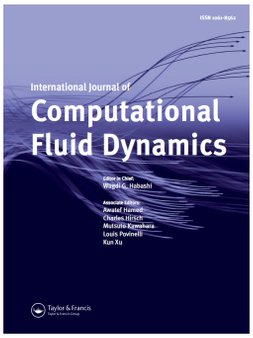

International Journal of Computational Fluid Dynamics

ISSN: 1061-8562 (Print) 1029-0257 (Online) Journal homepage: https://www.tandfonline.com/loi/gcfd20

High-order hybridisable discontinuous Galerkin method for the gas kinetic equation

Wei Su, Peng Wang & Yonghao Zhang

To cite this article: Wei Su, Peng Wang & Yonghao Zhang (2019): High-order hybridisable discontinuous Galerkin method for the gas kinetic equation, International Journal of Computational Fluid Dynamics, DOI: 10.1080/10618562.2019.1666110

To link to this article: https://doi.org/10.1080/10618562.2019.1666110

Published online: 16 Sep 2019.

Submit your article to this journal

Article views: 1

View related articles

View Crossmark data

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=gcfd20

INTERNATIONAL JOURNAL OF COMPUTATIONAL FLUID DYNAMICS
https://doi.org/10.1080/10618562.2019.1666110

Taylor & Francis
Taylor & Francis Group

Check for updates

High-order hybridisable discontinuous Galerkin method for the gas kinetic equation

Wei Su, Peng Wang and Yonghao Zhang

James Weir Fluids Laboratory, Department of Mechanical and Aerospace Engineering, University of Strathclyde, Glasgow, UK

ABSTRACT

The high-order hybridisable discontinuous Galerkin (HDG) method is used to find steady-state solution of gas kinetic equations on two-dimensional geometry. The velocity distribution function and its traces are approximated in piecewise polynomial space on triangular mesh and mesh skeleton, respectively. By employing a numerical flux derived from the upwind scheme and imposing its continuity on mesh skeleton, the global system for unknown traces is obtained with fewer coupled degrees of freedom, compared to the original DG method. The solutions of model equation for the Poiseuille flow through square channel show the higher order solver is faster than the lower order one. Moreover, the HDG scheme is more efficient than the original DG method when the degree of approximating polynomial is larger than 2. Finally, the developed scheme is extended to solve the Boltzmann equation with full collision operator, which can produce accurate results for shear-driven and thermally induced flows.

ARTICLE HISTORY
Received 13 December 2018
Accepted 18 July 2019

KEYWORDS
Hybridisable DG; Boltzmann equation; rarefied gas flow; upwind flux

1. Introduction

Gas flow widely encountered in high-altitude aerodynamics, vacuum technology, shale gas extraction and micro-electro-mechanical system (MEMS) is often beyond the continuum flow regime, where the evolution of macroscopic flow properties, such as density, bulk velocity and temperature, is no longer governed by the conventional Navier–Stokes (NS) equations. To predict such rarefied gas flows, microscopic description for the motion of gas molecules is necessary. In gas kinetic theory, the system state is described by the velocity distribution function (VDF) f. It is a function of time t, position $\boldsymbol{x}$ and molecular velocity $\boldsymbol{v}$, and is defined such that $f\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{v}$ is the number of gas molecules in the phase-space (physical space and velocity space) volume $\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{v}$. The macroscopic flow properties are derived via velocity moments of VDF. The evolution of f is governed by the Boltzmann equation (Chapman and Cowling 1970)

$$\frac{\partial f}{\partial t}+\boldsymbol{v}\cdot\frac{\partial f}{\partial \boldsymbol{x}}+\boldsymbol{a}\cdot\frac{\partial f}{\partial \boldsymbol{v}}=\mathcal{C},$$
$$\mathcal{C}=\iint B\cdot\left(f_*'f'-f_*f\right)\mathrm{d}\Omega\mathrm{d}\boldsymbol{v}_* \tag{1}$$

where $\boldsymbol{a}$ is the external force exerting on a volume of unit mass of gas molecules and $\mathcal{C}$ is the collision operator that describes change in VDF due to binary collisions. Nomenclature for other arguments is referred to Wu et al. (2013). The multidimensional nature (i.e. six dimensions in the phase space and one dimension in the temporal space) of the Boltzmann equation poses a real challenge to the numerical solution. Two categories of numerical approaches have been developed. One is the stochastic method that uses simulating particles to mimic the molecular behaviour and the other is the deterministic method that relies on the discretisation of governing equation over computational grid (Dimarco and Pareschi 2014). The deterministic approaches are based on the discrete velocity model, which uses a set of M_v discrete $\boldsymbol{v}^j$ to discretise the molecular velocity space, resulting in a system of equations evaluated at each $\boldsymbol{v}^j$

$$\frac{\partial f^j}{\partial t}+\boldsymbol{v}^j\cdot\frac{\partial f^j}{\partial \boldsymbol{x}}+\boldsymbol{a}\cdot\left(\frac{\partial f}{\partial \boldsymbol{v}}\right)_j=\mathcal{C}^j,\quad j=1,\ldots,M_\mathrm{v}, \tag{2}$$

where $f^j(t,\boldsymbol{x})=f(t,\boldsymbol{x},\boldsymbol{v}^j)$. Then, the equations that are still continuous in the physical space and time

CONTACT Wei Su wei.su@strath.ac.uk

© 2019 Informa UK Limited, trading as Taylor & Francis Group

are solved by computational fluid dynamic (CFD) technologies. Usually, a large number, e.g. several ten thousands, of discrete velocities (discrete equations) are required to accurately capture the variation of VDF. This makes the computational cost immediately prohibitive for realistic problems. Thus, the high-order CFD method is critical to reducing the number of degree of freedom (DoF).

One of the promising methods for this purpose is the discontinuous Galerkin (DG) method, which was first introduced for the neutron transport equation (Reed and Hill 1973). After combing an explicit Runge–Kutta (RK) time marching scheme, the method has great success in solving convection-dominated problems (Cockburn and Shu 2001). In recent years, the explicit DG method has been applied to solve the gas kinetic equations (Gobbert, Webster, and Cale 2007; Evans, Morgan, and Hassan 2011; Kitzler and Schöberl 2015). It was shown that the second-order DG method is 15 times more efficient than the second-order finite volume scheme in solving the kinetic model equations (Su, Alexeenko, and Cai 2015). Another attractive feature is that DG can naturally obtain fluxes at the boundaries with the same high-order accuracy as in the interior of the domain (Mieussens 2014). This is particularly important for the solution of flow in micro-systems, where the surface-to-volume ratio is significantly magnified.

Despite the above advantages, it was shown that higher order RKDG scheme is not superior to lower order one, e.g. the third-order scheme is about 4 times slower than the second-order one to obtain solutions with the same order of accuracy for rarefied Couette flow (Su, Alexeenko, and Cai 2015). It is mainly due to the two facts for the explicit high-order scheme: (1) number of DoF rapidly increases; (2) time step restricted by the Courant–Friedrichs–Lewy (CFL) condition becomes extremely small (Kubatko, Dawson, and Westerink 2008); thus the number of iteration becomes larger in finding steady-state solution. Although implicit scheme could be used to relax the CFL restriction, the classical DG methods are computationally expensive for implicit gas kinetic solvers, since the number of globally coupled DoF is significantly large. The hybridisable discontinuous Galerkin (HDG) method was proposed to overcome this disadvantage (Cockburn, Gopalakrishnan, and Lazarov 2009). By producing a final system in terms of DoF in approximating the traces of field variables, HDG could significantly reduce the number of global unknowns, since the traces are defined on cell interfaces and single-valued. The majority of HDG applications in fluid dynamics to date includes convection-diffusion flow (Cockburn, Gopalakrishnan, and Lazarov 2009), Stokes flow (Nguyen, Peraire, and Cockburn 2010), wave propagation problem (Giorgiani, Fernández-Méndez, and Huerta 2013) and incompressible/compressible NS flows (Peraire, Nguyen, and Cockburn 2010). In this paper, we present an HDG scheme that is designed for the gas kinetic equation for the first time.

2. HDG formulation

We present the HDG formulation to the linearised BGK model equation (Bhatnagar, Gross, and Krook 1954). When the flow velocity and the external acceleration are sufficiently small, we can linearise VDF about the global equilibrium state $f_{\text{eq}} = \exp(-|\boldsymbol{v}|^2)/\pi^{3/2}$ as $f = f_{\text{eq}}(1 + h)$, where h is the perturbed VDF. Then, interpreting f by h in Equation (1) and neglecting the nonlinear terms, we have the following equations for h^j (Cercignani 1988)

$$\begin{aligned} \boldsymbol{v}^j \cdot \frac{\partial h^j}{\partial \boldsymbol{x}} - 2\boldsymbol{a} \cdot \boldsymbol{v}^j &= \frac{\sqrt{\pi}}{2Kn}\left(\mathcal{L}^j - h^j\right), \\ \mathcal{L}^j &= \varrho + 2\boldsymbol{u} \cdot \boldsymbol{v}^j + \tau\left(|\boldsymbol{v}^j|^2 - \frac{3}{2}\right), \end{aligned} \tag{3}$$

where ϱ is the perturbed density, $\boldsymbol{u}$ is the bulk velocity and τ is the perturbed temperature. $\mathcal{L}^j$ is the linearised equilibrium distribution. Kn is the Knudsen number related to the characteristic flow dimension H and the gas properties (pressure p_0, temperature T_0 and viscosity μ_0) at reference condition as $Kn = \mu_0\sqrt{\pi RT_0}/\sqrt{2}p_0H$, where R is the specific gas constant. We have omitted the derivative with respect to time in Equation (3) since we are only interested in steady-state solutions. Macroscopic properties are calculated through quadratures

$$\begin{aligned} \varrho &= \sum_{j=1}^{M_{\text{v}}} h^j f_{\text{eq}}^j \varpi^j, \boldsymbol{u} = \sum_{j=1}^{M_{\text{v}}} \boldsymbol{v}^j h^j f_{\text{eq}}^j \varpi^j, \\ \tau &= \frac{2}{3}\sum_{j=1}^{M_{\text{v}}} |\boldsymbol{v}^j|^2 h^j f_{\text{eq}}^j \varpi^j - \varrho, \end{aligned} \tag{4}$$

where ϖ^j is the weight of the quadrature rule. All variables are given in a dimensionless form: $\boldsymbol{x}$ is

normalised by H; $\boldsymbol{v}$ and $\boldsymbol{u}$ are normalised by the most probable molecular speed $v_{\mathrm{m}} = \sqrt{2RT_0}$; T is normalised by T_0; VDF is normalised by n_0/v_{m}^3. Note that $\mathcal{L}^j$ contains macroscopic variables. To decouple the equations, Equations (3) are usually solved by the following semi-implicit iterative scheme

$$\frac{\sqrt{\pi}h^{j,(t+1)}}{2Kn} + \boldsymbol{v}^j \cdot \frac{\partial h^{j,(t+1)}}{\partial \boldsymbol{x}} = \frac{\sqrt{\pi}}{2Kn}\mathcal{L}^{j,(t)} + 2\boldsymbol{a}\cdot\boldsymbol{v}^j, \tag{5}$$

where the superscripts (t) and $(t+1)$ represent two consecutive iteration steps. The iteration is terminated when the convergence to a steady solution is achieved. For conciseness, we will omit the step index in the remainder of the paper unless necessary.

2.1. Weak formulation

Let $\Delta \in \mathbb{R}^2$ be a two-dimensional domain with boundary $\partial\Delta$ in the physical space, which is partitioned into M_{el} disjoint regular triangles $\{\Delta_i : \Delta = \cup_i^{M_{\mathrm{el}}}\Delta_i\}$. The boundaries of Δ_i (denoting as $\partial\Delta_i$) define a group of M_{fc} faces $\{\Gamma_c : \Gamma = \cup_i^{M_{\mathrm{el}}}\partial\Delta_i = \cup_c^{M_{\mathrm{fc}}}\Gamma_c\}$. The HDG method provides an approximate solution to h^j on Δ_i as well as an approximation to its trace $\hat{h}^j$ on Γ_c in some piecewise finite element spaces $\mathcal{V}\times\mathcal{W}$ of the following forms

$$\begin{aligned}\mathcal{V} &= \{\varphi :\ \varphi|_{\Delta_i} \in \mathcal{P}^k(\Delta_i), \forall\, \Delta_i \subset \Delta\},\\ \mathcal{W} &= \{\psi :\ \psi|_{\Gamma_c} \in \mathcal{P}^k(\Gamma_c), \forall\, \Gamma_c \subset \Gamma\},\end{aligned} \tag{6}$$

where $\mathcal{P}^k(D)$ denotes the space of kth-order polynomials on the domain D. Before describing the HDG formulation, we first introduce a collection of index mapping functions (Kirby, Sherwin, and Cockburn 2012) that relate the local edge of a triangle, namely $\partial\Delta_i^e$ to a global face Γ_c. Since the eth edge of the triangle $\partial\Delta_i$ is the cth face Γ_c, we set $\sigma(i,e) = c$ so that $\partial\Delta_i^e = \Gamma_{\sigma(i,e)}$. Similarly, since the interior face $\Gamma_c \in \Gamma\backslash\partial\Delta$ is the intersection of the two triangles, namely left triangle Δ_{i^-} and right triangle Δ_{i^+}, we set $\eta(c,+) = i^+$ and $\eta(c,-) = i^-$, then we denote $\Gamma_c = \partial\Delta_{\eta(c,+)} \cap \partial\Delta_{\eta(c,-)}$. At the boundary face $\Gamma_c \in \partial\Delta$, say, only the right triangle is involved.

The HDG method solves a problem in two steps (Cockburn, Gopalakrishnan, and Lazarov 2009). First, a global problem is set up to determine the trace $\hat{h}^j$ on Γ. Then, a local problem with $\hat{h}^j$ as the boundary condition on $\partial\Delta_i$ is solved element-by-element to obtain the solutions of h^j. Introducing $(\cdot)$ and $\langle\cdot\rangle$ as $(a,b)_D = \int_{D\subset\mathbb{R}^2}(a\cdot b)\mathrm{d}x_1\mathrm{d}x_2$ and $\langle a,b\rangle_D = \int_{D\subset\mathbb{R}^1}(a\cdot b)\mathrm{d}\Gamma$, respectively, the weak formulation of the local problem is

$$\begin{aligned}&-\left(\nabla\varphi, \boldsymbol{v}^j h^j\right)_{\Delta_i} + \sum_{e=1}^{3}\langle\varphi, \hat{\boldsymbol{F}}\cdot\boldsymbol{n}\rangle_{\partial\Delta_i^e}\\ &\quad + \frac{\sqrt{\pi}}{2Kn}(\varphi, h^j)_{\Delta_i} = (\varphi, s^j)_{\Delta_i}, \quad \text{for all } \varphi \in \mathcal{V},\end{aligned} \tag{7}$$

where $\boldsymbol{n}$ is the outward unit normal vector and $s^j = \sqrt{\pi}\mathcal{L}^j/2Kn - 2\boldsymbol{a}\cdot\boldsymbol{v}^j$. $\hat{\boldsymbol{F}}$ is the numerical flux, of which the definition will be given in the following section.

The global problem, used to determine $\hat{h}^j$, is obtained by imposing the continuity of normal fluxes at cell interfaces. The weak formulation is

$$\begin{aligned}&\langle\psi, \hat{\boldsymbol{F}}\cdot\boldsymbol{n}_{\eta(c,+)}\rangle_{\Gamma_c} + \langle\psi, \hat{\boldsymbol{F}}\cdot\boldsymbol{n}_{\eta(c,-)}\rangle_{\Gamma_c} = 0,\\ &\qquad \text{for all } \psi \in \mathcal{W},\end{aligned} \tag{8}$$

where $\hat{\boldsymbol{F}}\cdot\boldsymbol{n}_{\eta(c,+)}$ and $\hat{\boldsymbol{F}}\cdot\boldsymbol{n}_{\eta(c,-)}$ denote the numerical fluxes calculated from the left and right triangles, respectively. At the boundary faces, $\hat{\boldsymbol{F}}\cdot\boldsymbol{n}_{\eta(c,-)}$ should be replaced by $\hat{\boldsymbol{G}}\cdot\boldsymbol{n}$, i.e. fluxes flowing into the computational domain. Eliminating the field unknowns h^j in Equation (8) with Equation (7), and assembling over all the faces, the global problem becomes

$$\mathbb{K}^j\hat{\mathbf{H}}^j = \mathbb{R}^j, \tag{9}$$

where $\hat{\mathbf{H}}^j$ is the vector of DoF of $\hat{h}^j$ on all the faces. $\mathbb{K}^j$ is the global matrix of the linear system, which is highly sparse, since only face unknowns that involve in two adjacent triangles are coupled at each row. Compared to the standard linear DG system, the trace system is much smaller and sparser when $k>1$ (Huerta et al. 2013). Once the values of $\hat{h}^j$ are obtained, an element-by-element reconstruction of h^j is implemented according to Equation (7).

Note that the basic HDG formulation is not limited to the linearised BGK equation. It is straightforward to be extended to the nonlinear Boltzmann equation, where the unknowns in Equation (7) are the full VDF f and its trace $\hat{f}$ instead. Besides, the terms $\sqrt{\pi}(\varphi,h)_{\Delta_i}/2Kn$ and $(\varphi,s)_{\Delta_i}$ are replaced by $(\varphi, \iint Bf_*\mathrm{d}\Omega\mathrm{d}\boldsymbol{v}_*\cdot f)_{\Delta_i}$ and $(\varphi, \iint Bf_*'f'\mathrm{d}\Omega\mathrm{d}\boldsymbol{v}_*)_{\Delta_i} - (\varphi, \boldsymbol{a}\cdot\partial f/\partial\boldsymbol{v})_{\Delta_i}$ respectively. The fivefold integral $\iint\cdot\mathrm{d}\Omega\mathrm{d}\boldsymbol{v}_*$ and the derivative $\partial/\partial\boldsymbol{v}$ can be calculated by the fast spectral method (Wu et al. 2013). The detailed HDG formulations are listed in the Appendix.

2.2. Numerical flux and implementation of boundary condition

In this paper, we define the flux as (take the linearised BGK equation as an example)

$$\hat{\boldsymbol{F}}^j \cdot \boldsymbol{n} = \boldsymbol{v}^j \cdot \boldsymbol{n}\hat{h}^j + |\boldsymbol{v}^j \cdot \boldsymbol{n}| \left(h^j - \hat{h}^j\right). \tag{10}$$

If inserting the expression into continuity Equation (8) at interior faces, we immediately obtain $\langle \psi, \hat{h}^j \rangle = \frac{1}{2}\langle \psi, h^j_{\eta(c,+)} + h^j_{\eta(c,-)} \rangle$. That is, the trace $\hat{h}^j$ at the interior face is equal (in a weak sense) to the average of $h^j_{\eta(c,+)}$ and $h^j_{\eta(c,-)}$, which are the field unknowns evaluated at the interface from the left and right triangles, respectively. Then we have an equivalent expression for $\hat{\boldsymbol{F}} \cdot \boldsymbol{n}$:

$$\hat{\boldsymbol{F}} \cdot \boldsymbol{n}_{\eta(c,\pm)} = \begin{cases} \boldsymbol{v}^j \cdot \boldsymbol{n}_{\eta(c,\pm)} h^j_{\eta(c,\pm)}, & \boldsymbol{v}^j \cdot \boldsymbol{n}_{\eta(c,\pm)} \geq 0, \\ \boldsymbol{v}^j \cdot \boldsymbol{n}_{\eta(c,\pm)} h^j_{\eta(c,\mp)}, & \boldsymbol{v}^j \cdot \boldsymbol{n}_{\eta(c,\pm)} < 0, \end{cases} \tag{11}$$

which means, definition (10) can recover the first-order upwind principle. To be consistent with the evaluation of fluxes at interior faces, we calculate the boundary flux as $\hat{\boldsymbol{G}} \cdot \boldsymbol{n} = \boldsymbol{v}^j \cdot \boldsymbol{n}\hat{h}^j + |\boldsymbol{v}^j \cdot \boldsymbol{n}|$ $(g^j - \hat{h}^j)$, where g^j is the boundary value of h^j and $\boldsymbol{n}$ is the outward unit normal vector at the boundary pointing into the flow field.

3. Results and discussions

For verification, we first consider the steady gas flow along a channel of a square cross-section in the $x_1 - x_2$ plane, subject to a small constant acceleration $a_3 = 0.5$ in the x_3 direction, where (x_1, x_2, x_3) are the Cartesian coordinates in the physical space. It is assumed that the channel length is significantly larger than the dimension of its cross-section, thus the end effect can be neglected and the flow field only varies in the x_1 and x_2 directions. The HDG method of k up to 4 is applied to the linearised BGK equation. It is assumed that gas molecules undergo diffuse reflection on the surface and perturbed VDF at the boundary is always zero for this problem. The convergence criterion for iteration (5) is that the increment in flow velocity between two successive iteration steps $|\int u_3^{(t+1)} - u_3^{(t)} \mathrm{d}\Delta| / |\int u_3^{(t)} \mathrm{d}\Delta|$ is less than 10^{-5}. We call the routines in Intel Math Kernel Library (MKL) to invert a matrix, and the Intel MKL PARDISO (Schenk and Gärtner 2004) package to solve the global linear system. The test is done in double precision on a single Intel Xeon-E5-2680 processor. The velocity grid is fixed with 24 non-uniform points distributing within a truncation of $[-4, 4]$ in each direction (Su et al. 2017). Further refinement of the velocity grid would only improve the solutions by a magnitude no more than 0.5%. The flow is resolved on a domain of $\Delta = [0, 1] \times [0, 1]$, which is partitioned with uniform triangles, as shown in Figure 1(a). The typical flow velocity contours obtained by the HDG with $k = 4$ at $Kn = 0.018$, 0.089 and 0.89 are shown in Figure 1(b–d), respectively. The numbers of triangles in these cases are 50, 32 and 18, respectively. The maximum velocity emerges in the centre of the flow field. As Knudsen number (degree of rarefaction) increases, the maximum velocity reduces while the slip velocity in the vicinity of solid surfaces increases.

To investigate the performance of the proposed scheme, the L_1 error of the dimensionless mass flow rate (MFR, $M = \int u_3 \mathrm{d}\Delta$), the number of iterative steps and the CPU time are listed in Table 1, for various numbers of triangles and degrees of approximation polynomials. The solutions from the discrete unified gas kinetic scheme, which have been verified for a wide range of rarefactions (Wang et al. 2018), are used as a reference for error estimation. Note that, in some cases, the error does not reduce monotonically with an increasing number of triangles. This is due to the fact that the reference data are also numerical. As expected, on the same spatial grid, the HDG solutions of higher order accuracy are obtained using higher order approximating polynomials. Thereby, to achieve the same order of accuracy, the solvers with higher order polynomials require fewer triangles. On the other hand, for each rarefaction level, the solvers with different degrees of polynomials require almost the same number of iterations to obtain steady-state solutions. Since fewer triangles are needed, the higher the order of the solver, the less is the CPU time. For instance, at $Kn = 0.018$, the CPU time to obtain solution of $\sim 0.7\%$ error for $k = 4$ is about 40% and 14% of that for the solvers with $k = 3$ and $k = 2$, respectively. It is also interesting to compare the performances of the HDG and the original DG methods. The comparison is for the flow at $Kn = 0.089$. In the DG scheme, the same semi-implicit iterative scheme is applied. Unlike HDG, it doesn't resolve $\hat{h}^j$. Instead, a global linear system is directly built up for all the DoF of field variables h^j. The numerical flux is modelled through

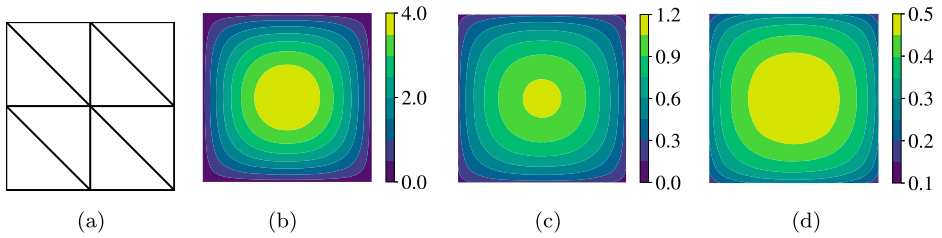

Figure 1. HDG solution ($k = 4$) of Poiseuille flow along a channel of square cross-section. (a) mesh, (b) $Kn = 0.018$, (c) u_3 at $Kn = 0.089$ and (d) u_3 at $Kn = 0.89$.

Table 1. Poiseuille flow along a channel with square cross-section solved by the HDG.

		$Kn = 0.018$			$Kn = 0.089$			$Kn = 0.89$		
k	M_{el}	L_1 error	Itr	t_c (s)	L_1 error	Itr	t_c (s)	L_1 error	Itr	t_c (s)
1	8	3.15×10^{-1}	1283	111.6	9.92×10^{-2}	131	10.8	1.05×10^{-2}	15	1.3
	18	1.49×10^{-1}	1468	284.5	4.28×10^{-2}	137	24.6	4.18×10^{-3}	15	2.6
	32	8.34×10^{-2}	1539	529.4	2.32×10^{-2}	139	45.0	2.20×10^{-3}	15	4.6
	50	5.26×10^{-2}	1570	852.0	1.46×10^{-2}	140	70.8	1.37×10^{-3}	15	9.3
2	8	2.46×10^{-2}	1595	214.3	6.62×10^{-3}	141	18.0	4.04×10^{-4}	15	1.9
	18	9.99×10^{-3}	1612	498.4	3.65×10^{-3}	141	39.8	3.98×10^{-4}	15	4.2
	32	7.16×10^{-3}	1615	887.0	2.93×10^{-3}	141	71.5	3.65×10^{-4}	15	8.0
	50	6.29×10^{-3}	1615	1380.8	2.64×10^{-3}	141	112.5	3.47×10^{-4}	15	12.2
3	8	6.54×10^{-3}	1615	305.5	2.84×10^{-3}	141	24.2	3.71×10^{-4}	15	3.6
	18	5.78×10^{-3}	1615	696.3	2.50×10^{-3}	141	55.6	3.43×10^{-4}	15	9.2
	32	5.59×10^{-3}	1615	1248.1	2.39×10^{-3}	141	97.8	3.29×10^{-4}	15	10.6
	50	6.75×10^{-3}	1615	1939.9	2.33×10^{-3}	141	160.7	3.24×10^{-4}	15	17.5
4	2	6.75×10^{-3}	1615	122.7	2.77×10^{-3}	141	10.0	6.90×10^{-5}	15	1.1
	8	5.67×10^{-3}	1615	490.0	2.44×10^{-3}	141	37.7	3.33×10^{-4}	15	4.0
	18	5.48×10^{-3}	1615	1125.0	2.33×10^{-3}	141	89.0	3.24×10^{-4}	15	9.4
	32	5.37×10^{-3}	1616	2008.3	2.29×10^{-3}	141	152.4	3.20×10^{-4}	15	18.2

Note: Itr denotes the number of iteration steps to satisfy the residual less than 10^{-5}, and t_c is the CPU time.

the upwind scheme, see Equation (11). The number of DoF N_{dof}, the L_1 error in MFR, the number of iterative steps, and the CPU time are listed in Table 2 for both approaches. It shows that HDG and DG with the same order approximating polynomial yield the same solution (MFRs have at least 7 same significant digits) on the same mesh and consume the same number of iterative steps. For HDG, the numbers of globally coupled DoF is $N_{dof} = M_v M_{fc}(k+1)$, while for DG, $N_{dof} = M_v M_{el}(k+1)(k+2)/2$. Here, M_{fc} is around 1.7 times of M_{el}. It is found that N_{dof} in HDG is smaller than the one in DG when $k \geq 2$. The higher order and

Table 2. Comparison of the implicit HDG and DG.

		HDG				DG			
k	M_{el}	N_{dof}/M_v	L_1 error	Itr	t_c (s)	N_{dof}/M_v	L_1 error	Itr	t_c (s)
1	8	32	9.92×10^{-2}	131	10.8	24	9.92×10^{-2}	131	4.9
	32	112	2.32×10^{-2}	139	45.0	96	2.32×10^{-2}	139	21.4
	72	240	1.02×10^{-2}	141	160.0	216	1.02×10^{-2}	141	54.5
	128	416	6.22×10^{-3}	141	180.2	384	6.22×10^{-3}	141	116.6
2	8	48	6.62×10^{-3}	141	18.0	48	6.62×10^{-3}	141	11.6
	32	168	2.93×10^{-3}	141	71.5	192	2.93×10^{-3}	141	52.5
	72	360	2.51×10^{-3}	141	169.7	432	2.51×10^{-3}	141	140.3
	128	624	2.39×10^{-3}	141	316.1	768	2.39×10^{-3}	141	276.6
3	8	64	2.84×10^{-3}	141	24.2	80	2.84×10^{-3}	141	23.2
	32	224	2.39×10^{-3}	141	97.8	320	2.39×10^{-3}	141	110.2
	72	480	2.30×10^{-3}	141	240.9	720	2.30×10^{-3}	141	284.5
	128	832	2.28×10^{-3}	141	465.8	1280	2.28×10^{-3}	141	523.5
4	8	80	2.44×10^{-3}	141	37.7	120	2.44×10^{-3}	141	44.0
	32	280	2.29×10^{-3}	141	152.4	480	2.29×10^{-3}	141	216.1
	72	600	2.27×10^{-3}	141	381.2	1080	2.27×10^{-3}	141	545.0
	128	1040	2.26×10^{-3}	141	735.1	1920	2.26×10^{-3}	141	893.8

Notes: Itr denotes the number of iteration steps to satisfy the residual less than 10^{-5}, N_{dof} denotes the number of globally coupled DoF, and t_c is the CPU time. Note that Itr and L_1 in both methods are exactly the same.

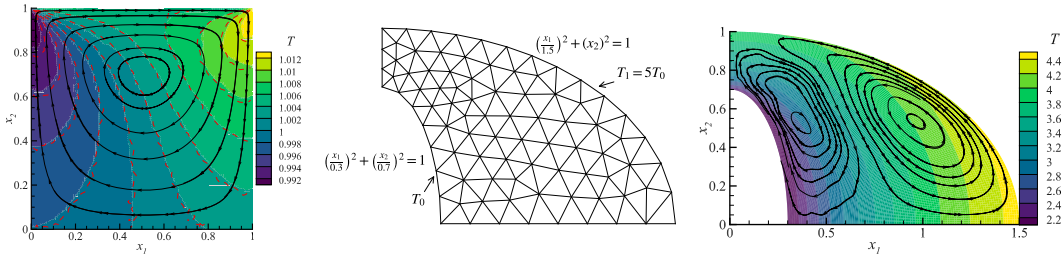

Figure 2. Examples of HDG solutions to the nonlinear Boltzmann equation. (a) Cavity flow driven by top lid with $U = 0.148$ at $Kn = 1$. (b) Configuration for flow induced by temperature inhomogeneity. (c) Thermal flow induced by temperature inhomogeneity at $Kn = 0.5$.

more triangles, the more significant this difference will be. As a consequence, the HDG method cost less CPU time than the DG method when $k > 2$. For example, HDG with $k = 4$ on 128 triangles can save 30% CPU time, compared to the original DG method. Note that N_{dof} of HDG with $k = 2$ is smaller than that of DG. However, it is not more efficient, since extra effort is required to recover field solution from trace solution in the HDG scheme.

Finally, we show the solutions of HDG to the nonlinear Boltzmann equation for the square cavity flow driven by its top lid of horizontal velocity $U = 0.148$ at $Kn = 1$ and thermal flow induced by non-uniform temperature field between two coaxial elliptic cylinders at $Kn = 0.5$. The temperature contours from HDG (background) and the direct simulation Monte Carlo (DSMC) method (dashed lines) (John, Gu, and Emerson 2010), as well as the streamlines for cavity flow are shown in Figure 2(a). The HDG solution, obtained with $k = 4$ on 72 uniform triangles using $108 \times 108 \times 24$ discrete velocities, agrees well with the DSMC result. The geometry and solutions of gas temperature and streamlines for the thermal flow are illustrated in Figure 2(b–c), where the cold cylinder is encompassed in a hot chamber. Only the upper right quarter of the domain is shown due to symmetry. Unlike the continuum flow where the bulk velocity is zero, at rarefied conditions, temperature inhomogeneity induces anisotropic momentum transfer that produces gas flow with two vortexes in the field. Temperature distribution from HDG ($k = 2$, on 142 unstructured triangles and $96 \times 96 \times 24$ discrete velocities) is closed to the DSMC results (Aoki, Sone, and Waniguchi 1998), which shows the capacity of the developed solver to deal with more complex geometry. The full diffuse wall condition is used in the two cases.

4. Conclusions

In summary, we have developed a high-order HDG solver for the solution of gas kinetic equations on an arbitrary triangular mesh. The velocity space is first discretised using a quadrature rule. Then, the discrete molecular VDFs and their traces are approximated in piecewise polynomial space of degree up to 4 on the spatial mesh and the mesh skeleton, respectively. Based on the upwind scheme, a numerical flux has been designed to evaluate the convection between adjacent cells. By imposing the continuity of the normal flux, a global system is setup in terms of the unknown traces only. Once the traces are resolved, the VDFs are updated in an element-by-element fashion. The boundary condition has been implemented in the same framework as the calculation of flux on interfaces. Finally, an implicit iterative scheme is employed to obtain the steady-state solution. Performance analysis shows that, to obtain the results with the same order of accuracy, the scheme with higher order approximating polynomial requires fewer triangles in spatial discretisation. As a result, the computational time and memory consumption are reduced. Furthermore, compared to the original DG scheme, the HDG solver is more efficient when the degree of approximating polynomial is larger than 2, since HDG has a fewer number of globally coupled degrees of freedom.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Peng Wang http://orcid.org/0000-0002-6595-0950

References

Aoki, K., Y. Sone, and Y. Waniguchi. 1998. "A Rarefied Gas Flow Induced by a Temperature Field: Numerical Analysis of the Flow Between Two Coaxial Elliptic Cylinders with Different Uniform Temperatures." *Computers & Mathematics with Applications* 35 (1): 15–28.

Bhatnagar, P. L., E. P. Gross, and M. Krook. 1954. "A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems." *Physical Review* 94: 511–525.

Cercignani, C. 1988. *The Boltzmann Equation and Its Applications*. New York: Springer-Verlag.

Chapman, S., and T. G. Cowling. 1970. *The Mathematical Theory of Non-uniform Gases*. 3rd ed. New York: Cambridge University Press.

Cockburn, B., J. Gopalakrishnan, and R. Lazarov. 2009. "Unified Hybridization of Discontinuous Galerkin, Mixed, and Continuous Galerkin Methods for Second Order Elliptic Problems." *SIAM Journal on Numerical Analysis* 47 (2): 1319–1365.

Cockburn, B., and C.-W. Shu. 2001. "Runge–Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems." *Journal of Scientific Computing* 16 (3): 173–261.

Dimarco, G., and L. Pareschi. 2014. "Numerical Methods for Kinetic Equations." *Acta Numerica* 23: 369–520.

Evans, B., K. Morgan, and O. Hassan. 2011. "A Discontinuous Finite Element Solution of the Boltzmann Kinetic Equation in Collisionless and BGK Forms for Macroscopic Gas Flows." *Applied Mathematical Modelling* 35 (3): 996–1015.

Giorgiani, G., S. Fernández-Méndez, and A. Huerta. 2013. "Hybridizable Discontinuous Galerkin p-Adaptivity for Wave Propagation Problems." *International Journal for Numerical Methods in Fluids* 72 (12): 1244–1262.

Gobbert, M. K., S. G. Webster, and T. S. Cale. 2007. "A Galerkin Method for the Simulation of the Transient 2-D/2-D and 3-D/3-D Linear Boltzmann Equation." *Journal of Scientific Computing* 30 (2): 237–273.

Huerta, A., A. Angeloski, X. Roca, and J. Peraire. 2013. "Efficiency of High-order Elements for Continuous and Discontinuous Galerkin Methods." *International Journal for Numerical Methods in Engineering* 96 (9): 529–560.

John, Benzi, Xiao-Jun Gu, and David R. Emerson. 2010. "Investigation of Heat and Mass Transfer in a Lid-Driven Cavity under Nonequilibrium Flow Conditions." *Numerical Heat Transfer, Part B: Fundamentals* 58 (5): 287–303.

Kirby, R. M., S. J. Sherwin, and B. Cockburn. 2012. "To CG or to HDG: A Comparative Study." *Journal of Scientific Computing* 51 (1): 183–212.

Kitzler, G., and J. Schöberl. 2015. "A High Order Space-Momentum Discontinuous Galerkin Method for the Boltzmann Equation." *Computers & Mathematics with Applications* 70 (7): 1539–1554.

Kubatko, E. J., C. Dawson, and J. J. Westerink. 2008. "Time Step Restrictions for Runge–Kutta Discontinuous Galerkin Methods on Triangular Grids." *Journal of Computational Physics* 227 (23): 9697–9710.

Mieussens, L. 2014. "A Survey of Deterministic Solvers for Rarefied Flows (Invited)." *AIP Conference Proceedings* 1628 (1): 943–951.

Nguyen, N. C., J. Peraire, and B. Cockburn. 2010. "A Hybridizable Discontinuous Galerkin Method for Stokes Flow." *Computer Methods in Applied Mechanics and Engineering* 199 (9): 582–597.

Peraire, J., N. Nguyen, and B. Cockburn. 2010. "A Hybridizable Discontinuous Galerkin Method for the Compressible Euler and Navier-Stokes Equations." 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, FL, 1–11.

Reed, W. H., and T. R. Hill. 1973. *Triangular Mesh Methods for the Neutron Transport Equation*. Technical Report 836.

Schenk, O., and K. Gärtner. 2004. "Solving Unsymmetric Sparse Systems of Linear Equations with PARDISO." *Future Generation Computer Systems* 20 (3): 475–487.

Su, W., A. A. Alexeenko, and G. Cai. 2015. "A Parallel Runge–Kutta Discontinuous Galerkin Solver for Rarefied Gas Flows Based on 2D Boltzmann Kinetic Equations." *Computers & Fluids* 109: 123–136.

Su, W., S. Lindsay, H. Liu, and L. Wu. 2017. "Comparative Study of the Discrete Velocity and Lattice Boltzmann Methods for Rarefied Gas Flows Through Irregular Channels." *Physics Review E* 96: 023309.

Wang, P., M. T. Ho, L. Wu, Z. Guo, and Y. Zhang. 2018. "A Comparative Study of Discrete Velocity Methods for Low-speed Rarefied Gas Flows." *Computers & Fluids* 161: 33–46.

Wu, L., C. White, T. J. Scanlon, J. M. Reese, and Y. Zhang. 2013. "Deterministic Numerical Solutions of the Boltzmann Equation Using the Fast Spectral Method." *Journal of Computational Physics* 250: 27–52.

Appendix

Unknowns are approximated by the nodal shape functions N_l in each triangle Δ_i or by $\hat{N}_l$ on each face Γ_c: $h_i^j = \sum_{l=1}^{K_{\rm el}} N_i^l H_{i,l}^j$, $\hat{h}_c^j = \sum_{l=1}^{K_{\rm fc}} \hat{N}_c^l \hat{H}_{c,l}^j$, where $K_{\rm el} = (k+1)(k+2)/2$ and $K_{\rm fc} = k+1$ are the numbers of DoF. Denote $\mathbf{H}^{i,j}$ as the vector of nodal value of h^j on each Δ_i, $\hat{\mathbf{H}}^{i,j}$ as the vector summing all the nodal value of $\hat{h}^j$ on the three faces of Δ_i, and $\hat{\mathbf{H}}^{c,j}$ as the vector of the nodal value of $\hat{h}^j$ on each Γ_c; the local and global problems can be rewritten in the matrix form

$$\mathbf{H}^{i,j} = \left[\mathbf{A}^{i,j}\right]^{-1} \mathbf{S}^{i,j} + \left[\mathbf{A}^{i,j}\right]^{-1} \hat{\mathbf{A}}^{i,j} \hat{\mathbf{H}}^{i,j}, \tag{A1}$$

and

$$\begin{aligned} \hat{\mathbf{B}}^{c,j} \hat{\mathbf{H}}^{c,j} &= \mathbf{B}^{\eta(c,+),j} \mathbf{H}^{\eta(c,+),j} + \mathbf{B}^{\eta(c,-),j} \mathbf{H}^{\eta(c,-),j}, \quad \text{on } \Gamma \backslash \partial\Omega, \\ \hat{\mathbf{B}}^{c,j} \hat{\mathbf{H}}^{c,j} &= \mathbf{B}^{\eta(c,+),j} \mathbf{H}^{\eta(c,+),j} + \hat{\mathbf{S}}^{c,j}, \quad \text{on } \Gamma \cap \partial\Omega, \end{aligned} \tag{A2}$$

where

$$\begin{aligned} \mathbf{A}_{ml}^{i,j} = \frac{\sqrt{\pi}}{2Kn} \left(N_i^m, N_i^l\right)_{\Delta_i} + \sum_{e=1}^{3} |\boldsymbol{v}^j \cdot \boldsymbol{n}| \langle N_i^m, N_i^l \rangle_{\partial \Delta_i^e} \\ - \left(\boldsymbol{v}^j \cdot \nabla N_i^m, N_i^l\right)_{\Delta_i}, \end{aligned}$$

$$\hat{\mathbf{A}}^{i,j,e}_{ml} = \left(|\boldsymbol{v}^j \cdot \boldsymbol{n}| - \boldsymbol{v}^j \cdot \boldsymbol{n}\right) \langle N_i^m, \hat{N}^l_{\sigma(i,e)} \rangle_{\partial \Delta_i^e}, \quad \mathbf{S}^{i,j}_m = \left(N_i^m, s^j\right)_{\Delta_i},$$
$$\hat{\mathbf{B}}^{c,j}_{ml} = \langle \hat{N}_c^m, \hat{N}_c^l \rangle_{\Gamma_c}, \mathbf{B}^{\eta(c,\pm),j}_{ml} = \frac{1}{2} \langle \hat{N}_c^m, N^l_{\eta(c,\pm)} \rangle_{\Gamma_c},$$
$$\hat{\mathbf{S}}^{c,j}_m = \frac{1}{2} \langle \hat{N}_c^m, g^j \rangle_{\Gamma_c}. \tag{A3}$$

By eliminating the unknowns $\mathbf{H}^{i,j}$ with Equation (A1) and assembling the equations of the global problem over all faces, the global problem becomes $\mathbb{K}^j \hat{\mathbf{H}}^j = \mathbb{R}^j$ with

$$\mathbb{K}^j = \mathop{\mathsf{A}}_{c=1}^{M_{\mathrm{fc}}} \left\{ \hat{\mathbf{B}}^{c,j} - \mathbf{B}^{\eta(c,\pm),j} \left[\mathbf{A}^{\eta(c,\pm),j}\right]^{-1} \hat{\mathbf{A}}^{\eta(c,\pm),j} \right\},$$
$$\mathbb{R}^j = \mathop{\mathsf{A}}_{c=1}^{M_{\mathrm{fc}}} \left\{ \mathbf{B}^{\eta(c,\pm),j} \left[\mathbf{A}^{\eta(c,\pm),j}\right]^{-1} \mathbf{S}^{\eta(c,\pm),j} + \hat{\mathbf{S}}^{c,j} \right\}. \tag{A4}$$

Here, A is the conventional assembly operator.

When the nonlinear Boltzmann equation is solved, the unknowns are the full VDF and its trace, which are approximated as $f_i^j = \sum_{l=1}^{K_{\mathrm{el}}} N_i^l F_{i,l}^j$, $\hat{f}_c^j = \sum_{l=1}^{K_{\mathrm{fc}}} \hat{N}_c^l \hat{F}_{c,l}^j$ with F and $\hat{F}$ being the corresponding nodal values. Then the terms $\sqrt{\pi}(N_i^m, N_i^l)_{\Delta_i}/2Kn$ and $(N_i^m, s^j)_{\Delta_i}$ in (A3) are replaced by

$$\sum_{r=1}^{K_{\mathrm{el}}} \left(N_i^m, N_i^r N_i^l\right)_{\Delta_i} \iint B F^j_{*,i,r} \mathrm{d}\Omega \mathrm{d}\boldsymbol{v}_* \tag{A5}$$

and

$$\sum_{r=1}^{K_{\mathrm{el}}} \sum_{l=1}^{K_{\mathrm{el}}} \left(N_i^m, N_i^r N_i^l\right)_{\Delta_i} \iint B F'^j_{*,i,r} F'^j_{i,l} \mathrm{d}\Omega \mathrm{d}\boldsymbol{v}_* - \boldsymbol{a} \cdot \sum_{l=1}^{K_{\mathrm{el}}} \left(N_i^m, N_i^l\right)_{\Delta_i} \frac{\partial F^j_{i,l}}{\partial \boldsymbol{v}}, \tag{A6}$$

respectively. The integral $\iint \cdot \mathrm{d}\Omega \mathrm{d}\boldsymbol{v}_*$ and derivative $\partial/\partial \boldsymbol{v}$ are calculated via the fast spectral method.