41st AIAA Thermophysics Conference
22–25 June 2009, San Antonio, TX

AIAA #2009–3742

Computational Experiments with Direct Simulation Quasi-Random Monte Carlo

José A. Camberos* and Robert B. Greendyke†
United States Air Force
Dayton, OH 45433

Larry A. Lambe‡
Multidisciplinary Software Systems
Research Corporation, Bloomingdale, IL 60108

Brook I. Bentley§
Air Force Institute of Technology
Dayton, OH 45433

The equations derived in kinetic theory express a desired quantity in terms of a probability density. The solution to these equations often requires computational techniques. For example, the Direct Simulation Monte Carlo (DSMC) method is a well-known powerful technique for computational rarefied gas dynamics. DSMC uses an algorithm that evolves an initial distribution in time using random sampling. The sampling is achieved classically through the use of a pseudo-random number generator. Alternatively, Quasi-Monte Carlo methods (QMCMs) can replace calls to a pseudo-random number generator by calls to a quasi-random number generator. QMCMs are known to have better convergence rates than Monte Carlo methods for high-dimensional integration, but it is not trivial to make QMCM work well in contexts outside of Monte Carlo integration, such as DSMC. In fact, naïve replacement of calls to a pseudo-random number generator by calls to a quasi-random generator can fail utterly. In a previous study, we illustrated these difficulties and discussed possible means to overcome them. In context of DSMC, we conclude that little can be gained through use of quasi-random sequences. In this work, we present results for an approach with an experimental method that uses a smoothed accept/reject stage.[1–3] Also, in the context of "direct methods" we find promising results.

Introduction: DSMC Background

We continue our report on a series of experiments with algorithms used for approximating solutions to the Boltzmann equation.[4–7] We briefly review the Discrete Simulation Monte Carlo (DSMC) method of Bird[5] as given by Garcia[8] and also revisit the subject of quasi-random number generators (QRNGs) as compared to pseudo-random number generators (PRNGs). Next we describe our experiments in which the PRNGs in Bird's programs[9] were replaced by quasi-random number generators QRNGs and review the results.

Bird[5] applied Monte Carlo methods to gas dynamics in the mid 1960s with an algorithm he named *Direct Simulation Monte Carlo* (DSMC). In Section 10.2 of his book, he explained that the algorithm was based on physical reasoning analogous to the derivation of the Boltzmann equation. The main idea of DSMC is to construct a stochastic process for a many-particle system that evolves in time. The process begins with an initial particle distribution ($t = 0$) and for $t > 0$ models gas dynamics realistically in the way the particle distribution evolves. Wagner[10] recently proved that Bird's method converges to a solution of the Boltzmann equation in a suitable limit.

Alternatively, Nanbu[11,12] derived a simulation procedure directly from the Boltzmann equation. A modification of this algorithm was given by Babovsky[13] and Nanbu's method was shown to converge[14] to a solution of the Boltzmann equation as the number of particles is allowed to grow indefinitely (and the solution is sufficiently smooth). The Kac N-particle equation[15] was used as the basis for another class of algorithms by Ivanov et al.[16,17] in the spatially uniform case. The authors modeled the Leontovich equation in the spatially non-uniform case. For more discussion of the relationship between the equations just mentioned and the Boltzmann equation, see the references just cited along with Nanbu's 1983 paper.[18]

*Research Aerospace Engineer, U. S. Air Force Research Lab, AIAA Associate Fellow (jose.camberos@us.af.mil)
†Associate Professor, Air Force Institute of Technology, AIAA Associate Fellow (robert.greendyke@afit.edu)
‡Chief Scientist, MSSRC (llambe@mssrc.com)
§1st Lt. USAF, Graduate Student, Air Force Institute of Technology, Member AIAA

Standard DSMC Algorithm

All the methods mentioned above have been implemented using PRNGs[19] and several authors have studied the effects of replacing the PRNGs in various schemes for approximating solutions to Boltzmann's equation in various special cases.[20–24] We are interested in investigating the effect of adapting Bird's algorithm to the quasi-random setting. Here we follow Garcia's description of DSMC.[8]

The region (finite volume of space) to be considered is partitioned into cells. Initially N particles are distributed in a uniformly random way throughout the system with a predetermined number particles in each cell. Each particle in the simulation represents a number N_a of actual particles in the physical system. In addition to its initial position $\mathbf{r_i}$, each particle is also given an initial velocity $\mathbf{v_i}$ according to the time $t = 0$ (initial) distribution. Particle evolution is carried out in discrete time steps Δt in two phases, advection and collision.

In the advection phase, particles move without interaction so that their positions are updated using the formula $\mathbf{r_i} \mapsto \mathbf{r_i} + \Delta t \mathbf{v_i}$. If a particle reaches a boundary it is processed according to appropriate boundary conditions. After all particles have moved, a given number are randomly selected for the collision stage. Only particles within the same cell are allowed to interact and a set of collisions is processed at each time step. All pairs of particles are candidates for collision regardless of their positions within the cell. A collision probability can be derived depending on the model. As is usually the case however it is difficult or expensive (or both) to sample this distribution directly. Instead, an accept/reject[25] procedure is implemented. If the pair is selected, the new positions and velocities are calculated as explained below. If the pair is rejected, the algorithm repeats the second step above. The calculation continues until the required number of candidate pairs has been processed. The post-collision velocities are computed using conservation of linear momentum and energy. This gives an incomplete set of equations that is closed by choosing two uniformly distributed random angles over a unit sphere. The required number of candidates is determined using the ratio of the total of accepted candidates to the total number of candidates.

Note that particle collisions are determined by a sample drawn from a discrete probability space (the space of all pairs of particles in a given cell) while the random angles over the sphere are drawn from a continuous uniform distribution.

Quasi–Random Monte Carlo

We begin with a brief description of a classic quasi-random number generator and then give two examples of the effect of adapting standard Monte Carlo methods to use quasi-random number generators.

In 1935 van der Corput[26] constructed a sequence that distributes numbers very uniformly in $[0, 1]$. For a non-negative integer n and digits $d_i \in \{0, 1\}$, write

$$n = d_0 + d_1\, 2 + d_2\, 2^2 + \ldots + d_k\, 2^k. \tag{1}$$

Let

$$\Phi_2(n) = \frac{d_0}{2} + \frac{d_1}{2^2} + \ldots + \frac{d_k}{2^{k+1}}. \tag{2}$$

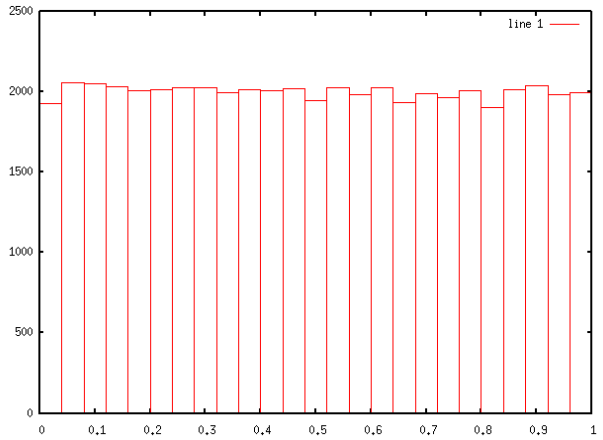
This can be done for any "base" b: For digits $\delta_i(n) \in \{0, \ldots, b-1\}$, write

$$n = \sum_{i=0}^{k} \delta_i(n)\, b^i \tag{3}$$

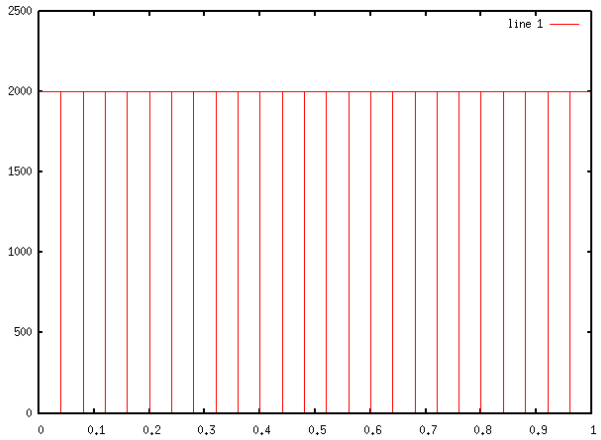
and let

$$\Phi_b(n) = \sum_{i=0}^{k} \delta_i(n)\, b^{-i-1}. \tag{4}$$

These sequences are called quasi-random or *low discrepancy* sequences.[27] There are many types of quasi-random sequences. For now, the ones described above are the only kind we need consider. Quasi-random sequences are generally much more uniform than pseudo-random sequences. This property is made quite precise in the book by Niederreiter.[27] For example, using the built-in function rand() from the `gcc` compiler, Figure(1) compares a pseudo-random sequence with a van der Corput sequence, base three, using histograms. In two dimensions, a pseudo-random sample of $10,000$ points in the unit square is compared with a quasi-random sample of $10,000$ points in the same domain using a Halton sequence[28] in Figure(2).

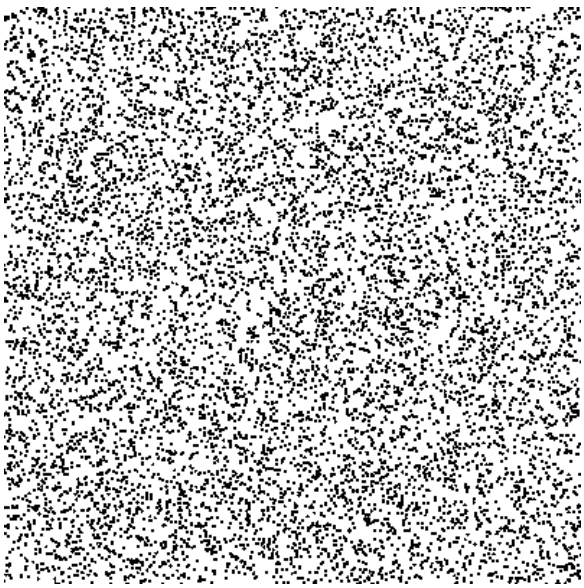


a) Histogram of a sample of $5,000$ pseudo-random points.

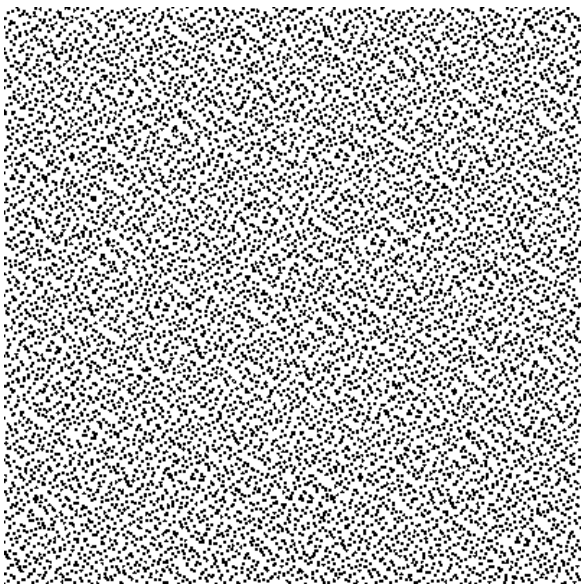


b) Histogram of a sample of $5,000$ base 3 van der Corput points..

Figure 1 Comparison of standard Monte Carlo sampling and quasi-random Monte Carlo when simulating sampling from a uniform distribution.



a) Pseudo-random points tend to clump (not optimal).



b) Quasi-random LDS (Halton Sequence) produces a much more uniform distribution of points.

Figure 2 Comparison of pseudo-random and quasi-random sequences in the unit sphere. The Halton sequence (b) generates a much more uniform distribution.

Quasi–Random Monte Carlo Integration

Recall[29] that the Monte Carlo integration of a function

$$f : [0,1]^s = [0,1] \times \ldots \times [0,1] \rightarrow \mathbb{R} \tag{5}$$

is an approximation of the form

$$\int_{[0,1]^s} \approx \frac{1}{N} \sum_{i=1}^{N} f(\bar{x}_i) \tag{6}$$

where $\{\bar{x}_1, \ldots, \bar{x}_N\}$ is a sequence of random elements of the hypercube $[0,1]^s$. The van der Corput sequence can be generalized to any dimension s by simply taking[28]

$$\Phi(n) = (\Phi_{p_1}(n), \ldots, \Phi_{p_s}(n)) \tag{7}$$

where $p_1, \ldots, p_s$ are the first s prime numbers. There are many other ways to obtain quasi-random sequences in dimension s.[27]

For example, consider the integral

$$I = \int_0^1 \int_0^1 \sin(x\,y)\,dx\,dy. \tag{8}$$

One may verify[30] that

$$I = \int_0^1 \frac{-\cos(x\,y)}{y}\,dy = -Ci(1) + \gamma \approx 0.2398117420005647259. \tag{9}$$

Let $\{\bar{r}_1, \ldots, \bar{r}_N\}$, be a pseudo random sample of the unit square $[0,1]\times[0,1]$ and $\{\bar{x}_1, \ldots, \bar{x}_N\}$ be a quasi-random sample. Let

$$I_{pr}(N) = \frac{1}{N}\sum_{i=1}^{N} f(\bar{r}_i), \quad \text{and} \quad I_{qr}(N) = \frac{1}{N}\sum_{i=1}^{N} f(\bar{x}_i). \tag{10}$$

We now study the sequences of errors generated with increasing sample points,

$$I_{pr}(10) - I, I_{pr}(20) - I, \ldots, I_{pr}(5000) - I, \quad \text{and} \quad I_{qr}(10) - I, I_{qr}(20) - I, \ldots, I_{qr}(5000) - I.$$

A useful way to visualize the result is to graph the points

$$\{(10, I_{pr}(10) - I), \ldots, (5000, I_{pr}(5000) - I)\} \quad \text{and} \quad \{(10, I_{qr}(10) - I), \ldots, (5000, I_{qr}(5000) - I)\},$$

and interpolating between the points in each case. Figure(3) shows the result. The reader may note that since the dimension $s = 2$ is small in this case, a fine mesh might be used along with a good quadrature to get an approximation better than the one obtained using pseudo-random sampling; however, the same sort of comparison in the error between Monte Carlo integration and Quasi-Monte Carlo integration persists to higher and higher dimensions because of the following well known inequality for reasonably behaved functions:[27]

$$\left| \frac{1}{N}\sum_{i=1}^{N} f(\bar{x}_i) - \int f(\bar{x})\,d\bar{x} \right| \leq V \cdot D_N \tag{11}$$

where V is a constant depending only on f and D_N is a quantity (called the *discrepancy*) depending on the quasi-random sequence $\{\bar{x}_1, \ldots, \bar{x}_N\}$. This inequality is called the *Koksma–Hlwaka* inequality and full details may be found in the book by Niederreiter.[27] We note here that because of this inequality, lower discrepancy results in better approximations to the integral. One might expect that quasi-random numbers may be useful in speeding up or otherwise increasing the quality of other Monte Carlo methods, however, we will see that there are some subtleties in achieving such adaptations.

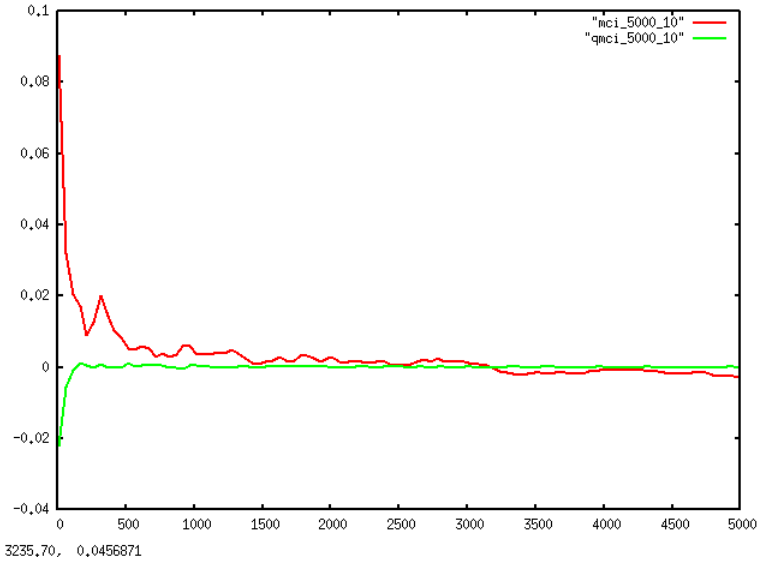
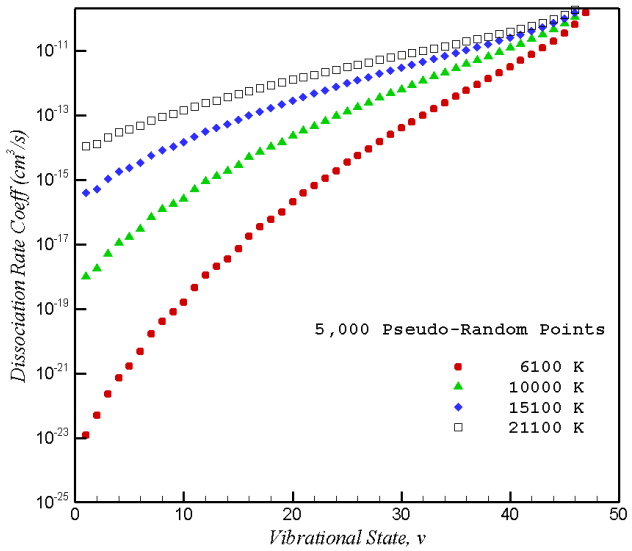
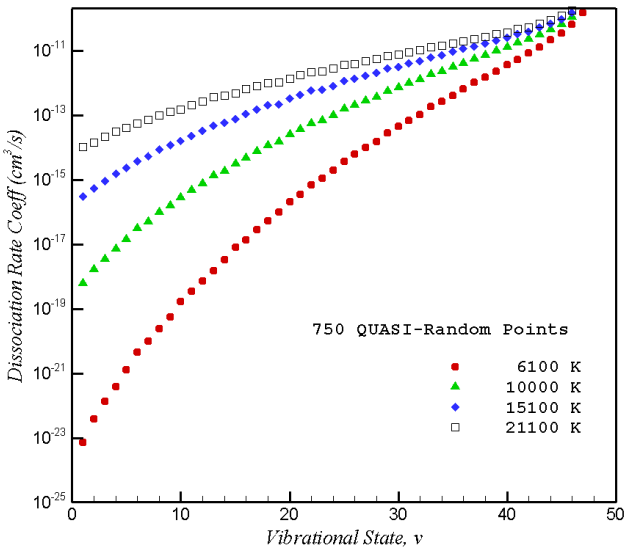


Figure 3 Monte Carlo integration error (red line approaching from above) vs Quasi-Monte Carlo integration error (green line approaching from below. The Quasi-Monte Carlo integration error converges quickly and monotonically improves with more points. The standard MC integration error fluctuates with no guarantee that adding more points will improve accuracy.



a) Standard Monte Carlo Method. The results shown here for the dissociation rate of nitrogen in modeling $N_2 - N_2$ collisions are assumed to be converged, using 5,000 sample points.



b) Quasi-Random Monte Carlo Method. The results shown here for the dissociation rate of nitrogen in modeling $N_2 - N_2$ collisions are of equal quality to the converged solution, using only 750 sample points.

Figure 4 The Quasi-Random Monte Carlo method yields results that are comparable with the best (converged) results using the standard method. Here, only 750 quasi random are used to obtain accuracy comparable to that obtained with 5,000 points using pseudo-random Monte Carlo.

A More Complicated Example

To explore the potential benefits of using quasi-random numbers in realistic applications, we implemented a procedure for obtaining the dissociation rates for state-to-state chemical kinetics in joint work of Camberos, Josyula, and Lambe[31] a part of which we will summarize here.

State-specific dissociation rates coefficients of $N_2 - N_2$ collisions were calculated based on the semiclassical theory of Macheret and Adamovich[32] who adapted a FORTRAN program by Billing.[33] We adapted the program by replacing calls from a pseudo-random number generator to a specially adapted eight dimensional quasi-random number generator. Figure (4) shows that the Quasi-Monte Carlo adaptation using only 750 samples compares well with the Monte Carlo method using 5000 pseudo-random samples. Numerically, the norm of the difference of the output is 0.0000000000309.

Experiments with DSMC

Results with DSMC and Quasi-Random Sequences (DSQMC)

In the late 1980s and early 1990s a number of authors investigated the effect of adapting Nanbu's algorithm to the quasi-random setting. Results by Babovsky et. al.[20] and Lécot[34] were fairly elaborate and the results were apparently not as striking as the difference that quasi-random sequences make with respect to Monte Carlo integration. In fact Bird comments on the work by Babovsky et. al. in Chapter 10 of his book.[5] A much more recent and successful attempt to adapt stochastic methods for gas dynamics to the quasi-random setting is the work of McNenly and Boyd.[23,24] McNenly studies the special case of collisionless flow and derives an algorithm that achieves near linear error convergence rate using quasi-random sequences. We decided to begin a sequence of experiments involving the full algorithm of Bird in this context. Our first results were a bit surprising and we reported them in some detail in our first paper.[35]

Scrambling

In our first experiment, we simply replaced all of the calls to the PRNG by calls to a van der Corput sequence. Not only was there no improvement in this case, *the resulting algorithm did not converge*! We actually anticipated this result however. It is due to a well-known problem in trying to apply quasi-random sequences in Monte Carlo algorithms which we believe was first reported by Morokoff and Caflisch in 1993[36] in studying quasi-random random walk methods for the heat equation. This problem is due to the following number theoretic fact concerning the van der Corput sequence base two. One has

$$\Phi_2(2n) < \frac{1}{2}, \;\; \Phi_2(2n+1) \geq \frac{1}{2}. \tag{12}$$

These inequalities doom to failure any naïve attempt to replace a PRNG by the van der Corput sequence or any other quasi-random sequence that has hidden correlations[37] built-in due to the mathematical nature of the algorithm used to implement them.

A general way to fix such problems is to *scramble* the quasi-random sequence – a procedure that goes at least back to Cranley and Patterson[38] in one form (1976) and Braaten and Weller[37] in another form (1979). The second form was considerably generalized by Owen.[39] A detailed analysis of these procedures is beyond the scope of this paper; we simply give a brief description of one form of scrambling. Recall that the building blocks of the van der Corput sequence base b are the digits $\{0, \ldots, b-1\}$ that make up the given integer n base b. Suppose that there are β such digits in the expansion of n. Instead of constructing the number in $[0, 1]$ as van der Corput does, scramble the digits first. That is, choose β permutations $\sigma_1, \ldots, \sigma_\beta$ of the digits *randomly* and apply them to each digit in order before doing the construction. Note that there are $b! \cdot \beta$ different possibilities for each n using this scheme just in dimension one! Note also that this is the first time that probability theory has entered the subject of quasi-random numbers. As will be seen from our second experiment below, this addition of "randomness" to quasi-random sequences is enough for the algorithm to converge once more.

The first two places where the PRNG is called in Bird's code require three calls. The first two are for selecting the two random particles as collision partners and the third involves the accept/reject inequality. In the quasi-random setting, this should be interpreted as a three-dimensional process. The reason for this is what is known as the *fundamental theorem of simulation* as it is called in the book by Christian and George,[25] viz. choosing a sample x according to a given distribution ρ is equivalent to conditionally sampling a pair (x, u) *uniformly* such that $u < \rho(x)$. In fact, this theorem leads directly to the classic accept/reject algorithm.

We replaced the three PRNG calls mentioned above with a scrambled three-dimensional quasi-random sequence for the accept/reject part of the algorithm. Next, we replaced the two PRNG calls to for the two post-collision deflection angles with a scrambled two-dimensional quasi-random sequence. This resulted in a computation that converged, but with only very slight improvement in the rate of convergence. A possible reason for this is described in the next Section.

Smoothed Accept/Reject

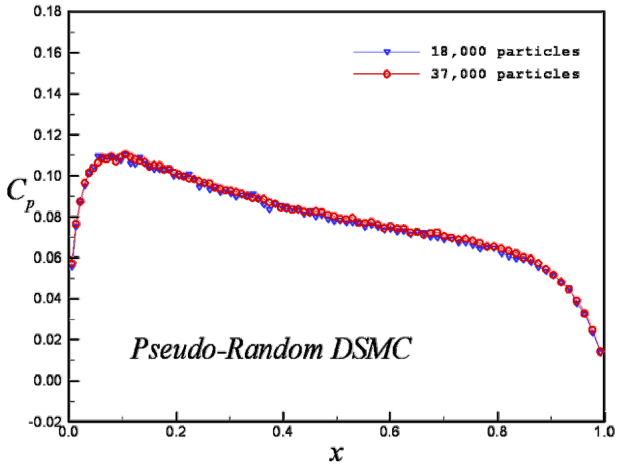
It has been observed[1–3] that the accept/reject algorithm for drawing a sample according to a given distribution corresponds to a discontinuous integrand and as such, the Koksma–Hlwaka does not apply. However, Moskowitz and Caflisch[3] showed that the integrand could be *smoothed* so that the Koksma–Hlwaka does apply to the new situation. This is done at the expense of introducing weights for the samples drawn. McNenly[23] also observed the inherent discontinuities of accept/reject and found a successful adaptation of a Monte Carlo method for the special case of free-molecule gas flow he studied by eliminating the accept/reject stage using another algorithm; however that method does not adapt to the general case.

Weighted particle methods for approximating the Boltzmann have already been introduced by Rjasanow and Wagner.[40–43] Since it was fairly trivial to adapt our previous code to the setting of smoothed accept/reject, we decided to run this sort of experiment as well. There was an issue about how the weights should evolve in Bird's approach and we formulated an *ad hoc* algorithm for doing this. We make no claim that our ad hoc algorithm is related to the work of Rjasanow and Wagner just cited.

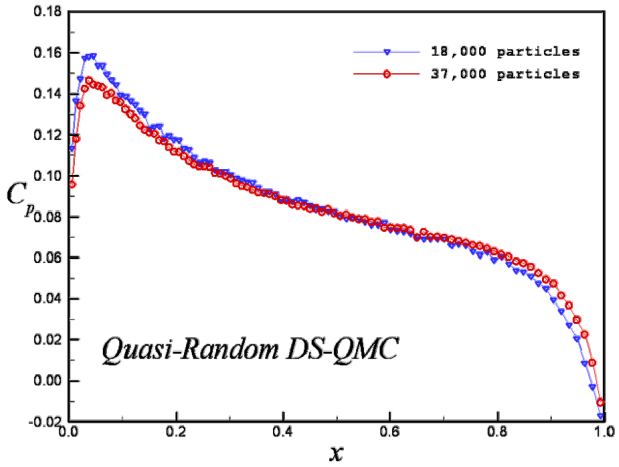
Previously, we compared the results of calculations using the supersonic leading-edge problem as described by Bird[5] pp. 340-348, with flow conditions set to simulate the supersonic flow of nitrogen gas at Mach 4. Convergence rates were not significantly improved. However, the algorithm was different enough from Bird's method to yield slightly different solutions for the pressure distribution (Figure 5) along the plate as well as heating (Figure 6(b)) and shearing (Figure 6(a)). Without experimental or other validated data for comparison, the result was inconclusive.

Our current effort sought to study a "Smoothed Accept/Reject," or SAR, model for collisions among simulated particles in the DSMC method. This new technique was observed to improve the accuracy of the DSMC method for calculating one dimensional shock structures in Argon for a variety of Mach numbers when compared to experimental data, which we will discuss below. We expect the method will extend to multi-dimensional codes and plan to compare further against any existing experimental data.

Our overall objective sought to improve the accuracy of the DSMC method by changing the standard acceptance/rejection (AR) criteria for intermolecular collision processes intrinsic to DSMC. The work presented here demonstrates an improvement in comparison to experimental data for a one-dimensional shock wave at varying Mach numbers. The results are offered as proof-of-concept and represent just one simple change to the conventional algorithm by the using a "smoothed accept/reject" (SAR) procedure – not a major revamping of the DSMC methodology. This is a direct outgrowth of initial studies by the authors[35] to accelerate the convergence and reduce the statistical variance of the DSMC method by replacing the PRNG function calls with LDS.

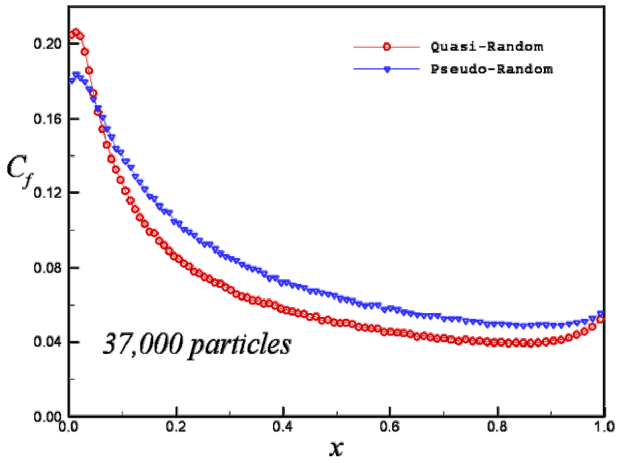


a) Pressure distribution along the plate with conventional DSMC.

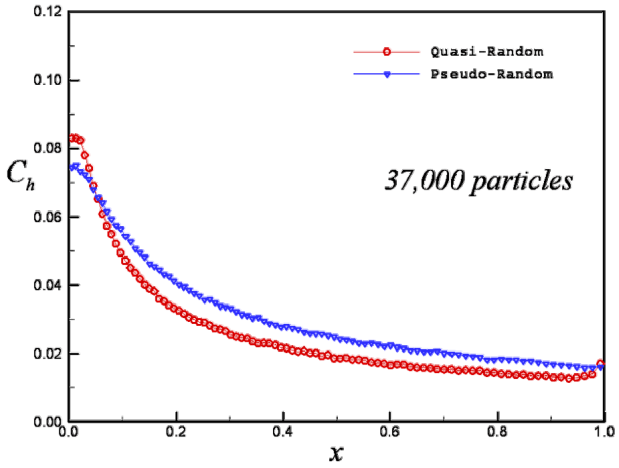


b) Pressure distribution along the plate with quasi-random DSQMC

Figure 5 The pressure distributions along the plate show some difference when comparing conventional DSMC with the quasi-random DSQMC results.



a) Comparison of friction coefficient with conventional DSMC and quasi-random DSQMC.



b) Comparison of heat transfer coefficient with conventional DSMC and quasi-random DSQMC

Figure 6 The friction and heat transfer coefficients along the plate show some difference when comparing conventional DSMC with the quasi-random DSQMC results.

Briefly for review, the core of the DSMC method is a (simulated) stochastic process for a many-particle system that evolves over time. The process begins with an initial distribution of particles at $t = 0$ about baseline values in terms of velocities and energies. For all times after the initialized values, the system of particles is allowed to evolve through a decoupled algorithm per time step of (simulated) collisions, followed by particle motion, advancement of the time step and then more collisions and particle motion, etc., until final convergence in a manner that "simulates" the behavior of the Boltzmann equation – hence the method's name: Direct Simulation Monte Carlo. Bird[5] notes that:

> simulation methods continue to suffer criticism for having a physical rather than a mathematical foundation. In reply to this criticism, it can be pointed out that the Boltzmann equation was not 'handed down carved in stone'

As noted previously, Wagner[10] has demonstrated that the DSMC method does indeed converge to the solution of the Boltzmann equation within an appropriate limit. The DSMC method at present uses pseudo-random number generation (PRNG) sequences of numbers to calculate the random numbers necessary for stochastic modeling of molecular simulation process including particle sampling, collision mechanics and post-collision velocity vector orientation.

The basics of the Bird algorithm[5] for collision calculation are fairly simple. Elementary kinetic theory is used to determine a collision frequency for each cell in the user defined Δt-time step. This per-cell frequency determines the number of simulated particle-pair collisions for the time interval. The probability of a pair of particles colliding in that time period is then given by:

$$P = \frac{F_N \sigma_T c_r \Delta t}{V_c} \tag{13}$$

where F_N is the ratio of actual particles per simulated particle, σ_T is the relative collision cross section for the particle pair, c_r is the magnitude of the relative velocity vector for the pair, and V_C is the volume of the computational cell. The maximum value of this parameter is computed and stored for each cell. Pairs of molecules are then selected randomly and compared to the maximum value for collision accept/reject criteria wherein the collision is accepted and computed if:

$$N_{\text{random}} < \frac{\sigma_T c_r}{(\sigma_T c_r)_{\max}} \tag{14}$$

where "N_{random}" is a random number obtained by either PRNG or LDS methods.

Our previous efforts sought to use LDS sequences of random numbers in place of the PRNG. As we noted above, LDS numbers are more uniformly distributed in their domain and we hoped that this would endow the DSMC sampling with a more uniform (smooth) distribution and reduce the statistical scatter in the sampled particles. The reduced scatter would then yield a faster overall convergence in the simulated flow-field. The LDS sequences are provably superior in generic Monte Carlo integration algorithms. Unfortunately, with the exception of one velocity component, the use of LDS only matched the performance of the conventional use of PRNG in DSMC methods in terms of convergence rates and variance. The flow-field solution for our test case yielded no difference within acceptable limits between the PRNG and the LDS use in the DSMC method. In order to improve the performance of the LDS, we thus attempted the construction of a modified "stochastic weighted-particle method" as inspired by Rjasanow & Wagner.[44]

Preliminary Work and Proof-of-Concept

The stochastic weighted-particle method suffers from a curious complexity however – particles are created with varying weights throughout the procedure, and then subsequently have to be destroyed in the procedure in order to maintain conservation of mass. The method was therefore altered to a more simplified weighting scheme with the use of LDS and compared to DSMC results from Bird's two-dimensional codes.[35] The results of this modification yet again showed no significant reduction in statistical variance from the standard DSMC treatment. However, the resulting flow-fields were markedly different than the results of the conventional DSMC code. Further analysis by the authors (Greendyke and Bentley) confirmed that it was not the use of the LDS that changed the results, but rather the use of the modified SAR algorithm. In the SAR algorithm used, rather than use the single collision probability of

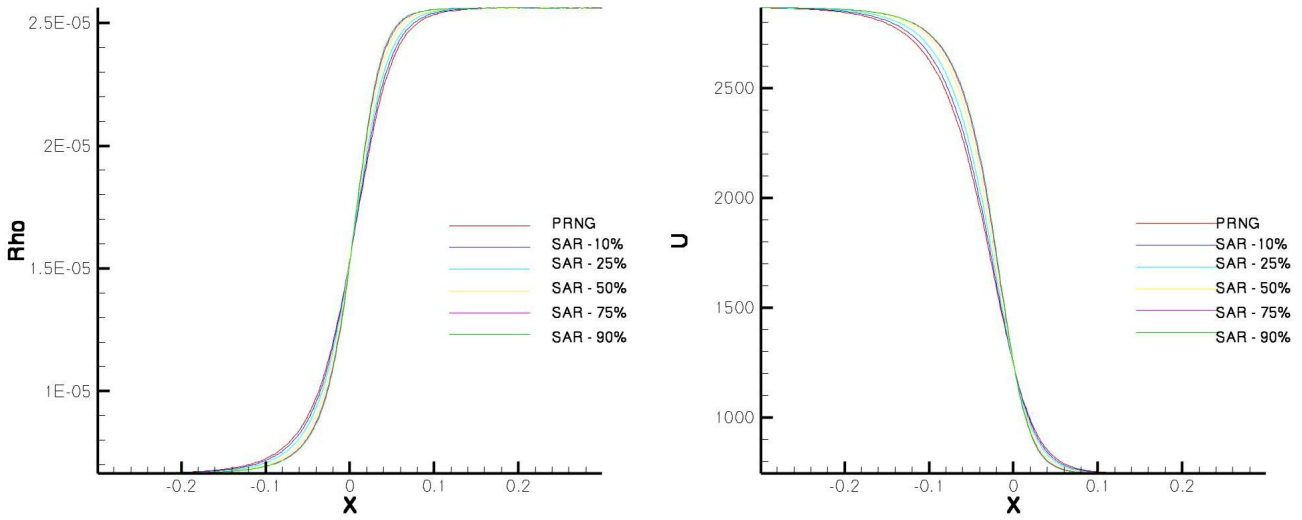
$$\frac{\sigma_T c_r}{(\sigma_T c_r)_{\max}}. \tag{15}$$

A "band" of accepted, but weighted collisions is determined with a parameter:

$$\frac{\sigma_T c_r}{(\sigma_T c_r)_{\max}} \pm \epsilon \tag{16}$$

where ϵ is a predetermined fraction of the $(\sigma_T c_r)_{\max}$ value. In other words, if the value of N_{random} is such that $N_{\text{random}} < [(\sigma_T c_r)/(\sigma_T c_r)_{\max} - \epsilon]$ the collision is accepted with an assigned weighting value of 1. If the random number is such that $N_{\text{random}} > [(\sigma_T c_r)/(\sigma_T c_r)_{\max} + \epsilon]$, the collision is rejected and both collision partners are assigned a weighting of zero for sampling purposes. However, if $[(\sigma_T c_r)/(\sigma_T c_r)_{\max} - \epsilon] < N_{\text{random}} < [(\sigma_T c_r)/(\sigma_T c_r)_{\max} + \epsilon]$, the collision is accepted, but with a weighting that varies linearly from zero to one. Note that the assignment of a value of zero to the rejected pair of collisions means that only those particles that have just collided – either fully accepted, or partially accepted (via weighting < 1) are included in overall sampling. It was assumed that the exclusion of non-accepted particles would speed the convergence of the solution to local equilibrium values for the computational cell in question, thereby accelerating the overall convergence. The weighting factors then re-appear in the flowfield calculations through weighted averaging:

$$\bar{Q} = \frac{\sum_i w_i Q_i}{\sum_i w_i}. \tag{17}$$



a) Shock profile (mass density). **b)** Shock profile (x-velocity).

Figure 7 1D Shock profile results for Mach 9 flow of Argon using Smoothed Accept/Reject parameter.

$\bar{Q}$ represents the weighted-average of a generic flowfield parameter sampled in a given cell, Q_i the value of Q for each molecule i, and w_i is the weighting assigned to the individual particle. Note that the above does NOT apply to density calculations, since the assignment of a zero weighting, or some weighting value less than one, would represent the destruction of mass in the system. For the calculation of density then, the denominator in the formula (17) is the summation of all particles in the cell and their weights are ignored.

Test Case: 1D Shock Profile

To demonstrate our approach, we considered quasi-one-dimensional supersonic flow. A unique solution exists, represented by a plane shock wave with flow direction determined by compression. These conditions have long been studied in developing computational algorithms and interest continues to this day because they offer an ideal case for comparing various gas dynamic models. Here, we calculate the profiles for various quantities (mass density, temperature, and velocity) across a planar, steady shock wave. The study was performed by two of the authors (Greendyke and Bentley) for a 1-D shock at Mach 9 in order to eliminate possible confusion about the effect of multiple dimensions. When the one-dimensional shock was examined using both PRNG and LDS sequences with a Smoothed Accept/Reject collision routine described above for a standing shock at Mach 9, no appreciable differences were found between the two results for a given smoothing factor ϵ. With or without smoothed A/R, LDS random numbers made no difference compared to conventional PRNG. What we do is that there is a clear effect when changing the value of ϵ which results in a different shock structure profile: Figure(7) shows the results for mass density (a) and x-velocity (b) for this case.

To pursue this line of reasoning further, the analysis was continued for standing shocks in Argon at a wider range of Mach numbers using only the PRNG random number generation already inherent in the DSMC – the use of LDS sequences involving far more computational complexity with apparently no perceivable improvement in the overall algorithm. Note that standing shocks represent an interesting study for DSMC methods since they have historically resulted in DSMC calculations indicating a much thicker shock than those found by experiment,[45] thereby highlighting a traditional weakness of the DSMC method. In addition to varying the values of ϵ, the limiting cases of Bird's standard method, and a modification of the Bird algorithm wherein the accept/reject criteria was eliminated completely (by taking collision pairs chosen at random and accepted without comparison to the maximum collision probability) were performed. The data was then compared to the experimental results of Alsmeyer[46] and data from other researchers included with Alsmeyer. Results were obtained for Mach number values of 1.5, 2, 3, 6, and 9. The resulting inverse shock thicknesses, δ, are plotted in Figure (8) along with the experimental data – the horizontal axis being Mach number, and the vertical being inverse shock thickness. Note that the solid black line is a curve fit of the Alsmeyer data alone. In order to incorporate the other researchers work presented in the Alsmeyer paper and reproduced in this figure, a 5th order curve fit of all data present in the paper is also portrayed in the solid blue line in the figure. The "SAR_Exx" values represent the SAR results wherein the "xx" values are the percentage values of ϵ. The standard DSMC values are shown as "Norm_DSMC" points while the cases where the accept/reject criterion was eliminated are given by the "NO_AR" points.

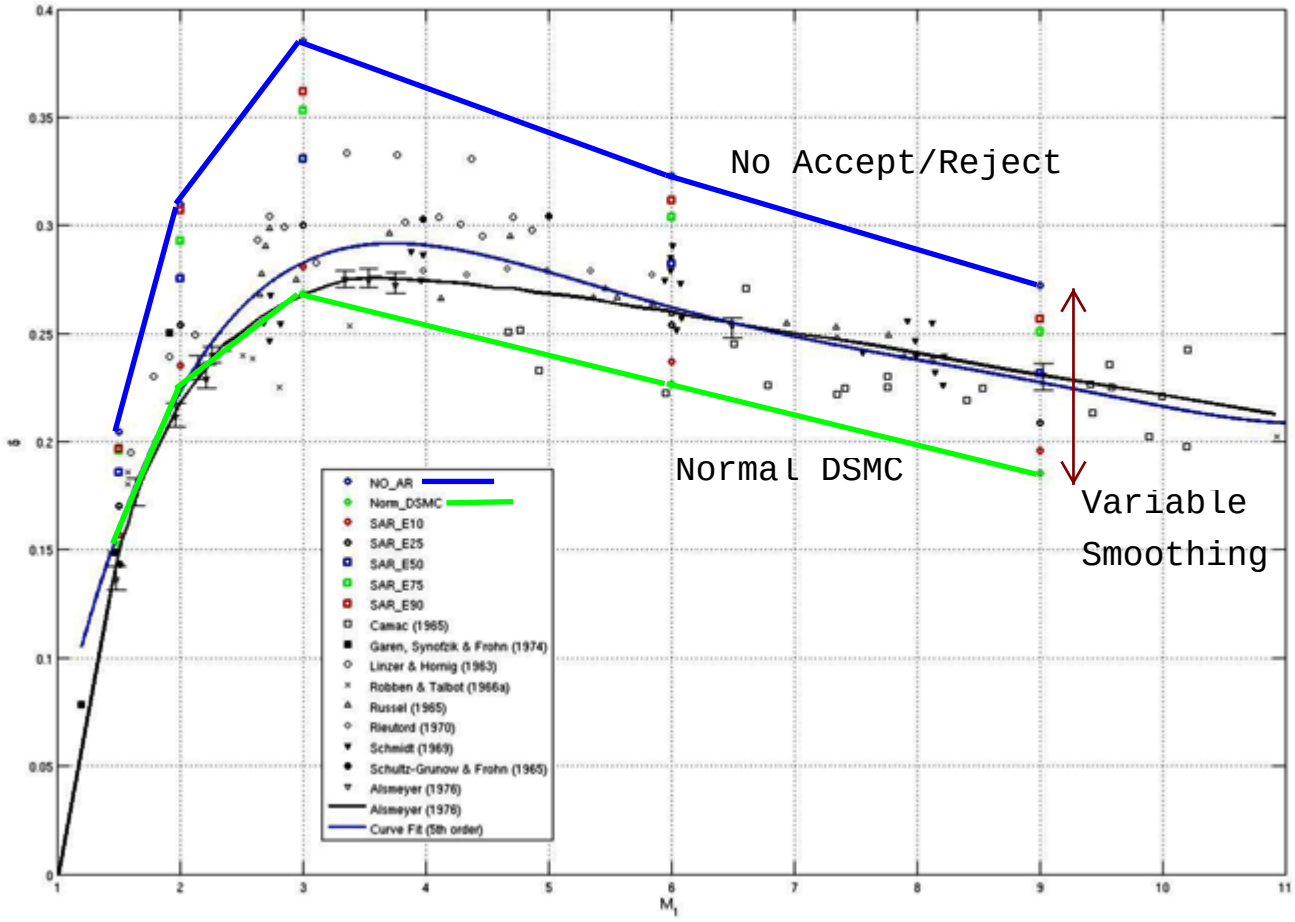


Figure 8 Comparison of inverse shock thickness with Smoothed Accept/Reject and conventional DSMC. The black solid line is a curve fit for the Alsmeyer data only; the solid blue line is a curve fit for all experimental data included. The adjustable parameter ϵ representing the authors' Smoothed Accept/Reject method yields inverse shock thickness values (vertical axis) that span the experimental data across the range of Mach numbers (horizontal axis). The simulated quasi-1D flow is for the monatomic gas Argon.

Close examination of Figure (8) clearly shows that as Mach number increases, the standard DSMC method predicts a much thicker shock (represented by a lower value of δ) than indicated in experimental results. This effect becomes more significant with increasing Mach number. Completely eliminating the accept/reject criteria in the collision simulation yields a much thinner shock than experiment – it is interesting to note that this "no accept/reject" criteria still gives better agreement with experiment than higher-fidelity BGK solvers[45] or modified Navier-Stokes solution methods.[47] The "NO_AR" results and standard DSMC cases thus represent the upper and lower bounds of the computational results compared to experiment for the range of Mach numbers considered. In contrast, our Smoothed Accept/Reject method yields excellent results in comparison to experimental data – both the Alsmeyer curve fit as well as the current curve fit. The accuracy displayed by the SAR method depends on the value of ϵ used, with better agreement at higher Mach numbers. The present comparison supports the conjecture that the appropriate value of ϵ may depend on local Mach number, hinting at a functional relationship of $\epsilon(M)$.

Given the results presented above, the SAR method represents at least a "tunable" parameter in DSMC calculations that can improve the accuracy of the DSMC method with only a small accompanying change in established DSMC codes. It is also equally likely that the new method represents a new way of simulating particle interactions that may be more representative of the actual physics of the situation than the conventional method. The SAR method alters the physics of the computation in three ways: (1) The use of a smoothed band of accepted collisions, (2) the value of ϵ, and (3) the use of only freshly collided molecules in the determination of flow-field quantities. We note finally that comparing 1D shock profiles for quantities such as mass density, heat transfer, temperature, etc. yielded results similar to the shock thickness comparison, with improvement noted upon using Smoothed A/R.

Future Work

The work presented herein is entirely a "labor of love". We have submitted funding proposals to continue the analysis with graduate student talent and effort. Future work is envisioned to take place in two phases. The first phase will consist of an extension of the work presented here. The SAR method will be incorporated into two and three dimensional codes. The results of these calculations will be compared to existing experimental data to determine the most appropriate values of the smoothing parameter – it should be pointed out that no rule-of-thumb exists for the values of ϵ, and the use of values above was purely to determine the range of effects obtained by various values. The SAR method outlined here is entirely new and novel – as such, there is no existing body of theory for the method. The original justification for the method was entirely ad hoc and phenomenological in nature – the effect of SAR calculations was observed, and noted to change DSMC results in the correct direction. Its basis in physical theory is not entirely understood at present.

We will focus the second phase of the project on developing the physical models behind the SAR method, as well as alternative modeling techniques. Not only do studies need to be performed to determine correct values of ϵ used in the SAR method, but a study of the physical basis for the observed effect needs to be conducted. Several alternative models for the convergence of the collision processes will also be examined, with the effects compared to appropriate experimental data. One such method would involve the abandonment of the stochastic determination of "acceptable" collisions in favor of a deterministic selection process among available molecular collision pairs inside a computational cell to speed convergence to local thermodynamic equilibrium. For instance, collision partners could be taken directly from the cell's particle velocity distribution wherein a high velocity particle is paired for a collision with a low velocity particle intentionally. The thought behind this concept is that by directly choosing particles from opposing sides of the velocity distribution function the resulting post-collision velocities for both colliding molecules will approach the most probable velocity of a localized Maxwellian distribution function. The effect will then be to force the computational cell more rapidly to a local equilibrium and thereby speed convergence as well as reduce the statistical variance observed in DSMC methods. The overall nonequilibrium of the flow-field will still be maintained by the code however – the method just described would only hasten local equilibrium – which is already intrinsic to the DSMC method. Nonequilibrium is maintained, as in the standard DSMC method, by "piecewise equilibrium" since the local cell's equilibrium is not dependent on the surrounding cells being in overall equilibrium with the cell in question.

Direct Methods

Another promising approach that we plan to pursue deals with the direct method[7,48–51] for approximating solutions to Boltzmann's equation. This involves approximating the Boltzmann collision integral directly and goes back to the 1960s work by Nordsieck and Hicks[48] and Desphande and Narasimha.[49] Deshpande and Narasimha[49] actually derive an analytic expression for the collision integral in the special case when the initial distribution is a sum of Maxwellian probability densities. In that case, the collision integral is explicitly

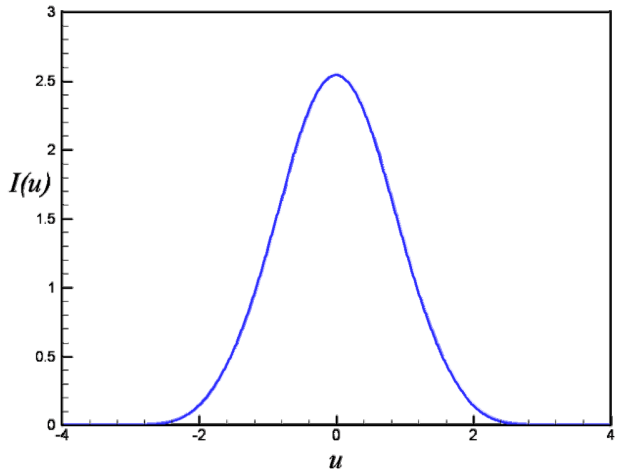
$$\frac{1}{\pi}\int_{\mathbb{R}^3}\int_0^{2\pi}\int_0^{\pi} ||\bar{w}|| e^{h(u,\bar{w},\phi,\psi)} e^{-||\bar{w}||^2} \sin\phi \, d\phi \, d\psi \, d\bar{w} \tag{18}$$

where u is a parameter, $\phi \in [0,\pi]$, $\psi \in [0,2\pi]$, $\bar{w} \in \mathbb{R}^3$, and the function h is given by

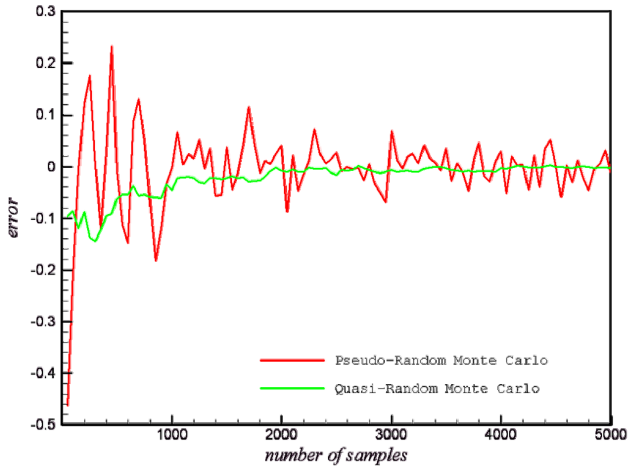
$$h(u,\bar{w},\phi,\psi) = -2u^2 + u\left(w_1 + w_2 + (w_2 - w_1)\cos\phi\right) + u\sin\phi\sin\psi\left(\sqrt{w_2^2 + w_3^2} + \frac{||\bar{w}||w_3\cot\psi + w_1 w_2}{\sqrt{w_2^2 + w_3^2}}\right). \tag{19}$$

The value of this integral at $u = 0.20$ is 2.4793846754936771044. We use importance sampling with respect to the distribution $e^{-||\bar{w}||^2}$ in the integrand and compare Monte Carlo integration to Quasi-Monte Carlo integration for the evaluation of this integral. Using the known value from the analytic solution, we can compare the errors using both methods as the number of samples increases. The graph of the analytic solution as a function of u is given in Figure(9(a)). The graph comparing the error obtained with MCI to the error obtained with QMCI are given in Figure(9(b)). Notice that the quasi-random integration has essentially converged at about 2500 samples (green line) whereas the pseudo-random integration is still oscillating at 5000 samples (red oscillating line). This is a considerable reduction of work; to quantify we would need to continue the standard MCI calculation to the same level of convergence as the QMCI result (not shown).

A general theory for importance sampling in the direct evaluation of the Boltzmann collision integral has been developed in a series of papers[50,51] in which it is shown that if the importance sampler is chosen carefully, a considerable reduction in the number of samples for Monte Carlo integration can be made. We believe that this reduction can be lowered still if these methods are combined with smoothed accept/reject in conjunction with quasi-random methods for drawing the samples. Results of experiments along these lines will be reported in our forthcoming analyses.



a) The graph of an analytic expression for the collision integral.



b) Error graphs for samples going from 50 to 5000 comparing MCI to QMCI for a 5D collision integral

Figure 9 Comparison of standard MC and quasi-random MC convergence for 5D collision integral. The quasi-random integration has essentially converged at about 2500 samples (green line) whereas the pseudo-random integration is still oscillating at 5000 samples (red oscillating line).

Conclusions

We have noted that some caution must be used in adapting Monte Carlo methods to the quasi-random case. In cases where the Koksma–Hlwaka inequality applied, we saw a great improvement in the adapted algorithm. It has been observed[1–3] that the accept/reject algorithm for drawing a sample according to a given distribution corresponds to a discontinuous integrand and as such, the Koksma–Hlwaka inequality does not apply. Considerable improvement has also been observed using quasi-random samples with *smoothed* accept/reject algorithms. As we pointed out, McNenly[23] also observed the inherent discontinuities of accept/reject and found a successful adaptation of a Monte Carlo method for the special case of free-molecule gas flow he studied by eliminating the accept/reject stage using another algorithm. We began a sequence of experiments to see what could be done to possibly improve Bird's DSMC algorithm and have seen, consistent with other researchers, that simply replacing calls to a PRNG can give incorrect results. Correcting that by scrambling can lead to correct results, but not necessarily a vast improvement in DSMC. Adapting the direct method to the quasi-random setting in connection with smoothed accept/reject for importance sampling however has a very promising outlook. We will continue with the SAR method by extending it to two and three dimensional modeling and the results compared to known experimental data in the literature. Further along, we will re-examine the fundamental physics upon which the Accept/Reject stages of the DSMC algorithm is based, not just ad-hoc improvements to the implementation.

Acknowledgements

The authors would like to thank the many fruitful discussions with numerous researchers about the subject matter. Also, we gratefully acknowledge the support of the USAF SBIR program which partially supported the development of the LDS mathematical library by MSSRC.

References

[1]Leydold, J. and Hörmann, W., "Smoothed Transformed Density Rejection," *Monte Carlo Methods and Applications*, Vol. 10, No. 3-4, 2004, pp. 393–401.

[2]Wang, X., "Improving the rejection sampling method in quasi-Monte Carlo methods," *J. Comput. Appl. Math.*, Vol. 114, No. 2, 2000, pp. 231–246.

[3]Moskowitz, B. and Caflisch, R. E., "Smoothness and dimension reduction in quasi-Monte Carlo methods," *Math. Comput. Modelling*, Vol. 23, No. 8-9, 1996, pp. 37–54, Monte Carlo and quasi-Monte Carlo methods.

[4]Harris, S., *An introduction to the theory of the Boltzmann equation*, Dover, Mineola, New York, 2004.

[5]Bird, G. A., *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford University Press, New York, 1995.

[6]Cercignani, C., *The Boltzmann Equation and its Applications*, Springer-Verlag, New York, 1988.

[7]Aristov, V. V., *Direct Methods for Solving the Boltzmann Equation and Study of Nonequilibrium Flows*, Vol. 60 of *Fluid Mechanics and its Applications*, Kluwer Academic Publishers, Dordrecht, 2001.

[8]Garcia, A. L., *Numerical Methods for Physics*, Prentice Hall, N.J., 2000, Second Edition.

[9]"Source code," http://www.gab.com.au/CORRIG.HTM.

[10]Wagner, W., "A convergence proof for Bird's direct simulation Monte Carlo method for the Boltzmann equation," *J. Statistical Phys.*, Vol. 66, No. 3-4, 1992, pp. 1011–1044.

[11]Nanbu, K., "Direct simulation scheme derived from the Boltzmann equation. I. Monocomponent gas," *Journal of the Physical Society of Japan*, Vol. 49, 1980, pp. 2042–2049.

[12]Nanbu, K., "Direct simulation scheme derived from the Boltzmann equation. II. Multicomponent gas," *Journal of the Physical Society of Japan*, Vol. 49, 1980, pp. 2050–2054.

[13]Babovsky, H., "On a Simulation Scheme for the Boltzmann Equation," *Mathematical Methods in the Applied Sciences*, Vol. 8, No. 2, 1986, pp. 223–233.

[14]Babovsky, H. and Illner, R., "A Convergence Proof for Nanbu's Simulation Method for the Full Boltzmann Equation," *SIAM Journal on Numerical Analysis*, Vol. 26, No. 1, 1989, pp. 45–65.

[15]Neyman, J., editor, *Foundations of Kinetic Theory*, Vol. III, Berkeley, 1956, University of California Press.

[16]Ivanov, M. S. and Rogasinsky, S. V., "Efficient schemes for statistical simulation of the rarefied gas flows," *Matemaicheskoe modelirovanie*, Vol. 1, No. 7, 1989, pp. 130–145.

[17]Ivanov, M. S., Rogasinsky, S. V., and Rudyak, V. Y., "Direct statistical simulation method and master kinetic equation," *Rarefied Gas Dynamics - Space Related Studies*, edited by E. P. Munz, D. P. Weaver, and D. H. Campbell, Vol. 116 of *AIAA Progress in Astronautics and Aeronautics*, 1989, pp. 171–181.

[18]Nanbu, K., "Derivation from Kac's master equation of the stochastic laws for simulating molecular collisions," *J. Phys. Soc. Japan*, Vol. 52, No. 12, 1983, pp. 4160–4165.

[19]Knuth, D. E., *The art of computer programming. Vol. 2*, Addison-Wesley Publishing Co., Reading, Mass., 2nd ed., 1981, Seminumerical algorithms, Addison-Wesley Series in Computer Science and Information Processing.

[20]Babovsky, H., Gropengiesser, F., Neunzert, H., Struckmeier, J., and Wiesen, B., "Application of Well-Distributed Sequences to Numerical Simulation of Boltzmann Equation," *Journal of Computational and Applied Mathematics*, Vol. 31, No. 1, 1990, pp. 15–22.

[21]Lécot, C. and Coulibaly, I., "A quasi-Monte Carlo scheme using nets for a linear Boltzmann equation," *SIAM J. Numer. Anal.*, Vol. 35, No. 1, 1998, pp. 51–70 (electronic).

[22]Lécot, C., "A quasi-Monte Carlo method for the Boltzmann equation," *Math. Comp.*, Vol. 56, No. 194, 1991, pp. 621–644.

[23]McNenly, M. J., *Toward the Improved Simulation of Microscale Gas Flow*, Ph.D. thesis, Univ. of Michigan, 2007.

[24]McNenly, M. J. and Boyd, I. D., "Investigating the use of low-discrepancy sequences in particle simulations for rarefied gas flows," *Department of Energy Computational Science Graduate Fellowship Annual Fellows Conference*, Krell Institute, Washington, DC, June 21-23 2005, http://www.krellinst.org/csgf/conf/2005/presentations/.

[25]Robert, C. P. and Casella, G., *Monte Carlo Statistical Methods*, Springer-Verlag, New York: NY, 2nd ed., 2004.

[26]van der Corput, J. G., "Verteilungsfunktionen," *Proc. Ned. Akad. v. Wet.*, Vol. 38, 1935, pp. 813–821.

[27]Niederreiter, H., *Random number generation and quasi-Monte Carlo methods*, Vol. 63 of *CBMS-NSF Regional Conference Series in Applied Mathematics*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[28]Halton, J. H., "On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals," *Numer. Math.*, Vol. 2, 1960, pp. 84–90.

[29]Faure, H., *Monte-Carlo and quasi-Monte-Carlo methods for numerical integration*, World Scientific, 2001.

[30]Abramowitz, M. and Stegun, I. A., editors, *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, John Wiley & Sons Inc., New York, 1984, Reprint of the 1972 edition, Selected Government Publications.

[31]Camberos, J. A., Josyula, E., and Lambe, L. A., "Quasi-Random Monte Carlo Integration for Computing Dissociation Rates," *AIAA Paper 2007–4260*, June 2007.

[32]Macheret, S. O. and Adamovich, I. V., "Semiclassical Modeling of State-Specific Dissociation Rates in Diatomic Gases," *Journal of Chemical Physics*, Vol. 113, No. 17, 2000, pp. 7351–7361.

[33]Billing, G. D., "Rate Constants and Cross Sections for Vibrational Transitions in Atom-Diatom and Diatom-Diatom Collisions," *Computer Physics Communications*, Vol. 32, 1984, pp. 45–62.

[34]Lécot, C., "A direct simulation Monte Carlo scheme and uniformly distributed sequences for solving the Boltzmann equation," *Computing*, Vol. 41, No. 1-2, 1989, pp. 41–57.

[35]Camberos, J. A., Greendyke, R. B., and Lambe, L. A., "On Direct Simulation Quasi-Monte Carlo Methods," *AIAA Paper 2008–3915*, June 2008.

[36]Morokoff, W. J. and Caflisch, R. E., "A quasi-Monte Carlo approach to particle simulation of the heat equation," *SIAM J. Numer. Anal.*, Vol. 30, No. 6, 1993, pp. 1558–1573.

[37]Braaten, E. and Weller, W., "An improved low-discrepancy sequence for multidimensional quasi-Monte Carlo integration," *Journal of Computational Physics*, Vol. 33, No. 2, 1979, pp. 249–258.

[38]Cranley, R. and Patterson, T. N. L., "Randomization of Number Theoretic Methods for Multiple Integration," *SIAM J. Numer. Anal.*, Vol. 13, 1976, pp. 904–914.

[39]Owen, A. B., "Randomly permuted (t, m, s)-nets and (t, s)-sequences," *Monte Carlo and quasi-Monte Carlo methods in scientific computing (Las Vegas, NV, 1994)*, Vol. 106 of *Lecture Notes in Statist.*, Springer, New York, 1995, pp. 299–317.

[40]Rjasanow, S. and Wagner, W., "A stochastic Weighted Particle Method for the Boltzmann Equation," *J. Comput. Phys.*, Vol. 124, No. 2, 1996, pp. 243–253.

[41]Rjasanow, S. and Wagner, W., "A generalized collision mechanism for stochastic particle schemes approximating Boltzmann-type equations," *Comput. Math. Appl.*, Vol. 35, No. 1-2, 1998, pp. 165–178, Simulation methods in kinetic theory.

[42]Rjasanow, S., Schreiber, T., and Wagner, W., "Reduction of the number of particles in the stochastic weighted particle method for the Boltzmann equation," *J. Comput. Phys.*, Vol. 145, No. 1, 1998, pp. 382–405.

[43]Rjasanow, S. and Wagner, W., "Stochastic Weighted Particle Method, Theory and Numerical Examples," *Bull. Inst. Math. Acad. Sin. (N.S.)*, Vol. 2, No. 2, 2007, pp. 461–493.

[44]Rjasnow, S. and Wagner, W., *Stochastic Numerics for the Boltzmann Equation*, Springer-Verlag, Berlin, 2005.

[45]Bentley, B. I. and Greendyke, R. B., "Using the Unified Flow Solver to Investigate the Normal Shock Wave Structure," *AIAA Paper 2009–1541*, January 2009.

[46]Alsmeyer, H., "Density Profiles in Argon and Nitrogen Shock Waves Measured by the Absorption of an Electron Beam," *J. Fluid Mechanics*, Vol. 74, No. Part 3, 1976.

[47]Claycomb, A. and Greendyke, R. B., "Extending CFD Modeling to the Transition Regime by Enhanced Thermophysical Modeling," *AIAA Paper 2008–3930*, June 2008.

[48]Nordsieck, A. and Hicks, B. L., "Monte Carlo evaluation of the Boltzmann collision integral," *Rarefied Gas Dynamics*, edited by C. Brundin, Vol. 1, Academic Press, 1967, pp. 695–710.

[49]Deshpande, S. and Narasimha, R., "The Boltzmann collision integrals for a combination of Maxwellians," *Journal of Fluid Mechanics Digital Archive*, Vol. 36, No. 3, 1969, pp. 545–554.

[50]Baker, L. L. and Hadjiconstantinou, N. G., "Variance Reduction for Monte Carlo Solutions of the Boltzmann Equation," *Physics of Fluids*, Vol. 17, 2005.

[51]Homolle, T. M. M. and Hadjiconstantinou, N. G., "A low-variance deviational simulation Monte Carlo for the Boltzmann equation," *J. Comput. Phys.*, Vol. 226, No. 2, 2007, pp. 2341–2358.