TA sl.

FORSCHUNG - AUSBILDUNG - WEITERBILDUNG

Bericht Nr. 83

BOLTZMANN SIMULATIONS WITH AXISYMMETRIC GEOMETRY

J. Struckmeier, K. Steiner

UNIVERSITÄT KAISERSLAUTERN
Fachbereich Mathematik
Postfach 3049

W-6750 Kaiserslautern

November 1992

Boltzmann simulations with axisymmetric geometry

J. Struckmeier, K. Steiner
AG Technomathematik
University of Kaiserslautern
Germany

Abstract

The paper presents theoretical and numerical investigations on simulation methods for the Boltzmann equation with axisymmetric geometry. The main task is to reduce the computational effort by taking advantage of the symmetry in the solution of the Boltzmann equation.
The reduction automatically leads to the concept of weighting functions for the radial space coordinate and therefore to a modified Boltzmann equation. Consequently the classical simulation methods have to be modified according to the new equation.
The numerical results shown in this paper — rarefied gas flows around a body with axisymmetric geometry — were done in the framework of the european space project HERMES.

1 Introduction

In the following paper we will consider the Boltzmann equation of the form

$$\frac{\partial f}{\partial t}(t,\vec{x},\vec{v}) + \vec{v}\cdot\nabla_x f(t,\vec{x},\vec{v}) = J(f,f)(t,\vec{x},\vec{v}) \tag{1}$$

with

$$\begin{aligned} J(f,f) &:= \int_{R^3}\int_{S^2_+} \sigma(|\vec{v}-\vec{w}|,\eta)\{f(t,\vec{x},\vec{v}')f(t,\vec{x},\vec{w}') \\ &\qquad -f(t,\vec{x},\vec{v})f(t,\vec{x},\vec{w})\}d\omega(\eta)d\vec{w} \\ \vec{v}' &= \vec{v}-\vec{\eta}\cdot(\vec{v},\vec{\eta}) \\ \vec{w}' &= \vec{w}+\vec{\eta}\cdot(\vec{w},\vec{\eta}) \\ \vec{\eta} &\in S^2_+ \end{aligned}$$

on a given spatial domain $\Omega \subset I\!R^3$.
The left hand side of equation (1) represents the free transport of the gas particles, the operator $J(f,f)$ the influence of binary interactions of gas particles during the free transport. The function $\sigma(|\vec{v}-\vec{w}|,\eta)$ corresponds to the differential cross section of the binary collisions.
The aim of the paper is to elaborate a simulation method in the case of an axisymmetric flow problem. We assume that the set $I\!R^3 \setminus \Omega \subset I\!R^3$ is compact and the boundary $\partial\Omega$ is axisymmetric with respect to the cartesian coordinate x. This means if the boundary $\partial\Omega$ is given by

$$\partial\Omega = \{(x,y,z) \in I\!R^3 \mid F_\Omega(x,y,z) = 0\}$$

then by introducing cylindrical coordinates (x',r,ϕ) according to the transformation

$$\begin{aligned} x &= x' \\ y &= r\cdot\cos\phi \\ z &= r\cdot\sin\phi \end{aligned}$$

the function F'_Ω is independent of the variable ϕ, i.e.

$$F'_\Omega = F'_\Omega(x,r)$$

If the Boltzmann equation is supplied with initial condition

$$f(t=0,\vec{x},\vec{v}) = f_0(\vec{x},\vec{v}) \tag{2}$$

and boundary condition

$$|\vec{v}\cdot\vec{n}|f(t,\vec{x},\vec{v}) = \int_{\vec{v}'\cdot\vec{n}<0} R(\vec{v}'\to\vec{v};t,\vec{x})|\vec{v}'\cdot\vec{n}|f(t,\vec{x},\vec{v}')d\vec{v}' \tag{3}$$

$$\vec{v}\cdot\vec{n} > 0, \vec{x} \in \partial\Omega \text{ and } \vec{n} \text{ the inner normal at } \vec{x} \in \partial\Omega$$

then under the assumption that the initial condition f_0 as well as the scattering kernel $R(\vec{v} \rightarrow \vec{v}; t, \vec{x})$ is independent of the angle ϕ in a cylindrical system, the solution f of (1) together with (2) and (3) will be independent of the angular variable ϕ, i.e. f will be invariant under the group of rotations around the cylinder axis.
In the context of numerical simulation methods for the Boltzmann equation, as given in [9], it is very useful to consider this property to reduce the computational effort for three dimensional axisymmetric flow problems.
The reduction makes it possible to calculate very small Knudsen number flows within a reasonable computational effort (see for example [5]).
Unfortunately it is not straightforward how to use the symmetry in the solution of the Boltzmann equation in numerical simulation methods. A way to implement an axisymmetric flow problem on a computer was first given by Bird ([4]), however without theoretical investigations. In this approach the space coordinates are reduced to the cylinder coordinate x and the radial distance r. The velocity vector remains unchanged in the cartesian system. Using cartesian velocity components makes it possible to use the collision routines of the standard simulation codes, which are widely elaborated also with regard to real gas effects.
The construction of a simulation method, which is based on a complete transformation, i.e. two dimensional velocity vectors, is much more complicated. The main problem is the transformation of the collision operator to axisymmetric geometry (see for example [12]).
In the following we give a detailled description of the modifications of the standard simulation codes, using axisymmetric space coordinates, but cartesian velocity components.
It is therefore necessary to introduce a weighting function $R(r)$ with respect to the radial direction r and to consider the Boltzmann equation for the new function

$$g(t, x, r, \phi, \vec{v}) = R(r) \cdot f(t, x, r, \phi, \vec{v}).$$

In the modified Boltzmann equation appears a new term on the left hand side of the equation, which influences the free transport of the particles.
The paper is organized as follows :
In the following section we describe briefly the general concept of particle methods for the Boltzmann equation as given in [9]. In part 3 of the paper we consider the modified Boltzmann equation for cylindrical coordinates and the modifications which are necessary in order to use a particle method for this equation. Finally we present some typical numerical results obtained with the presented method.

2 Numerical simulation methods

Numerical methods for the Boltzmann equation are first of all particle methods. Other numerical concepts like finite difference or finite element schemes play only a minor role in the computation of rarefied gas flows. The reason for this is the high dimensional phase space of the distribution function f as well as the highly nonlinear collision term on the right hand side of the equation.
The most popular numerical method for the Boltzmann equation is the Direct Simulation Monte Carlo Method introduced by Bird ([4]). Theoretical investigations on the connection of this method to the Boltzmann equation were given by Wagner ([15]).
Based on the work of Nanbu ([11]), Babovsky ([1], [2] and [3]) introduced a general concept to construct particle methods directly from the Boltzmann equation. This general concept, also called Finite Pointset Method, is further developed in several papers from authors at the University of Kaiserslautern ([9],[6], [7],[8] and [14]).
The idea of the finite pointset method (at the same time the general concept of a particle method) is to use a set of discrete points in the phase space $\Omega \times I\!R^3$, which approximates the function $f(t, \vec{x}, \vec{v})$ at a given time t. This kind of approximation is possible because every absolutely continuous probability measure on $\Omega \times I\!R^3$ can be approximated by a sum of discrete measures μ_N of the form

$$\mu_N(\vec{x}, \vec{v}) := \frac{1}{N} \sum_{i=1}^{N} \delta(\vec{x} - \vec{x}_i) \times \delta(\vec{v} - \vec{v}_i) \xrightarrow{w} f d\vec{v} d\vec{x} \qquad \text{for } N \to \infty,$$

where f is the density of the measure μ and "w" denotes the weak convergence.
The discrete measures can be used in the following way to approximate the solution of the Boltzmann equation :
The initial condition f_0 is approximated by a finite pointset of the form $(\vec{x}_i(0), \vec{v}_i(0))$.
The next step is to find a time evolution of the set $(\vec{x}_i(t), \vec{v}_i(t))$ such that the time dependent discrete measure converges weakly to the solution of the Boltzmann equation.
To construct the time evolution of the discrete particle set, the first step is to consider a time discretization of the Boltzmann equation (1) in the form

$$f(\Delta t, \vec{x}, \vec{v}) = f_0(, \vec{x}, \vec{v}) + \Delta t \cdot J(f_0(\vec{x}, \vec{v}), f_0(\vec{x}, \vec{v}))$$

Next, the spatial domain Ω is divided into a disjoint covering $\{\Omega_i\}_{i=1,\ldots,K}$ of the form

$$\Omega = \overset{\circ}{\bigcup_{i=1,\ldots,K}} \Omega_i$$

On Ω the solution of (1) is approximated by a step function according to the given cell system $\{\Omega_i\}_{i=1,\ldots,K}$. This spatial discretization leads to the following time and space discretized Boltzmann equation

$$\tilde{f}(\Delta t, \vec{x}, \vec{v}) = (1 + \Delta t \cdot J)(P\tilde{f}_0)(\vec{x}, \vec{v}) \tag{4}$$

where

$$\tilde{f}(\Delta t, \vec{x}, \vec{v}) = \sum_{i=1}^{K} f_{\Omega_i}(\Delta t, \vec{v})$$

and

$$(P\tilde{f})(\vec{x}, \vec{v}) := \frac{\int_{\Omega_i} \tilde{f}(0, \vec{x} - \Delta t\vec{v}, \vec{v}) d\vec{x}}{Vol(\Omega_i)} \qquad \text{for } \vec{x} \in \Omega_i \tag{5}$$

According to equation (4) it is obvious that the time evolution consists of two steps:

(i) apply the operator P to the approximation of the initial condition f_0. The approximation of f_0 is given by a sum of step functions on the spatial cell system. According to (5) every particle performs a free transport over the time step Δt with the given velocity v_i. If the approximation is given by a continuous function, the next step will be the projection to the given cell system. For discrete measures the particles have to be sorted with respect to the cell system in order to perform the collision process. During the free transport of the particle set, particles can interact with the given boundary $\partial\Omega$. In this case one has to consider the boundary condition given at $\partial\Omega$, i.e. to fulfill the relation (3).

(ii) apply the operator $(1 + \Delta t J)$ to the pointset of step (i). This operator describes the influence of binary collisions. The collision process is based on the homogeneous Boltzmann equation, because the spatial discretization is given by a step functions and the operator J acts locally on the space coordinate $\vec{x}$.

3 The Boltzmann simulation with axisymmetric geometry

We consider now the Boltzmann equation together with an axisymmetric boundary $\partial\Omega$. As already mentioned, if the initial condition f_0 as well as the scattering kernel R are independent of the angular component ϕ, the solution f will be invariant under the group of rotations about the x-axis. It is obvious to use this symmetry for a simulation method like the one described in section 2.

3.1 The Boltzmann equation in cylindrical coordinates

There are two ways to consider the Boltzmann equation in a cylindrical coordinate system. The first one is to use a complete transformation of the Boltzmann equation into the cylindrical system, which is given by the following transformation in the phase space $\Omega \times \mathbb{R}^3$

$$\begin{aligned} x &= x' \\ y &= r \cdot \cos\phi \\ z &= r \cdot \sin\phi \\ v_x &= v'_x \\ v_y &= \cos\phi \cdot v_r - \sin\phi \cdot v_\phi \\ v_z &= \sin\phi \cdot v_r + \cos\phi \cdot v_\phi \end{aligned}$$

Since the velocity space is also transformed into cylindrical coordinates, the collision operator $J(f,f)$ on the right hand side of the equation has to be transformed with respect to the new coordinates as well. Furthermore artificial force terms appear on the left hand side of the equation, because the velocity components v_r and v_ϕ do not only depend on the velocities v_y and v_z, but also on the space coordinate ϕ. The Boltzmann equation is given by

$$\frac{\partial f}{\partial t} + v_x \cdot \frac{\partial f}{\partial x} + v_r \cdot \frac{\partial f}{\partial r} + \frac{v_\phi}{r} \cdot \frac{\partial f}{\partial \phi} + \frac{v_\phi^2}{r}\frac{\partial f}{\partial v_r} - \frac{v_r \cdot v_\phi}{r}\frac{\partial f}{\partial v_\phi} = \hat{J}(f,f) \qquad (6)$$

where $\hat{J}$ denotes the transformed collision operator.
From a numerical point of view it is much more complicated to construct a numerical method with total axisymmetric geometry than a classical two or three dimensional code. The main problem is given by the transformation of the collision operator to axisymmetric geometry (see [12]).
Furthermore the new terms on the left hand side of equation (6) have to be modelized in the free transport of the particle set. Therefore we will not consider this approach here.

If one uses only the transformation of the spatial coordinates into cylindrical representation and leaves the velocity components unchanged, the Boltzmann equation can be written as

$$\frac{\partial f}{\partial t} + v_x \cdot \frac{\partial f}{\partial x} + (\cos\phi \cdot v_y + \sin\phi \cdot v_z) \cdot \frac{\partial f}{\partial r} + \frac{1}{r} \cdot (-\sin\phi \cdot v_y + \cos\phi \cdot v_z) \cdot \frac{\partial f}{\partial \phi} = J(f,f)$$

In a simulation method such as the one of section 2 the free transport of particles, i.e. the hyperbolic left hand side of the Boltzmann equation, is decoupled from the binary interactions described by the collision operator J.
The solution of the left hand side of the equation, i.e.

$$\frac{\partial f}{\partial t} + v_x \cdot \frac{\partial f}{\partial x} + (\cos\phi \cdot v_y + \sin\phi \cdot v_z) \cdot \frac{\partial f}{\partial r} + \frac{1}{r} \cdot (-\sin\phi \cdot v_y + \cos\phi \cdot v_z) \cdot \frac{\partial f}{\partial \phi} = 0 \tag{7}$$

with initial condition

$$f(t=0, x, r, \phi, \vec{v}) = a(x, r, \phi, \vec{v})$$

is given by

$$f(t, x, r, \phi, \vec{v}) = a(x - tv_x, T_r(-t), T_\phi(-t), \vec{v}) \tag{8}$$

where $T_r(t)$ and $T_\phi(t)$ are the solutions of the coupled system

$$\begin{aligned} \dot{r} &= \cos\phi \cdot v_y + \sin\phi \cdot v_z \\ r \cdot \dot{\phi} &= -\sin\phi \cdot v_y + \cos\phi \cdot v_z \end{aligned} \tag{9}$$

with initial condition $r(0) = r$, $\phi(0) = \phi$. The solutions $T_r(t)$ and $T_\phi(t)$ of the system (9) are given by

$$T_r(t) = [r^2 + 2tr \cdot (\cos\phi \cdot v_y + \sin\phi \cdot v_z) + t^2 \cdot (v_y^2 + v_z^2)]^{\frac{1}{2}} \tag{10}$$

$$T_\phi(t) = \arctan\left(\frac{r \cdot \sin\phi + t \cdot v_z}{r \cdot \cos\phi + t \cdot v_y}\right) \tag{11}$$

which correspond to the following linear functionals in cartesian coordinates

$$\begin{aligned} T_y(t) &= y + t \cdot v_y \\ T_z(t) &= z + t \cdot v_z \end{aligned}$$

and

$$\begin{aligned} T_r(t) &= (T_y^2(t) + T_z^2(t))^{\frac{1}{2}} \\ T_\phi(t) &= \arctan\left(\frac{T_z(t)}{T_y(t)}\right) \end{aligned}$$

Therefore the free transport of a finite pointset in cylindrical coordinates can be done according to equation (8), (10) and (11).

There is no change in the collision process, because the velocity vectors of the particles are expressed in the original cartesian coordinate system. The standard routines can be applied directly.
We will now consider the reduction to two dimensional space coordinates using the rotational invariance of the solution of the Boltzmann equation.
There are three main problems for reducing the space coordinates to the cylinder coordinates (x, r):

(i) a uniform spatial discretization according to the coordinates x and r leads to an increasing number of points in the radial direction using discrete point measures. Therefore it is necessary to introduce a weighting function in the radial direction r to achieve a reasonable space discretization.

(ii) the weighting function leads to a modified Boltzmann equation and to modifications in the free transport of the particles.

(iii) the modified Boltzmann equation for the weighted distribution function requires modifications in the radial distribution of the particles.

3.2 The spatial discretization with axisymmetric geometry

If we consider a uniform space discretization with cellsize Δ_r in the x–r– plane, where r is the radial component of the system, then the corresponding cell system in the original y–z–plane is given by rings with width Δ_r.
The area A_n of the n–th ring is given by

$$A_n = (2n-1) \cdot \pi \cdot \Delta r^2$$

If we now assume a uniform spatial distribution on Ω then the number of particles in the single cells must be chosen with respect to the area A_n. This means that the particle number N_n in radial direction at a fixed point x^0 is given by the sequence

$$N_n = (2n-1) \cdot N_1$$

where N_1 is the number of particles at the first radial cell.
From a numerical point of view it is not useful to work with increasing particle numbers in the cells with increasing radial distance to the cylinder axis, because the statistical fluctuations in the results at the cells near to the cylinder axis will be much higher than far away from the axis. Furthermore decreasing the fluctuations near the cylinder axis requires too much particles far away from the axis.
A similar problem occurs when using a sequence of radial cellsizes Δr_n such that the corresponding areas A_n remain constant with n. In this case the sequence Δr_n must be chosen according to the formula

$$\Delta r_n = (\sqrt{n} - \sqrt{n-1}) \cdot \Delta r, \; n = 1, 2, ...$$

The problem now is that the cellsizes far away from the cylinder axis are very small, so that one needs a lot of cells in the radial direction to cover a given spatial domain.

Both spatial discretization concepts described above are not useful in a numerical simulation method. It is necessary to introduce the concept of weighted particles in the spatial domain Ω, where the weights of the particles are given by a weighting function, which depends only on the radial distance . The task of the weighting function is to correct the differences in the areas A_n, such that the corresponding particle numbers remain constant with increasing radial distance.

3.3 Boltzmann equation with a radial weighting function

The concept of a weighting function $R(r)$ can be included in the Boltzmann equation by considering the dynamic behaviour of the function

$$g(t, x, r, \phi, \vec{v}) := R(r) \cdot f(t, x, r, \phi, \vec{v})$$

It is easy to verify that the time evolution for the function g in cylindrical space coordinates is given by

$$\begin{aligned} \frac{\partial g}{\partial t} &+ v_x \cdot \frac{\partial g}{\partial x} + (\cos\phi \cdot v_y + \sin\phi \cdot v_z) \cdot \frac{\partial g}{\partial r} \\ &+ \frac{1}{r} \cdot (-\sin\phi \cdot v_y + \cos\phi \cdot v_z) \cdot \frac{\partial g}{\partial \phi} \\ &- (\cos\phi \cdot v_y + \sin\phi \cdot v_z) \cdot \frac{\partial \ln R}{\partial r} \cdot g = \frac{1}{R} \cdot J(g, g) \end{aligned}$$

A numerical simulation procedure for the equation above can be set up exactly as for the original Boltzmann equation. The discretized equation can also be decoupled into a free transport according to the left hand side and a collision process according to the collision operator $J(g, g)$.
The new term on the left hand side influences only the free transport of the particles. The solution of the equation

$$\begin{aligned} \frac{\partial g}{\partial t} &+ v_x \cdot \frac{\partial g}{\partial x} + (\cos\phi \cdot v_y + \sin\phi \cdot v_z) \cdot \frac{\partial g}{\partial r} \\ &+ \frac{1}{r} \cdot (-\sin\phi \cdot v_y + \cos\phi \cdot v_z) \cdot \frac{\partial g}{\partial \phi} \\ &= (\cos\phi \cdot v_y + \sin\phi \cdot v_z) \cdot \frac{\partial \ln R}{\partial r} \cdot g \end{aligned} \tag{12}$$

with initial condition

$$g(t = 0, x, r, \phi, \vec{v}) = a(x, r, \phi, \vec{v}),$$

is given by

$$g(t, x, r, \phi, \vec{v}) = \frac{R(T_r(-t))}{R(r)} \cdot a(x - tv_x, T_r(-t), T_\phi(-t), \vec{v}), \tag{13}$$

which can be shown by direct calculation. The functions $T_r(t)$ and T_ϕ are given according to (10) and (11).
We show now that the difference between the solution (8) and (13)

$$\frac{R(T_r(-t))}{R(r)}$$

represents exactly the correction of the weight of a single particle moving in radial direction. The correction must be implemented in the simulation procedure when using cylindrical coordinates and a weighting function $R(r)$.

3.4 Simulations with a radial weighting function

We will now consider the numerical simulation for the solution g of equation (12).
In a numerical simulation method, as described in section 2, the solution of the Boltzmann equation is approximated by a finite pointset of the form

$$f(t,\vec{x},\vec{v}) \approx \frac{1}{N}\sum_{i=1}^{N} \delta(\vec{x}-\vec{x}_i(t)) \times \delta(\vec{v}-\vec{v}_i(t))$$

Hence we consider now an approximation of the solution g of (12) by a pointset of the form

$$g(t,x,r,\phi,\vec{v}) \approx \sum_{i=1}^{N} \alpha_i(r_i(t)) \cdot \delta(x-x_i(t)) \times \delta(r-r_i(t)) \times \delta(\phi-\phi_i(t)) \times \delta(\vec{v}-\vec{v}_i(t))$$

with weights α_i dependent on the radial coordinate r. The dynamical behaviour of the finite pointset is described by the discretized version of equation (12) according to a time discretization Δt and the cellsystem $\{\Omega_i\}_{i=1,\ldots,K}$ and is exactly constructed as in section 2.
After decoupling the free transport from the collision operator $J(g,g)$, the first step in the simulation method is to perform the free transport given by the solution (13) of equation (12). If we assume that at time $t=0$ the function $g(t=0,x,r,\phi,\vec{v})$ is given by a finite pointset with weights α_i^0, then the solution of (12) is

$$\begin{aligned} g(\Delta t,x,r,\phi,\vec{v}) = \sum_{i=1}^{N} \frac{R(T_{r_i}(-\Delta t))}{R(r_i)} \cdot \alpha_i^0 \cdot \delta(x-(x_i-\Delta t\cdot v_{x,i})) \\ \times\delta(r-(T_{r_i}(-\Delta t))) \times \delta(\phi-(T_{\phi_i}(-\Delta t))) \times \delta(\vec{v}-\vec{v}_i(t)) \end{aligned} \tag{14}$$

With a given uniform discretization in the x–r–plane at time $t=0$ there exist two possibilities for the weighting function $R(r)$. The first one is to choose the function $R(r)$ inverse proportional to the radial distance r, i.e.

$$R(r) = \frac{1}{r}$$

Then the number of particles is constant in every cell.
The main disadvantage is that the weights of the points vary with the radial distance r: in every cell, where the distribution function f is assumed to be homogeneous, the pointset consists of particles, which have different weights. The different weights make the collision procedure of a numerical simulation method much more complicated. Furthermore it is not possible to homogenize the weights over a single cell without changing the macroscopic quantities of the distribution function f.
Therefore it is much more useful to introduce the weighting function $R(r)$ as a step function over the radial distance r, i.e.

$$R(r) = \sum_{j=1,\dots,M} \frac{1}{2j-1} \chi_{[(j-1)\Delta r, j\Delta r]}(r) \tag{15}$$

where M is the total number of cells in radial direction. In this case the weights of the particles are elements of a discrete set and depend only on the index of the cell in radial direction, where the particles are actually located. It is obvious that it will be possible to homogenize the particle weights in a given cell without changing the macroscopic parameters, because in a given cell the weight is constant.
According to formula (15) the weight of a particle changes, if the particle crosses cell boundaries in radial direction. Therefore it is not possible to apply directly the collision procedure to the pointset (14), because particles located in the same cell can have different weights.
There are two possibilities to overcome this difficulty:
The first one is to homogenize the particles in one cell with respect to the weight prescribed by the weighting function $R(r)$. Such a homogenization step was already used in the context of weigthed particle simulations ([13]).
The second possibility was already mentioned by Bird ([4]) without a general theoretical background however. The underlying principle is to fix the weights in every step at the initial value $\alpha_i^0 = 1/N$. This is possible if one considers the change in the weights α_i given by

$$P_i(j \to k) = \frac{R(r_i)}{R(T_{r_i}(-\Delta t))}$$

(with $(k-1)\Delta r \le r_i \le k\Delta r$ and $(j-1)\Delta r \le T_{r_i}(-t) \le j\Delta r$) as the'staying alive' probability of the i-th particle. If a particle stays in the same radial section the weight α_i remains unchanged, whereas if a particle crosses a radial boundary during the free transport the particle has to be deleted or doubled according to the probability P_i.
For example, if a particle moves from the first to the second radial section the probability $P_i(1 \to 2)$ is equal to $\frac{1}{3}$, which means that the particle will be deleted with probability $\frac{2}{3}$, on the other hand the probability $P_i(2 \to 1)$ is equal to 3, which means that one has to create three identical particles instead of one.
Therefore it is unnecessary to save a weight α_i for every particle. With this modification in the free transport of the function g it is guaranteed that the fluxes across the radial boundary are approximated correct.

The modifications in the free transport described above still use three space coordinates in a cylindrical system. We will now consider the techniques to reduce the

space discretization to the coordinates x and r.
The free transport of the particles requires three dimensional space coordinates, because it is necessary to use the relation between the velocity vector in the cartesian and the cylindrical coordinate system. On the other hand it is no restriction to fix at a given time t_0 all particles at a given angular component ϕ_0, because the solution will be invariant under rotations about the x-axis. If all particles are located at the same angular component it is not necessary to save the information for all particles, such that the discrete measure can be written in the form

$$g(t_0, x, r, \vec{v}) \approx \sum_{i=1}^{N} \alpha_i \cdot \delta(x - x_i(t_0)) \times \delta(r - r_i(t_0)) \times \delta(\vec{v} - \vec{v}_i(t_0))$$

With this approximation and the global angular component ϕ_0 it is possible to perform the free transport according to equation (14). After applying (14) all particles will have a different angular component ϕ in dependence on the velocity vector $\vec{v}$. Now it is necessary to transform this local information of a single particle again to a global one in order to reduce the amount of information, which has to be stored for a single particle. In other words the individual angular coordinate ϕ_i of the particles have to be transformed somehow to the given value ϕ_0. This can be done by rotating the given velocity components $v_{y,i}$ and $v_{z,i}$ acoording to the angle $\phi_{rot} = \phi_0 - \phi_i$. After the rotation of the velocity components v_y and v_z all particles are again located at the fixed angular component ϕ_0. The spatial information which has to be stored for a single particle is now reduced to the cylinder coordinate x and the radial coordinate r.
After the free transport all particles, which are located in the same cell, have the prescribed weight, so it is possible to apply directly the collision operator on the cartesian velocity components as given in the classical simulation method.

Finally we want to focus again on the spatial distribution of particles according to the weighted distribution function. As an example we consider in the following a uniform distribution on a bounded spatial domain in $\mathbb{R}^3$ given by a cylinder with length L and radius R_0. Then the spatial measure for the modified distribution g is given by

$$g_m(x, r, \phi) d\phi dr dx = \frac{1}{2\pi \cdot L \cdot R_0} \sum_{j=1,\ldots,M} \frac{1}{2j-1} \chi_{[(j-1)\Delta r, j\Delta r]}(r) d\phi r dr dx$$

The corresponding particle approximation is then

$$\begin{aligned} x_i &= t_i^1 \cdot L \\ r_i &= [(2j-1) \cdot t_i^2 - (j-1)^2]^{\frac{1}{2}} \cdot \Delta r, \quad \text{for } r_i \in [(j-1)\Delta r, j\Delta r],\ j = 1, 2, \ldots \\ \phi_i &= t_i^3 \cdot 2\pi \\ \alpha_i &= \frac{1}{N} \end{aligned}$$

where the triple (t_i^1, t_i^2, t_i^3) is a random number in $[0,1]^3$ and N is the number of particles per cell.

It is important to notice that the radial direction is not uniformly distributed although the original function f is corrected by the radial weighting function R. The weighting function R can only correct the differences between radial sections, inside a given radial section the approximation must be performed according to the exact measure. The difference between the (incorrect) uniform distribution in radial direction and the (correct) nonuniform distribution decrease with increasing radial distance, but in sections near to the x–axis the differences are quite important. The use of the incorrect distribution can lead to undesired artificial effects in the numerical simulation method. These effects, among others, are illustrated in the next section.

4 Numerical results

The modified Boltzmann simulation method described in the previous section was used to calculate a rarefied gas flow around a body with axisymmetric geometry. This work was done in the framework of the european space research project HERMES. A detailed description of the numerical results can be found in ([10]).

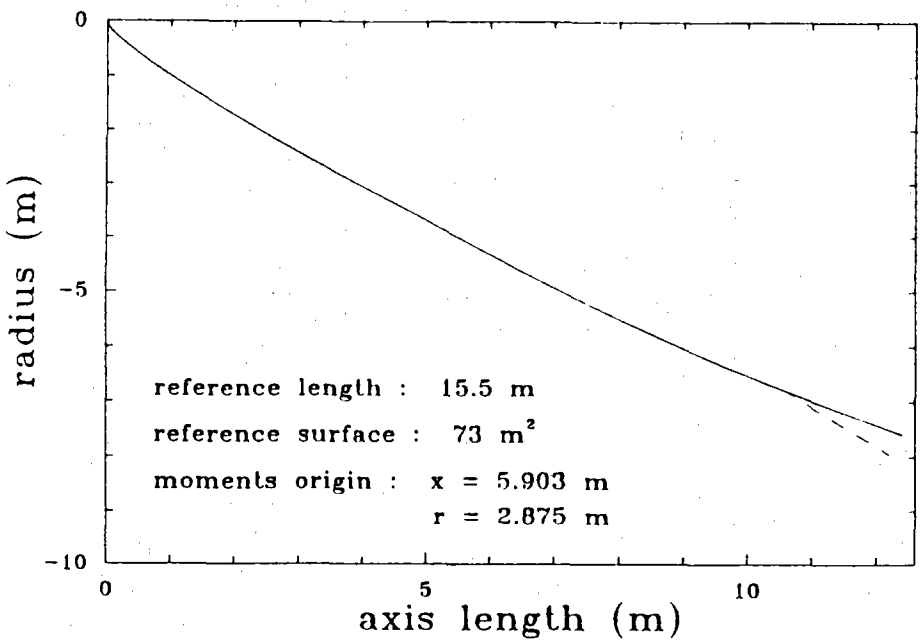

Fig. 1: HERMES windward centerline near the symmetry plane

The geometry of the axisymmetric body is given by the windward centerline of the HERMES space shuttle (in its symmetry plane), starting at the stagnation point for 30 degree angle of attack up to the leading edge. (see figure 1). The threedimensional axisymmetric body is created by rotating the centerline with respect to the x-axis. The physical parameters of the gas flow are choosen according to different reentry altitudes. Furthermore the geometry of the body is slightly modified by including a small flap at the leading edge (see also figure 1). The main task for this configuration is to determine the influence of the flap deflexion angle on the aerodynamic characteristics of the body. In a more extensive study the influence of real gas effects like vibrational or chemical effects will be examined.
The physical parameters of a gas flow at altitudes around 100 km demand a very fine spatial discretization. This requirement can easily lead to the limits of computer hardware, if the simulation method is based on standard threedimensional space coordinates. Therefore it is essential to consider the simulation procedure together with cylindrical space coordinates as the one described above.

Fig. 2: Channel flow at equilibrium with incorrect spatial distribution

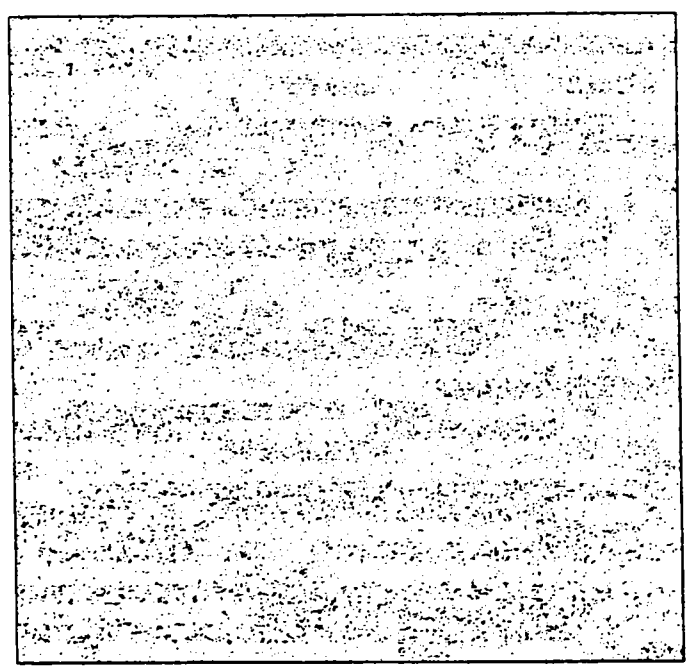

Fig. 3: Channel flow at equilibrium with correct spatial distribution

The amount of memory can be reduced by orders of magnitude using the axisymmetric simulation code.
All the calculations shown in the following were done on a nCUBE 2 parallel computer at the Laboratory of Technomathematics, University of Kaiserslautern. The implementation of the simulation code on a parallel computer is based on the work given in [14].

At first we want to focus on the spatial distribution of the particles at the beginning of the simulation procedure and the way how to construct the spatial distribution on the artificial boundaries. As mentioned in the previous section it is very important to consider the correct spatial measure for the radial direction r. The correct measure is given by a nonuniform distribution in radial direction, although the space distribution is uniform over the whole threedimensional domain.
The numerical simulation of a simple channel flow at equilibrium conditions shows the influence of the numerical solution on the spatial distribution:
The results for a (incorrect) discretization according to a uniform spatial distribution are shown in figure 2. Along the x–axis the deviations from equilibrium are evident. With increasing distance from the artificial boundary some artificial shock structures appear near the cylinder axis. Figure 7 shows the same flow problem together with the (correct) modified nonuniform space discretization. The equilibrium is preserved in the whole channel without artificial numerical error.

Altitude	Gas	$T_\infty\ [K]$	Ma	$T_w\ [K]$	$\lambda_\infty\ [m]$
120	N_2	368	20	1400	2.69
110	N_2	247	23	1400	0.60
100	N_2	194	25	1400	0.137
Altitude	Partnr	Cellnr	Part/Cell	Timesteps	CPU $[h]$
120	570,000	11,264	64	1000	1.5
110	925,000	11,264	64	1000	2.5
100	2,000,000	40,960	36	1000	4.0
Altitude	$C_{d,0°}$	$C_{l,0°}$	$(L/D)_{0°}$	$C_{h,0°}$	$C_{m,0°}$
120	2.191	.890	.406	.868	.882
110	1.688	1.048	.621	.539	.641
100	1.360	1.170	.860	.313	.490
Altitude	$C_{d,12°}$	$C_{l,12°}$	$(L/D)_{12°}$	$C_{h,12°}$	$C_{m,12°}$
120	2.304	.941	.408	.901	.974
110	1.785	1.109	.621	.557	.727
100	1.461	1.246	.853	.325	.584
	Drag	Lift	$\leftarrow coeff \rightarrow$	Heat	Pitching

Table 1 : Numerical parameters and results

Table 1 shows some typical results for the axisymmetric body described above. With the axisymmetric code it is possible to perform calculations with a very high dis-

cretization and therefore to obtain very accurate results. Furthermore the modifications in the simulation procedure are not very time consuming compared with the total CPU times. Moreover it is possible to reduce the specified total CPU times by using the techniques described in [14] to optimize the load balance of the parallel processors.
An illustration of a typical flow field is shown in figure 4. Part (a) of the figure shows the global Mach field at an altitude of 100 km, part (b) the rotational temperature of the flow near the flap with a flap deflexion angle of 0 degree and part (c) the corresponding rotational temperature for a flap deflexion angle of 12 degree.
Finally figure 5 shows the local pressure coefficient along the surface line in comparison with the modified Newton theory. The local surface quantities show only small fluctuations, although the number of averaging time steps is very small compared with other calculations.

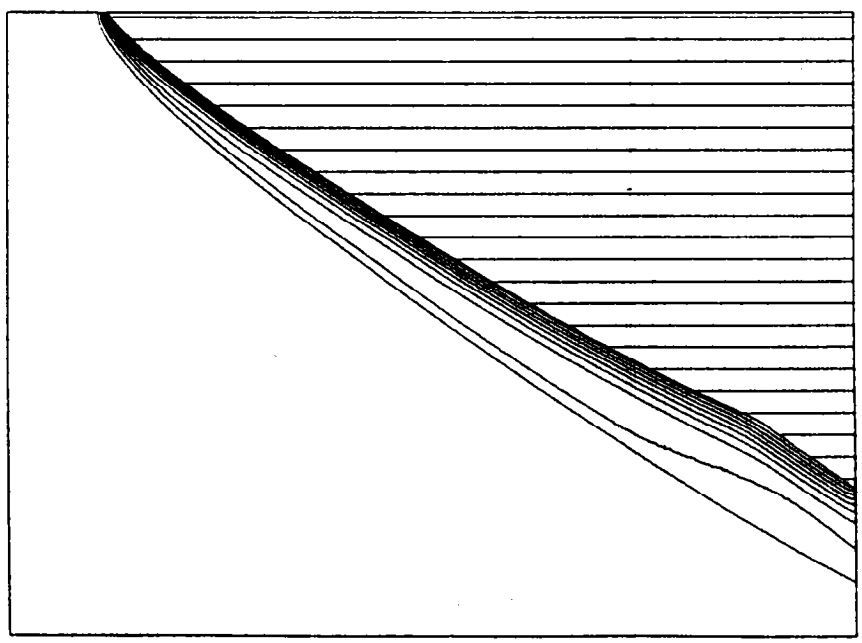

Fig. 4(a): Mach distribution at 100 km altitude

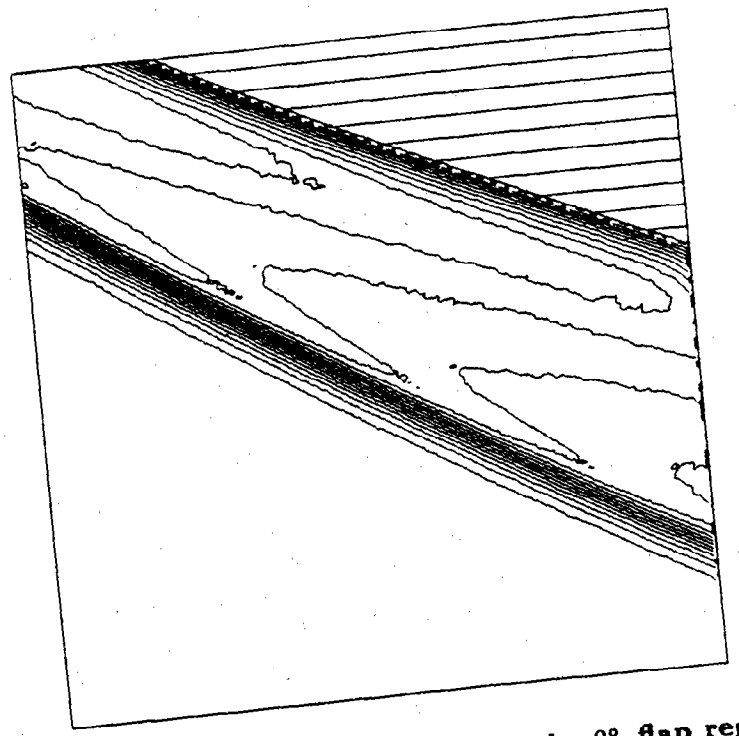

Fig. 4(b): Kinetic temperature near the 0°–flap region

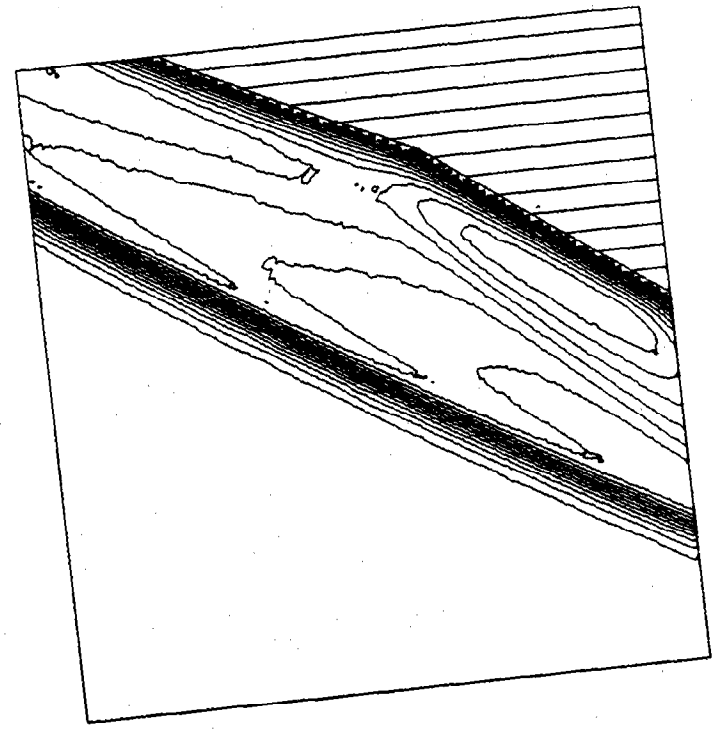

Fig. 4(c): Kinetic temperature near the 12°–flap region

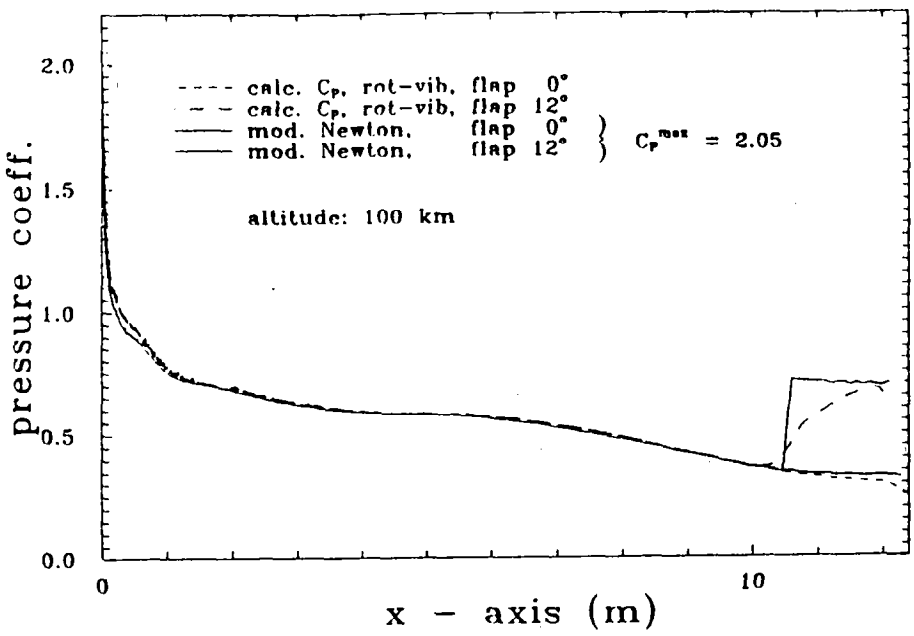

Fig. 5: Local pressure coefficient along the surface line

5 Conclusion

In the numerical simulation of axisymmetric rarefied gas flows it is reasonable to take advantage of the symmetry in the solution of the Boltzmann equation. On the other hand it is not straightforward how to include the symmetry in the classical simulation methods.

The transformation to cylindrical space coordinates leads automatically to the concept of weighting functions in the radial space coordinate. Otherwise it is not possible to achieve an adequate space discretization over the whole spatial domain. By introducing weighting functions it is necessary to consider a modified Boltzmann equation. This modified Boltzmann equation can be discretized in the same way as the original equation and the modifications of the numerical method affect only the free transport and the spatial distribution of the particles. The collision process remains unchanged. Numerical results illustrate the essential increase in computational efficiency which is achieved by the presented method.

References

[1] Babovsky, H.: *A convergence proof for Nanbu's Boltzmann simulation scheme*, European Journal of Mechanics, B/Fluids, 8, No. 1, 1989.

[2] Babovsky, H.: *On a simulation scheme for the Boltzmann equation*, Math. Meth. Apll. Sci., 8, 1986.

[3] Babovsky, H.; Illner, R.: *A convergence proof for Nanbu's simulation method for the full Boltzmann equation*, SIAM J. Num. Anal., 26, 1989.

[4] Bird, G.: *Molecular Gas Dynamics*, Clarendon Press, Oxford, 1976.

[5] Dogra, V.K.; Moss, J.N.; Wilmoth, R.G.: *DSMC Simulations of Hypersonic Low-Density Flow About an ASTV Inlcuding Wake Structure*, to appear

[6] Gropengießer, F.; Neunzert, H.; Struckmeier, J.; Wiesen, B.: *Rarefied gas flow around a disc with different angles of attack*, Proc. of the 17th Symp. on rarefied gas dynamics, Beylich, A.(ed.), Weinheim, 546–553, 1991.

[7] Gropengießer, F.; Neunzert, H.; Struckmeier, J.; Wiesen, B.: *Rarefied gas flow around a 3d–deltawing at low Knudsen numbers*, Proc. of the 17th Symp. on rarefied gas dynamics, Beylich, A.(ed.), Weinheim, 332–336, 1991.

[8] Gropengießer, F.; Neunzert, H.; Struckmeier, J.; Wiesen, B.: *Rarefied gas flow around a 3d–deltawing*, to appear in the Proc. of the workshop on hypersonic flows for reentry problems, Part II, Antibes.

[9] Neunzert, H.; Gropengießer, F.; Struckmeier, J.: *Computational methods for the Boltzmann equation*, in: Venice 1989: The state of art in appl. and ind. math., ed. Spigler, R., 1990.

[10] Neunzert, H.; Sack, W.: Steiner, K.; Struckmeier, J.: *Interim report for the project DPH 6486/91*, Internal report on the HERMES project, R&Q, 1992.

[11] Nanbu, K.: *Direct simulation scheme derived from the Boltzmann equation*, Journal of Physics Japan, 49, 1980.

[12] Niclot, B.: *The two particle Boltzmann collision operator in axisymmetric geometry*, Report nr. 164, Ecole Polytechnique, 1987.

[13] Schreiner, M. : *Weighted particles in The Finite-Pointset-Method*, Berichte der Arbeitsgruppe Technomathematik Nr. 62, Universität Kaiserslautern, 1991.

[14] Struckmeier, J.; Pfreundt, F.J.: *On the efficiency of simulation methods for the Boltzmann equation on parallel computers*, to appear in Parallel Computing.

[15] Wagner, W.: *A convergence proof for Bird's direct simulation Monte Carlo method for the Boltzmann equation*, Karl–Weierstrass Institut für Mathematik, 1990.