MONTE CARLO SIMULATION OF NONLINEAR RADIATION INDUCED PLASMAS

by

Benjamin Shaw-hu Wang

May, 1972

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS


N73-10700 Unclas G3/25 16254
MONTE CARLO SIMULATION OF NONLINEAR RADIATION
INDUCED PLASMAS

BY
BENJAMIN SHAW-hu WANG
B.S., National Taiwan University, 1960
M.S., University of Missouri (Rolla), 1964

THESIS
Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science in the Graduate College of the University of Illinois at Urbana-Champaign, 1972

Urbana, Illinois
A Monte Carlo simulation model for radiation induced plasmas with nonlinear properties due to recombination has been developed, employing a piecewise linearized predict-correct iterative technique. Several important variance reduction techniques have been developed and incorporated into the model, including an antithetic variates technique (negatively correlated variates) which has proven to be most useful. This approach is especially efficient for plasma systems with inhomogeneous media, multidimensions, and irregular boundaries, where other analytic or numerical solutions are either unrealistic or impractical. This model is quite general in scope and should be applicable to a variety of similar particle transport and diffusion-type plasma problems.

The Monte Carlo code developed in this work has been applied to the determination of the electron energy distribution function and related parameters for a noble gas plasma created by α-particle irradiation. The radiation induced plasma involved was characterized by the following features:

1) a continuous internal volume-source of high energy electrons
2) an applied electric field (optional)
3) secondary electron production via ionization processes
4) losses via recombination and leakage across boundaries

The calculations were specifically carried out for a helium gas medium with an electron source rate from $10^{14}$ to $10^{22}$ (#/cm$^3$-sec), initial electron energies from 70 to 1000 eV, pressures from 10 to 760 torr and E/P (electric field/pressure) ratios from 1 to 10 V/cm-torr.

For the lower source rates, it is observed that, in the zero electric-field case, the low-energy portion of the distribution function is quite close to a standard Maxwellian distribution. However, the high-energy portion (above the ionization potential) of the distributions is a rapidly decaying parabolic-shaped tail which can be crudely represented by a $1/\text{Energy}$ distribution. The addition of the electric field causes the distribution function in the low-energy region to shift towards Smit's (or Druyvesteyn's) distribution while the high-energy portion essentially retains its zero field shape. As the source rate increases beyond certain value (roughly $10^{18}$ cm$^{-3}$-sec$^{-1}$ in the present case), the low-energy region begins to depart from the standard Maxwellian or Smit distributions and this departure can become quite large.
A major emphasis in this development was placed on computation efficiency. The present computation scheme with its variance and accelerated convergence techniques is thought to be quite efficient compared to alternate approaches.
ACKNOWLEDGEMENT

The author wishes to express his sincere appreciation and gratitude to Professor George H. Miley for his patient guidance and invaluable help throughout the course of this work, without which this study would have been impossible. The author wishes to also thank Mr. Ronne Lo for his most valuable and helpful discussions.

The author would also like to thank Professor J. N. Snyder, the head of Computer Science Department, and the Nuclear Engineering Program for their financial support. This work was supported in part by the Physics Branch of the AEC Research Division under Grant No. AT(11-1)-2007 and by the NASA-AEC Space Power Office under Grant No. NGR 14-005-183.

Finally, the author wishes to thank Mrs. Sharon Taylor for her excellent typing of the final thesis manuscript.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>A. Statement of the Problem and Objectives</td>
<td>1</td>
</tr>
<tr>
<td>B. The Importance of Nuclear Radiation Produced Plasmas and the Physical Model</td>
<td>1</td>
</tr>
<tr>
<td>C. General Approaches and Methods of Solution</td>
<td>5</td>
</tr>
<tr>
<td>II. THE MATHEMATICAL MODEL: NUMERICAL SCHEME AND PHYSICAL PROBLEM</td>
<td>8</td>
</tr>
<tr>
<td>A. General Description of the Physics of the Problem</td>
<td>8</td>
</tr>
<tr>
<td>1. Physical Processes Involved</td>
<td>8</td>
</tr>
<tr>
<td>2. Relation to the Transport Equation</td>
<td>10</td>
</tr>
<tr>
<td>B. The Stochastic Model - Numerical Scheme</td>
<td>12</td>
</tr>
<tr>
<td>1. The Geometry and the Coordinate System</td>
<td>13</td>
</tr>
<tr>
<td>2. Tracing Procedure</td>
<td>20</td>
</tr>
<tr>
<td>a. The System Without Electric Field ($E = 0$)</td>
<td>21</td>
</tr>
<tr>
<td>b. With Electric Field Present</td>
<td>25</td>
</tr>
<tr>
<td>c. Types of Collisions and Cross Sections</td>
<td>27</td>
</tr>
<tr>
<td>3. The Prediction and Correction of Dominant System Parameters</td>
<td>29</td>
</tr>
<tr>
<td>III. VARIANCE REDUCTION TECHNIQUES</td>
<td>33</td>
</tr>
<tr>
<td>A. General Background</td>
<td>33</td>
</tr>
<tr>
<td>B. Importance Sampling</td>
<td>38</td>
</tr>
<tr>
<td>C. Russian Roulette and Splitting</td>
<td>42</td>
</tr>
<tr>
<td>D. Initial Guess</td>
<td>44</td>
</tr>
<tr>
<td>E. Antithetic Variates</td>
<td>46</td>
</tr>
<tr>
<td>F. Double History (Double Processing) Technique</td>
<td>50</td>
</tr>
<tr>
<td>CHAPTER</td>
<td>Page</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>IV. MONTE CARLO SIMULATION CODE</td>
<td>52</td>
</tr>
<tr>
<td>A. Introduction</td>
<td>52</td>
</tr>
<tr>
<td>B. Calculational Procedure and Program</td>
<td>52</td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>V. RESULTS</td>
<td>63</td>
</tr>
<tr>
<td>A. Convergence and Reliability</td>
<td>63</td>
</tr>
<tr>
<td>B. Error Estimation and Analysis</td>
<td>64</td>
</tr>
<tr>
<td>1. Error Analysis</td>
<td>64</td>
</tr>
<tr>
<td>2. Data Filtering</td>
<td>69</td>
</tr>
<tr>
<td>C. Solutions of the Problem</td>
<td>71</td>
</tr>
<tr>
<td>1. Preliminary Checks</td>
<td>71</td>
</tr>
<tr>
<td>2. The Electron Energy Distribution Function without Electric Field</td>
<td>77</td>
</tr>
<tr>
<td>3. The Electron Energy Distribution Function with an Applied Electric Field</td>
<td>90</td>
</tr>
<tr>
<td>D. Conclusion on Monte Carlo Results</td>
<td>104</td>
</tr>
<tr>
<td>VI. CONCLUSIONS</td>
<td>108</td>
</tr>
<tr>
<td>A. Prelude</td>
<td>108</td>
</tr>
<tr>
<td>B. Summary of Present Work</td>
<td>109</td>
</tr>
<tr>
<td>C. Practical Limitations and the Cures</td>
<td>112</td>
</tr>
<tr>
<td>D. Future Extensions</td>
<td>116</td>
</tr>
<tr>
<td>E. Concluding Remarks</td>
<td>119</td>
</tr>
<tr>
<td>LIST OF REFERENCES</td>
<td>121</td>
</tr>
<tr>
<td>APPENDIX</td>
<td>125</td>
</tr>
<tr>
<td>VITA</td>
<td>126</td>
</tr>
</tbody>
</table>
CHAPTER I
INTRODUCTION

A. Statement of the Problem and Objectives

Two major goals were set for this study. The first was to develop an efficient Monte Carlo simulation technique for solving radiation-induced plasma problems. This required adapting several variance reduction techniques in order to increase the Monte Carlo efficiency. The second goal was to apply the model developed herein to study the characteristics of a noble-gas plasma created by alpha-particle irradiation.

The major point of interest is the determination of the electron energy distribution in the nuclear radiation induced plasma. This distribution is of direct importance to current experimental studies at the University of Illinois where the possibility of directly pumping* gas lasers with nuclear radiation is under study (6). Recent results from these studies also show that nuclear radiation can be used to enhance the output from some type of electrically pumped lasers (5).

B. The Importance of Nuclear Radiation Produced Plasmas and the Physical Model

Nuclear radiation can be used in two ways: to directly pump the laser, or alternately, where an electric discharge

---

"Pumping" of a laser is defined as the introduction of energy into the lasing medium required for population inversion. The electrical current which creates the discharge in the conventional gas laser is the pump in that case. Flash lamps are used for many solid state lasers.
serves as the main pump, to enhance the operation of lasers. The possibility of efficient coupling of lasers and nuclear reactors has generated an interest in the use of high-energy ion beams to pump lasers. To avoid radiation damage problems, gas lasers are generally thought of in this connection.

There are several important reasons for searching for more direct methods of coupling energy from reactor sources into lasers (8). One is the possibility that the resulting laser may offer unique characteristics, e.g., new frequencies and new methods of modulation and control. Simplified coupling could also offer weight reductions and improved reliability, which may be of advantage in space applications. Also, it may eventually become feasible to simultaneously extract several energy forms, e.g., laser and electrical power from a single nuclear station.

In most gas lasers the excited states, and consequently the inversion of population necessary for laser operation, are formed directly or indirectly by electronic collisions in a plasma generated by an electrical discharge. However, an alternate way to accomplish this goal is via nuclear radiation which can produce ionization and excitation through both direct interactions and interactions due to secondary-electron production. Given the high neutron fluxes available from nuclear reactors, the particles from neutron-induced nuclear reactions (e.g., α-particles from a Boron coating on the wall of a laser tube) can furnish the nuclear radiation
source. The description of the transport of both the primary radiation and the secondary electrons, from their point of birth throughout their path in the gas, is a key problem in the quantitative evaluation of the plasma. Therefore the development of nuclear-radiation pumping requires a thorough understanding of the radiation-induced plasma.

The following physical model is applied to the situation of a radiation-induced plasma. Pertinent quantities of interest include: electron-energy distribution spectrum; average electron energy; and rate of energy loss.

The heavy-charged particles (α-particles) are born in coating and enter an adjacent gas. In the present calculations, the cylindrical geometry of the actual laser tube is approximated by a slab geometry (cf. Figure 1) with two parallel plane boron coatings separated by a gaseous medium. The α-particle radiation continuously creates high energy electrons throughout the gas volume (with energy $E_o$). The resulting plasma is a weakly ionized, three-component plasma (electrons, positive ions, and neutral gas atoms). It is assumed that the alpha-particle trajectory can be treated analytically using methods such as developed in (7). Further, this reference shows how to obtain the primary secondary-electron (δ-ray) energy spectrum. The difficult problem, however, is to follow the thermalization and further secondary production of these primary electrons.
Thus, the Monte Carlo simulation starts off by following these electrons on their lifetime histories, i.e., by simulating the course of their thermalization. Electrons go through various processes of collisions and scattering, resulting in a complex process of diffusion, energy degradation and secondary production, but finally, they recombine with positive ions or escape from the system, ending their histories.

With the radiation rates of interest here, the plasma is only weakly ionized and electrons mostly thermalize by collisions with the neutral background-gas atoms. In other words, collisions with ions and other free electrons can be neglected. In this sense, the problem is simplified in that it remains linear. However, some non-linearity enters in the following way. The recombination process depends on the ion density and spatial distribution. However, the ions are originally formed by ionizations which occur as the electrons slow down.

Another important characteristic is that the plasma is highly non-equilibrium. The thermalized electrons and ions form an approximately Maxwellian background, but superimposed on this is the high energy tail due to the electrons in the process of slowing. Also, the alpha particles themselves represent a non-equilibrium component.

In summary, the special features of the plasma system are:
1) Continuous internal volume source.

2) An applied electric field may or may not be present.

3) Finite cylindrical geometry--approximated here by slab geometry.

4) Electron-ion pair production and recombination processes.

5) The energy range of the electrons range from kilovolts to thermal (eV.).

6) The system represents a non-equilibrium plasma, but in a steady-state condition (or stationary).

7) Due to relatively small densities of electrons and ions compared with that of gas atoms, direct interactions between the electrons themselves or electron-ion collisions are neglected.

C. General Approaches and Methods of Solution

Possible approaches toward solutions of these types of problems (particle transport and diffusion) can be categorized as follows:

1) Analytical solution of the governing Boltzmann transport equation.

2) Numerical solution of the Boltzmann transport equation.

3) Direct analog simulation (Monte Carlo).

4) Mixtures of the above approaches.

The governing equation for this type of problem is the Boltzmann equation, and most classical methods have involved approximate analytic solutions of it. Solutions are possible only after various approximations and simplifications are made, and such solutions are often very restrictive and/or difficult to extend to actual situations of interest. More
or less the same drawbacks apply to numerical solutions. However, in this case approximations are necessary or the computation time becomes too long, especially as the dimensionality and the complexity of the problem increases. In fact, the computation time increases roughly exponentially with the dimensionality of the problem. Usually the governing partial differential and integral equations (coupled Boltzmann equations) are reduced to finite difference equations, and the requirements for stability and convergence severely limit the maximum step sizes of the independent variables. Further difficulties with the practical application of such approaches arise because it is often difficult to include realistic boundary conditions.

Direct simulation by the application of Monte Carlo techniques seems to be the better approach; sometimes the only approach for such problems. It by-passes the direct solution of the partial differential-integral equation, and generally no approximations and/or simplifications are necessary. Complicated boundary conditions can be handled easily, and the computation time increases at the most linearly with the dimensionality of the problem. However, computer time still may be a problem so that judicious selection of numerical schemes and variance reduction techniques are required.

Many methods have been devised in the past which involve a mixture of analytic, numerical, and Monte Carlo approaches.
In such situations, Monte Carlo is usually applied to the most difficult part of the problem. For instance, in the evaluation of the non-linear, five-dimensional collision integral in the Boltzmann equation, straightforward numerical quadrature would require months of computation time even on the present day fast computers. The Monte Carlo (statistical sampling techniques which closely resemble the actual collision phenomena) techniques can produce results within reasonable computation time (24). The Monte Carlo method may be defined as a numerical device (numerical experimentation) for studying an artificial stochastic model of a physical or mathematical process. This study may be approached by one of two methods. The first is the process of simulation (direct analog simulation) in which the particles - electrons and ions in the case of plasma systems, or photons in the case of radiative transfer - are represented by a game of chance in the computer. In the second method, which is non-simulation, the Monte Carlo technique is strictly used as a numerical stochastic device for the solution of a given integral or partial differential equations (24,25).
CHAPTER II

THE MATHEMATICAL MODEL: NUMERICAL SCHEME AND PHYSICAL PROBLEM

A. General Description of the Physics of the Problem

The plasma under study is a weakly-ionized three-component plasma, and is non-equilibrium as described in Chapter I. The α-particle radiation continuously creates high-energy electrons throughout the gas volume. The Monte Carlo simulation is to trace these electrons' life histories and their complex processes of diffusion and energy degradation. Since the diffusion process of the electrons through the medium is affected by the density of the positive ions and the other electrons present (background electrons), the resulting problem is non-linear.

1. Physical Processes Involved

The mechanisms by which particles interact with the atoms of the medium are elastic and inelastic collisions, by which the incident particle is slowed and dissipates some or all of its kinetic energy to the medium by the combined action of the above mentioned processes. The particle can also experience so called superelastic collision by which the particle collides with the excited atoms gaining some energy. Finally, the particle can collide and recombine with positive ions or can leak out of the system thus ending its life.
As described in Chapter I, the type of interaction which is involved in any particular collision is governed by the laws of chance. For large numbers of interactions, the frequency of occurrence of any given interaction is determined by the relative magnitudes of the interaction cross sections. Secondary knock-on electrons due to the ionization process may be capable of further interacting with the atoms and producing additional secondary electrons.

As the system reaches steady-state, a balance is reached such that the birth rate for new particles, i.e., the rate at which primary or source electrons enter the system plus the production rate of secondary electrons due to ionization should approximately equal the loss rate due to leakage and absorption or recombination.

Since the transport of radiation through a medium affects its absorption properties, the transport of individual particles is indirectly affected by the density of the other particles present, and this introduces some nonlinearities into the system. We shall adopt a piecewise predict-correct technique to approximate the nonlinear system with a linear model.

We are concerned with the simulation, by random sampling of the scattering of the charged particles. The direct simulation of the physical scattering processes in such a way that the diffusion process is imitated by letting
the particles carry out a random walk, each step of which takes into account the combined effect of many collisions. The mechanisms by which electrons interact with the atoms of the medium are elastic, inelastic, and superelastic collisions as well as recombination with positive ions. The incident source electron is slowed and dissipates some or all of its energy to the medium by the combined action of the various scattering processes, until it escapes, or recombines with an ion (radiative energy losses are neglected). The probability that an interaction occurs is determined by the total collision cross-section. The type of interaction is then determined by the relative magnitudes of the individual cross-sections.

2. Relation to the Transport Equation

Although no direct use will be made of the governing Boltzmann transport equation in the sequel, we shall write it down briefly, in order to indicate the mathematical problems to be solved implicitly by the Monte Carlo method. It is a nonlinear integro-differential equation of the form

\[
\frac{\partial f}{\partial t} + \vec{u} \cdot \nabla f - \frac{\vec{m}}{m} (\vec{E} + \vec{u} \times \vec{B}) \cdot \nabla_u f = \left( \frac{\partial f}{\partial t} \right)_{\text{collisions}} \tag{1}
\]

where \( f \): number density distribution function

\( \vec{B} \): magnetic field

\( \vec{u} \): velocity of the particles

\( \vec{E} \): electric field strength
In our situation $\mathbf{B}=0$ since the magnetic field does not exist. Further, if we assume an infinite medium and uniform source, $\nabla r \cdot \mathbf{f}=0$, the distribution function $f$ depends only on energy. Then the above equation reduces to

$$\frac{\partial f}{\partial t} - \frac{\mathbf{E}}{m} \cdot \nabla \mathbf{u} f = (\frac{\partial f}{\partial t}) \text{ collisions} \tag{2}$$

The right hand side term is the collision term, which is the most difficult part of the equation. The effects of all relevant particle interactions must be included in this term. For the case of electron collisions with stationary gas atoms of uniform density, we can write the collision term as:

$$(\frac{\partial f}{\partial t}) \text{ collision} = \frac{d N_{\text{in}} - d N_{\text{out}}}{dV \cdot dt} \tag{3}$$

where $dN_{\text{in}}$ gives the number of phase space points (electrons) that are scattered into the phase-space volume $dV$ by collision in time $dt$. Similarly, $dN_{\text{out}}$ represents the number of points scattered out of $dV$ by collisions in time $dt$. This Eq. 2 states that the change in the distribution function $f$ are caused by collisions of various kinds and the external force due to the electric field.

The assumptions that enter into the derivation of the transport equation which carry over to the Monte Carlo
simulation include:

a) the scattering centers (background atoms and ions) are distributed at random, i.e., possible correlations between the positions of various atoms and ions are not taken into account.

b) In the absence of an electric field, the trajectory of the particle is idealized as a zig-zag path, consisting of flights interrupted by sudden collisions in which the energy and direction of the particle changed. The quantum-mechanical interference (electron diffraction) resulting from the coherent scattering by several centers are neglected, and a particle, in the course of traversing the medium, interacts with one scattering center at a time.

The diffusion process in terms of the transport equation is analogous to the use of Eulerian coordinates in hydrodynamics, in that one asks about the flux at given points in space. By contrast, the Monte Carlo method uses Lagrangian coordinates; one attaches a label to a particular bit of fluid, i.e., a diffusing particle, and follows its history. Then, by sampling many histories, one is able, at least in principle, to solve any diffusion problem.

B. The Stochastic Model - Numerical Scheme

The essential approach to the nonlinear problem adopted here is the use of piecewise linearized steps. In each such step (an amount of time $\Delta t$ along the time axis), the
dominate parameters, which characterize the nonlinear properties, are predicted and considered to be fixed with respect to the independent variables. During the time $\Delta t$, the system can be considered as more or less linear, so the conventional Monte Carlo simulation technique is used. The quantities pertaining to the given system are computed for each of a reasonably large number of independent particle histories. The desired parameter values are obtained by averaging the results over all of the histories. At the end of the step, the dominate parameters are corrected before beginning the next step. In each step (time interval $\Delta t$, or cycle), instead of computing the trajectory and angular deflection of an electron after each interaction, we perform a random walk calculation. Since each step of the random walk takes into account the combined effects of many individual interactions, we form a so-called condensed history by this approach. Let us describe the model in some detail, step by step.

1. The Geometry and the Coordinate System

The initial spatial geometry in this problem is a one-dimensional, finite slab (from $z=0$ to $z=D$) which is subdivided into uniform intervals called zones as shown in Figure 1. The energy range is from 0 to $E_0$ (=1keV) which is also subdivided into three regions: 0~1 (thermal energy region), 1 to ED2, and ED2 to $E_0$. Each of these
Figure 1. Slab geometry of the medium with subdivided zones.
regions is further divided into smaller uniform or logarithmic spacing (intervals). The particle number density distribution \( f(E,z) \) is a function of energy \( E \) and spatial position \( z \) and is generated such that particles with energy and spatial position falling into the same intervals are grouped together and normalized by the total particles processed. Although the system is stationary, we deliberately introduce the time variable for the convenience of recording output information and carrying out the calculation steps. The time serves as a clock, or we can imagine that the system begins in a transient state. Then, as time goes on it will eventually reach steady state. The storage of the timing information for individual particles is not necessary. Instead of following a single particle from the beginning to the end of its history, we follow a large number of particles, say \( N \) (which represent the sample particles of the whole system population) through a small time interval \( \Delta t \) (called the cycle time). Within this time interval, the system is considered to be linear, and the particles are processed more or less independently as in a conventional linear Monte Carlo model. At the end of \( \Delta t \), called census time, we record data information for each particle (record particle state information). During this time interval (or present time cycle) some particles may experience collisions of some kind, some may not, and new particles (source particles, secondary particles produced
by ionization) enter the system. Some particles may be absorbed or escape from the system; their histories are therefore ended. This represents a random walk calculation, and each step of the random walk takes into account the combined effect of many individual interactions. The trajectory of a particle which is computed by this approach is called a condensed history. The residue particles (ionization produced secondary particles) are also set into motion, and their histories are followed separately. They are like all the other source particles in all respects.

The complete description of particle histories is approximated by the condensed histories, or we take sequences of snapshots of all the particles in the system at various times to provide a moving picture of the history that can be used to estimate the quantities of interest. The path lengths travelled by the particles are also used as a clock to measure time, or equivalently, the variable \( t \) may be replaced by \( S(t) = \int_{t_0}^{t} v(x)dx \). The trajectories of the particles (the condensed histories) are described by the arrays as indicated in Figure 2. The superscript denotes the cycle time number; the subscript denotes particle number where the index 0 refers to the initial state of the particle. Other notation includes \( E \): the particle energy, \( z \): spatial position, and \( U \): the direction, and \( w_t \): the weight. When the particle has traveled a path-length \( DCFN \) (in the same duration \( \Delta t \)), a condensed history
<table>
<thead>
<tr>
<th>i</th>
<th>t₀</th>
<th>t₁</th>
<th>tₐ</th>
<th>tₘ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>E₁⁰</td>
<td>E₁¹</td>
<td>...</td>
<td>Eₖ</td>
</tr>
<tr>
<td></td>
<td>E₁</td>
<td>...</td>
<td>Eₖ</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>Z₁⁰</td>
<td>Z₁¹</td>
<td>Z₁</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>U₁⁰</td>
<td>U₁¹</td>
<td>U₁</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>wt₁⁰</td>
<td>wt₁¹</td>
<td>wt₁</td>
<td>...</td>
</tr>
</tbody>
</table>

| 2 | E₂⁰ | E₂¹ | ... | E₂ | ... | Eₘ₂ |
|   | E₂ | ... | E₂ | ... | Eₘ₂ |
|   | Z₂⁰ | Z₂¹ | Z₂ | ... | Zₘ₂ |
|   | U₂⁰ | U₂¹ | U₂ | ... | Uₘ₂ |
|   | wt₂⁰ | wt₂¹ | wt₂ | ... | wtₘ₂ |

<table>
<thead>
<tr>
<th>N</th>
<th>Eᵱ</th>
<th>Eᵱ</th>
<th>Eᵱ</th>
<th>Eᵱ</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Zᵱ</td>
<td>Zᵱ</td>
<td>Zᵱ</td>
<td>Zᵱ</td>
</tr>
<tr>
<td></td>
<td>Uᵱ</td>
<td>Uᵱ</td>
<td>Uᵱ</td>
<td>Uᵱ</td>
</tr>
<tr>
<td></td>
<td>wtᵱ</td>
<td>wtᵱ</td>
<td>wtᵱ</td>
<td>wtᵱ</td>
</tr>
</tbody>
</table>

Figure 2. Snapshops of the particles' condensed histories.
is sampled. Letting the particle carry out a random walk in each step (from state M to state M+1) takes into account the combined effect of many collisions. The sizes of the steps of the random walk, i.e., the pathlength intervals DCEM (equivalent to Δt) are chosen such that the total number of steps should be kept as small as possible. Too large a step size (Δt) may cause inaccuracies in the approximation. However, the length of the Monte Carlo calculation is directly proportional to the total number of steps. Thus, a compromise must be reached such that we choose as big a step size as possible without introducing intolerable inaccuracies. A reasonable initial value is chosen such that the particle with the average energy of the time cycle travels one mean free path, i.e., Δt=D_{avg}/V_{avg}. Different time cycles have different E_{avg} (V_{avg}) and D_{avg} so Δt may vary, but as steady-state is approached, Δt will not change much. By this technique we not only randomize and average the different particle samples, ensemblewise, but we also do the same (on the particle ensemble sets) along the time axis (averaging independent but equivalent time cycles). Trial and error results indicate that the optimum values of Δt (or DCEM) lie between 0.5 (D_{avg}/V_{avg}) and 2.0 (D_{avg}/V_{avg}).

The maximum number of particles (N) that can be processed in each cycle is limited by the amount of storage available in the computer system. For the IBM 360/75 system
at the Illinois Computer Center, the storage of the data for 5,000 particles is easily accomplished. By playing Russian roulette, splitting, or similar techniques, the total number of artificial particles (N) can be increased by an order of magnitude. The end of one time cycle \( t_i \) is the start of the next time cycle \( t_{i+1} \). While keeping track of the particles in a current time cycle at one time, we simultaneously record the data information after each particle reaches the census and extract the output parameter values after all the particles have been processed. In this way the storage requirements are reduced to a minimum level possible.

The limitation on \( N \) by the storage requirement may be further relaxed by the following technique. Instead of recording and tallying the output information and clearing or reinitializing all parameter values at the end of every cycle, we record the accumulated information for \( K \) continuous, and similar but statistically independent time cycles, then renormalize and reinitialize. In this way the total number of particles processed is \( KN \), and if processing of a large number of particles is desired, we can use a larger value of \( K(K>1) \) at the expense of more processing time. The reason for this is based on the fact that the system being simulated is time independent. We have introduced the time variable artificially so that ergodicity holds, namely ensemble averaging is the same as time averaging.
2. Tracing Procedure

The simulation is initialized by introducing N particles with energy $E_{in}$ [sampled from an initial guess for the number density distribution vs. energy], with spatial position $z_i^0$ (uniformly distributed throughout the spatial region), with direction $u_i^0$ ($u=\cos \theta$; $\theta$: angle of flight with z axis) which is uniformly distributed between -1 and +1 and with weight $w_{ti}$. Volume particles are emitted uniformly throughout the medium and isotropically in direction; namely, at $t=t_0$:

\begin{align*}
E_i^0 &= E_{in} \quad (4a) \\
Z_i^0 &= r_i \cdot D \quad (4b) \\
u_i^0 &= 2r_i - 1 \quad (4c)
\end{align*}

where $r_i$: random number, uniformly distributed between 0, 1, or $r_i \in (0, 1)$

d: slab width or cylinder diameter

$w_{ti}, E_{in}$: weight and particle energy determined from initial distribution.

Once the particle data are determined, the geometry routine is entered. This includes the determination of the distance to boundary $d_B$, the distance to collision $d_{col}$, and the distance to census time $d_{cen}$ (clock distance equivalent to $\Delta t$). These distances are defined by:
\[ d_B = [(j+1)\Delta z - Z]/u \text{ if } u > 0 \] (5)
\[ = (Z - j\Delta z)/|u| \text{ if } u < 0 \]
\[ = DLB \text{ (a large number } DLB >> \Delta z \text{) if } u \approx 0 \]

\[ d_{col} = \lambda(E) |lnr| = |lnr|/\sum_t(E) \] (6)

\[ d_{cen} = D_{avg} \cdot DI = V_{avg} \cdot \Delta t \cdot DI \] (7)

where:
- \( r \): random number uniformly distributed
- \( j \): zone number
- \( \Delta z \): spatial interval size
- \( \sum_t \): the total cross-section evaluated at the particles' energy
- \( D_{avg} \): the average value of \( D_{col} \) of \( N \) particles of the previous cycle. It is approximately one mean free path corresponding to \( E_{avg} \): the average energy of the cycle.
- \( DI \): input parameter for controlling the cycle time interval.

The next step depends on which of these distances is the smallest.

a. **The System Without Electric Field (E = 0)**

i) If the smallest distance is \( d_{cen} \): the particle is advanced in position and time to census time.

\[ z' = z + u \cdot d_{cen} \] (8a)
\[ u' = u \] (8b)
\[ E' = E \] (8c)

The particle data is stored, the energy is tallied, and
the density counter is incremented. Then we go on to process the next particle.

ii) If the smallest distance is $d_B$, the particle is advanced to the boundary such that:

$$z' = z + (d_B + 0.001 \Delta z) \cdot u$$  
$$j = \text{INT} \left( \frac{z'}{\Delta z} \right) + 1$$  
$$d'_{\text{col}} = d_{\text{col}} - d_B$$  
$$d'_{\text{cen}} = d_{\text{cen}} - d_B$$

(9a)  
(9b)  
(9c)  
(9d)

where $j$ is the zone number. If $j > JZ$ or $j < 1$, then the particle is considered lost or has escaped from the system, and its history is ended. Otherwise the calculation and selection is repeated until the particle reaches the census time.

iii) If $d_{\text{col}}$ is the smallest, the particle is first advanced to collision point and

$$z' = z + u \cdot d_{\text{col}}$$  
$$d'_{\text{cen}} = d_{\text{cen}} - d_{\text{col}}$$

(10)

Then the type of collision is specified. For specifying the collisional processes, we divide the whole energy range into two regions. Region I: energy range 0 to $E_{D1}$ eV.
Region II: energy range $E_{D1}$ eV to $E_o$ eV. $E_{D1}$ and $E_o$ are specified by input parameters and are referred to as the energy dividing line and source particle energy, respectively.
If the particle's energy is in the range of Region I, three kinds of collisions are possible: elastic collisions, electron-ion recombination, and superelastic collisions (where electrons hitting an excited atom pick-up the excitation energy). If the particle's energy falls in Region II, it can experience four kinds of collisions; namely, elastic collisions and three types of inelastic collisions—excitation to metastable level, excitation to other levels, and ionization. For ionization, the secondary electron so produced shares energy with the primary electron, and the energy of primary electron before collision less the ionization potential is shared between the two electrons. The partition of the shared energy could be determined according to Goodrich's experimental distribution curve (31). But based upon Thomas' (1) calculation experience, the way in which the energy is partitioned does not have a strong influence on any of the calculated mean properties. Thus, for computational simplicity, as Thomas did in all his calculations, we choose the partition ratio 9:1. The secondary electrons produced via ionization are considered as new particles introduced into the system. Thus they are followed from their point of birth just like other particles. This is equivalent to the generation method for processing branched trees of samples.

The type of collision is determined according to Figure 3 using a random number (uniformly distributed
<table>
<thead>
<tr>
<th>Elastic</th>
<th>Excitation to metastable level</th>
<th>Excitation to other levels</th>
<th>Ionization</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
</tbody>
</table>

\[
A = P_{el}
\]

\[
B = P_{el} + P_{met}
\]

\[
C = P_{el} + P_{met} + P_{exc}
\]

\[
P_{el} + P_{met} + P_{exc} + P_{ion} = 1
\]

Figure 3. The probability model for determining the types of collision.
between 0 and 1) to play the law of chances.

b. With Electric Field Present

The coordinate axis z is defined as antiparallel to the electric field \( E_f \). In the presence of electric field \( E_f \), the state of motion of an electron is specified by its speed \( V \) and the direction cosine \( \cos \theta \), where \( \theta \) is the angle made by the line of flight with the z-axis at the time \( t \). If an electron is in a state \([E(V), u(\cos \theta), z, t]\), it will assume (after a flight along the orbit by \( \Delta t \)) the new state \([E'(V'), u'(\cos \theta'), z+\Delta z, t+\Delta t]\). These parameters are connected by the following relations:

\[
\begin{align*}
V' \cdot \sin \theta' &= V \cdot \sin \theta \\
\frac{mV'}{\cos \theta'} &= \frac{mV}{\cos \theta} + eE_f \cdot \Delta t \\
\frac{1}{2} mV'^2 &= \frac{1}{2} mV^2 + eE_f \cdot \Delta z \\
E' &= E + eE_f \cdot \Delta z
\end{align*}
\]

where \( m \) and \(-e\) are the mass and the electric charge of the electron, respectively. Here \( \Delta z \) is the projection of the path length \( L \) on the z axis. It can be obtained as follows:

\[
\Delta z = \int_0^{\Delta z} V_z \, dt = \int_0^{\Delta t} \left( V \cdot u + \frac{E_f}{m} \Delta t \right) \, dt \\
= V \cdot u \cdot \Delta t + \frac{1}{2} \frac{E_f}{m} \cdot \Delta t^2
\]

In carrying out the geometry routine, it is suitable to use time directly as the clock. The determination of the distance to collision \( d_{col} \) (DCOL) and the distance to census
d_{cen}(DCEN) is accomplished as follows:

\[
DCOL = \frac{1}{\lambda(E)} + \frac{1}{\lambda(E')} \tag{16}
\]

\[
DCEN = \sqrt{V_y^2 + V_z^2} \cdot \Delta t \tag{17}
\]

\[
V_y = V \cdot \sin \theta \tag{18}
\]

\[
V_z = V \cdot \cos \theta + \frac{E_f}{2m} \cdot \Delta t \tag{19}
\]

where \(E\): the initial energy of the particle

\(E'\): the energy of the particle after flight time \(\Delta t\).

The next step is to determine which of the distances is smaller. If \(DCEN < DCOL\), then the particle is advanced to the new state

\[
E' = E + E_f \cdot \Delta z \tag{20}
\]

\[
z' = z + \Delta z
\]

\[
\Delta t' = \Delta t - \frac{DCOL \cdot \Delta t}{DCEN} \tag{21}
\]

If \(E' < 0\) or \(z' > D\), then the particle is considered to have escaped from the system. If \(DCEN > DCOL\), the particle is advanced to collision point.

\[
\Delta t' = \Delta t - \frac{DCOL \cdot \Delta t}{DCEN} \tag{22}
\]

then the specific collision type must be determined.
c. **Types of Collisions and Cross Sections**

For the elastic collisions, the particle scatters off with angle of deflection $\theta$ and energy loss $\Delta E$, the following formula is used (36).

$$\Delta E = \frac{2m}{M} (E-E_n) (1-\cos\theta_s)$$ (23)

where $\theta_s$: scattering angle (angle of deflection)

$E$: particle energy

$E_n$: target gas atom energy

$m$: electron mass

$M$: gas atom mass

$E_n$ is selected according to a Maxwellian distribution, and the $\Delta E$ so formed may have either positive or negative values, indicating the loss or gain of energy respectively. For calculational simplicity, instead of selecting the deflection from the elaborate distributions, we use $\pi/4$ or $\pi/2$ deflection approximations, namely $\theta_s = \pi/4$ or $\pi/2$, then $\cos\theta_s = 0$ or 0.70711. Another reasonable alternative is to assume that the direction after collision is equally probably in all directions (isotropic scattering). We select this direction at random with uniform distribution, and based on Eq. 23 calculate $\cos\theta_s$ and $\Delta E$, respectively, where

$$\cos\theta = 2r_i - 1$$ (24)

$$\theta_s = \theta - \theta_i$$
where $\theta_i$: incident angle
$\theta$: outgoing angle after collision

For the three kinds of inelastic collision cross sections we use approximate formula based on experimental data as correlated by Itoh and Musha (2). The formulas are expressed in the form:

$$\sum(E) = \frac{B(E-E_i)}{A+(E-E_i)} + C$$

(25)

where the constants $A$, $B$, $C$, and $E_i$ depend on the kind of collision and scattering angle. For the elastic cross section in helium, we use a single approximate formula for whole effective energy range based on Heylen and Lewis's paper (10), namely:

$$\sum_{el}(E) = 26\exp(-0.04E) \text{ (cm}^{-1})$$

(26)

For the recombination cross section for helium (35), we assume the simple form:

$$\sum_{RCB}(E) = \mu(E).N^+$$

$$\mu(E) = 10^{-9}/V(E)$$

(27)

where $N^+$: ion density
$V$: electron velocity

the superelastic cross section is

$$\sum_{Sel}(E) = \frac{0.109(19.8-E)}{0.25 + (E-19.8)^2} \cdot \frac{N^+}{3\cdot n}$$

(28)
where $N^+$: positive ion density
$D_n$: gas atom density
$E$: electron's energy

3. The Prediction and Correction of Dominant System Parameters

There exists a single parameter, the ion density distribution $N^+(E,z)$, which completely characterizes the nonlinearity of the system. This simplifies the problem and suggests the following predict-and-correct scheme. We assume that the average mean free path (or $\Delta t$) and the ion density distribution $N^+(E,z)$ in the present time cycle also applies to the next time cycle. In other words, in each time cycle we use the predicted values of $\Delta t$ and $N^+(E,z)$ from previous time cycle. At the end of present time cycle, we use the up to date information to renormalize and correct the value to be used for the next time cycle. As steady state is reached, the predicted and corrected values of the parameters converge. From the continuity equation

$$\nabla \cdot J = S - L$$

(29)

where $J$: the particle current flow in or out of the system
$S$: the source
$L$: the sum of losses

To start with, we assume that $\nabla \cdot J = 0$, so that

$$S = L = \alpha(E) N^+(E,z) N^-(E,z)$$

(30)
Here $N^-(E,z)$ is the electron number density, the key quantity we are calculating. To initialize the run we have to predict $N^+(E,z)$. It is reasonable to initially assume this distribution is uniform in $z$, and a maxwellian distribution in $E$, i.e., $N^+(E,z) = N_0 M(E)$. We also assume that $N^-(E,z)$ is the same as $N^+(E,z)$. Then we obtain

$$S = L = \alpha(E) N^+(E,z) N^-(E,z)$$

$$= \alpha(E) [N^+(E,z)]^2 \quad (31)$$
or

$$S_0 \delta(E-E_0) = \alpha(E) N_0^2 M^2(E)$$

Integration of both sides gives

$$S_0 = N_0^2 \int_0^{E_0} \alpha(E) M^2(E) \, dE$$

$$\Rightarrow N_0 = [S_0 / \int_0^{E_0} \alpha(E) M^2(E) \, dE]^{1/2} \quad (32)$$

This gives a starting value for the calculations. In later cycles the renormalization process involves integration of Eq. 29, with $\nabla \cdot J \neq 0$, i.e.,

$$\int \int \frac{dJ(E,z)}{dz} \, dz \, dE = \int \int S(E,z) \, dz \, dE$$

$$- \int \int \alpha(E) N^+(E,z) N^-(E,z) \, dE$$

$$= J_{esc} + J_s$$

which reduces to

$$J(Z=d) - J(Z=0) = S_0 \, d - \int \int \alpha(E) N^+(E,z) N^-(E,z) \, dE$$

$$- J_{esc} + J_s$$
\[
N_i^2 = \frac{S_0 d + J_s - [J(z=d) - J(z=0)] - J_{\text{esc}}}{\int \alpha(E) M(E) n^-(E, z) \, dz \, dE}
\] (33)

where \( i \) refers to \( i \)th time cycle

- \( J_{\text{esc}} \): current due to escaped electrons
- \( J_s \): current due to secondary electrons produced by ionizations
- \( M(E) \): Maxwellian distribution (normalized)
- \( n^-(E) \): electron number density distribution (normalized)
- \( S_0 \): source rate
- \( N_i \): ion density distribution (normalized)

The value of \( N_i \) replaces \( N_0 \) in equation (33) at later time cycles. The value of \( N_i \) (\( N^+ \)) enters the formula for the recombination cross section and superelastic cross section, thus it directly affects the rates of recombination and superelastic collision. Indirectly, it affects the balance of the system, or equivalently the rate of convergence of the Monte Carlo calculation. An adaptive correction procedure is used here. It is based on the value of the "balance factor" \( BF \), defined as the ratio of the particles introduced into the system by source and ionization processes to the particles removed from the system due to recombination and leakage. We adopt a over-relaxation type of correction technique, the over-correction is used to speed up the over-all convergence rate. As defined before, if \( BF > 1 \), the system has more particles introduced
into than eliminated, but if $BF < 1$, the opposite is true.
The over-correction factor is proportional to $(BF)^k$, i.e.,
to the $k$th power of the balance factor. The value of $k$
initially chosen was $k=2$, but an optimum value can be
obtained by trial and error.
CHAPTER III
VARIANCE REDUCTION TECHNIQUES

A. General Background

The Monte Carlo method (or method of statistical trials) consists of solving various problems of computational mathematics by the construction of some random process for each such problem, with the parameters of the process set equal to the required quantities of the problem. These quantities are then approximated by observations of the random process and the computation of its statistical characteristics, which are approximately equal to required parameters [from Schreider (15)]. Every Monte Carlo computation that leads to quantitative results may be regarded as estimating the value of a multiple integral. Suppose we have M random numbers in the computation system. The results will be a vector valued function \( R(\xi_1, \xi_2, \ldots, \xi_M) \) involving the sequence of random numbers \( \xi_1, \xi_2, \ldots, \xi_M \). This is an unbiased estimator of \( \int_0^1 \ldots \int_0^1 R(x_1, x_2, \ldots, x_M) \, dx_1 \ldots dx_M \). For the sake of simplicity, we take the one-dimensional integral as a standard example. Let

\[
\theta = \int_0^1 f(x) \, dx
\]

(34)

where \( f \in L^2(0,1) \) or \( \int_0^1 [f(x)]^2 \, dx \) exists.

Define the relative efficiency of two Monte Carlo methods as follows. Let the methods call for \( n_1 \) and \( n_2 \) units of computing time, respectively, and let the resulting
estimates of \( \theta \) have variances \( \sigma_1^2 \) and \( \sigma_2^2 \). Then the
efficiency of method 2 relative to method 1 is defined as

\[
\frac{n_1 \cdot \sigma_1^2}{n_2 \cdot \sigma_2^2}
\]

(35)

Equation 35 is the product of two terms, the variance ratio
\( \sigma_1^2 / \sigma_2^2 \) and the labor ratio \( n_1/n_2 \). If \( \gamma_1, \gamma_2, \ldots, \gamma_N \)
are independent random numbers with uniform distribution
between 0, 1 or \( \gamma_i \in (0,1) \); then the quantities \( f_i = f(\gamma_i) \)
are the independent random variates with expectation \( \theta \).
Therefore

\[
\bar{f} = \frac{1}{N} \sum_{i=1}^{N} f_i
\]

(36)
is an unbiased estimator of \( \theta \), and its variance is

\[
\frac{1}{n} \int_0^1 (f(x)-\theta)^2 dx = \frac{\sigma^2}{n}
\]

(37)

Accordingly the quantity \( \bar{f} \), which has been determined by
observation of the random process, is approximately equal
to the required quantity \( \theta \) (or in other words, it is an
unbiased estimator of \( \theta \)) with a probability which can be
made as close as required to unity if a large number of
trials is practical. We refer to the estimator \( \bar{f} \) as the
crude Monte Carlo estimator of \( \theta \) (commonly referred to as
Monte Carlo estimation).

Let us introduce another even less efficient method,
namely hit or miss Monte Carlo. Suppose that \( 0 \leq f(x) \leq 1 \).
when $0 \leq x \leq 1$. We may draw the curve $y = f(x)$ in the unit square $0 \leq x, y \leq 1$, and $\Theta$ is the proportion of the area of the square beneath the curve. Stating this formally, we write

$$f(x) = \int_0^1 g(x, y) \, dy$$

(38)

where $g(x, y) = 0$ if $f(x) < y$

$= 1$ if $f(x) \geq y$

Then

$$\Theta = \int_0^1 \int_0^1 g(x, y) \, dx \, dy$$

(39)

The estimator

$$\bar{g} = \frac{1}{n} \sum_{i=1}^{n} g(\xi_{2i-1}, \xi_{2i}) = \frac{n^*}{n}$$

(40)

$n^*$ is the number of occasions where $f(\xi_{2i-1}) > \xi_{2i}$

In other words, we take $n$ trials at random in the unit square, and count the proportion of them which lie below the curve $y = f(x)$.

Historically, a hit or miss method was first propounded in the explanation of Monte Carlo techniques. It is the easiest to understand but one of the least efficient techniques. The rate of convergence and the variance (statistical error) is in general proportional to $1/\sqrt{N}$ where $N$ is the number of samples. It is obviously impractical to gain a significant improvement of accuracy by merely increasing $N$. Therefore more efficient techniques for reducing the variance and/or increasing the efficiency
are very important. Monte Carlo experimentalists need wide experience and background in both the mathematics and the physics of the problem, and they have to exercise considerable ingenuity in distorting and modifying problems in the pursuit of variance reduction techniques. Statistical and inferential procedures are also important in order to extract the most reliable conclusions from the observational data.

The basic requirement of the Monte Carlo calculations is to establish how random numbers may be used to sample a function that describes, in a probabilistic fashion, a physical event. The procedure by which this is done follows from the fundamental principle of Monte Carlo (16), which may be stated as follows. If $p(x)dx$ is the probability of $x$ lying between $x$ and $x+dx$, with $a \leq x \leq b$

$$\int_{a}^{b} p(x)dx = 1$$  \hspace{1cm} (41)

Then \[ r = P(x) = \int_{a}^{x} p(\gamma)d\gamma \]  \hspace{1cm} (42)

determines $x$ uniquely as a function of the random number $r$ which is uniformly distributed on range $(0, 1)$. The quantities $p(x)$, $P(x)$ are the probability density and the probability distribution functions, respectively.

With the development of large computers, the use of the Monte Carlo method in a wide range of problems has
increased rapidly. In particular, the method is applied to various physical problems such as plasmas (15), radiative transfer problems (21), neutron transport and nuclear reactor physics (29), etc.

The random walk type simulation can be briefly described in the following manner. One first introduces the basic physics of the problem into the computer in a probabilistic fashion. A system of coordinates and boundaries are defined and then, as a computer experiment, particles are released from the source. These particles are traced as they diffuse through a prescribed medium following the probabilistic interaction laws. The particles are followed until they escape from the medium or are absorbed or are converted to the thermal field.

The parameters pertinent to the evaluation of the desired quantities are recorded. One continues processing additional particles until adequate statistical estimates of the quantities of interest have been obtained.

When doing a Monte Carlo problem one focusses attention on three main topics. They are:

1. The development of the analogy for the probability processes (simulation).
2. The generation of sample values of the random variables on a given computing machine.
3. The design and use of variance reduction techniques.
The variance reduction methods are often strongly dependent on the probabilistic model, thus the greatest gains are often made by exploiting specific details of the problem, rather than by routine application of general principles.

B. **Importance Sampling**

The general idea of importance sampling is to draw samples from a distribution other than the one suggested by the problem. Then an appropriate weighting factor is introduced to correct the biasing caused by changing the original distribution. If done correctly, the final results are essentially unbiased. The object is to concentrate the distribution of sample points in the parts of the region that are most "important" instead of spreading them out evenly. Thus, the probability of sampling from an interesting region is increased; the probability of sampling from unimportant or less interesting region is correspondingly decreased. Stating this formally, we have

\[ \theta = \int_0^1 f(x)dx = \int_0^1 \frac{f(x)}{g(x)} g(x)dx = \int_0^1 \frac{f(x)}{g(x)} dG(x) \]  

(43)

for any functions \( g \) and \( G \) satisfying

\[ G(x) = \int_0^x g(y)dy \]  

(44)

where \( g \) must be a positive valued function such that
if \( \gamma \) is a random number sampled from the distribution \( G \),
then \( f(\gamma)/g(\gamma) \) has the expectation \( \theta \) and variance

\[
\sigma^2_{f/g} = \int_0^1 \left( \frac{f(x)}{g(x)} - \theta \right)^2 dG(x)
\] (46)

We notice that if \( g(x) = cf(x) \), or if \( g \) is proportional to \( f \) and if \( c = \frac{1}{\theta} \) then \( \sigma^2_{f/g} = 0 \). This perfect situation does not exist, since we do not know \( \theta \). We always obtain an unbiased estimate for positive function \( g \). Our object is to select \( g \) to reduce the standard error of our estimate.

Because the estimate is the average of observed values of \( f/g \), we choose \( g \) such that \( f/g \) is as constant as possible in order to achieve a small sampling variance. While we intend \( g \) to mimic \( f \), we generally restrict our choice of \( g \) to functions simple enough to allow analytic integration.

For the present plasma problem, the importance sampling technique is developed as follows: during the particle's (electron's) random walk, it suffers a large number of collisions with gas atoms, and these collisions may be elastic or inelastic or of other kinds. The probabilities of their occurrence depend on their cross section \( \Sigma_i \). We have

\[
\Sigma_t = \Sigma_1 + \Sigma_2 + \ldots + \Sigma_M
\] (47)
where $\Sigma_i$: the cross section for $i$th process

$\Sigma_t$: total cross section or the summations of the cross sections of all the possible processes that the particle may experience.

Dividing both sides of equation 47 by $\Sigma_t$, we obtain:

$$1 = p_1 + p_2 + \cdots + p_M \tag{48}$$

where $p_i = \frac{\Sigma_i}{\Sigma_t}$ the probability of $i$th process occurring as the particle experiences a collision. If we introduce weights $w_i$ into equation 47 we obtain

$$\Sigma_t' = w_1 \cdot \Sigma_1 + w_2 \cdot \Sigma_2 + \cdots + w_M \cdot \Sigma_M \tag{49}$$

Similarly we have

$$1 = p_1' + p_2' + \cdots + p_M' \tag{50}$$

After the application of the weights, we put more weights on the interesting or more important events. Then their artificial probabilities are greater, therefore we can get better estimates of the important parameters. Coincidentally, such a linear weighting scheme can only be applied to the elastic collision process in our system. The other collision processes involve appreciable changes in the particle’s energy (catastrophic collision). Also cross sections are energy dependent, these nonlinear effects prevent the application of above mentioned weighting scheme to inelastic collision processes. By looking at the cross section values, in the low energy region, the elastic cross-section is by
far the largest by order of magnitude. Others have shown that the elastic process does not strongly influence the estimated parameters that we are interested in (1, 2). As an electron makes an elastic collision, it loses only a small amount of energy while making a small angle deflection. In fact, most other workers simply neglect elastic loses if an electric field is present (1, 2, 10). In our case, with $E_f=0$, elastic effects must be included. In the low energy region ($\approx 0$ eV to 20 eV), the energy of a majority of the particles lies below 1 eV and over 90% collisions are elastic collisions, and only a few collisions are inelastic.

The collision process subroutines are a time-consuming part of the simulation calculations. If we can supress the elastic processes by a factor of $k_{el}$, the chances of all other processes are increased by approximately the same factor. Therefore we convert more computing time to processes which contribute more strongly to the quantities of interest. The following formulation provides the theoretical support for this. In lieu of equations 47-50, the $p_i$ are the original probabilities (undistorted probabilities) while the $p'_i$ are the weighted probabilities. Then the correction factors are introduced as follows; at energy $E$, we have

$$c_1 = \frac{p_1}{p'_1}, \quad c_2 = \frac{p_2}{p'_2}, \ldots \quad c_M = \frac{p_M}{p'_M} \quad (51)$$
or \( c_i = \frac{p_i}{p_1} = \frac{\sum_i \cdot \Sigma^i}{\Sigma_t \cdot \Sigma_i w_i} = \frac{\Sigma^i}{\Sigma_t} \frac{1}{w_1} \) \( \quad (51a) \)

If we let \( w_1 = \frac{1}{k_{el}}, w_2 = w_3 = \ldots w_M = 1 \)

then from (51) we have

\[ C_i = \frac{\sum_i \cdot \Sigma^i}{\Sigma_t \Sigma_i w_1} = \frac{\Sigma^i}{\Sigma_t} \frac{1}{w_1}, \quad c_i = \frac{\Sigma^i}{\Sigma_t} = 1, \quad i \neq 1 \] \( \quad (52) \)

let \( \Sigma_1 = \Sigma_{el} \) namely we introduce weight on elastic cross section only. Since \( \Sigma_{el}(E) \gg \Sigma_i(E), \quad i > 1, \quad \forall \) then

\[ \frac{\sum_i'E_i(E)}{\Sigma_t(E)} \equiv \frac{\Sigma_{el}'(E)}{\Sigma_{el}(E)} \quad w_1 = \frac{1}{k_{el}} \] \( \quad (53) \)

and \( C_i \approx w_i \frac{1}{w_1} = 1, \quad c_i \approx \frac{1}{k_{el}} \quad (i \neq 1) \) \( \quad (53a) \)

Equation 53 holds for low energy region, which is approximately independent of energy.

C. Russian Roulette and Splitting

In general, the sampling is done such that we examine the sample and classify it as being in some sense "interesting" or "uninteresting". We are willing to spend more than average amount of work on the interesting samples and on the contrary, we want to spend less effort on the uninteresting ones. This can be done by splitting the interesting samples into independent branches, thus resulting in more of them, and by killing off some of the uninteresting ones. The first process is splitting and the second Russian Roulette.
The "killing off" is done by a supplementary game of chance. If this game is lost, the sample is killed; if it is won, the sample is counted with an extra weight to make up for the fact that some other samples have been "killed". The game has a certain similarity to the Russian game of chance played with revolvers - whence the name.

For the present plasma system we divide the whole energy spectrum into two regions, the low energy region I (0 eV to $E_{D1}$ eV) and the high energy region II ($E_{D1}$ to $E_0$). Here $E_0$ is energy of source particles, which is the highest energy the particle can have. $E_{D1}$ is the energy region dividing line, which can be chosen as any value between 5 eV and 100 eV depending on the initial guessed energy distribution. In region I, the number density is high, and the elastic process is the dominant process. As we mentioned earlier, the elastic process does not contribute much, so Region I is classified as uninteresting. Each particle in this region (particles with energy less than $E_{D1}$) is assigned a weight $k_1 (k_1 \geq 1)$. In other words, each particle plays the role of $k_1$ particles.

In region II, classified as interesting region, each particle carries a weight $w_2 \left( w_2 = \frac{1}{k_2}, w_2 \leq 1 \right)$. Namely, each particle in this region is split into $k_2$ particles with weight $w_2$.

As particles cross over to other regions, care must be taken to account for killing off and splitting effects.
For instance, a particle in region I may acquire enough energy, either from electric field or by superelastic collision with the excited atoms, to cross over to region II. Then we should generate $k_3$ particles ($k_3 = k_1 \cdot k_2$) in region II with the same energy.

Particles from the high-energy region may lose part of their energy by inelastic collision such that they cross over to the low-energy region. Then killing off should be effective. For every $k_3$ such particles that cross over from region II to region I, only one particle survives. The values of $k_1$ and $k_2$ can be chosen according to the total number of particles in the system to be processed, the source rate, and related accuracy requirements. They are input parameters.

D. Initial Guess

Monte Carlo methods comprise that branch of experimental mathematics which is concerned with experiments on random numbers. The user, like the experimental physicist needs theory and knowledge to give structure and purpose to his experiments, and as experimental work provides growing insight into the nature of a problem and suggest appropriate theory. Good Monte Carlo practice should keep this relationship as a general maxim.

The basic procedure of the Monte Carlo method is the manipulation of random numbers, but these numbers should not
be employed prodigally. Each random number is a potential source of added uncertainty in the final result. Thus it will usually pay to scrutinize each part of a Monte Carlo experiment to see whether that part can not be replaced by an exact theoretical analysis contributing no uncertainty. In other words, exact analysis should replace Monte Carlo sampling wherever possible. Sometimes reliable intuition would aid in increasing the efficiency of Monte Carlo calculations. For the plasma system we are simulating, the straight-forward approach would be to follow N source particles on their life time history, until they lose energy sufficient by collisions to reach a thermal equilibrium. Over 50 inelastic collisions generally occur, and the probability (inelastic cross sections) for inelastic processes is low. Consequently, the particles travel a relatively large distance which consumes a good amount of time. Thus a large number of time cycles would be required before the system reaches steady state. To avoid this, we use our intuition, to start off with a guessed initial distribution. The electron energy distribution f(E) (or electron number density distribution) is a function of the electron's energy E. Initial particle data is generated using this distribution, and this can save considerable irrelevant computations. This idea is simple, yet very useful and effective. Without an initial distribution, the problem would require hundreds of time cycles to
converge. However, with a reasonable guess, good results have been obtained in as little as tens of time cycles. Of course, the better the initial guess, the faster the convergence.

E. Antithetic Variates

When we estimate an unknown parameter $\theta$ by means of an estimator $t$, we may seek another estimator $t'$ having the same (unknown) expectation $\theta$ as $t$ but a strong negative correlation with $t$. Then $\frac{1}{2}(t + t')$ will be an unbiased estimator of $\theta$, and its variance is

$$\text{Var}[\frac{1}{2}(t+t')] = \frac{1}{4}\text{var}(t) + \frac{1}{4}\text{var}(t') + \frac{1}{2}\text{cov}(t,t')$$

(54)

where $\text{cov}(t,t')$ is negative, and

$$\text{var}(t) = \mathcal{E}[(t-u)^2]$$

$$\text{cov}(t,t') = \mathcal{E}[(t-u)(t'-u')]$$

$u,u'$ are the mean of $t,t'$ respectively, and $\mathcal{E}(x)$ is the expectation of the random variable $x$.

It is possible to make $\text{var}(t+t')$ smaller than $\text{var}(t)$ by suitably selection $t'$. For example, $1-r$ is uniformly distributed whenever $r$ is, and if $f(r)$ is the unbiased estimator of $\theta$, so is $f(1-r)$. When $f$ is a monotone function, $f(r)$ and $f(1-r)$ will be negatively correlated. Thus we could take

$$\frac{1}{2}(t+t') = \frac{1}{2} t(r) + \frac{1}{2} f(1-r)$$

(55)
as an estimator of θ.

The name and idea of antithetic variates was first introduced by Hammersley and Morton (13, 14) during attempts to improve Buffon's needle experiment for estimating the value of π (3.1415926).

The main idea of antithetic variates is based on the underlining theorem [from Hammersley and Morton (12)] which can be stated informally as follows: "Whenever we have an estimator consisting of a sum of random variables, it is possible to arrange for a strict functional dependence between them, such that the estimator remains unbiased, while its variance comes arbitrarily close to the smallest that can be attained with these variables". As the name implies, antithetic variates are the set of estimators which mutually compensate each other's variations. Essentially, we rearrange the random variables by permuting finite subintervals in order to make the sum of the rearranged functions as nearly constant as possible. From the practical viewpoint, the mathematical conditions imposed by antithetic variates to the Monte Carlo calculation are quite loose and flexible. It is relatively easy to find a negatively correlated unbiased estimator to produce an efficient variance reduction scheme.

A system of antithetic variates is obtained by simple transformations based on a stratification of the interval (12). The transformations are constructed in such a way.
that the efficiency of the method will increase as a higher power of $M$ (the number of antithetic variates taken). The transformations considered are linear combinations of the values of the function $f$ at a number of points. Thus a correlated stratified sampling technique is developed, and its efficiency generally exceeds that of crude sampling by a factor $M^3$.

The following useful transformations are obtained in this manner:

$$t_1 = \frac{1}{n} \sum_{i=1}^{n} \{af(x_i) + (1-\alpha) f[\alpha+(1-\alpha)x_i]\}$$  \hspace{1cm} (56)$$

$$= \frac{1}{n} \sum_{i=1}^{n} \alpha f(x_i)$$

$$t_2 = \frac{1}{n} \sum_{i=1}^{n} \{af(x_i) + (1-\alpha) f[1-(1-\alpha)x_i]\}$$  \hspace{1cm} (57)$$

$$= \frac{1}{n} \sum_{i=1}^{n} \alpha f(x_i)$$

The value of $\alpha$ is between 0 and 1, the variance $\text{var}(t_j)$ has a minimum at some value of $\alpha$. An adequate rule of thumb is to choose $\alpha$ by finding the root of

$$f(\alpha) = (1-\alpha) f(1) + \alpha f(0)$$  \hspace{1cm} (58)$$

Another useful transformation is

$$t_3 = \frac{1}{m} \sum_{j=0}^{m} \frac{f(x+j)}{m} = \sum_{m} f(x)$$  \hspace{1cm} (59)$$
In general, the variance of crude Monte Carlo is 
\[ \text{var}(t) = O(m^{-1}) \] while the variance of \( t_3 \) is 
\[ \text{var}(t_3) = O(m^{-2M}), \quad (M > 1). \] The simplest form of antithetic transformations have proved quite successful in neutron transport calculations.

The general notion and a simple form of antithetic variates are applied in our present calculations. High efficiency gain can be obtained easily if it is used sparingly and at judicious points in the computation. We have used the random number \( r_i \) (\( r_i \in (0,1) \)) most of the time for playing the law of chances, and \( 1-r_i \) is another random number of the same kind which is the image of \( r_i \). The two are used together in good many places where repeated use of \( r_i \) is required. This serves the purpose of two antithetic variates and cuts the labor in half (same random number \( r_i \) being used twice). Equations 55 and 59 have been used in the initial data preparation program and source routine where initial particles states \( (E_i, u_i, z_i, w_i) \), \( E_i \) are generated according to input initial distribution.

For instance:

\[
\begin{align*}
  u_i &= 2r_i - 1, \quad u_{i+1} = 2(1-r_i) - 1 \\
  z_i &= r_i \cdot d, \quad z_{i+1} = (1-r_i) \cdot d \\
  E_i &= E_k + r_i \Delta E_k, \quad E_{i+1} = E_k + (1-r_i) \Delta E_k \\
  E_i &= E_0 \text{ for source particles.}
\end{align*}
\]

Here \( k \) is an energy index, or \( k \)th energy interval.

Wherever \( r_i \) appears, \( r_i \) and \( 1-r_i \) are used interchangeably.
Thus we are generating source particles from diametrically opposite directions, or from symmetric locations.

F. Double History (Double Processing) Technique

In the particle tracing routine, using the general idea of antithetic variate technique, we have adopted a "so-called" double history processing technique which is described as follows:

a) In the course of tracing the particle's history, in each time cycle, we use the transformed random number $\lambda \cdot |\ln z|$ to decide the distance to collision $d_{\text{coll}}$ and go on until the particle reaches census time. However, we process the same particle twice under the identical initial condition except that we use $\lambda \cdot |\ln(1-r)|$ as the distance to collision $d_{\text{coll}}$. Both sets of history are tallied but only one set of particle data is stored. This way each particle is processed twice independently but correlated in such a way that they compensate each other's variation (variance).

b) Another alternative for double history processing is to play a dual direction random walk instead of dual collision distances. We process the same particle twice in such a way it looks like there are two identical particles emerging into the system in two diametrically opposite directions. Again both histories are tallied, but only the original particle's data is stored.
The two versions of the double history processing technique increase the labor (computation time) by a factor of 1.2 to 1.5. The efficiency gain is difficult to specify. At least it is certain that the gain is more than double the total number of the particles in the system.

The purpose of dual processing is to compensate each particle's variation and minimize the variance (statistical errors). One set of particle data in dual processing is thrown away, while the other set is stored just like normal processing. Which one of the two resulting sets is saved is completely random.
CHAPTER IV

MONTE CARLO SIMULATION CODE

A. Introduction

This chapter presents the complete Monte Carlo simulation program, including an input, output, main flow chart, and the background theory and techniques incorporated. This program implements the condensed history approach that was described in Chapter II together with various variance reduction techniques described in Chapter III.

The basic Monte Carlo calculation can be described most easily by reference to the simplified flow chart shown in Figure 4. By means of this flow chart we shall follow the particle tracing process, making relatively brief comments on the various steps implied by each box, on each routine, and on the main program variables. The program is written in FORTRAN IV and is designed to be run on IBM 360/75 under the HASP-MVT system.

B. Calculational Procedure and Program Description

To obtain an overall view of the calculations, we note that there are three basic steps involved in the simulation process. The first is the initialization of all parameter values, i.e., prepare the particle data and predict the unknown dominate parameter values. The second is the main loop of particle tracing which generates census information. The third uses the census information and takes tallies to estimate the histograms (electron energy or number density
1. Input control

2. Initialization and particle data generation (E, Z, U, WT)

3. Follow initial source particles to census, tally particle states
   - last source particle? No
   - Yes
     - Follow residue particles to census and tally states
     - last residue particle? No
     - Yes
       - From state tallies (particle histories) calculate estimated parameter values
       - Renormalize and correct the ion density, reinitialize parameter values increment t^n to t^{n+1} time cycles and print the output quantities

4. If a prescribed number of time cycles have been processed, output and stop.

Figure 4. Flow chart of the Monte Carlo simulation code.
frequency distribution vs. energy), and parameter values. Finally the dominate parameter values are corrected and the program returns to the first step.

Let us follow the logic flow diagram (Figure 4) in the order of the number marked on the left hand side of the boxes, and describe their individual functions.

1. Input Control: The main program is usually compiled and the loaded version is then stored on disk storage, ready for execution. In order to offer a wide flexibility, the following input parameters are used to control the program runs.

a) NDISK: program start control. The program can start from the beginning (cycle 0), or it can start from the stopping point of the previous run. This feature enables long runs to be broken up into several short runs. It also offers debugging flexibility and economy of computer time. While a long run may run up to 150 time cycles, we can always break it up into multiple numbers of fives and tens cycles. Thus we can adjust other parameter values on a cut and try basis without wasting much computing time. The input parameter for this control purpose is NDISK: input 0 for storing from very beginning; input 1 for starting from the latest tenth cycle of last
run; input 2 for starting from the latest fifth cycle.

b) K1, K2, and ED1: Regional weights and dividing energy. The Russian Roulette and splitting variance reduction techniques use three input parameters to control the amount of killing off and/or splitting of particles. This also determines where to divide energy regions for the Russian Roulette and splitting, assuming that the option to use Roulette is selected. The input parameters are K1: the weight of particles in region I (K1 ≥ 1), K2: the inverse of the weight w2 of particles in region II where w2 = 1/K2 (K 2 ≥ 1), ED1: the energy dividing line, namely (0 to ED1) is region I, (ED1 to E₀) region II where the energies are in electron volts.

c) NTC: The total number of time cycles to be run.

DTC: The input parameter to control the length of census time Δt of the time cycles. In terms of distance travelled by the particles, Δt is equal to a fraction or multiple of one mean free path , corresponding to average energy of particles in the system.
d) NTP: The total number of particles that actually exist in the simulation system.

NTW: The artificial* total number of particles in the system during Russian Roulette and splitting games.

NS: The number of source particles injected into the system during each time cycle.

e) KELC: The input parameter that controls the importance sampling option of the program, i.e., the multiplying factor used to suppress the elastic cross section (reduces the probability of elastic process by a factor KELC).

f) EDP: The input value of the ratio of the electric field strength to the pressure.

PDPO: The input value for the reduced pressure of the system \( \frac{P}{P_0} \) where \( P_0 = 1 \) torr.

g) NIP: The input parameter used to control double processing: \( NIP = 2 \) for double processing, \( NIP = 1 \) for single processing.

IUSN: IUSN = 1 for dual-collision type double processing

IUSN = -1 for dual-direction type double processing

*The total number modified by scaling and weighting factors.
h) D: The input value of the width of the slab (diameter of the cylinder) which defines the external boundaries of the medium.

JZ: The number of zones the slab is divided into, or the number of internal zones.

i) NCUM: Input control to give cumulative results (time average) over the time cycles or a cumulative average over NCUM time cycles.

NAVJ: Input constant to control the power of $\Delta t$, (or $\Delta D$) namely $\Delta t_i^{NAVP}$ is the weight of the cycle to be used for cumulative results. These two parameters are used to implement antithetic variates techniques along the time axis.

j) EO: The energy of source particles.

SR: The rate of the source particles entering the system.

k) FID: An array that contains the initial guess distribution (number density vs. energy).

l) NUSC: An input parameter used to control the up-scattering elastic collision process; $NUSC = 1$: with upscattering, $NUSC = 0$: without upscattering.

2. Data preparation: The initialization is entered only once during the calculation, where the time cycle is 0 (or at time $t_0$). After the input phase, all the pertinent
parameters are set to their proper permanent or temporary values. Before entering the next phase to trace the particle histories, several dominate parameter values must be predicted. The data preparation routine is called in to generate particle data \( (E_i: \text{energy}, u_i: \text{direction of travel}, Z_i: \text{position}, W_{ti}: \text{weight}) \) based on the guess for the initial distribution (generate machine particles). Then the average energy of the system is calculated and is used to predict the census time \( (\Delta t \text{ or equivalently DCEN}) \) as well as the number of ions \( (N^+) \).

3. Particle tracing: With the point of origin, direction of propagation, and energy (velocity) known for the initial source particles, the next phase follows the particles through a series of stochastic interactions until the census time is reached, and the particle data and parameters values are tallied and updated. During the course of tracing the particle histories, all the relevant events are recorded and later on can be interrogated to provide the output quantities of interest. This section of the program is the central part of the main program. It contains the main loop for tracing all the particles in the system in a self consistent way. Most of the variance reduction techniques are incorporated here, including double processing, Russian Roulette, and splitting, as well as the various versions of
importance sampling techniques.

4. Secondary particle tracing: The residue particles, i.e., particles generated by ionization collision process (the secondary electrons so produced) are treated in all respects like the regular source particles, except that their starting positions and time coordinates as well as directions are those associated with the particles that generated them. These residue particles may go further and produce other residue particles (secondary electrons) if their energy is high enough, thus forming a short production chain. The probability of such chain production is small however, since the average energy of the secondaries is quite low. The end of the main loop comes after all such residue particles are processed.

5. Data calculations: The details of the phase of the computations depend upon the specific goals and character of study. Our primary interest is the particle energy distribution (or number density vs. energy). This frequency distribution and other pertinent parameters are estimated (or calculated) using the stored data from various parameter counters in the main loop (tallied information) as well as tallied histories. The quantities of interest include local values (for the current time cycle only) and the cumulated values from several time cycles. After these calculations, the next phase is to prepare for the processing of new time
60

cycle.

6. Normalization and printout: Up to this stage, all the output quantities have been calculated. Next the output information is printed out, and at the same time, if the cycle number is the multiple of five, all the particle data and related parameter values are stored on the disk. Then the next run can be read off of the disk and the process can proceed from where we left off the last time. All the parameter values are again initialized, and new source particles are generated for the next time cycle. Particles that have escaped from the system or have been absorbed create holes in the particle data storage which are filled up by the new source particles and the residue particles produced during the time cycle. Before going on to the next cycle, the ion density is renormalized and corrected in a semiadaptive way according to a balance factor BF. The ion density distribution parameter \( N^+ \) is a dominate parameter since it affects several processes such as the rate of recombination and the rate of super-elastic collision (refer to Equations 27 and 28). It is corrected in such a way as to accelerate convergence, till the system reaches a balanced situation. Then increment time from \( t^n \) to \( t^{n+1} \).

The output quantities are the following:

a) The kind of gas in the medium, source particle energy, pressure, slab thickness, and \( \Delta t \) the time interval of this cycle.
b) The number of time cycles, the time in seconds, the number of source particles generated during the cycle time, and the average energy of the particles in the system.

c) The number density distribution vs. energy, the single cycle distribution, the cumulated distribution, the number-density deviation, the average-energy deviation, the average energy, and the energy interval vs. energy.

d) The number density and the average energy vs. z (position zone).

e) The ion distribution ($N_+$), the number of ionizations ($N_{IZN}$), the number of escaped particles ($N_{ES}$), the number of absorbed or recombined particles ($N_{AS}$), the number of super-elastic collisions ($N_{SE}$), the number of excitations to metastable levels ($N_{MET}$), and the number of excitations to other levels ($N_{EXC}$).

f) All the input control parameters as listed in section IV-1.

g) The electric field strength (EF), the source rate, the W-value, the average energy of escaped particles, and the average energy of absorbed particles.

h) The number check on the density distributions.

All the above outputs include both current cycle values and the cumulated values over the past cycles.
7. Final checks: The last stage is to check whether the prescribed number of time cycles have been processed, adequate statistics have been collected, and all the useful data have been stored on the disk. If so, the program stops.

Further programming details are provided in Appendix
CHAPTER V

RESULTS

A. Convergence and Reliability

The present Monte Carlo Model is a predict-correct iterative scheme, with semiadaptive control ability. For such an iterative procedure, our main criteria for the evaluation and determination of convergence are based on the following inferences:

1. The balance factor (BF) as defined in Chapter II (Section II-B-3) signifies whether or not the system has reached steady state, i.e., converged to a final solution. At steady state, the value of BF should be approximately equal to unity. It may approach to unity in both directions, and due to semiadaptive control procedure used, it may oscillate around unity. Larger amplitude oscillations are observed at first, but they rapidly damp out.

2. The estimated values of the major parameters, e.g. the average energy and the W-value show a similar fluctuation, and then they converge to nearly constant values as steady state is approached.

3. The validity of the convergence has been checked using test problems where analytic results are available. A good comparison has been obtained, showing that the program and the model are working correctly, free of obscure errors.
B. Error Estimation and Analysis

1. Error Analysis

1. As mentioned in early chapters, Monte Carlo simulation is equivalent to performing mathematical experiments, the results being based on observations of such experiments. From this analogy we can derive the following procedures for the analysis of errors or accuracies of our calculations.

It is convenient to subdivide experimental errors into three broad types, namely: random errors, systematic errors, and blunders. In general, the experimental error is some additive function of all three, while blunders can hopefully be eliminated. We shall describe them in detail in following sections.

a. Random errors - such errors are of great concern in Monte Carlo type calculations. They generally involve statistical fluctuations or deviations, representing the difference between the singly measured value and the best value of a set of measurements, i.e., the difference between the arithmetic mean and the true mean. Sometimes such errors are referred to as elementary (or inherent) errors in the measurements. The commonly used measure of random errors is the variance or standard deviation.

b. Systematic error - a systematic error tends to have the same algebraic sign, i.e., it is either an additive or subtractive quantity introduced in the measurement
process. It is an unpleasant and insidious contribution which is not generally amenable to statistical treatment. Thus such errors seriously impair the reliability of the estimation. Typical examples include:

i) Incorrect assumptions or approximation in the representation of certain processes.

ii) Constructional faults and mistakes in the algorithms or subroutines.

iii) Inadequate regard of constancy of experimental conditions or inadequate sampling techniques (biased).

c. Blunders - these are outright mistakes which should be corrected by all means. Possible examples include:

i) Incorrect logic, misunderstanding of the problems.

ii) Errors in transcription of data.

iii) Mistakes in constants used.

iv) Confusion of units.

In any type of calculation, the systematic error as well as blunders should be removed. One way for detecting such errors is to compare the program against some known reliable solutions. Only the random errors are subject to reduction by the various treatments amenable to attack by variance reduction techniques.

The precision in an estimated (mean) value is proportional to the reciprocal of the statistical error and is
high if the statistical error is small, i.e. the accuracy is high if the net systematic error is small. Usually, but not necessarily, high accuracy implies small statistical errors as well.

As discussed in later sections, our model and program has been checked against several test problems, and good agreement has been obtained. Therefore, it can be assumed that any serious systematic errors have been removed and that the program is free of blunders. Thus, we shall concentrate on the analysis of random errors.

No thorough analysis of variance has been made. Due to the intricate way that the histories calculated during a given time cycle depend on previous time cycles, a meaningful variance estimate beyond the first time cycle seems to be out of the question. However as the system reaches steady state, the deviations or variations (statistical fluctuations) of the estimated parameters can be estimated by means of standard deviation (or variance) over subsequent time cycles. Namely, we use the following formulation:

\[ s_i = \left( \frac{1}{n} \sum_{j=1}^{n} \left( f_{ij} - \bar{f}_i \right)^2 \right)^{1/2} \]  

(61)

where \( s_i \): the standard deviation at the \( i \)th energy interval over \( n \) consecutive time cycles.

\( f_{ij} \): the parameter value (density distribution function of \( i \)th energy interval, at \( j \)th time cycle.

\( \bar{f}_i \): the average parameter value of \( i \)th interval over \( n \) time cycles.
A heuristic approach was adopted to measure local deviations or statistical fluctuations at each time cycle. This provides a relative magnitude of deviation or dispersion which can be used to compare the various variance reduction techniques. The approach is described as follows:

The estimated major frequency distribution (the electron energy density distribution function in our case) is obtained in the form of histogram. The abscissa of the histogram is divided into so-called class intervals (energy intervals in our calculations). In each such interval an erect rectangle (block) of height \( f_i \) and width \( E_i \) is formed. This block-area type of distribution is the fitted frequency distribution curve. We devise a numerical description of local deviations by defining:

i) the location index of the center of the erected rectangles in the histogram as \( i \) and with abscissa value \( E_i \).

ii) the spread or dispersion around the center.

The total number of particles in each class interval (energy interval) is proportional to the area of the rectangular flock \( (f_i \Delta E_i) \), the particle's energy in this class interval falls in the range \( (E_i - \Delta E_i/2, E_i + \Delta E_i/2) \) with overall average energy \( \bar{E}_i = E_i \). Then the following relationship holds (refer to Figure 5):
Figure 5. Rectangular block representation of the electron energy distribution histogram.
\[ \frac{\Delta E_i}{\Delta E_i} = \frac{\Delta f_i}{f_i} \]

or \[ \Delta f_i = f_i \frac{\Delta E_i}{\Delta E_i} \] (62)

where \( \Delta E_i \) is the deviation of the average energy \( E_i \) from \( E_i \) or \( \Delta E_i = E_i - E_i \). Also \( \Delta f_i \) is the corresponding deviation of the frequency distribution at location index \( i \).

Equation (62) provides a relative measure of the amplitude of the deviation of density distribution curve at energy \( E_i \) which can be used for comparing the efficiency of different variance reduction techniques as well as provide an heuristic clue to the modification of the fitted density distribution curve.

2. Data Filtering

The random errors (statistical fluctuations) result in bumps and ridges in the estimated frequency distribution function. It may be possible to alleviate many of these distortions by means of data smoothing techniques (data filtering).

Two types of data smoothing techniques commonly used in the processing of time series might be used here:

a. Moving averages - this technique operates by replacing each point of the frequency distribution function (height of the histogram at \( E_iV_i \)) at time cycle \( t_j \) with an average value of several subsequent points in the time cycle series. Thus if \( f_{ij} \) is the value
of frequency function at \(i\)th energy interval and \(j\)th time cycle, then \(f_{ij}\) is replaced by

\[
\sum_{j=1}^{M} f_{ij} \cdot \frac{W_j}{M}
\]

where \(W_j\), the weighting factor for the \(j\)th time cycle, usually has the value of 1 or we can define it as a function of \(\Delta t_j\). The value of \(M\) is the number of successive points of the time cycles to be included in each average.

b. Parabolic filtering - this is similar to the moving averages technique, but \(M\) is selected as an odd number, the point to be replaced is located in the center of these \(M\) points, and a parabola is fitted to these points by means of least squares. Thus it is replaced by the corresponding point \(f_i\) on the derived parabola. The process is repeated by shifting the center point to the next one in the time cycle series.

The technique of moving averages (a) is employed in the program, because of its simplicity and the fact that the parabolic smoothing technique is too sensitive to the changes in the distribution function. Note that this smoothing (filtering) technique only becomes effective as the simulated system reaches steady state.
C. Solutions of the Problem

1. Preliminary Checks

In order to verify that the model is valid and correctly working, and the computer program is free of bugs, blunders, and systematic errors, we solved a specific problem first reported by J. A. Smit (11) and later extended by Heylen and Lewis (10). Their solutions for the electron energy distribution were obtained analytically based on the Boltzmann transport equation for the special case characterized as follows:

i) electric field present

ii) no external sources

iii) infinite size medium of background noble gases.

The comparisons are made for helium gas at $\frac{E_f}{p} = 4, 5, 10 \text{ V}\cdot\text{cm}^{-1}$. ($\frac{E_f}{p}$: electric field strength to pressure ratio). The results are shown in Figures 6, 7 and 8. Both Smit and Heylen’s results have taken into account all the collision processes including those due to elastic loss as well as inelastic losses. Heylen’s results is more up-to-date, for low $\frac{E_f}{p}$ values only $\frac{E_f}{p} = 5$ is given, which was chosen for the present comparison. In so far as possible, the same cross sections as Smit and Heylen and Lewis used were incorporated in the present calculations. The slight discrepancies apparent at the low-energy end is probably due to differences in treating the scattering process, and
Figure 6. Comparison of the electron energy distribution for He at $E_f/p = 4$ for
a) Smit's calculation (1936) ($E = 6.4$)
b) present calculation ($E = 6.51$)
Figure 7. Comparison of electron energy distribution for He at $E_f/p = 5$

- a) present calculation ($E = 6.64$)
- b) Heylen and Lewis' result ($E = 6.51$)
Figure 8. Comparison of electron energy distribution for He at $E_f/p = 10$

a) Smit's result ($E = 7.7$)

b) present calculation ($E = 7.83$)
the many approximations required in their analytic solutions. Also, some of the cross sections used in this calculations are experimental values, which may differ by some amount from that used for analytic solutions.

For the high energy region, (near the source energy) checks have been made against the analytic solutions reported by Lo and Miley (32). As shown in Figure 9, the general shape of the electron energy flux distributions are in reasonable agreement. In this case it was not practical to use the same cross sections as incorporated in the analytic solutions, and this may account for some of the differences observed. Also the analytic solutions are obtained for an infinite medium vs. a 1 cm slab in the present case. Thus the analytic solutions do not allow leakage, although this becomes important in the present calculation for low pressures. Lo and Miley's results are plotted in Figure 9 for the source energy $E_0 = 1500$ eV and $E_0 = 500$ eV, whereas the present results are for $E_0 = 1000$ eV. The plot shown is for the flux density (normalized) instead of energy number density. They are related as follows:

$$\varphi(E) = f(E) \cdot V(E)$$

(63)

where $\varphi(E)$: electron flux density

$f(E)$: electron number density

$V(E)$: Electron kinetic velocity
Figure 9. Comparison of the high-energy portion of flux density curve
a) $E_0 = 1500$ eV and $E_0 = 500$ eV: Lo and Miley’s result
b) $E_0 = 1000$ eV: present calculations
Since we only intend to compare general shapes of distribution, the normalizations used for the various curve, shown here are arbitrarily chosen for optimum display of information.

2. The Electron Energy Distribution Function without Electric Field

The calculations for electron energy distribution functions for helium with a high energy electron source, but no applied electric field (Figures 10 to 14) were carried out at two different pressures and various source rates. The slab thickness D is chosen to be fixed at $D = 1 \text{ cm}$, which is equal to the actual diameter of cylindrical tube used in the laboratory (7). The two pressures are chosen to be $P = 10 \text{ torr}$ and $P = 760 \text{ torr}$, corresponding to a typical low-pressure case and a normal atmospheric pressure.

We have chosen two different source energies ($E_0$), one at $E_0 = 1,000 \text{ eV}$ and the other at $E_0 = 70 \text{ eV}$. These values roughly correspond to the highest energy and the average energy in the primary electron energy spectrum for MeV alpha-particle irradiation of helium as done in the experimental studies conducted at the University of Illinois (6).

The range of interest for the source rates for a boron coated tube irradiated in the Illinois TRIGA reactor lies between $10^{14}$ to $10^{18} \text{ (#/cm}^3\text{-sec)}$. However, results shown for source rates above about $10^{16}$ are questionable since the resulting ionization density becomes large enough that
electron-electron collisions become important, and this process was neglected here.

The major quantity of interest here is the electron energy distribution function which were defined as $n^-(E)$ in Chapter II. For simplicity, we represent the normalized distribution corresponding to $n^-(E)$ by $f(E)$. This normalization is defined by:

$$\int_0^E f(E) \, dE = 1 \tag{64}$$

For the plasma system without electric field ($E_f = 0$), the distribution functions with source energy $E_o = 1000$ eV are shown in Figures 10 and 11 for the pressure $p = 10$ torr, and $p = 760$ torr respectively. Similar distribution functions for $E_o = 70$ eV are shown in Figures 12 and 13.

An idealized Maxwellian distribution plus $1/E$ tail are shown in Figures 10 to 13 for reference. (The Maxwellian plus $1/E$ distribution was used as the initial distribution for these calculations. Note that since this curve is for reference only, it is arbitrarily normalized to a point on the Maxwellian curve and does not satisfy Equation 64.)

For the range of source rates mentioned, the calculated energy distribution for helium (He) are reasonably Maxwellian at low energy region. The differences due to variations in the source rates show up at the high energy end, especially near the source energy, where the magnitudes seem to depend linearly on the values of source rate. This is somewhat
deceptive, however, since the absolute magnitude at any point is due to the normalization. (This point is stressed again later in connection with Figure 24).

The fact that the distributions drop off rapidly for energies above about 20 eV is attributed to the fact that the threshold energy for inelastic processes is 20 eV, above which energy the ionization and excitation processes bring more electrons into the low-energy region.

In order to investigate the change in distribution function at very high source rates, a calculation for $S_o = 10^{22}$ particles/cm$^3$-sec was carried out. As mentioned earlier, the neglect of electron-electron collisions makes the accuracy of this calculation questionable. Still the trends are of interest, and it is seen from Figure 14 that the distribution curve changes drastically. In fact, it is no longer Maxwellian at the low-energy end. Due to the high source rate, more high-energy particles and more particles with energy in the range from 0.3 eV to 20 eV are present in the system.

As discussed earlier, the distribution curve drops sharply for energies above the excitation threshold energy (at $E = 20$ eV) due to the dominance of ionization and excitation collision processes. At the higher pressure, the distribution curves decrease more rapidly compared to low pressure, since increased scattering slows the electrons down more rapidly. This effect is shown in Figures 15 and 16,
Figure 10. The electron energy distribution for He without an electric field but different source rates at $p = 10$ torr, $E_o = 1000$ eV and Maxwellian plus $1/E$ distribution.
Figure 11. The electron energy distribution for He without an electric field but different source rates at $p = 760$ torr, $E_0 = 1000$ eV and Maxwellian plus $1/E$ distribution.
Figure 12. The electron energy distribution for He without an electric field but different source rates at p = 10 torr, $E_0 = 70$ eV and Maxwellian plus $1/E$ distribution.
Figure 13. The electron energy distribution for He without an electric field but different source rates at $p = 760$ torr, $E_0 = 70$ eV and Maxwellian plus $1/E$ distribution.
Figure 14. Comparison of $f(E)$ for He without an electric field (source rate $S_0 = 10^{22}$ #/cm$^3$-sec) with the standard Maxwellian distribution plus $1/E$ tail.
Figure 15. The electron energy distribution for He, without an electric field, for different pressure at $S_o = 10^{16}$, $E_0 = 1000$ eV.
Figure 16. The electron energy distribution for He, without an electric field, for different pressure at $S_0 = 10^{16}$, $E_0 = 70$ eV.
for the same source rate \( S_0 = 10^{16} \, (#/\text{cm}^3 \cdot \text{sec}) \) and source energy, but two different pressures. The reduction in magnitude of the high energy tail as the pressure increases is clearly illustrated.

In these calculations, the rate of both leakage and absorption (recombination) are strongly dependent on the pressure. At higher pressures (e.g., \( p = 760 \, \text{torr} \)), very few particles leak out of the system, and the loss are dominated by the absorption process. For lower pressure (\( p = 10 \, \text{torr} \)), fewer particles are absorbed, while more escape or leak out of the system. Also, more particles escape in the high-energy region, but the absorbed particles are mainly in the low-energy region.

The estimated major parameters are summarized in Table 1, which includes the average energy of the system (\( \bar{E} \)), the number of positive ions (\( N^+ \)) and the W-values for various combinations of source rate \( S_0 \), source energy \( E_0 \) and pressure \( p \).

The W-value is defined as the amount of energy required (input) to produce an ionization pair (positive ion and electron), or

\[
W\text{-value} = \frac{(N_s E_0 - E_{Tes} - E_{Tab})}{N_z} \tag{65}
\]

Where \( N_s \): the number of source particles introduced into the system during the period \( \Delta t \).

\( E_{Tes} \): the total energy lost with escaping particles during the time \( \Delta t \).
Table 1. Calculated Parameter Values for the Plasma Without an Electric Field

<table>
<thead>
<tr>
<th>*P</th>
<th>E₀</th>
<th>S₀</th>
<th>̅E</th>
<th>N⁺</th>
<th>W-value</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>(torr)</td>
<td>(eV)</td>
<td>(#/cm²·sec)</td>
<td>(eV)</td>
<td>(#)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1000</td>
<td>10¹⁸</td>
<td>0.171</td>
<td>0.125 x 10¹⁴</td>
<td>41 ~ 63</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1000</td>
<td>10¹⁶</td>
<td>0.157</td>
<td>0.125 x 10¹³</td>
<td>35 ~ 53</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1000</td>
<td>10¹⁴</td>
<td>0.111</td>
<td>0.125 x 10¹²</td>
<td>49 ~ 58</td>
<td></td>
</tr>
<tr>
<td>760</td>
<td>1000</td>
<td>10¹⁸</td>
<td>0.167</td>
<td>0.583 x 10¹⁵</td>
<td>32 ~ 38</td>
<td></td>
</tr>
<tr>
<td>760</td>
<td>1000</td>
<td>10¹⁶</td>
<td>0.123</td>
<td>0.446 x 10¹⁴</td>
<td>40 ~ 65</td>
<td></td>
</tr>
<tr>
<td>760</td>
<td>1000</td>
<td>10¹⁴</td>
<td>0.104</td>
<td>0.67 x 10¹³</td>
<td>35 ~ 54</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>70</td>
<td>10¹⁸</td>
<td>0.104</td>
<td>0.125 x 10¹⁴</td>
<td>50 ~ 62</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>70</td>
<td>10¹⁶</td>
<td>0.082</td>
<td>0.125 x 10¹³</td>
<td>47 ~ 63</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>70</td>
<td>10¹⁴</td>
<td>0.075</td>
<td>0.124 x 10¹²</td>
<td>40 ~ 57</td>
<td></td>
</tr>
<tr>
<td>760</td>
<td>70</td>
<td>10¹⁸</td>
<td>0.082</td>
<td>0.16 x 10¹⁵</td>
<td>47 ~ 65</td>
<td></td>
</tr>
<tr>
<td>760</td>
<td>70</td>
<td>10¹⁶</td>
<td>0.075</td>
<td>0.141 x 10¹⁴</td>
<td>42 ~ 58</td>
<td></td>
</tr>
<tr>
<td>760</td>
<td>70</td>
<td>10¹⁴</td>
<td>0.075</td>
<td>0.157 x 10¹³</td>
<td>39 ~ 55</td>
<td></td>
</tr>
</tbody>
</table>

Maxwellian distribution

|        | 0    | 0.05 | ---  | ---  |

*P: pressure, E₀: source energy, S₀: source rate, ̅E: average energy of the particles, N⁺: number of positive ions.
$E_{ab}$: the total energy lost with recombining particles during the time $\Delta t$

$N_z$: the total number of ionizations produced during the time period $\Delta t$.

As seen from Table 1, the average energy $E$ tends to be higher for higher source rates and higher source energy, and it is lower for higher pressures. However, the differences are in general very small. (All the average energies are consistently greater than that of a Maxwellian distribution). The values for ion density $N^+$ are roughly proportional to the square root of source rate, and are higher for higher pressures, virtually independent of source energy.

The $W$-values are quite sensitive to statistical fluctuations, and the ranges of variation obtained over the last 10 time cycles in a given calculation are shown in Table 1. The $W$-values obtained here do not show any clear dependence on the system parameters, thus variation indicated may be totally due to the statistical fluctuations. This is not too surprising since the $W$-value is known to be roughly independent of particle energy, etc. The $W$-value for helium obtained experimentally by Jesse and Sadouski (34) is 43. A reasonable value to choose for the $W$-value from Table 1 would be 50, which is little bit higher than the experimental result. The reason for this discrepancy is not clear, but it may be due to small inaccuracies in the ionization-excitation cross section employed here. The $W$-value is, by definition, quite sensitive to the details
of the ratio of the ionization to the excitation cross section as a function of energy.

3. The Electron Energy Distribution Function with an Applied Electric Field

The calculated electron energy distribution functions in He with an applied electric field \( E_f/p = 10 \) are plotted in Figures 17 to 21, in a manner similar to that in the previous section. All these calculations are carried out at the electric field to pressure ratio \( E_f/p = 10 \). Smit's result (without source or \( S_o = 0 \)) are plotted together with present calculations at given pressure and source energy \( (p = 10, 760 \text{ and } E_o = 1000 \text{ eV, } 70 \text{ eV}) \), with three different source rates. It is observed that the distribution functions are highly non-Maxwellian. For the source rate in the range \( 10^{14} \text{#/cm}^3\text{-sec} \) to \( 10^{18} \text{#/cm}^3\text{-sec} \), the distribution curves at low energy end are quite close to the Smit's results (11). At the high energy end, they have a shape similar to the curves of no electric fields. Again differences due to the source rate show up at high-energy region. Below 10 eV, all the distribution functions fall on the top of each other, and they begin to spread out as the energy increases above 10 eV. For high pressures or low source energy, more difference (between \( S_o = 0 \) and \( S_o \neq 0 \)) show up at high-energy end than that of low pressure and/or high source energy. Again at very high source rates \( S_o = 10^{22} \text{#/cm}^3\text{-sec} \) the distribution function changes appreicably.
In Figure 21, the comparison is made with a very high source rate \( S_0 = 10^{22} \text{cm}^3\text{-sec} \) and with one in the region of interest \( S_0 = 10^{16}, p = 760 \text{ torr} \). For very high source rate, the distribution function changes quite a lot; it still preserves the general shape, but at low-energy end, it is much smaller in amplitude. (However, as stressed earlier, the high source rate calculation is questionable due to the neglect of electron-electron interactions.)

The changes in these distribution functions due to different pressures are also shown in Figures 22 and 23. The observed trends are similar to those in the previous section for the electric field \( E_f = 0 \) case.

So far, the normalized distribution functions \( f(E) \) have been displayed. However, to stress the fact that the absolute magnitudes of the curves depends strongly on the source (shapes have received our interest to this point), Figure 24 shows the unnormalized electron energy distribution functions for He (with electric field or \( E_f \neq 0 \)) are plotted, with \( E_f/p = 10, p = 10 \text{ torr}, E_o = 1000 \text{ eV}, \) and three different source rates \( S_o = 10^{14}, S_o = 10^{16}, S_o = 10^{18} \). The magnitude (unnormalized) distribution is defined as

\[
N^-(E) = N^+ \cdot f(E)
\]

(66)

where \( N^+ \): number of positive ions

\( f(E) \): normalized distribution

The values of \( N^+ \) are \( 0.24 \times 10^{15}, 0.24 \times 10^{14}, \) and \( 0.24 \times 10^{13} \) corresponding to the three source rates \( S_0 = 10^{18}, S_0 = 10^{16} \).
Figure 17. The electron energy distribution for He with an electric field at $E_f/p = 10$, $p = 10$ torr, $E_o = 1000$ eV.
Figure 18. The electron energy distribution for He with an electric field at $E_f/p = 10$, $p = 760$ torr, $E_o = 1000$ eV.
Figure 19. The electron energy distribution for He with an electric field at $E_f/p = 10$, $p = 10$, $E_o = 70$ eV.
Figure 20. The electron energy distribution for He with an electric field at $E_f/p = 10$, $p = 760$ torr, $E_0 = 70$ eV.
Figure 21. The electron energy distribution function at $E_f/p = 10$, $p = 760$ torr for $S_0 = 10^{22}$ and $S_0 = 10^{16}$ ($\#$/cm$^3$-sec).
Figure 22. The electron energy distribution for He, with an electric field at $S_0 = 10^{16}$, $E_o = 1000$ ev.
Figure 23. The electron energy distributions for He, with an electric field at $S_o = 10^{16}$, $E_o = 70$ eV.
Figure 24. The unnormalized electron energy distribution for He at $E_f/p = 10$, $p = 10$ torr, $E_0 = 1000$ eV.
Figure 25. The effect of recombination on the electron energy distribution
a) with normal recombination cross section x 100
b) with normal recombination cross section x 1.
and $S_0 = 10^{14}$, respectively.

In order to show the effect of recombination, a normal case is compared in Figure 25 to a calculation where the absorption cross section was arbitrarily increased 100 times. It is observed that the major effect comes in at low-energy region. More and more electrons recombine as they thermalize and hence, the lower energy region of the distribution is depleted. Since recombination does not occur at high energies, this region is not affected.

The calculated major parameter values for the system with an electric field are summarized in Table 2. As expected on physical grounds the values of the average energies for this case are much higher than the corresponding values with no electric field (compare to Table 1). However, other general trends are the same, e.g. the value of the average energies are slightly higher for higher source rate and/or higher source energy, but are lower for higher pressures.

For comparison the average energy (7.7 eV) found by Smit for an electric field and $E_f/p = 10$ but with no source present is included in Table 2. It is in general higher than the present values. The reason is not entirely clear. At first thought the additional high-energy source electrons might be expected to raise the average energy. However, as found by T. Ganley (37), the presence of high-energy electrons to produce ionization allows the electrons in the Maxwellian region to decrease in energy, and as a result the overall average may actually decrease.
Table 2. Calculated Parameter Values for Plasma with Applied Electric Field ($E_f/p = 10$)

<table>
<thead>
<tr>
<th><em>P</em> (torr)</th>
<th>$E_0$ (eV)</th>
<th>$S_{o3}$ (#/cm$^2$-sec)</th>
<th>$E$ (eV)</th>
<th>$N^+$ (#)</th>
<th>Modified W-value range</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1000</td>
<td>$10^{18}$</td>
<td>8.1</td>
<td>$0.24 \times 10^{15}$</td>
<td>35 $\sim$ 44</td>
</tr>
<tr>
<td>10</td>
<td>1000</td>
<td>$10^{16}$</td>
<td>7.5</td>
<td>$0.21 \times 10^{14}$</td>
<td>34 $\sim$ 54</td>
</tr>
<tr>
<td>10</td>
<td>1000</td>
<td>$10^{14}$</td>
<td>7.3</td>
<td>$0.24 \times 10^{13}$</td>
<td>37.5 $\sim$ 60</td>
</tr>
<tr>
<td>760</td>
<td>1000</td>
<td>$10^{18}$</td>
<td>7.0</td>
<td>$0.23 \times 10^{17}$</td>
<td>37 $\sim$ 50</td>
</tr>
<tr>
<td>760</td>
<td>1000</td>
<td>$10^{16}$</td>
<td>6.6</td>
<td>$0.204 \times 10^{16}$</td>
<td>30 $\sim$ 51</td>
</tr>
<tr>
<td>760</td>
<td>1000</td>
<td>$10^{14}$</td>
<td>5.4</td>
<td>$0.24 \times 10^{15}$</td>
<td>34 $\sim$ 57</td>
</tr>
<tr>
<td>10</td>
<td>70</td>
<td>$10^{18}$</td>
<td>5.86</td>
<td>$0.21 \times 10^{15}$</td>
<td>31 $\sim$ 35</td>
</tr>
<tr>
<td>10</td>
<td>70</td>
<td>$10^{16}$</td>
<td>5.18</td>
<td>$0.21 \times 10^{14}$</td>
<td>34 $\sim$ 50</td>
</tr>
<tr>
<td>10</td>
<td>70</td>
<td>$10^{14}$</td>
<td>4.46</td>
<td>$0.213 \times 10^{13}$</td>
<td>31 $\sim$ 56</td>
</tr>
<tr>
<td>760</td>
<td>70</td>
<td>$10^{18}$</td>
<td>5.7</td>
<td>$0.284 \times 10^{16}$</td>
<td>35 $\sim$ 46</td>
</tr>
<tr>
<td>760</td>
<td>70</td>
<td>$10^{16}$</td>
<td>5.0</td>
<td>$0.26 \times 10^{15}$</td>
<td>39 $\sim$ 48</td>
</tr>
<tr>
<td>760</td>
<td>70</td>
<td>$10^{14}$</td>
<td>4.3</td>
<td>$0.24 \times 10^{14}$</td>
<td>32 $\sim$ 49</td>
</tr>
</tbody>
</table>

Smit's distribution

<table>
<thead>
<tr>
<th>E</th>
<th>N+</th>
<th>Modified W-value range</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>7.7</td>
<td>---</td>
</tr>
</tbody>
</table>

*P: pressure; $E = source energy; S: source rate; $E_f$: average energy; $N^+$: number of positive ions.*
The values of $N^4$ are roughly proportional to the square root of the source rate (recall Equation 33) and are independent of the pressure and source energy.

Strictly speaking, the definition of the $W$-value for the field case should include an allowance for energy gained by electrons in the electric field which then result in ionization. There is no way to include this energy transfer in the present case, so the values listed in Table 2 are based on the ion pair production divided by the energy input with the high-energy source electron alone (no field contribution). This is, then the same formulation (Eq. 65) as that of no field case, and we label the result as the "effective" or "modified" $W$-value. Again these values fluctuate considerably, so the ranges observed in the last 10 time cycles are given. As would be expected from the neglect of the field contribution, the $W$-values in this table are in general smaller than that of no field case. As in the earlier calculation, no general trend can be observed, as so far as the dependence of the parameters ($E_o$, $S_o$, $p$) are concerned, and this is consistent with the earlier observations and the literature which indicate that the $W$-values are fairly insensitive to these parameters.

We have concluded so far that, the distribution functions for non-field case ($E_f = 0$) are more or less assume a Maxwellian distribution in the low-energy region for source rates below $10^{18}$. However, when an electric field
is added, the distribution shifts to that found by Smit in the low-energy region. This is, in effect, the well-known Druyvesteyn distribution. It is significant that the addition of a source adds a tail to this distribution, but it does not drastically change the low-energy region. By intuition, it might be expected that Smit's distribution (with electric field or $E_f \neq 0$) would change back to a Maxwellian distribution as the electric field approaches zero. In order to display the transition region, the distribution functions for smaller electric field ($E_f/p = 1$) are plotted in Figure 26 together with the calculated distribution functions at $E_f/p = 10$ and $E_f = 0$. The high-energy tail of the distribution curve are not significantly affected by reducing the electric field $E_f$, but the distribution function in the low energy region starts to approach a Maxwellian distribution as expected.

D. Conclusion on Monte Carlo Results

Based on the Monte Carlo solutions we obtained so far, we can make the following comments.

1. For the normal range of source rate ($S \leq 10^{18}$), where the neglect of electron-electron interactions remains valid, the Maxwellian plus $1/E$ distribution is a good initial approximate distribution for the no electric field case. Likewise the Smit (or Druyvesteyn) plus $1/E$ distribution is a good approximate initial distribution to be used for the system with an electric field present.
The electron energy distribution for He at $p = 10$ torr, $S_o = 10^{16}$ for $E_f/p = 0$, $E_f/p = 1$ and $E_f/p = 10$.

Figure 26. The electron energy distribution for He at $p = 10$ torr, $S_o = 10^{16}$ for $E_f/p = 0$, $E_f/p = 1$ and $E_f/p = 10$. 

Energy in eV
2. The dependence of the distribution functions (with or without electric field) upon the pressure and source rate only shows up in the high-energy region, where the distribution functions have higher tails for higher source rate, and have smaller values for higher pressures.

3. The range of variations observed in the W-value calculation reflects statistical fluctuations which limit the calculational accuracy of the absolute numerical W-values. Still the results demonstrate that the W-value is more or less independent or insensitive to the changes of pressure, source rate and source energy.

4. Due to a tight budget, the primary emphasis was on increasing the computation efficiency and reducing the computation time to a possible minimum. Thus most of the calculations were carried out with 2000 to 5000 actual particles and for any particular set of parameter values, and the results were generally obtained within 10 minutes of (IBM 360/75) computer time. Compared with hours of computer time and 5 to 10 times more particles used by others for the similar calculations (e.g., see References 3, 4, and 9) the gain in computation efficiency seems to be significant. Of course, a complete comparison is not possible since all of these calculations achieved different accuracies in the final results.

5. The present results also show close agreement with the analytic solutions of Lo and Miley (32,33) on the
following points:

a. The shape of high-energy portion of the distribution curves are in reasonable agreement as shown in Figure 9.

b. The high-energy part of the distribution function is not changed by the presence of an electric field for field values $E_f / p \leq 10$.

c. With an electric field, the low-energy part of the distribution function agrees with Smit's result; and this part is not affected appreciably by source rates $< 10^{18}$ particles/cm$^3$-sec.
CHAPTER VI
CONCLUSIONS

A. Prelude

The application of Monte Carlo techniques to solve particle transport problems is of fundamental interest in many fields of physics and engineering; especially when analytic and/or numerical solutions of the basic governing equations are too complex to be practical. Yet direct application of Monte Carlo techniques to simulate individual interactions is sometimes prohibitively costly because of the large amount of computational time required. Thus techniques for improving the efficiency are required. Various existing variance reduction techniques can be used, but they are problem dependent. Thus techniques that are efficient for one type of problem might not be as effective for other types of problems. Thus such techniques must be judiciously selected and modified for the particular problem under consideration. One basic underlying principle which applies to any Monte Carlo calculation is: Apply as much known information as one can (given in analytic and/or numerical form) to reduce the uncertainties of the problem. Whenever possible Monte Carlo experiments should be checked and replaced by exact theoretical analysis to reduce uncertainties.

The Monte Carlo experimentalist has to exercise ingenuity in distorting and modifying problems in the pursuit
of variance-reduction techniques. Although Monte Carlo methods are general devices, still a great deal of work depends upon the individual's originality to create special methods or numerical schemes to suit their needs. Unlike the physical experimentalist, the mathematical experimentalist using Monte Carlo Methods has the advantage that his experimental material consists of mathematical objects which can be distorted, controlled, and modified more easily.

The greatest successes of the Monte Carlo method have arisen where the basic mathematical problem itself consists of the investigation of some random processes. However, there are exceptions involving deterministic problem-solution of boundary value problems and partial differential equations. The solutions of these problems are closely connected with the characteristics of certain random diffusion type processes (or can be converted to such type of processes), therefore these problems are reduced to the modeling of such processes. Thus the model and the special techniques developed in present research are quite general in scope, and it should be possible to apply them to many classes of Monte Carlo calculations with a certain amount of modification.

B. Summary of Present Work

In this research study, we have devised a general mathematical model for the particle transport or diffusion
type plasma problems which have inherent nonlinear properties due to recombination. Many special variance reduction techniques are incorporated into the system. The model can be used for calculating both time-dependent and steady-state nonlinear transport problems provided that suitable methods exist to calculate, predict, and correct the important parameters which dominate the nonlinearities. A wide range of nonlinear problems can be formulated and handled in this way. The principle idea, the piecewise linearized predict-correct model, is a general, simple, and efficient approach for solving nonlinear problems. In principle, most nonlinear problems can be cast in terms of the basic ideas and algorithms of present work. This represents an extension of earlier works of Musha and Itoh (8), Thomas (9), and Fleck Jr. (11). As for the amount of computation time and efficiency involved, the present study has made significant improvements. Some of the improvements are a natural consequence of the large storage space available in modern computer systems. Additional improvements come from a few simple, but effective, special techniques incorporated into the present model. As described in Chapter II, these include:

1) The initial distribution - straightforward particle tracing procedures would require following every simulated particle from its birth (from source) with initial energy $E_0$ at $t = t_0$ to the end of its life time history. This type of simulation would require hundreds
of cycle times to achieve a final steady state condition. Instead, we start with an approximately steady-state system by guessing an initial particle energy distribution. Then we simulate the system by tracing particles starting with energy corresponding to this input distribution. This is equivalent to skipping over the initial transient time cycles and greatly reduces the computation time and increases the calculation efficiency.

2) The application of the negatively-correlated variates (antithetic variates) technique greatly reduces statistical fluctuations with a minimum number of simulating particles (machine particles).

3) The introduction of a weighting scheme and an artificial game of chance (using weighted cross sections) contributed some reduction in calculation time and resolved some practical difficulties.

4) For the system with an applied electric field, the exact formulation for a parabolic flight path was used to advance the particles in larger steps than would have been possible using the small-step linear approximation employed by Musher and Itoh (8) and also by Thomas (9). This improves the efficiency appreciably.

Throughout the entire calculation, one basic balance equation (Eq. 30) played a very important role. It merely states that at steady state, the source is balanced by losses.
By means of this equation, we predict the nonlinear parameter values and provide checks for convergence of the solution.

The need for large amounts of storage space for Monte Carlo calculations no longer creates serious problems or drawbacks on modern large computer systems, such as was the case as recent as seven or ten years ago. In addition, techniques exist for further reducing the storage requirement. In the present study, we suggest a technique to eliminate major storage requirements, and use only a small amount of storage as a temporary buffer storage (described later).

In other types of numerical solutions, e.g., in finite difference solutions of the diffusion equation, the step size $\Delta t$ is restricted by stability requirements. However, in this present model, $\Delta t$ is only restricted by the degree and amount of nonlinearities present in the system. This restriction is far less stringent.

C. Practical Limitations and the Cures

It was necessary to solve several difficulties in the present simulation study in order to provide practical solutions. We shall describe them item by item.

1) The electron energy distribution is the quantity of major interest. The difference between the magnitudes of its maximum and minimum non-zero values is of the order $10^3$. This is due to high density in the low-energy region (thermal energy) and the relatively low density
in the high-energy region (near the source energy).
It is unrealistic to hope to produce reasonable numbers of particles in the low-density (high-energy) region and at the same time have $10^9$ times as many in the low-energy region. This problem is resolved by combining the following techniques:

a) Russian Roulette and splitting.
b) Weight factors which produce fractional particles.
c) Logarithmic energy intervals. The number of particles in each energy class interval $\Delta E_i$ is directly proportional to $f(E_i)\cdot \Delta E_i$ ($f(E_i)$: number density at $E_i$).

By using logarithmic energy intervals, we have small intervals at low energy region and large intervals at high energy so that the product $f(E_i)\cdot \Delta E_i$ assumes a practical magnitude.

2) The total number of simulating particles (machine particles) $N_T$ is limited by the storage requirement and the economy of the calculation (computing time is proportional to $N_T$). We have to set an upper limit ($N_{max}$) on the total number of simulating particles in any time cycle. During each time cycle of duration $\Delta t$, $N_s$ source particles are introduced into the system, and $N_2$ secondary particles are produced (by ionization). At the same time $N_{es}$ particles escape from the system, and $N_{Ab}$ particles are lost due to recombination.

There are two conditions to be satisfied at all times:
\[ \begin{align*}
\text{a)} & \quad N_c + N_z + N_s \leq N_T \leq N_{\text{max}} & \quad (67a) \\
\text{b)} & \quad N_z + N_s \approx N_{\text{es}} + N_{\text{AB}} & \quad (67b)
\end{align*} \]

where \( N_c \): the number of census particles (background particles).

Condition a) insures that total number of particles in the system is always under the limit (\( N_{\text{max}} \)). Condition b) indicates that a particle balance should be satisfied at steady state.

In the program, there are two equivalent problems to be considered. One is the balance of the machine particles [corresponding to condition (a)], and the other is the balance of simulated particles [weighted particles, corresponding to condition (b)]. Both conditions must be satisfied simultaneously. Condition (a) allows a certain amount of leeway since the number of particles that survive the integration cycle (time cycle) and hence require storage will be less than \( N_c + N_z + N_s \). Here \( N_s \) is chosen to be an input number, the weights of these particles are determined by the source rate. If condition (a) can not be met, \( N_s \) is taken to be

\[ N_s = N_{\text{max}} - N_c - N_z - JZ - 1 \tag{68} \]

where \( JZ \) is the number of spatial zones. It has been shown (12) that Equation 68 insures the stabilization of census population without overly reducing the number of source particles.
The proper selection of the energy dividing line (ED1) for Russian Roulette and splitting can insure the satisfaction of condition (b). Also as mentioned in earlier chapters, the balance factor (BF) is used to speed up convergence and force condition (b) to be satisfied sooner.

3) In any Monte Carlo calculation, finite samples are used to estimate or simulate the behavior of a large parent population. This is the main source of statistical fluctuations (variance or uncertainty). In the present simulating system, less than 5000 particles were used to simulate approximately $10^{14}$ physical particles. This creates the problem of matching the values of input and estimated output quantities. Due to the huge scaling factor ($10^{14}/5000$), a small amount of statistical fluctuation or estimation error in the estimated value will be amplified about $10^{11}$ times. To avoid this, we use the estimated (or equivalent) input quantities which carry the same order of magnitude of statistical fluctuation and weight.

4) There are several reasons that make the logarithmic energy scale the logical choice for the group intervals in the energy axis of phase space. These include:
   a) the wide energy span (0 to 1500 eV)
   b) the relatively high number density at low energies combined with a low rate and small amount of energy
exchange due to collision processes. In the high-
energy region, the situation is just the opposite.
c) to produce reasonable number of particles in each
energy interval [for the reason mentioned in the first
section (II-B-1)]. The size and number of intervals
are mainly dictated by the range of data (energy
range in this case) and number of observations
available. Excessive fragmentation of the data
may produce many intervals with few occupants. On
the other hand, insufficient divisions may obscure
important dispersions. Thus an optimum number of
intervals will be a compromise between these extremes.

D. Future Extensions

The Monte Carlo methods are applicable to the most
widely diverse branches of computational mathematics includ-
ing particle physics (neutron transport for instance),
operations research problems such as the investigation of
servicing processes, modelling the processes of information
transmission in communication theory, the evaluation of
definite integrals, the solution of partial differential
equations (e.g. boundary-value problems); the solution
of systems of linear equations - inversion of matrices, and
many other applications. Problems handled by such methods
are in general of two types, the probabilistic and the
deterministic, according to whether or not they are directly
concerned with the behavior and outcome of random processes.
Their basis lies in simulating statistical experiments by means of computational techniques. Based on the underlying principles described earlier, various variance reduction techniques can be applied. The present work serves as an example and guide for using the important new techniques such as correlated sampling.

1) The code developed can readily be used to calculate electron energy distribution functions for other gaseous media by merely changing the cross sections input. The model can also be modified to include the electron-electron and electron-ion interactions although this adds more nonlinearities into the system. This would allow calculation for higher source rates and cases where the fractional ionization is large.

2) As for the complex transport problems encountered in areas such as plasmas physics and astrophysics, the model and techniques can readily be extended to the problems with more complex geometry and/or boundary conditions, with inhomogeneous media and multi-dimensions. Also many kinds of nonlinear features can be studied in a realistic fashion.

3) More elaborate error checking and correction procedure (based on the known physical phenomena or theoretical formulations) such as the correction matrix used by Berger (9) should be developed to treat the combined effect of statistical fluctuations and other types of
uncertainties. This is essential for generating good results for Monte Carlo calculation.

4) More elaborate algorithms or techniques for playing the artificial games of chance by means of weighted cross sections would be a good direction for future research. This must be based on the investigator's intuition and experience, but still it is a promising approach for increasing the efficiency.

5) The present model and techniques can be also implemented on smaller computer systems such as PDP-11 or other similar systems where large amount of fast storage space are not available. The following techniques could be used to process the calculation without storing all the particle data (namely \( E_i, u_i, z_i, Wt_i, \phi_i \)). At the beginning of each time cycle, we generate a small number (some reasonable amount, \( \approx 100 \) for instance) of particles. The corresponding particle data can be generated from the initial input distribution or from last time cycle. After these particles have been processed, only the accumulated information is stored. A new group of particles come into the system by generating a new set of data, based on the same distributions, which are stored in the same temporary storage space. The previous particle data is destroyed after they have been processed. The same process is repeated until all the necessary prespecified number of particles have been
processed, and the new distributions are obtained. This eliminates the necessity of storing all the particle data at each time cycle, and this eliminates the large arrays that demand the major amounts of storage space. Due to the random nature of the processing algorithms, and the Markovian behavior of the time cycle process (these time cycle processes are a kind of Markov process), this storage saving algorithm can accomplish the desired results.

E. Concluding Remarks

In developing any Monte Carlo models and performing calculations, as mentioned in Chapter V the systematic errors and outright blunders must be detected and removed by testing and comparing with known results.

The various variance reduction techniques are used to increase the relative computation efficiency, reduce inherent random errors or statistical fluctuations. For better calculational results, more samples are favorable even though not economic. In any Monte Carlo calculation there exists a minimum number of samples below which the results or statistics are no longer valid and dependable regardless of the variance reduction schemes employed. In the other words, variance reduction techniques help to reduce statistical fluctuations, but they can not be used to remove all the fluctuations caused by small samples.
The minimum sample size is, of course, problem dependent. There are no definite rules or formula for it, thus a cut and try method is necessary until the calculated variance is acceptable.

Monte Carlo calculations are perhaps best in the preliminary stages where they help to give a general idea of the situation and give hints or feelings for trends. If more accurate results are necessary, Monte Carlo calculations may not always be practical good method. In any case, the Monte Carlo method is an important tool in the inferential analysis of the mathematical-physics problems, but it should be viewed as supplementing, rather than replacing, analytic and other numerical methods.
LIST OF REFERENCES


33. R. H. Lo, Department of Nuclear Engineering, University of Illinois, Private Communication, April, 1972.


FORTRAN IV G LEVEL 16

0033 CJU=0.0
0034 E1=8.846
0035 FP=EDP=PDPO
0036 EIS=24.4
0037 C1=2.0 C=0.0005949/4.000873
0038 ETX=8072(2.0+1.0202320.1018+1.02+0)
0039 VTE=1.0/ETV=ETV
0040 CALL UNDER('APPP)
0041 PDZ=PDP=PDPO
0042 NQO=(1.0-0.0)/ETV
0043 CMUO=1.0/ETV
0044 CMUO=CMUO
0045 IF (TDPO.IEQ.0.01) CMUO=CMUO.0.01
0046 PDZO=CMUO/ETV
0047 ELS=1.0
0048 NUP=0
0049 NDN=0
0050 NNDN=0
0051 JND=0
0052 JRN=0
0053 NAF=0
0054 NFIELD=0
0055 MDZN=0
0056 MIZN=0
0057 NIH=0
0058 LK=0
0059 LINV=99999
0060 MN=0
0061 MNE=0
0062 MEXE=0
0063 ETAS=0.0
0064 ETSP=0.0
0065 MTR5=0
0066 HM=0
0067 HSTF=0
0068 NCOL=0
0069 WNCA=0.0
0070 WNLZ=0.0
0071 WNZC=0.0
0072 WNAS=0.0
0073 WNAEs=0.0
0074 WNSES=0.0
0075 CLOSS=0.0
0076 CSFZ=0.0
0077 CME=0.0
0078 CSFL=0.0
0079 CETAS=0.0
0080 CETSP=0.0
0081 CETSS=0.0
0082 JE=48
0083 IE=36
0084 DEP=3.5E11+PDPO
AC85 DETZ=0/22
FORTRAN IV G LEVEL 18

```
0060  DIV=0.69612
0067  CONST=1.6/LOG(2.01)
0068  ED2=0.66953
0069  ED3=19.9
0070  PH=ED2*AHY
0071  AHY=2.0*PI/360
0072  AHT=20.0*PI/360
0073  CALASC=PI360/20.0
0074  SP=1.0
0075  AHT= AHT/10.0
0076  DE2=0.013
0077  DE2=0.038
0078  DE3=0.0
0079  DI1=1.0/0.014
0080  DI2=1.0/DE2
0081  DI3=1.0/DE3
0082  ANU=SORT(12.0*2.0)*AKT*ED2
0083  DD_11=1.0679
0084  EBL=(1.0-DE2)-4.0
0085  EBL=EBL-0.1608
0086  EBL=EBL-0.20*DE3
0087  EBL=EBL-0.1405
0088  EBL=EBL-0.20*DE3
0089  EBL=EBL-0.20*DE3
0090  EBL=EBL-0.20*DE3
0091  EBL=EBL-0.20*DE3
0092  EBL=EBL-0.20*DE3
0093  EBL=EBL-0.20*DE3
0094  EBL=EBL-0.20*DE3
0095  EBL=EBL-0.20*DE3
0096  EBL=EBL-0.20*DE3
0097  EBL=EBL-0.20*DE3
0098  EBL=EBL-0.20*DE3
0099  EBL=EBL-0.20*DE3
0100  EBL=EBL-0.20*DE3
0101  EBL=EBL-0.20*DE3
0102  EBL=EBL-0.20*DE3
0103  EBL=EBL-0.20*DE3
0104  EBL=EBL-0.20*DE3
0105  EBL=EBL-0.20*DE3
0106  EBL=EBL-0.20*DE3
0107  EBL=EBL-0.20*DE3
0108  EBL=EBL-0.20*DE3
0109  EBL=EBL-0.20*DE3
0110  DD_12=1.0*DE3
0111  EBL=7.005+11.30*DE3
0112  EBL=EBL-0.1405
0113  EBL=EBL-0.20*DE3
0114  EBL=EBL-0.20*DE3
0115  EBL=EBL-0.20*DE3
0116  DD_10=1.0
0117  EBL=1.0
0118  EBL=1.0
0119  EBL=1.0
0120  EBL=1.0
0121  EBL=1.0
0122  IF IMPRINT(1) GO TO 2
0123  READ(101) EAVG, DANG, THF, NTW, HT, NTP, NION, DEEG, NSW, SHW, E U
0124  READ(101) HT, SHID, NSW
0125  NS=NSW
0126  WHT=1.0
0127  IN=IMB, 10
0128  NT=NT + 1
0129  NT=NT + 1
0130  IF IMPRINT(1) GO TO 2
0131  READ(101) EAVG, DANG, THF, NTW, HT, NTP, NION, DEEG, NSW, SHW, E U
0132  READ(101) HT, SHID, NSW
0133  NS=NSW
0134  WHT=1.0
0135  REWIND 9
0136  NT=NT + 1
0137  CONTINUE
0138  IF IMPRINT(1) GO TO 122
```
FORTRAN IV, G LEVEL 18	 MAIN	 DATE = 72128	 05/31/70

0139 NION = SORT(SO*10/ATIG)
0140 ANGAM=NION*0.33333/DEN
0141 CALL SOURCE(INTP, EAVG, NTP, PM, P1D)
0142 (ALPHA,LEAVE)
0143 DAVG=AL
0144 DAVE=DAYGNND/SORT(EAVG)
0145 SF2=1.0
0146 IF (NION.NLE.0) SF2=MNT/NION
0147 SN=SO+DFZ/ETV=SORT(EAVG)
0148 NS=NSM*0.5
0149 IF (SN.GT.ANSW) NS=SN+0.5
0150 NS=NSM*2
0151 SW=NSM/SN
0152 SF=SF2
0153 SF2=SF2
0154 SF=SF2
0155 CALL SOURCE(IN, NTP)
0156 THE=0.0
0157 ANIOS=NION
0158 WGT=1.0
0159 *122 CONTINUE

C GENERATE SOURCE PARTICLES TO INITIALIZE THE PROGRAM
0160 ANCE=0.00000/DEN
0161 WRITE(6710)
0162 70 FORMAT(11, *PROGRAM INITIALIZATION AND THE INITIAL PARAMETERS*, /)
0163 WRITE(16,711) ALTG, NION, NTP, AL.EAVG
0164 71 FORMAT(11, *E14.6, 3X, NION =E14.6, 3X, NTP =E11.3, 3X, MNT =E11.3, 3X, TNP =E11.3, 3X, 1)
0165 CITEN(EO) = E14.6, 3X, E14.6, 3X, E14.6, 3X, E14.6, 3X, E14.6, 3X, E14.6, 3X, E14.6, 3X, E14.6, 3X, E14.6, 3X, E14.6, 3X)
0167 WRITE(4721)
0168 WRITE(4721)
0169 RIP=ANS210(1)
0169 LI=1

C WITHIN THE OUTER I LOOP THE I TH PARTICLE HISTORY
C OBTAINED AN INITIAL ENTRYING HERE
0170 7777 IF ((E11,LGE.0.0) OR (E11,G.EQ.1.5)) GO TO 9000
0171 TCE=MOD(E10, ETV)
0172 7778 IF (E1(E11,GT.0.0) AND, (1, GT. NTP)) TCE= DIZIN(D11)
0173 E11=M11+1211
0174 LG2=0
0175 ETAG=1.0
0176 IF (E11.GT.ED1) ETAG=1.0
0177 SG=SORIE(11)
0178 OVA=E11*ETV=M11*0.5
0179 VYS=SQR(ETV**U11) + DVA
0180 VS1=E11*ETV**U11 + U11*ETV
0181 TCE=SORIE(VA11,+5**.5) +ogle
0182 IF LG2=0
0183 900 IF (E11.LGE.0.0) GO TO 9000
0184 IF (NFLG.GEQ.0) GO TO 809
0185 SG=SORIE(11)
0186 OVA=E11*ETV=M11*0.5
0167 IF (ENEM.EQ.                           0) GO TO 9009
0168 809 DS=VS+TGEN
0169 ENEW = E11 + EFPOE
0170 IF (ENEM.LE.0.01) GO TO 9008
0171 SGEP=SGRTEN
0172 UNEW = SGEPUG(SGEP*DVAM2./(SGEP*ETV))
0173 ALL=ALGT(ETT)
0174 AL=ALD(GEN)
0175 AL1=1.0/AL + 1.0/AL2
0176 AL2=2.0/AL
0177 RI+RANSZ(DI)
0178 AL=ARSIL0G(RI)
0179 DCOL=ALXALD
0180 IF (ENFLG.EQ.11) GO TO 901
0181 IF((1000.I.GT.0.0) WRITE(6,75) DCOL,OCEN,ETT,UTT,WT(1)
0182 DCOL=ARSIL0G(RI)-ALXALD
0183 DTOT=DTOT+DCOL*DCOL
0184 ETOT=ETOT+ETI
0185 IF (MDP+LY.E2) GO TO 901
0186 TCEN-TCEN
0187 DCE=N DCE
0188 E10=E1I
0189 U10=U1I*1USN
0190 Z10=E11
0191 ENDE=ENEW
0192 U10=UNEW
0193 DI0=02
0194 901 NFLG=1
0195 902 IF (DCEN.LT.DCOL) GO TO 905
0196 TCEN=TCEN - DCOL*DCEC/DCEC
0197 DCE=DCE+DCOL/DCEC
0198 211=E1I + EFZC
0199 211=21I + DZC
0200 DCEC=DCEN - DCOL
0201 NCOL=NCOL+1
0202 IF (MOD(NCOL,MC1).EQ.0) R1=RANSZ(0)
0203 IF (MOD(NCOL,NC1).EQ.0) R1=RANZ(0)
0204 CALL COLFAD11, TCEN, NTP, R1, NUST, 69008, 69045
0205 IF((1000.L.GE.0.0).AND.((LF2.EQ.11)) GO TO 905
0206 GO TO 900
0207 905 E11=ENEW
0208 U1I=UNEW
0209 Z11=Z1I + DI
0210 IF (1111.LE.0.0).OR.(1111.GT.0.0) GO TO 9008
0211 J=INT(1111/4ZIV)+1
0212 IF (J.GT.Z2) J=J2
0213 K=INDX+1(E11)
0214 IF (K.GT.J1) K=J1
0215 IF (K.GT.J1) J=K
0216 IF ((E11.LE.0.0) .OR. (E11.GT.0.0) ) GO TO 9008
0217 ETAG=E11
0218 IF (ETAG*ETAG2.EQ.1.0) ETAG=ETAG2=1.0
0219 IF (ETAG*ETAG2.GE.1.0) GO TO 904
0220 IF (ETAG.EQ.1.0) ETAG2=1.0
FORTRAN IV G LEVEL 18

MAIN

DATE = 72128  05/31/87

0240 NUP=NUP+1
0241 ELSE(NUP)=E(I)
0242 M(I)=M(I)/K3
0243 MSE(NUP)=M(I)
0244 GO TO 904
0245 903 IF(LFG2.EQ.1) OR(NOPM.LT.2) NOPM=NPM
0246 IF(LFG2.EQ.1) OR(NOPM.LT.2) R=NIN008=
0247 IF(NOPM.NE.1) GO TO 9045
0248 IF (MOD(NONK,3)).NE.0 GO TO 9045
0249 WT(I)=WT(I)+K3
0250 9045 NEIK(JI)=NEIK(JI)+WT(I)
0251 KES(KI,J) = KES(KI,J)+WT(I)
0252 9045 IF(LFG2.EQ.1) OR(NOPM.LT.2) GO TO 9009
0253 LFG2=L
0254 E(I)=E(I)
0255 L(I)=L(I)
0256 (I) = I (I)
0257 TGEN=TGEN +2
0258 DCEN=DCEN +2
0259 DCOL=DCOL +2
0260 ENEW=ENED
0261 UNEW=UNED
0262 DI=DI +2
0263 IF(JUSN.LT.0) GO TO 900
0264 GO TO 902
0265 9008 NES=NES+1
0266 IF (E(I).NE.0.0) WT(I)=WT(I) +K3
0267 IF (WT(I).LT.1) WT(I)=1.0
0268 WNES=WNES +WT(I)
0269 IF (NES NE 1) GO TO 9009
0270 IF (E(I).GT.0.0) ETNP=ETNP +E(I)
0271 9009 K = I
0272 LF2=0
0273 IF ((I).LT.0) GO TO 7777
0274 IF (I.EQ.1) AND (NIZN.GT.0) NOZN=NOZN +1
0275 IF (NIZN.GT.0) NIZN=1
0276 IF (MIN.GT.0) AND (NIN.EQ.0) GO TO 7778
0277 NIZN=MIZN +0.5
0278 NWS=NWS +0.5
0279 NAS=NAS +0.5
0280 WNAS=WNAS +KELC
0281 WSE=WNSE +KELC
0282 MSE=WNSE +0.5
0283 GO 191 M=1,JE
0284 GO 191 N=1,J2
0285 ELEM=ELEM +1
0286 191 NEM=NEM +1
0287 IF (E(I).EQ.1) OR(NUP.EQ.0) GO TO 23
0288 N3=N3+2
0289 IF (LFG2.EQ.1) M=1
0290 N3=N3+2
0291 IF (LFG2.EQ.1) M=1
0292 IF (M3.GE.1) M=3
```fortran
FORTRAN IV G LEVEL 1B

0293 DO 23 N=1,NUP
0294 EK=K0*K0*(ESE(N))
0295 EE(K)=EE(K)*((K3-1)*ESE(N)+WSE(H))
0296 NE(K)=NE(K)*(K3-1)*WSE(H)
0297 22 IF (NK3.GE.J11) IZ=JE
0298 DD 21 L=L1,JE
0299 NEZ(L)=NEZ(L)+M*NSE(N)
0300 21 EZ(L)=EZ(L)+ESE(N)*M*NSE(N)
0301 NK3=NK3-1,JZ
0302 IF (NK3.LT.J1) IZ=JE
0303 M=1
0304 IF (NK3+GT+Q) GO TO 22
0305 23 CONTINUE
0306 24 CONTINUE
0307 DD 192 N=1,JE
0308 DD 1915 N=1,JE
0309 NEZ(M)=NEZ(M)+NZE(N,M)
0310 1915 EZ(M)=EZ(M)+NZE(N,M)
0311 192 CONTINUE
0312 NTPH=NTP+M*NES+MAS
0313 NTPW=0
0314 ETOL=0.0
0315 DO 14 I=1,NJE
0316 ETOL=ETOL+EE(I)
0317 14 NTPW=NTP+M*NES
0318 ETOL=ETOL
0319 NTP*=NTPN
0320 DAC=DAV+ND
0321 DAVG=DAVG+1000.1*NTPV
0322 DAVS=DAVS+DAVG
0323 DTAVS=DTAVS+DAVS
0324 DAVG=DAVG+1000.1*NTPV
0325 DAVS=DAVS+DAVG
0326 NTP=RTSP-NTP+NTPH*.D0.0
0327 NTP=NTP+NTPH+DAYS
0328 NTP=NTPH
0329 EMAG=ETOL+ANTPO
0330 NSD=NS
0331 NSD=NSD+12*0.5
0332 NTP=NTP12Z
0333 NTTP=NTPH
0334 DAC=DATOP+3/NTP
0335 EAVG=ETOT/NTP
0336 DBR=0000/ETV
0337 TME=TME+TPE
0338 TM=E-TME*TPE
0339 C IF SOURCE PARTICLES SOINSIC NES THEN ELIMINATE AS MANY RESIDUE PARTICLES AT TH
0339 NTP=NTP+K*NES+NASW
0340 NTP=NTP+W
0341 OREN=A000A001
0342 DREV=DAVG/WNS/SORT( EAVG)
0343 RE=0.0
0344 IF ((MNES+WNAS+GT.O.O) .EQ. (WZ2N+SN)/(WNES+WNAS))
```
C INTRODUCE THE SOURCE PARTICLES FOR THE NEXT CYCLE
C EVERY FIFTH CYCLE RENORMALIZE NION AND OUTPUT QUANTITIES ARE PRINTED
C RENORMALIZATION PROCESS OF NION
0405  ANSTP= 1.0/NTS5
0406  ANTP5=DSAVS/NTS5
0407  N5TP=N5TP
0408  DO 16 JZ
0409  CNREZIMI= CNREZ(M) + NREZ(M) + DAVS
0410  16  CEZIMI= CEZ(M) + EZ(M) + DAVS
0411  DO 12 JZ
0412  PNAEZIMI= NREZ(M)*ANTPO
0413  10 EZIMI= EZ(M) + NREZ(M)
0414  SUMN=0.0
0415  DO 15 M
0416  SUMN= SUMN + NE(M)
0417  193 M
0418  TNEIMI= TNE(M) + NE(M)/EDEIM
0419  1755 M
0420  EAV3=0.0
0421  DO 20 M
0422  EAV3= EAV3 + NE(M)/EDEIM + EAV3
0423  1755 M
0424  CNION=CJOINP*2.0*AKY/SQRT(AKTOP)
0425  ANION=0.0
0426  DO 20 M
0427  ANIO2= ANIO2 + FM1LEGIMI + NLEIMI + EDEIM
0428  ANION= ANION + FM1LEGIMI + NLEIMI + EDEIM
0429  20 CONTINUE
0430  CNION=ANION+CNION
0431  DNO2=ANIO2*CNION
0432  CJOIN=CJOIN
0433  IF (ALOS,LT,0.0) CJD=0.0
0434  DNO2=ABS(CJOIN)+DNO2
0435  TNO2=ABS(CJOIN)/DNO2
0436  TNO2=ABS(CJOIN)/DNO2
0437  TNO2=SQRT(TNO2)
0438  TNO2=SQRT(TNO2)
0439  CNIO2=CNIO2 + TNO2*DAVS
0440  ANIODS-CNIO2*D130
0441  W=1.0
0442  IF (JAMES+ANAS,NE,0.) W=(ANION+AM5G0)/(JAMES+ANAS)
0443  IF (MODINT,NE,0.) GO TO 1790
0444  NION=ANIODS
0445  IF (W.F,GT,0.05),AND.(W.F,LT,2.0)
0446  DAVT=DAVT+ND
0447  1790 CONTINUE
FORTRAN IV G LEVEL 18

MAIN DATE • T2126 05/31%02

0446 ALPHA=WN11N/MTBM=DAVY)
0449 ANAEH=NO=33333/DEN
0450 1785 FORMAT(1D8,16M14,3X,16M13,4,3X,16M13,4,3X,
0451 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0452 WRITE(4,1785) ANION, DNO2, TNIO2, ALPHA, ALOSS
0453 171 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0454 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0455 WRITE(4,171) CD, POPD, DTE, DEG, NPOP
0456 180 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0457 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0458 WRITE(4,180) MT, TEM, NS, NW
0459 181 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0460 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0461 WRITE(4,181) FLX1M, FLX2M, FLX3M, FLX4M
0462 182 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0463 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0464 WRITE(4,182) X, Y, Z, W, M, N
0465 183 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0466 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0467 WRITE(4,183) X, Y, Z, W, M, N
0468 184 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0469 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0470 WRITE(4,184) X, Y, Z, W, M, N
0471 185 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0472 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0473 WRITE(4,185) X, Y, Z, W, M, N
0474 186 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0475 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0476 WRITE(4,186) X, Y, Z, W, M, N
0477 187 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0478 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0479 WRITE(4,187) X, Y, Z, W, M, N
0480 188 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0481 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0482 WRITE(4,188) X, Y, Z, W, M, N
0483 189 FORMAT(1D8,3X,16M14,3X,16M13,4,3X,16M13,4,3X,
0484 X,16M14,3X,16M13,4,3X,16M13,4,3X)
0485 WRITE(4,189) X, Y, Z, W, M, N

DO 281 M = 1, JF
0460 CSMH=CSHM + CMH=CMH+EMH=EMH/HEH(M)
0461 IF (HEH(M) .EQ. 0) DEAG=0.0001-EEH(M)-HEH(M)
0462 IF (HEH(M) .EQ. 0) DEAG=0.0001-EEH(M)-HEH(M)
0463 IF (HEH(M) .EQ. 0) DEAG=0.0001-EEH(M)-HEH(M)
0464 IF (HEH(M) .EQ. 0) DEAG=0.0001-EEH(M)-HEH(M)
0465 IF (HEH(M) .EQ. 0) DEAG=0.0001-EEH(M)-HEH(M)
0466 FNOV=CMH=FNEH(M)
0467 XPE=CMH=EMH=MPE
0468 FLX1=CMH=EMH=MPE
0469 FLX2=CMH=EMH=MPE
0470 IF (HEH(M) .EQ. 0) GO TO 281
0471 IF (HEH(M) .EQ. 0) GO TO 281
0472 IF (HEH(M) .EQ. 0) GO TO 281
0473 IF (HEH(M) .EQ. 0) GO TO 281
0474 IF (HEH(M) .EQ. 0) GO TO 281
0475 IF (HEH(M) .EQ. 0) GO TO 281
0476 IF (HEH(M) .EQ. 0) GO TO 281
0477 IF (HEH(M) .EQ. 0) GO TO 281
0478 IF (HEH(M) .EQ. 0) GO TO 281
0479 IF (HEH(M) .EQ. 0) GO TO 281
0480 IF (HEH(M) .EQ. 0) GO TO 281
0481 IF (HEH(M) .EQ. 0) GO TO 281
0482 IF (HEH(M) .EQ. 0) GO TO 281
0483 IF (HEH(M) .EQ. 0) GO TO 281
0484 IF (HEH(M) .EQ. 0) GO TO 281

XMI=EMH(M)
0474 XMI=EMH(M)
0475 XMI=EMH(M)
0476 XMI=EMH(M)
0477 XMI=EMH(M)
0478 XMI=EMH(M)
0479 XMI=EMH(M)
0480 XMI=EMH(M)
0481 XMI=EMH(M)
0482 XMI=EMH(M)
0483 XMI=EMH(M)
0484 XMI=EMH(M)
FORTRAN IV G LEVEL

135

FORTRAN IV G LEVEL

135

FORTRAN IV G LEVEL

135

FORTRAN IV G LEVEL

135

FORTRAN IV G LEVEL

135

FORTRAN IV G LEVEL

135

FORTRAN IV G LEVEL

135

FORTRAN IV G LEVEL

135
**FORTRAN IV G LEVEL 16**

**MAIN**

**DATE = 72124**

**05/31/00**

```
0509  E N D F I L E I O
0510  R E W I N D I O
0511  4 C O N T I N U E
0512  I F ( M O D I N T . N C U M I . N E . 0 ) G O T O 6 6 6 6
0513  N S T P = 0
0514  N T R N = 0
0515  D T A V S = 0.0
0516  N A S T = 0
0517  N I N T = 0
0518  N W L E = 0.0
0519  C N S I = 0.0
0520  C N S S = 0.0
0521  C M E T = 0.0
0522  C N E X C = 0.0
0523  C E S A V G = 0.0
0524  C E A S A G = 0.0
0525  C S F 2 = 0.0
0526  C S F 3 = 0.0
0527  C M 1 0 2 = 0.0
0528  C S F 1 = 0.0
0529  D O 5 4  N = 1 , J Z
0530  C N R E I M ( M ) = 0 . 0
0531  9 4  C E Z I M ( M ) = 0 . 0
0532  D O 5 5  M = 1 , J E
0533  T H E I M ( M ) = 0 . 0
0534  5 5  C N R E I M ( M ) = 0 . 0
0535  C N E S = 0 . 0
0536  C N A S = 0 . 0
0537  D T A V S = 0 . 0
0538  C T E T M = 0 . 0
0539  C E T A S = 0 . 0
0540  I F ( M O D I N T . N C U M I . N E . 0 ) G O T O 6 6 6 6
0541  6 6 6 6  C O N T I N U E
0542  N T = N T + 1
0543  D O 5 6  M = 1 , 9 0 0
0544  D I Z I M ( M ) = 0 . 0
0545  I E S I M ( M ) = 0
0546  I R E X I M ( M ) = 0
0547  5 6  I A S I M ( M ) = 0
0548  D O 5 1  N = 1 , 9 0 0
0549  W S E I M ( M ) = 0 . 0
0550  5 1  E S E I M ( M ) = 0 . 0
0551  1  1  C. R E I N I T I A L I Z E T H E O U T P U T Q U A N T I T I E S
0552  D O 5 2  M = 1 , J E
0553  5 2  E S I M ( M ) = 0 . 0
0554  5 2  N E I M ( M ) = 0 . 0
0555  D O 5 3  M = 1 , J E
0556  N A E R Z ( M ) = 0 . 0
0557  E Z I M ( M ) = 0 . 0
0558  D O 5 3  N = 1 , J E
0559  E K Z I M ( M ) = 0 . 0
0560  5 3  N E Z I M ( M ) = 0 . 0
```
<table>
<thead>
<tr>
<th>FORTRAN IV LEVEL 16</th>
<th>MAIN</th>
<th>DATE = 72126</th>
<th>0573/02</th>
</tr>
</thead>
<tbody>
<tr>
<td>0561</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0562</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0563</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0564</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0565</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0566</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0567</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0568</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0569</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0570</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0571</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0572</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0573</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0574</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0575</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0576</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0577</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0578</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0579</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0580</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0581</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0582</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0583</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0584</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0585</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0586</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0587</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0588</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0589</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0590</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0591</td>
<td>NIN=0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FUNCTION INDEXE(E)
COMMON /AREA/, CONS,DEL2, ELT, IE, D11, D12, D13, DEL1
IF (E.LT.0.08) GO TO 11
IF (E.GT.0.159 .1539) GO TO 13
1 INDEXE = 5* (ALOG(2)*CONST + 0.01*D12)
RETURN
11 INDEXE = 1 + END
11 RETURN
13 INDEXE = 36 + (ALOG(T)/CONS - 0.0359*D13)
0010 RETURN
0011 END
FORTRAN IV source code:

```fortran
0040 IF (NI(I), LT, 1) GO TO 300
0050 JI = NI(I) + IE - 1
0051 DD 301 J = IE, JI, 2
0052 RI = RANS2(I)
0053 RZ = 1.0 - RI
0054 E(J) = ER0(I) + ED2(I)*RI
0055 E(J-1) = FRO(I) + FNE(I)*R2
0056 WT(J) = WJ
0057 WT(J+1) = WJ+1
0058 EAG = EAG + E(J) + E(J+1)
0059 IF (MU0(J-1, J), EQ, 0.1, AND, (J, GE, J1)) EAG = EAG + E(J+1)
0060 CONTINUE
0061 IE = IE + N(I, 1)
0062 CONTINUE
0063 NSO = NSO
0064 EAVG = EAG/NSO
0065 I2 = NSO
0066 ETW = 0.9
0067 DD 400 IE = IE + NSO
0068 IF (E(I, 1, LE, ED1)) WT(I) = K1 * WT(I)
0069 IF (E(I, 1, GT, ED1)) WT(I) = WNZ * WT(I)
0070 IF (E(I, 1, LE, G.C)) E(J) = E(J-1)
0071 FTW = ETW + E(I, 1, WT I)
0072 CONTINUE
0073 400 ETAVG = ETW/NTNZ
0074 WRITE(16, G07) EAVG, ETAVG
0075 G07 = 0.0, 25
0076 I = 1
0077 101 RZ = RANS2(I)
0078 RU = RANS2(I)
0079 U(I) = 0.0 - RZ
0080 U(I) = 2.0 - RU
0081 I = I + 1
0082 102 Z(I) = 0.0 - RZ
0083 U(I) = 2.0 - RU
0084 I = I + 1
0085 103 Z(I) = 0.0 - RZ
0086 U(I) = 2.0 - RU
0087 I = I + 1
0088 104 Z(I) = 0.0 - RZ
0089 U(I) = 1.0 - RZ
0090 I = I + 1
0091 IF (Z(2) - 1, I, GE, 0) GO TO 101
0092 IF (Z(2) - L, I, 1) GO TO 105
0093 RZ = RANS2(I)
0094 RU = RANS2(I)
0095 IF (Z(2) - 1, EQ, 2) GO TO 102
0096 IF (Z(2) - L, EQ, 1) GO TO 103
0097 IF (Z(2) - L, EQ, 0) GO TO 104
0098 105 RETURN
0099 END
```
SUBROUTINE SOURCE(NS,NTP)
COMMON D, E(S100), I(S100), U(S100), IAS(S000), IRR(S000), NRR, IES(S000)
X,KT(S100), SWF
COMMON /AREA2/ C1, MUO, E1, E15, SF, AAME, PDPO, PI, E0, NION, JE, JZ, NASH1,
INSEW, AKT, AIKT, AM1, AMCI, ED1, DMOUO, PD26, K1, K2, K4, WNN2, ED2, E03
REAL MUO, NION
C SOURCE PARTICLE PARAMETERS GENERATING PROEDURE
C II: THE INITIAL INDEX VALUE FOR THE SOURCE PARTICLE
105 RETURN
END

DO 100 K=1, NS, 2
E(I)=EO
RU=RN3Z(I)
Z(I)=D*RZ
U(I)=RU+RU-L, C
WT(I)=MN2/SWF
IF ((NS-K).LE.3) GO TO 100
E(I+1)=EC
I(I+1)=I(I)+1
WT(I+1)=MN2/SWF
1=I+2
100 CONTINUE
NTP=NTP+NS
105 RETURN
END
C WHENEVER THERE ARE MORE HOLES THAN SOURCE PARTICLES, FILL IN THE EXTRA HOLES

SUBROUTINE FILHOL(NTP, NES, NAS)

COMMON D, E(5100), Z(5100), UI(5100), IA(5100), INR(950), NNR(951), EIS(6001)

K=NTP

IF (NES.LE.0) GO TO 105

DO 100 I=1, NES

J=IES(I)

E(J)=E(K)

Z(J)=Z(K)

UI(J)=UI(K)

WT(J)=WT(K)

100 K=K-1

NTP=NTP-NES

105 IF (NAS.LE.0) RETURN

DO 300 I=1, NAS

J=IAS(I)

E(J)=E(K)

Z(J)=Z(K)

UI(J)=UI(K)

WT(J)=WT(K)

300 K=K-1

NTP=NTP-NAS

200 RETURN

END
SUBROUTINE RRPROC(NTP,K3)
COMMON D, E(5100), Z(3100), U(5100), IAS(500), IRR(850), NKR, IES(600)
X, W1(5100), SMF
COMMON /A6/ ESE(50), NUP, NDNR, NDN, WSE(50).
NR=NDNR
K=NTP
IF (K3.LE.1) RETURN
100 CONTINUE
I=IRR(NR)
E(I)=E(K)
U(I)=U(K)
Z(I)=Z(K)
WT(I)=WT(K)
K=K-1
200 NR=NR-1
IF (NR.GT.0) GO TO 100
NTP+K=1
300 RETURN
END
FUNCTION ALDA(E)

COMMON/AREA2/C1,MUO,E1,E1S,SH,ANEM,POD0,P1,E0,NI0N,JE,JZ,NASW1,
INSEN,AKT,AXT,AM1,AMCI,ED0,POD0,POD6,K1,K2,K3,WNZ,ED2,ED3
COMMON/AREA3/XSEL,XELC,XRCB,XMET,XEXC,XION,XTOL,XELC

REAL MUO,NI0N
WELC=1.0/KELC

IF (E.LE.5.) XELC=23.0*POD0
IF (E.GT.5. ) AND (E.LE.19.8) XELC=23.0*POD0*SRT(5./E)
IF (E.GT.19.8) XELC=EXP(-0.02*E)*16.76*POD0
XELC=XELC*WELC

IF (E.LE.3 1 GO TO 5001

4001 AE1= 0.25 +E - 19.8O)**2

XMET= 0.109*(E - 19.80)/AE1 + 0.0311

XMET=XMET*POD0

E19= E - 21.4

E24= E - 21.4

E24= E - E

E3= 4900. + E24**2

XION= 172.*E24/AE3

XION=XION*POD0

IF (E.LT.24.46) XION=0.0

XTOL= XMET +XEXC +XION +XELC

XRCB=0.0

XSEL=0.0

GO TO 6001

5001 BE1=0.250*(E-19.8)**2

XSEL=0.109*(19.8 -E )/BE1

XSEL=XSEL*POD0*ANEM

XRCB=POD0*(NI0N*SRT(E))

XION=0.0

XEXC=0.0

XMET=0.0

XTOL= XSEL *XELC +XRCB

6001 ALDA=1.0/XTOL

RETURN

END
SUBROUTINE COLPRO1, DCEN, YM, KF, NUJ, N + 1

COMMON D, E(0,0), Z(0,0), U(0,0), T(0,0), X(0,0), N

COMMON /AREA1/ NIZN, NOIZN, NAS, DIZN(500), NSE, NNET, NEXC, ETA, XMET, WNEC, WNET, WISE


COMMON /AREA3/ SEL, XELC, XRCB, XMET, XNEC, XION, XION, XION, XION, XION


COMMON /AREA5/ ISEL, XELC, XRCB, XMET, XNEC, XION, XION, XION, XION, XION


DIMENSION PX(14)

FM31E1 = EXP(-E*AIK1)*SORT(E)*AMCI

REAL MUO, _NION

C DECIDE WHICH KIND OF COLLISION

XTI = ALDA(E(1))

IF (E(1), LE, ED3 ) GO TO 5000

C, ENERGY REGION 1 19.0EV TO 1 KEV

4000 PMET = XNET*XTI

4010 PEXC = XNC*XTI

4014 PION = XION*XTI

4015 PELC = XELC*XTI

4016 PX(1) = PEXC

4017 PX(2) = PEXC + PMET

4018 PX(3) = PX(2) + PION

4019 [PM]

4020 DO 45 IPM = 1, 9

4021 IF (RIPM, GT, PX(I)) IP = IPM + 1

4022 GO TO (4015, 4013, 4017, 4011), IP

4023 4011 R1 = RAN32(10)

4024 U11 = 2.0*R1 - 1.0

4025 RETURN

C INELASTIC COLLISION PROCESS EXCITE TO META STABLE STATE

C EXCITATION TO META STABLE LEVEL

4026 4013 WMET = WNET + FT1

4027 ELJ = E(1) - 19.9

4028 IF (ELJ, LE, 0.0) RETURN

4029 WMAT = WMET

4030 R1 = RAN32(10)

4031 U11 = R1*R11 - 1.0

4032 RETURN

C EXCITATION TO OTHER LEVELS

4033 4015 WNET = WNET + FT

4034 E(1) = E(1) - 21.46

4035 IF (E(1), LE, 0.0) RETURN

4036 WNEC = WNEC

4037 R1 = RAN32(10)

4038 U11 = R1*R11 - 1.0

4039 RETURN

C IONIZATION PROCESS PRODUCE SECONDARY ELECTIONS

4040 4017 WHEN = WHEN + WT1

4041 E(1) = E(1) - E1

4042 IF (E(1), LE, 0.0) RETURN

4043 NIZN = NIZN

4044 NODN = NODN + 1
C IONIZATION PRODUCED PARTICLES ALWAYS PUT TO THE END OF RESIDUES

0045 DIIZNINOEN= OGEN
0046 RI= RANS(0)
0047 UE(I)=RI+1.0
0048 NNTP = NTP + NIZN
0049 E(NNTP)=E(I)+0.1
0050 Z(NNTP)=Z(I)
0051 U(NNTP)=U(I)
0052 E(I)=E(I)+0.9
0053 WT(NNTP)=WT(I)
0054 IF (E(NNTP) .GT. ED1) OR (E(I)+1.111.E11.LT.FD11) GO TO 4018
0055 NON=NON+1
0056 NON=NON+1
0057 WT(NNTP)=WT(I)*K3
0058 IRR(NOND)=NTP+NIZN
0059 CONTINUE
0060 RETURN

C ENERGY REGION 2 LOW ENERGY RANGE DOWN TO THERMAL REGION

0061 5000 PELC= ACEL*XTI
0062 PSEL= XSEL*XTI
0063 PRCB= XRCB*XTI
0064 PX[I]= PRCB
0065 PX[I]= PSEL + PRCB
0066 IP=1
0067 GO TO 55
0068 55 IF(RP.GT.PX[I]) IP=IP+1
0069 GO TO (5013,5015,5011,IP)

C ELASTIC COLLISION PROCESS

0070 5011 IF (ELS.LE.0.0) GO TO 5013
0071 IF (LINS(S.LT.8.1),OR.(E(I),G.T.E.2.5)) GO TO 70 5012
0072 50115 RE=RANS(0)
0073 R2=RANS(0)
0074 E1=ED2*RE
0075 E2=ED2*RE
0076 FR=PM3(I)
0077 FR2=PM3(E2)
0078 IF (RE*LE,FR2) RS=RE
0079 IF (E2,FR,FR1) RS=RE
0080 IF ((RE,LE,FR2),OR.(RLE,FR1)) GO TO 5012
0081 GO TO 50115
0082 5012 ENS=RS*ED2
0083 50124 IF(LINS(L.GT.ED1),OR.(E(I),G.T.E.2.5)) ENS=0.0
0084 R1=RANS(0)
0085 U1=2.*RI-1.0
0086 UM=SQRT(1.0-U1**2) (1.0-U1)**2)
0087 US2=UM+2.0
0088 COSR=UM(1+SIGNUM,US)
0089 DELE=CI(I1)-ENS(1.0+COSS)*ELS*KELC
0090 E11=E11 -DELE
0091 Go TO 5012
0092 50125 IF (ELS.LE.0.0) U11= 2.0*RANS(0) -1.0
0093 RETURN

C RECOMBINATION PROCESS
FORTRAN IV G LEVEL 18 COLPRO DATE = 72126 05/31/02

0094 5013 NAS = NAS + 1
0095 IAS(NAS) = I
0096 WNASE = WNASE + WT(I)
0097 IF (IE(I) .GT. 0.0) ETAS = ETAS + ET(I) * WT(I)
0098 RETURN

G SUPERELASTIC COLLISION PROCESS

0099 5015 IE(I) = IE(I) + 1
0100 WNSE = WNSE + WT(I)
0101 NSE = NSE + 1
0102 RP = RAN31(10)
0103 UI(I) = RP * RP - 1.0
0104 RETURN
0105 END
```
119 FORTRAN, IV G LEVEL 19 MAIN DATE = 72129 01/20/29

0033 C4=2.*5.030549/4.033973
0034 ETV=5012.*1.040203/4.01081*1.06*08
0035 VEC=1.0(ETV*ETV)
0036 CALL UNERZ('OFF')
0037 PUD6*PUDU=25.0
0038 MUC=(1.0<91)/ETV
0039 CHUQ=1.0*09
0040 CHUPD=CMUJ*PDPP
0041 IF (PDPP<0.10*C) CMUUPD=CMUQ*10*0
0042 PDHUO=CMUPO/ETV
0043 CLUSS=","
0044 NUP=0
0045 NUN=0
0046 NNP=0
0047 INO=0
0048 NKR=0
0049 MAS=0
0050 NEE=0
0051 NSE=0
0052 NQIN=0
0053 NIM=0
0054 NIN=0
0055 LK=0
0056 INIV=999999
0057 NMET=0
0058 NEX=0
0059 ETA5=5.0
0060 ETEP=0.0
0061 NIPS=0
0062 NW=0
0063 NISP=0
0064 DTAVS=3.0
0065 CSFZ=0.0
0066 CNO2=0.0
0067 CSEL=0.0
0068 CETAS=0.0
0069 CETEMP=0.0
0070 NGOS=0
0071 WNFET=0.0
0072 WNSE=0.0
0073 WNEC=0.0
0074 WNES=0.0
0075 WNAS=0.0
0076 WNI24=0.0
0077 JEE=0.0
0078 IE=JE
0079 DEC=3.566E16*PDPO.
0080 DETA=0/JZ
0081 DIVZ=1.0/DEZ
0082 CMBT=1.0/LOG102.0
0083 E02=0.4472313
0084 E03=19.0
0085 PM=ED2*A1K
```
FORTRAN IV G LEVEL IN MAIN DATE = 72123 01/28/69

0086 AMI2,A1KTY/SORT(P1*AKT)
0087 CALL FH3(IN19#G32211)
0088 S0=PDP+SM1,DE+16
0089 A1K=} FMUPP/((AKT*P1)
0090 DE1=0,010
0091 DE2=3,55
0092 DE3=0,3
0093 DE4=0,016
0094 UI2=1/DE2
0095 UI3=1/DE3
0096 AMI1= SORT12.C*2.71R3*AKT1*ED2
0097 DO 11 I=8,16
0098 ERL=(I-3)*DE2-4,0
0099 EMV= ERL-1,1685
0100 FOR(I1= 2,0**ERL
0101 FG(I-1)= 2,0**EMV
0129 N1 PDF(I)=END(1)(2,0**DE2-1.0)
0130 JEL=J+1
0131 DO 12 I=37, JEL
0132 FAL= 7,015*(I-11)*DE3
0133 EMV= ERL-1,1405
0134 FOR(I1= 2,0**ERL
0135 FG(I-1)= 2,0**EMV
0136 DO 112 I=2,0**DE3-1,0)
0137 DO 10 I=1, 5
0138 FDE(I)= 0.016
0139 EGI(I)= EDE(I)+I-0,008
0140 EDE(I1)= EDE(I)+I-11
0141 CDE(I1)=29,6
0142 EGI4(I)= 1737,482
0143 IF INTAPE.NE.11 GO TO 2
0144 READ1(I) IAVG, TAVG, NTW, NT, NTP, NION, DDEG, NSW, SWF, F, U
0145 READ1(I) Z, WT, ANIOS, SF3
0146 N5=NSW*K2
0147 REWIND 10
0148 NT=NT+1
0149 NT=NT+1
0150 2 CONTINUE
0151 IF (NTAPE.NE.2I GO TO 5
0152 READ(I) IAVG, DAVG, THE, NTW, NT, NTP, NION, DDEG, NSW, SWF, F, U
0153 READ(I) Z, WT, ANIOS, SF3
0154 N5=NSW*K2
0155 RFVNO 9
0156 "J", I+1
0157 5 CONTINUE
0158 IF (INTAPE.NE.0I GO TO 122
0159 NIOD= SORT(50*10/AITG)
0152 ANEAM=MUON*0,3333/DEN
0153 CALL SOURCE2(NTP, EAVG, NTW, PH, P1)
0154 AL=ALOA(EAVG)
0155 DAVG=AL
0156 DDEG=DAYG*ND/SORT(EAVG)
0157 SF2=1,0
0158 IF (NION.NE.0). SF2=MTW/HION
FORTAN IV G LEVEL 18

MAIN

0100 NSM=ANSW*0.5
0101 IF (NSM GT ANSW) NSM=NSM-ANSW
0102 SW=NSM/SN
0103 SF1=SW
0104 ANS=SF1
0105 CALL SOURCE(NS,TNP)
0106 SF1=SF2
0107 SF3=SF2
0108 THEO,C
0109 122 CONTINUE

123 C GENERATE SOURCE PARTICLES TO INITIALIZE THE PROGRAM
124 D E N = N I O N = 0.33333/DEN
125 WRITE(6,71)
126 70 FORMAT('1. PROGRAM INITIALIZATION AND THF INITIAL PARAMETERS',/)
127 WRITE(16,71) ALTC, N N N, TNP, AL, EVG
128 71 FORMAT('1. A T I G * E14.6, 3X * N I O N * , E14.6, 3X * TNP * , E12.3X)
129 72 FORMAT('1. R X , * NCOL, 13X * DCER, 13X * ENERGY, 11X * , 14X * U, 10X, 14X * WEIGHT, */
130 WRITE(14,72)
131 N I N T * DEN*310)
132 73 FORMAT('1. C WITHIN THE OUTER I LOOP THE I TH PARTICLE HISTORY
133 C GEOMETRIC ROUTINE ENTERING HERE
134 722 IF (I(E11),LE,0,1,OR,(E11),EQ,1) GO TO 900
135 D C E N * D I L G * S Q R T (E11)
136 777 IF (((I(E1)+0.1),AND,((1+0.1),TNP),DCER*RN(S(N)*1))
137 775 N F L G = D O
138 WRITE(14,E11)
139 IF (E11,EQ,0) ETAG1=-1.0
140 3000 IF (E11,EQ,0) OR,(E11),EQ,0,1) GO TO 900
141 J=INT(Z11)*DIV+1
142 R1=RAN32(D1)
143 ALD=ALD(A(E11))
144 NCOL=ALD*ALD(101)
145 IF (NFLG,EQ,1) GO TO 809
146 IF((NFLG,EQ,1)) EO WRITE(16,72A) (DCOL,DCEN,E11),I11),U(I11),MT11)
147 IF((NFLG,EQ,1)) EO WRITE(16,72A) (DCOL,DCEN,E11),I11),U(I11),MT11)
148 IF((NFLG,EQ,1)) EO WRITE(16,72A) (DCOL,DCEN,E11),I11),U(I11),MT11)
149 809 NFLG=1
150 800 IF (DCEN,LT,NCOL) GO TO 900

151
FORTRAN IV G LEVEL 18

M A I N

D A T E = 7 2 1 2 5 1 0 / 2 4 / 2 4

0187 I G N + D G N + D G C L
0188 I I T + C ( C ) + D C L + U ( I )
0189 N C D + N C O L +
0190 I F ( M O D ( N C L + N C ) , E Q . 0 ) R I P = M A N Z I G O I
0191 I F ( M O D ( N C L + N C ) , N E . 0 ) R I P = 1 . 0 + R I P
0193 G O T O 3 " . 0
0194 9 0 2 2 ( I ) + ( I ) + D G C N + U ( I )
0195 I F ( ( I ) . L T . E G O ) O P . ( ( Z I I ) . L T . 0 ) G O T O 9 0 0 6
0196 J I N T ( ( I ) . O D E U + 1)
0197 I F ( J . G T . J ) J = J
0198 K = I N D E X ( E ( I I ) )
0199 I F ( K . G T . J ) K = J
0200 I F ( K , N C O L ) G O T O 9 0 4
0201 E T A G 2 = 1 . 0
0202 I F ( E ( I I ) . G T . 1 ) E T A G 2 = - 1 . 0
0203 I F ( ( E T A G 1 ) . E T A G 2 ) . G T . 0 ) G O T O 9 0 4
0204 I F ( E T A G 2 . G T . 0 ) G O T O 9 0 3
0205 N U P = N U P + 1
0206 F S C ( N I P ) = E ( I I )
0207 W T ( I ) = W ( I ) / K 3
0208 W S F ( N U P ) = W ( I )
0209 G O T O 9 0 4
0210 9 0 3 3 I F ( ( L F G 2 . G T . 1 ) . O R . ) ( ( N D P N . L T . 2 ) ) N D N R = N D N R + 1
0211 I F ( ( L F G 2 . L T . 1 ) . O R . ) ( ( N D P N . L T . 2 ) ) I R R = N D N R + 1
0212 N D N R = N D N R + 1
0213 I F ( ( M O D ( N D N R , K 3 ) ) . N E . 0 ) G O T O 9 0 4 5
0214 W T ( I ) = W ( I )
0215 N E Z ( K , J ) = N E Z ( K , J ) + W T ( I )
0216 F K ( K , J ) = E K 2 ( K , J ) + W T ( I )
0217 9 0 4 5 I F ( ( L F G 2 . G T . 1 ) . O R . ) ( ( N D P N . L T . 2 ) ) G O T O 9 0 0 9
0218 L F G 2 = 1
0219 F ( I ) = E D
0220 U ( I ) = U ( I )
0221 Z ( I ) = Z ( I )
0222 D C E N = D C F K 2
0223 N C D = N C D L 2
0224 G O T O 9 0 0 9
0225 9 0 0 3 N E S = N E S + 1
0226 I F ( ( E ( I ) . L T , E O D ) ) W T ( I ) = W T ( I ) / K 3
0227 W N S = W N S + W T ( I )
0228 L E S ( N E S ) = 1
0229 I F ( ( E ( E ) . G T . 0 ) ) E T E P = E T E P + E ( E ) W T ( I )
0230 9 0 0 9 L = L + 1
0231 L F G 2 = 0
0232 I F ( ( L F . N T P ) ) G O T O 7 7 7 7 7
0234 I F ( N O N Z , L T , 0 ) N I N = 1
0235 I F ( ( N I N Z , G T , 0 ) ) A N D ( ( N I N Z , E Q , 0 ) ) G O T O 7 7 7 8
0236 N I N Z = N I N Z + 0 , 5
0237 N W N S = W N S + 0 , 5
0238 W N S = W N S + W N S
0239 W N S = W N S + 5
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0240</td>
<td>F0RTRAN IV G-LEVEL: 10</td>
</tr>
<tr>
<td>0241</td>
<td>NWSE = WNSF/KRIG</td>
</tr>
<tr>
<td>0242</td>
<td>NSF = WSSE + .5</td>
</tr>
<tr>
<td>0243</td>
<td>DD 191 M = 1,1,JE</td>
</tr>
<tr>
<td>0244</td>
<td>DD 191 M = 1,1,JE</td>
</tr>
<tr>
<td>0245</td>
<td>E(IN) = E(M) + E(ZM,N)</td>
</tr>
<tr>
<td>0246</td>
<td>(IN) = NE(M) + NE(N)</td>
</tr>
<tr>
<td>0247</td>
<td>IF (K3, SO, 1) OR (IN, DQ, 01) GO TO 24</td>
</tr>
<tr>
<td>0248</td>
<td>M3 = K3/JJ</td>
</tr>
<tr>
<td>0249</td>
<td>IF (N3, LT, 1) M3 = 1</td>
</tr>
<tr>
<td>0250</td>
<td>IF (N3, LT, 1) IZ = NK</td>
</tr>
<tr>
<td>0251</td>
<td>IF (N3, DQ, 1) M3 = NT</td>
</tr>
<tr>
<td>0252</td>
<td>MJ 23 NE/MUP</td>
</tr>
<tr>
<td>0253</td>
<td>K1 = N0E(FE(E(N))</td>
</tr>
<tr>
<td>0254</td>
<td>E(K1) = E(K) * (K1-1)*ESE(FN)*WSE(N)</td>
</tr>
<tr>
<td>0255</td>
<td>NSE(K1) = NSE(K) + (K1-1)*WSE(N)</td>
</tr>
<tr>
<td>0256</td>
<td>22 IF (K1, GE, J) IZ = JZ</td>
</tr>
<tr>
<td>0257</td>
<td>DD 21 L = 1,1Z</td>
</tr>
<tr>
<td>0258</td>
<td>31 <em>FIL</em>E(Z) + FSE(E(N)) <em>M2</em>WSE(N)</td>
</tr>
<tr>
<td>0259</td>
<td>NK3 = K3 - JZ = M3</td>
</tr>
<tr>
<td>0260</td>
<td>IF (NM3, LT, J) IZ = NK3</td>
</tr>
<tr>
<td>0261</td>
<td>M3 = 1</td>
</tr>
<tr>
<td>0262</td>
<td>IF (NK3, GT, J) GO TO 22</td>
</tr>
<tr>
<td>0263</td>
<td>23 CONTINUE</td>
</tr>
<tr>
<td>0264</td>
<td>24 CONTINUE</td>
</tr>
<tr>
<td>0265</td>
<td>MJ 192 M = 1,1 JF</td>
</tr>
<tr>
<td>0266</td>
<td>DU 1915 N = 1,1,JE</td>
</tr>
<tr>
<td>0267</td>
<td>WP1 = WP1 + NEZ(N)</td>
</tr>
<tr>
<td>0268</td>
<td>1915 E(IN) = E(IN) + FK7(Y, M)</td>
</tr>
<tr>
<td>0269</td>
<td>122 CONTINUE</td>
</tr>
<tr>
<td>0270</td>
<td>NTM = NTM + NIIN = NES = NAS</td>
</tr>
<tr>
<td>0271</td>
<td>NTP = N1,0</td>
</tr>
<tr>
<td>0272</td>
<td>FTUL = U</td>
</tr>
<tr>
<td>0273</td>
<td>OD 14 M = 1,1,JE</td>
</tr>
<tr>
<td>0274</td>
<td>FTUL = FTUL * D6(M)</td>
</tr>
<tr>
<td>0275</td>
<td>14 NTPX = NTPX + NE(M)</td>
</tr>
<tr>
<td>0276</td>
<td>13 EWTUL = EWTUL</td>
</tr>
<tr>
<td>0277</td>
<td>NTP + NTP</td>
</tr>
<tr>
<td>0278</td>
<td>DAVG = DAVG + NW</td>
</tr>
<tr>
<td>0279</td>
<td>DAVS = DAVS + DAVS</td>
</tr>
<tr>
<td>0280</td>
<td>21 NTP = NTP + NTP</td>
</tr>
<tr>
<td>0281</td>
<td>20 NTP = NTP + NTP</td>
</tr>
<tr>
<td>0282</td>
<td>DAVG = DAVG + NW</td>
</tr>
<tr>
<td>0283</td>
<td>NTP = NTP + NTP</td>
</tr>
<tr>
<td>0284</td>
<td>000 NTP = NTP</td>
</tr>
<tr>
<td>0285</td>
<td>DAVS = DAVS + DAVS</td>
</tr>
<tr>
<td>0286</td>
<td>NTP = NTP + NTP</td>
</tr>
<tr>
<td>0287</td>
<td>NTP = NTP</td>
</tr>
<tr>
<td>0288</td>
<td>DAVG = DAVG + NTP</td>
</tr>
<tr>
<td>0289</td>
<td>NTP = NTP</td>
</tr>
<tr>
<td>0290</td>
<td>NTP = NTP + NW + .5</td>
</tr>
<tr>
<td>0291</td>
<td>NTP = NTP + NW</td>
</tr>
<tr>
<td>0292</td>
<td>NTP = NTP</td>
</tr>
</tbody>
</table>

Note: The code appears to be a FORTRAN program with several variables and conditional statements.
FORTRAN IV G LEVEL 1B  MAIN  DATE = 72125  01/28/29

0293  DAVG=DOTOT*0.9/NTP
0294  EAVG=ETOT/NTP
0295  DEG=ODDFG/ETV
0296  TMP=TMP+DEG
0297  TMS=TMS+DEG
0298  C IF SUVERSE PARTICLES SO(NS)C NES THEN ELIMINATE AS MANY RESIDUE PARTICLES AT TH
0299  NTPS=NTPS - NMS - NAS
0300  DGENL=ALDAI EAVG
0301  NT=NTPW + 5
0302  DDEG=DAVO/ND/SQRT(EAVG).
0303  SFR=1.0
0304  IF (ANIOS,NE,0) SFR=NTPS/ANIOS
0305  EAVG=Q.O
0306  IF(NWES,NE,0.0) EAVG=EYEP/WNES
0307  EAVG=Q.0
0308  IF (NWAS,NE,0.0) EAVG=ETAS/WNAS
0309  IF (NWES+WNAS) GT.0.0, SF=SF(WHT/FETV)
0310  IF (NWES+WNAS) GT.0.0, RF=1.0
0311  CN55=CN55+SN*DAVS
0312  CSF2=CSF2 + (ETEP*ETAS*4.0*WHT/ETV)
0313  IF(15MOD(INT,5)>=0.01,AND,(INT,NE,0)) SF=CSF2 /IS0*0.95.
0314  CSF1+ SF+ DAVS
0315  IF (15MOD(INT,5)>=0.01,AND,(INT,NE,0)) SF=CSF1*DISG
0316  IF (15MOD(INT,5)>=0.01,AND,(INT,NE,0)) SF=CSF2*DISG
0317  AVOG=SF*SF
0318  WHT=SF1/3F
0319  SN=SU*IDEG*SF3*WHT/ETV
0320  IF (SN,GT,ANSW) NS=SN + 0.5
0321  WSF=NSF/S4
0322  NS=NS*K3
0323  SFR=SF
0324  NE=NE+MET + 0.5
0325  NE=NE+EXC + 0.5
0326  WVEL=WSVLE + WVALUE*DAVS
0327  CETEP=CEtep + FTEP*DAVS
0328  CETAS=CETAS + ETAS*DAVS
0329  CN52=CN52*WHT/ETV
0330  CNES=CNES*NWS*DAVS
0331  CN55=CN55*NWS*DAVS
0332  CNMET=CNMET+WMET*DAVS
0333  CNEXC=CNEXC+NEC*DAVS
0334  CEAVG=CEAVG + ESAVG*DAVS
0335  CEASG=CEASG + EAVG*DAVS
0336  CLONG=CLONG + (WHT+WNES)*DAVS/(TMS*SF3*WHT)
0337  WVL1=WVEL+DISG
0338  ALISS=CLONG+DISG
0339  ANISG=CN55+DISG
0340  ANISVG=CN55+DISG
0341  ATEP=AFTEP+DAS
0342  ETA5=CETAS/DAS
0343  IF (ANISVG,NE,0) WVL2=(EO*ANISG-ATEP*ETAS)/ANISVG.
FORTRAN IV, LEVEL 1B  MAIN  DATE = 72128  01/30/20

0344  ANES=CNFS/DTAVS
0345  ANAS=CNAS/DTAVS
0346  AESAV=CESAVG/DISG
0347  AERG=CEASG/DISG
0348  ANHET=CNHET/DISG
0349  ANEX=CNEX/DISG
0350  ANS=ANES+ANAS-ANIZVG

0351  IF (ANS.LT.0.0) ANS=NS/NW
0352  IF (WINT.0.0) WVALUE=LENEO-AETEP-AETAS/ANIZVG
0353  IF (WINT.0.0) WVALUE=ANSEP-EETEP/ANIZVG
0354  IF (NS.GE.1) TNS=0.0
0355  MET=NESSNAS

CALL SOURCE(NS,NTP)

0356  IF (NET.GT.0) CALL FIELDLINTP,NES, NAS
0357  NINT=NINT+NIE
0358  IF (YUP.GT.0) CALL SEPROCINTP,K31
0359  IF (NONA.GT.0) CALL RRPINTP,K31
0360  NASS=NAST+NWS

C INTRODUCE THE SOURCE PARTICLES FOR THE NEXT CYCLE

C EVERY FIFTH CYCLE RENORMALIZE NEON AND OUTPUT QUANTITIES ARE PRINTED

C RENORMALIZATION PROCESS OF NEON

1362  ANSTP=1.0/NSTP
1363  ANTPS=DAVS/NTPS
1364  NTPS=NSTP
1365  DO 18,J=1,TE
1366  CNEZ(M)+CNEZ(M)+NREZ(M)*DAVS
1367  16 CEE(M)+CEE(M)+EE(M)*DAVS
1368  DO 19,J=1,TE
1369  PNEZ(M)+NREZ(M)+ANTPO
1370  16 EE(M)=F7M1/NREZ(M)
1371  SUM=EE
1372  DO 199,M=1,JE
1373  SUM+=SUME+NE(M)
1374  TNe(M)+TNE(M)+NE(M)/EDE(M)
1375  193 FNE(M)+NE(M)+ANTPO/EDE(M)
1376  EAV=0.0
1377  DO 1755,M=1,JE
1378  EAV=EE(M)+FNE(M)/EDE(M)+EAV
1379  1755 CNE(M)=TNE(M)*ANSTP
1380  CND=CND+CTP**2.7*AIK/SORT(AIK*PI)
1381  ANINV=0.0
1382  ANINV=0.0
1383  DO 20,J=1,JE
1384  ANINV=ANINV+F(1E6(M),GEN(M)+FNE(M)*EDE(M)
1385  ANINV=ANINV+F(1E6(M),FNE(M)*EDE(M)
1386  20 CONTINUE
1387  CNION=ANCH+CN1ION
1388  CNION=ANCH+CN1ION
1389  CONTINUE

0390  IF (ALCSS.LT.0.0) CJD=0.0
0391  IF (ALCSS.LT.0.0) CJD=110*50
0392  THION=ANIS+CJDI/DNION
0393  -ANIO2+ASS+G+DION2
FORTRAN IV G LEVEL 18

0394   TNI02=SQR(TNI02)
0395   TNI02=SQR(TNI02)
0396   CN102=CN102 + TNI02*DAVS
0397   ANIOS=ANIOS*0.156
0398   DNT=1.0
0399   IF (ANES+ANAS,NE.0,0) WFL=ANIZ*GAMSG/7(ANES+ANAS)
0400   IF (IOMUN,0,0) GO TO 1700
0401   NIONS=NIONS
0402   IF ((WFL.GT.0.005).AND.(WFL.LT.200.0)) NIONS=ANIOS*WFL*HAF
0403   1790 CONTINUE
0404   DAVT=DAVO+DNT
0405   ALPHA=ANIZ/NTPW*DAVT
0406   ANEOS=NIONS*0.33333/10
0407   1799 FORMAT(T*ANION*,E13.4,E13.4,13X,14X,9E12.5,9X/4
0408   WRITE(6,1799) ANION, ON102, TNI02, ALPHA, ALOSS
0409   WRITE(6,1799) EG, PDPO, D, DETI, DEG, NDPN
0410   172 FORMAT(T*GAS,HE,2X,SOURCE ENERGY*,F5.0,3X, PRESSURE*,F4.0,
0411   X,M1Z**,3X,SLAB THICKNESS*,F4.2,E1**, DX*,F4.1**,EM#,4X,DELT,
0412   X*E13.5,3X,19**PROG!,12**
0413   WRITE(6,1800) NT, TME, NS,NSW
0414   140 FORMAT(T*NUMBER OF TIME CYCLES*,16,3X,TIME =,12.5, SECT*,
0415   1X,*GENERATED SOURCE PART =",15,3X,"NOTED # OF SOURCE PART","115 )
0416   WRITE(6,1805) DAVT, DGEN, DAVG, EWAVG, EAVG
0417   1H:5 FORMAT(T*DAVG =*,F12,5,3X,"DGEN =",F12,5,3X,"OVERALL",FAVG,*
0418   IF (F10.5,3X,"WRITE DAVG=",F10,5,3X,"EAVG CHECK=",F10,5,3/,
0419   WRITE(6,1811)
0420   1H:4 FORMAT(T*1X,NUMBER DISTRIBUTION*,2X,"FLUX DENSITY*,4X,"AVG *
0421   IF (F10.5,3X,"CUMULATED DISTRIBUTION*,3X,"ENERGY INTERVAL",2X,"CUM FLUX
0422   XEN*,4X,"AVG.DEV =",4X/
0423   WRITE(6,1821) DAVG, EAVG
0424   1H:6 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0425   IF (F10.5,3X,"WRITE DAVG=",F10,5,3X,"EAVG CHECK=",F10,5,3/,
0426   1H:7 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0427   IF (F10.5,3X,"WRITE DAVG=",F10,5,3X,"EAVG CHECK=",F10,5,3/,
0428   1H:8 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0429   IF (F10.5,3X,"WRITE DAVG=",F10,5,3X,"EAVG CHECK=",F10,5,3/,
0430   1H:9 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0431   WRITE(6,1831) DAVG, EAVG, FLX1, GEMI, EDE(M), FLX2, DNAV
0432   1H:10 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0433   WRITE(6,1841) DAVG, EAVG, DET, DAVGO, NCPD
0434   1H:11 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0435   WRITE(6,1851) DAVG, EAVG, DET, DAVGO, NCPD
0436   1H:12 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0437   WRITE(6,1861) DAVG, EAVG, DET, DAVGO, NCPD
0438   1H:13 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0439   WRITE(6,1871) DAVG, EAVG, DET, DAVGO, NCPD
0440   1H:14 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0441   WRITE(6,1881) DAVG, EAVG, DET, DAVGO, NCPD
0442   1H:15 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0443   WRITE(6,1891) DAVG, EAVG, DET, DAVGO, NCPD
0444   1H:16 FORMAT(T*CFSUM*,E12.5,5X,E12.6,5X,E12,5,2X,E12,5,2X/
0445   WRITE(6,1901) DAVG, EAVG, DET, DAVGO, NCPD
FORTRAN IV G LEVEL 14  MAIN  DATE = 72122  01/28/29

0435 100 FORMAT(6/3X, 'TOTAL # OF PARTICLES PROCESSED = ',I5, 3X, 'CUMULAT
0436
0437 197 FORMAT(6/3X, 'ZONE NUMBER', 5X, 'AVERAGE ENERGY', 5X, 'NUMBER OF
0438  IF ELECTRONS', 3X, 'CUMULATIVE # DEN', '3X, CUMU. AVG ENERGY', 3X,
0439  'ZONE # DENSITY', 5X, )
0440
0441 1195 FORMAT(5, 5X, 15, 5X, F14.6, 3X, F14.6, 5X, F14.5, 7X, F14.6, 7X,
0442  LE12.5)
0443
0444 185 FORMAT(10, 5X, 'NUMBER OF POSITIVE ION', 'E14.6', 1X, 'NUMBER OF IONIZAT
0445 1IONS OCCURRED: ', F9.3, '5X, 'ACTUAL', '15, '3X, 'CUMULATIVE AVG', 'F10.5,
0448 X, 'CUMULATIVE AVG: ', F9.2, '2X, ' / TOTAL # OF PARTICLES IN THE NEW CYC
0449 L: ', 13, '1X, ', K1 = 15, '3X, ', K2 = 112, '3X, ', X1 = 8, ' X1 = 8, ' BALANCE FACT OR	WF = ',
0449 APFl(3, 3X, 'RF2 =', F10.5, ')
0450
0451 197 FORMAT(5, 5X, 'E12.4', 1X, 'ENERGY DIVIDING LINE FOR RR PLAY E =', F10.4, '4X,
0452 'EVERY', 12 X, 'CYCLE CHANGE N	N+', 3X, 'TIME INTERVAL MULTIPLY. FACTOR =', E12.4, 4X,
0453 XX, 'ELASTIC X SEC / BY', 112.4, 1X
0454
0455 185 FORMAT(10, 5X, 'NUMBER OF SOURCE PARTICLES GENERATED OLD = ', I12.4,
0456 1X, 'SOURCE PARTICLES (WEIGHTED) IN THE NEW CYCLE = ', I12.4,
0458 1X, ' # OF SUPER ELASTIC COLLISIONS = ', F10.4, 1X, 1
0459 
0460 118 FORMATT(5, 5X, ' # OF EXCITATIONS TO HPA START LEVEL = ', F10.4,
0460 3X, 'CUMULATIVE AVG = ', F9.3, 1X, ' # OF EXCITATIONS TO OTHER LEVELS = ', F10.4,
0460 3X, 'CUMULATIVE AVG = ', F9.2, 2X, 1X, ' # OF PARTICLES CROSS OVER TO HIGH
0460 ENERGY REGION NUP = ', I15, 'XX, OVERALL MT = ', E12.4, 1X
0461
0461 185 FORMAT(10, 5X, 'EF, EDP, PDD', '5F1, SP2, SF3
0461 X5, '3F1 = ', 112.4, 'XX, '5F2 = ', E12.4, '3X, '5F3 = ', E12.4, 1X
0462
0462 197 FORMAT(5, 5X, 'EQV, VRCT', 'MVALVE', 'MVE1, MVE2, MVSEG, MSAVG, MSAVG, MSSAVG',
0462 3X, 'XAVG
0463
0463 170 FORMAT(5, 5X, 'SOURCE RATE = ', E11.3, 1X, 'PER SEC', '4X, 'W-VALUE (LOCAL I =',
0463 X10.3, 3X, 'SOURCE RED. FACT = ', F13.4, 1X, 'W-VALUE =',
0463 X10.3, 3X, 'CUM AVG W1 = ', F9.3, 3X, 'CUM AVG MV2 = ', F9.3,
0463 1X, ' AVERAGE ENERGY OF ESCAPED PARTICLES = ', F12.4, 3X, 'CUMULAT
0463 XAVE AVG = ', F11.4, 1X, 'AVERAGE ENERGY OF ABSORBED PARTICLES = ', F12.5,
0463 X5, 'CUMULATIVE AVG = ', F11.4, 1X
```
1 CALCULATED N USE P. USE CN = 1.14, 2.3K
0.144, 2.3K, CALC: N USE CN = 1.14, 2.3K.

0457 IF (MOD(INT1, 10) .NE. 0.) GO TO 3
0458 WRITE(11) EURO, AY, THEE, NY, NT, NTP, NIDN, ODEG, NSW, SWF, E, U
0460 ENDFILE 9
0461 REWIND 9
0462 CONTINUE
0463 IF (MOD(INT1, 10) .NE. 0.) GO TO 4
0464 WRITE(11) IAO, AY, THEE, NY, NT, NTP, NIDN, ODEG, NSW, SWF, E, U
0466 ENDFILE 10
0467 REWIND 10
0469 IF (MOD(INT1, NCM) .NE. 0.) GO TO 6666
0470 NSTP = 0
0471 NTP = 3
0472 CSF = 1.
0473 NST = 0
0474 NJ = 0
0475 NICE = 0.
0476 CJS = 0.
0477 CSF = 1.
0478 CN12 = 0.
0499 CKE = 0.
0480 CN9 = 1.
0481 CNMET = 0.
0482 CKE = 0.
0493 CSEAV = 0.
0494 CESA = 0.
0485 DO 54 M = 1, J1
0500 DO 55 M = 1, J1
0486 CNE2(M) = 0.
0487 CNE2(M) = 0.
0498 DS = 0.
0499 DS = 0.
0500 NME = 0.
0501 NEE = 0.
0502 TES = 0.
0503 TES = 0.
0504 DO 51 N = 1, 50
0505 WSM(N) = 0.
0506 WSM(N) = 0.
0507 1 = 1
```
C. REINITIALIZE THE OUTPUT QUANTITIES

0508    DO 52 M=1,JE
0509    RE(M)=Q
0510    DO 52 NE(M)=0.0
0511    DO 52 NE(M+1)=NE(M)
0512    NEZ(M)=0.0
0513    EZ(M)=0.0
0514    DO 53 N=1,JE
0515    KE(N)=KE(M)
0516    KE(N+1)=KE(N)
0517    NID(N)=0
0518    NID(N+1)=NID(N)
0519    NID(N+2)=NID(N)
0520    NES(N)=0
0521    NSES(N)=0
0522    NCU(N)=0
0523    NAS(N)=0
0524    DTP(N)=0.
0525    NME(N)=0
0526    NEXC(N)=0
0527    WNAM(N)=0
0528    WNSP(N)=0
0529    WNEX(N)=0
0530    WNLZ(N)=0
0531    WNET(N)=0
0532    WEXC(N)=0
0533    ETAS(N)=0
0534    FTP(N)=0
0535    LIT(N)=0
0536    NUP(N)=0
0537    NQUP(N)=0
0538    NNV(N)=0
0539    INV(N)=1.0
0540    INV=INV+1.0
0541    CALL NMS(INV)
0542    IF INT(N,I) GO TO 7777
0543    7777 STOP
0544    DEBUG SHUCHK
0545    END
VITA

Benjamin Shaw-hu Wang was born in [redacted]. He is the son of Mr. C. H. Wang, the former Chinese Army General now the Congressman of the Republic of China. He received the B.S. degree in Electrical Engineering from the National Taiwan University of Taipei, Taiwan, China in 1960, he then served as an ensign in Chinese Naval Academy for one and a half years. He came to the United States in September, 1962 and enrolled in the University of Missouri at Rolla, Missouri. After he received the M.S. degree in Electrical Engineering in 1964, he joined the Westinghouse Research Laboratories and served as a research engineer for three and a half years.

In September 1967, he came to the University of Illinois at Urbana-Champaign to undertake studies for a Ph.D. in Computer Science Department. He has been a research assistant and teaching assistant in the Department of Computer Science and a research assistant in the Nuclear Engineering Program. He is the member of Kappa Mu Epsilon and Pi Mu Epsilon the two national honor societies of mathematics.