Commun. Comput. Phys.
doi: 10.4208/cicp.291013.240314a

Vol. **16**, No. 4, pp. 956-982
October 2014

A Local Velocity Grid Approach for BGK Equation

Florian Bernard[1,2,*], Angelo Iollo[2] and Gabriella Puppo[3]

[1] *Department of Mechanical and Aerospace Engineering, Politecnico di Torino, 10129 Torino, Italy.*
[2] *Univ. Bordeaux, IMB, UMR 5251, F-33400 Talence, France.*
INRIA, F-33400 Talence, France.
[3] *Dip. di Scienza ed Alta Tecnologia, Università dell'Insubria, Como, Italy.*

Received 29 October 2013; Accepted (in revised version) 24 March 2014

Available online 8 August 2014

Abstract. The solution of complex rarefied flows with the BGK equation and the Discrete Velocity Method (DVM) requires a large number of velocity grid points leading to significant computational costs. We propose an adaptive velocity grid approach exploiting the fact that locally in space, the distribution function is supported only by a sub-set of the global velocity grid. The velocity grid is adapted thanks to criteria based on local temperature, velocity and on the enforcement of mass conservation. Simulations in 1D and 2D are presented for different Knudsen numbers and compared to a global velocity grid BGK solution, showing the computational gain of the proposed approach.

AMS subject classifications: 76P05

Key words: Kinetic models, BGK model, discrete velocity method.

1 Introduction

In hydrodynamic regimes, fluid flows can be simulated thanks to standard models such as Navier-Stokes or compressible Euler equations. However, some regimes cannot be qualified as hydrodynamic and the continuum equations are not able to correctly describe the dynamics of the flow. The parameter that dictates whether or not a flow is hydrodynamic is the Knudsen number Kn. It is defined as the ratio between the mean free path λ between the particles and the characteristic length of the physical problem L. When this number goes towards zero, the hydrodynamic regime is reached. For large

*Corresponding author. *Email addresses:* florian.bernard@polito.it (F. Bernard), angelo.iollo@math.u-bordeaux1.fr (A. Iollo), gabriella.puppo@uninsubria.it (G. Puppo)

Knudsen numbers (usually higher than 10^{-2}), the regime is qualified as rarefied and the governing equation is the Boltzmann equation [9].

Directly solving the Boltzmann equation is computationally prohibitive because of the high-dimensionality and of the complexity of the collision operator, see for instance [14]. However, several numerical methods exist in the literature to get around this difficulty. One of the most popular methods is the Direct Simulation Monte Carlo (DSMC, [7]). This is a statistical approximation of the Boltzmann equation that is numerically viable for high Knudsen numbers, but becomes very costly for low Knudsen numbers as the number of collisions increases and a strong restriction appears on the time step [21]. For these regimes it also has the disadvantage of generating noisy results.

Hence, attempts have been made to derive numerical solvers for the Boltzmann equation which are not based on particles [25]. For a recent review of deterministic methods for Boltzmann equation, see [15], and the references therein. Numerical solvers for Boltzmann equation have to deal with the complexity of the collision term. For this reason, simplified kinetic models can be extremely attractive. A particularly successful model is the BGK model [6], in which the collision term of Boltzmann equation is approximated as a relaxation towards a Maxwellian distribution function. More recently, an extension to BGK has been widely studied to include thermal effects which are not correctly represented in the standard BGK model. It is the ES-BGK model [3]. The ability of this model to correctly approximate the full Boltzmann equation has been studied in [2].

For small relaxation times, a strong restriction on the time step exists also for explicit schemes to integrate the BGK model, but this constraint can be treated by using implicit schemes, such as IMEX [19,22] which have been successfully applied to the BGK model, [23] and [13]. Recently, this technique has been extended to the efficient integration of the ES-BGK model [1,17]. IMEX numerical schemes provide efficient solvers which give the correct asymptotic properties [16,18,24].

Such models are usually implemented thanks to the discrete velocity method (DVM) [8], that requires a discretization of the velocity space. The same global velocity grid is employed for each space discretization point. A major bottleneck of such an approach is the size of the velocity grid. Indeed, the boundaries of the velocity grid have to be determined according to the maximum velocity and temperature in the flow while its spacing depends on the minimum temperature. In realistic test cases like hypersonic atmosphere re-entries or satellite engines, very strong gradients in temperature can be observed. Then, the solution needs a very large and fine grid, so that the computational requirements increase quickly and make the phenomenon difficult to simulate especially in 3D.

This important problem has already been addressed in [11]. There, an AMR technique is presented in velocity space for the unified gas-kinetic scheme [27]. A careful work has been carried out on the error estimation due to quad/octree meshes to make the method accurate. Since the grids are not connected, interpolation is required in the transport step, to match the velocity grids in the neighbouring space cells. This interpolation can lead to approximation errors and possible increasing computational time. In [4] a criteria to

decide whether or not the grid has to be refined or coarsened is presented and proven to be efficient with and AMR algorithm for steady flows. However, dealing with unsteady flows requires a grid adaptation at each time step and thus, interpolation of the solution between two successive grids.

In this paper, a local velocity grid approach is presented for the BGK model. The idea is to build local velocity grids in each space cell, keeping a high accuracy and saving as much computational time as possible, thanks to a simple algorithm. The proposed approach is at worse as expensive as a standard DVM.

The new approach presented here avoids interpolations and refinement steps. Our numerical tests show that this method significantly decreases the computational time since the velocity grid is often drastically reduced with a negligible overhead. The efficiency of the method will be presented on 1D and 2D test cases for different regimes.

2 Governing equations

In this section, we detail the full model and its 1D and 2D reductions.

2.1 The BGK model

A typical property of the Boltzmann equation is given by the Chapman-Enskog expansion [10]. It ensures the asymptotic limit towards Navier-Stokes equation for small Knudsen numbers ($\mathrm{Kn}\simeq 10^{-3}$) and towards compressible Euler equations for $\mathrm{Kn}\to 0$. The BGK model respects this property and is an approximation of the Boltzmann equation particularly viable for moderate Knudsen number, in the kinetic regime (roughly $\mathrm{Kn}\leq 1$). This model gives a simple representation of the collision term:

$$\frac{\partial f}{\partial t}(x,\xi,t)+\xi\cdot\nabla_x f(x,\xi,t)=\frac{1}{\tau}\left(M_f(x,\xi,t)-f(x,\xi,t)\right), \tag{2.1}$$

where τ is the relaxation time and f is a density distribution function that depends on space $x\in\mathbb{R}^d$, microscopic velocity $\xi\in\mathbb{R}^N$, and time t with $f_0(x,\xi,t=0)$ as initial data. M_f is a Maxwellian distribution function obtained as follows:

$$M_f(x,\xi,t)=\frac{\rho(x,t)}{(2\pi RT(x,t))^{N/2}}\exp\left(-\frac{|\,\xi-U(x,t)\,|^2}{2RT(x,t)}\right), \tag{2.2}$$

where R is the universal gas constant and $T(x,t)$, $U(x,t)$ and $\rho(x,t)$ are macroscopic values of temperature, velocity and density calculated from the moments of f defined by:

$$\begin{pmatrix}\rho(x,t)\\ \rho(x,t)U(x,t)\\ E(x,t)\end{pmatrix}=\int_{\mathbb{R}^N} f(x,\xi,t)m(\xi)d\xi \quad\text{with}\quad m(\xi)=\begin{pmatrix}1\\ \xi\\ \frac{1}{2}|\,\xi\,|^2\end{pmatrix}. \tag{2.3}$$

$E(x,t)$ is the total energy obtained as follows:

$$E(x,t)=\frac{N}{2}\rho RT(x,t)+\frac{1}{2}\rho(x,t)\,|\,U(x,t)\,|^2\,. \tag{2.4}$$

We consider a monoatomic gas for which the ratio of specific heats γ can be calculated as:

$$\gamma=1+\frac{2}{N}.$$

In the following, $N=3$ and therefore γ=5/3.

The relaxation time for the BGK model can be written as:

$$\tau^{-1}=c\rho T^{1-\nu},$$

where c is a constant that depends on the reference conditions and ν is the exponent of the viscosity law of the gas. The constant c can be expressed as:

$$c=\frac{RT_0^{\nu}}{\mu_0},$$

where μ_0 is the reference viscosity of the gas at the reference temperature T_0.

In the following, we will consider the BGK equation in dimensionless form, that is:

$$\frac{\partial f}{\partial t}(x,\xi,t)+\xi\cdot\bigtriangledown_x f(x,\xi,t)=\frac{1}{\tau}\left(M_f(x,\xi,t)-f(x,\xi,t)\right), \tag{2.5}$$

$$M_f(x,\xi,t)=\frac{\rho(x,t)}{(2\pi T(x,t))^{N/2}}\exp\left(-\frac{|\,\xi-U(x,t)\,|^2}{2T(x,t)}\right), \tag{2.6}$$

$$\tau^{-1}=Kn_{\infty}\rho T^{1-\nu}. \tag{2.7}$$

ρ, U and T are now the dimensionless macroscopic quantities and Kn_{∞} is the Knudsen number in reference conditions.

For 1D and 2D cases, the non-leading dimensions can be dropped in velocity (two in 1D and one in 2D) by integrating the distribution function directly along these dimensions. Two distribution functions ϕ and ψ of lower dimensions are obtained to create the reduced model (see [12] and [5] for more details). For instance in 1D with $\xi=(\xi_u,\xi_v,\xi_w)$:

$$\begin{cases}\partial_t\phi(x,\xi_u,t)+\xi_u\partial_x\phi(x,\xi_u,t)=\dfrac{1}{\tau}\left(M_\phi(x,\xi_u,t)-\phi(x,\xi_u,t)\right),\\ \partial_t\psi(x,\xi_u,t)+\xi_u\partial_x\psi(x,\xi_u,t)=\dfrac{1}{\tau}\left(M_\psi(x,\xi_u,t)-\psi(x,\xi_u,t)\right),\end{cases} \tag{2.8}$$

where $\phi=\int_{\mathbb{R}^2} f d\xi_v d\xi_w$, $\psi=\int_{\mathbb{R}^2}\frac{1}{2}(\xi_v^2+\xi_w^2)f d\xi_v d\xi_w$, $M_\phi=\int_{\mathbb{R}^2} M_f d\xi_v d\xi_w$ and $M_\psi=\int_{\mathbb{R}^2}\frac{1}{2}(\xi_v^2+\xi_w^2)M_f d\xi_v d\xi_w$.

2.2 The discrete model in velocity space

By construction the distribution function f and the corresponding Maxwellian distribution function M_f satisfy:

$$\int_{\mathbb{R}^3} M_f m(\xi) d\xi = \begin{pmatrix} \rho(x,t) \\ \rho(x,t)U(x,t) \\ E(x,t) \end{pmatrix} = \int_{\mathbb{R}^3} f m(\xi) d\xi.$$

This is essential to ensure conservation of mass, momentum and energy because multiplying (2.1) by $m(\xi)$ and integrating in velocity space one recovers the conservation laws for mass, momentum and energy:

$$\partial_t \int_{\mathbb{R}^3} m(\xi) f(\xi) d\xi + \partial_x \int_{\mathbb{R}^3} \xi m(\xi) f(\xi) d\xi = 0.$$

In the discrete case, a grid must be introduced in velocity space and the integrals are evaluated by quadrature. Let $\langle .,. \rangle$ denote the quadrature rule and:

$$\begin{pmatrix} \rho \\ \rho U \\ E \end{pmatrix} = \langle f(\xi), m(\xi) \rangle$$

be the discrete moments of f. Now:

$$\langle M_f(\xi), m(\xi) \rangle \neq \begin{pmatrix} \rho \\ \rho U \\ E \end{pmatrix} = \langle f(\xi), m(\xi) \rangle$$

due to the error introduced by the quadrature rule and conservation would not hold exactly.

In [20], Mieussens proved that a discrete Maxwellian can be expressed as $\widetilde{M}_f = \exp(\alpha \cdot m(\xi))$ and such that:

$$\langle \widetilde{M}_f(\xi), m(\xi) \rangle = \begin{pmatrix} \rho \\ \rho U \\ E \end{pmatrix}. \tag{2.9}$$

This form holds true also in the continuous case for the exact Maxwellian with:

$$\alpha_c = \left(\ln \left(\frac{\rho(x,t)}{(2\pi RT(x,t))^{d/2}} \right) - \frac{|U(x,t)|^2}{2RT(x,t)}, \frac{U(x,t)}{RT(x,t)}, -\frac{1}{RT(x,t)} \right).$$

For the 1D reduced model, this expression becomes:

$$\langle \widetilde{M_\phi}, m_1(\xi) \rangle + \langle \widetilde{M_\psi}, e_3 \rangle = \langle f, m_1(\xi) \rangle \tag{2.10}$$

with $e_3 = (0,0,1)^T$ and $m_1 = (1, \xi_u, |\xi_u|^2/2)^T$.

In 2D, the discrete Maxwellian of ϕ and ψ is such as:

$$\langle \widetilde{M_\phi}, m_2(\xi)\rangle + \langle \widetilde{M_\psi}, e_4\rangle = \langle f, m(\xi)\rangle \tag{2.11}$$

with $e_4 = (0,0,0,1)^T$ and $m_2 = (1,\xi_u,\xi_v,|\xi_u+\xi_v|^2/2)^T$.

Eqs. (2.10) and (2.11) are non-linear systems in α which can be solved with a Newton-Raphson method. The solution in the continuous case (α_c) will be the starting value for the Newton-Raphson algorithm.

In our case, the trapezoidal quadrature rule is used because it has spectral accuracy for smooth and periodic functions on a uniform grid. For this reason we use a uniform grid symmetric with respect to zero and such that f is negligible outside the grid. In 1D, the velocity grid is defined as:

$$\mathcal{G} = (\xi_i)_{i=-n_v,\cdots,n_v} \quad \text{with} \quad \xi_i = i\Delta\xi.$$

For multidimensional cases, the same discretization is performed independently in all directions.

3 The numerical method

Since the two equations for ϕ and ψ have the same structure, the numerical method will be presented for one equation only where the function f will stand for ϕ and ψ and M_f the corresponding Maxwellian. We use a finite volume method and it will be described for the 2D reduced model.

3.1 The space discretization

The space Ω is discretized on a Cartesian grid with $n \times m$ cells:

$$\Omega = \bigcup_{\substack{i=1\cdots n\\ j=1\cdots m}} \Omega_{i,j} = \bigcup_{\substack{i=1\cdots n\\ j=1\cdots m}} [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$$

such that (x_i, y_j) are the coordinates of the center of the cell (i,j) and $(x_{i+1/2}, y_j)$ are the coordinates of the center of the interface between cells (i,j) and $(i+1,j)$. Eq. (2.5) is integrated on the space cell $\Omega_{i,j} = [x_i - \Delta x/2, x_i + \Delta x/2] \times [y_j - \Delta y/2, y_j + \Delta y/2]$:

$$\frac{\partial f_{i,j}}{\partial t} + \xi \cdot \int_{\partial\Omega_{i,j}} f n_{\partial\Omega_{i,j}} d\sigma = \frac{1}{\tau_{i,j}} \left(M_{f_{i,j}} - f_{i,j} \right), \tag{3.1}$$

where $f_{i,j} = \frac{1}{|\Omega_{i,j}|} \int_{\Omega_{i,j}} f dx dy$ and $M_{f_{i,j}} = \widetilde{M}_f(x_i, y_j)$ the discrete Maxwellian computed from (2.9).

Since a uniform Cartesian grid is considered, the equation can be simply rewritten in terms of fluxes at each numerical interface:

$$\frac{\partial f_{i,j}}{\partial t}+\frac{1}{\Delta x}\left(F_{i+\frac{1}{2},j}-F_{i-\frac{1}{2},j}+F_{i,j+\frac{1}{2}}-F_{i,j-\frac{1}{2}}\right)=\frac{1}{\tau_{i,j}}\left(M_{f_{i,j}}-f_{i,j}\right) \tag{3.2}$$

with $F_{i+\frac{1}{2},j}$ the numerical flux between cell $\Omega_{i,j}$ and cell $\Omega_{i+1,j}$ (with a similar notation for the other fluxes) which is expressed as:

$$F_{i+\frac{1}{2},j}=\max(0,\xi)f_l+\min(0,\xi)f_r \tag{3.3}$$

with f_r and f_l the values of f on the two sides of the interface. The numerical expression of the distribution functions f_l, f_r depends on the reconstruction used at the numerical interface. For a first-order reconstruction, $f_l=f_{i,j}$ and $f_r=f_{i+1,j}$. One can also use a MUSCL reconstruction with slope limiters (MinMod for example) to get second order accuracy:

$$\begin{cases} f_l=f_{i,j}+\dfrac{1}{2}\text{MinMod}\left(f_{i+1,j}-f_{i,j},f_{i,j}-f_{i-1,j}\right), \\ f_r=f_{i+1,j}-\dfrac{1}{2}\text{MinMod}\left(f_{i+1,j}-f_{i,j},f_{i+2,j}-f_{i+1,j}\right). \end{cases} \tag{3.4}$$

3.2 The time discretization

The time discretization can be performed explicitly for all terms. But in this case, the time step would be determined by the space discretization (Δx), the maximum velocity of the velocity grid and the relaxation time τ. For small Knudsen numbers, the relaxation part becomes very stiff (τ very small) and imposes a very strong restriction on the time step. For this reason, the IMEX scheme [23], [22] is chosen. Here the relaxation term is treated implicitly while the convective part is non stiff an explicit scheme can be more efficient.

The time integration for a ν-stages IMEX Runge-Kutta scheme [19] applied to (3.2) reads as follows:

$$\begin{aligned} f_{i,j}^{n+1}&=f_{i,j}^n-\Delta t\sum_{k=1}^{\nu}\tilde{\omega}_k\xi\nabla_x f_{i,j}^{(k)}+\frac{\Delta t}{\tau_{i,j}}\sum_{k=1}^{\nu}\omega_k\left(M_{f_{i,j}}^{(k)}-f_{i,j}^{(k)}\right), \\ f_{i,j}^{(k)}&=f_{i,j}^n-\Delta t\sum_{l=1}^{k-1}\tilde{A}_{k,l}\xi\nabla_x f_{i,j}^{(l)}+\frac{\Delta t}{\tau_{i,j}}\sum_{l=1}^{k}A_{k,l}\left(M_{f_{i,j}}^{(l)}-f_{i,j}^{(l)}\right), \\ f_{i,j}^{(1)}&=f_{i,j}^n+\frac{\Delta t}{\tau_{i,j}}A_{1,1}\left(M_{f_{i,j}}^{(1)}-f_{i,j}^{(1)}\right), \end{aligned} \tag{3.5}$$

where A and $\tilde{A}$ are $\nu\times\nu$ matrices, with $\tilde{A}_{i,s}=0$ if $s\geq i$ and $A_{i,s}=0$ if $s>i$. These coefficients are derived from a double Butcher's tableaux:

$$\begin{array}{c|c} & \tilde{A} \\ \hline & \tilde{\omega}^T \end{array} \qquad\qquad \begin{array}{c|c} & A \\ \hline & \omega^T \end{array}$$

It is worth to remark that to compute the stage k ($f_{i,j}^{(k)}$), we need to know the Maxwellian at stage k which depends a priori on the moments of $f_{i,j}^{(k)}$. But we observe that if we compute the moments of $f_{i,j}^{(k)}$:

$$\begin{aligned}
&< m(\xi), f_{i,j}^{(k)} > \\
=& < m(\xi), f_{i,j}^{n} > -\Delta t \sum_{l=1}^{k-1} \tilde{A}_{k,l} < m(\xi), \xi \nabla_x f_{i,j}^{(l)} > + \frac{\Delta t}{\tau} \sum_{l=1}^{k} A_{k,l} < m(\xi), (M_{f_{i,j}}^{(l)} - f_{i,j}^{(l)}) > \\
=& < m(\xi), f_{i,j}^{n} > -\Delta t \sum_{l=1}^{k-1} \tilde{A}_{k,l} < m(\xi), \xi \nabla_x f_{i,j}^{(l)} >
\end{aligned}$$

because the moments of $M_{f_{i,j}}^{(l)}$ and $f_{i,j}^{(l)}$ are the same. The moments of $f_{i,j}^{(k)}$ can be then computed using already known quantities and therefore the discrete Maxwellian $M_{f_{i,j}}^{(k)}$ at stage k is obtained. All the quantities until stage $k-1$ are known so the equation for stage k becomes:

$$f_{i,j}^{(k)} = \frac{\tau_{i,j}}{A_{k,k}\Delta t + \tau_{i,j}} \left(f_{i,j}^{n} - \Delta t \sum_{l=1}^{k-1} \tilde{A}_{k,l} \xi \nabla_x f_{i,j}^{(l)} + \frac{\Delta t}{\tau_{i,j}} \sum_{l=1}^{k-1} A_{k,l} (M_{f_{i,j}}^{(l)} - f_{i,j}^{(l)}) + \frac{A_{k,k}\Delta t}{\tau_{i,j}} M_{f_{i,j}}^{(k)} \right),$$

$f_{i,j}^{(k)}$ can be then computed explicitly since the right hand side is known.

In our case, we will be interested in first and second order schemes.

IMEX 1st order scheme

0	
1	

Explicit scheme

1	
1	

Implicit scheme

IMEX 2nd order scheme

0	0	0
0	0	0
0	1	0
0	$\frac{1}{2}$	$\frac{1}{2}$

Explicit scheme

$\frac{1}{2}$	0	0
$-\frac{1}{2}$	$\frac{1}{2}$	0
0	$\frac{1}{2}$	$\frac{1}{2}$
0	$\frac{1}{2}$	$\frac{1}{2}$

Implicit scheme

4 A local velocity grid approach

Here, the local velocity approach is presented. In a first part, the algorithm will be described in 1D. Then the approach will be extended to a multidimensional space.

4.1 The 1D algorithm

We start describing the construction of the local velocity grid in 1D. For test cases with strong shocks or with high temperature or velocity differences, the boundaries of the velocity grid have to be very extended with a fine grid spacing. Usually the grid is defined with the following conditions:

$$\mathcal{G}=(\xi_i)_{i=-n_v,\cdots,n_v} \quad \text{such that} \begin{cases} \xi_{\max}=|U|_{\max}+5\sqrt{T_{\max}}, \\ \xi_{\min}=-\xi_{\max}, \\ \Delta\xi\leq\sqrt{T_{\min}}, \end{cases} \tag{4.1}$$

with $|U|_{\max}$ the maximum absolute value of the macroscopic velocity and $T_{\max}$ the maximum temperature over space and time. The velocity grid is then taken symmetric with respect to zero. The first condition ensures that most of the distribution function (more than 99.99%) is contained in the grid for every space cell at any time t if it is not too far from the Maxwellian distribution function. The last condition ensures that the Maxwellian distribution function will be always defined on at least three grid points. In spatial cells, the distribution function will be concentrated on a sub-part of the global velocity grid and will be zero (or negligible) elsewhere.

The idea of the present work is to avoid the computation of the distribution function where it is negligible and more specifically, where the convective fluxes will be zero or negligible.

Local grids in velocity are chosen for each space cell based on the macroscopic quantities. Each local grid is a sub-set of the global velocity grid which would be the one chosen thanks to (4.1).

In particular, considering a global velocity grid with a set of N_v velocity points $(\xi_k)_{k=1,N_v}$, the local grid on a cell i is set as follows:

$$\begin{aligned} &\hat{\xi}^i_{\min}=\xi_l \quad \text{such that} \quad \xi_l=\max_{k\in[1,N_v]}(\xi_k\leq U_i-\alpha\sqrt{T_i}), \\ &\hat{\xi}^i_{\max}=\xi_l \quad \text{such that} \quad \xi_l=\min_{k\in[1,N_v]}(\xi_k\geq |U_i|+\alpha\sqrt{T_i}), \end{aligned} \tag{4.2}$$

where U_i and T_i are the macroscopic velocity and temperature in cell i. $\hat{\xi}^i_{\min}$ and $\hat{\xi}^i_{\max}$ are respectively the minimum and maximum velocity node contained in the local grid for the space cell i. Note that in this way, the interfaces of velocity cells belonging to different local grids are aligned. Then, no interpolation is needed to compute the numerical fluxes

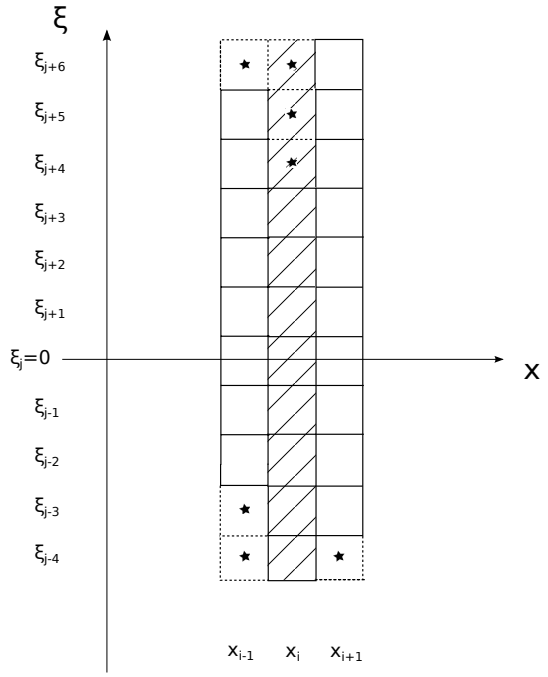

Figure 1: Representation of velocity-space cells in phase space in 1D. Local grids defined in cell $i-1$, i, $i+1$ before the union are in solid lines and stencil used for the transport in space cell i after the union is in dashed lines. Cells in the local grid for space cell i are hatched. Cells with marker ★ are cells added after the union.

in the transport step. Indeed, all the local grids are based on the global grid. Thus, if ξ is outside the local grid at t^n, $f(x,\xi,t^n)$ is set to zero.

Defining in such a way the local velocity grids is not enough as the gas evolves. Let us consider the case where a velocity point ξ_k exists in the velocity grid of a given space cell but not in the velocity grid of the neighbour cell. Even small, the value of the distribution function $f_{i,k}$ is not zero in the first cell while $f_{i+1,k}=0$. Thus a small flux is created and it must contribute to update $f_{i,k}$ and $f_{i+1,k}$. But for the cell that does not contain this microscopic velocity, the value of the distribution function cannot be updated unless that point is included in the local grid and the value of f in that cell may become non negligible in long time. Hence, at the beginning of the time step, the local grids will be set as the union of the local grids contained in the stencil of the numerical flux (see Fig. 1). For a first-order scheme based on a three points stencil:

$$\begin{cases} \xi^i_{\min} = \min(\hat{\xi}^{i-1}_{\min}, \hat{\xi}^{i}_{\min}, \hat{\xi}^{i+1}_{\min}), \\ \xi^i_{\max} = \max(\hat{\xi}^{i-1}_{\max}, \hat{\xi}^{i}_{\max}, \hat{\xi}^{i+1}_{\max}). \end{cases} \tag{4.3}$$

$\xi^i_{\min}$ and $\xi^i_{\max}$ are respectively the new minimum and maximum velocity contained in the local grid for cell i. This step ensures that not only the distribution function is zero

outside the local grid but the numerical fluxes at the grid interfaces are also zero outside $(\xi^i_{\min},\xi^i_{\max})$.

For a second-order scheme in space, there are actually two more cells involved in the stencil of the transport step for each cell. Indeed, to calculate the flux between cell i and cell $i+1$, one needs the slopes in cell i and in cell $i+1$. In the cell $i+1$ the slopes are calculated with (3.4) which involves the distribution function in cell $i+2$. Thus, the union is performed as follows:

$$\begin{cases} \xi^i_{\min} = \min(\hat{\xi}^{i-2}_{\min}, \hat{\xi}^{i-1}_{\min}, \hat{\xi}^{i}_{\min}, \hat{\xi}^{i+1}_{\min}, \hat{\xi}^{i+2}_{\min}), \\ \xi^i_{\max} = \max(\hat{\xi}^{i-2}_{\max}, \hat{\xi}^{i-1}_{\max}, \hat{\xi}^{i}_{\max}, \hat{\xi}^{i+1}_{\max}, \hat{\xi}^{i+2}_{\max}). \end{cases} \tag{4.4}$$

With this approach, the maximum microscopic velocity used for the computation of the transport step is not necessarily the maximum velocity of the global grid $\mathcal{G}$. It may also vary in time but it remains at most the maximum velocity of the global grid. To improve the computational time, at each time step a new Δt can be computed considering the maximum velocity $\xi_{\max}$ of all the local velocity grids. This modification may reduce the CPU time in all cases where the maximum velocity of the global grid is not contained in any local grid.

This approach ensures that for a Maxwellian distribution function and $\alpha \geq 5$ the density error is less than 10^{-6}. However, the error can be larger for high Knudsen numbers, when the distribution function is not Maxwellian. In order to account for these cases, the local grid at the new time step is chosen as follows. First, in a space cell i at time t^n, the local grid $\mathcal{G}^n_i$ is estimated as in (4.2). Then a test is performed on the distribution function to check that the new local grid contains a significant part (set with a tolerance) of the distribution function stored in the old grid $\mathcal{G}^{n-1}_i$. This condition reads as follows:

$$\frac{|\int_{\mathcal{G}^n_i} f^n_i d\xi - \int_{\mathcal{G}^{n-1}_i} f^n_i d\xi|}{\int_{\mathcal{G}^{n-1}_i} f^n_i d\xi} < tol, \tag{4.5}$$

where *tol* is a tolerance (set to 10^{-5} in all numerical test cases). If (4.5) does not hold, $\mathcal{G}^n_i = \mathcal{G}^{n-1}_i$. Finally, the union of the neighbour velocity grids is performed as in (4.4).

Defining the local velocity grids in such a way ensures that if all the distribution function is contained in the local grids at the first time step, the convective fluxes outside the local grids are zero even if the distribution function is not a Maxwellian.

Let us summarize the algorithm to define the local grid in the space cell i at the beginning of each time step for a 1D case. Let $\mathcal{G}^{n-1}_i$ be the grid used in the previous time step:

1. Compute the maximum and minimum velocity required with respect to the parameter α, the macroscopic velocity and the temperature as in (4.2).
2. Check if the distribution function f^n_i is contained in the new local grid $\mathcal{G}^n_i$ with respect to a tolerance as in (4.5). Otherwise set $\mathcal{G}^n_i = \mathcal{G}^{n-1}_i$.

3. Perform the union with the neighbour local velocity grids with (4.4).
4. Compute the numerical fluxes using the new grid and evolve f_i^{n+1}.

Note that step 2 is not performed at t=0 because the distribution function at t=0 is initialized with the local Maxwellian in each space cell. Thus the criteria on the standard deviation is enough to initialize the local velocity grids with good accuracy.

4.2 The multidimensional space algorithm

The extension to multidimensional space is quite straightforward. The same approach is used along all dimensions independently. By considering the local macroscopic velocity and local temperature, criteria (4.2) is applied to construct the local grid in each space direction with the corresponding component of the macroscopic velocity. The maximum and minimum microscopic velocity ($\hat{\xi}_{\max}=(\hat{\xi}_{u,\max},\hat{\xi}_{v,\max},\hat{\xi}_{w,\max})$ and $\hat{\xi}_{\min}=(\hat{\xi}_{u,\min},\hat{\xi}_{v,\min},\hat{\xi}_{w,\min})$ in 3D) are then computed along each space direction independently. To make sure of including all non zero flux on non-existing velocities (as described in 1D), a union of the neighbour velocity grids has to be performed. In multidimensional cases, the fluxes are calculated with a directional splitting. Each local grid has to include all the local grids involved in the transport step. For example, in 2D, in a space cell (i,j) and for a second-order scheme in space:

$$\xi_{\min}^{i,j}=\min_{k=-2,\cdots,2}\left(\hat{\xi}_{\min}^{i+k,j},\hat{\xi}_{\min}^{i,j+k}\right)\qquad\text{and}\qquad\xi_{\max}^{i,j}=\max_{k=-2,\cdots,2}\left(\hat{\xi}_{\max}^{i+k,j+l},\hat{\xi}_{\max}^{i,j+k}\right).\tag{4.6}$$

As in the 1D case, this kind of approach might not be accurate away from the hydrodynamic regime. Therefore, the same correction of the local velocity grid is applied based on the integral of the distribution function (Eq. (4.5)).

The local grids are then a sub-rectangle of the global grid in 2D and a brick in 3D.

5 Numerical results

In this section numerical results will be presented in 1D and 2D for both hydrodynamic and rarefied regimes. The computations for the 2D test cases are performed in parallel.

5.1 Test case 1: Two interacting blast waves

5.1.1 Hydrodynamic regime

This test case was initially introduced by Woodward and Colella [26]. It is a popular test case that shows the ability of a scheme to correctly capture very strong shock waves in the continuum regime. Initially, the domain [0,1] is decomposed in three sub-domains:

$$\begin{cases}\rho=1, & u=0, & p=1000 & \text{in [0,0.1]} & \text{(region A)}\\ \rho=1, & u=0, & p=0.01 & \text{in [0.1,0.9]} & \text{(region B)}\\ \rho=1, & u=0, & p=100 & \text{in [0.9,1]} & \text{(region C)}\end{cases}$$

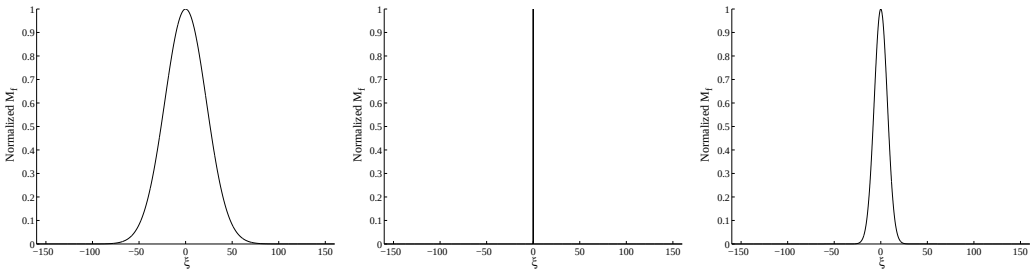

Figure 2: Test Case 1: Normalized Maxwellian distribution functions in region A (left), B (center) and C (right).

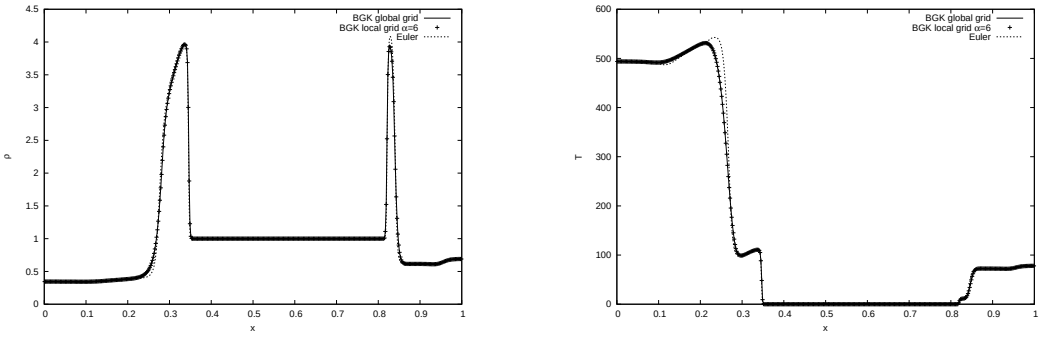

Figure 3: Test case 1, $Kn_\infty = 10^{-5}$: Density (left) and temperature (right) solution with 500 grid points in space.

In each subdomain, the distribution function is initialized as a Maxwellian. Due to the large temperature differences, the support of each Maxwellian is very different as shown on Fig. 2.

At the boundaries of the domain, solid walls reflect the waves. The classic specular reflection is applied. The Knudsen number is set to 10^{-5} and the simulation is stopped at t=0.01. Second-order schemes in space and time are used. The global velocity grid is computed as in (4.1). In this case, the velocity space is $[-160,160]$ and is discretized uniformly with 3200 grid points for the global grid. The solution is shown on Fig. 3 for the temperature and the density. The solid line corresponds to the solution of the BGK equation on the global grid, the dashed line corresponds to the solution of compressible Euler equations while the solution for the BGK equation with the local grid method and $\alpha = 6$ is represented with + signs.

The local velocity grids are defined as in (4.2) and (4.5), and we investigate the performance of the scheme as a function of the parameter α. Note that increasing α means that the local grid boundaries are enlarged. Fig. 4 shows the boundaries of the local grids as a function of space when only criteria (4.2) is used with $\alpha = 6$. Taking into account the second criteria enlarges the local grid around shocks (see Fig. 5). Comparing it with the global grid (in solid lines) highlights that the local grid includes only a fraction of the global grid points.

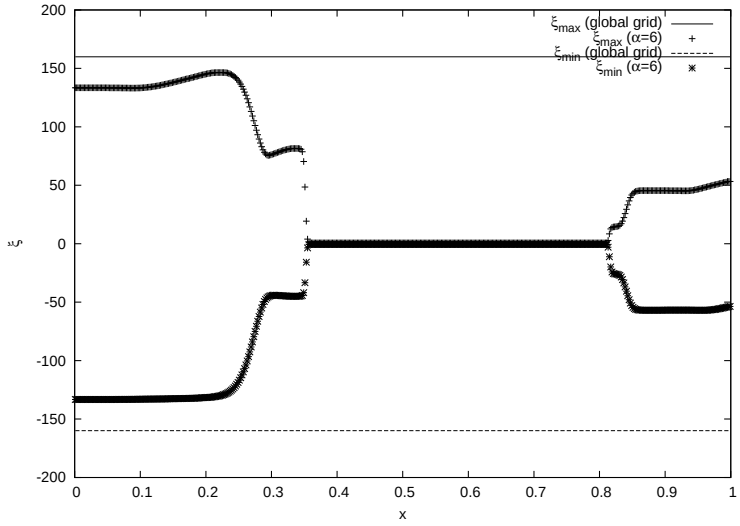

Figure 4: Test case 1, $Kn_\infty = 10^{-5}$: Maximum and minimum velocity in the local grid for $\alpha = 6$ (first criteria).

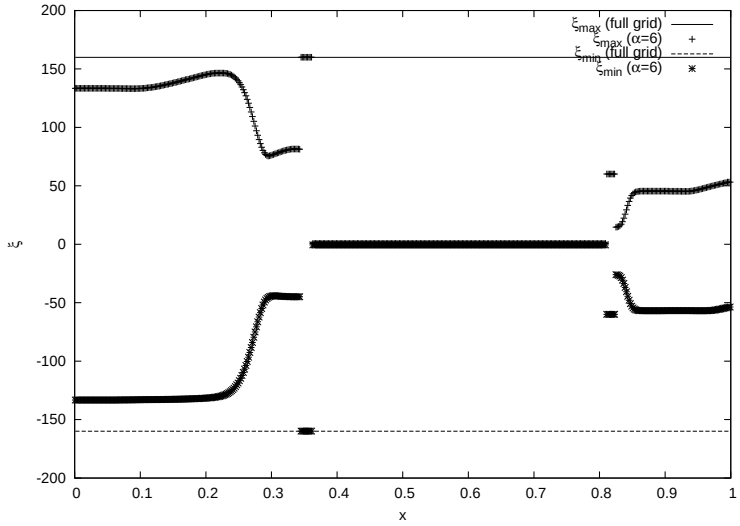

Figure 5: Test case 1, $Kn_\infty = 10^{-5}$: Maximum and minimum velocity in the local grid for $\alpha = 6$ (both criteria).

Figs. 6 and 7 show the normalized errors in L_1, L_2 and L_∞ norm for the density, velocity and energy for several values of α. The behaviour for large values of α is also presented in L_∞norm on Fig. 7. Defining $\underline{\rho} = (\rho, U, E)$, the different norms are normalized as follows:

$$L_{p,normalized}(\underline{\rho}) = \frac{||\underline{\rho}_{\mathcal{G}} - \underline{\rho}_{\mathcal{G}_v}||_{L_p}}{||\underline{\rho}_{\mathcal{G}}||_{L_p}}, \tag{5.1}$$

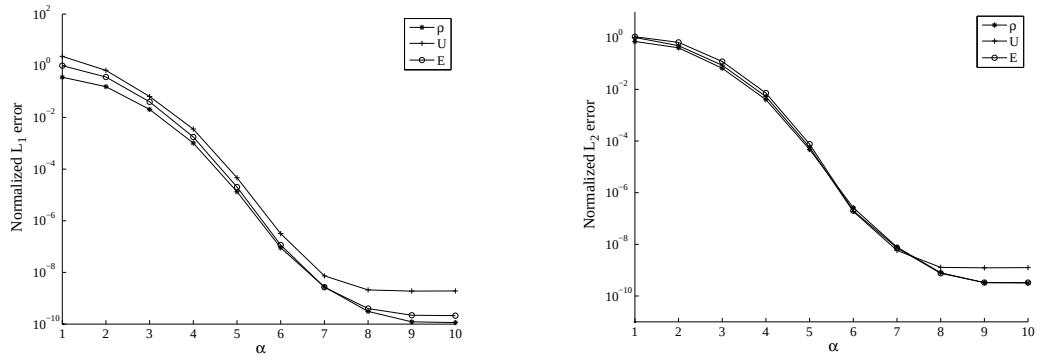

Figure 6: Test case 1, $Kn_\infty = 10^{-5}$: L_1 (left) and L_2 (right) error on density, velocity and energy.

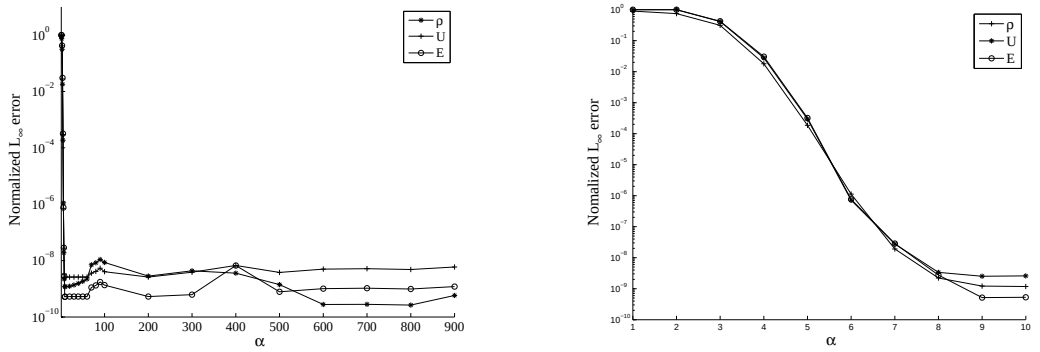

Figure 7: Test case 1, $Kn_\infty = 10^{-5}$: L_∞ error on density, velocity and energy.

where L_p is the standard $L_p$norm, $\underline{\rho}_{\mathcal{G}}$ is the solution for $\underline{\rho}$ on the global velocity grid, $\underline{\rho}_{\mathcal{G}_v}$ is the solution for $\underline{\rho}$ with the local velocity grid method.

It is observed that the error decreases fast and for $\alpha \geq 6$ the error is already lower than 10^{-6} in all norms. It remains stable for large values of α.

One can also compute the conservation error on mass and energy normalized by the initial value of mass and energy respectively (Fig. 8) as a function of α. This error has a similar behaviour and reaches quickly machine precision (which corresponds to the conservation error on the global grid) since the scheme is conservative.

Results on Fig. 9 show the fractions of the computational time and of the total number of velocity grid points used with respect to the global grid calculation as function of α. These results are obtained with a constant time step defined with the maximum velocity of the global grid. Even for $\alpha = 600$ the velocity grid does not coincide with the global grid because of the presence of regions where the temperature is small. Indeed, in such areas the standard deviation of the Maxwellian is small (~ 0.1) and a large value of α is required to get the whole grid ($\alpha \sim 1600$). If the time step is defined at each iteration with the maximum velocity contained in the local grids (but still constant in space), the com-

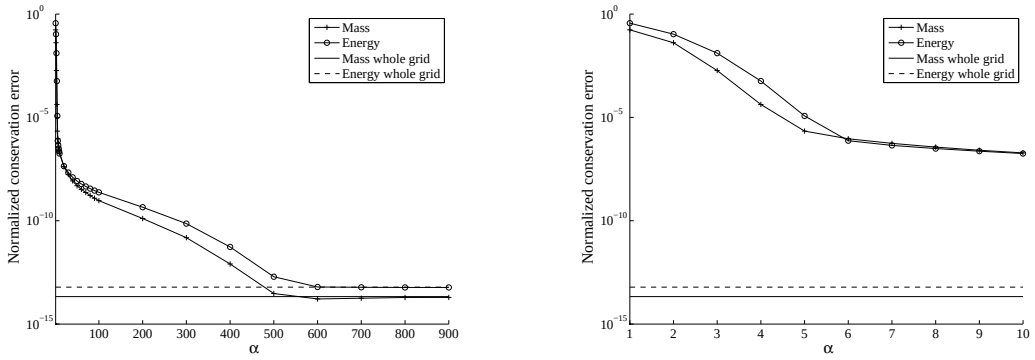

Figure 8: Test case 1, $Kn_\infty = 10^{-5}$: Conservation error.

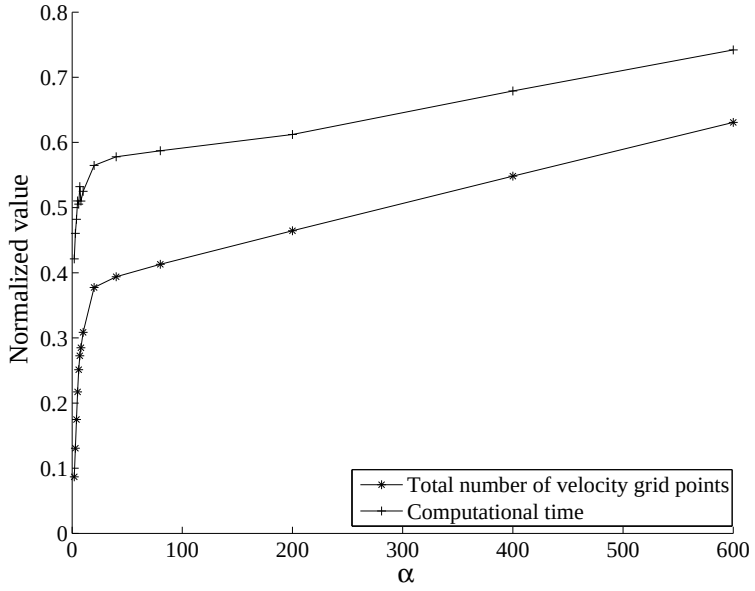

Figure 9: Test case 1, $Kn_\infty = 10^{-5}$: Normalized number of velocity grid points used and computational time with respect to the global grid calculation.

putational time can be significantly reduced when the maximum velocity of the global grid is not contained in any local grid. Fig. 10 shows the comparison of the computational time obtained with a constant time step and a variable time step defined as above.

The largest gain is obtained for small values of α. But even with higher values, there is still a gain. For instance, for $\alpha = 6$, the error is below 10^{-6} with a conservation error below 10^{-5} while the gain in CPU time is around 50% using less than 75% of the global grid.

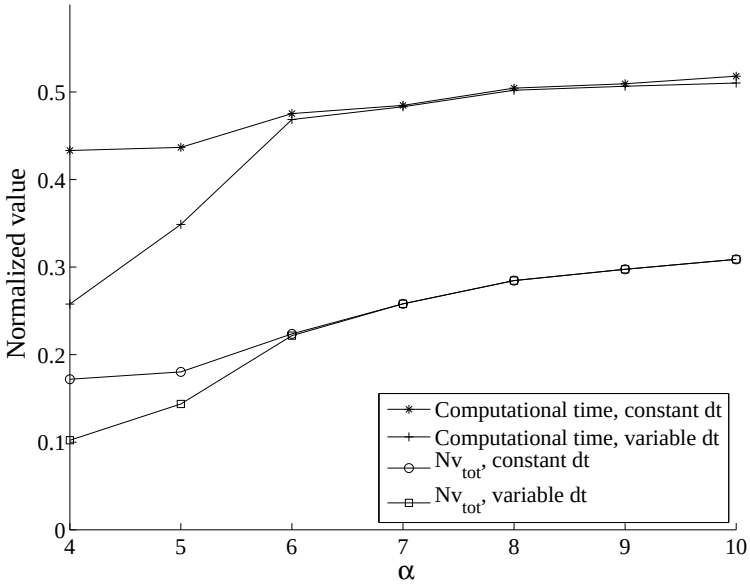

Figure 10: Test case 1, $Kn_\infty = 10^{-5}$: Normalized number of velocity grid points used and computational time with respect to the global grid calculation.

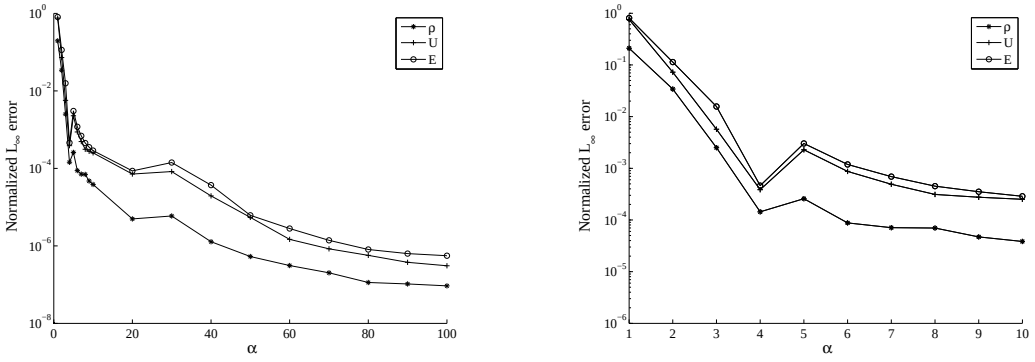

Figure 11: Test case 1, $Kn_\infty = 10^{-2}$: L_∞ error on density, velocity and energy (left) with a zoom on small values of α (right).

5.1.2 Rarefied regime

Here the method is tested on the same test case but with a higher Knudsen number. In particular, we consider $Kn_\infty = 10^{-2}$ as an intermediate regime, and $Kn_\infty = 1$ for the fully kinetic regime.

In these cases, the error of the method is larger with respect to the case at low Knudsen number because the distribution function may be more diffusive than the Maxwellian and thus the conservation error may increase as Kn_∞ grows. Since the shape of the distribution function is not close to a Maxwellian distribution function, criteria (4.5) on the

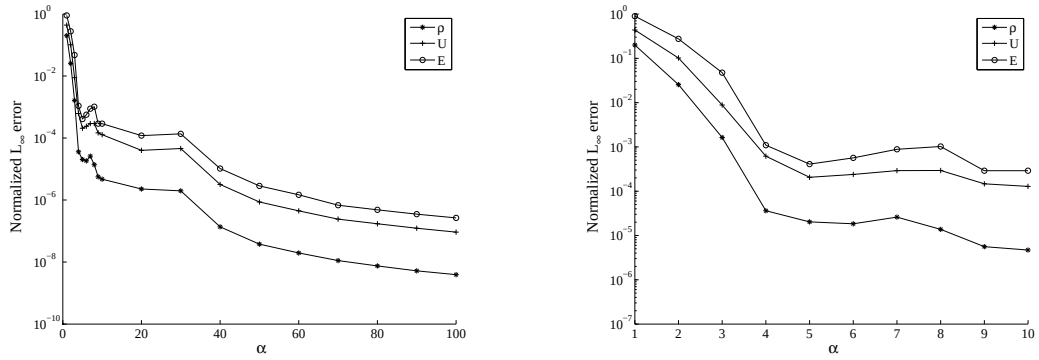

Figure 12: Test case 1, $Kn_\infty = 1$: L_∞ error on density, velocity and energy (left) with a zoom on small values of α (right).

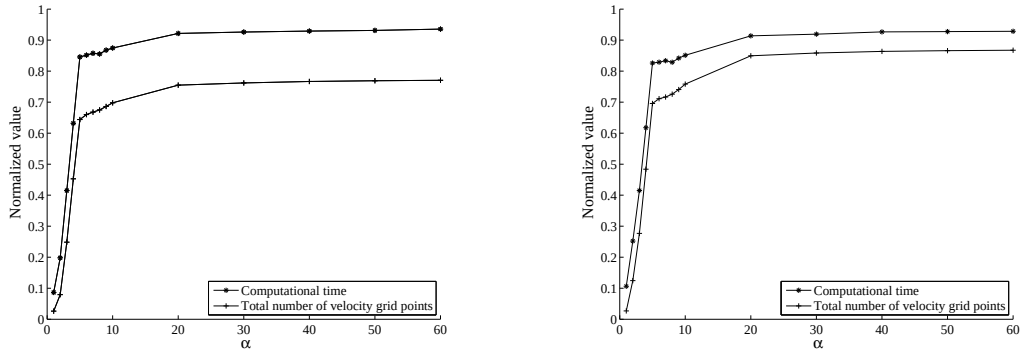

Figure 13: Computational time and total number of velocity grid points used for $Kn_\infty = 10^{-2}$ (left) and $Kn_\infty = 1$ (right).

integral of the distribution function has to be used. Hence, the error still goes towards zero for reasonable values of α. We show the results on the error in L_∞ on Figs. 12. In other norms, the errors behave similarly. These results also show that for α around 5, there is a kink in the error. This occurs when the conservation criteria (4.5) becomes less restrictive than the criteria on α. Typically, for these values of α (around 5) the more restrictive criteria switches from one to another during the calculation. The same behaviour can be observed in the following error curves relative to 2D hydrodynamic or kinetic regime.

The gain in the computational time (see Fig. 13) is much smaller than for $Kn_\infty = 10^{-5}$ because the relaxation time is too high to redistribute the particles as a Maxwellian distribution function after the convective step. Then, after several time steps, the distribution function lies on a large part of the velocity grid in all cells. But even in this case a gain of about 15% can be obtained for a relative error of 10^{-4}, which is obtained at $\alpha = 6$.

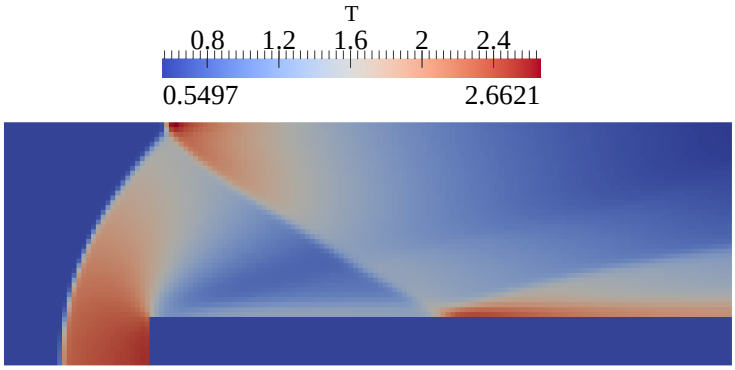

Figure 14: Solution for the temperature ($Kn_\infty = 10^{-5}$).

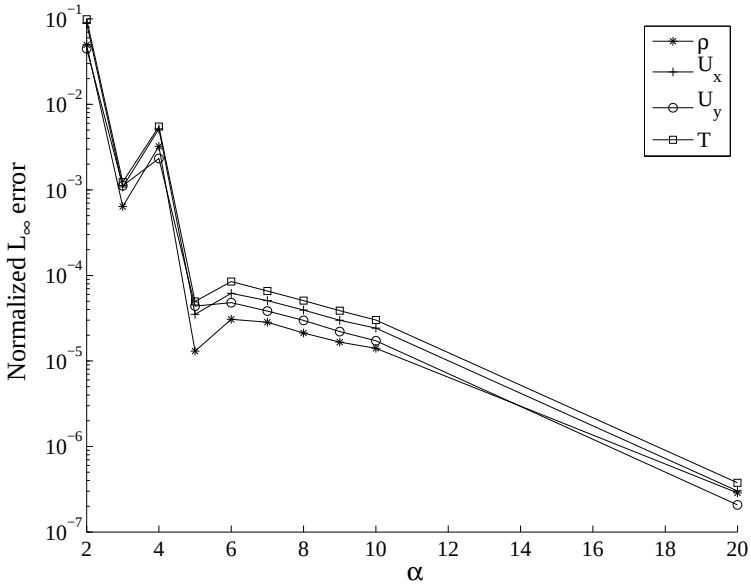

Figure 15: Test case 2, $Kn_\infty = 10^{-5}$: L_∞ error on density, velocity and temperature.

5.2 Test case 2: A Mach 3 wind tunnel with a step

5.2.1 The hydrodynamic regime

Here, the local grid method is tested on a 2D case. A flow at $M = 3$ propagates in a tunnel with a step. A shock is created on the step and then is reflected on the top of the tunnel. Strong shocks propagate and interact with themselves and with the boundaries. This is also a difficult test case for the discrete velocity model because of the step geometry. The temperature profile is shown in Fig. 14.

The domain is $[0,3] \times [0,1]$. The step lies from $x = 0.6$ to $x = 3$ and goes from $y = 0$ to

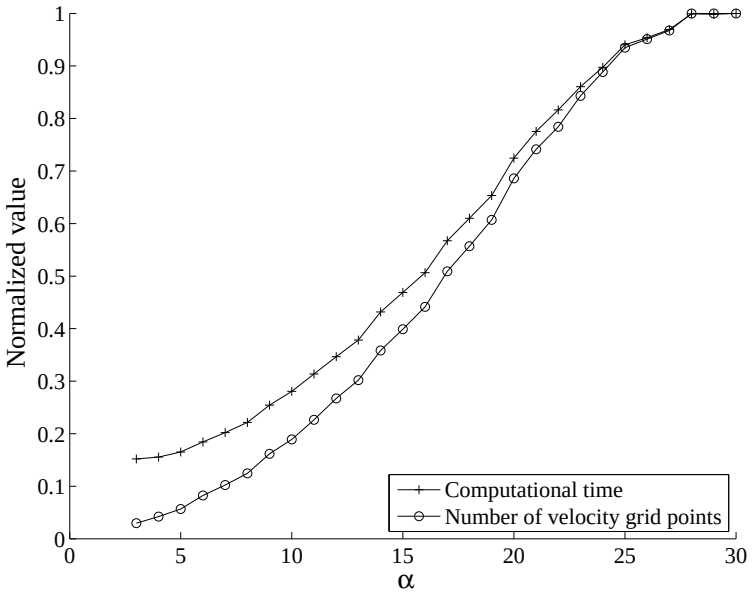

Figure 16: Normalized number of velocity grid point used and computational time with respect to the full grid calculation for one iteration on one processor ($Kn=10^{-5}$).

$y=0.2$. The spatial domain is discretized with 150×50 cells. The Knudsen number is set to 10^{-5}. The global velocity space is $[-20,20]\times[-20,20]$ discretized on a Cartesian grid with 61 grid points in each direction. It is built to satisfy conditions (4.1).

The error is calculated with respect to the BGK solution on the whole velocity grid for different values of α for density, velocity and temperature. The L_∞ norm of this error is plotted on Fig. 15. Other norms behave similarly.

As for the 1D case, the error on macroscopic variables goes quickly towards zero as α grows. For values of α between 5 and 10, the error is lower than 10^{-4} in all norms.

Fig. 16 shows the computational time and the total number of velocity grid points used for one iteration on one processor. It shows a good scalability of the method with respect to the number of velocity grid points. For small values of α (≤5) the computational time becomes less sensitive to the number of velocity grid points used because the execution time of some parts of the code is independent or almost independent of the number of velocity grid points. For example, the time required to compute the Maxwellian distribution function is mostly due to the solution of the non-linear system in (2.11) and the number of space cells.

The computational time to get the solution at t=1.2 has also been compared for constant and variable time steps, as well as the total number of velocity grid points used for the whole calculation (see Fig. 17). It shows that the method is efficient because the computational time scales well with the number of velocity grid points. With $\alpha=10$, less than 30% of the whole grid is used and the computational time is reduced by more than

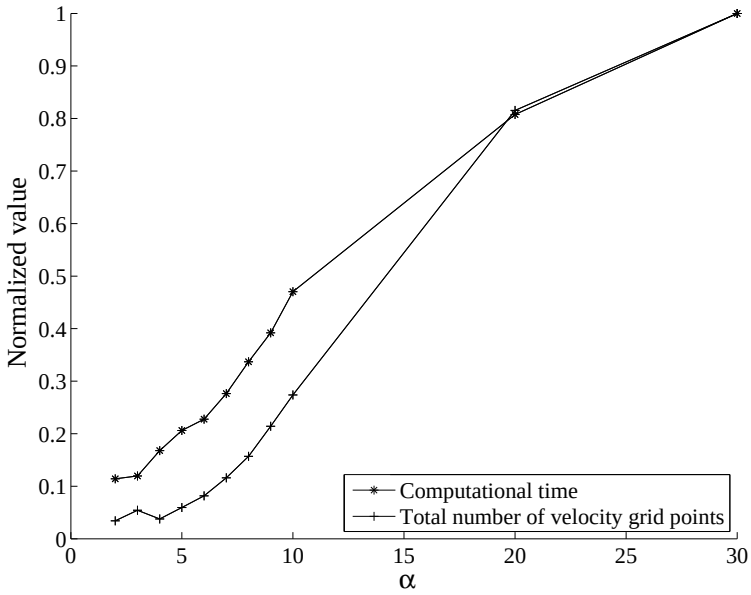

Figure 17: Normalized number of velocity grid point used and computational time with respect to the full grid calculation ($Kn_\infty = 10^{-5}$).

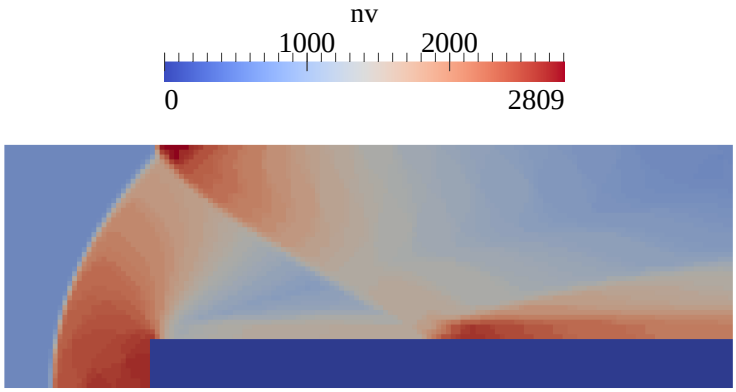

Figure 18: Number of velocity grid points used in each cell at t=1.2 for $\alpha = 10$ ($Kn_\infty = 10^{-5}$).

50% with a good accuracy (the L_∞norm is lower than 10^{-4} for all variables).

We also display the number of velocity grid points used in each cell at the final time (Fig. 18). More detail is needed in the regions where there are sharp gradients, and where the gas is far from equilibrium. The width of the grid is determined by the temperature and thus it follows the temperature profile (compare Figs. 14 and 18).

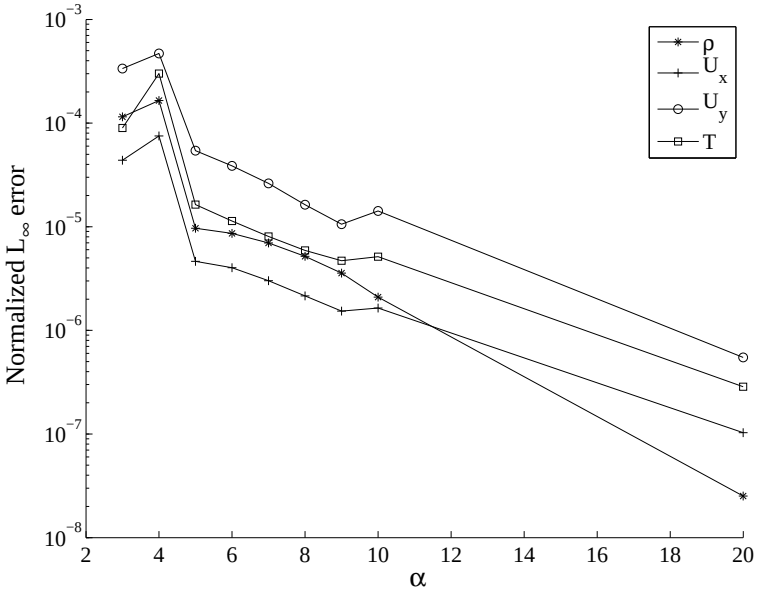

Figure 19: Test case 2, $Kn_\infty = 10^{-2}$: L_∞ error on density, velocity and temperature.

5.2.2 The rarefied regime

The same test case is presented but with a Knudsen number set to 10^{-2}. Here, the distribution function will no longer be close to a Maxwellian. The criteria (4.5) on the integral of the distribution function is more restrictive than the criteria involving α especially for small values of α. The results on the error in L_∞norm are shown in Fig. 19 while the computational gains appear in Fig. 20. For $\alpha = 10$, the error is of the order of 10^{-5} and the CPU time is again reduced by more than 50%.

5.3 Test case 3: the blunt body at Mach 10

Another 2D test case is presented to show the efficiency of the method on a Cartesian grid with immersed boundaries. A cylinder is immersed in a flow at Mach 10. A shock develops on the body and very strong differences of temperature and velocity will appear in the field. We consider $Kn = 10^{-5}$ and therefore we impose the Euler-AP boundary condition on the cylinder [5].

This test requires a very large and fine velocity grid to capture all the phenomena. The domain is $[-2,0]\times[0,4]$ and is discretized with $50\times$ 100 cells in space. The global velocity space is discretized with 101×101 cells in $[-50,50]\times[-50,50]$.

Fig. 21 shows the solution for the temperature and the number of velocity grid points in each cell at steady state for $\alpha = 6$. As expected and observed in the case of the wind tunnel, the grid is enlarged around the shock and contains more velocity grid points in high temperature areas.

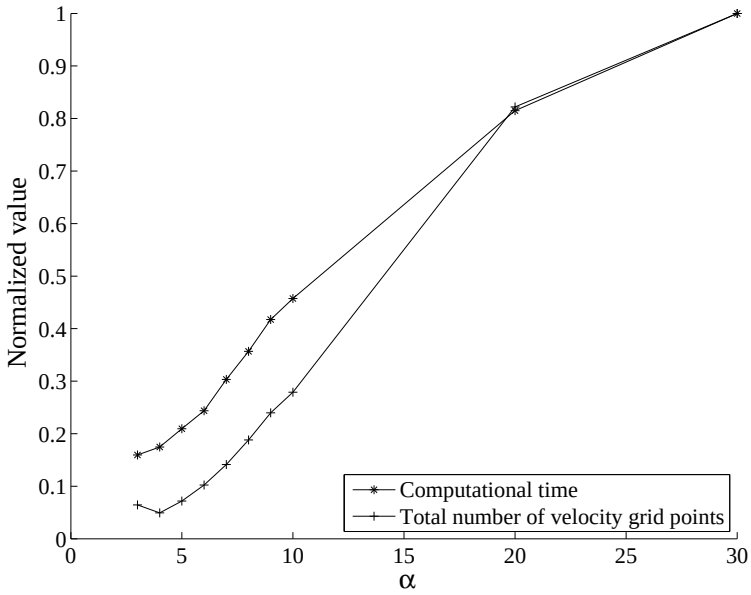

Figure 20: Test case 2, $Kn_\infty = 10^{-2}$: Normalized number of velocity grid points used and computational time with respect to the full grid calculation.

The error obtained with the local grids is calculated with respect to the solution on the global velocity grid (Fig. 22).

Taking an error of the order of 10^{-6} (see Figs. 22) corresponds to $\alpha = 6$ and gives a gain about 70% of the computational time (see Fig. 23).

In Fig. 23, one can note that for the smallest value of α ($\alpha = 2$), the number of grid points used is actually larger than for $\alpha = 3$. This is due to the criteria on the integral (4.5) which is more restrictive and imposes to enlarge the grid to include all points where f cannot be disregarded. This occurs also in the case of the wind tunnel (see Fig. 20).

6 Conclusion

We presented an approach to solve problems governed by the BGK equation where high temperature or velocity gradients are present in the field. This method is based on the construction of local velocity grids. In each space cell, the local grid is determined by the local macroscopic velocity and temperature. These quantities a-priori determine the shape of the corresponding Maxwellian distribution function. Thus, at each time step, the local grid is defined ensuring that all regions where the local Maxwellian is not negligible are included. Next a criteria on conservation is added to the algorithm to avoid mass and energy loss in the kinetic regime where the distribution function is far from the local Maxwellian.

The local grids are characterized by having always the same grid spacing and each

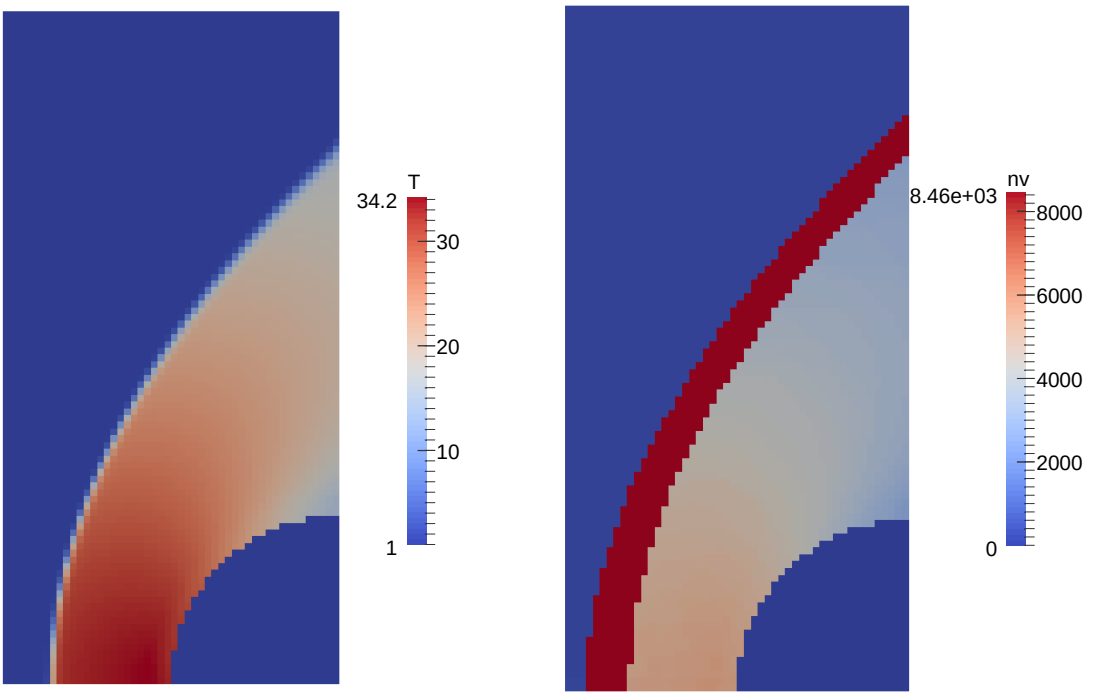

Figure 21: Test case 3, $Kn_\infty = 10^{-5}$: Blunt body solution for the temperature (left) and number of velocity grid points in each cell at steady state (right).

local grid is designed to be a subset of an ideal global velocity grid. This ensures that no interpolation is required between neighbouring space cells, avoiding the overhead typical of non-uniform grids. Although the scheme is simple, the computational gain is relevant, especially close to the hydrodynamic regime. These gains are observed also in parallel computations. In future work, this method will be extended to the ES-BGK model to deal with the correct Prandtl number and to 3D test cases where the gain expected is even larger.

Acknowledgments

This study has been carried out with financial support from the French State, managed by the French National Research Agency (ANR) in the frame of the "Investments for the future" Programme IdEx Bordeaux (ANR-10-IDEX-03-02), Cluster of excellence CPU.

This work was partially supported by "National Group for Scientific Computation (GNCS-INDAM).

Experiments presented in this paper were carried out using the PLAFRIM experimental testbed, being developed under the Inria PlaFRIM development action with support

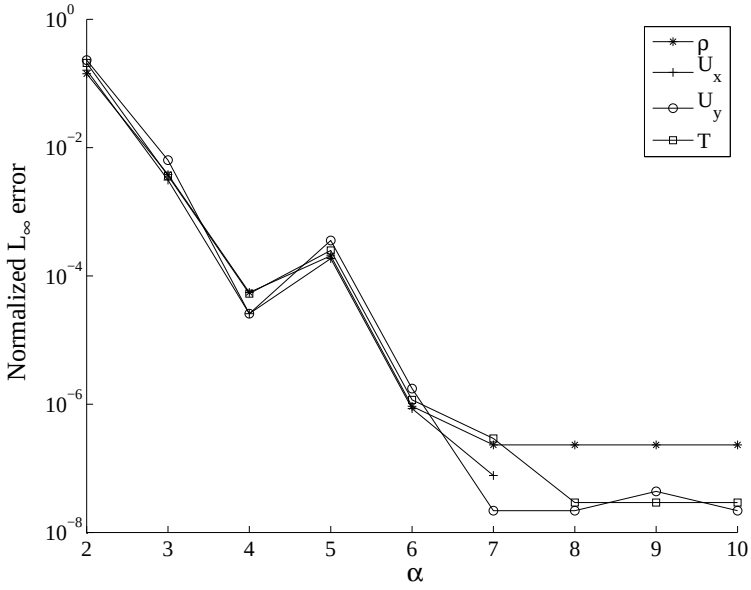

Figure 22: Test case 3, $Kn_\infty = 10^{-5}$: L_∞ error on density, velocity and temperature.

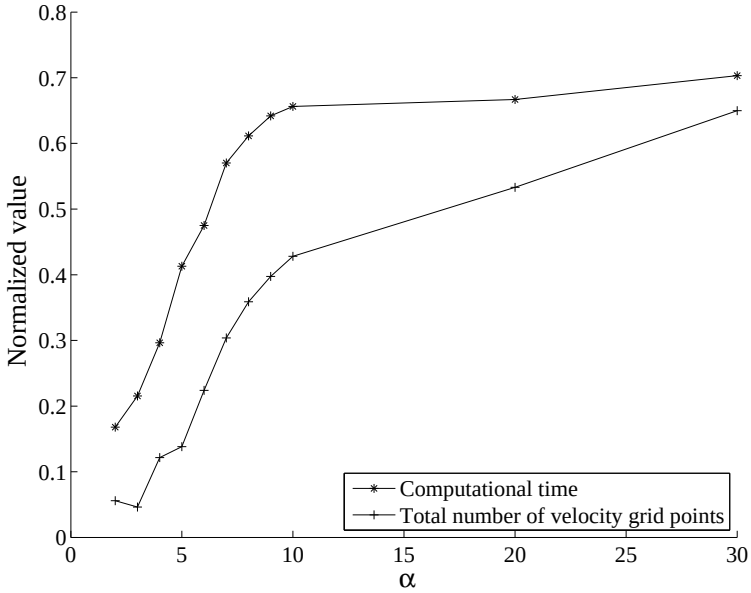

Figure 23: Test case 3, $Kn_\infty = 10^{-5}$: Normalized number of velocity grid points used and computational time with respect to the full grid calculation.

from LABRI and IMB and other entities: Conseil Régional d'Aquitaine, FeDER, Université de Bordeaux and CNRS (see https://plafrim.bordeaux.inria.fr/).

References

[1] Alaia, A., and Puppo, G. A hybrid method for hydrodynamic-kinetic flow - Part II - Coupling of hydrodynamic and kinetic models. Journal of Computational Physics 231, 16 (June 2012), 5217–5242.

[2] Andries, P., Bourgat, J.-F., Le Tallec, P., and Perthame, B. Numerical comparison between the Boltzmann and ES-BGK models for rarefied gases. Computer Methods in Applied Mechanics and Engineering 191, 31 (May 2002), 3369–3390.

[3] Andries, P., Le Tallec, P., Perlat, J.-P., and Perthame, B. The Gaussian-BGK model of Boltzmann equation with small Prandtl number. European Journal of Mechanics. B. Fluids 19, 6 (2000), 813–830.

[4] Baranger, C., Claudel, J., Hérouard, N., and Mieussens, L. AIP Conference Proceedings. In 28th International Symposium on Rarefied Gas Dynamics 2012, AIP, pp. 389–396.

[5] Bernard, F., Iollo, A., and Puppo, G. Accurate Asymptotic Preserving Boundary Conditions for Kinetic Equations on Cartesian Grids. Rapport de recherche RR-8471, INRIA, Feb. 2014.

[6] Bhatnagar, P. L., Gross, E. P., and Krook, M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one- component systems. Physics Review 94, (May 1954), 511–525.

[7] Bird, G. A. Molecular Gas Dynamics and the Direct Simulation of Gas Flows. Oxford Engineering Science Series. Clarendon Press, 1994.

[8] Buet, C. A discrete-velocity scheme for the Boltzmann operator of rarefied gas dynamics. Transport Theory and Statistical Physics 25, 1 (1996), 33–60.

[9] Cercignani, C. The Boltzmann Equation and Its Applications. Springer-Verlag GmbH, 1988.

[10] Chapman, S., and Cowling, T. The Mathematical Theory of Non-uniform Gases: An Account of the Kinetic Theory of Viscosity, Thermal Conduction and Diffusion in Gases. Cambridge Mathematical Library. Cambridge University Press, 1970.

[11] Chen, S., Xu, K., Lee, C., and Cai, Q. A unified gas kinetic scheme with moving mesh and velocity space adaptation. Journal of Computational Physics 231, 20 (Aug. 2012), 6643–6664.

[12] Chu, C. K. Kinetic-theoretic description of the formation of a shock wave. Physics of Fluids 8, (1965), 12–22.

[13] Coron, F., and Perthame, B. Numerical passage from kinetic to fluid equations. SIAM Journal on Numerical Analysis 28, (1991), 26–42.

[14] Degond, P., Pareschi, L., and Russo, G. Modeling and Computational Methods for Kinetic Equations. Modeling and Simulation in Science, Engineering and Technology. Springer, 2004.

[15] Dimarco, G., and Pareschi, L. Numerical methods for kinetic equations. Acta Numerica 23, (2014), 369–520.

[16] Filbet, F., and Jin, S. A class of asymptotic-preserving schemes for kinetic equations and related problems with stiff sources. Journal of Computational Physics 229, 20 (Oct. 2010), 7625–7648.

[17] Filbet, F., and Jin, S. An Asymptotic Preserving Scheme for the ES-BGK Model of the Boltzmann Equation. Journal of Scientific Computing 46, 2 (June 2010), 204–224.

[18] Jin, S. Efficient asymptotic-preserving (AP) schemes for some multiscale kinetic equations. SIAM Journal on Scientific Computing 21, 2 (1999), 441–454.

[19] Kennedy, C. A., and Carpenter, M. H. Additive Runge-Kutta schemes for convection-diffusion-reaction equations. Applied Numerical Mathematics 44, 1-2 (Jan. 2003), 139–181.

[20] Mieussens, L. Discrete-Velocity Models and Numerical Schemes for the Boltzmann-BGK

Equation in Plane and Axisymmetric Geometries. Journal of Computational Physics 162, 2 (Aug. 2000), 429–466.

[21] Pareschi, L., and Russo, G. An introduction to the numerical analysis of the Boltzmann equation. Rivista di Matematica della Università di Parma. Serie 7 4**, (2005), 145–250.

[22] Pareschi, L., and Russo, G. Implicit–Explicit Runge–Kutta Schemes and Applications to Hyperbolic Systems with Relaxation. Journal of Scientific Computing 25, 1 (Oct. 2005), 129–155.

[23] Pieraccini, S., and Puppo, G. Implicit–Explicit schemes for BGK kinetic equations. Journal of Scientific Computing 32, 1 (2007), 1–28.

[24] Pieraccini, S., and Puppo, G. Microscopically Implicit-Macroscopically Explicit schemes for the BGK equation. Journal of Computational Physics 231, (2012), 299–327.

[25] Tcheremissine, F. Conservative evaluation of Boltzmann collision integral in discrete ordinates approximation. Computers & Mathematics with Applications 35, 1 (1998), 215–221.

[26] Woodward, P., and Colella, P. The numerical simulation of two-dimensional fluid flow with strong shocks. Journal of Computational Physics (ISSN 0021-9991) 54, (Apr. 1984), 115–173.

[27] Xu, K., and Huang, J.-C. A unified gas-kinetic scheme for continuum and rarefied flows. Journal of Computational Physics 229, 20 (Oct. 2010), 7747–7764.