[19] Kolobov VI, Arslabekovm RR, Aristov VV, frolova AA, Zabelk SA. Unified solver for rarefied and continuum flows with adaptive mesh and algorithm refinement. J Comput Phys 2007;223(2):589–608.
[20] Mieussens L, Struchtrup H. Numerical comparison of Bhatnagar–Gross–Krook models with proper Prandtl number. Phys Fluids 2004;16(8):2797–813.
[21] Morinishi K. Numerical simulation for gas microflows using Boltzmann equation. Comput Fluids 2006;35:978–85.
[22] Titarev VA. Efficient deterministic modeling of three-dimensional rarefied gas flows. Commun Comput Phys 2012;12(1):161–92.
[23] Reed WH, Hill TR. Triangular mesh methods for neutron transport equation. Technical report LA-UR-73-479, 1973, Los Almos Scientific Laboratory, Los Alamos.
[24] Cockburn B, Shu CW. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: General framework. Math Comput 1989;52(186):411–35.
[25] Cockburn B, Shu CW. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws III: One dimensional systems. J Comput Phys 1989;84(1):90–113.
[26] Cockburn B, Shu CW. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws IV: The multidimensional case. Math Comput 1990;54(190):545–81.
[27] Cockburn B, Shu CW. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws V: Multidimensional systems. J Comput Phys 1998;141(2):199–224.
[28] Zhou T, Li Y, Shu CW. Numerical comparison of WENO finite volume and Runge–Kutta discontinuous Galerkin methods. J Sci Comput 2001;16(2):145–71.
[29] Toulorge T, Desmet W. CFL conditions for Runge–Kutta discontinuous Galerkin methods on triangular girds. J Comput Phys 2011;230(12):4657–78.
[30] Liu H, Xu K. A Ringe–Kutta discontinuous Galerkin method for viscous flow equations. J Comput Phys 2007;224(2):1223–42.
[31] Gobbert MK, Webster SG, Cale TS. A Galerkin method for the simulation of the transient 2D/2V and 3D/3V linear Boltzmann equation. J Sci Comput 2006;30(2):237–73.
[32] Baker LL, Hadjiconstantinou NG. Variance-reduced Monte Carlo solutions of the Boltzmann equation for low-speed gas flows: a discontinuous Galerkin formulation. Int J Numer Meth Fluids 2008;58(4):381–402.
[33] Alekseenko AM. Numerical properties of high order discrete velocity solutions of the BGK kinetic equation. Appl Numer Math 2011;61(4):410–27.
[34] Alexeenko AA, Galitzine C, Alekseenko AM. High-order discontinuous Galerkin method for Boltzmann model equations. In: 40th thermophysics conference. Seattle, Washington (USA); 2008.
[35] Shu CW. Efficient implementation of essentially non-oscillatory shock-capturing schemes. J Comput Phys 1988;77(1):439–71.
[36] Bhatnagar BL, Gross EP, Krook M. A model for collision processes in gases I: small amplitude processes in charged and neutral one-component systems. Phys Rev 1954;94(3):511–25.
[37] Holway LH. New statistical models for kinetic theory: methods of construction. Phys Fluids 1966;9(9):1658–73.
[38] Andries P, Tallex PL, Perla JP, Perthame B. The Gaussian-BGK model of Boltzmann equation with small Prandtl number. Eur J Mech B – Fluids 2000;19(6):813–30.

[39] Chigullapalli S, Alexeenko AA. Unsteady 3D rarefied flow solver based on Boltzmann-ESBGK model kinetic equation. In: 41st AIAA fluid dynamics conference and exhibit. Honolulu, Hawaii (USA); 2011.

[40] Cockburn B, Shu CW. Runge–Kutta discontinuous Galerkin method for convection dominated problems. J Sci Comput 2001;16(3):173–261.

[41] Kubatko EJ, Dawson C, Westerink JJ. Time step restrictions for Runge–Kutta discontinuous Galerkin methods on triangular grids. J Comput Phys 2008;227(23):9697–710.

[42] Cowper GR. Gaussian quadrature formulas for triangles. Int J Numer Method Eng 1973;7(3):405–8.

[43] Qiu JX, Khoo BC, Shu CW. A numerical study for the performance of the Runge–Kutta discontinuous Galerkin method based on different numerical fluxes. J Comput Phys 2006;212(2):540–65.

[44] Mieussens L. Discrete-velocity models and numerical schemes for the Boltzmann-BGK equation in plane and axisymmetric geometries. J Comput Phys 2002;162(2):429–66.

[45] Chigullapalli S, Venkatrramman A, Ivanov MS, Alexeenko AA. Entropy considerations in numerical simulations of non-equilibrium rarefied flows. J Comput Phys 2010;229(6):2139–58.

[46] Yap YW, Sader JE. High accuracy numerical solutions of the Boltzmann Bhatnagar–Gross–Krook equation for steady and oscillatory Couette flows. Phys Fluids 2012;24:032004.

[47] Hadjiconstantinous NG. Oscillatory shear-driven gas flows in the transition and free-molecular-flow regimes. Phys Fluids 2005;17:100611.

[48] Incropera Frank P. Fundamentals of heat and mass transfer. Hoboken: John Wiley; 2007.

[49] Alexander FJ, Garcia AL, Alder BJ. Cell size dependence of transport coefficients in stochastic particle algorithms. Phys Fluids 1998;10(6):1540–2.

[50] Sharipov F, Seleznev V. Data on internal rarefied gas flows. J Phys Chem Ref Data 1998;27(3):657–706.

[51] Arkilic EB, Schmidt MA, Breuer KS. Gaseous slip flow in long microchannels. J Microelectro Mech Syst 1997;6(2):167–78.

[52] Guo XH, Huang C, Alexeenko AA, Sullivan J. Numerical and experimental study of gas flows in 2D and 3D microchannels. J Micromech Microeng 2008;18(2):025034.

[53] Alexeenko AA, Gimelshein SF, Levin DA. Reconsideration of low Reynolds number flow-through constriction microchannels using the DSMC method. J Microeletromech Syst 2005;14(4):847–56.