

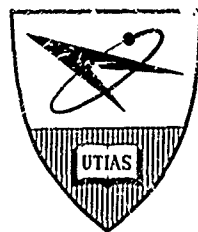
INSTITUTE
FOR
AEROSPACE STUDIES

UNIVERSITY OF TORONTO

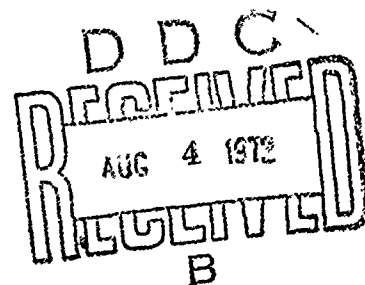
REVIEW OF MONTE CARLO
METHODS IN KINETIC THEORY

by

N. A. Derzko



Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
U S Department of Commerce
Springfield VA 22151



April, 1972.

UTIAS Review No. 35

AFOSR-TR-72-1197

AD 745963

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) UNIVERSITY OF TORONTO INSTITUTE FOR AEROSPACE STUDIES TORONTO 5, ONTARIO, CANADA		2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	
		2b. GROUP	
3. REPORT TITLE REVIEW OF MONTE CARLO METHODS IN KINETIC THEORY			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Scientific Interim			
5. AUTHOR(S) (First name, middle initial, last name) N A DERZKO			
6. REPORT DATE April 1972		7a. TOTAL NO. OF PAGES 80	7b. NO. OF REFS 35
8a. CONTRACT OR GRANT NO AFOSR-71-2091		8b. ORIGINATOR'S REPORT NUMBER(S) UTIAS Review No. 35	
b. PROJECT NO. 9783-01			
c. 61102F		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d. 681307		AFOSR-TR-72-1197	
10. DISTRIBUTION STATEMENT Approved for public release; distribution unlimited.			
11. SUPPLEMENTARY NOTES TECH, OTHER		12. SPONSORING MILITARY ACTIVITY AF Office of Scientific Research (NAM) 1400 Wilson Boulevard Arlington, Virginia 22209	
13. ABSTRACT The present review contains a description with comments of the three main Monte Carlo methods which have been used to date in solution of the full Boltzmann equation of kinetic theory - namely (1) test particle method of J. K. Haviland (2) simulation method of G. A. Bird and, (3) integral evaluation method of A. Nordsieck and B. L Hicks. A chapter containing the necessary formulas for kinetic theory and one on probability theory is included at the beginning. The author's first hand experience with the simulation method has made possible the inclusion of some new material in chapter 4.			

DD FORM 1473
1 NOV 66

ia

UNCLASSIFIED

Security Classification

Security Classification

ib

UNCLASSIFIED

REVIEW OF MONTE CARLO
METHODS IN KINETIC THEORY

BY

N. A. DERZKO

Submitted March, 1971.

April, 1972.

UTIAS Review No. 35

ACKNOWLEDGEMENTS

I wish to express my appreciation to Dr. G. N. Patterson for initiating this review and for his sustained interest during its preparation.

Numerous useful conversations with Dr. J. H. deLeeuw are also gratefully acknowledged.

Financial support for this work was obtained from the United States Air Force Office of Scientific Research under Grant Number AFOSR 71-2091.

SUMMARY

The present review contains a description with comments of the three main Monte Carlo methods which have been used to date in solution of the full Boltzman equation of kinetic theory - namely

1. Test particle method of J. K. Haviland.
2. Simulation method of G. A. Bird.
3. Integral Evaluation method of A. Nordsieck and B. L. Hicks.

A chapter containing the necessary formulas from kinetic theory and one on probability theory is included at the beginning. The author's first hand experience with the simulation method has made possible the inclusion of some new material in Chapter 4.

TABLE OF CONTENTS

	<u>Page</u>
Acknowledgements	
Summary	
1. PRELIMINARY NOTATION AND THEORY	1
1.1 Elastic Collision of Two Molecules Having Equal Mass	1
1.1.1 Classical Description	1
1.1.2 Statistical Description - Scattering Cross-Section	1
1.1.3 The Hard Sphere Model	2
1.1.4 Diagrams	
1.2 The Boltzman Equation	3
1.2.1 The Distribution Function f	3
1.2.2 Equation for the Classical Description	3
1.2.3 Simplification for Hard Spheres	4
1.2.4 Equation in Terms of $G(\hat{w}, g)$	4
1.2.5 Cylindrical Coordinates in Velocity Space	5
1.3 Invariance Properties	5
1.3.1 Normalization of f	5
1.3.2 Change of Scale	6
1.4 The Boltzman Equation from a Computational Standpoint	7
1.5 List of Symbols	10
2. BASIC STATISTICS AND PROBABILITY THEORY	12
2.0 Introduction	12
2.1 Probability Space	12
2.1.1 Definition	12
2.1.2 Coin Tossing	12
2.1.3 Dart Throwing	13
2.1.4 Conditional Probability and the Urn Model	14
2.2 Random Variables	14
2.2.1 Definition and Examples	14
2.2.2 Distribution and Density Functions	15
2.3 Expectation and Variance	15
2.4 Independence	17
2.4.1 Product Spaces	17
2.4.2 Motivation and Definition	17
2.4.3 Independence and Expectation	18
2.5 The Uniform and Normal Distributions	19
2.6 The Central Limit Theorem	20
2.7 Histograms	21

	<u>Page</u>
2.8 Algorithms and Monte Carlo	23
2.8.1 General Structure	23
2.8.2 Monte Carlo Integration	25
2.9 Random Number Generators	26
2.9.1 The Uniform Distribution	26
2.9.2 Method I	27
2.9.3 Method II	28
2.9.4 Generating the Normal Distribution	29
2.9.5 Uniform on the Unit Sphere	30
2.9.6 Effusive Flow	30
2.10 References	32
2.11 List of Symbols	33
3. THE TEST PARTICLE METHOD	35
3.0 Introduction	35
3.1 Preliminary Theory	35
3.1.1 Outline	35
3.1.2 Trajectories to Distribution Function	36
3.1.3 Computation of Trajectories	37
3.1.4 Specialization to One Space and Two Velocity Dimensions	37
3.1.5 A Finite Velocity Range is Sufficient	38
3.2 The Algorithm	39
3.2.1 Storage	39
3.2.2 Data and Formulas	39
3.2.3 Normalization	40
3.2.4 Trajectory Initialization	40
3.2.5 Cell Coordinates	41
3.2.6 Translation	41
3.2.7 Collision	42
3.2.8 Exit Criteria	42
3.3 Variances	43
3.3.1 Variance of FI (i,j,k)	43
3.3.2 Variance of the Moments	43
3.4 Conclusion	44
3.5 References	45
3.6 List of Symbols	46
4. THE MONTE CARLO SIMULATION METHOD	48
4.1 General Description	48
4.2 The Algorithm	48
4.2.1 Storage Layout and Initialization	48
4.2.2 Steps Comprising One Iteration	49

	<u>Page</u>
4.3 Analysis	51
4.3.1 Steady and Time Dependent Problems	51
4.3.2 Cell Size and Δt_m	52
4.3.3 Storage Requirements	52
4.3.4 Computing Time Requirements	53
4.3.5 The Time Increment Formula	53
4.3.6 Boundary Strategies	55
4.3.7 Standard Deviation and Limitations	57
4.4 Variants and Refinements	58
4.4.1 Weighted Cells	58
4.4.2 Mixtures of Gases	59
4.5 References	60
4.6 List of Symbols	62
4.7 Definitions	64
5. MONTE CARLO EVALUATION OF THE COLLISION INTEGRAL	65
5.1 Introduction	65
5.2 Theoretical Preliminaries	65
5.2.1 Nordsieck's Units	65
5.2.2 The Discrete Analogue	66
5.2.3 The Monte Carlo Estimate	67
5.2.4 The Corrections	69
5.2.5 The Finite Difference Scheme	70
5.3 The Algorithm	71
5.3.1 Storage Requirements and Initialization	71
5.3.2 Steps in One Iteration	71
5.4 Comments	73
5.5 References	75
5.6 List of Symbols	77

1. PRELIMINARY NOTATION AND THEORY

1.1 Elastic Collision of Two Molecules Having Equal Mass

1.1.1 Classical Description

A Collision between two such molecules M and M' is adequately described in this setting whenever two pre-collision velocity vectors $\vec{v}$, $\vec{v}'$ and two post-collision velocity vectors $\vec{V}$, $\vec{V}'$ are given. The terms pre-collision and post-collision are only meaningful when the molecules move freely before and after collision, that is when the forces between them act only when they are close together. This assumption is necessary to even speak of a collision process.

The theory shows that $\vec{V}$, $\vec{V}'$ are functions of $\vec{v}$, $\vec{v}'$ and two other parameters b , ϵ whose geometrical meaning is explained in Fig. 1 (b is called the miss distance and ϵ specifies the orientation of the collision plane). Figures 1-3 refer to a coordinate system moving with velocity $\vec{v}$ in which M is initially at rest at the origin. The pre-collision and post-collision relative velocity vectors are denoted by $\vec{v}_r = \vec{v}' - \vec{v}$ and $\vec{V}_r = \vec{V}' - \vec{V}$ respectively and χ is the deflection angle of M. It can be shown that $\vec{V}_r$ lies in the collision plane ($\vec{v}_r$ lies in this plane by definition, so $\vec{V}_r$, $\vec{v}_r$ are coplanar), and that $\chi = \chi(b, g)$ ($g = |\vec{v}_r|$) where the function χ is determined by the collision interaction. For purposes of kinetic theory it is convenient to assume that there is a value b_m such that $\chi = 0$ for $b > b_m$ as would be the case for a purely local interaction. It is still a very good approximation in cases of physical interest.

The collision vector $\vec{e}$ in Fig. 3 is a unit vector in the direction of momentum transfer during collision and is completely determined in the collision plane by b and $\vec{v}_r$, and in three dimensions by v , $\vec{v}_r$ and ϵ ($\vec{e} = \vec{e}(\vec{v}_r, b, \epsilon)$). Figure 4 is the analogue of Fig. 3 for an arbitrary coordinate system in which M and M' have pre-collision velocity vectors $\vec{v}$ and $\vec{v}'$ respectively. The following formulas for the functions $\vec{V}$ and $\vec{V}'$ are useful for subsequent work:

$$\begin{aligned}\vec{V} &= \vec{v} + (\vec{v}_r \cdot \vec{e})\vec{e} \\ \vec{V}' &= \vec{v}' - (\vec{v}_r \cdot \vec{e})\vec{e}\end{aligned}\tag{1}$$

$$\begin{aligned}\vec{V} &= 1/2(\vec{v}' + \vec{v} - \vec{V}_r) \\ \vec{V}' &= 1/2(\vec{v}' + \vec{v} + \vec{V}_r)\end{aligned}\tag{2}$$

1.1.2 Statistical Description - Scattering Cross-Section

The basic element in the statistical description of the collision process is the conditional probability density (differential cross-section) $G(\hat{\omega}, g)$ having the property that $G(\hat{\omega}, g) d\hat{\omega}$ is the probability that the post collision velocity vector of M' has direction within a solid angle $d\hat{\omega}$ about the unit vector $\hat{\omega}$ given that a collision has taken place.

G can be obtained in terms of χ by first observing that the distribution for (b, ϵ) given $b \leq b_m$ has the density function $b/\pi b_m^2$. The calculation is simple if we assume that an inverse function $b(\chi, g)$ can be found with the property that

$$\chi(b(\chi, g), g) = \chi$$

Then the probability that χ lies in the range $(\chi, \chi + d\chi)$ and ϵ lies in the range $(\epsilon, \epsilon + d\epsilon)$ is given by

$$\frac{b}{\pi b_m^2} \frac{\partial b}{\partial \chi}(\chi, g) d\chi d\epsilon$$

Since the range $(\chi, \chi + d\chi) \times (\epsilon, \epsilon + d\epsilon)$ corresponds to a solid angle $\sin \chi d\chi d\epsilon$, it follows that

$$G(\hat{w}, g) = \frac{b}{\pi b_m^2 \sin \chi} \frac{\partial b}{\partial \chi}(\chi, g) \quad (3)$$

where $\hat{w} = (\cos \chi \cos \epsilon, \cos \chi \sin \epsilon, \sin \chi)$

1.1.3 The Hard Sphere Model

This model constitutes one of the rare cases for which χ and G have explicit expressions and is the one most often used in Monte Carlo calculations. The key formula for our purposes, evident from Fig. 5 is

$$b_m \cos \chi = b.$$

It follows that

$$db = -b_m \sin \chi d\chi,$$

and the probability that (χ, ϵ) lies in $(\chi, \chi + d\chi) \times (\epsilon, \epsilon + d\epsilon)$ is

$$\begin{aligned} & \frac{b_m \cos \chi \cdot b_m \sin \chi d\chi d\epsilon}{\pi b_m^2} \\ &= \frac{\sin 2\chi d(2\chi) d\epsilon}{4\pi} \end{aligned}$$

Referring again to Fig. 5, we see that if the deflection angle is χ then the change in direction of the relative velocity vector is 2χ . Now the solid angle corresponding to the range $(2\chi, 2\chi + d(2\chi)) \times (\epsilon, \epsilon + d\epsilon)$ is

$$\sin 2\chi d(2\chi) d\epsilon.$$

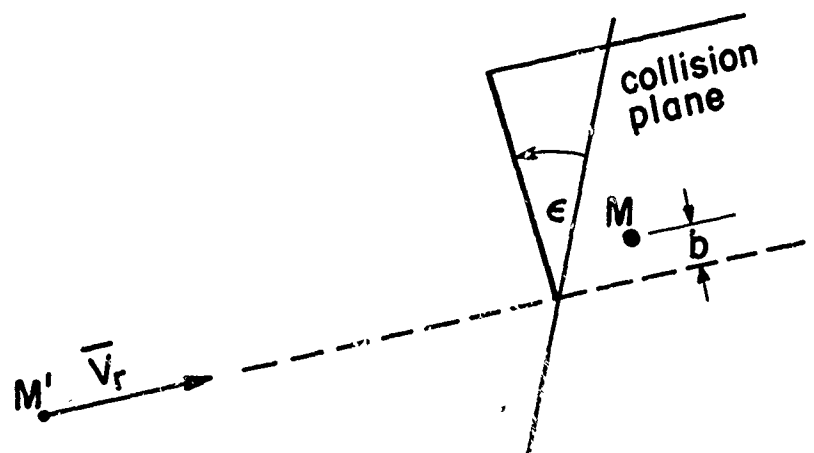


FIGURE 1: LONG BEFORE COLLISION

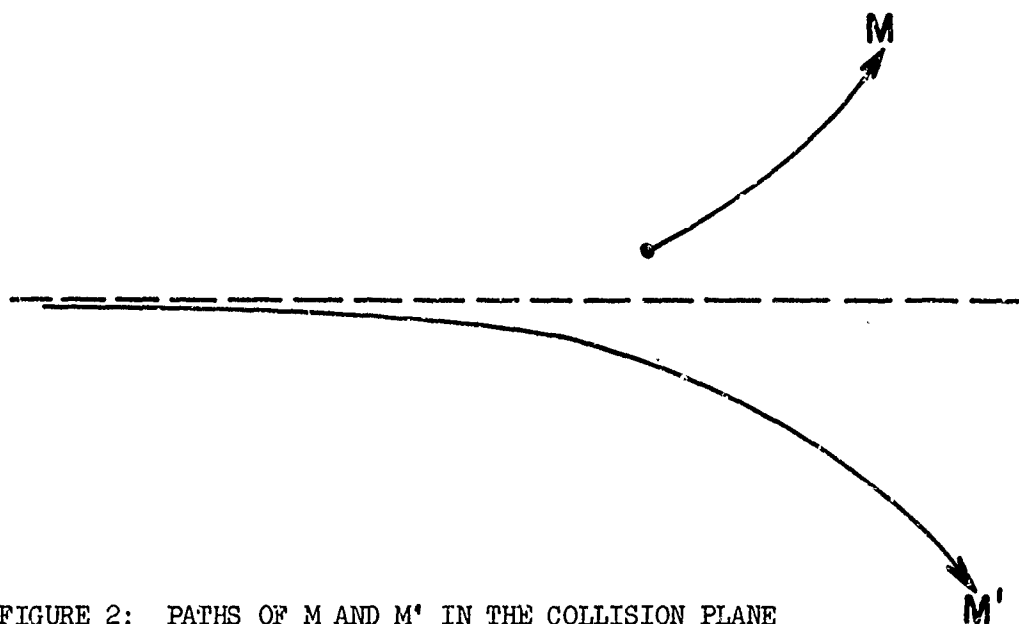


FIGURE 2: PATHS OF M AND M' IN THE COLLISION PLANE

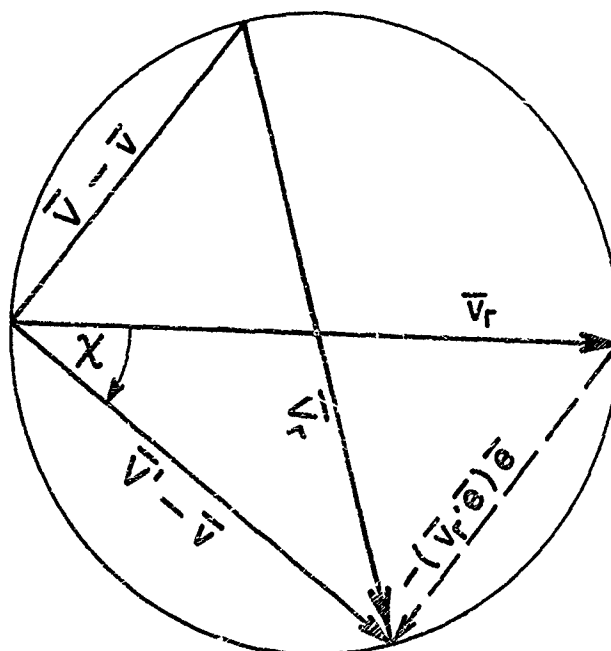


FIGURE 3: VELOCITY VECTORS OF FIGS. 1 and 2

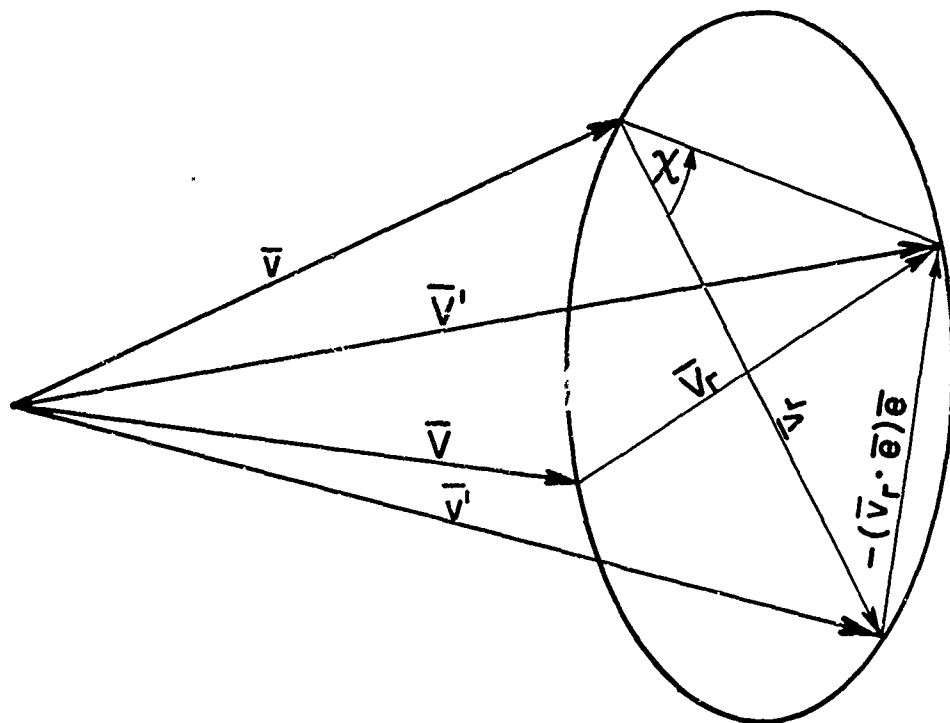


FIGURE 4: RELATIONSHIP BETWEEN VELOCITY VECTORS IN GENERAL

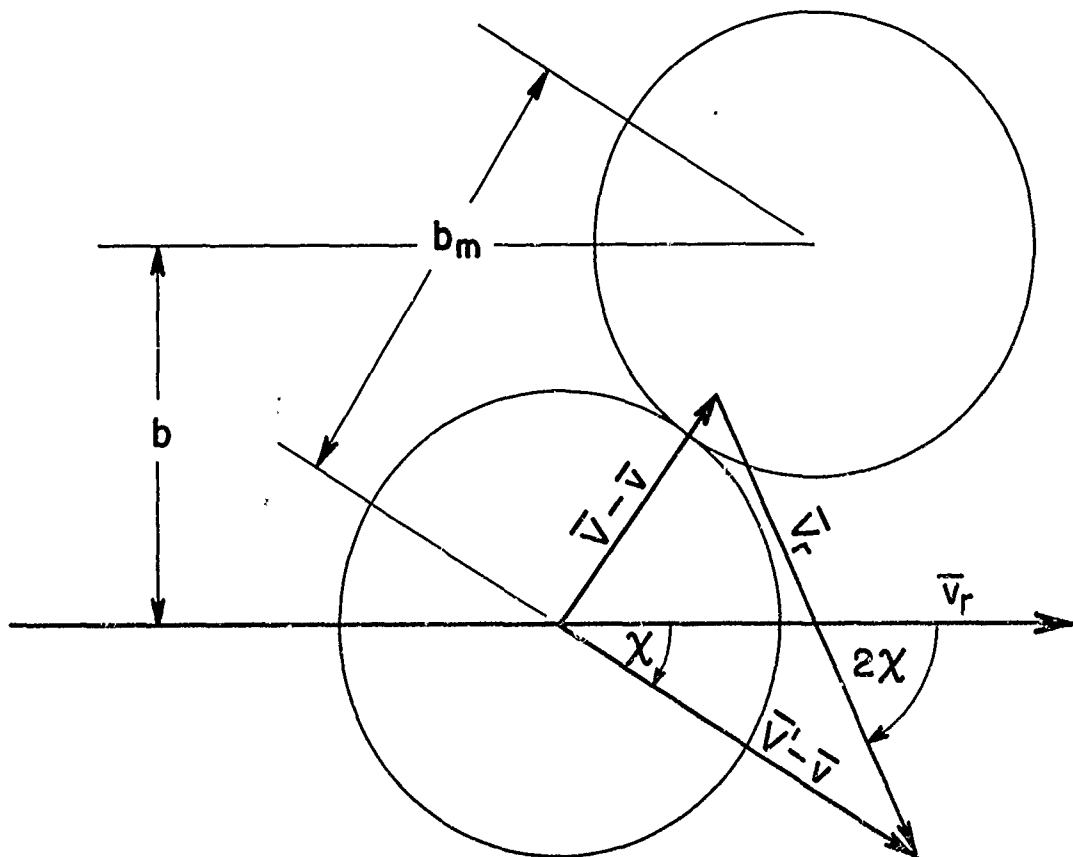


FIGURE 5: HARD SPHERE MODEL

From the preceding two facts we deduce that $\bar{V}_r$ is uniformly distributed on the surface of the unit sphere, and most surprisingly, distributed independently of $\bar{v}_r$. This fact makes the calculation of post-collision velocities very simple in the hard sphere model.

1.2 The Boltzman Equation

1.2.1 The Distribution Function f

Suppose we wish to give a non-detailed description at time t of a gas of like molecules in some region $\mathcal{D}$ of three dimension Euclidean space E^3 . One way of doing this is to specify a positive function $f(\bar{x}, \bar{v}, t)$ for $\bar{x} \in \mathcal{D}$, $\bar{v} \in E^3$ such that $f(\bar{x}, \bar{v}, t) d\bar{x} d\bar{v}$ yields the expected number of molecules in the small six-dimensional rectangular cell about the point $(\bar{x}, \bar{v})$ whose sides have lengths given by $(d\bar{x}, d\bar{v})$. The set of all pairs $(\bar{x}, \bar{v})$ is known as phase space and is denoted by $\Omega = \mathcal{D} \times E^3$.

1.2.2 The Equation for the Classical Description

The Boltzman equation is a conservation equation for f . Its basic form is

$$\begin{aligned} & \left(\frac{\partial f}{\partial t} + \bar{v} \cdot \nabla f \right) (\bar{x}, \bar{v}, t) \\ &= \int d\bar{v}' \int_0^{b_m} b db \int_0^{2\pi} d\epsilon |\bar{v}_r| f(\bar{x}, \bar{v}', t) f(\bar{x}, \bar{v}, t) \\ & - \int d\bar{v}' \int_0^{b_m} b db \int_0^{2\pi} d\epsilon |\bar{v}_r| f(\bar{x}, \bar{v}, t) f(\bar{x}, \bar{v}', t) \end{aligned} \quad (4)$$

where $\bar{v}'$ and $\bar{v}$ are functions of $\bar{v}'$, $\bar{v}$, b , ϵ as described in the foregoing section. The three terms which have been displayed in the above equation are known respectively as the convection term, the gain term, and the loss term.

Because the expressions involved are lengthy ones, certain conventions have been adopted in order to shorten the amount of writing involved. Let f denote $f(\bar{x}, \bar{v}, t)$, f' denote $f(\bar{x}, \bar{v}', t)$, F denote $f(\bar{x}, \bar{v}, t)$, and F' denote $f(\bar{x}, \bar{v}', t)$. Then (4) can be written as

$$\frac{\partial f}{\partial t} + \bar{v} \cdot \nabla f = \int d\bar{v}' \int_0^{b_m} b db \int_0^{2\pi} d\epsilon |\bar{v}_r| (F'F - f'f) \quad (5)$$

An examination of the loss term reveals that $f(\bar{x}, \bar{v}, t)$ is independent of the variables of integration and can therefore be taken outside the integral sign. Accordingly the gain term is often denoted by A and the loss term by Bf . Thus

$$\frac{\partial f}{\partial t} + \bar{\mathbf{v}} \cdot \nabla f = A - Bf \quad (6)$$

The combination $A-Bf$ is often referred to as the collision integral and written $C(f)$ to emphasize that it is an operator on f .

The mathematical representation of a physical problem requires that, in addition to equation (4) initial and boundary conditions on the function f must be satisfied. If the representation is complete it is then expected that the solution f will exist and be unique.

1.2.3 Simplification for Hard Spheres

If we recall that $b = b_m \cos \chi$ (see Fig. 5) where χ is the angle between $\bar{\mathbf{v}}_r$ and $\hat{\omega}$, it follows that

$$|\bar{\mathbf{v}}_r| b \, db \, d\epsilon = b_m^2 (\hat{\omega} \cdot \bar{\mathbf{v}}_r) \, d\hat{\omega}$$

Consequently (5) becomes

$$\frac{\partial f}{\partial t} + \bar{\mathbf{v}} \cdot \nabla f = b_m^2 \int d\bar{\mathbf{v}}' \int_{\hat{\omega} \cdot \bar{\mathbf{v}}_r \geq 0} d\hat{\omega} (\hat{\omega} \cdot \bar{\mathbf{v}}_r) (F'F - f'f)$$

and by making a minor extension in the definition of $\bar{\mathbf{v}}$ and $\bar{\mathbf{v}}'$ this can be written as

$$\frac{\partial f}{\partial t} + \bar{\mathbf{v}} \cdot \nabla f = 1/2 \, b_m^2 \int d\bar{\mathbf{v}}' \int d\hat{\omega} |\hat{\omega} \cdot \bar{\mathbf{v}}_r| (F'F - f'f) \quad (7)$$

1.2.4 The Equation in Terms of $G(\hat{\omega}, g)$

The Boltzman equation can also be written in terms of the differential cross-section $G(\hat{\omega}, g)$. The expressions in this case are

$$\begin{aligned} & \left(\frac{\partial f}{\partial t} + \bar{\mathbf{v}} \cdot \nabla f \right) (\bar{\mathbf{x}}, \bar{\mathbf{v}}, t) \\ &= \int d\bar{\mathbf{v}}' \int d\hat{\omega} |\bar{\mathbf{v}}_r| G(\hat{\omega}, |\bar{\mathbf{v}}_r|) \left(f(\bar{\mathbf{x}}, \bar{\mathbf{v}}', t) f(\bar{\mathbf{x}}, \bar{\mathbf{v}}, t) - f(\bar{\mathbf{x}}, \bar{\mathbf{v}}', t) f(\bar{\mathbf{x}}, \bar{\mathbf{v}}, t) \right) \end{aligned} \quad (8)$$

where now, of course, $\bar{\mathbf{v}}, \bar{\mathbf{v}}'$ must be given as functions of $\bar{\mathbf{v}}, \bar{\mathbf{v}}', \hat{\omega}$, instead of $\bar{\mathbf{v}}, \bar{\mathbf{v}}', b, \epsilon$. The integral over $\hat{\omega}$ in this equation is non-zero only over the forward hemisphere $0 < \chi < \pi/2$ since $G = 0$ for $\pi/2 < \chi < \pi$ to allow conservation of momentum.

1.2.5 Cylindrical Coordinates in Velocity Space

In problems requiring only one space dimension to describe, the distribution function f often has the form

$$f(x, y-z, v_x, v_y, v_z, t) = \tilde{f}(x, v_x, (v_y^2 + v_z^2)^{1/2}) \quad (9)$$

$$= \tilde{f}(x, v_x, v_\perp)$$

where
$$v_\perp = (v_y^2 + v_z^2)^{1/2}$$

In this case the left side of (4) has the form

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x}$$

Although no reduction in the dimension of the collision integral follows, it is fairly easy to show that for any f of this form, $C(f)$ has the same form.

To this end it is useful to introduce the representations

$$\vec{v} = (v_x, v_\perp \cos \varphi, v_\perp \sin \varphi), \quad \vec{v}' = (v'_x, v'_\perp \cos \varphi', v'_\perp \sin \varphi'),$$

and

$$\hat{w} = (w_x, w_\perp \cos \psi, w_\perp \sin \psi), \quad (10)$$

where

$$(w_x, w_\perp) = (\cos \theta, \sin \theta),$$

and to write the collision integral with respect to $dv_x dv_\perp d\varphi' d\theta dv$. It turns out that the integrand in $C(f)$ depends only on the differences $(\varphi - \psi)$ and $(\varphi' - \psi')$ and not on the individual angles. Since the integrals with respect to $d\varphi'$ and dv are over the interval $(0, 2\pi)$, a simple change of variables yields the result that $C(f)$ is independent of φ . The Boltzmann equation using cylindrical coordinates in velocity space is then

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} = \frac{b^2}{2} \int_{-\infty}^{\infty} dv'_x \int_0^{\infty} v'_\perp dv'_\perp \int_0^{2\pi} d\varphi' \int \hat{w} \cdot \vec{v}_r |FF' - ff'| \quad (11)$$

where the $\vec{v}$'s have been omitted for convenience.

1.3 Invariance Properties

1.3.1 Normalization of f

If f satisfies equation (8) with differential cross-section $G(\hat{w}, g)$ and $\alpha > 0$ then αf satisfies (5) with differential cross-section $G(\hat{w}, g)/\alpha$. This makes it possible to work with a normalized distribution

function $\bar{f}$ having the property that

$$\int_{E^3} \bar{f}(\bar{x}_0, \bar{v}, t_0) d\bar{v} = 1$$

at some reference point $\bar{x}_0$ and time t_0 . That is,

$$\bar{f}(\bar{x}, \bar{v}, t) = f(\bar{x}, \bar{v}, t) / n_0$$

where

$$n_0 = \int_{E^3} f(\bar{x}_0, \bar{v}, t_0) d\bar{v}.$$

The new differential cross-section is then

$$G(\hat{w}, g) = n_0 G(\hat{w}, \hat{g})$$

and can be interpreted as a differential cross-section density because of the equation

$$\int G(\hat{w}, g) d\hat{w} = n_0 \int \hat{G}(\hat{w}, g) d\hat{w} = n_0 A(g).$$

The quantity $(\sqrt{2} n_0 A(g))^{-1}$ is the mean free path at speed g . In the case of hard sphere collisions both A and the mean free path are independent of g . In this case, if L is a characteristic dimension in the problem,

$$L (n_0 A)^{-1}$$

constitutes a useful dimensionless matching parameter commonly known as the Knudsen number.

1.3.2 Change of Scale

In theoretical work it is frequently convenient to express the basic equations in terms of time and length units characteristic of the problem at hand. To this end let

$$t = \tau t_1,$$

$$\bar{x} = \lambda \bar{x}_1,$$

$$\text{and} \quad \alpha = \lambda / \tau,$$

from which it follows that

$$\bar{v} = \alpha \bar{v}_1.$$

Put

$$\tilde{f}(\bar{x}_1, \bar{v}_1, t_1) = (\lambda \alpha)^3 f(\lambda \bar{x}_1, \alpha \bar{v}_1, \tau t_1).$$

Then

$$\int_{\Omega_1} \tilde{f}(\bar{x}_1, \bar{v}_1, t_1) d\bar{x}_1 d\bar{v}_1 = \int_{\Omega} f(\bar{x}, \bar{v}, t) d\bar{x} d\bar{v}.$$

Furthermore, if $f(\bar{x}, \bar{v}, t)$ satisfies equation (4) with

$$\bar{v} = \bar{v} + (\bar{v}_r \cdot \bar{e}) \bar{e}$$

$$\bar{v}' = \bar{v}' - (\bar{v}_r \cdot \bar{e}) \bar{e}$$

$$\bar{e} = \bar{e}(\bar{v}_r, b, \epsilon),$$

then a straightforward calculation reveals that $f(\bar{x}, \bar{v}, t)$ satisfies

$$\frac{\partial f}{\partial t_1} + v_1 \cdot \nabla_1 \tilde{f} = \int d\bar{v}_1' \int_0^{b_m/\lambda} b_1 db_1 \int_0^{2\pi} d\epsilon |\bar{v}_{1r}| (\tilde{f}' \tilde{F} - \tilde{f} \tilde{F}') \quad (12)$$

where

$$\tilde{f} \equiv f(\bar{x}_1, \bar{v}_1, t_1)$$

$$\tilde{f}' \equiv f(\bar{x}_1, \bar{v}_1', t_1)$$

$$\tilde{F} \equiv f(\bar{x}_1, \bar{V}_1, t_1)$$

$$\tilde{F}' \equiv f(\bar{x}_1, \bar{V}_1', t_1)$$

$$\bar{V}_1 \equiv \bar{v}_1 - (\bar{v}_{1r} \cdot \bar{e}_1) \bar{e}_1$$

$$\bar{V}_1' \equiv \bar{v}_1' + (\bar{v}_{1r} \cdot \bar{e}_1) \bar{e}_1$$

$$\bar{e}_1(\bar{v}_{1r}, b_1, \epsilon) = \bar{e}(\alpha \bar{v}_{1r}, \lambda b_1, \epsilon)$$

That is, the Maxwell-Boltzman equation is invariant under scale changes if the proper change is made in the collision vector $\bar{e}$ and the miss distance b_m .

1.4 The Boltzman Equation from a Computational Standpoint

Any attempt to solve the full Boltzman equation by computational means must offer a solution to two basic problems - a feasible evaluation

of the collision integral and a convergent strategy for obtaining a solution to the first order nonlinear partial differential integral equation which constitutes the Boltzman equation. A straightforward resolution of these problems fails because both the fast access storage capacity and the computing speed of even our fastest machines are inadequate for dealing with even simple problems. There are, of course, other questions to be considered such as boundary conditions, but they pose difficulties of a lesser order of magnitude.

Even the storage requirements for the distribution function $f(\bar{x}, \bar{v}, t)$ at fixed time t are considerable, for suppose that f were tabulated by storing its values at 10 grid points for each degree of freedom. Then in the general case of 3-dimensional $\bar{x}$ and $\bar{v}$, 10^6 fast access storage locations would be required which is beyond the capacity of present day computers. In a highly symmetrical problem such as the normal plane shock, one can get by using one dimension for $\bar{x}$ and two dimensions for $\bar{v}$ by making use of the symmetry. In this example only 10^3 values are required under the assumption of 10 grid points per degree of freedom. Since 10^3 is well within the range of most computers, this particular problem has been done by several authors.

The evaluation of the collision integral by numerical quadrature is considerably more intractable. To gain an appreciation of the difficulties involved it is sufficient to consider a typical calculation. Since $\bar{x}$ and t remain fixed during one evaluation of the collision integral, the distribution function will be written $f(\bar{v})$; and since it is expected that f decreases very rapidly (like $\exp(-c|\bar{v}|^2)$) for large $\bar{v}$, consider f only in a subregion S of velocity space with the property that the integral of f outside of S is negligible. Suppose, furthermore that S is subdivided into bins denoted by S_k $1 \leq k \leq n$, and that f will be represented in computer memory by storing the n numbers $f_k = \int_{S_k} f(\bar{v}) d\bar{v}$ $1 \leq k \leq n$. That is, the actual f used in the computation will be an approximation given by

$$\tilde{f}(\bar{v}) = \sum_{k=1}^n f_k \chi_k(\bar{v}) \quad (13)$$

where $\chi_k(\bar{v})$ is a function taking the value 1 for $\bar{v} \in S_k$ and 0 otherwise. Then

$$C(\tilde{f}) = \sum_{i,j} f_i f_j \int d\bar{v}' \int_0^b \int_0^b b db \int_0^{2\pi} d\epsilon [\chi_i(\bar{v}') \chi_j(\bar{v}) - \chi_i(\bar{v}') \chi_j(\bar{v}')] \quad (14)$$

and the resultant contribution to bin k is given by

$$\tilde{C}_k(f) = \sum_{i,j} p_k^{ij} f_i f_j \quad (15)$$

where

$$P_k^{ij} = \int_{S_k} d\vec{v} \int d\vec{v}' \int_0^{b_m} b db \int_0^{2\pi} d\epsilon [\chi_i(\vec{v}') \chi_j(\vec{v}) - \chi_i(\vec{v}) \chi_j(\vec{v}')]]$$

Even under the course assumption that there be 10 bins in each of three directions in velocity space, n has to be 10^3 . This means that the evaluation of C requires 2×10^6 multiplications and 10^6 additions at each point k or 2×10^9 multiplications and 10^9 additions altogether. Since even the fastest present-day computers have multiplication times of the order of 10^{-6} sec., it is evident that capabilities fall short of requirements even without considering the colossal task of computing and/or storing the coefficient array P_k^{ij} .

The problem of finding the best strategy for solving the nonlinear differential equation

$$\frac{\partial f}{\partial t} + \vec{v} \cdot \nabla f = C(f)$$

is at least partially a theoretical one and presumably³ solvable even in the absence of a satisfactory means for evaluating the collision integral. Regretably, very little theory is known concerning equations of this type. At least two methods have been tried in the case when C is replaced by a simpler approximation C_a . The first is concisely described by the following formula for $f(\vec{x}, \vec{v}, t + \Delta t)$ in terms of $f(\vec{x}, \vec{v}, t)$:

$$f(\vec{x}, \vec{v}, t + \Delta t) = f(\vec{x}, \vec{v}, t) + [-\vec{v} \cdot \nabla f + C_a(f)] \Delta t, \quad (16)$$

In the case of time dependent problems (16) yields a sequence of approximations to the distribution function at intervals of time Δt . In the case of time independent problems, it is expected that the sequence of approximations will converge to a solution of the equation

$$\vec{v} \cdot \nabla f = C_a(f) \quad (17)$$

in much the same way as a non-equilibrium distribution function relaxes to its equilibrium steady state value in a physical situation. This has been found to be a fairly slow method and greater speed is achieved by solving (17) directly by means of a finite difference scheme.

In spite of the major hurdles that must be overcome, a tremendous amount of work has been done adding to our knowledge of the Boltzman equation. A whole volume would be far from sufficient to describe it. The major categories of solution methods include the moment methods, use of the BGK collision integral, and most recently, the Monte Carlo methods.

In this review we describe the Monte Carlo methods of J. K. Havilland, G. A. Bird, and A. Nordsiek. An attempt has been made to include a large number of related works in the bibliography.

In order to make the discussion as self contained as possible, a chapter on the probability theory relevant to Monte Carlo calculations is included at the beginning.

1.5 List of Symbols

<u>Roman</u>	<u>First Occurrence</u>
A	1.1.1
b, t_m	1.1.1
b_1	1.3.2
B	1.2.2
$C(f)$	1.2.2
$\bar{e}$	1.1.1
$\bar{e}_1$	1.3.2
E^3	1.2.1
$f(\bar{x}, \bar{v}, t)$	1.2.1
f, f', F, F'	1.2.2
$f(x, v_x, (v_y^2 + v_z^2)^{1/2})$	1.2.5
$f(x_1, v_1, t_1)$	1.3.2
g	1.1.1
$G(\hat{w}, g)$	1.1.2
M, M'	1.1.1
t	1.2.1
t_1	1.3.2
$\bar{v}, \bar{v}', \bar{v}, \bar{v}', \bar{v}_r, \bar{v}_r, \bar{v}_r $	1.1.1
v_x, v_y, v_x, v	1.2.5
$\bar{v}_1, \bar{v}_1', \bar{v}_1, \bar{v}_1', \bar{v}_{1r}, \bar{v}_{1r}$	1.3.2
$\hat{w}$	1.1.2
$\bar{x}$	1.1.1
$\bar{x}_1$	1.3.2
y	1.2.5
z	1.2.5

<u>Greek</u>	<u>First Occurrence</u>
α	3.1
ϵ	1.1
φ, ϕ	2.5
$\chi, \chi(b, g)$	1.2
λ	3.2
ν, ν'	2.5
θ	2.5
τ	3.2
Ω	2.1
Ω_1	3.2

<u>Other</u>	<u>First Occurrence</u>
$\mathcal{D}$	2.1
$\nabla \equiv (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$	2.2

2. BASIC STATISTICS AND PROBABILITY THEORY

2.0 Introduction

The purpose of this chapter is to assemble some of the ideas and concepts from probability theory which the author has found useful in dealing with Monte Carlo methods. No attempt shall be made to lay down a precise enough structure to permit rigorous proofs. The definitions are more complete than the remainder of the discussion to permit the reader to easily find additional information in a probability theory textbook, but in most cases we do not take the space to fully explain all the hypothesis mentioned in a definition if they are commonly satisfied in the usual examples of probability spaces. The notation in this section is characteristic of probability theory and independent of that used up to this point.

2.1 Probability Space

2.1.1 Definition

The central notion in statistics and probability theory is that of a sample space or probability space, and we begin by describing this entity in very general abstract terms. Real understanding, of course, comes from seeing how the concept is used in a number of familiar situations and we shall get to that immediately afterwards. We assume for the purposes of the next paragraph that the reader is familiar with the concepts of set, element, subset, union and intersection.

Let Ω be a set and $\mathcal{F}$ be a σ -field of subsets of Ω . The subsets in $\mathcal{F}$ will turn out to represent events. The term σ -field requires that $\mathcal{F}$ contain the empty set ϕ and that it be closed under the operations of countable union and set difference. Suppose, in addition, that we are given a function P whose domain is $\mathcal{F}$ and whose range is the interval $[0,1]$ satisfying $P(\phi) = 0$, $P(\Omega) = 1$ and $P(\bigcup_{n=1}^{\infty} A_n) = \sum_{n=1}^{\infty} P(A_n)$ whenever the A_n , $n = 1, 2, \dots$ are disjoint. This last condition implies that $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ as well as a certain continuity property for P . We call the triple $(\Omega, \mathcal{F}, P)$ a probability space.

2.1.2 Coin Tossing

We will try to show that the abstract structure of the last paragraph provides us with a model fitting all sorts of situations which according to our intuition are of a probabilistic nature. Take, for example the common coin tossing experiment.

Consider first a single toss of a fair coin. We claim that $(\Omega, \mathcal{F}, P)$ with $\Omega = \{H, T\}$, where H denotes heads and T denotes tails, $\mathcal{F} = \{\phi, \{H\}, \{T\}, \{H, T\}\}$ and $P(\phi) = 0$, $P(\{H\}) = P(\{T\}) = 1/2$, $P(\{H, T\}) = 1$ model for this experiment. The reason it is suitable is based upon the fact that if we toss the coin many times then roughly half of the tosses yield heads and half yield tails. In addition, there are two analogous theoretical results proved on the basis of our model, the weak law of large numbers and the strong law of large numbers which strengthens the bond between intuition, experiment, and theory.

The common model for two independent tosses of a fair coin is almost as simple. The probability space can be taken to be as follows:

$\Omega = \{HH, HT, TH, TT\}$, $\mathcal{F}$ consists of all subsets of Ω and if $A \in \Omega$ ("e" meaning "an element of") then $P(A) = (\text{Number of elements of } A)/4$. Suppose we ask now "what is the probability of obtaining both a head and a tail?" In our model this means "what subset A corresponds to 'both a head and a tail' and what is $P(A)$?" The answer is $A = \{HT, TH\}$ and $P(A) = 2/4 = 1/2$.

Similar examples can be given for dealing cards, drawing marbles out of urns and many other real life situations where there is an element of chance involved. We refer the reader to a book on probability theory such as (2, Vol.1) if his interest moves him to find out more about them. It is important for an understanding of the subject to discuss one example which is essentially different from the coin tossing experiment.

2.1.3 Dart Throwing

The coin tossing example had the property that $P(A) > 0$ whenever A contained only a single element of Ω . If this were always true it would be unnecessary to introduce the collection $\mathcal{F}$ into the model. We now describe a situation where it isn't. Suppose a dart is tossed at an infinite dartboard equipped with a cartesian coordinate system. It is aimed at the origin. As any enterprising dart thrower realizes, the chances of hitting exactly the origin, or exactly any other point is zero. On the other hand, the probability of hitting a nonzero area on the dartboard is generally non zero.

For this example let Ω consist of all the points on the dartboard, or better yet, all pairs of real numbers (x,y) which will be coordinates of points on the dartboard (bull's eye at the origin). Let $\mathcal{F}$ be the smallest σ -field of sets containing all rectangles. The satisfying this description turns out to be quite large and contains, for example, all regions whose boundaries are continuous curves. Finally we let

$$P(A) = \frac{1}{\pi c^2} \int_A e^{-(x^2+y^2)/c^2} dx dy$$

Then $P(\phi) = 0$, $P(\Omega) = 1$, and P satisfies the additivity condition because of the properties of the integral. Furthermore, $P(A) = 0$ whenever A is a single point in the plane or, for that matter, even a smooth curve. The function is called the probability density or distribution density function. The constant c is related to the skill of the dart thrower (the smaller c the better). It has been found that this particular type of density function describes an actual dart throwing experiment particularly well. That is, if the dart is thrown N times, where N is large, and the number of times a hit is made inside a given set A is N_A , then N_A/N approximates $P(A)$ and the approximation improves as N increases.

2.1.4 Conditional Probability and the Urn Model

The machinery of sample spaces is ideally suited to deal with problems involving what is known as conditional probability. Let us consider a problem where this concept is involved. Suppose we have two urns numbered 1 and 2 and each contains a red ball and a black ball. A ball is drawn from urn 1 and placed (without looking at the colour) into urn 2. Then a ball is drawn from urn 2 and it is black. What is the probability that the ball originally drawn from urn 1 was black? To solve this problem notice first that there are two possible compositions for urn 2, each equally likely $R_1R_2B_2$ and $B_1R_2B_2$ (to clarify matters we have used subscripts to denote the urn of origin of the various balls). Let $\Omega = \{R_1, R_2, B_2, B_1, R_2, B_2\}$, $\mathcal{F}$ consist of all subsets of Ω and P be defined by $P(A) = (\text{number of elements in } A)/6$. This assignment of probabilities is consistent with our intuitive prejudice that each ball in a given urn has an equal chance of being drawn. The subset of Ω corresponding to the event "a black ball is drawn from the second urn" is $A = \{B_2, B_1, B_2\}$. Since we know this event has occurred we restrict our attention to a new sample space $(\Omega_1, \mathcal{F}_1, P_1)$ where $\Omega_1 = A$, $\mathcal{F}_1$ consists of all subsets of Ω_1 and, since each single element set has equal probability in $(\Omega, \mathcal{F}, P)$, we ask for the same property in $(\Omega_1, \mathcal{F}_1, P_1)$. This means $P_1(B) = (\# \text{ elements in } B)/3$ whenever $B \in \mathcal{F}_1$. Such a definition leads to the conclusion that the probability of a black ball on the first draw, which corresponds to the primed letters in A , is $2/3$. Experience shows $2/3$ is correct, in support of the usefulness of this definition of sample space.

In general if $(\Omega, \mathcal{F}, P)$ is a sample space and Ω_1 , with $P(\Omega_1) > 0$ is a subset specifying a condition known to be satisfied, then the conditional sample space is defined to be $(\Omega_1, \mathcal{F}_1, P_1)$ where $\mathcal{F}_1 = \{\Omega \cap A : A \in \mathcal{F}\}$ and $P_1(B) = P(B)/P(A)$ when $B \in \mathcal{F}_1$.

2.2 Random Variables

2.2.1 Definition and Examples

Having described what is meant by a sample space, we are now ready to discuss random variables, a concept which is indispensable in the study of Monte Carlo simulators. A random variable X on a probability space $(\Omega, \mathcal{F}, P)$ is a function whose domain is the set Ω and whose range is in the real numbers such that $\{\omega : \alpha < X(\omega) < \beta\} \in \mathcal{F}$ for all real numbers $\alpha < \beta$. In practice it is useful to deal with random variables whose range is an n -dimensional Euclidian space, but we shall avoid this complication for the moment. The probability that a random variable assumes a value in an interval (α, β) is denoted by $P(\alpha < X < \beta)$ and given by $P(A)$ where A is the set $\{\omega : \alpha < X(\omega) < \beta\}$, i.e. the set of elements of Ω which lead to a value of X between α and β . It is clear now why we assumed such sets were in $\mathcal{F}$ since $\mathcal{F}$ is the domain of the probability function P .

"Natural" examples of random variables abound in real experiments where chance is involved. If a coin is tossed N times the total number of heads is a random variable. In the dart throwing experiment the x coordinate of a hit is a random variable, so is the y coordinate

and the distance from the origin $(x^2 + y^2)^{1/2}$. There will be many more examples in the Monte Carlo methods we are about to discuss.

2.2.2 Distribution and Density Functions

In working with a single random variable X it is occasionally useful to devise a new sample space $(\Omega, \mathcal{F}, P)$ which is simpler in many cases than the original one and yet leads to the same conclusions as far as probability theory is concerned. This sample space is defined as follows: Ω is the real numbers; $\mathcal{F}$ consists of all the Borel-measurable subsets of the real numbers and P is completely specified by giving its value on every interval (α, β) :

$$P_x((\alpha, \beta)) = P(\alpha < X < \beta)$$

P_x called the probability distribution function for X .

In many cases it is possible to find a real valued function $\varphi(x)$ with the property that

$$P_x((\alpha, \beta)) = \int_{\alpha}^{\beta} \varphi(x) dx$$

for all (α, β) . Then φ is called the probability density function for the random variable X .

2.3 Expectation and Variance

We now come to an exceedingly useful concept. The expectation of a random variable corresponds to the answer we obtain if we make many observations of the random variable and take their average. It is defined most generally as follows:

$$E(x) = \int_{\Omega} X(\omega) P(d\omega).$$

The expression on the right is an integral over the probability space and is defined to be the limit of finite sums of the form

$$\sum_k s_k P(A_k)$$

where A_k is a disjoint decomposition of Ω and the s_k are chosen in such a way that the step function with constant value s_k on A_k ($k = 1, 2, \dots$) approximates X in a certain sense. The theory we have dealt with so far is not equal to the task of making this definition precise.

We shall therefore give another definition which is equivalent to the above one but applies only in the case where the random variable X possesses a density function φ . Then

$$E(X) \equiv \int_{-\infty}^{\infty} x \varphi(x) dx.$$

One of the annoying facts of the business is that the integral in either definition may fail to exist. There is nothing that can be done about this, and we shall not deal with random variables whose expectation fails to exist in the sequel. Expectations and random variables have a number of useful algebraic properties; we now state a most basic one. Let X_1 and X_2 be any two random variables whose expectation exists and α be a real number. Then $\alpha X_1 + X_2$ is again a random variable whose expectation exists and

$$E(\alpha X_1 + X_2) = \alpha E(X_1) + E(X_2)$$

The expectation $E(X)$ is often called the mean of X . This is because if we perform an experiment whose outcome is X N times in such a way that each repetition is independent of the previous ones and take the average of the values of X we obtain in this way, this average turns out to get closer and closer to $E(X)$ as N gets large. This process of repeating an experiment is amenable to a much more careful analysis which we postpone momentarily.

Instead we pursue another idea. Just as it is useful to know the mean of a random variable X , it is useful to have a measure of how much the values of X can be expected to deviate from the mean. The concepts which have been introduced for this purpose are the variance and the standard deviation of X . The variance is defined to be the expectation of the random variable

$$(X - E(X))^2 = X^2 - 2 X E(X) + (E(X))^2$$

That is,

$$\text{Var}(X) = E(X^2) - 2 (E(X))^2 + (E(X))^2 = E(X^2) - (E(X))^2$$

In case X has a probability density function $\varphi(x)$

$$\text{Var}(X) = \int_{-\infty}^{\infty} (x - E(x))^2 \varphi(x) dx$$

The standard deviation of X is defined to be the square root of the variance, i.e.

$$\sigma(X) = (\text{Var}(X))^{1/2}$$

In order to make our statement that the variance and standard deviation give us an idea of how much the values of X can be expected to deviate from the mean, we state a result known as the Chebychev inequality [1, Prop. 1.7, p. 4]:

$$P(\{\omega : |X(\omega) - E(X)|/\sigma \geq \varepsilon\}) \leq \varepsilon^{-2}$$

2.4 Independence

2.4.1 Product Spaces

In order to build a model to handle such situations as repeating a chance experiment we introduce the concept of product of independent probability spaces. To this end let $(\Omega_k, \mathcal{F}_k, P_k)$ $k = 1, \dots, n$ be probability spaces. Let Ω be the sample space whose elements are n -tuples of elements of Ω_k $k = 1, \dots, N$. That is,

$$\Omega = \{(\omega_1, \omega_2, \dots, \omega_N) : \omega_k \in \Omega_k, k = 1, \dots, N\}$$

or equivalently

$$\Omega = \Omega_1 \times \Omega_2 \times \dots \times \Omega_N$$

We let $\mathcal{F}$ be the smallest σ -field containing all sets of the form

$$A = A_1 \times A_2 \times \dots \times A_N,$$

where

$$A_k \in \mathcal{F}_k, k = 1, \dots, N.$$

We define P first on sets of the form $A = A_1 \times \dots \times A_N$ by setting

$$P(A) = P_1(A_1) P_2(A_2) \dots P_N(A_N)$$

and then extend it to all of $\mathcal{F}$ by making use of the addition axiom and of continuity properties of the P_k . It can be checked that $(\Omega, \mathcal{F}, P)$ is a probability space. We shall call it the product of the probability spaces $(\Omega_k, \mathcal{F}_k, P_k)$, $k = 1, \dots, N$, and denote it occasionally by

$$\prod_{k=1}^N (\Omega_k, \mathcal{F}_k, P_k).$$

We shall also have occasion to make use of an infinite product

$$\prod_{k=1}^{\infty} (\Omega_k, \mathcal{F}_k, P_k)$$

of sample spaces. The concept and definition are analogous but too complicated to attempt here.

2.4.2 Motivation and Definition

Sets of the form $\Omega_1 \times \dots \times \Omega_{k-1} \times A_k \times \Omega_{k+1} \times \dots \times \Omega_N$ which we denote by $\hat{A}_k$ exhibit a very important property --- namely,

$$P(\hat{A}_{k_1} \cap \dots \cap \hat{A}_{k_\ell}) = P(\hat{A}_{k_1}) \dots P(\hat{A}_{k_\ell})$$

whenever $k_1 \neq k_2 \neq \dots \neq k_\ell$. It is important enough to have a definition all its own. Events $B_1, B_2, \dots, B_\ell$ are said to be independent of

$$P(B_1 \cap \dots \cap B_\ell) = P(B_1) P(B_2) \dots P(B_\ell)$$

The notion of independence in our model is a rigorous statement of the notion as derived from experiments involving chance. We shall make much use of its consequences for random variables, and we proceed to derive some of them presently.

If X_1 is a random variable on $(\Omega_1, \mathcal{F}_1, P_1)$, then it can be treated as a random variable on

$$(\Omega, \mathcal{F}, P) = \prod_{k=1}^N (\Omega_k, \mathcal{F}_k, P_k)$$

a natural way by letting

$$\hat{X}_1(\omega_1, \dots, \omega_N) = X_1(\omega_1)$$

We find then that

$$P(\alpha < \hat{X}_1 < \beta) = P(\alpha < X_1 < \beta)$$

so that as far as probabilities are concerned X_1 and $\hat{X}_1$ are identical. Suppose X_k is a random variable on $(\Omega_k, \mathcal{F}_k, P_k)$ and $\hat{X}_k$ is the corresponding random variable on the product space, $k = 1, \dots, N$. Then

$$P(X_{k_1} \in J_1, \dots, X_{k_\ell} \in J_\ell) = P(X_{k_1} \in J_1) \dots P(X_{k_\ell} \in J_\ell)$$

where $k_1 \neq k_2 \neq \dots \neq k_\ell$ and $J_i, i = 1, \dots, \ell$ are Borel sets of real numbers. This follows easily from the independence property of events which we discussed in the preceding paragraph. Hence we define the notion of independence for any ℓ random variables $X_1, \dots, X_\ell$ defined on a common probability space by the statement:

$$P(X_1 \in J_1 \text{ \& \dots \& } X_\ell \in J_\ell) = \prod_{i=1}^{\ell} P(X_i \in J_i).$$

2.4.3 Independence and Expectation

If $X_1, \dots, X_N$ are random variables, then so is $\prod_{k=1}^N X_k$. If they are in addition independent it can be proved that

$$E\left(\prod_{k=1}^N X_k\right) = \prod_{k=1}^N E(X_k).$$

For our purposes this property is not as important as one of its consequences --- namely

$$\text{Var}\left(\sum_{k=1}^N X_k\right) = \sum_{k=1}^N \text{Var}(X_k)$$

which we prove here when $n = 2$ for the sake of illustration:

$$\begin{aligned}\text{Var}(X_1 + X_2) &= E[(X_1 + X_2)^2] - [E(X_1 + X_2)]^2 \\&= E(X_1^2 + 2X_1 X_2 + X_2^2) - ((E(X_1))^2 + 2E(X_1)E(X_2) + (E(X_2))^2) \\&= E(X_1^2) + 2E(X_1 X_2) + E(X_2^2) - (E(X_1))^2 - 2E(X_1)E(X_2) - (E(X_2))^2 \\&= E(X_1^2) - (E(X_1))^2 + E(X_2^2) - (E(X_2))^2 = \text{Var}(X_1) + \text{Var}(X_2).\end{aligned}$$

2.5. The Uniform and Normal Distributions

So far we have dealt with a large number of definitions and theory. In order to make use of this theory we need to have some more specific knowledge about probability spaces which recur frequently in building statistical models for real situations. We shall describe two such spaces now.

The first of these is known as the uniform distribution and is defined as follows: Let Ω be the interval (a, b) and let $\mathcal{F}$ consist of all the Borel measurable subsets of (a, b) . The probability function is defined to be

$$P(A) = \int_A \frac{dx}{b-a}$$

where $\frac{1}{b-a}$ is the density function. We then calculate the mean

$$E(X) = \int_a^b \frac{xdx}{b-a} = \frac{a+b}{2} = \mu$$

and the variance

$$E((X-\mu)^2) = \int_a^b \frac{x^2 dx}{b-a} - \mu^2 = \frac{(b-a)^2}{12}$$

and remember these facts for future reference.

The second space we must mention is the normal distribution which is defined as follows: Let Ω be the whole real line $(-\infty, \infty)$ and let $\mathcal{F}$ consist of all the Borel measurable subsets of the real line. We fix values of the parameters μ and σ and define the probability of a set A by

$$P(A) = \frac{1}{\sigma\sqrt{2\pi}} \int_A e^{-(x-\mu)^2/2\sigma^2} dx$$

so that the density function is

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}.$$

The mean then is given by

$$E(X) = \int_{-\infty}^{\infty} \frac{x}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} dx$$

and the variance by

$$E((X-\mu)^2) = \int_{-\infty}^{\infty} \frac{(x-\mu)^2}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} dx = \sigma^2$$

from which the standard deviation is σ . The preceding integrals cannot be evaluated by elementary means and one usually transforms them into linear combinations of

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}, \quad \int_{-\infty}^{\infty} xe^{-x^2} dx = 0, \quad \int_{-\infty}^{\infty} x^2 e^{-x^2} dx = \frac{\sqrt{\pi}}{2}$$

which are commonly found in tables of integrals.

2.6 The Central Limit Theorem

This pretty well completes the outline of the most essential statistical ideas necessary to plunge into a study of Monte Carlo methods. There remains unmentioned, however, a fundamental and far reaching theorem which proves invaluable in understanding the answers a Monte Carlo calculation cranks out. The central limit theorem assumes we have an unlimited number of independent random variables X_k ($k=1,2,\dots$) and each has the same distribution; that is, given (α, β) , then $P(\alpha < X_k < \beta)$ is the same for all $k=1,2,\dots$ leading to a mean μ and a standard deviation σ . Then we can conclude that for N sufficiently large, the random variable

$$\frac{1}{N} \sum_{k=1}^N X_k$$

has an approximately normal distribution with mean μ and standard deviation

$$\frac{\sigma}{\sqrt{N}}$$

Strictly speaking, the term "approximately" above requires more precise definition. We therefore refer the reader to [1, p. 170] where the following precise statement can be found:

Theorem

Let $X_1, X_2, \dots$ be independent, identically distributed random variables each of mean μ and variance σ . Then

$$S_n = \frac{X_1 + X_2 + \dots + X_n}{\sigma \sqrt{n}} \xrightarrow{D} N(0,1)$$

The symbol $\xrightarrow{D} N(0,1)$ stands for converges in distribution to the normal distribution of mean zero and standard deviation 1. It means that if $F_n(\alpha) = P(S_n < \alpha)$ is the distribution function of S_n then

$$\lim_{n \rightarrow \infty} F_n(\alpha) = \int_{-\infty}^{\alpha} \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt$$

2.7 Histograms

Suppose we are given a device which is able to generate random numbers chosen from a distribution with $\Omega = (a,b)$ and density $\varphi(x)$. In this section we discuss the problem of obtaining an approximation to $\varphi(x)$ by plotting a histogram from number N of random numbers $(r_1, \dots, r_N)$ produced by our generator.

First, we state what is meant by a histogram. Let a partition $\{a = x_0 < x_1 < \dots < x_n = b\}$ of the interval (a,b) be fixed. Thus is subdivided into cells $\Omega_k = (x_{k-1}, x_k)$ $k = 1, \dots, n$. The approximation $\varphi_H(x)$ is defined to be the step function whose constant value on the interval Ω_k is obtained by counting the number of the r_k which fall into this interval and dividing the result by $N(x_k - x_{k-1})$.

There is another way of describing the histogram which is more useful for theoretical purposes. Let X_k be the characteristic function of the cell Ω_k , that is,

$$X_k(x) = \begin{cases} 1 & x \in \Omega_k \\ 0 & \text{otherwise} \end{cases}$$

Then X_k is a random variable on $(\Omega, \mathcal{F}, P)$. We can imagine X_k to be a random variable on the i -th factor of the product space

$$\prod_{i=1}^N (\Omega, \mathcal{F}, P)$$

and denote its extension to the whole product space by $X_{k,i}$. For fixed k then $X_{k,i}$ $i = 1, \dots, N$ are identically distributed independent random variables. Furthermore, if we let $\varphi_{H,k}$ denote the value of $\varphi_H(x)$ on Ω_k , we obtain

$$\varphi_{H,k} = \frac{1}{N(x_k - x_{k-1})} \sum_{i=1}^N X_{k,i}$$

$\varphi_{H,k}$ is also a random variable on the product space. We calculate the mean and variance of X_k :

$$E(\bar{x}_k) = \int_{\Omega_k} 1 \cdot p(x) \, dx = \mu_k;$$

$$\text{Var}(\bar{x}_k) = \int_{\Omega_k} 1^2 p(x) \, dx - \mu_k^2 = \mu_k(1 - \mu_k)$$

It follows then that

$$E(\rho_{H,k}) = \mu_k / (x_k - x_{k-1})$$

$$\text{Var}(\rho_{H,k}) = \mu_k(1 - \mu_k) / N(x_k - x_{k-1})^2$$

and

$$\sigma(\rho_{H,k}) = (\mu_k(1 - \mu_k))^{1/2} / N^{1/2} (x_k - x_{k-1})$$

If N is large then we can use the central limit theorem to conclude that $\rho_{H,k}$ will be approximately normally distributed implying that it will fall within two standard deviations of $\mu_k / (x_k - x_{k-1})$ about 98% of the time. It is perhaps more illuminating to look at the "relative deviation"

$$\sigma(\rho_{H,k}) / E(\rho_{H,k}) = (1/\mu_k - 1)^{1/2} N^{-1/2}$$

Then if $\mu_k \ll 1$, which must be the case if the partition points are finely spaced in order to make the histogram show detail, the relative deviation is approximately $(N\mu_k)^{-1/2}$ or (expected number of counts in cell Ω_k) $^{-1/2}$. This is a useful rule of thumb to remember when designing Monte Carlo algorithms. It is heavily dependent, of course, on the assumptions of independence and small Ω_k .

We have worked out the histogram theory for a distribution with $\Omega = (a, b)$ and probability density p . The theory is not really restricted to these assumptions. More generally we can start from an arbitrary probability space $(\Omega, \mathcal{F}, p)$ and choose a partition

$$\Omega = \bigcup_{k=1}^N \Omega_k,$$

Ω_k disjoint and $\Omega_k \in \mathcal{F}$, $k=1, 2, \dots, n$ of the set Ω . Again we choose a sample $(\omega_1, \omega_2, \dots, \omega_N)$ of N random points from the distribution $(\Omega, \mathcal{F}, p)$.

Since P may not be given by a density function it is not possible to define the histogram as a stepfunction approximation to the density function. Instead we let $\rho_{H,k}$ equal the fraction of the ω_i falling into Ω_k and consider it an approximation to

$$P(\Omega_k) = \mu_k.$$

We define x_k as before and find that

$$\phi_{H,k} = \frac{1}{N} \sum_{i=1}^N x_{k,i}$$

is a random variable on the space

$$\prod_{i=1}^N (\Omega, \mathcal{F}, P)$$

whose mean turns out to be μ_k and standard deviation turns out to be

$$\mu_k(1-\mu_k)^{N-1/2}$$

Again the relative standard deviation is

$$(1/\mu_k - 1)^{1/2} N^{-1/2}$$

and our rule of thumb that relative standard deviation $\approx$ (expected number of counts in Ω_k)^{-1/2} still applies.

2.8 Algorithms and Monte Carlo

2.8.1 General Structure

We shall not define precisely what is meant by an algorithm. It is sufficient for our purposes to say that an algorithm is a finite sequence of operations and of things to be operated on, instructions giving the order in which the operations are to be performed and when the execution is to stop, together with a guarantee that the operations can always be carried out under stated conditions. A computer program, of course, fits the description we have just given and it is in this context that we shall be discussing algorithms from now on.

Since we shall not be discussing any particular algorithm in the next few paragraphs and consequently shall not be concerned with the internal details of how a particular algorithm is put together, it is appropriate to use the well known function notation as a shorthand for an arbitrary given algorithm. That is, let U be the vector of numbers which the algorithm uses as initial parameters and V be a vector whose components are the numbers which it produces. Then we write

$$V = \phi(U)$$

to denote the action of the algorithm.

Generally algorithms are devised so that we can use a computer to calculate some function which is at least theoretically known. For example, let A be an $n \times n$ matrix of complex numbers. Then it is known that A has n complex eigenvalues. Alternatively we could say that there exists a function $F(A)$ whose value is a vector of n components - the eigenvalues of A in some order. It is an important problem to devise an algorithm $\phi(A, k)$ which produces a close approximation to the eigenvalues of A . We ask that the algorithm at least satisfy

$$\lim_{k \rightarrow \infty} \phi(A, k) = F(A)$$

Experience tells us that such an algorithm would utilize some iterative method and that the time $T(k)$ required to execute the algorithm would go to ∞ as $k \rightarrow \infty$. That is something one must live with.

With this preamble we now turn our attention to what we shall call a Monte Carlo algorithm. We describe it at first in a somewhat idealized form in order to clarify what is meant.

Suppose that our computer has attached to it a source of random numbers whose distribution is known so that in addition to the usual slate of instructions there is one which enables us to introduce a number from the random source into the computation. If an algorithm makes use of these random numbers, we should intuitively expect that the answer will also be described by a probability distribution. We can describe a Monte Carlo algorithm in terms of our function notation if we imagine that instead of having a random number generator built into the computer we allow the computer to have access to any finite part of an infinite sequence of numbers produced separately by a random number generator. Let such a sequence be denoted by $\bar{r} = (r_1, r_2, \dots)$. The words "finite part" in the second to last sentence deserve emphasis. We wish to exclude such functions as

"Compute the mean of the sequence r "

or

"If $\lim_k r_k$ is greater than 10 commence step 3".

Fortunately in most algorithms it is possible to place an upper bound N on the length of the sequence required by the algorithm. Then it suffices to work with $\bar{r} = (r_1, r_2, \dots, r_N)$.

For theoretical purposes we let $(\Omega_o, \mathcal{F}_o, P_o)$ be the probability space describing the random number generator. Then $\bar{r}$ may be treated as a draw from the probability space

$$(\Omega, \mathcal{F}, P) = \prod_{\ell=1}^N (\Omega_o, \mathcal{F}_o, P_o),$$

or

$$\prod_{\ell=1}^{\infty} (\Omega_o, \mathcal{F}_o, P_o)$$

as the case may be. The Monte Carlo algorithm is denoted by

$$V = \phi(U, \tilde{r}, k).$$

The parameter k has been introduced to allow for convergence properties. It is evident that for fixed U , V is a vector valued random variable on $(\Omega, \mathcal{F}, P)$. It is therefore reasonable to ask for the probability distribution of V . Generally, this is too difficult a question to answer, and for practical purposes certain properties of this distribution suffice.

2.8:2 Monte Carlo Integration

It is time now to describe an algorithm which is, historically speaking, a prototype for all Monte Carlo calculations. The purpose of this algorithm will be to approximate the integral of a given function $f(x)$ on the interval $(0,1)$. It is presumed that f is given to us by specifying d parameters - they could be the coefficients of a polynomial of order $d - 1$, or they might be the values at d points in the interval $(0,1)$ if f is assumed piecewise linear. The exact method of specification is of no direct concern to us. Let us simply assume that once the d parameters are given, we know how to calculate $f(x)$ for any $x \in (0,1)$.

The actual computation of the integral proceeds as follows. Given N , choose N independent random numbers $r = (r_1, r_2, \dots, r_N)$ from the uniform distribution on $(0,1)$ and then compute

$$I = \phi(f, r, N) = \frac{1}{N} \sum_{k=1}^N f(r_k)$$

we claim that I is an approximation to

$$\int_0^1 f(x) dx$$

in the following sense:

$$E(I) = \frac{1}{N} \sum_{k=1}^N E(f) = E(f) = \int_0^1 f(x) dx \equiv \mu$$

$$\text{Var}(I) = \frac{1}{N} \text{Var}(f) = \frac{1}{N} \left(\int_0^1 f^2(x) dx - \mu^2 \right)$$

$$\sigma(I) = N^{-1/2} \left(\int_0^1 f^2(x) dx - \mu^2 \right)^{1/2} = C N^{-1/2}$$

We summarize: The mean of $\phi(f, r, N)$ is the required integral and its standard deviation tends to zero as $N \rightarrow \infty$. Furthermore, ϕ is an average of random variables that for sufficiently large N we can assume it will be approximately normally distributed with mean μ and standard deviation $C N^{-1/2}$. In particular the probability that

$$|\phi - \mu| < 2 C N^{-1/2}$$

is about 0.98. from which we can conclude that with probability 0.98 will approximate $\int_0^1 f(x) dx$ to within ϵ if $N > (2C/\epsilon)^2$.

Before leaving this example, it is important to notice that the number of computations of f and hence the time taken to execute the algorithm increases as N but the standard deviation only decreases as $N^{-1/2}$. If we compare this with a quadratically convergent standard algorithm, we can see that Monte Carlo is a preposterously expensive way to get accuracy.

We can also use this example as a guide to formulate a criterion for convergence of a Monte Carlo algorithm. We say that the algorithm

$$\phi(U, r, n)$$

converges to $F(U)$ as $n \rightarrow \infty$ if

$$\mu_n = E(\phi(U, \cdot, n)) \rightarrow F(U)$$

and

$$\text{Var}(\phi(U, \cdot, n)) \rightarrow 0.$$

Unfortunately, neither this criterion, nor any alternative has been proved for the Monte Carlo algorithms proposed to date in Kinetic Theory. We state it here primarily to throw light on the nature of the answer a Monte Carlo algorithm produces.

2.9 Random Number Generators

2.9.1 The Uniform Distribution

We stated at the beginning of the preceding section on Monte Carlo algorithms that the random number generator attached to our computer was an idealization. In practice a substitute is used.

There are several standard algorithms known which have the property that they produce a sequence of numbers which as far as its statistical properties are concerned could have been produced by sampling from the uniform distribution on $(0,1)$ (there is nothing basic about this interval, it has been adopted by convention mainly). The algorithms must be given a starting value r_0 so that we could write a typical one as

$$R(r_0, n)$$

which gives the n -th pseudo random number if the starting number is r_0 . Among the statistical properties, we demand that histograms built from the sequence $R(r_0, n)$ be statistically indistinguishable from a histogram built from the same number of samples from a uniform distribution.

It is tempting, but inconsistent with our goals, to go into detailed descriptions of the tests which are used in this connection. We refer the reader instead to [3, Ch.12].

The best methods known to date for generating uniformly distributed random numbers are known as the recursive congruential generators (5). The n -th pseudo-random number $R(r_0, n)$ is given by

$$br_0^n \equiv R(r_0, n) \pmod{m}$$

for a suitable choice of the parameters (b, m) . This choice requires a considerable amount of number theory to make properly. The congruential methods are both fast and pass the statistical tests with flying colours.

There are two basic methods for devising an algorithm to generate random numbers from a distribution given by an arbitrary density function $p(x)$ on $(-\infty, \infty)$ which make use of a source of uniformly distributed random numbers.

2.9.2 Method I

Let

$$\Phi(x) = \int_{-\infty}^x p(\xi) d\xi$$

Φ is a nondecreasing continuous function which assumes every value in the interval $(0, 1)$. Consequently, there exists at least one nondecreasing inverse function Φ^{-1} on the domain $(0, 1)$ such that

$$\Phi(\Phi^{-1}(y)) = y.$$

Φ^{-1} is a random variable on the uniform probability space with $\Omega = (0, 1)$. Since

$$x_1 \leq \Phi^{-1}(y) \leq x_2$$

implies

$$\Phi(x_1) \leq y \leq \Phi(x_2)$$

it follows that

$$P(x_1 \leq \Phi^{-1} \leq x_2) = P(\Phi(x_1) \leq y \leq \Phi(x_2)) = \Phi(x_2) - \Phi(x_1)$$

Hence Φ^{-1} is distributed according to the density function $p(x)$, and if $R(r_0, n)$ is a random number generator for the uniform distribution on $(0, 1)$ then

$$\Phi^{-1}(R(r_0, n))$$

is a random number generator for the distribution with density $p(x)$.

Method I, though completely general in theory, is not always useful from a practical point of view. The problem arises when there is no easily computable expression for ϕ^{-1} or, even worse, if we are given ϕ and there is no easily computable expression for either ϕ or ϕ^{-1} . In such cases Method II may be preferable.

2.9.3 Method II

Let ϕ be a given function not identically zero on the interval (a,b) and satisfying the inequality

$$0 \leq \phi(x) \leq c.$$

We shall describe a method for obtaining a random number from the distribution defined by the density

$$\phi(x) / \int_a^b \phi(x) dx$$

Let x be a random number from the uniform distribution on (a,b) and y be a random number drawn from the uniform distribution on $(0,c)$. Then

$$P(\phi(x) \leq y) = \int_a^b \phi(x) dx / (b-a)c$$

and the distribution for x given that the condition $\phi(x) \leq y$ is satisfied is

$$P(\alpha < x < \beta \text{ on the condition } \phi(x) < y) \\ = \int_{\alpha}^{\beta} \phi(x) dx / c(b-a) \left(\int_a^b \phi(x) dx / c(b-a) \right)^{-1} = \int_{\alpha}^{\beta} \phi(x) dx / \int_a^b \phi(x) dx$$

That is, the above conditional distribution for x is the distribution we seek to compute.

On the basis of the theory just derived, we devise the random number algorithm as follows:

A. Choose r_1 and r_2 from the uniform distribution on $(0,1)$. Then

$$x = a + (b-a) r_1, \text{ and } y = cr_2$$

will be uniformly distributed on (a,b) and $(0,c)$ respectively.

B. If $\phi(x) \leq y$ then x is the required random number and we are finished. Otherwise return to A to repeat the process.

One of the interesting features of this algorithm is that the number of steps necessary to generate a random number x is itself a random variable. Some further analysis is in order. The probability that x is accepted on the first pass is

$$p = \int_a^b \phi(x) dx / c(b-a).$$

The probability of acceptance on exactly the n -th pass is the same as the probability of non-acceptance on the first $(n-1)$ passes and acceptance on the n -th pass, or

$$(1-p)^{n-1}p.$$

Since

$$\sum_{n=1}^{\infty} (1-p)^{n-1}p = 1$$

the probability of eventual acceptance is one. On the other hand, the expected waiting time is

$$\sum_{n=1}^{\infty} n(1-p)^{n-1}p = \frac{1}{p}$$

From this it is clear that the speed of Method II is improved by increasing p . This can be done by first of all choosing

$$c = \varphi_{\max}$$

Secondly, $(b-a)$ can be decreased by truncating any long thin tails of $\varphi(x)$.

Under any circumstances Method II has the serious disadvantage that no useful upper bound on its running time can be given.

2.9.4. Generating the Normal Distribution

Fortunately Method I and Method II do not exhaust the possibilities for designing random number generators. As an example of the exception we shall now describe a unique method for generating random numbers from the normal distribution with mean 0 and standard deviation 1. In this case Method I is not easily applicable because there is no simple formula for the function $\varphi(x)$, and Method II is undesirable for the reasons we have already mentioned.

First of all, let us note that Method I yields an efficient random number generator for the distribution with density function

$$x e^{-x^2/2}.$$

Here

$$\Phi(x) = \int_0^x t e^{-t^2/2} dt = 1 - e^{-x^2/2},$$

and

$$\Phi^{-1}(y) = (-\ln(1-y))^{1/2}.$$

Furthermore, if we choose r from the above distribution and θ from the uniform distribution $(0, 2\pi)$, the distribution density function for the pair (r, θ) is given by

$$\frac{r e^{-r^2/2}}{2\pi}$$

Now suppose that (r, θ) are the polar coordinates of a point in the plane. Then if we change to Cartesian coordinates

$$x = r \cos \theta, y = r \sin \theta$$

the density function becomes

$$\frac{1}{2\pi} e^{-(x^2+y^2)/2} = ((2\pi)^{-1/2} e^{-x^2/2})((2\pi)^{-1/2} e^{-y^2/2})$$

Because the density factors in this way, we conclude that x and y are independent and each is normally distributed.

2.9.5 Uniform on the Unit Sphere

A variant of Method II is often used to sample from the uniform distribution on the unit sphere using the following procedure

- A. Choose three independent random numbers r_1, r_2, r_3 from the uniform distribution on $(0,1)$.
- B. Retain if $0 < \sum r_i^2 < 1$; otherwise go to A.
- C. The required unit vector is $(r_1^2 + r_2^2 + r_3^2)^{-1/2} (r_1, r_2, r_3)$.

2.9.6 Effusive Flow

Finally, we describe a modification of Method II for generating random numbers from a distribution with a density function proportional to

$$x e^{-(x-x_0)^2/c^2} \text{ on } (0,b).$$

Ideally we would like to consider this function on $(0, \infty)$, but the cut-off at b must be introduced to control the expected waiting time. This particular distribution is important because it occurs frequently in dealing with boundary conditions for the Boltzman equation.

The algorithm is as follows:

- A. Choose r from the normal distribution with mean zero and standard deviation one. Then

$$x = x_0 + r c/2^{1/2}$$

has density function

$$\frac{1}{c\sqrt{\pi}} e^{-(x-x_0)^2/c^2}$$

B. If $x \notin (0, b)$ we reject this value of x and repeat step 1. Otherwise, we proceed to C.

C. Choose a random number y from the uniform distribution on $(0, b)$. If $x \leq y$ we accept x as the required random number. Otherwise, the program goes to A to repeat the process.

We shall omit the details of the theory in this case since they follow closely our discussion of Method II. The algorithm has an unreasonably large expected waiting time if $x_0 < -c$. The waiting time can be reduced if $x_0 < 0$ by using only the positive part of the normal distribution in step 1, i.e.

$$x = x_0 + |r|c/2^{1/2}$$

If x_0 is large and negative then the standard Method II is more efficient than this one.

2.10 References

1. Breiman, L. Probability, Addison-Wesley, 1968.
2. Feller, An Introduction to Probability Theory and Its Applications, J. Wiley, New York, Vol. 1 (1957), Vol. 2 (1965).
3. Fisz, M. Probability Theory and Mathematical Statistics, John Wiley, New York, 1963.
4. Hammersely, J.M.
Handscomb, D.C. Monte Carlo Methods; Methuen's Monographs on Applied Probability and Statistics, John Wiley, New York, 1964.
5. Hull, T.E.
Dobell, A.R. Random Number Generators, SIAM Review, Vol. 4, No. 3, 1962.

2.11 List of Symbols

<u>Roman</u>	<u>First Occurrence</u>
A_n	2.1.1
$\hat{A}_k$	2.4.2
A	2.1.4
B_ℓ	2.4.2
c	2.1.3
$E(X)$	2.3
$F(A)$	2.8
$F_n(\alpha)$	2.6
$f(x)$	2.8.2
I	2.8.2
J_ℓ	2.4.2
$N(0,1)$	2.6
P	2.1.1
P_1	2.1.4
P_X	2.2.2
$\bar{r}$	2.8
$R(r_0, n)$	2.9
s, s_k	2.3
S_n	2.6
$T(k)$	2.8
$\text{Var}(X)$	2.3
V	2.7
x	2.1.3
X	2.2.1
y	2.1.3

<u>Script</u>	<u>First Occurrence</u>
$\mathcal{D}$	2.6
$\mathbb{F}$	2.1.1
$\mathbb{F}_1$	2.1.4
$\mathbb{F}_x$	2.2.2
<u>Greek</u>	
$\varphi(x)$	2.2.2
$\varphi_H(x), \varphi_{H,k}$	2.7
Φ	2.1.1
$\Phi(U), \Phi(A,k)$	2.8
$\chi_k(x), \chi_{k,i}$	2.7
μ	2.5.3
μ_k	2.7
σ	2.1.1
$\sigma(x)$	2.3
Ω	2.1.1
Ω_1	2.1.4
Ω_x	2.2.2
<u>Other</u>	
$(0,1), \bigcup_{n=1}^{\infty}, \cup, \cap, \epsilon, \{H,T\}$	2.1.1
(α, β)	2.2.1

3- THE TEST PARTICLE METHOD

3.0 Introduction

J. K. Haviland in 1961 introduced a Monte Carlo method for obtaining time independent solutions of the Boltzman equation which is now descriptively called the "test particle method". This method requires enough storage for two approximations to the distribution functions to be in memory at any one time and has therefore been limited to problems which could be reduced to one space and two velocity dimensions. The computing times are of the order of a half hour on the I.B.M. 709 to obtain a density profile with a standard deviation of the order of 1/20 of the density at any given point. In this case the density histogram was accumulated in 10 cells each having a width of the order of 1/10 mean free path.

3.1 Preliminary Theory

3.1.1 Outline

The basic idea in Haviland's method is as follows: In order to compute the distribution function $f(\vec{x}, \vec{v})$ satisfying a given Boltzman equation, and/or some of its moments in a given 3-dimensional region $\mathcal{D}$ with given boundary conditions on the boundary $\partial\mathcal{D}$ of $\mathcal{D}$. Denote the phase space for this problem by $\Omega = \mathcal{D} \times E^3$. Since the computation is designed to only produce a histogram approximating the distribution function, divide the phase space into a finite number of cells Ω_k (in practice k will represent a vector of 3 integers).

Initially one makes an educated guess (usually an appropriate linear combination of Maxwellians) for an approximate distribution function. This is considered the target distribution for the first iteration. The target distribution is then used to statistically compute trajectories in Ω of a molecule having the same collision cross-section with the target molecules as appears in the given Boltzman equation. As the trajectories are computed, their contribution to an incident distribution function is recorded. Clearly, if the initial guess was an exact solution to Boltzman's equation, one should find that as the number of trajectories increased the partially accumulated incident distribution function would tend to the target distribution function. Otherwise, the incident distribution approximation is different from the target distribution even after a large enough number of the trajectories have been computed to reduce the variance to an acceptable level.

Nevertheless, it has been found by experience that the incident distribution is a better approximation to the correct answer in the sense that if the process is repeated with the target distribution replaced by the previously computed incident distribution then eventually the incident distribution will tend to the target distribution. This is interpreted as convergence of the iteration scheme to the correct distribution (= final incident distribution = final target distribution). The required number of iterations depends on the initial guess and the error tolerances. Haviland used fewer than 5 in his calculations.

3.1.2 Trajectories to Distribution Function

To explain and motivate Haviland's procedure for accumulating the incident distribution function, suppose that $\mathcal{D}$, Ω , Ω_k , have the same meaning as before. Imagine that the molecules in $\mathcal{D}$ are numbered with an index α ($\alpha = 1, 2, \dots, N$), and have trajectories given by

$$\bar{v}_\alpha(t)$$

and

$$\bar{x}_\alpha(t) = \bar{x}_\alpha(0) + \int_0^t \bar{v}_\alpha(\tau) d\tau.$$

At this point it is worthwhile to remark that each $\bar{v}_\alpha(t)$ must closely approximate a stepfunction (piecewise constant function) in order that the Boltzman equation be valid for this system.

The first task is to define what is meant by the statement "f is the distribution of the dynamical system whose trajectories are $\bar{x}_\alpha(t)$, $\alpha = 1, 2, \dots, N$ ":

$$\frac{1}{\text{Vol}(\Omega_k)} \int_{\Omega_k} f(\bar{x}, \bar{v}) d\bar{x} d\bar{v} = \sum_{\alpha=1}^N \delta_k(\bar{x}_\alpha(t), \bar{v}_\alpha(t)) + R \quad (1)$$

where δ_k is a function having the property that $\delta_k(\bar{x}, \bar{v}) = 1$ when $(\bar{x}, \bar{v})$ belongs to the cell Ω_k and $\delta_k(\bar{x}, \bar{v}) = 0$ otherwise; and R is a random variable having mean zero and variance tending to zero as the volume of Ω_k and therefore N becomes large. Strictly speaking, one should define the probability space underlying R but it is best to avoid such a project at this point for the sake of brevity. In fact, in order to elucidate the essentials R shall be omitted henceforth and $=$ replaced with $\approx$ to indicate the omission.

Since (1) is true for all t it follows that

$$\frac{1}{T} \int_0^T dt \left(\frac{1}{\text{Vol}(\Omega_k)} \int_{\Omega_k} f(\bar{x}, \bar{v}) d\bar{x} d\bar{v} \right) \approx \sum_{\alpha=1}^N \frac{1}{T} \int_0^T \delta_k(x_\alpha(t), v_\alpha(t)) dt \quad (2)$$

It is easy to see, however, that

$$\int_0^T \delta_k(x_\alpha(t), v_\alpha(t)) dt$$

is nothing other than the transit time $\tau_{\alpha,k}$ that the trajectory of the α -th molecule spends in the k -th cell. The factor $\frac{1}{T}$ is the same for each cell, and it is more efficient in actual computation to simply accumulate the $\tau_{\alpha,k}$ -s in a suitable array and then divide by the normalizing factor at the end of the computation. Finally, there is a very important observation to be made at this point. It is not really necessary to compute the actual trajectories of N molecules in order to apply the preceding formula. It suffices to generate trajectories which have the same probability distribution as the actual ones. It must be noted that to precisely specify this

distribution would entail a certain amount of basic work which does not yet appear in the literature. One must be satisfied with describing a generation procedure which mimics statistically the one occurring in nature and arguing by analogy rather than proof.

3.1.3 Computation of Trajectories

Suppose a molecule starts at $(\bar{x}_0, \bar{v}_0)$ in phase space at time $t = 0$. Then the probability that the molecule will undergo its first collision in the time interval $(t, t+dt)$ is given by

$$P(\bar{x}_0, \bar{v}_0, t) = (\Sigma e^{-t\Sigma})dt$$

where

$$\Sigma(\bar{x}_0, \bar{v}_0) = \int d\bar{v} \int_0^b db \int_0^{2\pi} d\epsilon g f(\bar{x}_0, \bar{v}) \quad (3)$$

is the collision rate with the target molecules. As is customary, $g = |\bar{v}_r|$, and $\bar{v}_r = \bar{v} - \bar{v}_0$.

Let τ be chosen at random from a distribution with density $P(\bar{x}_0, \bar{v}_0, t)$. The position at time of collision is then $\bar{x}_1 = \bar{x}_0 + \tau \bar{v}_0$. In order to obtain a post collision velocity for the molecule reason as follows: The collision partner has a velocity chosen from a distribution with density

$$\frac{f(\bar{x}, \bar{v})}{\rho(\bar{x})} (\rho(\bar{x}) = \int f(\bar{x}, \bar{v}) d\bar{v}) \quad (4)$$

The probability that the collision parameters b, ϵ fall into the intervals $(b, b+db), (\epsilon, \epsilon+d\epsilon)$ is given by

$$\frac{bdbd\epsilon}{\pi b_m^2} \quad (5)$$

Let them be chosen accordingly. Then the post collision velocity $\bar{v}_0$ may be computed using (see 1.1.3)

$$\bar{v}_0 = \bar{v}_0 + (\bar{v}_r \cdot \hat{e}) \hat{e} \quad (6)$$

3.1.4 Specialization to One Space and Two Velocity Dimensions

Haviland considered two problems which require only one space dimension and two velocity dimensions for representing the distribution function: the flow between infinite parallel walls at two different temperatures and a plane shock extending to infinity in both directions. In both these problems the distribution function can be immediately written $f(x, v_x, v_\perp)$ (see 1.2.5).

Let the (x, u, v) -space be partitioned into cells having dimensions

$$\Delta x, \Delta v_x, \Delta v_\perp$$

The number of molecules in a typical cell is

$$\int_x^{x+\Delta x} dx' \int_{v_x}^{v_x+\Delta v_x} dv'_x \int_{v_l}^{v_l+\Delta v_l} dv'_l \int_0^{2\pi} v'_l d\theta f(x, v'_x, v'_l) \quad (6)$$

$$= 2\pi f(x, v_x, v_l) v_l \Delta x \Delta v_x \Delta v_l$$

molecules per unit area where (x, v_x, v_l) is some interior point of the cell. We now define

$$F(x, v_x, v_l) = 2\pi v_l f(x, v_x, v_l). \quad (7)$$

Suppose each cell in phase space is now indexed with subscripts (i, j, k) giving the coordinate numbers of the cell in the x, v_x and v_l directions respectively, and (x_i, v_{xj}, v_{lk}) gives the midpoint coordinates of the $i j k$ -th cell. If F_{ijk} gives the accumulated trajectory residence time in the ijk -th cell, then

$$F(x_i, v_{xj}, v_{lk}) \approx \frac{C}{\Delta x \Delta v_x \Delta v_l} F_{ijk} \quad (8)$$

where C is a normalization factor chosen so that the distribution function predicts the correct average number density of molecules. In order to compute collision rates and moments of the distribution, the following formula given by Haviland is very useful

$$\begin{aligned} & \int f(x, v_x, v_y, v_z) \phi(x, v_x) dv_x dv_y dv_z \\ &= \int_{-\infty}^{\infty} dv_x \int_0^{\infty} dv_l \int_0^{2\pi} v_l d\theta f(x, v_x) v_l \cos\theta, v_l \sin\theta \phi(x, v_x, v_l \cos\theta, v_l \sin\theta) \\ &= \int_{-\infty}^{\infty} dv_x \int_0^{\infty} dv_l F(x, v_x, v_l) \int_0^{2\pi} d\theta \phi(x, v_x, v_l \cos\theta, v_l \sin\theta) \quad (9) \\ &\approx \frac{1}{2\pi} \sum_{j,k} F_{ijk} \int_0^{2\pi} d\theta \phi(x_i, v_{xj}, v_{lk} \cos\theta, v_{lk} \sin\theta) \end{aligned}$$

where x_i is the closest cell midpoint to x .

3.1.5 A Finite Velocity Range is Sufficient

Since it is expected that for large values of v_x, v_l the distribution function will decay like $\exp[-(v_x/c_x)^2 + (v_l/c_l^x)^2]$, only the cells near the mean velocity should contain any significant number of molecules. This leads to the very important simplification of working with only a finite number of cells in velocity space. Haviland (p.151, table IV) has studied the effects of restricting the velocity distribution to a rectangle of the form

$$(u_x - ac_0, u_x + ac_0) \times (u_l - ac_0, u_l + ac_0)$$

(where $(u_x, u_\perp)$ is the mean velocity and c_0 is the most probable deviation from the mean) for various values of α . He points out that if $\alpha = 2.0$ then one is assured of including over 99% of the molecules of an equilibrium distribution. Any molecules found outside the rectangle are simply assigned to the closest cell on the boundary of the rectangle.

3.2 The Algorithm

3.2.1 Storage Requirements

The largest arrays are the target distribution function FT; the incident distribution function FI each of dimensions I,J,K, a cross-section array XSECT of dimensions K, $(2J - 1)$, K and a collision rate array SIGMA of dimensions I,J,K. Both the cross-section and collision rate arrays are used to save time by storing functions which would otherwise have to be recomputed a large number of times. Haviland used $I = 10$, $J = 24$, $K = 12$ in all of his calculations.

3.2.2 Data and Formulas

In the case of flow between parallel plates at different temperatures with perfect accommodation at the walls, there are two non-dimensional parameters - the temperature ratio of the walls and the distance between them in mean free paths. This is also the stage at which any constants which will be used frequently during the program should be read in or computed. They include the number $r_{\max}$ of iterations necessary to get convergence of the distribution function, the number $n_{\max}$ of cell transits necessary to accumulate the next approximation to the distribution function, and the array XSECT. In Haviland's calculations $r_{\max} \sim 3$ and $n_{\max} \sim 50,000-100,000$.

The formulas for both XSECT and SIGMA are derived from

$$\Sigma(x, v_x, v_\perp) = 1/2 \int_{-\infty}^{\infty} dv'_x \int_0^{\infty} dv'_\perp F(x, v'_x, v'_\perp) \int_0^{2\pi} d\theta g b_m^2 \quad (10)$$

which is obtained by using (3), (7), and (9). The evaluation of

$$\int_0^{2\pi} d\theta g b_m^2$$

at the cell midpoint values $(v_{xj}, v_{\perp k})$ and $(v'_{xj}, v'_{\perp k})$ yields

$$\begin{aligned} & \text{XSECT}(k, j'-j+J, k') \\ &= 1/2 \int_0^{2\pi} b_m^2 [(v_{xj} - v'_{xj})^2 + (v'_{\perp k} \cos \theta' - v_{\perp k} \sin \theta)^2 + (v'_{\perp k} \sin \theta' - v_{\perp k} \cos \theta)^2]^{1/2} \end{aligned}$$

A couple of remarks pertain to the preceding formula. Firstly, since

$$(v'_{\perp k} \cos \theta' - v_{\perp k} \cos \theta)^2 + (v'_{\perp k} \sin \theta' - v_{\perp k} \sin \theta)^2 = v_{\perp k}^2 + v_{\perp k}^2 - 2v'_{\perp k} v_{\perp k} \cos(\theta' - \theta),$$

and the integration is over a full period of the cosine function, the integral is independent of θ' . Secondly, since the cells in the v_x direction are equally spaced, $v_{xj} - v_{xj}$ will depend only on $j' - j$. This is the justification for writing XSECT ($k, j' - j + J, k'$) instead of XSECT (k, j, j', k'). The J has been thrown in for the sole purpose of keeping indices positive - a requirement of the FORTRAN language in which this program was originally written. It is also worthwhile to note that SIGMA should be computed after FT which changes with each iterate.

Initially, the starting target distribution function is stored in FT. For this purpose, it is a good idea to use the best analytical approximation which is known and reasonably simple. Haviland used the free molecule solution for the heat transfer problem and the bimodal Mott-Smith distribution in the shock problem. If

$$f(x, v_x, v_{\perp} \cos \theta, v_{\perp} \sin \theta) = f(x, v_x, v_{\perp})$$

is the initial analytic approximation, we begin by setting

$$FT(i, j, k) = 2\pi \int_{x_i - \Delta x/2}^{x_i + \Delta x/2} dx \int_{v_{xj} - \Delta v_x/2}^{v_{xj} + \Delta v_x/2} dv_x \int_{v_{\perp k} - \Delta v_{\perp}/2}^{v_{\perp k} + \Delta v_{\perp}/2} dv_{\perp} v_{\perp} f(x, v_x, v_{\perp})$$

for the interior cells. For boundary cells, the tail of the distribution must be included. For example, if $j = J$ above then

$$\int_{v_x - \Delta v_x/2}^{v_x + \Delta v_x/2}$$

is replaced by

$$\int_{v_{xj} - \Delta v_x/2}^{\infty}$$

3.2.3 Normalization

The normalization constant C_F is computed from the formula

$$n_m \Delta x = C_F \sum_{ijk} F(i, j, k)$$

where n_m is the mean number density. Note that the left side of this formula is nothing other than the total number of molecules per unit area (since the system is supposed to extend to ∞ in the y and z directions). Any moments which one wishes to compute and output can be dealt with simply once the normalization constant for the distribution function is known.

3.2.4 Trajectory Initialization

The first task which must be performed even before any trajectories are considered is the computation of the array SIGMA

according to the formula

$$\text{SIGMA}(i,j,k) = (C_F/\Delta x) \sum_{j',k'} \text{FT}(i,j',k') \text{XSECT}(k,j'-j+J,k').$$

The actual computation of a trajectory starts with the molecule at either the left boundary or right boundary having velocity (v_x, v_z) chosen according to the boundary conditions and is repeated every time the molecule crosses a boundary. For many kinds of boundary condition the v_x, v_z are chosen from an appropriate probability distribution. The matter of boundary conditions is a study in itself, and the present discussion is restricted to equilibrium flow normal to a plane and to emission from a perfectly accommodating wall.

Let

$$\phi(u,c;v_x,v_z) = C_\phi v_x \exp\left(-((v_x-u)^2 + v_z^2)/c^2\right)$$

where C_ϕ is chosen so that $\int \phi dv_x dv_z = 1$, and c is the most probable speed after the mean is subtracted out. In the case of a flow in the x direction of speed u , the probability that a molecule which crosses a fixed plane normal to the flow has velocity in the range $v_x, v_x+dv_x; v_z, v_z+dv_z$ is given by $\phi(u,c;v_x,v_z) dv_x dv_z$. In the case when the molecule is emitted from a perfectly accommodating wall the probability is $\phi(0,c;v_x,v_z) dv_x dv_z$. Efficient procedures for choosing v_x, v_z at random from a distribution with density function ϕ have already been described (2.9.6).

3.2.5 Cell Coordinates

Once the velocity components of the test molecule are chosen, its cell indices i,j,k , are computed for future reference, but the actual coordinates are not changed to the cell midpoint values. The molecule now has two choices: to cross a cell division, (crossing the calculation field boundary is a subcase of this) or to undergo a collision within its cell. In order to decide which occurs, the program computes τ_c , the time to next collision and τ_d the time to cross a cell division. The expression for τ_d follows from a simple geometrical exercise.

As a basis for discussing the computation of τ_c refer to (10). The array SIGMA has already been computed and the indices i,j,k are computed in this section. Choose τ_c from the distribution whose density is

$$\text{SIGMA}(i,j,k) e^{-t \text{SIGMA}(i,j,k)}$$

using Method I.

If $\tau_c \geq \tau_d$, 2.6 is applied to move the molecule into the next cell. Otherwise, 2.7 is used to compute a collision.

3.2.6 Translation

The x coordinate of the molecule is changed to the appropriate cell division value and τ_d is added to FI (i,j,k) . If the molecule

happened to end up at the edge of the x interval 2.5 is applied to start a new trajectory. Otherwise, the collision/transit calculation is repeated in the new cell.

3.2.7 Collision

To start with, τ_0 is added to $FI(i,j,k)$. Then the cell (indices j',k') where the collision partner will originate is chosen in accordance with the unnormalized probability function

$$FT(i,j',k') \text{ XSECT } (k,j'-j+J,k').$$

Although Haviland used Method I to make his choice of j',k' it seems that in this case Method II would be faster, even though it would entail finding the maximum values of FT and $XSECT$ in 2.4 in order to determine a suitable range for the second random number.

Once the choice of j',k' is made, the collision partner is assumed to have velocity components (v'_x, v'_y) equal to the cell midpoint velocity coordinates. In order to obtain cartesian components, choose θ' from the uniform distribution on $(0,2\pi)$ and let

$$v'_y = v'_x \cos \theta', \quad v'_z = v'_x \sin \theta'.$$

The most efficient way to compute the post collision velocity $\bar{V}$ of the incident molecule depends to a large extent on the intermolecular interaction. In the case of hard spheres, where it is known that the post-collision relative velocity (1.1.3) has direction uniformly distributed on the surface of the unit sphere, it is fastest to choose a unit vector $\hat{w}$ according to 2.9.5 whereupon

$$\bar{V}_r = |\bar{v}' - \bar{v}| \hat{w}$$

and (see 1.1.1)

$$\bar{V} = 1/2(\bar{v} + \bar{v}' - \bar{V}_r)$$

In general, when the distribution of $\hat{w}$ is not explicitly known, the general theory of Chapter 1 must be used to compute collisions.

3.2.8 Exit Criteria

The collision-transit computation (2.5, 2.6, 2.7) is repeated until the total number of collisions plus transits reach a preassigned number sufficiently large for the required standard deviation in the results (see 3.3). The attainment of this number signals the end of the iteration and the beginning of the next one or the completion of the calculation.

In proceeding to the next iteration FT is stored in FI . The need for an actual storing operation can be eliminated by a suitable arrangement of the program and it is best to refer to Haviland's original flow diagrams to see how this is done.⁽¹⁾

This completes the description of the algorithm.

3.3 Variances

3.3.1 Variance of FI (i,j,k)

Haviland's method lends itself easily to a theoretical estimate of the variance of the answers it produces. In order to obtain this estimate, Haviland did not analyse his algorithm directly but instead assumed a model which is supported by the conclusions it produces when the distribution function is Maxwellian. The model is along the following lines: Suppose that each $FI(i,j,k)$ is the sum of N independent random variables X_{α} each taking the value τ with probability F_{ijk} and 0 with probability $1 - F_{ijk}$ where

$$\sum_{ijk} F_{ijk} = 1.$$

Then

$$E(X_{\alpha}) = \tau F_{ijk}$$

and

$$\text{Var}(X_{\alpha}) = E(X_{\alpha}^2) - (E(X_{\alpha}))^2 = \tau^2 F_{ijk} (1 - F_{ijk})$$

It follows that

$$E(F(i,j,k)) = E(\sum_{\alpha} X_{\alpha}) = N\tau F_{ijk}$$

and

$$\text{Var}(\sum_{\alpha} X_{\alpha}) = N\tau^2 F_{ijk} (1 - F_{ijk}).$$

(See 2.2.7)

In practice

$$\hat{F}_{ijk} = FI(i,j,k) / \sum_{ijk} FI(i,j,k)$$

is used as an estimate of F_{ijk} . For reasons of simplicity however, suppose τ is known and one can therefore use

$$F_{ijk} = \frac{1}{\tau N} FI(i,j,k)$$

Then

$$E(\hat{F}_{ijk}) = F_{ijk}$$

as required and

$$\text{Var}(\hat{F}_{ijk}) = \frac{1}{N} F_{ijk} (1 - F_{ijk})$$

3.3.2 Variance of the Moments

If $\psi(x, v_x, v_{\perp})$ (or $\psi(v_x, v_{\perp})$) is a given function, then the moment of ψ with respect to the distribution $F(x, v_x, v_{\perp})$ is defined to be

$$\int \psi(x, v_x, v_z) F(x, v_x, v_z) dx dv_x dv_z$$

(or $\int \psi(v_x, v_z) F(x, v_x, v_z) dv_x dv_z$ respectively)

Consider only the first case which, in its discrete version, will turn out to contain the second. The discrete approximation for the moment of ψ is given by

$$\hat{\psi} = \sum_{i,j,k} \psi_{i,j,k} \hat{F}_{ijk}$$

and is again a random variable. Then

$$E(\hat{\psi}) = \sum_{i,j,k} \psi_{ijk} F_{ijk}$$

and

$$\text{Var}(\hat{\psi}) = \frac{1}{N} \sum_{i,j,k} \psi_{ijk}^2 F_{ijk} (1-F_{ijk})$$

This is not the same formula for the variance of as derived by Haviland, because the initial model was slightly different. It has the advantage of predicting a larger variance than Haviland's formula; this seems to be desirable in view of the fact that Haviland's theoretical variances turned out to be smaller than his experimental (computed) ones.

3.4 Conclusion

Haviland's method was the first attempt to solve the Boltzman equation by a Monte Carlo technique and consequently has played an important role in the development of the subject. Its storage and computing time requirements pretty well limit its applicability to the two types of problem originally treated by Haviland. It does have the advantage, however, of being amenable to a close mathematical analysis (which at the time of writing has not been fully carried out) yielding a proof that statistically computed quantities have the required mean, and estimates of the variances of these quantities. It is fairly reliable as such methods go, and so far the only unexplained predictions have been regular density fluctuations in the high temperature high density regions of a flowfield.

3.5 References

1. Haviland, J.K. The Solution of Two Molecular Flow Problems by the Monte Carlo Method, Methods in Computational Physics (Ed. Alder), Vol. 4, 1965, Academic Press.
2. Sinclair, A.M.R. UTIAS Report No. 130 (1968), University of Toronto, Toronto, Canada.
3. Starr, A.W. Monte Carlo Simulation of the Flow of a Rarefied Gas Around a Cylinder, MIT Lincoln Laboratory Technical Note 1967-7.

3.6 List of Symbols

<u>Roman</u>	<u>First Occurrence</u>
b, b_m	3.1.3
c_x, c, c_o	3.1.5
c_F, c_ϕ	3.2.3
$\mathcal{D}, \partial\mathcal{D}$	3.1.1
E	3.3.1
E^3	3.1.1
$f(\bar{x}, \bar{v})$	3.1.2
$F(x, v_x, v_\perp), F_{ijk}$	3.1.4
FT, FI	3.2.1
$FT(i, j, k)$	3.2.2
I, J, K	3.2.1
N	3.1.2
n_m	3.2.3
$P(\bar{x}_o, \bar{v}_o, t)$	3.1.3
$SIGMA$	3.2.1
$SIGMA(i, j, k)$	3.2.4
T	3.1.2
$v_\alpha(t), vol(\Omega)$	3.1.2
$\bar{v}_o$	3.1.2
$v_\perp, \bar{v}_r, v_x, v_y, v_z, \bar{v}_o, v_{\perp 1}, v_{\perp k}$	3.1.4
Var	3.3.1
$\hat{w}$	3.2.7
$\bar{x}(t)$	3.1.2
$\bar{x}_o, \bar{x}$	3.1.3
x_1, x_i	3.1.4
$XSECT$	3.2.1

RomanFirst OccurrenceXSECT ($k, j' - j + J, k'$)

3.2.2

 X_α

3.3.1

Greek ε

3.1.3

 δ_k

3.1.2

 $\Delta x, \Delta u, \Delta v, \phi(x, v_x, v_y, v_z)$

3.1.4

 $\phi(u, c, v_x, v_1)$

3.2.4

 $\Psi(v, v_x, v_1), \psi_{ijk}$

3.3.2

 $\rho(x), \Sigma(\bar{x}_0, \bar{v}_0)$

3.1.3

 $\tau_{\alpha, k} \quad \tau$

3.1.2

 τ_c, τ_d

3.2.5

 τ

3.3.1

 θ

3.1.4

 Ω

3.1.1

4. Monte Carlo Simulation Method

4.1 General Description

The normalizability (1.3.1) of the Boltzman equation suggests that an actual flow involving in the vicinity of 10^{20} molecules could be studied by means of a model using a small number of molecules, each of proportionally larger cross section. There are a number of difficulties which arise from such a trade-off. The higher probability of multiple collisions must be dealt with in the model and the statistical fluctuations arising from the small sample size necessitate averaging a large number of observations to obtain a useful approximation to the distribution function.

Nevertheless, G.A. Bird and others have had considerable success using a model with as few as 1000 molecules in conjunction with a Monte Carlo procedure for obtaining post collision velocities consistent with the cross-section function.

Bird's method is an algorithm in which each iteration operates on a sample of N (of the order of a thousand) molecules from a distribution with distribution function

$$f(\bar{x}, \bar{v}, t)$$

to generate a sample from a distribution with density function

$$f(\bar{x}, \bar{v}, t + \Delta t)$$

where f satisfies the Boltzman equation and suitable boundary conditions. Although the foregoing statement has never been precisely proved, the method has found extensive application in the study of problems in the transition regime where the mean free path is of the same order of magnitude as the flow properties under study. In almost all cases, the results are quite reasonable. The method is applicable in both time dependent and time independent problems. In the case of time dependent problems it is necessary to repeat the calculation using different random number sequences a sufficient number of times to reduce the standard deviation of any sampled quantities to an acceptable level. In the case of time independent problems the usual procedure is to start with an arbitrary distribution function $f(\bar{x}, \bar{v}, 0)$ and to run the iteration without sampling until f has converged sufficiently to its steady state value $f(\bar{x}, \bar{v}, \infty)$. Then sampling is done at regular intervals until a sufficient sample size is reached.

4.2 The Algorithm

4.2.1 Storage Layout and Initialization

Each molecule is represented in computer memory by storing its velocity and position coordinates. This requires $3 + 3 = 6$ locations but in many cases geometrical symmetry allows a reduction in the number of position coordinates which must be stored. Many physically interesting problems (for example, the flow around spherically or cylindrically symmetrical obstacles) require only two

position coordinates; and some studies on the relaxation of the distribution require no position coordinates. Space is assigned in the fast access memory of the computer to accommodate the coordinates of up to $N_{\max}$ simulator molecules.

There remains the problem of referencing the molecules in such a way as to make those in an arbitrary region of position space easily accessible. This is accomplished by dividing position space into cells. Some notation will be helpful later. Let the flow region under study be denoted by D and subdivided into a finite number K of subregions D_k which will be called cells. The cells must be small enough so that the velocity distribution function is approximately constant on each one. This judgement must be made before the solution to the problem is known, but experience has shown that intuition is an adequate guide in setting up the cell size. Factors affecting the choice of cell size are considered in Section 3.3.

Once the cell boundaries have been decided the programming problem of easily referencing the subset of molecules in each cell can be tackled. There are several well known ways of doing this (refer to any work on list processing) of which one possibility is to set up K auxiliary tables in such a way that the k -th table contains the addresses or sequence numbers of the molecules in the k -th cell. The method described presently combines access speed with storage economy.

Suppose cell k contains n_k molecules, $1 \leq k \leq K$. Then the molecules from cell 1 are stored in locations 1 to n_1 of the molecule list; from cell 2 in locations $n_1 + 1$ to n_2 etc. A vector MAP with K entries containing the numbers $n_1, n_1 + n_2, \dots, n_1 + n_2 + \dots + n_K$ is also stored. Whenever an addition, deletion or move occurs, only the last molecule in each cell section is moved and MAP is updated to describe the new configuration.

Preliminary to starting the iteration, a number $N_0 < N_{\max}$ of molecules is chosen from an appropriate starting distribution and stored in the molecule table of the programme. Any additional information necessary to reference the molecules, a table of cell volumes, cross-section constants, boundary condition parameters, etc. must also be assembled at this stage. The appropriate starting distribution depends on whether the algorithm will be applied to a steady or unsteady flow problem and is discussed in Section 3.1.

4.2.2 Steps Comprising One Iteration

A. Collision Increment to the Distribution Function

The following procedure is repeated to each cell k , $1 \leq k \leq K$. A pair of molecules is chosen at random from those in cell k in such a way that the probability of choosing a given pair is proportional to the relative velocity. To avoid computing $1/2n_k(n_k-1)$ relative velocities, it is convenient to estimate the largest relative velocity $s_{\max}$ which can occur with any frequency and to make the choice as follows: Choose two independent integers from the uniform distribution on

$$(1 + \sum_{i=1}^{k-1} n_i, \sum_{i=1}^k n_i),$$

rejecting the pair and repeating the choice if the two integers turn out to be equal, and compute the relative speed $|\vec{v}_r| = g$ of the accepted pair. Choose a random number r from the uniform distribution on $(0, s_{\max})$, accept the pair for collision if $r < g$ and otherwise reject, repeating the procedure from the choice of two random integers. As soon as a pair is accepted for collision the collision parameters are chosen at random and the velocity vectors of the pair in question are replaced with post collision velocity vectors computed on the basis of the collision parameters. Details of this calculation are given in Ch. 1.1 (see also Ch. 2.9.5).

After each collision a time counter t is advanced according to the formula

$$\Delta t = \frac{2 \text{ vol}_k}{A n_k^2 g}$$

where A is the cross-section per simulator molecule, and vol_k is the volume of cell k . Collisions are computed until the total time advance exceeds a preset value Δt_m . Both the time advance formula and the value of Δt_m raise a number of delicate questions and are discussed in sections 3.5 and 3.3 respectively. Once the collision calculation has been completed, the programme proceeds to B.

B. Convection Increment to the Distribution Function

Each molecule is moved to a new location corresponding to the time interval Δt_m . This means, of course, that its position coordinate $\bar{x}$ is replaced with $\bar{x} + \bar{v}\Delta t_m$ in case of cartesian coordinates. It may happen that this displacement of the molecule takes it outside the region Ω , across a cell boundary, or across the boundary of some obstacle in the flow. In each of these cases an appropriate strategy must be applied. For instance, crossing an exterior boundary entails deletion of the molecule from the list; crossing a cell boundary necessitates appropriate changes in the bookkeeping tables; and collision with an obstacle entails replacement with a molecule chosen from a distribution appropriate to the boundary condition (see Section 3.6). For example, the molecule may be reflected, emitted diffusely, or it may choose one of these fates with probability α . When the whole molecule list has been dealt with, the programme proceeds to C.

C. Influx of Molecules

Facilities must be provided in the programme for introducing molecules into the calculation which enter the region because of the flow. These are usually introduced with their position coordinates consistent with entry into the region at some random time in the interval $(t, t + \Delta t_m)$, and velocity coordinates appropriate to the incoming distribution. The boundary point of entry is chosen at random when no better criterion is available and the number of molecules

introduced in this way is derived from the theoretical input flux and the elapsed time Δt_m (details in Section 3.6).

D. Data Collection

The description to this point constitutes one iteration. References to data collection have been omitted purposely in order to make the description of the main steps as concise as possible. Basically, data collection falls into two categories; estimation of various transfer rates across specified boundaries in the flow, and estimation of moments of the distribution function at fixed time. Sampling for the first category is most easily carried out as part of step B, and for the second category at this point in the program. The latter is not normally done after every iteration and the actual strategy is dependent upon whether a steady or unsteady flow is being simulated as described in Section 3.1.

E. Exit Criterion

When the programme has reached this point, two alternatives are possible: commence another iteration at A or prepare and output final data. The decision can be made either on the basis of the total number of iterations or the accumulated time advance. The number of iterations necessary to accumulate the required data depends on the acceptable standard deviation of the answer (see Section 3.7).

4.3 Analysis

4.3.1 Steady and Time Dependent Problems

The treatment of steady and time dependent problems differs in the choice of the initial distribution function and in the sampling strategy. In the case of steady problems a good approximation to the actual distribution function is seldom known. Consequently the initial distribution function is seldom known. Consequently the initial distribution for the iteration is usually taken to be a drifting Maxwellian of approximately the right velocity, temperature, and density. This initial distribution is expected to converge to the final steady state distribution fairly quickly. An early study done by Bird [3] suggests that convergence takes three to four collision times. Sampling is delayed until convergence has occurred and then carried out at regular intervals until a sufficient number of observations has been made. If the sampling interval is too short then the successive observations are not independent and consequently little is gained in the way of reduced standard deviation. On the other hand, if the sampling interval is too long, there is a waste of computer time. The proper choice of interval depends on many factors (see 4.3.7) but generally speaking, it has been found that $1/4$ to 1 collision time is a satisfactory range. In the case of time dependent problems, the initial distribution function is assumed known and can therefore be used as a source distribution for the initial sample of molecules. Sampling is done at those times when a snapshot of the flowfield is desired, and the iteration is continued until the flow has developed sufficiently. In order to reduce the

standard deviation of the results to an acceptable level it is necessary to repeat the calculation a large number of times using a different random number sequence each time, and to average corresponding results.

4.3.2 Cell Size and Δt_m

From comments in the literature [6, Fig. 2] and from the author's own experience it seems that a wide range is available for both the cell diameters and the time advance interval without significantly altering the results. There are extremes, however which should be avoided. Since the method treats the distribution function as if it were constant over each cell, the cell must be small enough to accommodate steep gradients in the flow properties such as occur in a shock wave. Similarly Δt_m must be small enough to avoid gross changes in the distribution function caused by either the collision or the convection increment over the time interval Δt_m . There is also a rough lower bound on the cell size and on Δt_m . This is because enough collisions must be computed in each cell during Δt_m to assure that the computed time increment is fairly close to Δt_m . If cell k contains n_k molecules then approximately $1/2 n_k$ collisions will be computed per collision time. Thus, if $\Delta t_m = 1/3$ collision times and $n_k = 30$, 5 collisions will be computed during each time interval. The applications of this method thus far carried out have used 30 or more molecules per cell, although Bird reports in one of his papers [6] that satisfactory results are obtained with as few as six molecules per cell.

4.3.3 Storage Requirements

The fast access storage capacity of the computer limits the number of molecules that can be used in the simulation. The programs designed by the author have used 500-700 Fortran instructions leaving space for close to 4000 5-coordinate molecules when run on the University of Toronto IBM 7094 II computer. A typical program divides naturally into an initialization, iteration, and output section of which only one needs to be in fast access storage at one time, thereby reducing further the storage requirements for program.

A general principle is that the larger the number of simulator molecules, the more reliable the results. In the case of time independent problems there is also a disadvantage in using a large number of molecules because computing time per collision time increases linearly with the number of molecules and a fixed number of collision times is required to achieve convergence to the steady distribution. This unproductive computing time can be minimized in a series of runs where the parameters are incremented only a small amount each run by using the final sample from one run as the initial sample for the next run, thereby reducing the convergence time.

4.3.4 Computing Time Requirements

For purposes of estimating the actual running time of a given program, the basic parameters are τ_c , the time required to compute one collision, τ_T , the time required to move one molecule and check boundaries, and μ the number of Δt_m -s. per collision time. Since approximately $1/2 N$ collisions must be computed per collision time (N is the total number of molecules and each collision computation accounts for two molecules), it is easy to see that an approximate formula for the running time of the algorithm is

$$\text{time} = 1/2 N \tau_c + N \mu \tau_T = 1/2 N (\tau_c + 2 \mu \tau_T)$$

Since the number of observations of a given variable varies as the product of the number of molecules in the flowfield and the number of collision times involved, it follows that the computing time per observation is constant for a fixed variable.

Unfortunately, the best data available in the literature gives only the computing time necessary to compute a certain number of collisions. Consequently it is only possible to determine an approximate value for the combination $\tau_c + 2 \mu \tau_T$ rather than for the individual parameters. The table below yields a rough idea of the times involved.

<u>Reference No.</u>	<u>Computer Used</u>	<u>Computing time/1000 collisions</u>
(3)	Silliac, U. Sidney	$60 \times 1000/30000 = 2 \text{ min.}$
(8)	IBM 7040/32K	$900 \times 1000/2 \times 10^6 = .45 \text{ min.}$
(12)	CDC 6600	$20 \times 1000/6 \times 10^5 = .03 \text{ min.}$
Author	IBM 360/65	$3 \times 1000/10000 = .3 \text{ min.}$

4.3.5 The Time Increment Formula

It is essential in a simulator of the type we have been discussing that collisions occur at the right rate relative to the motion of the molecules, and that each pair of molecules have the correct probability of undergoing collision. To be more specific, consider a cell Ω_k in position space of volume $\text{vol}(\Omega_k)$ and containing n_k molecules. This means that Ω_k contains $1/2 n_k(n_k-1)$ distinct pairs of molecules. Let g_{ij} denote the relative speed of molecule i and molecule j . If the collision cross-section is A , the pair must be found in a volume $g_{ij} A \Delta t_m$ for a collision to occur during time Δt_m . Since it is already known that the pair is found in $\text{vol}(\Omega_k)$, the probability that the pair i, j undergoes collision during time Δt_m is

$$P_{ij} = \frac{g_{ij} A \Delta t_m}{\text{vol}(\Omega_k)}$$

and the probability of a collision is

$$\sum_{i < j} P_{ij},$$

where there are $L = 1/2 n_k(n_k-1)$ terms in the sum. Let the probabilities above be renumbered from 1 to L so that given i, j there exists an ℓ such that $P_\ell = P_{ij}$, and let $q_\ell = 1 - p_\ell$. This means that the probability h_m of exactly m collisions is the sum of all products of the form

$$r_1 r_2 \dots r_L,$$

where $r_\ell = p_\ell$ or q_ℓ and there are exactly m p_ℓ -s in each term.

On the basis of the foregoing theory, the choice of collision partners could proceed in one of two ways:

A. Accept pair ℓ for collision with probability p_ℓ and repeat for all possible pairs $\ell = 1, \dots, L$. Since in practice there are about 100 molecules per cell yielding roughly 50,000 pairs of which only a small fraction collide during time Δt_m , this procedure would turn out to be prohibitively time consuming.

B. Decide the number of collisions at the start by choosing a positive integer m from the distribution h_m and then choose m successive pairs from the distribution

$$\hat{p}_\ell = \frac{p_\ell}{\sum_{\ell=1}^L p_\ell}.$$

This method, too, is unfeasible because it requires the computation of both p_ℓ and h_m which is out of the question.

The method proposed by Bird is to choose ℓ at random in the range $(0, L)$ and r from the uniform distribution on $(0, p_{bd})$ where p_{bd} is any number larger than all the p_ℓ -s. The integer ℓ is accepted if $r < p_\ell$; otherwise another choice is made. Once ℓ is chosen, a counter c is advanced by $1/L$ and the collision computation is complete as soon as c surpasses 1.

It will be shown presently that this procedure gives the correct expected number of collisions, but in general the wrong distribution. To derive the expectation, let X_ℓ $\ell = 1, \dots, L$ be random variables defined by

$$X_\ell = \begin{cases} 1 & \text{when pair } \ell \text{ collides} \\ 0 & \text{otherwise.} \end{cases}$$

Then

$$S = \sum_{\ell=1}^L X_\ell$$

gives the total number of collisions in cell k during time Δt_m .

Furthermore

$$E(S) = \sum_{\ell=1}^L E(X_{\ell}) = \sum_{\ell=1}^L p_{\ell}$$

which is the actual expectation. In order to find the Bird expectation, observe that the criterion for ending the collision computation is a first passage in a random walk with nonzero mean (see Ref. 2, 2.10). In such a process the expectation of the time to first passage is given by

$$\frac{E(\tau \text{ at time of first passage})}{E(\text{each increment})}$$

Since in practice the increments are much smaller than 1, it is safe to assume that the numerator will be very close to 1. In order to compute the denominator, note that the Bird procedure for choosing ℓ yields an integer from the distribution $(\hat{p}_{\ell})$. Then

$$E(\text{increment}) = \sum_{\ell=1}^M \frac{1}{Lp_{\ell}} \hat{p}_{\ell} = \frac{L}{L \sum p_{\ell}} = \frac{1}{\sum p_{\ell}}$$

from which it follows that on the average

$$\frac{\sum_{\ell=1}^L p_{\ell}}{1}$$

choices are required to advance the parameter c by one unit. In order to show that in general the Bird technique does not produce the correct distribution for the number of collisions it is sufficient to display a counter example. Consider a hypothetical extreme case in which $p_{\ell} = p$ $\ell = 1, \dots, L$. Then the probability that m collisions occur is

$$\binom{L}{m} p^m q^{L-m},$$

$\binom{L}{m}$ being the binomial coefficient, whereas the Bird method would advance c by $1/Lp$ for each collision leading to the probability

$$P(m) = \begin{cases} 1 & \text{if } m = [Lp] + 1 \\ 0 & \text{otherwise} \end{cases}$$

where $[Lp]$ denotes the greatest integer less than or equal to Lp .

4.3.6 Boundary Strategies

There are two commonly occurring boundary situations which must be handled in a Monte Carlo simulation.

a) An imaginary boundary in a drifting Maxwellian flow. That is, a flow whose distribution function at each point x is given by

$$f(v) = n_x (c \sqrt{\pi})^{-3} \exp(-c^2 \sum_{i=1}^3 (v_i - u_i)^2) \quad (1)$$

where the vector u with components u_i $i = 1, 2, 3$ specifies the velocity of the flow, c is a constant related to the temperature of the gas, and n_x is the local number density. Any molecules reaching such a boundary from the interior of the region are simply abandoned. At the end of each time interval Δt_m , it is necessary to introduce a number of molecules representing those that have crossed the boundary into the region during this time. To be specific, consider an area element δA with interior normal I in such a flow. Let (I, J, K) be an orthonormal frame such that J lies in the plane of I and the velocity vector u of the flow. Then u can be resolved as follows

$$u = u_I I + u_J J. \quad (2)$$

The molecules crossing δA into the region have velocity components in the I, J, K directions which are independently distributed and have density functions

$$\begin{aligned} (1/c \pi^{1/2}) v_I \exp(-(v_I - u_I)^2/c^2) \\ (1/c \pi^{1/2}) \exp(-(v_J - u_J)^2/c^2), \end{aligned} \quad (3)$$

and $(1/c \pi^{1/2}) \exp(-(v_K - u_K)^2/c^2)$

respectively, modulo normalizing factors. The total flux across δA is given by

$$\phi = n_x \int_0^\infty (v/c \pi^{1/2}) \exp(-(v - u_I)^2/c^2) dv \quad (4)$$

In practice, n_x can be determined from the formula

$$n_x = (\lambda A_m)^{-1}, \quad (5)$$

where $2^{-1/2} \lambda$ is the local mean free path, and A_m is the cross-section per simulator molecule. Then

$$\phi \delta A \Delta t_m$$

gives the total number of simulator molecules entering the region through δA during time Δt_m . In practice, a time of entry is chosen from the uniform distribution on $(0, \Delta t_m)$, a point of entry is chosen from the uniform distribution on δA , and velocity components v_I, v_J, v_K are chosen from the distribution described in (3) (Sec. 2.9.5). The molecule is then stored with the position coordinates it would have at the end of the time interval Δt_m .

b) Surface interactions. The most common surface interaction models to date have been specular and diffuse reflection. In the case of specular reflection, a molecule which collides with a boundary during time Δt_m , has its velocity vector replaced with the reflected vector at time of collision and moves with the new velocity for the remainder of the time interval. The reflected velocity $\bar{v}'$ is given in terms of the incident velocity $\bar{v}$ by

$$\bar{v}' = \bar{v} - 2(\bar{v} \cdot I)I$$

where I is the normal at point of impact. In the case of diffuse reflection, the velocity vector is replaced by one chosen from the distribution (3) (2.9.6) in which $u = 0$ and c is characteristic of the surface temperature.

4.3.7 Standard Deviation and Limitations

The basic theorem relevant to the discussion of the standard deviation in any Monte Carlo method asserts that the standard deviation of an average of independent random variables decreases as the inverse square root of the number of variables. The hypothesis of independence is essential for it is even intuitively clear that in the simulation of a sample of 1000 molecules for one collision time with regular observations of the distribution function during this time there comes a point after which nothing more is gained by increasing the frequency of observations. This is because eventually successive observations will cease to be independent of one another. This issue only arises of course in sampling moments of the distribution function at regularly spaced intervals. It does not arise in the accumulation of information on boundary crossings for the purposes of estimating fluxes because successive boundary crossings are automatically independent.

There are as yet no studies in the literature of how the standard deviation of a typical moment depends on the sampling interval. The only work remotely relevant is an early publication by Bird (3) on the time taken for the distribution function to relax to a Maxwellian form. It was of the order of 3 collision times. This is longer than the sampling interval used by most authors in the field where $1/4$ to 1 collision times has appeared to be satisfactory for most moments. The price paid for the wrong sampling interval is the wrong theoretical estimate for the standard deviations and wasted computer time. If the interval is too short, time is wasted in taking needless samples, and if too long, time is wasted in computation between samples.

At present state of development, Bird's Monte Carlo method cannot cheaply yield a satisfactory approximation to the complete distribution function of even a simple practical problem. To see this, consider the flow around a cylinder at Knudsen number 1 and make the modest demand that the distribution function be accumulated on a grid having 10 points in each direction, leading to 10^5 grid points altogether. In order to achieve a relative standard deviation of $1/10$ at each grid point, roughly 100 observations per cell are required, which is 10^7 observations in all. On the assumption of a computer memory capacity of 5×10^3 molecules, $10^7/5 \times 10^3 = 2 \times 10^3$ independent observations on the flowfield are necessary. On the basis of sampling interval of one collision time, it is found that the program would have to run for 2000 collision times. In the author's experience, the calculation requires about one minute of IBM 7094 II or IBM 360/65 computer time per collision time. Therefore more than 30 hours of computer time would be needed.

The method is at present adequate for finding certain moments of the distribution function. In order to obtain a density field with 10% relative standard deviation at 10^2 grid points using the hypothetical program of the last paragraph would require about 20 minutes of computer

time which is well within reason. Of course, a relative standard deviation of 10% is rather large, leading to numbers in each cell which are within 20% of their mean value with probability 0.95, or within 30% with probability 0.997. Results of such coarseness are usually of little interest. In order to improve them by a factor of $\sqrt{10}$ requires a tenfold increase in computing effort, producing a relative standard deviation of 0.036. Then one is assured numbers within 7.2% of the mean with probability 0.95 or 9.5% with probability 0.997. Such accuracy is acceptable for many practical applications.

4.4 Variants and Refinements

4.4.1 Weighted Cells

In the simulation technique described thus far it has been understood that the cross-section per molecule was the same for each molecule in the flowfield. Another way of looking at it is that each simulator molecule represented a fixed number of physical molecules. There are many situations where such a representation is unsatisfactory because it is desirable to obtain more detail about the flow in some subregion of the flowfield, or in the case of spherical and cylindrical coordinates, to obtain information near the origin or near the axis where there are few molecules because of the small cell volumes. Consequently a weighting scheme has been tried by both G. A. Bird and the author.

In this scheme the collision computation and the time advance formula remain unchanged, except that the cross-section per molecule becomes dependent on the cell. Difficulties arise, however, when a molecule crosses a cell boundary during the convection phase of the iteration. To be specific, let the cross-section per molecule in cell k be $A_k = A/w_k$ $k = 1, \dots, K$ where w_k is a given positive cell weight, and A is a constant dependent on the desired total number of simulator molecules and the average cross-section density. If a molecule moves from one cell to another having twice the weight, it is reasonable to expect that a satisfactory strategy would be to replace it with two copies of itself. The dilemma arises when the molecule moves in the opposite direction. A resolution which has been found satisfactory by the author makes use of a random integer function defined as follows:

$$\text{Intr}(x) = [x] + i,$$

where $[x]$ is the greatest integer which is less than or equal to x , and i is a random variable with the distribution

$$i = \begin{cases} 0 & \text{with probability } 1 - (x - [x]) \\ 1 & \text{with probability } x - [x] \end{cases}$$

Then, when a molecule moves from a cell with weight w_α to a cell with weight w_β , it is replaced with

$$\text{intr}(w_\alpha/w_\beta)$$

molecules having the same coordinates.

A value for the constant A compatible with the molecule capacity can be obtained as follows: Suppose it is given that the cross-section density ($1/\sqrt{2}$ x reciprocal mean free path) at some point and time in the flow is A_0 . A useful first approximation can be based on the assumption that the average cross-section density in the whole flowfield will also be A_0 . Then cell k will contain

$$A_0 \text{vol}_k / A w_k^{-1}$$

simulator molecules, where vol_k is the volume of cell k . Then the total number of molecules in the flowfield will be

$$N_{av} = \sum_{k=1}^K A_0 w_k \text{vol}_k / A \quad (6)$$

where N_{av} depends on the memory capacity of the computer. Solving (6) it is found that

$$A = (A_0 / N_{av}) \sum_{k=1}^K w_k \text{vol}_k$$

4.4.2 Mixtures of Gases

The Monte Carlo simulation method has also been extended by Bird to deal with binary mixtures of gases. Aside from minor complications arising from the labelling of the molecules in computer memory, the biggest modification that must be made is in the time advance procedure in order to ensure that the correct number of each of the four possible types of collision is computed. Let the molecule types be numbered 1 and 2, and denote the collision types by (1,1), (1,2), (2,1), and (2,2). The basic idea is to compute (1,1) collisions until the time parameter has been advanced by $1/4 \Delta t_m$, (1,2) collisions for another $1/4 \Delta t_m$ etc., using the same time increment formula as before with the appropriate cross-section for each collision type. The somewhat more refined procedure adopted by Bird entails keeping a separate time counter for each collision type and using the formula

$$\Delta \tau_{\alpha\beta} = 4 \frac{2 \text{vol}}{A_{\alpha\beta} n_{\alpha} n_{\beta} g}$$

to increment the (α, β) counter. At each step in the collision computation, a collision corresponding to the counter with the lowest value is computed and computation ceases when each counter has surpassed Δt_m . The formulas for computing post collision velocities for molecules of unequal mass are commonly found in textbooks and present no more of a problem than do those for like molecules.

4.5 References

1. Bienkowski, G.K.
McKinley, R.R.
Siskind, S.M. Monte Carlo Solutions for Hypersonic Flow over Axially Symmetric Bodies. Preprint, Princeton U., and R.C.A. Box 800, Princeton, 1970.
2. Denisik, S.A.
Yu, G. Malama,
Polak, L.S.
Rezvanov, R.A. Application of the Monte Carlo Method in the Solution of Some Problems of Physical and Chemical Kinetics, Teplofizika Vysokikh Temperatur, 5, 1011, (1967).
3. Bird, G.A. Approach to Translational Equilibrium in a Rigid Sphere Gas, Phys. Fluids 6, 1518 (1963).
4. Bird, G.A. Shock Wave Structure in a Rigid Sphere Gas, Rarefied Gas Dynamics IV (J.H. deLeeuw, Ed.), vol. 1, 216 (1965).
5. Bird, G.A. Aerodynamic Properties of Some Simple Bodies in the Hypersonic Transition Regime, AIAA J. 4, 55 (1966).
6. Bird, G.A. The Velocity Distribution Function within a Shock Wave, J. Fluid Mech. 30, 479 (1967).
7. Bird, G.A. The Structure of Normal Shock Waves in a Binary Gas Mixture, J. Fluid Mech. 31, 657 (1968).
8. Bird, G.A. The Formation and Reflection of Shock Waves, Rarefied Gas Dynamics VI (L. Trilling and H.Y. Wachman, Ed.) Vol. 1, 85 (1969).
9. Bird, G.A. Direct Simulation Monte Carlo Method - Current Status and Prospects, Rarefied Gas Dynamics VI (L. Trilling and H.Y. Wachman, Ed.), Vol. 1, 301 (1969).
10. Macpherson, A.K. A Collisional Treatment of the Reflection of an Imploding Hemispherical Shock Wave, Preprint, Institute for Aerospace Studies, University of Toronto.
11. Starr, A.W. Monte Carlo Simulation of the Flow of a Rarefied Gas Around a Cylinder, Technical Note, Lincoln Laboratory, MIT, 1967-7.

12. Vogenitz, F.W.
Bird, G.A.
Broadwell, J.E.
Rungaldier, H.
Theoretical and Experimental Study of
Low Density Supersonic Flows About
Several Simple Shapes, AIAA Paper No.
68-6, Aerospace Sciences Meeting, New
York, Jan. 1968.
13. Vogenitz, F.W.
Broadwell, J.E.
Bird, G.A.
Leading Edge Flow by the Monte Carlo
Direct Simulation Technique, AIAA J.
8, 504 (1970).

4.6 List of Symbols

<u>Roman</u>	<u>First Occurrence</u>
A.	4.2.2
A_m, A_o	4.3.6
c, g_{ij}	4.3.5
$D, \mathcal{D}_k$	4.2.1
g	4.2.2
h_m	4.3.5
K, k	4.2.1
L, ℓ, m	4.3.5
MAP	4.2.1
N_o, N_{max}, n, n_k	4.2.1
n_x	4.3.6
$p, p_{bd}, p_r, \hat{p}_\ell, p_{ij}$	4.3.5
q_ℓ	4.3.5
$r_1, \dots, r_L$	4.3.5
s_{max}	4.2.2
S	4.3.5
t	4.1.0
$\Delta t, \Delta t_m$	4.2.2
u	4.3.6
$\bar{v}, \bar{v}', v_I, v_J, v_K$	4.3.6
vol_k	4.4.1
w_k	4.4.1
X_ℓ	4.3.5
<u>Greek</u>	
α	4.2.2
ϕ	4.3.6
λ	4.3.6

μ

4.3.4

τ, τ_c, τ_T

4.3.4

4.7 Definitions

$d\vec{x}$	Volume differential in 3-dimensional space (= $dx_1 dx_2 dx_3$, where x_1, x_2, x_3 are components of the vector $\vec{x}$).
$E(X)$	Expectation of the random variable X .
$f(\vec{x}, \vec{v}, t)$	A function such that $f(\vec{x}, \vec{v}, t) d\vec{x} d\vec{v}$ is the average number of molecules in a cell of 6- dimensional space containing the point and having 6-dimensional volume $d\vec{x} d\vec{v}$.
$(\vec{x} \cdot \vec{y})$	Inner product of the vectors $\vec{x}$ and $\vec{y}$.
ϵ	'Belongs to'
(α, β)	Interval with endpoints α and β .
$\binom{L}{m}$	$= \frac{L(L-1) \dots (L-m+1)}{1.2. \dots m}$
$[x]$	the greatest integer less than or equal to x .
$ \vec{v} $	Length of the vector $\vec{v}$.

5. MONTE CARLO EVALUATION OF THE COLLISION INTEGRAL

5.1 Introduction

The last method to be discussed in this review has been developed almost exclusively by Nordsieck, Hicks, and their coworkers at the University of Illinois, and described reasonably completely in a series of reports of the Coordinated Science Laboratory. Its essential features include Nordsieck's proposal from as early as 1957 (2, p.1) for choosing a random sample of points in the region over which the collision integral is to be taken, a conventional finite difference scheme for obtaining a distribution function satisfying the Boltzman equation once the collision integral is known, and a series of refinements designed to overcome approximation and random errors.

The method has so far only been applied to problems described by the hard sphere model and requiring only one space dimension such as the plane normal shock wave and certain relaxation problems. For such problems the method takes about one hour of CDC 1604 time for the iteration to converge. Each iteration yields a statistical approximation to the iterate one would obtain with an exact evaluation of the collision integral. The approximation has a standard deviation of about 3% at each of $10 \times 226 = 2260$ histogram bins (10 position cells and 226 velocity bins).

Extension of the method to problems requiring higher space dimension seems difficult because of computer storage and time requirements and also because of the problems involved in devising a successful finite difference scheme in more than one dimension.

The discussion presented here will also deal exclusively with problems requiring one space dimension and will treat as an example Hick's and his coworkers' calculations of the plane normal shock wave.

5.2 Theoretical Preliminaries

5.2.1 Nordsieck's Units

Hicks and his coworkers have made use of special units derived basically from the standard procedure for nondimensionalizing the Boltzman equation. The units are based on conditions in some reference region of the flow (in the shock wave problem, the far upstream conditions are convenient).

If b_m , m are the molecular diameter and mass respectively, k is Boltzman's constant, and T, n are the reference temperature and number density respectively, the Nordsieck units for length, velocity and time are respectively $\ell = 1/2\pi n b_m^2$, $c = (2 kT/m)^{1/2}$, and ℓ/c , and the distribution function is given in terms of n/c^3 .

If the substitutions

$$v = \ell x, \quad v_x = c \tilde{v}_x, \quad v_x' = c \tilde{v}_x', \quad v_1 = c \tilde{v}_1, \quad t = (\ell/c) \tilde{t}, \quad f = (n/c^3) \tilde{f}, \\ f' = (n/c^3) \tilde{f}', \quad F = (n/c^3) \tilde{F}, \quad \text{and} \quad F' = (n/c^3) \tilde{F}'$$

are made in Boltzmann's equation, then after dropping the $\sim$ -s, it becomes (See (11), (12.5)).

$$\frac{\partial f}{\partial t} + \mathbf{v}_x \frac{\partial f}{\partial x} = \int_{-\infty}^{\infty} d\mathbf{v}_x' \int_0^{\infty} v_{\perp}' dv_{\perp}' \int_0^{2\pi} d\phi' \int_0^{2\pi} \frac{d\hat{w}}{4\pi} |\hat{w} \cdot \hat{v}_x| (FF' - ff') \quad (1)$$

In their calculations, Nordsieck, Hicks, et al make use of a further velocity scale change in order to allow the use of velocity bins of a size convenient for computation. Since this change is not necessary for explaining the essential features of the method, it will be omitted here.

5.2.2 The Discrete Analogue

In the computational algorithm, both velocity and position space are quantized. Furthermore, in the case of velocity space, some means is needed to represent infinite two dimensional space using only a finite number of cells. This last problem is handled in the Nordsieck-Hicks method by restricting attention to a semi-circular region in $(v_x, v_{\perp})$ space which is still large enough to include 99% of the molecules x under conditions expected in the flow.

With regard to quantization, Nordsieck and Hicks have reported good results with a non-uniform net of about 10 points in the x -direction and an array of 226 square bins in $(v_x, v_{\perp})$ -space covering the semicircular region with 24 bins along the diameter. The net in the x -direction is non-uniform because of varying density gradients. In fact, in their later work on the shock problem, Hicks and Smith found that x was not a good variable to work with because of the problems in making the x -cells fine enough to take care of the large flow gradients in the shock itself. After some experimentation with various monotonic transforms of x , it was found that the best variable to use was the density n itself.

Since it is beyond the scope of the present work to go into the details of a particular solution it will henceforth be assumed for sake of simplicity that there are j_m position cells with partition points x_j $j=1, \dots, j_m$, and s square velocity bins with midpoints $(v_{xs}, v_{\perp s})$, $s=1, \dots, s_m$. The values of the distribution function will be correspondingly indexed as $f(j,s)$. Then, if

$$S(s, s', \bar{n}) \text{ and } S'(s, s', \bar{n})$$

denote the cells containing the post-collision velocities arising out of a collision between molecules having velocities $\bar{v}_s$ and $\bar{v}_{s'}$ with collision vector $\bar{n}$, the discrete approximations to F and F' are

$$f(j, S) \text{ and } f(j, S')$$

respectively.

The discrete approximation to (1) suggested by Nordsieck for the time independent problem where

$$\frac{\partial f}{\partial t} = 0$$

is then

$$v_{xs} \frac{f(j+1,s) - f(j,s)}{x_{j+1} - x_j} = 1/2 [(A-Bf)_{j,s} + (A-Bf)_{j+1,s}] \quad (2)$$

where $(a-bf)_{j,s}$ is an approximation to the collision integral given by

$$(a-bf)_{j,s} = \sum_{s'=1}^{s_m} v_{1s'} \Delta v_{xs'} \Delta v_{1s'} \sum_{q=1}^{q_m} \Delta \varphi'_q \sum_{r=1}^{r_m} \frac{1}{r_m} |\bar{n}_r \cdot (\bar{v}_s - \bar{v}_{s'})| \times \\ \times [f(j,s)f(j,s') - f(j,s)f(j,s')] \quad (3)$$

where $\bar{n}_r, r=1, \dots, r_m$ is an equidistant array of points on the unit sphere and

$$\sum_{q=1}^{q_m} \Delta \varphi'_q = 2\pi.$$

5.2.3 The Monte Carlo Estimate

The evaluation of the discrete approximation to the collision integral in (6) (a sum repeated $j_m s_m$ times) is still beyond the capabilities of the present-day computer. To date, the only resolution of the difficulty is the Monte Carlo method proposed by Nordsieck. Nordsieck's method is based on conventional Monte Carlo evaluation of the sum approximating the collision integral and not of the integral itself. The essence of the procedure is as follows: In order to evaluate the sum at fixed (j,s) , a random sample of N values of the quadruple

$$(v_x', v_{1s}', \varphi', \bar{n})$$

is chosen, where (v_x', v_{1s}') is uniformly distributed over the cell centres of velocity space, φ' is uniformly distributed over an equally spaced set of points in $(0, 2\pi)$, and $\bar{n}$ is uniformly distributed over an equally spaced set of points on the surface of the unit sphere. The Monte Carlo approximation to $(a-bf)_{j,s}$ is then given by

$$\frac{1}{N} \sum_{\text{sample}} \frac{1}{4\pi} v_{1s}' |\bar{n} \cdot (\bar{v}_s - \bar{v}_{s'})| [f(j,s)f(j,s') - f(j,s)f(j,s')] \quad (3)$$

The method as it is presently used differs from the above in a way which takes advantage of certain symmetry, exchange and reflection properties of the collision process in order to expand the effective size of the sample at a considerable saving in work. In order to make this possible it is necessary to evaluate v_{1s} times the collision integral

$$\int_{-\infty}^{\infty} dv_x' \int_0^{\infty} v_{1s} v_{1s}' dv_{1s}' \int_0^{2\pi} \frac{d\bar{n}}{4\pi} |\bar{n} \cdot (\bar{v} - \bar{v}')| (ff' - ff') \quad (4)$$

instead of the collision integral itself.

The details are as follows: two three-dimensional precollision vectors $\bar{v}, \bar{v}'$ are chosen instead of only one. Each choice entails two steps.

A. (v_x, v_1) is chosen from among the cell centres in velocity space.

B. ϕ is chosen from an equally spaced set of points in $(0, 2\pi)$.

The unit vector $\bar{n}$ is then chosen at random from an equally spaced set of points on the surface of the unit sphere, and the post-collision velocities $\bar{V}, \bar{V}'$ are computed. It is convenient to call a sample of 3-vector 5-tuples

$$\{(\bar{v}, \bar{v}', \bar{n}, \bar{V}, \bar{V}')\}$$

admissible (AS) if the individual vectors have the properties just described. Then the expression

$$\frac{226}{N} \sum_{AS} \bar{v}_x = \bar{v}_{xs} \frac{1}{4\pi} v_1 v_1' |\bar{n} \cdot (\bar{v} - \bar{v}')| (FF' - ff') \quad (5)$$

where s, s', S, S' are cell numbers of v, v', V, V' respectively is an approximation to (4).

The advantages of this approach become apparent when it is observed that if

$$\{(\bar{v}, \bar{v}', \bar{n}, \bar{V}, \bar{V}')\}$$

is admissible then so is

$$\{(\bar{v}', \bar{v}, \bar{n}, \bar{V}', \bar{V})\} \quad (6)$$

Furthermore, if the velocity space is symmetrical with respect to reflection about $v_x = 0$ and the subscript R denotes this reflection, then

$$\{(\bar{v}_R, \bar{v}'_R, \bar{n}, \bar{V}_R, \bar{V}'_R)\} \quad (7)$$

is also admissible.

Finally, if $\bar{v}, \bar{v}', \bar{n} \rightarrow \bar{V}, \bar{V}'$ denotes a collision, then $V, V', -\bar{n} \rightarrow v, v'$ also denotes a collision called the inverse collision. In addition, since the Jacobian of the transformation

$$(v, v') \rightarrow (V, V')$$

has absolute value 1 for each $\bar{n}$, it follows that if each of v, v' are uniformly distributed over some region of velocity space, V, V' will be uniformly distributed over the image region. The image region in Nordsieck's case is slightly different from the original region so that collisions yielding $\bar{V}, \bar{V}'$ outside if the original region must be discarded. The discarded fraction is quite small (less than 0.16 [9]) and occurs on the fringe of the finite velocity space where the distribution function f is

also very small. It is expected that the error arising from this discrepancy is insignificant so that

$$\{(\bar{v}, \bar{v}', -\bar{n}, \dot{\bar{v}}, \bar{v}')\} \quad (8)$$

can also be considered admissible.

In view of the preceding discussion, it is clear that starting any admissible sample, seven new admissible samples can be constructed by utilizing (6), (7), and (8), and the composite sample will be eight times as large as the original one. It is equally clear, however, that the composite sample will not consist of statistically independent choices, although it will be a fair sample for purposes of evaluating the collision integral. This lack of independence does not receive comment in the literature on Nordsieck's method, but it seems likely that if anything it is beneficial in reducing the variance of the estimate for the collision integral. The actual strategy for incorporating the composite sample into the collision integral estimate is described as part of the algorithm.

5.2.4 The Corrections

Hicks, Yen, and Reilly have published a report dealing exclusively with various techniques of error reduction and giving the formulas which they have found through experience to be most effective. In most cases, theoretical support for the correction procedures is lacking.

Errors enter into the collision integral estimate from two sources: the replacement of the collision integral by a discrete approximation and from the statistical fluctuations inherent in the Monte Carlo method. It was found very early that if nothing was done to normalize f after each iteration, a systematic drift in the values of f would lead to an eventual violation of the laws of conservation of mass, momentum, and energy. This difficulty is presently handled by the least squares (LS) correction.

In the (LS) correction, corrected values p_s $s=1, \dots, s_m$, of the collision integral are chosen in such a way as to minimize the sum of squares

$$\sum_{s=1}^{s_m} [(A-Bf)_s - p_s]^2 / p_s^2$$

subject to constraints arising from the conservation laws. In the shock problem, mass momentum, and energy fluxes are conserved leading respectively to the constraint equations

$$\begin{aligned} \sum_s p_s v_{\perp s} v_{xs} &= \text{const} \\ \sum_s p_s v_{\perp s} v_{xs}^2 &= \text{const} \\ \sum_s p_s v_{\perp s} v_{xs} (v_{xs}^2 + v_{\perp s}^2) &= \text{const} \end{aligned}$$

The correction

$$\Delta_s = p_s - (A-Bf)_s, s = 1, \dots, s_m \quad (9)$$

is split evenly between A_s and $(Bf)_s$ except in the case of velocity bins where one of the corrected values would be negative, when it is split in such a way as to leave both terms positive.

An attempt is made to control the error arising from statistical fluctuations by the Maxwell-Boltzman (MB) correction. In this correction, the values of the collision integral are calculated in such a way that the collision integral would be zero for a gas in equilibrium which has the same values of density, temperature, and velocity as exist at the given point in the non-equilibrium gas. In carrying out this correction, the required moments of f must first be computed in order to obtain for each position station j the local density n_j , the mean speed u_j and a measure of the deviation from the mean c_j . Let f_{MB} denote the Maxwell-Boltzman distribution function based on these parameters, i.e.

$$f_{MB}(v_x, v_\perp) = 2n_j \pi^{-1/2} c_j^{-3} v_\perp \exp[-(v_x - u_j)^2 + v_\perp^2]/c_j^2$$

The correction corresponding to a collision $(\bar{v}, \bar{v}', \bar{n}, \bar{V}, \bar{V}')$ is given by

$$\Delta(\bar{v}, \bar{v}', \bar{n}) = 1/2 v_\perp v_\perp' \sqrt{n} \cdot (\bar{v}' - \bar{v}) [f_{MB}(\bar{V}) f_{MB}(\bar{V}') - f(\bar{v}) f(\bar{v}')] \quad (10)$$

Then each time during the statistical computation of the collision integral that $(v_\perp A)(\bar{v})$ and $(v_\perp Bf)(\bar{v})$ are incremented as a consequence of such a collision, each is immediately corrected according to the formulas

$$\begin{aligned} (v_\perp A)_c(\bar{v}) &= (v_\perp A)(\bar{v}) - \Delta(\bar{v}, \bar{v}', \hat{w}) \\ (v_\perp Bf)_c(\bar{v}) &= (v_\perp Bf)(\bar{v}) + \Delta(\bar{v}, \bar{v}', \hat{w}) \end{aligned} \quad (11)$$

5.2.5 The Finite Difference Scheme

Nordsieck's proposal for a stable finite difference scheme (one in which old errors decrease from step to step) makes essential use of the fact that the loss term Bf in the collision integral is proportional to f and that Boltzman's equation in the one dimensional time independent case is (see 1.3.I).

$$\frac{\partial f}{\partial x} = \frac{1}{v_x} (A - Bf) \quad (12)$$

The elementary theory of first order linear ordinary differential equations says that a perturbation of the initial value for (12) will decay as x increases if $v_x > 0$.

Consequently, the direction of integration in any finite difference scheme for this equation should be to the left if $f_x < 0$ and to the right if $v_x > 0$. In the case when $x_j - x_{j-1} = \delta x = \text{const}$,

a typical recursion formula for integration to the right based upon the approximation to the derivative given in (6) is given by

$$f_{j+1} = f_j \frac{1-B_j}{1+B_{j+1}} + \frac{A_j + A_{j+1}}{1+B_{j+1}}$$

where

$$A_j = A_j \delta x / 2v_x, B_j = B_j \delta x / 2v_x, j=1, \dots, j_m$$

This simple formula did not prove satisfactory for shock wave studies because of large gradients in flow properties near the centre of the shock. Recent work by Hicks et al on this problem makes use of n as an independent variable instead of x . To write Boltzman's equation in terms of n requires knowing the function $\frac{dn}{dx}(n)$. In practice it is

obtained at each step of the integration procedure by using a discrete version of the formula

$$\frac{dn}{dx} = \int (A-Bf)_{MC} \frac{d\bar{v}}{v_x} \quad (13)$$

where $(A-Bf)_{MC}$ is the Monte Carlo approximation to the collision integral.

5.3 The Algorithm

5.3.1 Storage Requirements and Initialization

The major blocks of storage in this method must be assigned to storing the current iterate of the distribution function f and the two parts A , Bf , of the Monte Carlo estimate of the collision integral. If j_m stations are used along the x -axis and s_m bins are used in velocity space, then j_ms_m locations are required for each of these functions. The values of A and Bf must be stored separately (instead of the difference $A-Bf$) because they are required in the stable difference scheme used in integrating the s_m ordinary differential equations (one for each velocity bin).

In addition to these major blocks of storage, general workspace is required as well as space to store certain function values which are used repeatedly and which would take an excessive amount of time to recalculate each time they are used.

The main part of initialization entails giving initial values to the distribution function f , that is specifying the zeroth iterate. Not unexpectedly, best results are obtained when the initial approximation is fairly close to the correct steady state distribution. In the case of Hicks and coworkers shock wave studies, the initial distribution was taken to be Mott-Smith.

5.3.2 Steps in One Iteration

A. Monte Carlo estimate for the collision integral. As has already been mentioned in 5.2.3, it is convenient to first estimate $v_1 A$ and $v_1 Bf$

and then to divide by v_1 and $v_1 f$ respectively, in order to obtain A and B. Also, the Maxwell-Boltzman correction, if it is used, must be incorporated here. Finally, it is worth pointing out that the estimation procedure about to be described must be repeated j_m times, once for each value of the position variable.

Before commencing accumulation of the integrals, it is necessary to zero the storage for $v_1 A$ and $v_1 B f$, and to compute the factor

$$w = 2\pi(j_m/N_s) \times (\text{area of velocity space}),$$

where N_s is the number of collisions in the collision sample, which according to the experience of Hicks must be about 2^{13} or larger. Thus the sampling procedure about to be described is repeated N_s times.

In this procedure a velocity pair (v_x, v_1) is chosen at random from the discrete two dimensional velocity space, and an angle φ is chosen from an equidistant set of values in $(0, 2\pi)$. It must be remarked that since the actual value is not needed in the computation, it is time-saving to store only the sets of values $\sin\varphi$ and $\cos\varphi$ and to choose the appropriate one of these at random when the need arises. In any case, the triple (v_x, v_1, φ) defines a velocity vector henceforth denoted by $\bar{v}$. The procedure is thus repeated to choose another velocity vector $\bar{v}'$ independently of the first. Finally, the collision vector $\bar{w}$ is chosen at random from an equally spaced set of points on the surface of the unit sphere.

It is now possible to compute the function

$$X = v_1 v_1' |\hat{w} \cdot \bar{v}_r|, (\bar{v}_r = \bar{v}' - \bar{v})$$

and the post-collision velocities $\bar{V}, \bar{V}'$ (1.1.1, Eq. 1,2). Let s, s', S, S' denote the cell numbers of the velocities v, v', V, V' respectively. Finally, if $\bar{v}$ is any vector, let $\bar{v}_R$ denote the reflection of $\bar{v}$ in the plane $x=0$, and let s_R, s'_R, S_R, S'_R denote the reflected subscripts corresponding to s, s', S, S' respectively. One further piece of notation is convenient in writing down the subsequent formulas. That is:

$$f = f(j, s), \quad f' = f(j, s')$$

$$F = f(j, S), \quad F' = f(j, S'),$$

with f_R, f'_R, F_R, F'_R defined correspondingly.

Eight contributions to the collision integral are then computed from the triple $(\bar{v}, \bar{v}', \bar{n})$ by adding

$$wXFF \text{ to } v_1 A(s), v_1 A(s'), v_1 Bf(S), v_1 Bf(S'),$$

$$wXff \text{ to } v_1 A(S), v_1 A(S'), v_1 Bf(s), v_1 Bf(s'),$$

$$wX^F_R F'_R \text{ to } v_1 A(s_R), v_1 A(s'_R), v_1 Bf(S_R), v_1 Bf(S'_R),$$

$$wX^f_R f'_R \text{ to } v_1 A(S_R), v_1 A(S'_R), v_1 Bf(s_R), v_1 Bf(s'_R).$$

At the same time the MB correction can be applied according to the formulas

$$\begin{aligned} (v_1 A)_c(s) &= v_1 A(s) - \Delta(s, s', \bar{w}); \quad (v_1 bf)_c(s) = v_1 bf(s) + \Delta(s, s', \bar{w}), \\ (v_1 A)_c(s') &= v_1 A(s') - \Delta(s', s, \bar{n}); \text{ etc.} \end{aligned}$$

As soon as N_s sets of eight contributions apiece have been computed, the whole Monte Carlo collision estimate is repeated at the next position station. Hicks and his coworkers found that the properties of the general iteration procedure were improved if the statistical fluctuations between position stations were eliminated by making use of the same collision sample at each position station. This can be simply achieved by reinitializing the random number generator to the same value at the start of each integral computation. When all the collision integrals have been computed, the program proceeds to B.

B. The least squares correction. The LS correction is the smallest modification of the s_m values of the collision integral consistent with conservation of mass, momentum, and energy. The correction is made in accordance with the formulas given in 2.14.

C. The numerical integration phase. Once the corrections have been made, values of a and b suitable for use in the numerical integration scheme can be calculated. The numerical integration itself involves a solution of s_m uncoupled ordinary differential equations, one for each fibre $s_m = \text{const}$ in the discrete phase space. The relevant recursion relations are discussed in 2.14. The j_{msm} values of f obtained as a result of the integrations constitute the next iterate.

D. Calculation of moments. Some or all parts of this subprogram are generally bypassed until the final iteration. Normally, only enough information needs to be output from the intermediate iterations to give evidence that the iterations are in fact converging. It would be too lengthy to give a complete list of what type of information about the distribution function has or should be calculated. Nordsieck, Hicks, et al have computed moments up to the fourth in the shock problem, graphical displays of the distribution function, and estimates of standard deviation.

E. Exit criterion. The common criterial for convergence of an iteration scheme are applicable here. The simplest, of course, is a test that subsequent iterations are close enough together. It has been found that, starting with the Mott-Smith distribution in the shock problem, around 12 iterations give satisfactory convergence.

5.4 Comments

The only aspect of this method which is rigorously understood from the standpoint of theory is the Monte Carlo estimate of the collision integral. Here, standard results on evaluating integrals by means of a random sample of points in the domain of integration apply. The fact that the eight variants arising from each collision are dependent has not been treated theoretically but presumably comes under the heading of the theory of antithetic variates. In addition, in Nordsieck's method it is not the collision integral itself that is estimated, but only a discrete approximation. The closeness of this approximation has only been

investigated in an empirical sort of way. In the numerical integration phase, the recently opened study of differential and difference equations with coefficients chosen from some distribution could conceivably throw light on possible inaccuracies in Woidsieck's solution of the 226 difference equations whose coefficients have been obtained from the collision integral estimate.

Hicks and his coworkers did initially run into difficulties on the numerical integration phase which they first tried to surmount by smoothing the values of a and b after they were computed, and most recently by using a fixed collision sample for each position station. The latter has succeeded in stabilizing the calculation, but does not help in obtaining a measure of how far the calculated solution is to the actual one. A great deal of work remains to be done.

At the moment, faith in the method must rest on the exhaustive empirical error studies that have been done for each of the problems undertaken by the Chicago team, and described in detail in their CSL reports. The care with which this work has been done evokes confidence in the actual figures that have been published. Nevertheless, the sparseness of theory dealing with the method, the intricacy of the corrections which seem essential to its success, and the time required to both develop the program for a particular problem and for the computation itself, limit its usefulness as a general purpose tool for solving kinetic theory problems. Finally, the method has been developed for zero or one space dimension, whereas many of the more interesting problems require at least two.

5.5 References

The letters CSL in these references refer to the Coordinated Science Laboratory, University of Illinois, Urbana, Illinois.

1. Hicks, B.L. The Pseudo-Shock: A Nonlinear Problem of Translational Relaxation, CSL Report I-236.
2. Hicks, B.L. Numerical Studies of Strong Shock Waves,
Aggarwal, J.K. Part I: Illiac Solution of a Boltzman Equation by Nordsieck's Method, CSL Report I-111, 1962.
3. Hicks, B.L. Numerical Studies of Strong Shock Waves,
Aggarwal, J.K. Part II: Results of Illiac Calculations, CSL Report I-117, 1963.
4. Hicks, B.L. Numerical Studies of Strong Shock Waves,
Smith, M.A. Part VIII: Properties of a Shock Wave for a Mach Number of 2.5, CSL Report R-347, 1967.
5. Hicks, B.L. Numerical Studies of Strong Shock Waves,
Smith, M.A. Part IX: Error Analysis for a Mach Number of 2.5, CSL Report R-348, 1967.
6. Hicks, B.L. On the Accuracy of Monte Carlo Solutions of the Nonlinear Boltzman Equation, Journal of Computational Physics, Vol. 3, No. 1.
7. Nordsieck, A. Bull. Amer. Phys. Soc., 10, 1183 (1965).
Hicks, B.L.
8. Nordsieck, A. Bull. Amer. Phys. Soc., 10, 1127 (1965).
Hicks, B.L.
9. Nordsieck, A. Monte Carlo Evaluation of the Boltzman Collision Integral, CSL Report R-307, 1966.
Hicks, B.L.
10. Smith, M.A. Numerical Studies of Strong Shock Waves,
Hicks, B.L. Part VII: A Study of Monte Carlo Errors in the Evaluation of the Collision Integral, CSL I-131, 1965.
11. Yen, S.M. Approximate Solution of the Boltzman Equation for a Shock Wave, AVCO RAD Tech. Memo, TM-66-9 (1966).
12. Yen, S.M. On the Accuracy of Approximate Solutions to the Boltzman Equation, CSL Report R-308, 1966.
Hicks, B.L.

13. Yen, S.M.
Hicks, P.L. Initial Behaviour of the Pseudo-Shock,
CSL Report R-350, 1967.
14. Yen, S.M.
Schmidt, H.J. Rarefied Gas Dynamics 6, (L. Trilling
and H. Y. Wachman Ed.), Vol. 1, 85 (1969)
Monte Carlo Solutions for Heat Transfer
Problems.

5.6 List of Symbols

<u>Roman</u>	<u>First Occurrence</u>
a	5.2.2
A, A_s	5.2.4
$A_j, (A-Bf)_{MC}$	5.2.5
b_m	5.2.1
b	5.3.2
$E (Bf)_s$	5.2.4
B_j	5.2.5
c	5.2.1
c_j	5.2.4
$\hat{c}, \hat{c}', \tilde{c}, \tilde{c}', F, F', \tilde{F}, \tilde{F}'$	5.2.1
$f(j, s), \tilde{f}(j, s')$	5.2.2
f_{MB}	5.2.4
f_j	5.2.5
$f_R, f'_R, \tilde{f}_R, \tilde{f}'_R$	5.3.2
j_l, j_m	5.2.2
l	5.2.1
m	5.2.1
$n, \bar{n}$	5.2.1
n_r	5.2.2
N_s	5.3.2
R	5.2.3
$s, s_m, S(s, s', \bar{n}), S(s, s', \bar{n})$	5.2.2
T	5.2.1
u_j	5.2.4

<u>Roman</u>	<u>First Occurrence</u>
$v_{\dot{x}}, \tilde{v}_{\dot{x}}, v'_{\dot{x}}, \tilde{v}'_{\dot{x}}, v, v'$	5.2.1
$v_{xs}, v_s, \tilde{v}_s, v'_s$	5.2.2
$(v_{\perp a})(\bar{v}), (v_{\perp bf})(\bar{v}), (v_{\perp a})_c(\bar{v}), (v_{\perp bf})_c(\bar{v})$	5.2.4
v_R	5.3.2
$\hat{w}$	5.3.2
$x, \tilde{x}$	5.2.1
x_j	5.2.2
<u>Greek</u>	
δx	5.2.5
$\Delta v_{xs}, \Delta v'_s, \Delta \phi_q$	5.2.2
$\Delta(\bar{v}, \bar{v}', \bar{n})$	5.2.4
ϕ, ϕ'	5.2.1
χ	5.3.2