

TA sl.

FORSCHUNG - AUSBILDUNG - WEITERBILDUNG

Bericht Nr. 91

A COMPARISON OF SIMULATION METHODS

FOR RAREFIED GAS FLOWS

Jens Struckmeier, Konrad Steiner

**UNIVERSITÄT KAISERSLAUTERN
Fachbereich Mathematik
Postfach 3049**

W-6750 Kaiserslautern

Juni 1993

A Comparison of Simulation Methods for Rarefied Gas Flows

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

Abstract

Simulation methods like DSMC are an efficient tool to compute rarefied gas flows. Using supercomputers it is possible to include various real gas effects like vibrational energies or chemical reactions in a gas mixture. Nevertheless it is still necessary to improve the accuracy of the current simulation methods in order to reduce the computational effort. To support this task the paper presents a comparison of the classical DSMC method with the so called Finite Pointset Method. This new approach was developed during several years in the framework of the European space project HERMES. The comparison given in the paper is based on two different testcases: a spatially homogeneous relaxation problem and a 2-dimensional axisymmetric flow problem at high Mach numbers.

Keywords: Rarefied gas flows; Direct Simulation Monte Carlo (DSMC); Finite Pointset Method (FPM); Boltzmann equation.

1 Introduction

The paper presents a comparison of the classical DSMC method for rarefied gases with the Finite Pointset Method (FPM) for the Boltzmann equation, developed at the University of Kaiserslautern. The investigation is based on two different testcases: the first one is a spatially homogeneous relaxation problem, the second one a twodimensional axisymmetric flow problem at high Mach numbers. In the first testcase the initial condition at time $t = 0$ is a nonequilibrium distribution, which relaxes for $t \rightarrow \infty$ to equilibrium based on the collision invariants at time $t = 0$. The second testcase is a typical multidimensional situation where one wants to compute the aerodynamical characteristics of a reentry body.

The main interest of the paper is to compare two numerical methods; consequently, the parameters describing the physical nature of the gas were chosen as simple as possible. The binary collisions between the gas particles are described by a hard sphere interaction law (for monoatomic gases); the gas surface interaction is assumed to be total thermal accommodation with diffusive reflection. The collision process is the most interesting part in a simulation procedure. Therefore in the multidimensional testcase the different collision routines were checked within a fixed environment, i.e. a fixed spatial discretization with equal parameters for all cases.

The first method considered is the classical DSMC method as originally proposed by Bird ([5]). The DSMC method is based on the Time Counter approach together with an acceptance-rejection algorithm to detect real collisions. This approach is still the mainly used method to compute rarefied gas flows. A modified version of the classical approach, the so called No Time Counter method, proposed by Bird ([6]) too is also considered in this paper. The implementation of the FPM is based on the work given in [9]. Two different realizations are investigated. In the first version the collision pairs are chosen with aid of a randomly chosen permutation of the integer set $\{1, \dots, \frac{N}{2}\}$, where N is the number of particles in a given cell. In the second version the collision pairs are chosen according to an 'update' procedure where one particle can have more than one collision during a fixed timestep.

An important part of a simulation method for rarefied gas flows is the choice of the pseudo random generator. In [11] it was shown that low discrepancy sequences can be created with the same effort as random numbers with a linear congruential method. Therefore a comparison of the simulation methods with both types of generators will be also considered. The paper is organized as follows: First we describe briefly the two simulation methods used in the current investigation. Chapter 3 presents the results obtained for two different testcases using the DSMC and the FPM method. Finally we discuss the results of the numerical tests.

2 Simulation methods for rarefied gas flows

Still the most widely used simulation method for rarefied gas flows is the DSMC method developed by Bird in the mid seventies. The classical DSMC method is based on the Time Counter approach, which is used to approximate the exact collision frequency within a reduced number of real gas particles.

In order to make use of vector computer architectures Bird presented a modified version of his method, the so called No Time Counter approach, which should overcome the problem of vectorizing the collision process of the DSMC method.

Due to the fast development of the hardware architecture it is nowadays possible to include much more physical effects, like vibrational energies or chemical reactions. Nevertheless the computational effort for such complicated calculations is very high until today and there is still a need to improve the numerical methods.

The second simulation method used for the description of rarefied gases within realistic applications is the so called Finite Pointset Method, developed at the University of Kaiserslautern. Different from DSMC which computes the evolution of a N -particle system, FPM is a numerical method for solving the Boltzmann equation. Based originally on a paper by Nanbu ([8]), it was first improved by Babovsky ([1],[2],[3]) and experienced afterwards a lot of modifications resulting in what we called FPM ([9],[10]).

Both methods were several times tested versus windtunnel measurements, where the numerical results obtained by threedimensional simulation codes.

In the first part we describe briefly the general feature of the DSMC method; the second part investigates the Finite Pointset Method.

2.1 Bird's DSMC method

The general approach of the DSMC method is described by the following:

'The direct simulation Monte Carlo method is a technique for the computer modelling of a real gas flow by some thousands of simulated molecules' (Bird ([5]), p. 118).

Instead of consider the real number of molecules (i.e. $\sim 10^{19}/m^3$) the DSMC method tries to simulate the behaviour of the real gas flow by a much smaller number of molecules.

Nevertheless the dynamical behaviour of the reduced number of molecules is based on the real gas flow:

'The velocity components and position coordinates of the simulated molecules are stored in the computer and are modified with time as the molecules are concurrently followed through representative collisions and boundary interactions in simulated physical space'

(Bird ([5]), p. 118).

In order to perform the simulation of the real dynamical process Bird proposed the following algorithm:

'The molecular motion and the intermolecular collisions are uncoupled over the small time intervall Δt_m by the repetition of the following procedure:

- (i) All the molecules are moved through distances appropriate to their velocity components and Δt_m . Appropriate action is taken if the molecule crosses boundaries representing solid surfaces, lines or surfaces of symmetry, or the outer boundary of

the flow. New molecules are generated at boundaries across which there is an inward flux.

- (ii) A representative set of collisions, appropriate to Δt_m , is computed among the molecules. The pre-collision velocity components of the molecules involved in the collisions are replaced by the post-collision values. Since the change in flow variables across a cell is small, the molecules in a cell at any instant may be regarded as a sample of the molecules at the location of the cell. This enables the relative positions of the molecules within the cell to be disregarded when choosing collision pairs.'

(Bird ([5]), p. 120)

For the construction of the representative set of collisions the question remains how to select the appropriate collision pairs as well as how to determine the appropriate number of collisions.

In order to determine the correct total number of collisions during the time step Δt_m it would be necessary to compute the modulus of the relative velocity vector for all possible pairs of molecules (this can be directly verified by considering a N -particle system with a fixed N). An algorithm which uses this approach would have a computational effort of order N^2 , where N is the total number of simulated molecules. This effort cannot be handled for realistic applications.

To overcome this difficulty Bird proposed to introduce a parameter V (i.e. an upper bound for the exact maximum), which should be updated during a binary collision if the actual relative velocity is greater than V . Furthermore the parameter V is used to determine a real collision:

- (1) Determine randomly two particles with velocities v and w out of the N^2 possible collision pairs.

- (2) If

$$\frac{|v - w|}{V} > rand$$

where $rand$ is a uniformly distributed random number in $[0, 1]$,

then perform the collision.

Otherwise go back to (1).

This is an acceptance-rejection method for the approximation of the representative set of collisions.

To get the correct number of collisions during the time step Δt_m , every collision is provided with a small time increment $\Delta\tau$, the so called time counter. The collision procedure is repeated as long as the sum over these increments $\Delta\tau$ remains less than Δt_m .

The expression for the time increment of a single collision (based on a hard sphere model) is given by

$$\Delta\tau = \frac{C}{N \cdot |v - w|}$$

where C is a gas dependent constant.

In the Time Counter approach the total number of collision within the given time discretization Δt_m depends on the relative velocity $|v - w|$ of a single collision. It is not possible to

calculate the total number of collisions at the beginning of the iteration step. Therefore a complete vectorization of the collision process is not possible.

In the No Time Counter approach ([6]) the time counter is independent of the actual relative velocity $|v - w|$ and is approximated with help of the parameter V :

$$\Delta\tau = \frac{C}{N \cdot V}$$

In this case the parameter V is fixed during a time iteration and modified after performing all collisions.

A theoretical investigation of the DSMC method without the modifications which are necessary to reduce the N^2 -effort was given by Wagner ([14]).

2.2 The Finite Pointset Method

FPM is a numerical method for solving the Boltzmann equation. It assumes that the Boltzmann equation describes the real flow behaviour accurately and that this equation has a unique \mathcal{L}^1 -solution decaying sufficiently fast for $|v| \rightarrow \infty$. Since the Boltzmann equation deals with the density function f and FPM with particles, we have to explain in which sense particle systems approximate densities. The theoretical foundation of the Finite Pointset Method is the approximation of a finite probability measures by a sum of discrete measures of the form

$$\frac{1}{N} \sum_{i=1}^N \delta(v - v_i) \rightarrow d\mu$$

where the convergence for $N \rightarrow \infty$ holds in the sense of the weak convergence of measures. The density function $f(v)$ of the gas ensemble is interpreted as the density of a probability measure μ and the Boltzmann equation describes the dynamic behaviour of μ .

Based on this kind of approximation a simulation method for rarefied gas flows can be derived from a discretized version of the Boltzmann equation. Since we are mainly interested in the simulation of the collision process we will only consider the spatially homogeneous Boltzmann equation

$$\frac{\partial f}{\partial t} = \frac{1}{\epsilon} J(f, f) \quad (1)$$

with

$$J(f, f) = \int_{w \in \mathbb{R}^3} \int_{\eta \in S_+^2} |v - w| \sigma(|v - w|, \eta) \{f(v')f(w') - f(v)f(w)\} d\omega(\eta) dw$$

$$v' = T_{v,w}(\eta) = v - \langle v - w, \eta \rangle \cdot \eta$$

$$w' = T_{w,v}(\eta) = w - \langle w - v, \eta \rangle \cdot \eta$$

and initial condition

$$f(t = 0, v) = f_0(v)$$

In the following we assume $\epsilon = 1$. Taken a discretization of the left hand side of (1)

$$\frac{\partial f}{\partial t} \approx \frac{f(\Delta t, v) - f(0, v)}{\Delta t}$$

the time discretized Boltzmann equation is given by

$$f(\Delta t, v) = (1 + \Delta t \cdot J)f(0, v) \quad (2)$$

Since the approximation of the function f is based on the weak convergence of measures it is useful to consider the weak formulation of (2)

$$\int_{v \in \mathbb{R}^3} \Phi(v) f(\Delta t, v) = \int_{v \in \mathbb{R}^3} \int_{w \in \mathbb{R}^3} (R\Phi)(v, w) f(0, v) f(0, w) dw dv \quad (3)$$

for all $\Phi \in C_b(\mathbb{R}^3)$ where

$$\begin{aligned} (R\Phi)(v, w) = & \Delta t \cdot |v - w| \int_{\eta \in S_+^2} \sigma(|v - w|, \eta) \Phi(v') d\omega(\eta) \\ & + \{1 - \Delta t \cdot |v - w| \int_{\eta \in S_+^2} \sigma(|v - w|, \eta) d\omega(\eta)\} \cdot \Phi(v) \end{aligned}$$

A more compact formulation for the operator R was given by Babovsky ([2]):

$$(R\Phi) = \int_{b \in B_{0, \sqrt{\pi}}} \Phi(T_{v, w} \circ K_{v, w}(b)) db$$

where $K_{v, w}$ is a suitable transformation of the impact parameter.

Equation (3) is used to construct a simulation method for the homogeneous Boltzmann equation:

Consider a given approximation of the initial condition in the form

$$f(0, v) \approx \frac{1}{N} \sum_{i=1}^N \delta(v - v_i)$$

Then in order to construct the approximation of f at time Δt according to equation (3) the first task is to construct an approximation of the density

$$f(0, v) f(0, w) dx dv dw \quad (4)$$

on $B_{0, \sqrt{\pi}} \times \mathbb{R}^3 \times \mathbb{R}^3$ by a discrete measure

$$\frac{1}{N} \sum_{i=1}^N \delta(b - b_i) \times \delta(v - v_i) \times \delta(w - w_i)$$

The solution at time $t = \Delta t$ is then given by the transformation

$$v_i(\Delta t) = T_{(v_i, w_i)} \circ K_{v_i, w_i}(b_i)$$

and the discrete measure at time $t = \Delta t$ by

$$f(\Delta t, v) \approx \sum_{i=1}^N \delta(v - v_i(\Delta t))$$

The convergence of such an procedure for $N \rightarrow \infty$ can be proved using the convergence theorems given for example in ([4]).

This procedure is a general concept for the construction of simulation methods for the Boltzmann equation. It is clear that there exist a lot of different realizations for the approximation of the product measure (4) for a given approximation of $f(0, v)$. An approximation of the product measure means a set of N collision pairs out of the given N initial points and a set of impact parameters for the N collision pairs. Two different realizations are explained in the following.

Equation (3) includes a conservation of energy and momentum of the gas ensemble and one useful aspect for a realization of (3) will be to construct a numerical method which also includes the conservation laws.

The standard procedure for the conservation of energy and momentum is to use the symmetry of the collision probability between two particles with index i resp. j :

$$P_{ij} = P_{ji}$$

If only both particles together can have a collision and the impact parameters are chosen such that

$$b_i = b_j \quad (5)$$

the above relation automatically leads to a conservation of energy and momentum.

Another way to include the conservation law is to construct the set of impact parameters $(x_i)_{i=1, \dots, N}$ such that the total energy and momentum is conserved. This procedure can be useful for the simulation of gas mixtures, where the single gas components appear in different concentrations in the mixture. The two different realizations presented here are based on the standard approach (5).

The first one uses permutations on the index set $\{1, \dots, \frac{N}{2}\}$ for the approximation of the product measure $f(v)f(w)dv dw$. This realization will be called 'FPM with permutation':

1st realization: FPM with permutation

The total set of points $(v_i)_{i=1, \dots, N}$ is divided into two disjoint sets $(v_i^1)_{i=1, \dots, \frac{N}{2}}$ and $(v_i^2)_{i=1, \dots, \frac{N}{2}}$ (we assume that N is an even number).

Based on a permutation Π of the index set $\{1, \dots, \frac{N}{2}\}$ the approximation of the product measure $f(0, v)f(0, w)dv dw$ is given by the formula

$$\frac{1}{N} \left[\sum_{i=1}^{N/2} \delta(v - v_i^1) \times \delta(w - v_{\Pi(i)}^2) + \sum_{i=1}^{N/2} \delta(v - v_{\Pi(i)}^2) \times \delta(w - v_i^1) \right]$$

The collision pairs are given by the set

$$(v_i^1, v_{\Pi(i)}^2) \text{ for } i = 1, \dots, \frac{N}{2}$$

and if a real collision occurs the post-collisional velocities are given by

$$\begin{aligned} v_i'^1 &= T_{v_i^1, v_{\Pi(i)}^2} \circ K_{v_i^1, v_{\Pi(i)}^2}(b_i^1) \\ v_{\Pi(i)}'^2 &= T_{v_{\Pi(i)}^2, v_i^1} \circ K_{v_{\Pi(i)}^2, v_i^1}(b_i^1) \end{aligned}$$

□

The collision procedure for this realization can be very efficiently implemented on a vector computer, since no recurrence relation occurs; all collisions can be handled inside a vector register.

According to equation (3) the time discretization parameter Δt has to be chosen small enough such that the term $(1 - \Delta t \cdot |v - w|)$ remains positive. For inhomogeneous flows the parameter Δt is further coupled with the space discretization Δx ; the gas particles should move in average over a spatial length Δx during the time step Δt . For inhomogeneous flows the first restriction on the parameter Δt is much more stronger. This strong restriction on Δt can be weakened by dividing the collision process into a fixed number of cycles N_C and performing in each cycle the collision procedure as given above.

For the spatially homogeneous Boltzmann equation the concept of collision cycles is exactly equal to a time discretization with parameter $\frac{\Delta t}{N_C}$. The sensitivity of the numerical solution on the number of cycles will be studied in chapter 3.

The second realization of equation (3), called 'FPM with update' in the following, is obtained in considering a fixed number of cycles equal to $\frac{N}{2}$:

2nd realization: FPM with update

$\frac{N}{2}$ collision pairs are chosen uniformly out of the N^2 possible pairs where a collision pair may even consist of two identical particles. The set of impact parameters are chosen such that the conservation of energy and momentum holds for a single collision. Furthermore the post-collisional velocities are directly updated in the ensemble, such that if the same particle occurs again in a collision pair within the same time step the new velocity is used. \square

The connection of this update procedure with the discretized Boltzmann equation is as follows:

If we assume that the time step Δt is chosen such that the term $(1 - \Delta t \cdot |v - w|)$ is nonnegative for all N^2 possible collision pairs, then the expected number of collisions during a time step of length $\frac{2\Delta t}{N}$ is less than 1. Therefore, if we consider the FPM with permutation for the time step Δt and a cycle number of $\frac{N}{2}$, it is not necessary to consider $\frac{N}{2}$ different collision pairs; at most one real collision will actually occur. On the other hand it is very time consuming to determine in every cycle the set of collision pairs and to decide if the collision is real or fictitious.

The computational effort for a cycle number of $\frac{N}{2}$ can be drastically reduced in considering only one collision pair out of the N^2 possible pairs and decide if a real collision occurs only for this pair. This is exactly done in the realization given above.

The total collision frequency for this realization is the same as in the FPM with permutation. Therefore the update scheme can be regarded as the FPM with permutation based on the time discretization of length Δt and a cycle number of $\frac{N}{2}$.

If the assumption on Δt that $(1 - \Delta t \cdot |v - w|)$ is nonnegative for all N^2 pairs is not valid, a global cycle number N_C , as in the case of the FPM with permutation, can be introduced. Then the algorithm given above is repeated N_C times during the timestep of length Δt .

A difference to the FPM with permutation is that it is possible to choose two identical

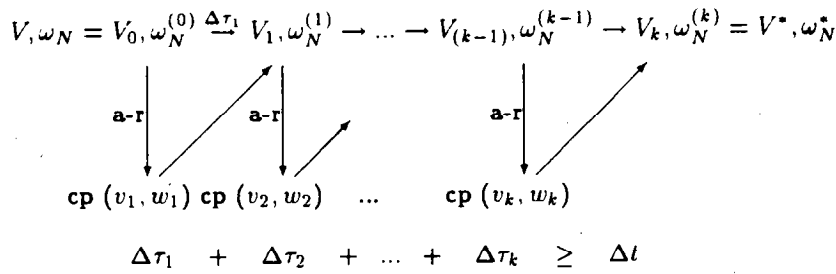
particle as a collision pair. This is not possible using permutations because the total set of particles is divided a priori into two disjoint sets; identical particle can not constitute a collision pair.

Furthermore it is clear that a higher (internal) cycle number $N_C \cdot \frac{N}{2}$ will give a better approximation for the spatially homogeneous Boltzmann equation, because the cycle number is coupled with the time discretization.

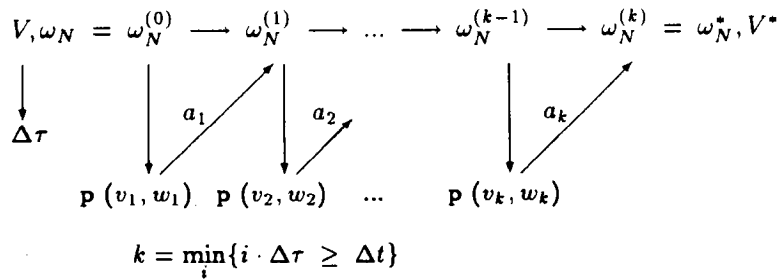
The disadvantage of the update approach is the efficiency on vector computers: it is not possible to perform the total collision procedure inside a vector register, because the actual velocities obey a recurrence relation.

The following diagrams explain the four different approaches schematically:

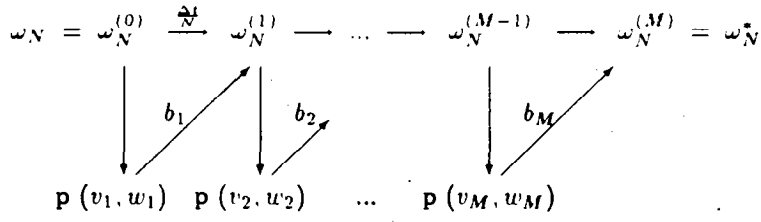
Bird's Time-Counter



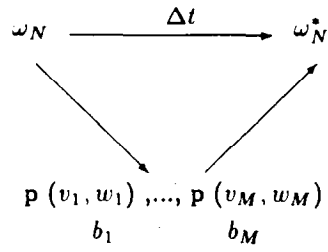
Bird's No-Time-Counter



FPM Update



FPM Permutation



Notations:

$\omega_N^{(i)} = \omega_N^{(i-1)} \setminus (v_i, w_i) \cup (v'_i, w'_i)$	$M = N/2$
$p(v_i, w_i) =$ fictitious collision pair	$cp(v_i, w_i) =$ real collision pair
$V_i = \max\{V_{i-1}, v_i - w_i \}$	$V^* = \max\{V_i, v_i - w_i , i = 1, \dots, k\}$
a-r = acceptance-rejection procedure	$\Delta\tau = \frac{C}{N \cdot V}$
$a_i = \frac{ v_i - w_i }{V}$	$b_i =$ transformed impact parameter

3 Numerical results

The first test case is a spatially homogeneous relaxation problem. We compare the influence of low discrepancy sequences and linear congruential generated sequences on the quality of the numerical solution and study the sensitivity of the numerical parameters like particle number and time discretization.

Then we investigate a twodimensional axisymmetric flow problem at high Mach numbers and try to compare the two proposed simulation methods.

3.1 Spatially homogeneous relaxation problem

The relaxation problem considered here is governed by the spatially homogeneous Boltzmann equation completed with a nonequilibrium distribution at time $t = 0$.

$$\frac{\partial f}{\partial t} = \frac{1}{\epsilon} J(f, f)$$

$$f(0, v) = \frac{\rho}{2(2\pi RT)^{\frac{3}{2}}} \left(\exp\left(-\frac{(v-u)^2}{2RT}\right) + \exp\left(-\frac{(v+u)^2}{2RT}\right) \right)$$

For the limit $t \rightarrow \infty$ the solution tends to an equilibrium distribution with prescribed density ρ and temperature T . The relaxation to equilibrium is forced by the collision operator $J(f, f)$ at the right hand side of the Boltzmann equation. The parameter ϵ only leads to a scaling of the instationary curves. Therefore in general there exists only one relaxation curve; all other curves can be constructed using a linear transformation in the time variable. Differences in the numerical methods can occur in the instationary behaviour of the solution as well as in the accuracy (i.e. the fluctuations) of the stationary state.

First we present results on the fluctuations of the moments of the different methods in the stationary state with respect to the required CPU-time. Then we compare the instationary curves given by the different methods. Furthermore we try to investigate the sensitivity of the numerical solution to the different parameters used in the different methods.

Algorithm	Type	Version	Pseudo-random numbers
<i>A</i>	DSMC	Time-Counter	Linear Congruential
<i>B</i>	DSMC	Time-Counter	Low-Discrepancy
<i>C</i>	DSMC	No-Time-Counter	Linear Congruential
<i>D</i>	DSMC	No-Time-Counter	Low-Discrepancy
<i>E</i>	FPM	Permutations	Linear Congruential
<i>F</i>	FPM	Permutations	Low-Discrepancy
<i>G</i>	FPM	Update	Linear Congruential
<i>H</i>	FPM	Update	Low-Discrepancy

Table 1: Test constellation

Table 1 shows the different constellations used in the comparison.

The first and most important parameter in simulation methods is the number of particles

used for the approximation of the exact solution. This number N determines the accuracy of the numerical solution. With N to infinity the numerics should converge to the exact one.

For a finite number of particles the fluctuations of independent samples around the averaged solution indicates the accuracy of the given approximation. The fluctuations are strongly influenced by the random numbers used in the single steps of the numerical method. Referring to chapter 2 for the actual homogeneous problem random numbers are used in the following steps of the calculation:

- (1) Approximation of the initial distribution function of the gas particles.
- (2) For every discrete timestep determine a potential collision pair and decide if a real collision occurs.
- (3) For every collision determine the post collisional velocities.

It is obvious that using low discrepancy sequences the quality of the approximation in step (1) will be better than using a standard linear congruential generator. Low discrepancy sequences always have the optimal order of approximation in the sense of discrepancy and an averaging procedure can be included by choosing uniformly distributed starting points (for a detailed description see ([1])).

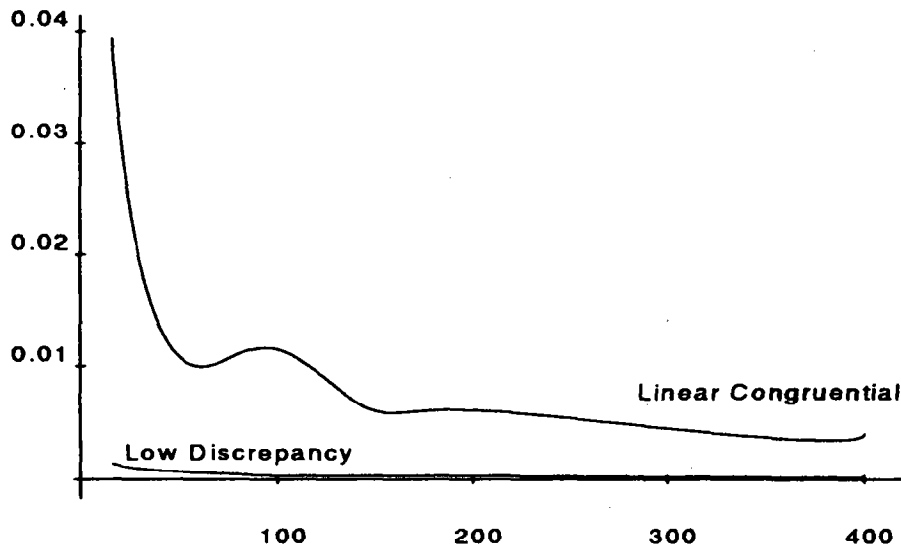


Fig. 1 : Approximation quality for the 2nd moment

In figure 1 the plotted curves show the fluctuations around the sampled value for the second moment in the v_x -direction of the distribution function $f(0, v)$ using low discrepancy sequences and standard random numbers.

The results demonstrate the superiority of low discrepancy sequences in this case.

During the time iteration the approximation quality is influenced by the random numbers which are necessary in step (2) and (3). After several timesteps all particles will have suffered a collision so that the error depends mainly on step (3) of the iteration procedure. The following results show the fluctuations around the averaged value using low discrepancy sequences in step (1) and (3) of the iteration versus standard random numbers. The fluctuations using low discrepancy sequences in step (1) and (3) are about 20% lower than using standard random numbers (see figure 2). Comparing the results with the required CPU-times (table 2) the gain in the CPU-time for a prescribed accuracy is about a factor of 2.

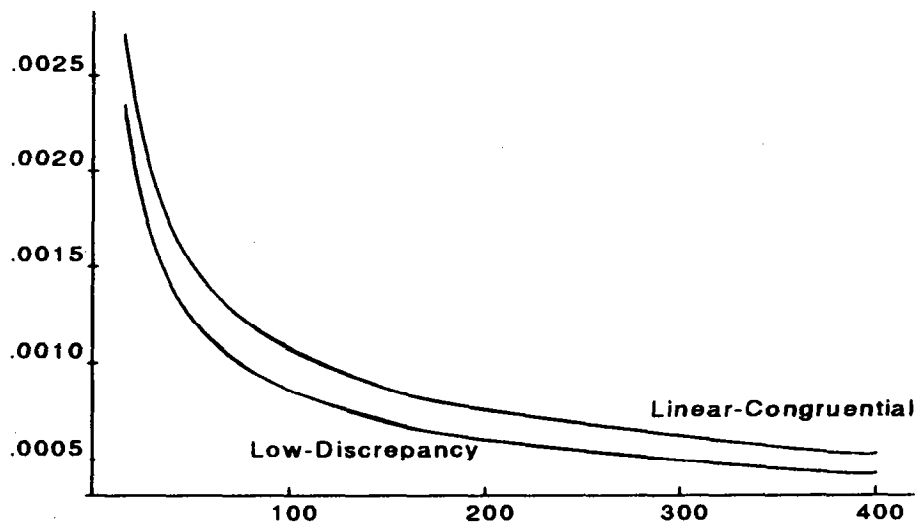


Fig. 2 : Fluctuations versus particle number

Algorithm	$N = 16$	$N = 64$	$N = 196$	$N = 324$
A	5.6	21.2	64.5	107.0
B	6.2	23.2	70.5	116.9
C	5.8	21.2	—	106.9
D	6.1	23.0	70.0	—

Table 2: CPU-time[s] versus particle number

The results shown above hold for all versions used in the comparison: DSMC Time Counter, DSMC No Time Counter, FPM with permutations and FPM with update. As mentioned earlier the second interesting part in a relaxation problem is the transition

to equilibrium. In the following we investigate the solution obtained by the different algorithms after the first time step. The sensitivity of the solution on the various parameters is the main part of the investigation.

The most important parameter in simulation methods is the number of points used to approximate the exact solution. The number of particles controls for example the amount of memory necessary for the computation. For multidimensional flows the number of particles per cell can not be extended too much because of the limited computer memory. Therefore the numerical results should not depend too strong on the number of particles per cell. The lower approximation quality with low particle numbers can be improved by increasing the number of averaging steps.

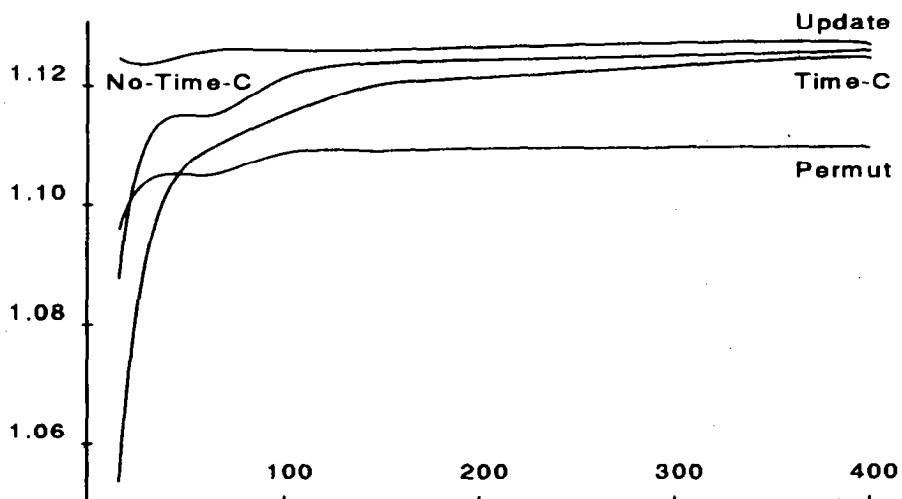


Fig. 3 : 2nd moment at time $t = \Delta t$ versus particle number

Figure 3 presents the dependence of the numerical solution on the number of particles used for the different simulation methods. The original Time Counter method turned out to be very sensitive on the number of particles. The No Time Counter approach shows also a dependence on the particle number as well as the Finite Pointset Method with permutations. Only the Finite Pointset Method with update procedure is nearly independent on the number of particles. Furthermore the Finite Pointset Method with permutation with a fixed number of cycles do not converge to the solution of the three other methods. The following table shows the number of particles necessary with the DSMC Time Counter resp. No Time Counter approach to be within a given error bound of the FPM with update algorithm.

Algorithm	$\Delta = 5\%$	$\Delta = 2\%$	$\Delta = 1\%$	$\Delta = 0.5\%$
A or B	≈ 24	≈ 64	≈ 100	≈ 196
C or D	≈ 16	≈ 36	≈ 64	≈ 100

Table 3: Particle number versus accuracy

The behaviour of the Finite Pointset Method with permutation can be explained by looking at the time discretization of the method. In contrast to the other three approaches this method is the only one working with the explicit time discretization Δt . DSMC as well as FPM with update can be characterized as 'semi implicit' time discretization schemes with variable step control. In the limit $N \rightarrow \infty$ the FPM with permutation still works with the given time step Δt , whereas all three other methods split the time increment Δt in small parts dependent on N ; one particle may have more than one collision during the time increment Δt .

Both methods, the DSMC as well as the FPM, include a further parameter, which can influence the numerical solution.

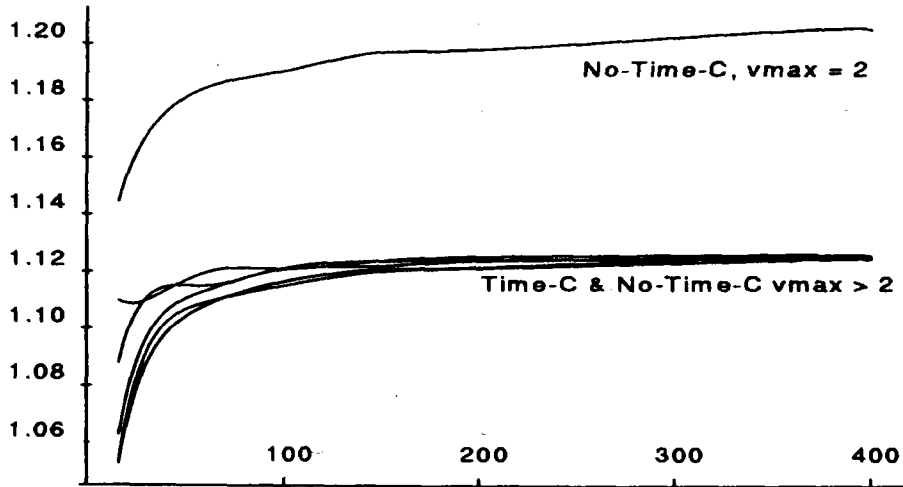


Fig. 4 : DSMC solution versus parameter V

In the case of the DSMC the parameter is the modulus of the maximal relative velocity vector in a collision. By including this parameter in the DSMC method Bird overcomes the $O(N^2)$ effort. In the Time Counter approach the parameter V is fixed at the beginning of the computation. In every collision this parameter is modified if the actual value is greater

than V . Furthermore the expression for the time counter depends on V .

In the No Time Counter approach the parameter V is fixed over one iteration step and modified only after a complete iteration. The time counter depends only on V and remains constant over the whole timestep.

Figure 4 shows that both methods are not sensitive against different starting values for V . On the other hand the CPU-time increases if the starting value increases.

Algorithm	$V = 2$	$V = 4$	$V = 6$
A	130.9	132.2	147.8
C	129.9	132.4	149.0

Table 4: CPU-time[s] versus V

This behaviour is obvious because the parameter V is the parameter for the acceptance-rejection procedure in the DSMC method and therefore the probability for the acceptance of a given collision pair decreases with increasing V .

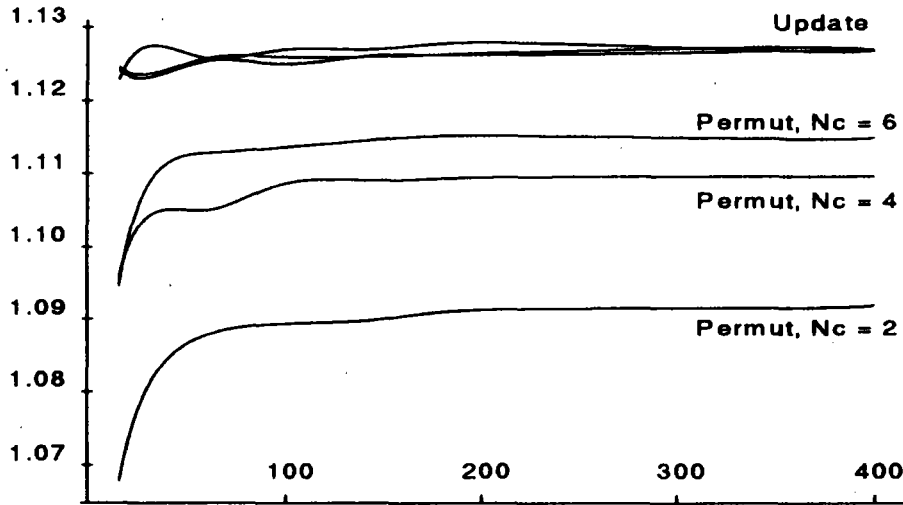


Fig. 5 : FPM solution versus cycle number N_C

The collision process in the Finite Pointset Method is described by the number of cycles N_c used for dividing the given time discretization step Δt into smaller parts with length $\Delta \tau = \frac{\Delta t}{N_c}$. For the FPM with permutation this parameter exactly presents a change in the explicit time parameter Δt by a factor of N_c . Therefore one can expect that the solution will depend especially on this parameter. Figure 5 shows the results for the FPM with

permutation and with update. The solution for the FPM with update does not depend on the parameter N_c .

Finally figure 6 shows some typical curves for the fluctuations during the transition to equilibrium.

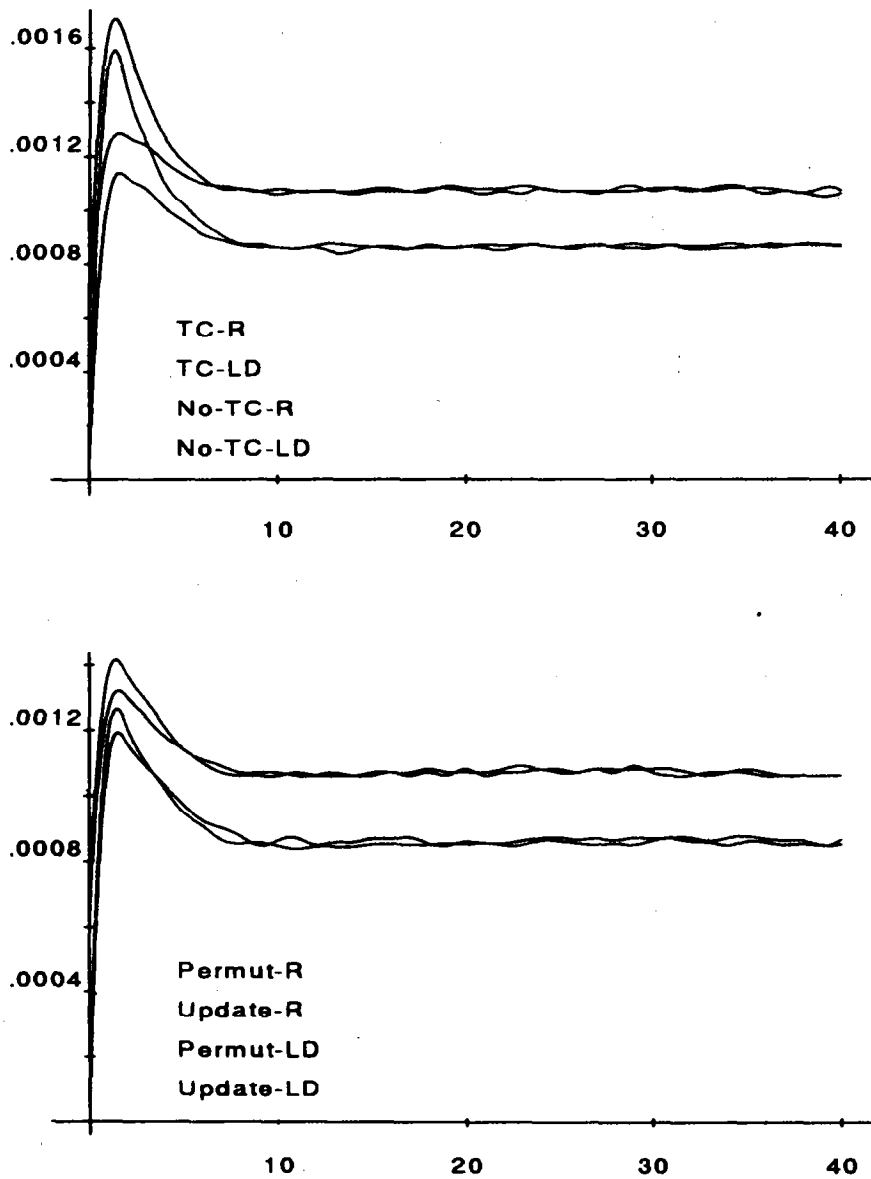


Fig. 6 : Fluctuations during transition to equilibrium

3.2 Twodimensional axisymmetric flow problem

The twodimensional axisymmetric flow problem was already investigated in connection with the development of an axisymmetric flow code ([13]). The body is defined by the HERMES windward centerline at an angle of attack of 30° degree. The comparison was done by exchanging the different collision routines within the existing code.

The main interest of this comparison is to investigate the influence of the number of particles per cell on the numerical solution and to compare the different collision routines within a given environment. Focusing on these aspects only standard random number generators were used in the twodimensional flow problem, except for the approximation of the initial condition and the boundary conditions at the artificial boundaries.

All the calculations were done on a nCUBE 2S parallel machine with 64 nodes. The parallelized code is based on the work of Struckmeier and Pfreundt ([12]). Because the nCUBE 2S parallel computer has no vector architecture the better performance of the FPM with permutations compared with the FPM with update resp. the No Time Counter scheme compared with the Time Counter approach cannot be investigated in this paper.

Global and local surface quantities, like drag or heat coefficient, are the main quantities in the description of the aerodynamic behaviour of a space vehicle. The numerical results should be as accurate as possible in order to study the influence of real gas effects like vibrational energy or chemical reactions on the aerodynamic characteristics.

For that reason we confine the comparison to the global surface quantities of the given axisymmetric body.

The main results obtained from the relaxation problem was the strong dependence of the DSMC Time Counter method on the number of particles used in the calculation. Furthermore the influence of the explicit time discretization in the FPM with permutation was an evident result in former investigation.

The following figures present results for the drag coefficient and the heat transfer coefficient in dependence on the number of particles per cell. Other surface quantities like pitching moment show the same behaviour.

Table 5 shows the corresponding CPU-times.

Algorithm	$N = 36$	$N = 64$	$N = 196$	$N = 400$
DSMC TC	24 : 49	42 : 50	128 : 50	262 : 16
DSMC NTC	24 : 26	42 : 27	128 : 39	264 : 03
FPM P	26 : 43	45 : 40	138 : 33	281 : 54
FPM U	26 : 23	45 : 51	138 : 59	286 : 04

Table 5: CPU-time[min] versus particle number

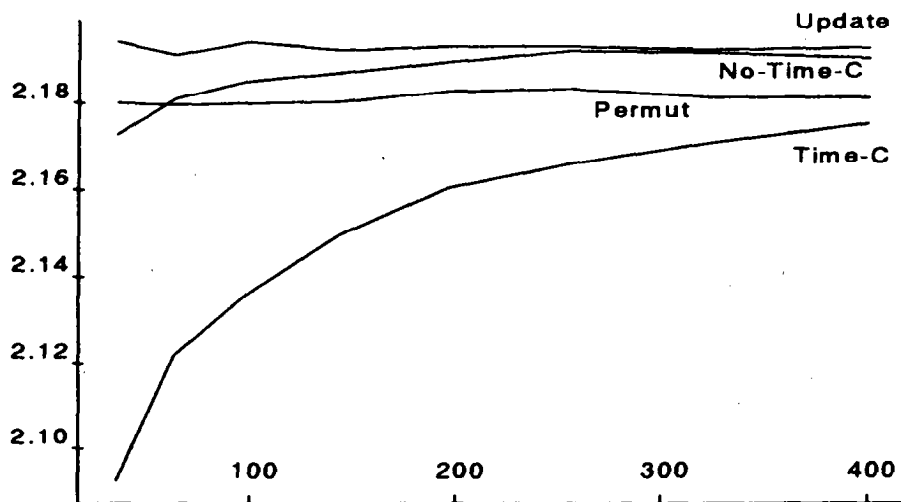


Fig. 8a : Drag coefficient versus particle number

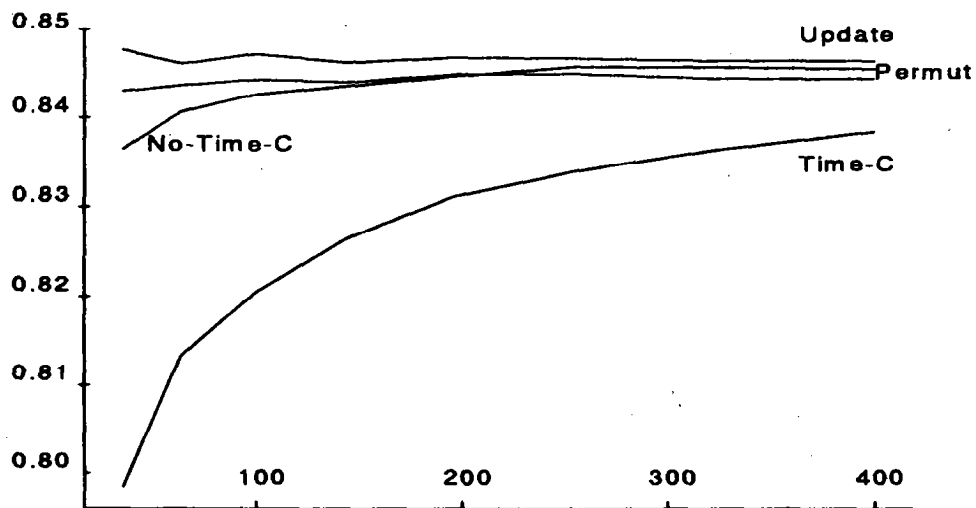


Fig. 8c : Heat transfer coefficient versus particle number

4 Conclusion

Although simulation methods for rarefied gas flows look very similar differences in the approximation quality can be detected. Nevertheless the DSMC approach as well as the FPM were several times proofed against 'real' windtunnel measurements.

The differences discovered in the current paper can be mainly characterized by the sensitivity of the numerical results on the parameters used for the computation like the particle number or the time discretization parameter. Only the FPM with update is independent on the numerical parameters. The FPM with permutations is an explicit scheme in time, such that the numerical results depend on the particle number and the time discretization parameter. The DSMC method – especially the Time Counter version – turned out to be very sensitive on the particle number. On the other hand the time discretization parameter only influences the computational effort. A further theoretical investigation on the various modifications of the Time Counter approach of the DSMC method seems to be necessary.

The use of low discrepancy sequences lead to a higher accuracy of the numerical results in all methods. Nevertheless a further research on this topic is necessary to validate this result.

References

- [1] Babovsky, H.: *On a Simulation Scheme for the Boltzmann Equation*, Math. Methods in the Applied Sciences 8 (1986), pp. 223-233.
- [2] Babovsky, H.: *A Convergence Proof for Nanbu's Boltzmann Simulation Scheme*, European J. of Mech. B/Fluids, 8, no. 1 (1989), pp. 41-55.
- [3] Babovsky, H., Illner, R.: *A Convergence Proof for Nanbu's Simulation Method for the Full Boltzmann Equation*, SIAM J. on Numerical Analysis, Vol. 26, No. 1, (1989), pp. 45-65.
- [4] Billingsley, P.: *Convergence of Probability Measures*, John Wiley, New York (1968).
- [5] Bird, G.A.: *Molecular Gas Dynamics*, Clarendon Press, Oxford (1976).
- [6] Bird, G.A.: *Perception of Numerical Methods in Rarefied Gasdynamics*, in: Muntz, E.P.; Weaver, D.P.; Campbell, D.H. (Eds.), Progress in Astronautics and Aeronautics: Rarefied Gas Dynamics: Theoretical and Computational Techniques, vol. 118 (1989), pp. 221-226.
- [7] Lecot, C.: *A Direct Simulation Monte Carlo Scheme and Uniformly Distributed Sequences for Solving the Boltzmann Equation*, Computing 41 (1989), pp. 41-57.
- [8] Nanbu, K.: *Direct Simulation Scheme derived from the Boltzmann Equation*, Journal of Physics Japan 49 (1980), pp. 2042-2049.
- [9] Neunzert, H.; Gropengießer, F.; Struckmeier, J.: *Computational method for the Boltzmann equation*, in: Spigler, R.(Ed.), Applied and Ind. Mathematics, Venice-1, 1989, Kluwer Acad. Publ., Dordrecht (1991), pp. 111-140.
- [10] Neunzert, H.; Struckmeier, J.: *The Finite-Pointset-Method for hypersonic flows in the rarefied gas regime*, in: Bertin, J.J; Glowinski, R.; Periaux, J. (Eds.), Hypersonics, Vol. III, Birkhäuser Boston (1989)
- [11] Struckmeier, J.: *Fast Generation of Low Discrepancy Sequences*, to appear
- [12] Struckmeier, J.; Pfreundt, F.-J.: *On the efficiency of simulation methods for the Boltzmann equation on parallel computers*, Parallel Computing 19 (1993), pp. 103-119.
- [13] Struckmeier, J.; Steiner, K.: *Boltzmann simulations with axisymmetric geometry*, Berichte der Arbeitsgruppe Technomathematik, Nr. 83 (1992), to be published.
- [14] Wagner, W.: *A convergence proof for Bird's direct simulation Monte Carlo method for the Boltzmann equation*, J. Stat. Phys.,
- [15] Yakowitz, S.J.: *Computational Probability and Simulation*, Addison Wesley, Massachusetts (1977).