A PARALLEL WEIGHTED PARTICLE ALGORITHM FOR SOLVING THE BOLTZMANN (B.G.K.) EQUATION ON THE CONNECTION MACHINE

Didier Issautier [1], Frédéric Poupaud[1],Loula Fezoui[2] and Jean-Pierre Cioni[3]

Abstract

We present a parallel algorithm based on a deterministic particle method for solving the Boltzmann equation (B.G.K. model). Two test cases are performed : the Sod shock tube and a rarefied gas flow on a flat plate. We present performance results on the Connection Machine CM200 and compare them to those obtained on a CRAY-YMP.

RESOLUTION DE L'EQUATION DE BOLTZMANN (B.G.K.) PAR UNE METHODE PARTICULAIRE DETERMINISTE SUR LA CONNECTION MACHINE

Résumé

Nous proposons un algorithme parallèle basé sur une methode particulaire déterministe pour résoudre l'équation de Boltzmann (modèle B.G.K.). Nous présentons des résultats numériques pour deux cas tests : le tube à choc de Sod et l'écoulement d'un gaz raréfié sur une plaque plane. Nous présentons également des résultats de performance obtenus sur la Connection Machine CM200 et les comparons à ceux obtenus sur un CRAY-YMP.

[1]Laboratoire J.-A. Dieudonné, U.R.A. 168 du CNRS Université de Nice Sophia-Antipolis Parc Valrose, BP 71, 06108 NICE Cédex 02

[2]CERMICS-INRIA, Sophia-Antipolis

[3]INRIA, Sophia-Antipolis

Introduction

The **Boltzmann** equation is the basic model in the kinetic theory of gases. Boltzmann type equation are also involved in modeling electron transport in solids and plasmas, neutron transport in nuclear reactors, and radiative transfer in planetary and stellar atmospheres.
We are concerned here in the Bhatnagar-Gross-Krook (B.G.K.) model which is a relaxation model describing the evolution of a gas through a kinetic theory. This model does not contain relevant features of the classical Boltzmann equation but despite its simplicity, it contains most of the basic properties of fluid dynamics such as the conservation of mass, momentum and energy [1].
Therefore, B.G.K. equations are widely used for testing numerical methods for kinetic equations. It explains the growing interest in numerical approaches to solve B.G.K. problems.
Nevertheless, approximated solutions need to deal with a very large number of particles; pratically millions of particles have to be used in realistic computations. Well it is clear that whatever scheme one may use, the computation time on serial machines will grow at best linearly with the number of particles. To be more exact, the computation time per particle is expected to be constant in the best cases.
The advent of parallel systems raised hopes in going beyond this limit. Indeed, one could expect that the CPU time per particle will decrease if the number of particles grow together with the number of processors. However, interactions between particles may translate into complex and numerous interprocessor communications in parallel implementations.
Aiming at parallel implementations, one has to pay a great attention in the choice of the numerical method. There are mainly two approaches in numerical kinetic: the probabilistic approach which is the most used and the deterministic one which is in rise nowadays.
The numerical method to be used here belongs to the latter class and was introduced by S. Mas-Gallic [4]. Our aim here was to propose a parallel algorithm issued from the weighted particle method and its implementation on a massively parallel computer. A Connection Machine (CM200 model) was chosen for the numerical experiments to be presented here.
This paper consists of four sections. In section 1, we recall the B.G.K model. Section 2 is devoted to the presentation of the particle method. In section 3, we present briefly the Connection Machine and describe the parallel algorithm. Finally we present some numerical solutions and performance results obtained on the Connection Machine. Comparisons with the CRAY-YMP results are also reported.

1 Mathematical Model

1.1 The Boltzmann Equation

The evolution of the particle distribution in the six-dimensionnal space (x,ξ) is modellised by the **Boltzmann** equation [1]. In the case of a monoatomic gas

where particles interact through a finite range conservative force, the evolution of the distribution f is governed by the following Boltzmann equation :

$$\frac{\partial f}{\partial t} + \xi.\nabla_x f = \int_{IR^3 \times S^2} (f' f'_* - f f_*) q(\xi_* - \xi, n)\, d\,\xi_*\ d\,n \tag{1}$$

where $f = f(t, x, \xi)$, t is the time, x is the position, ξ is the velocity and q is the diffusion cross section. The prime superscript refers to before-collision states and the star subscript denotes the dummy variable :

$$f_* = f(t, x, \xi_*),\ f'_* = f(t, x, \xi'_*),\ f' = f(t, x, \xi').$$

The before-collision velocities are related here to the after-collision ones through the following elastic shock law:

$$\xi' = \xi + n(\xi_* - \xi, n) \text{ and } \xi'_* = \xi_* - n(\xi_* - \xi, n)$$

The right-hand side of the Boltzmann equation (1) usually denoted by $\mathbf{Q}(f, f)$ is called the **collision operator**.

1.2 The B.G.K. Model

Because of the complicated structure of the collision operator, simpler collision terms have been suggested.

In fact, it was observed that a large amount of details of the two-body interaction does not affect significantly the values of many measures in physical experiments. Thus, it was expected that the fine structure of the collision operator $\mathbf{Q}(f, f)$ could be replaced by a simpler one $\mathbf{J}(f)$ which retains only the qualitative and average properties of $\mathbf{Q}(f, f)$. Consequently, $\mathbf{J}(f)$ must satisfy the following relations :

$$\int_{IR^3} \phi_k \mathbf{J}(f) d\,\xi = 0 \tag{2}$$

$$\int_{IR^3} \log f \mathbf{J}(f) d\,\xi \leq 0 \tag{3}$$

where $\phi_{k=0,\ldots,4} = (1, \xi_{l=1,2,3}, \mid \xi \mid^2)$ are the elementary invariants by collision. One gets the equality in (3) if, and only if, the density f is a **Maxwellian** [1].
Equation (2) expresses the conservation of mass, momentum and energy and the Boltzmann inequality (3) shows the tendency of the density to a Maxwellian distribution. The simplest way to take (3) into account is to assume that the average effect of collisions changes the density f by an amount proportional to its departure from a Maxwellian M.
We introduce the Knudsen number $Kn = l/d$ where l denotes the mean free path and d is a characteristic length scale. The Knudsen number is related to the relaxation time τ by: $Kn = \tau\bar{\xi}/d$, $\bar{\xi}$ being a typical molecular velocity.
By setting

$$\mathbf{J}(f) = \frac{1}{\tau}\,(\,M\ -\ f\,) \tag{4}$$

we get the **Bhatnagar, Gross** and **Krook** (**B.G.K.**) model [2].
The above Maxwellian M depends on the disposable scalar parameters $(\rho, \rho U, E)$ which represent respectively the density, the momentum and the energy . According to equations (2) and (4), we have :

$$\int_{IR^3} \phi_k M(f) d\xi = \int_{IR^3} \phi_k f \, d\xi$$

M is usually called the local Maxwellian and may be written :

$$M(\rho, \rho U, E) = \frac{\rho}{(2\pi T)^{\frac{3}{2}}} \exp(-\frac{(\xi - U)^2}{2T}) \tag{5}$$

where :

$$\begin{pmatrix} \rho \\ \rho U \\ E \end{pmatrix} = \int_{IR^3} \begin{pmatrix} 1 \\ \xi \\ \frac{|\ \xi\ |^2}{2} \end{pmatrix} f \, d\xi \tag{6}$$

The temperature T is deduced from the total energy of monoatomic molecules :

$$E = \rho \frac{|\ U\ |^2}{2} + \frac{3}{2} \rho T. \tag{7}$$

Diatomic case

The above model describes the evolution of a monoatomic gas. In the case of diatomic molecules, one has to take into account the rotational energy in the definition of the internal energy (see [3],[8]). This leads to slight modifications concerning the collision operator $\mathbf{J}(f)$. More precisely, the local Maxwellian M is replaced by an other equilibrium function also denoted M and defined by :

$$M(\rho, \rho U, E, I) = \frac{\rho}{(2\pi T)^{\frac{3}{2}+\frac{1}{\delta}}} \exp(-\frac{(\xi - U)^2}{2T} - \frac{I^\delta}{T}) \tag{8}$$

I represents the internal energy term associated to rotation mode and $\delta = \dfrac{2(\gamma - 1)}{2 - 3(\gamma - 1)}$ where γ is the ratio of specific heats. One may observe that the two definitions of M are almost the same if we set γ equals to $5/3$ (monoatomic case) in (8).

The moments of f are now defined by :

$$\begin{pmatrix} \rho \\ \rho U \\ E \end{pmatrix} = \int_{IR^3 \times IR^+} \begin{pmatrix} 1 \\ \xi \\ \frac{|\ \xi\ |^2}{2} + I^\delta \end{pmatrix} f \, d\xi \, dI \tag{9}$$

We use this extension to perform diatomic gas simulations to be presented later. We refer the reader to [7] for a mathematical analysis of this model.

It is well known that Euler system of equations can be derived from the conservation laws for the moments of the distribution f in the limit process $\tau \to 0$. This property will be used in part 3.1 to validate the numerical method to be presented in the next section. For the sake of simplicity, the method will be described in the case of a monoatomic gas.

2 Numerical Method

For solving numerically the kinetic equations, one generally use probabilistic approaches and more precisely the Monte-Carlo method (see [6]). However, parallelising the Monte-Carlo method on fine grained machines is not straightforward (see [12] for example).
More recently, deterministic approaches for particle methods have been suggested. The aim of this paper is to prove that determimistic particle methods are well adapted to massive parallelism.

We propose here a parallel algorithm based on the weighted particle method introduced by S. Mas-Gallic [4]. In this method, particles move according to a collisionless dynamic. The collision part is then considered by allocating a weight to each particle. Integrals involved in (6) are evaluated by a quadrature formula where the particles themselves play the role of quadrature points. We shall describe briefly the method and refer to [4],[5] for more details.

2.1 Description of the method

Given an initial distribution f^0 on a set of points $(x_i^0, \xi_i^0)_{i \in I \subset \mathbf{Z}}$, we search $f_i(t)$, $x_i(t)$ and $\xi_i(t)$ such that the measure

$$f_h = \sum_{i \in I} w_i f_i(t) \delta(x - x_i(t)) \otimes \delta(\xi - \xi_i(t)) \tag{10}$$

is some approximation of the solution f and w_i are the associated positive weights. Using (10), we introduce a quadrature formula associated to the points (x_i, ξ_i) and the weights w_i defined as follows :

$$\int \int g(t,x,\xi)\, d\,x\, d\,\xi \;\approx\; \sum_{i \in I} w_i g(t, x_i(t), \xi_i(t)) \tag{11}$$

where g is a continuous function in (x, ξ) variables.
The main idea of deterministic particle methods is to point out that f_h is a good approximation of f in the space of measures if (11) is a good quadrature formula and if f_i is an approximated value of $f(t, x_i(t), \xi_i(t))$, see [15].

Convective terms

$x_i(t)$ and $\xi_i(t)$ are solutions of the following differential system :

$$\begin{cases} \dfrac{d\,x_i}{d\,t}(t) = \xi_i \quad , \; x_i(0) = x_i^0 \\ \dfrac{d\,\xi_i}{d\,t}(t) = 0 \quad\;\; , \; \xi_i(0) = \xi_i^0 \end{cases} \tag{12}$$

Since ξ_i are constant w.r.t. time, solutions are given by :

$$x_i(t) = x_i^0 + \xi_i^0 t \;\text{ and }\; \xi_i(t) = \xi_i^0$$

Note: The jacobian determinant of the change of variables $(x_i^0, \xi_i^0) \to (x_i(t), \xi_i(t))$ takes the constant value 1 . Consequently, if (w_i, x_i^0, ξ_i^0) is a good quadrature formula, the same is true for $(w_i, x_i(t), \xi_i(t))$.

Collision terms

The main difficulty here consists in computing the macroscopic quantities ρ, ρU and E which define the Maxwellian (5). Let us note that integrations are done in the velocity space. The quadrature formula (11) which is set in the phase space will be used to define approximate values of the integrals (6). This is done by introducing a continuous function ζ which transforms the integral in ξ-space into an integral in the phase space. The function ζ has the following properties:

$$\left\{ \begin{array}{l} \displaystyle\int_{I\!R^3} \zeta(x)\, d\,x = 1 \\ \displaystyle\int_{I\!R^3} x^l \zeta(x)\, d\,x = 0 \qquad 1 \leq l \leq k-1 \ ; \ \ k \geq 2 \\ \displaystyle\int_{I\!R^3} \mid x^k \mid \mid \zeta(x) \mid d\,x < +\infty \end{array} \right.$$

Let us set : $\zeta_\epsilon(x) = \dfrac{1}{\epsilon^3}\zeta(\dfrac{x}{\epsilon})$ for $\epsilon > 0$. Approximate discrete values of ρ, ρU and E, at time t are then defined by :

$$\left(\begin{array}{c} \rho_h(t,x) \\ (\rho U)_h(t,x) \\ E_h(t,x) \end{array} \right) \ = \ \sum_{j \in I} w_j \left(\begin{array}{c} 1 \\ \xi_j \\ \mid \xi_j \mid^2 \end{array} \right) f_j\, \zeta_\epsilon\,(x - x_j(t)) \tag{13}$$

In our study, we use the cut-off functions ζ_ϵ with compact support in $[-\epsilon, \epsilon]$. Consequently, the only particles which occur in the computation of ρ_h, $(\rho U)_h$ and E_h are those which verify $\mid x - x_j \mid < \epsilon$.
Using (5), one can define the associated local Maxwellian at each time t by :
$M_h(t,x,\xi) = M(\rho_h, \rho U_h, E_h)(t,x,\xi)$ and we put $M_i(t) = M_h(t, x_i(t), \xi_i(t))$

2.2 Time integration

We want a time integration scheme which remains stable for both large and small values of the Knudsen number. However, we restrict ourselves to explicit schemes for they are in general more adapted to unsteady calculations and are also nicely parallelisable. Under these constraints, we choose a scheme based on the time integration of the B.G.K. solution :

$$f(t+\Delta t, x, \xi) = f(t, x - \Delta t \xi, \xi) \exp(-\frac{\Delta t}{\tau}) + \frac{1}{\tau}\int_0^{\Delta t} M(t+s, x - s\xi, \xi) \exp(-\frac{\Delta t - s}{\tau}) ds \tag{14}$$

One may easily verify that f given by the above formula is a solution of the B.G.K. equation by deriving along characteristics : $x(t) = x(0) + \xi t$.
In the following we note $M(t+s) = M(t+s, x - s\xi, \xi)$.

If we write the implicit expression (14) for $t = t^n = n\Delta t$ where Δt is the time step, we get:

$$f^{n+1} = \exp(-\frac{\Delta t}{\tau}) f^n + \frac{1}{\tau}\int_0^{\Delta t} M(t+s)\exp(-\frac{\Delta t - s}{\tau})ds$$

Assuming the Maxwellian M to be constant w.r.t. time on $[t^n, t^{n+1}]$, the above integral may be evaluated as:

$$\int_0^{\Delta t} M(t_n+s)\exp(-\frac{\Delta t - s}{\tau})ds \simeq M^n \int_0^{\Delta t} \exp(-\frac{\Delta t - s}{\tau})ds$$

Finally, the resulting explicit scheme may be written :

$$\begin{array}{rcl} f_i^0 & = & f_0(x_i^0, \xi_i^0) \\ f_i^{n+1} & = & \exp(-\frac{\Delta t}{\tau}) f_i^n + (1 - \exp(-\frac{\Delta t}{\tau}))M_i^n \end{array} \qquad (15)$$

where the subscript i refers to the phase-space discretisation described previously. It is easy to check that for a fixed τ the above scheme is a first-order time accurate approximation of (14) and is unconditionally stable in L^1. The latter assertion is based on the fact that the Maxwellian is a L^1 isometric application.

3 Numerical experiments

In order to validate the method, we performed two test cases :

1. The Sod shock tube: this classical one-dimensional test case was selected here for we can compare the numerical results to the exact solution in the limit $\tau \to 0$.

2. Leading edge problem: this academic test case was chosen for it is a true two-dimensional case and thus allows us to validate the numerical treatment of the boundary conditions.

3.1 Sod shock tube

In terms of fluid dynamics, the Sod shock tube problem [13] corresponds to a Riemann problem in $\mathbb{R}$ with the following initial data :

$$\left\{ \begin{array}{llll} \rho_L^0 = 1 & U_{x,L}^0 = 0 & U_{y,L}^0 = 0 & P_L^0 = 1 \\ \rho_R^0 = \frac{1}{8} & U_{x,R}^0 = 0 & U_{y,R}^0 = 0 & P_R^0 = \frac{1}{10} \end{array} \right.$$

The subscripts L and R refer to left and right states respectively. The initial distribution function f^0 is given by: $f^0 = M(\rho^0, (\rho U)^0, E^0)$ where M is given by (5). The classical slip condition turns here into specular reflection :

$$f(t, x, \xi, I) = f(t, x, -\xi, I) \quad x = 0,\ 1.$$

In this experiment we used the described scheme with $\tau = 10^{-8}$. The computed density, pressure,temperature and velocity are plotted on the figures 1 to 4 at time $t = 0.16$ and compared to the exact Euler solution (dotted lines). The results show a slight amount of numerical dissipation particularly visible on the rarefaction and the contact discontinuity waves. This dissipation is usually attributed to the presence of large velocities which induce some diffusion [8]. Despite this diffusion, one can claim that the results are quite good.

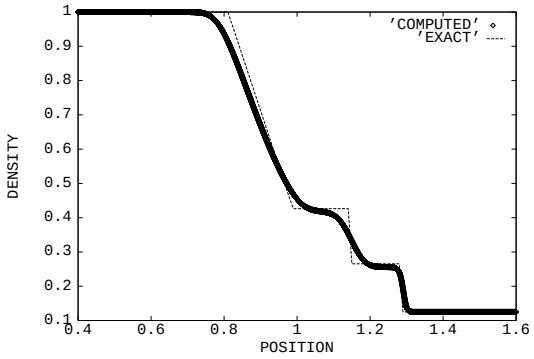

Fig. 1. Density at time t=0.16

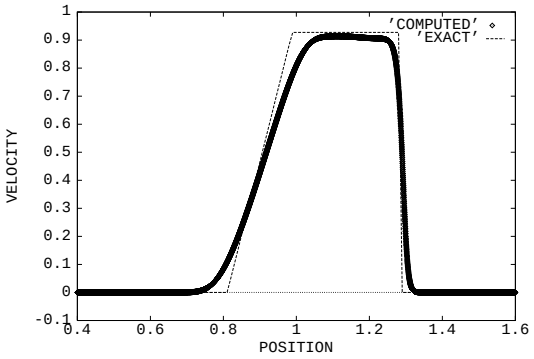

Fig. 2. Velocity at time t=0.16

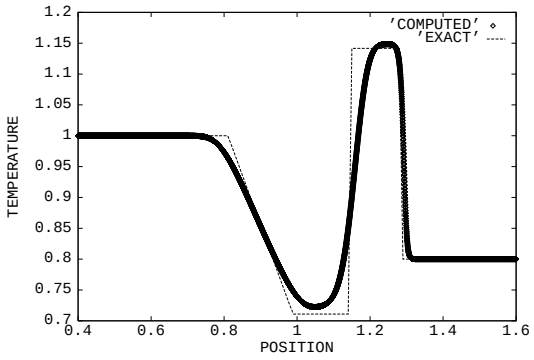

Fig. 3. Temperature at time t=0.16

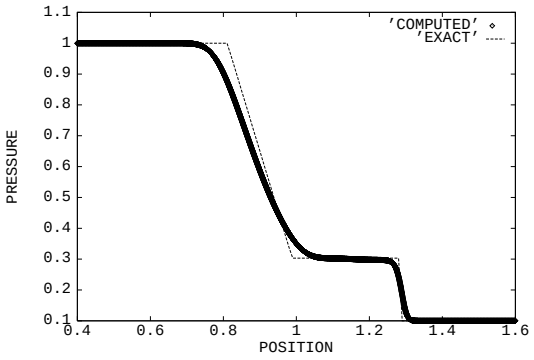

Fig. 4. Pressure at time t=0.16

3.2 Leading edge problem

We use here the physical data given in [12]. We consider a flow of 1200 m/s of air at a number density of 7.10^{20} and a temperature of 300 K past a plate of length 0.21 m. The flow data are taken to be the values of the free stream flow and are denoted as U_∞, ρ_∞ and T_∞ respectively. The surface temperature is of 300 K. Numerical solutions of this problem, get by a Monte-Carlo method, can be found in [11]. It was assumed that there is a complete thermal accomodation on the plate. The domain of computation Ω is shown on the figure 5.

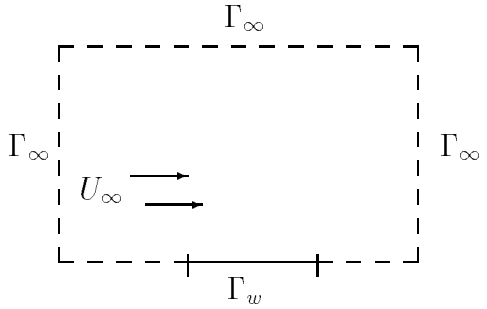

Fig. 5: Computation Domain Ω

Boundary conditions

- On Γ_∞, the gas is assumed to be in thermal equilibrium:

$$f(t,x,\xi) = M(\rho_\infty, (\rho U)_\infty, E_\infty) \quad \xi.n < 0 \quad x \in \Gamma_\infty$$

where T_∞, is evaluated from the free stream flow values given above through the energy definition (7).

- On Γ_w, the boundary conditions describe the interactions of the gas molecules with the solid boundary. Let us note $R(\xi^{'} \rightarrow \xi, x, t)$ the probability density that a molecule stricks the solid boundary with velocity between $\xi^{'}$ and $\xi^{'} + d\xi^{'}$ at the point x and time t and re-emerges at pratically the same point with velocity between ξ and $\xi + d\xi$. If R is known, a boundary condition beared on f can be deduced. Because of the complexity of the phenomena taking place on the surface, physical models defining a corresponding kernel $R(\xi^{'} \rightarrow \xi)$ were constructed. The first attempt to solve such a problem is due to **Maxwell** who suggested for R (see [1]) :

$$R(\xi^{'} \rightarrow \xi, x, t) = (1-\alpha)\delta(\xi^{'} + 2n(\xi.n)) + \alpha f_0(\xi, I) \mid \xi.n \mid \quad (\xi.n > 0 \quad \xi^{'}.n < 0)$$

where α is the fraction of evaporated molecules and f_0 is defined by :

$$f_0 = \frac{1}{T_0 (2\pi T_0)^{\frac{1}{2}+\frac{1}{\delta}}} \exp(-\frac{\xi^2}{2T_0} - \frac{I^\delta}{T_0})$$

T_0 is the temperature of the solid boundary; α is also called the **accomodation coefficient**. For $\alpha = 0$, we obtain a pure specular reflection. Complete thermal accomodation corresponds to $\alpha = 1$ and leads to the following boundary condition:

$$f(t,x,\xi,I) = f_0(\xi,I) \int_{(\xi^{'}.n<0)\times IR^+} f(t,x,\xi^{'},J) d\xi^{'} dJ \quad x \in \Gamma_w = \partial\Omega \ ; \ \xi.n > 0$$

As in the previous case, we assume that at time $t = t_0$ the gas is in thermal equilibrium, the initial distribution function is a Maxwellian which depends on the characteristics of the upstream flow. The Knudsen number is about 10^{-2} in this calculation.

We present on the figures 6 and 7 the pressure and shear stress distributions along the wedge. They are respectively defined by : $C_p = \dfrac{2\,p}{\rho_\infty U_\infty^2}$ and $C_f = \dfrac{2\,\tau_w}{\rho_\infty U_\infty^2}$.

We compare our results to those obtained by Bird [11] who used the DSMC (Direct Simulation Monte Carlo) method. We can observe that the difference in amplitude does not exceed 10% except at the tips of the plate where the effect of the upstream flow is particularly perceptible. It is not easy to conclude which result is the most accurate although the kinetic model we used is unlikely to give the best results in this case. One may note however that the results are in good agreement.

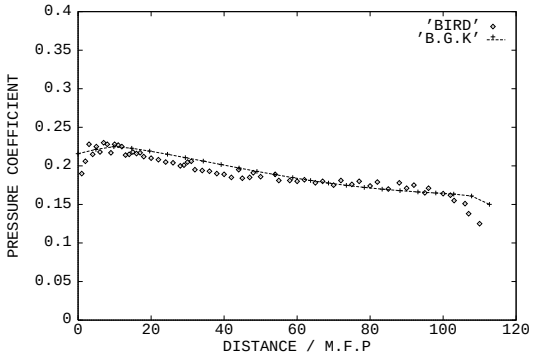

Fig. 6: Pressure coefficient

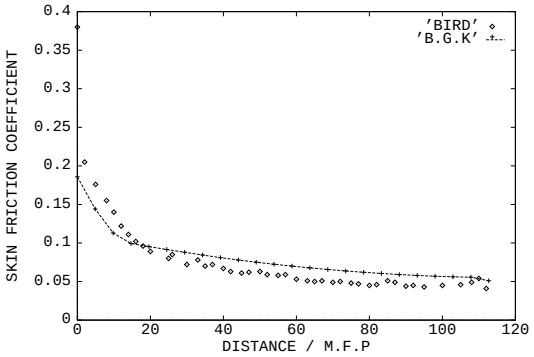

Fig. 7: Skin friction coefficient

Incident case

We consider again the same problem with now an incidence angle of 20^o and a surface temperature of $678K$. The pressure and skin friction coefficient distributions are plotted on the figures 8 and 9 and are compared with Bird's results. Although this test case is appreciably more difficult than the non incident one, we obtain comparable results with no more extra work on the code. The numerical parameters ($\Delta t, \epsilon$, number of particles...) were unchanged and we got the solution with the same time cost as in the previous case.

We believe that these experiments demonstrate the reliability and the robustness of the scheme used.

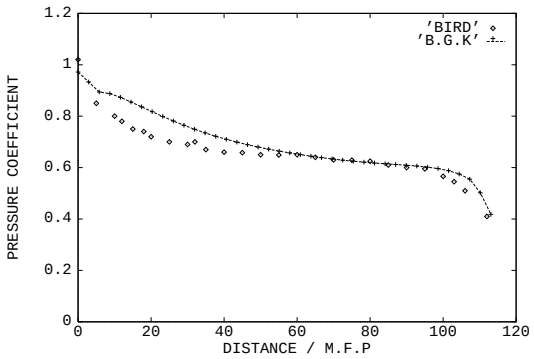

Fig. 8: Pressure coefficient

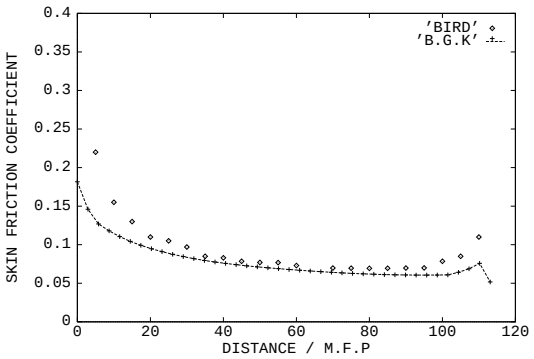

Fig. 9: Skin friction coefficient

4 Parallel implementation

4.1 The Connection Machine System

We first recall the main features of the Connection Machine CM200. The CM200 is a **SIMD** (Single Instruction Multiple Data) distributed memory machine containing up to $64\,K$ (K=1024) bit-serial processors having their own memory. These processors are arranged 16 to a chip in a hypercube topology of dimension 12 in the $64\,K$ configuration. Each pair of chips may be supported by a 32 or 64-bit floating point accelerator (FPA) which increases floating point calculations rate by a factor of 20.

The Connection Machine is provided with two communication models:

- The NEWS (North-East-West-South) network is a structured mechanism which optimizes grid communications by exploiting the four nearest neighbour connections.

- The general mechanism (ROUTER) allows parallel accesses among the distributed memory so that any processor can access the memory of any other processor. This mechanism uses the hypercube topology. NEWS network is the fastest but requires regular patterns.

The Connection Machine system allows to use more processing nodes than it is physically available through a splitting of the distributed memory. Each physical processing node simulates some number of virtual processing nodes and the ratio between the virtual ones to the physical ones is called the **V**irtual **P**rocessor **R**atio (VPR). Connection Machine system software is provided with three high-level languages, among which is the *CM* Fortran: it is based on Fortran 77 but also uses the array extensions in the draft Fortran 90 standard to express data parallel operations. We refer to [14] for more details on the system.

4.2 Description of the algorithm

Parallelising well an algorithm on a massively parallel computer consists first in finding a "good" distribution of the computation data among thousands of processors.
An appropriate data distribution is a one that :

1. Minimises inter-processor communications.

2. Involves no redundant operations and needs no extra storage.

Such a distribution does not always exist and we more usually have to find a compromise between communications, operations and storage.
For the B.G.K. model, there are mainly two ways in distributing data :

1. One may associate one particle to each processor. The convective part of the algorithm becomes perfectly parallelisable i.e. no communication occur while computing new positions of particles. However, the collision part which is the most costly in floating point operations would involve a large amount of communications. This would result in an inefficient algorithm on the Connection Machine and we rejected such a distribution.

2. Another way to distribute data is then to privilege the collision part. This is done in a such way that no communication may occur now during the computation of the collision operator. However, as it will be explained in the following, the privilege given to the collisions will be paid by an extra storage which may result in a limitation of the number of numerical particles.

Since we choose a cut-off function with compact support of size ϵ, we can introduce a fictitious discretisation of the physical space into cells of the same size (ϵ). Finally, we associate to each processor the content of a cell and the data of the neighbouring cells. Thus, the collision operator is computed without performing any communication.
Communications will occur in the convective part of the algorithm. Indeed, at each time step of length Δt we have to compute the new position of each particle:

$$x_i^{n+1} \;=\; x_i^n \;+\; \Delta t\, \xi_i^0$$

Thus, a particle may leave its initial cell to a new one which may be located in an arbitrary physical processor. In this case, the general mechanism of communication (ROUTER) is required and the communication cost can be very important.
In order to minimise time communication, we choose the time step Δt such as the particles can only reach the neighbouring cells. Thus, we communicate only with the neighbouring processors and then we can use the NEWS network. The resulting algorithm may be summarised as follows:

For each time step do:

1. Convect particles
2. Communicate data to the neighbours
3. Integrate the boundary conditions
4. Compute the collision operator
5. Update the solutions

5 Performance results

We present now the performance results and compare them to those get on a CRAY–YMP. Let us precise that the code has been implemented in **CM-Fortran** and that no optimisation was performed. The results have been get on $8\,K$ and $16\,K$ CM200 configurations.
We have reported on Table 1 the CPU Times and Gflops (i.e. 10^9 floating point operations per second). The number of spatial cells used equals the number of virtual processors. The number of particles per cell was constant, taken as 99 leading to a total number of particles of 99 times the number of spatial cells. We note a decrease of 30% in performance with double precision (DP) in comparison with single precision (SP) (fig. 10). Nevertheless, the results of Table 1 show a very high level of parallelism since the part of the communication time is less than 5% of the

total CPU time. One can also observe a good scaling by comparing for example the results obtained on the $8\,K$ physical configuration at a VPR of 2 with the corresponding results on the $16\,K$ configuration at a VPR of 1. This allows us to expect a performance of 8 Gflops on a $64\,K$ configuration in single precision and of 5.5 Gflops in double precision. However, the main observation one can make from these results is that the CPU time per particle is decreasing with the number of physical processors on the CM200 while it remains constant on the CRAY, as it was expected.

Machine Type	Spatial Cells	CPU Time per iter per particle (μs)	CPU Time per iter (s)	% Comm.	GFlops
CM $8\,K$	8192	56	45.6	4.1	1.01
CM $8\,K$	16384	54	88.7	3.6	1.04
CM $16\,K$	16384	28	45.5	4.1	2.0

Table 1: CM200 results

Comparisons in terms of CPU time and performance between the $8\,K$ CM200 results and the CRAY-YMP ones are plotted on the figures 10 to 12. The results show that a $8\,K$ CM200 is about 2.5 faster than the CRAY-YMP one-processor in single precision and about 1.6 in double precision. Indeed, the latter observation makes sense in terms of CPU time since the performance in flops depends too much on the configuration used (cf. Table 1). More precisely, the algorithms being slightly different on the two machines, we decided to consider mainly the CPU time needed to obtain the physical solution. Let us note that the number of iterations is the same in both cases (CRAY and CM200) allowing us to consider the cost per iteration as a good measure of performance. However, the figure 12 shows that the method reaches a high level of parallelism on the CM200 (the performance increases with the number of cells) together with a good level of vectorisation on the CRAY (150 Mflops).

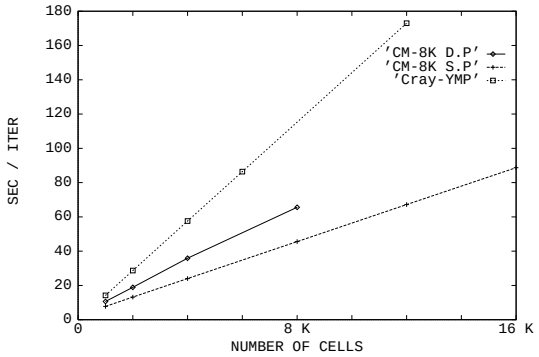

Fig. 10: CPU time / iteration

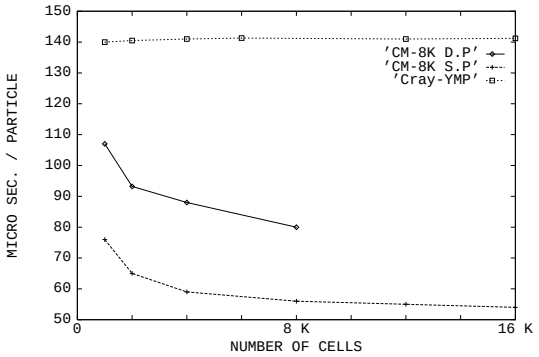

Fig. 11: CPU time / particle

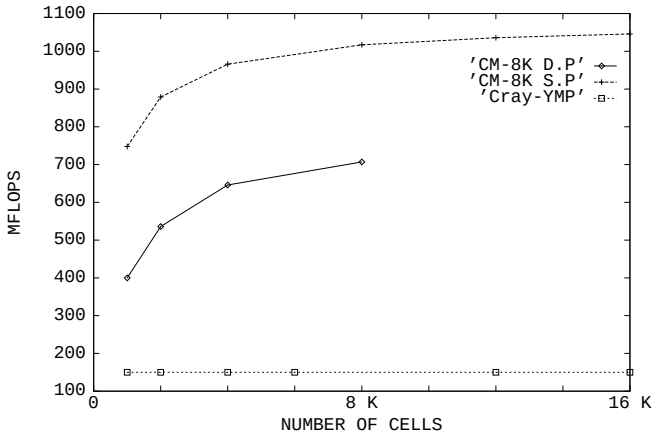

Fig. 12 Performance in MFlops

6 Conclusion

We have reported here on our efforts in kinetic numerical simulations using the B.G.K. model in 2-D space dimension. The numerical approach is based on a deterministic particle method and on a non classical time integration scheme. Numerical solutions of the Sod shock tube and of a flat plate were presented and compared to the exact solution in the first case and to Monte Carlo simulations in the second one. We mainly observed that the B.G.K. numerical solutions compare well with the full Boltzmann simulations. The plate experiments where the Knudsen number is about 10^{-2} demonstrate that the B.G.K. model used remains reliable even for rarefied gas flows.

A particular attention was given to the implementation on a massively parallel machine : the Connection Machine CM200. Performance results show that the $16\,K$ CM200 is about five times faster (in CPU time) than the CRAY-YMP one-processor. The ratio reduces to 3.5 in favour of the $16\,K$ CM200 in double precision. These excellent results are, for a large part, a consequence of the fact that collision phase is performed with no communications. This was achieved via the storage of the neighbouring cells. Such a choice is not restrictive for 2-D simulations since one can use about two million of particles. However, this could result in more severe limits for 3-D simulations.

The high level of parallelism achieved for the presented method allows us to suggest an extension to the full Boltzmann equation. Besides, implementation of the algorithm on a MIMD parallel machine is in progress.

References

[1] C. CERCIGNANI, *The Boltzmann Equation and Its Applications*, Springer Verlag, New-York, Berlin, (1988).

[2] P.L. BHATNAGAR, E.P. GROSS and M. KROOK, *A model for Collision Processes in Gases*, Phys.Rev.,**94**, (1954) p.511.

[3] P.C. PHILIPPI and R. BRUN, *Kinetic modeling of polyatomic Gas Mixture*, Physica **105** A (1981), p.147.

[4] S. MAS–GALLIC, *A deterministic particle method for the linearized Boltzmann Equation*, Trans.Theory Stat.Phys., **16**, (1987) pp.885-887.

[5] S. MAS–GALLIC and F. POUPAUD, *Approximation of the Transport Equation by a weighted particle method*, Trans.Theory Stat.Phys., **17**, (1988) pp.311-345.

[6] C. JACOBINI and L. REGGIANI, Rev.Mod.Phys., **55**, (1983) p.645.

[7] B. PERTHAME, *Global Existence to the B.G.K. model of Boltzmann Equation*, J.Diff.Eq, **82**, (1989) pp.191-205.

[8] B. PERTHAME, *Boltzmann type schemes for gas dynamics and the entropy property* SIAM J. Numer. Anal., **27** (1990) pp. 1405-1421.

[9] C. BARDOS, F. GOLSE, D. LEVERMORE, *Fluid Dynamic Limits of Kinetic Equations : Formal Derivations*, J. of Stat. Phys., **63**, (1991) pp.323-344.

[10] L.N. LONG, M. KAMON and J.MYCZKOWSKI, *A Massively Parallel Algorithm to solve the Boltzmann (B.G.K.) Equation*, AIAA Paper, 92–0563.

[11] G. A. BIRD, *Shock wave–boundary layer interactions in rarefied gas flows*, AIAA Paper, 91–1312.

[12] F. CORON,P. HOMSI, *Utilisation du Parallélisme Massif en calcul d'écoulements raréfiés*, Contrat DRET, 90/587.

[13] G.A. SOD, *A survey of several finite difference methods for systems of non linear hyperbolic conservation laws* J. Comp. Phys., **27**, (1977).

[14] *Technical summary*, Thinking Machines Corporation, Version 6.0 (1990).

[15] P. A. RAVIART, *An analysis of particle methods in Numerical Methods in Fluids Dynamics*, Lect. Notes in Math. **1127**, F. Bezzi ed., Springer Verlag, Berlin, (1985).