MONTE CARLO SIMULATION OF NONLINEAR RADIATION INDUCED PLASMAS

by

Benjamin Shaw-hu Wang

May, 1972
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
MONTE CARLO SIMULATION OF NONLINEAR RADIATION INDUCED PLASMAS

Benjamin Shaw-hu Wang, Ph.D.
Department of Computer Science
University of Illinois at Urbana-Champaign, 1972

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 torrs 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

CHAPTER                      Page

I. INTRODUCTION              1
   A. Statement of the Problem and Objectives  1
   B. The Importance of Nuclear Radiation Produced Plasmas and the Physical Model  1
   C. General Approaches and Methods of Solution  5

II. THE MATHEMATICAL MODEL: NUMERICAL SCHEME AND PHYSICAL PROBLEM  8
   A. General Description of the Physics of the Problem  8
      1. Physical Processes Involved  8
      2. Relation to the Transport Equation  10
   B. The Stochastic Model - Numerical Scheme  12
      1. The Geometry and the Coordinate System  13
      2. Tracing Procedure  20
         a. The System Without Electric Field (E_f=0)  21
         b. With Electric Field Present  25
         c. Types of Collisions and Cross Sections  27
      3. The Prediction and Correction of Dominant System Parameters  29

III. VARIANCE REDUCTION TECHNIQUES  33
     A. General Background  33
     B. Importance Sampling  38
     C. Russian Roulette and Splitting  42
     D. Initial Guess  44
     E. Antithetic Variates  46
     F. Double History (Double Processing) Technique  50
<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<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 Description</td>
<td>52</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 $E_0$). 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 (β-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 life time 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).
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

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \nabla f - \frac{\mathbf{u}}{m} (\mathbf{E} + \mathbf{u} \times \mathbf{B}) \cdot \nabla \mathbf{u} f = \left( \frac{\partial f}{\partial t} \right)_{\text{collisions}}$$

where $f$: number density distribution function
B: magnetic field
$\mathbf{u}$: velocity of the particles
$\mathbf{E}$: 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, $\nabla r f=0$, the distribution function $f$ depends only on energy. Then the above equation reduces to

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

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:

$$\frac{\partial f}{\partial t}_{\text{collision}} = \frac{dN_{\text{in}} - dN_{\text{out}}}{dV \cdot dt} \tag{3}$$

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: $0 \sim 1$ (thermal energy region), 1 to $ED_2$, and $ED_2$ 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 \( w \): the weight. When the particle has traveled a path-length \( DCEN \) (in the same duration \( \Delta t \)), a condensed history
### Figure 2. Snapshops of the particles' condensed histories.

<table>
<thead>
<tr>
<th>i</th>
<th>t</th>
<th>t₀</th>
<th>t₁</th>
<th>tᵢ</th>
<th>tₘ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>E₀</td>
<td>E₁</td>
<td>Eᵢ</td>
<td>Eₘ</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Z₁</td>
<td>Z₁</td>
<td>Zᵢ</td>
<td>Zₘ</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U₁</td>
<td>U₁</td>
<td>Uᵢ</td>
<td>Uₘ</td>
</tr>
<tr>
<td></td>
<td></td>
<td>wt₀</td>
<td>wt₁</td>
<td>wtᵢ</td>
<td>wtₘ</td>
</tr>
</tbody>
</table>

| 2 |   | E₀ | E₂ | Eᵢ | Eₘ |
|   |   | Z₂ | Z₁ | Zᵢ | Zₘ |
|   |   | U₂ | U₁ | Uᵢ | Uₘ |
|   |   | wt₀| wt₂| wtᵢ| wtₘ|

<table>
<thead>
<tr>
<th>N</th>
<th>E₀</th>
<th>Eₙ</th>
<th>Eᵢ</th>
<th>Eₘ</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Zₙ</td>
<td>Zₙ</td>
<td>Zᵢ</td>
<td>Zₘ</td>
</tr>
<tr>
<td></td>
<td>Uₙ</td>
<td>Uₙ</td>
<td>Uᵢ</td>
<td>Uₘ</td>
</tr>
<tr>
<td></td>
<td>wtₙ</td>
<td>wtₙ</td>
<td>wtᵢ</td>
<td>wtₘ</td>
</tr>
</tbody>
</table>
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 $D_{CEN}$ (equivalent to $\Delta t$) are chosen such that the total number of steps should be kept as small as possible. Too large a step size ($\Delta t$) 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., $\Delta t = D_{avg}/V_{avg}$. Different time cycles have different $E_{avg}$ ($V_{avg}$) and $D_{avg}$ so $\Delta t$ may vary, but as steady-state is approached, $\Delta t$ 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 $\Delta t$ (or $D_{CEN}$) 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 $t=t_0$:

$$E_i^0 = E_{in} \quad (4a)$$

$$z_i^0 = r_i \cdot D \quad (4b)$$

$$u_i^0 = 2r_i - 1 \quad (4c)$$

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:
\[ dB = \frac{(j+1)AZ - Z}{u} \text{ if } u > 0 \]
\[ = \frac{Z - jAZ}{|u|} \text{ if } u < 0 \]
\[ = DLB \text{ (a large number } DLB \gg AZ) \text{ if } u \approx 0 \]

\[ d_{\text{col}} = \lambda(E) \left| \ln r \right| = \left| \ln r \right| / \sum_t(E) \]  

\[ d_{\text{cen}} = D_{\text{avg}} \cdot DI = V_{\text{avg}} \cdot At \cdot DI \]

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

\( D_{\text{avg}} \): the average value of \( d_{\text{col}} \) of \( N \) particles of the previous cycle. It is approximately one mean free path corresponding to \( E_{\text{avg}} \): 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 \((E_z=0)\)

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

\[ z' = z + u.d_{\text{cen}} \]  
\[ u' = u \]  
\[ E' = E \]

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:

$$z' = z + (d_B + 0.001 \Delta z) \cdot u$$

(9a)

$$j = \text{INT} \left( z'/\Delta z \right) + 1$$

(9b)

$$d'_{\text{col}} = d_{\text{col}} - d_B$$

(9c)

$$d'_{\text{cen}} = d_{\text{cen}} - d_B$$

(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 \( \mathbf{E}_f \). In the presence of electric field \( \mathbf{E}_f \), 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:

\[
\begin{align*}
V' \cdot \sin \theta' & = V \cdot \sin \theta \tag{11} \\
mV' \cos \theta' & = mV \cos \theta + e \mathbf{E}_f \cdot \Delta t \tag{12} \\
1/2 mV'^2 & = 1/2 mV^2 + e \mathbf{E}_f \cdot \Delta z \tag{13} \\
E' & = E + e \mathbf{E}_f \cdot \Delta z \tag{14}
\end{align*}
\]

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:

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

\[
= V \cdot u \cdot \Delta t + 1/2 \cdot \frac{E_f}{m} \cdot \Delta t^2 \tag{15}
\]

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

\begin{align*}
DCOL & = \lambda|1n\lambda| \\
& = 2/(1/\lambda(E) + 1/\lambda(E')) \\
DCEN & = \sqrt{V_y^2 + V_z^2 \cdot \Delta t} \\
V_y & = V \cdot \sin \phi \\
V_z & = V \cdot \cos \phi + \frac{E_f}{2m} \cdot \Delta t
\end{align*}

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

\begin{align*}
E' &= E + E_f \cdot \Delta z \\
z' &= z + \Delta z \\
u' &= \sqrt{E'/E} \cdot u + eE_f \cdot \Delta t \\
\Delta t' &= \Delta t - DCOL \cdot \Delta t/DCEN \\
DCEN &= DCEN - DCOL \\
z' &= z + \Delta z \cdot DCOL/DCEN \\
E' &= E + \Delta z \cdot E_f \cdot DCOL/DCEN
\end{align*}

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

\begin{align*}
\Delta t' &= \Delta t - DCOL \cdot \Delta t/DCEN \\
DCEN &= DCEN - DCOL \\
z' &= z + \Delta z \cdot DCOL/DCEN \\
E' &= E + \Delta z \cdot E_f \cdot DCOL/DCEN
\end{align*}

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).

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

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 $\cos^2 \theta_s = 0$ 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 $\cos^2 \theta_s$ and $\Delta E$, respectively, where

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

$$\theta_s = \theta - \theta_i$$
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:

\[
\sum(E) = \frac{B(E-E_i)}{A+(E-E_i)^2} + C
\]

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:

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

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

\[
\sum_{RCB}(E) = \mu(E) \cdot N^+
\]

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

where \( N^+ \): ion density

\( V \): electron velocity

the superelastic cross section is

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

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

\[
= \alpha(E) [N^+(E, z)]^2
\]

or

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

Integration of both sides gives

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

or

\[
N_0 = \left[ S_0 / \int_0^{E_0} \alpha(E) M^2(E) \, dE \right]^{1/2}
\]

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.,

\[
\iiint \frac{dJ(E, z)}{dz} \, dz \, dE = \iiint S(E, z) \, dz \, dE
\]

\[
- \iiint \alpha(E) \, N^+(E, z) N^-(E, z) \, dz \, dE = J_{esc} + J_s
\]

which reduces to

\[
J(Z=d) - J(Z=0) = S_0 \, d - \iiint \alpha(E) \, N^+(E, z) N^-(E, z) \, dz \, dE
\]

\[
- J_{esc} + J_s
\]
\[ \frac{N_i^2}{N_0^2} = \frac{S_0 + J_s - [J(z=d) - J(z=0)] - J_{esc}}{\int \alpha(E) M(E) n^-(E,z) \, dz \, dE} \]  

(33)

where \( i \) refers to ith time cycle

- \( J_{esc} \): 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 kth 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 \( R(x_1, x_2, \ldots, x_M) \) involving the sequence of random numbers \( x_1, x_2, \ldots, x_M \). This is an unbiased estimator of \( \int_0^1 \cdots \int_0^1 R(x_1, x_2, \ldots, x_M) \, dx_1 \cdots dx_M \). For the sake of simplicity, we take the one-dimensional integral as a standard example. Let

\[
\theta = \int_0^1 f(x) \, dx \tag{34}
\]

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 θ have variances \( \sigma_1^2 \) and \( \sigma_2^2 \). Then the efficiency of method 2 relative to method 1 is defined as

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

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

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

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

\[
\frac{1}{n} \int_{0}^{1} (f(x) - \theta)^2 dx = \frac{\sigma^2}{n}
\]

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

$$f(x) = \int_0^1 g(x,y) \, dy$$  \hspace{1cm} (38)$$

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

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

Then

$$\theta = \int_0^1 \int_0^1 g(x,y) \, dx \, dy$$  \hspace{1cm} (39)$$

The estimator

$$\bar{g} = \frac{1}{n} \sum_{i=1}^{n} g(\xi_{2i-1}, \xi_{2i}) = \frac{n^*}{n}$$  \hspace{1cm} (40)$$

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

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 \( a \leq x \leq b \)

\[
\int_{a}^{b} p(x)dx = 1 \tag{41}
\]

Then \( r = P(x) = \int_{a}^{x} p(y)dy \) \( \tag{42} \)

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

$$\theta = \int_0^1 f(x) dx = \int_0^1 \frac{f(x)}{g(x)} g(x) dx = \int_0^1 \frac{f(x)}{g(x)} dG(x)$$  \hspace{1cm} (43)

for any functions $g$ and $G$ satisfying

$$G(x) = \int_0^x g(y) dy$$ \hspace{1cm} (44)

where $g$ must be a positive valued function such that
\[ G(1) = \int_{0}^{1} g(y) dy \] (45)

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 ith 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:

$$1 = p_1 + p_2 + \cdots + p_M$$  \hspace{1cm} (48)

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

$$\Sigma'_t = w_1 \cdot \Sigma_1 + w_2 \cdot \Sigma_2 + \cdots + w_M \cdot \Sigma_M$$  \hspace{1cm} (49)

Similarly we have

$$1 = p'_1 + p'_2 + \cdots + p'_M$$  \hspace{1cm} (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. Coincidently, 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 losses 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 suppress 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

$$c_1 = \frac{p_1}{p'_1}, \quad c_2 = \frac{p_2}{p'_2}, \ldots, \quad c_M = \frac{p_M}{p'_M}$$

(51)
or \[ C_1 = \frac{p_i}{p_i^1} = \frac{\Sigma_i \cdot \Sigma_i^t}{\Sigma_t \cdot \Sigma_i^t \cdot w_i} = \frac{\Sigma_i^t}{\Sigma_t} \cdot \frac{1}{w_i} \] (51a)

If we let \( w_1 = \frac{1}{k_{e_1}} \), \( w_2 = w_3 = \ldots \) \( w_M = 1 \)

then from (51) we have

\[ C_1 = \frac{\Sigma_1 \cdot \Sigma_1^t}{\Sigma_t \cdot \Sigma_1^t \cdot w_1} = \frac{\Sigma_1^t}{\Sigma_t} \cdot \frac{1}{w_1}, \quad C_i = \frac{\Sigma_i^t}{\Sigma_t} = 1, \quad i \neq 1 \] (52)

let \( \Sigma_1 = \Sigma_{e_1} \) namely we introduce weight on elastic cross section only. Since \( \Sigma_{e_1}(E) \gg \Sigma_i(E), \quad i > 1, \quad \forall i \)

then

\[ \frac{\Sigma_t^i(E)}{\Sigma_t(E)} \approx \frac{\Sigma_{e_1}(E)}{\Sigma_{e_1}(E)} \quad w_1 = \frac{1}{k_{e_1}} \] (53)

and \[ C_1 \approx w_1 \frac{1}{w_1} = 1, \quad C_i \approx \frac{1}{k_{e_1}} \quad (i \neq 1) \] (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 $w_2 (w_2 = \frac{1}{k_2}, w_2 \leq 1)$. 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 p.ocesses 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 \( 1/2(t + t') \) will be an unbiased estimator of \( \theta \), and its variance is

\[
\text{Var}[\frac{1}{2}(t+t')] = \frac{1}{4}\text{var}(t) + \frac{1}{4}\text{var}(t') + \frac{1}{2}\text{cov}(t,t') \tag{54}
\]

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

\[
\text{var}(t) = \mathbb{E} [ (t-u)^2 ]
\]
\[
\text{cov}(t,t') = \mathbb{E} [ (t-u)(t'-u') ]
\]

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

It is possible to make \( \text{var}(t+t') \) smaller than \( \text{var}(t) \) by suitably selecting \( t' \). 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

\[
\frac{1}{2}(t+t') = \frac{1}{2} t(r) + \frac{1}{2} f(1-r) \tag{55}
\]
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 \( \pi \) \( (3.1415926) \).

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:

\[
 t_1 = \frac{1}{n} \sum_{i=1}^{n} \left\{ \alpha f(x_i) + (1-\alpha) f[\alpha+(1-\alpha)x_i] \right\} 
\]

\[
 = \frac{1}{n} \sum_{i=1}^{n} G_{\alpha} f(x_i) 
\]

\[
 t_2 = \frac{1}{n} \sum_{i=1}^{n} \left\{ \alpha f(x_i) + (1-\alpha) f[1-(1-\alpha)x_i] \right\} 
\]

\[
 = \frac{1}{n} \sum_{i=1}^{n} \tilde{G}_{\alpha} f(x_i) 
\]

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

\[
 f(\alpha) = (1-\alpha) f(1) + \alpha f(0) 
\]

An adequate rule of thumb is to choose \( \alpha \) by finding the root of

Another useful transformation is

\[
 t_3 = \frac{1}{m} \sum_{j=0}^{m} f\left(\frac{j+1}{m}\right) = \bigcup_{m} f(x) 
\]

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

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 \( d_{\text{col}}^2 \). 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

4. Follow residue particles to census and tally states

   last residue particle? No

   Yes

5. From state tallies (particle histories) calculate estimated parameter values

6. Renormalize and correct the ion density, reinitialize parameter values increment \( t^n \) to \( t^{n+1} \) time cycles and print the output quantities

7. 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) $K_1$, $K_2$, and $ED_1$: 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 $K_1$: the weight of particles in region I ($K_1 \geq 1$), $K_2$: the inverse of the weight $w_2$ of particles in region II where $w_2 = 1/K_2$ ($K_2 \geq 1$), $ED_1$: the energy dividing line, namely (0 to $ED_1$) is region I, ($ED_1$ to $E_0$) 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 $\Delta t$ of the time cycles. In terms of distance travelled by the particles, $\Delta t$ 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 ($P/p_0$ 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.

NAVP: Input constant to control the power of $\Delta t$, (or $\Delta D$) namely $\Delta t^\text{NAVP}$ 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
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 \((N_+)^\text{+}\), the number of ionizations \((NIZN)\), the number of escaped particles \((NES)\), the number of absorbed or recombined particles \((NAS)\), the number of super-elastic collisions \((NSE)\), the number of excitations to metastable levels \((NMET)\), and the number of excitations to other levels \((NEXC)\).

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
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:

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

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 \( \bar{E}_i \approx E_i \). 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 \( \bar{E}_i \) from \( E_i \) or \( \Delta E_i = \bar{E}_i - E_i \). 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 \( E_i, V_i \)) 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

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

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 it's 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 $E_f/p = 4, 5, 10 \text{ V cm}^{-1}$. ($E_f/p$: 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 $E_f/p$ values only $E_f/p = 5$ 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) ($\bar{E} = 6.4$)

b) present calculation ($\bar{E} = 6.51$)
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 ($\overline{E} = 7.7$)

b) present calculation ($\overline{E} = 7.83$)
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:

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

(63)

where  
\begin{align*}
\psi(E): & \text{ electron flux density} \\
f(E): & \text{ electron number density} \\
V(E): & \text{ Electron kinetic velocity}
\end{align*}
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 \( D = 1 \) cm, which is equal to the actual diameter of cylindrical tube used in the laboratory (7). The two pressures are chosen to be \( P = 10 \) torr and \( P = 760 \) torr, corresponding to a typical low-pressure case and a normal atmospheric pressure.

We have chosen two different source energies \( (E_0) \), one at \( E_0 = 1,000 \) eV and the other at \( E_0 = 70 \) 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 \( 10^{18} \) (#/cm\(^3\)-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:

$$\int_{0}^{E_{0}} f(E) \, dE = 1$$  \hspace{1cm} (64)

For the plasma system without electric field ($E_{f} = 0$), the distribution functions with source energy $E_{0} = 1000$ eV are shown in Figures 10 and 11 for the pressure $p = 10$ torr, and $p = 760$ torr respectively. Similar distribution functions for $E_{0} = 70$ 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 \( S_0 = 10^{22} \text{ particles/cm}^3\text{-sec} \) 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 \( E = 20 \text{ eV} \)) 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, $E_0 = 1000$ 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 $S_o = 10^{16}$, $E_o = 1000$ 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 \( S_o = 10^{16} \) \((#/\text{cm}^3\cdot\text{sec})\) 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., \( p = 760 \) torr), very few particles leak out of the system, and the loss are dominated by the absorption process. For lower pressure (\( p = 10 \) 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 \( S_o \), source energy \( E_o \) 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

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

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 a Electric Field

<table>
<thead>
<tr>
<th><em>P</em> (torr)</th>
<th><em>E</em>₀ (eV)</th>
<th><em>S</em>₀ (#/cm²·sec)</th>
<th><em>E</em> (eV)</th>
<th><em>N</em>⁺ (#)</th>
<th><em>W</em>-value</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<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

<table>
<thead>
<tr>
<th><em>E</em></th>
<th><em>S</em>₀</th>
<th><em>N</em>⁺</th>
<th><em>W</em>-value</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.05</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

$E_{\text{tab}}$: 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 $\bar{E}$ 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 appreciably.
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 \((S_0 = 10^{14}, S_0 = 10^{16}, S_0 = 10^{18})\). 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 \(0.24 \times 10^{15}, 0.24 \times 10^{14}, \text{ and } 0.24 \times 10^{13}\) 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 $S_0 = 10^{16}$ ($#/cm^3\cdot sec$).
Figure 22. The electron energy distribution for He, with an electric field at $S_o = 10^{16}$, $E_o = 1000$ ev.
Figure 23. The electron energy distributions for He, with an electric field at $S_0 = 10^{16}$, $E_0 = 70$ 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 ($E_z/p = 10$)

<table>
<thead>
<tr>
<th>$p$ (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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 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 ~ 49</td>
</tr>
</tbody>
</table>

Smit's distribution

<table>
<thead>
<tr>
<th>$p$</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></td>
<td>7.7</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

* $p$: pressure; $E$: source energy; $S$: source rate; $E_z$: average energy; $N^+$: number of positive ions.
The values of $N^1$ 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 ($E_0$, $S_0$, $p$) 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 addi-
tion 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 distri-
bution 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.
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 $E_f/p = 10$. 

$E$ (Energy in eV) 

Distribution Function $f(E)$ 

- $E_f/p = 0$ 
- $E_f/p = 1$ 
- $E_f/p = 10$
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
fellowing 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 Musha 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 \( 10^9 \). 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 propor-
tional 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 propor-
tional to $N_T$). We have to set an upper limit ($N_{\text{max}}$)
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 $N_Z$ secon-
dary 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:
a) \[ N_c + N_z + N_s \leq N_T \leq N_{\text{max}} \]  
(67a)

b) \[ N_z + N_s = N_{es} + N_{AB} \]  
(67b)

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

\[ N_s = N_{\text{max}} - N_c - N_z - JZ - 1 \]  
(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 including 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 $E_i$, $u_i$, $z_i$, $W_t$, $W_i$). 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.


C MASTER PROGRAM FOR MONTE CARLO SIMULATION OF NONLINEAR \#ALPHA PLASMA ELECTRON ENERGY DISTRIBUTION STUDY
C TICES INDUCED PLASMA ELECTRON ENERGY DISTRIBUTION STUDY
C DEFINITIONS AND TERMINOLOGY
C DI DIAMETER OF THE CYLINDER THE LENGTH OF THE FINITE REGION
C EQNITIAL ENERGY OF THE SOURCE PARTICLES (JOINT) SOURCE RATE
C NT TOTAL NUMBER OF IONICS X 3 TOTAL NUMBER OF ENERGY RANGES
C I(A) ARRAY FOR STORING THE INDEX OF ABSORBED PARTICLES NEXT THE NO. OF ION
C DistANCE TO BOUNDARY DOCl DISTANCE TO COLLISION DECENT DISTANCE TO CENSUS
C OZ! ZONE INTERVAL
C ARRAYS E. J. II STORING ENERGY, DISTANCE, AND DIRECTION OF PARTICLES
C NT TOTAL NUMBER OF PARTICLES TO BE PROCESSED EACH CYCLE
C NT TOTAL NUMBER OF TIME CYCLES
C M1 NUMBER OF SOURCE PARTICLES PER UNIT TIME DISTANCE ON SOURCE RATE
C NINI NUMBER OF IONIZATION PROCESSES OCCURRED EACH TIME CYCLE

REAL NE(50), NE(51), E(41), E(42), E(43), E(44), E(45), E(46)
DIMENSION P(151), P(152), P(153), P(154), P(155), P(156), P(157), P(158)
COMMON O. E(100), O(101), O(102), O(103), O(104), O(105), O(106), O(107)
COMMON X(1), X(2), X(3), X(4), X(5), X(6), X(7), X(8), X(9), X(10)
COMMON X(11), X(12), X(13), X(14), X(15), X(16), X(17), X(18), X(19), X(20)
COMMON X(21), X(22), X(23), X(24), X(25), X(26), X(27), X(28), X(29), X(30)
COMMON X(31), X(32), X(33), X(34), X(35), X(36), X(37), X(38), X(39), X(40)
COMMON X(41), X(42), X(43), X(44), X(45), X(46), X(47), X(48), X(49), X(50)
COMMON X(51), X(52), X(53), X(54), X(55), X(56), X(57), X(58), X(59), X(60)
COMMON X(61), X(62), X(63), X(64), X(65), X(66), X(67), X(68), X(69), X(70)
COMMON X(71), X(72), X(73), X(74), X(75), X(76), X(77), X(78), X(79), X(80)
COMMON X(81), X(82), X(83), X(84), X(85), X(86), X(87), X(88), X(89), X(90)
COMMON X(91), X(92), X(93), X(94), X(95), X(96), X(97), X(98), X(99), X(100)
COMMON X(101), X(102), X(103), X(104), X(105), X(106), X(107), X(108), X(109), X(110)
COMMON X(111), X(112), X(113), X(114), X(115), X(116), X(117), X(118), X(119), X(120)
COMMON X(121), X(122), X(123), X(124), X(125), X(126), X(127), X(128), X(129), X(130)
COMMON X(131), X(132), X(133), X(134), X(135), X(136), X(137), X(138), X(139), X(140)
COMMON X(141), X(142), X(143), X(144), X(145), X(146), X(147), X(148), X(149), X(150)
COMMON X(151), X(152), X(153), X(154), X(155), X(156), X(157), X(158), X(159), X(160)
COMMON X(161), X(162), X(163), X(164), X(165), X(166), X(167), X(168), X(169), X(170)
COMMON X(171), X(172), X(173), X(174), X(175), X(176), X(177), X(178), X(179), X(180)
COMM...
FORTRAN IV LEVEL 16

0033 CJ=0.0
0034 E1=2.6446
0035 IP=EDP+PDPO
0036 ES=21.4
0037 CI=2.240.0034944.003873
0038 ET= 0.8720328.1081 0.02 08
0039 VTE= 1.07/ETV
0040 CALL UNDER(4FP)
0041 PD2=PDPO+26.0
0042 MU=1.0E-09/ETV
0043 CMU=1.0E-09
0044 CMUPDP=CMUPDP
0045 IF IFPDPO.GT.10.00 CMUPDP=CMUPDP
0046 PD2CMUPDP/ETV
0047 ELS=1.0
0048 NUP=0
0049 NDN=0
0050 NDN=0
0051 NH=0
0052 NH=0
0053 NH=0
0054 NH=0
0055 NSE=0
0056 NSE=0
0057 NSE=0
0058 NSE=0
0059 NSE=0
0060 NSE=0
0061 NHET=0
0062 NHET=0
0063 NHET=0
0064 NHET=0
0065 NHET=0
0066 NHET=0
0067 NHET=0
0068 NHET=0
0069 WP=0.0
0070 WP=0.0
0071 WP=0.0
0072 WP=0.0
0073 WP=0.0
0074 WP=0.0
0075 WP=0.0
0076 WP=0.0
0077 WP=0.0
0078 WP=0.0
0079 WP=0.0
0080 WP=0.0
0081 WP=0.0
0082 JV=48
0083 JV=48
0084 J=48
0085 J=48
```fortran
      DATE = 72126
      0976 D10 = 1.0/DE1
      0977 CONST = 1.0/ALOG12.01
      0978 E02 = 0.46775
      0979 E03 = 19.6
      0980 PM = E03/ATMT
      0981 A(T) = 2.0*SQRTPT*PT4AKT
      0982 CALL KASTE(0.04,3,32,1)
      0983 SQ = RPOD*SR1.0,0*16
      0984 SP = 1.0
      0985 AITG = C(NP0)A(1)*AKT*E1
      0986 OE1 = 0.018
      0987 OE2 = 0.335
      0988 OE3 = 0.0
      0989 DI1 = 1.0/0.019
      0990 DI2 = 1.0/DE2
      0991 DI3 = 1.0/DE3
      0992 AH(1) = SQRT(1.021.0E8)*AKT*E1
      0993 11 OE(1) = 1.25*OE(3) - 1.0
      0994 DD = 1.0
      0995 E01 = 0.1
      0996 EQ1 = EQ(1)+E1 - 0.006
      0997 EQ11 = EQ(1)*E1 - 1.1
      0998 EQ(1) = SQR(1)*AQ(1)*AQ(1)
      0999 EQ(2) = SQ(1)*AQ(1)*AQ(1)
      1000 IF INFAPEN<1.0 GO TO 2
      1001 READ(101) EVG, DNG, THE, NTM, HT, NTP, MION, DDEG, HSW, SWF, E, U
      1002 READ(101) HT, MION, SWF
      1003 HTTPS
      1004 WGNHT = 1.0
      1005 RE = PM2, 10
      1006 NT = NT - 1
      1007 2 CONTINUE
      1008 IF INFAPEN<0.01 GO TO 3
      1009 READ(101) EVG, DNG, THE, NTM, HT, NTP, MION, DDEG, HSW, SWF, E, U
      1010 READ(101) HT, MION, SWF
      1011 HTTPS
      1012 WGNHT = 1.0
      1013 REWIND 9
      1014 NT = NT + 1
      1015 CONTINUE
      1016 IF (INFAPEN<0.0) GO TO 122
```
FORTRAN IV, G LEVEL 18

0139  NION= SORT(SO*10/AITG)
0140  ANGAM=NION+33333/DEN
0141  CALL SOURCE1NTP, EAVG, NTP, KM, PID
0142  ALMADAL(EAVG)
0143  DAV=AL
0144  DDEC=DAV+DEN/SORT(EAVG)
0145  SF=1.0
0146  IF (NION.NE.0) SER=MT/NION
0147  SN=50+AL*SF/ETV*SORT(EAVG)
0148  NSF=NS+0.5
0149  IF (SN.GT.ANSW) NSF=SN+0.5
0150  NSF=NSW*2
0151  SNP=NSW/SN
0152  SFL=SF
0153  SFL=SF2
0154  CALL SOURCEINS1NTP
0155  THE=0.0
0156  ANIOS=NION
0157  WGT=1.0
0158  122 CONTINUE
0159  '122 CONTINUE
0160  C GENERATE SOURCE PARTICLES TO INITIALIZE THE PROGRAM
0161  ANGAM=NION+33333/DEN
0162  WRITE(A6271)
0163  70 FORMAT(*,1X,PO,12X,LO,12X,MT,12X,MD,12X,PT,12X,MT,12X,MD)
0164  WRITE(A6271)
0165  71 FORMAT(*,12X,MD,12X,MD,12X,MD,12X,MD,12X,MD,12X,MD,12X,MD,12X,MD)
0166  WRITE(A6271)
0167  122 CONTINUE
0168  R=ANG2(N)
0169  122 CONTINUE
0170  C WITHIN THE OUTER I LOOP THE I TH PARTICLE HISTORY
0171  C GEOMETRIC ROBOTIC ENTERING HERE
0172  IF (IE11.LE.0.0) GO TO 9008
0173  IF (IM14.GT.0.1) GO TO 7778
0174  IF (IC12 .EQ.0) GO TO 609
0175  IF (IE11.GT.0.1) ETA=1.1
0176  IF (IE11.GT.0.1) ETA2=1.0
0177  SOR=SORT(IE11)
0178  OVA=EF*TCEN*EMASSI*0.5
0179  VZA=SQ2*ETV*U1+DVA
0180  VYS=(E11*ETV*U1*U1)*U1+DVA
0181  OCEM=SORTI1*2*VYS*TCEN
0182  NPCL=0
0183  900 IF (IE11.LE.0.0) GO TO 9008
0184  IF (INFLG .EQ.0) GO TO 809
0185  SOR=SORT(IE11)
0186  OVA=EF*TCEN*EMASSI*0.5
```
<table>
<thead>
<tr>
<th>FORTRAN IV G LEVEL</th>
<th>MAIN</th>
<th>DATE = 72126</th>
<th>05/31/702</th>
</tr>
</thead>
<tbody>
<tr>
<td>0187</td>
<td>809 D2=VIA+TCEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0188</td>
<td>ENEW = E11 + EP#DZ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0189</td>
<td>IF (ENW #L.#O, 0.01) GO TO 9009</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0190</td>
<td>SQEP = SQETEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0191</td>
<td>UNEW = SQEP(1)/SQEP + DVA2.* (SQEP*ETV1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0192</td>
<td>AL# = ALD(EI1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0193</td>
<td>AL# = ALD(ENEM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0194</td>
<td>AL# = 1.0/AL# + 1.0/AL#</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0195</td>
<td>AL# = 2.0/AL#</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0196</td>
<td>R# = RANS2(0)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0197</td>
<td>AL# = ARSIALOG(R11)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0198</td>
<td>DCOL# = ALR#ALD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0199</td>
<td>IF (ENL#G.EQ.11) GO TO 901</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0200</td>
<td>TFID=I(EI1, EQ.0) WRITE(6,75) DCOL,DCEN,E11,TTT,UTT,W(11)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0201</td>
<td>DCOL# = ARSIALOG(11-R11) AL#ALD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0202</td>
<td>ETOT=ETOT + DCOL*DCOL2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0203</td>
<td>IF (NDPM.LT.21) GO TO 901</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0204</td>
<td>TCEN#=TCEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0205</td>
<td>DCEN#=DCEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0206</td>
<td>E10=EI1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0207</td>
<td>UD# = U11*1USN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0208</td>
<td>Z10=EI1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0209</td>
<td>ENM# = UNEW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0210</td>
<td>UNM# = UNEW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0211</td>
<td>OZ# = OZ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0212</td>
<td>901 NFLG=1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0213</td>
<td>902 IF (DOCEN.LT, DOCOL) GO TO 905</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0214</td>
<td>DOCEN#=DOCOL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0215</td>
<td>E11=EI1 + EP#DC #</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0216</td>
<td>E11+Z11 + DC #</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0217</td>
<td>DOCEN = DOCEN ##</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0218</td>
<td>NCOL# = NCOL+1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0219</td>
<td>IF (MOD(NCOL, MC1.EQ.0) RIP=RAN32(0)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0220</td>
<td>TP=MOD(NCOL,NC1.MOD#RIP=1.0-RIP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0221</td>
<td>CALL CDLPDD11, TCEN, NTP, R#P, NUST, E9008, E9045</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0222</td>
<td>IF (EI11.LE.0.01) AND (LFG2.EQ.11) GO TO 9008</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0223</td>
<td>GO TO 900</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0224</td>
<td>905 E11=ENW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0225</td>
<td>U11= UNEW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0226</td>
<td>Z11= EI11 + OZ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0227</td>
<td>IF (Z111.LE.0.0) OR (Z111.GT.0) ) GO TO 9008</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0228</td>
<td>J=INT(Z111/0.1) J=J2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0229</td>
<td>IF (J=J2) J=J2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0230</td>
<td>J# = INDEXE(EI11)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0231</td>
<td>IF (KE.##EJ1 = J)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0232</td>
<td>IF (K#EQ.11) GO TO 906</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0233</td>
<td>ETAG2=1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0234</td>
<td>IF (EI11.GT.ED11) ETAG2=1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0235</td>
<td>IF (ETAG2)=ETAG211, GT.0, 0) GO TO 904</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0236</td>
<td>IF (ETAG2.GT.0.0) GO TO 903</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
FORTRAN IV G LEVEL 18

MAIN

DATE = 72126

0240  NUP=NUP+1
0241  ESE(NUP)=E(I)
0242  MT(II)=MT(II)/K3
0243  MSE(NUP)=MT(II)
0244  GO TO 904
0245  903 IF (LFG2.EQ.1) OR (NDP.LT.21) NDIN=NDIN+1
0246  IF (LFG2.EQ.1) OR (NDP.LT.21) RFIN=RFIN+1
0247  IF (MOD(NDN,K3).NE.0) GO TO 9045
0248  WT(II)=MT(II)/K2
0249  904 NEXJ,J= NEJ(N,J) + MT(II)
0250  IF (LFG2.EQ.1) OR (NDP.LT.21) NDIN=NDIN+1
0251  IF (EII.GT.0.01) ETEP=EII*WT(II)
0252  9045 IF (LFG2.EQ.1) OR (NDP.LT.21) GO TO 9009
0253  LFG2=1
0254  EII=EII+10
0255  NII=NII+10
0256  ZII=ZII+10
0257  TCN=TCN+2
0258  DCE=DCEN2
0259  DCOL=DCOL2
0260  ENEW=ENND
0261  UN=UN+1000
0262  D2=DO1
0263  9009 IF (IUSN.LT.01) GO TO 900
0264  GO TO 902
0265  9008 NEX=NEX+1
0266  IF (111.LT.601) WT(II)=MT(II)/K3
0267  IF (WT(II).LT.1.0) WT(II)=1.0
0268  WN1=WN1+MT(II)
0269  IF (EII.GT.0.01) ETEP=EII*WT(II)
0270  9071 9= 11
0271  LFG2=0
0272  IF (LFG2.NE.0) GO TO 7777
0273  IF (EII.GT.0.01) AND (111.LT.601) NOIN=NOIN+1
0274  IF (111.LT.601) NIN1+1
0275  IF (111.NE.0.01) GO TO 7778
0276  NII=111+11
0277  NII=111+11
0278  NII=111+11
0279  NII=111+11
0280  WN1=WN1+KELC
0281  WN1=WN1+KELC
0282  NSE=WNSE+0.5
0283  WN1=WNSE+0.5
0284  NSE=WNSE+0.5
0285  WN1=WNSE+0.5
0286  IF (EII.GT.0.01) AND (111.LT.601) GO TO 25
0287  N3=K3/2
0288  IF (111.LT.11) M3=1
0289  M3=K3-1
0290  IF (M3.GE.11) M3=M3
0291  IF (M3.GE.11) M3=M3
0292  GO TO 7777

05/31/67
```
<table>
<thead>
<tr>
<th>FORTRAN IV G Level</th>
<th>MAIN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>0293</td>
<td>DD 23 N=1, NUP</td>
</tr>
<tr>
<td>0294</td>
<td>K = KDOSE(E, E1)</td>
</tr>
<tr>
<td>0295</td>
<td>EE(K) = EE(K) + (K3-1)*ESE(N)*WSE(N)</td>
</tr>
<tr>
<td>0296</td>
<td>NB(E) = ME(K) + (K3-1)*ESE(N)</td>
</tr>
<tr>
<td>0297</td>
<td>22 IF (NK3 .GE. 21) IZ = IZ</td>
</tr>
<tr>
<td>0298</td>
<td>DO 21 = 1, IZ</td>
</tr>
<tr>
<td>0299</td>
<td>NREZ(L) = NREZ(L) + M*NSE(N)</td>
</tr>
<tr>
<td>0300</td>
<td>EZ(L) = EZ(L) + ESE(N)<em>M</em>NSE(N)</td>
</tr>
<tr>
<td>0301</td>
<td>NK3 = NK3 - 1</td>
</tr>
<tr>
<td>0302</td>
<td>IF (NK3 .LT. 21) IZ = IZ</td>
</tr>
<tr>
<td>0303</td>
<td>M3 = M3 + 1</td>
</tr>
<tr>
<td>0304</td>
<td>IF (NK3 .GT. 0) GO TO 22</td>
</tr>
<tr>
<td>0305</td>
<td>23 CONTINUE</td>
</tr>
<tr>
<td>0306</td>
<td>24 CONTINUE</td>
</tr>
<tr>
<td>0307</td>
<td>DO 192 N=1, JZ</td>
</tr>
<tr>
<td>0308</td>
<td>DO 1915 N=1, JZ</td>
</tr>
<tr>
<td>0309</td>
<td>NREZ(N) = NREZ(N) + NREZ(N)</td>
</tr>
<tr>
<td>0310</td>
<td>1915 EZ(N) = EZ(N) + EZ(N)</td>
</tr>
<tr>
<td>0311</td>
<td>192 CONTINUE</td>
</tr>
<tr>
<td>0312</td>
<td>NTPN = NTP + NTPN</td>
</tr>
<tr>
<td>0313</td>
<td>NTPW = NTPW</td>
</tr>
<tr>
<td>0314</td>
<td>ETOL = 0.0</td>
</tr>
<tr>
<td>0315</td>
<td>DO 14 M=1, JE</td>
</tr>
<tr>
<td>0316</td>
<td>ETOL = ETOL + EE(M)</td>
</tr>
<tr>
<td>0317</td>
<td>14 NTPW = NTPW + EE(M)</td>
</tr>
<tr>
<td>0318</td>
<td>EWTOL = EWTOL</td>
</tr>
<tr>
<td>0319</td>
<td>NTPF = NTPF</td>
</tr>
<tr>
<td>0320</td>
<td>DAVGO = DAVGO</td>
</tr>
<tr>
<td>0321</td>
<td>DAVS = DAVS + DAVS</td>
</tr>
<tr>
<td>0322</td>
<td>DSAVS = DSAVS + DAVS</td>
</tr>
<tr>
<td>0323</td>
<td>DTAVS = DTAVS + DAVS</td>
</tr>
<tr>
<td>0324</td>
<td>DTAVS = DTAVS + DAVS</td>
</tr>
<tr>
<td>0325</td>
<td>NTPO = NTPO + 1.0/NTPW</td>
</tr>
<tr>
<td>0326</td>
<td>NTPO = NTPO + NTPO</td>
</tr>
<tr>
<td>0327</td>
<td>NTP5 = NTP5 + NTPO</td>
</tr>
<tr>
<td>0328</td>
<td>NTP5 = NTP5 + NTPO</td>
</tr>
<tr>
<td>0329</td>
<td>EMAG = EMAG</td>
</tr>
<tr>
<td>0330</td>
<td>NSD = NSD</td>
</tr>
<tr>
<td>0331</td>
<td>NSD = NSD + H2 N0.5</td>
</tr>
<tr>
<td>0332</td>
<td>NTP = NTP + NTP</td>
</tr>
<tr>
<td>0333</td>
<td>NTPP = NTP</td>
</tr>
<tr>
<td>0334</td>
<td>DAVG = DAVG + DAVG</td>
</tr>
<tr>
<td>0335</td>
<td>EAVG = EAVG</td>
</tr>
<tr>
<td>0336</td>
<td>DAVG = DAVG</td>
</tr>
<tr>
<td>0337</td>
<td>TME = TME</td>
</tr>
<tr>
<td>0338</td>
<td>TMS = TMS</td>
</tr>
</tbody>
</table>

C IF SOURCE PARTICLES SOINSK NNS THEN ELIMINATE AS MANY RESIDUE PARTICLES AT TH |
| 0339             | NTP = NTP |
| 0340             | NTP |
| 0341             | QCN = QCN |
| 0342             | QDG = QDG |
| 0343             | BEG = BEG  |
```

<table>
<thead>
<tr>
<th>FORTRAN IV G LEVEL 18</th>
<th>MAIN</th>
<th>DATE = 72126</th>
<th>09/31/72</th>
</tr>
</thead>
<tbody>
<tr>
<td>0345</td>
<td>SF2=1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0346</td>
<td>IF LANIDS .NE.01, SEX=MTPS/ANIDS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0347</td>
<td>ESAVG=0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0348</td>
<td>IF (WNHS .NE.01), ESAVG = EEP/WHNS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0349</td>
<td>CASAVG=0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0350</td>
<td>IF (WNHS .NE.01), ESAVG = ETAS/WHNS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0351</td>
<td>SWPF=SWF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0352</td>
<td>CNS=CNS+SN+SN-DAVS*WGHIT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0353</td>
<td>CSF1= CSF1 +ETEP +ETAS<em>G0.0</em>WNZ12/1/2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0354</td>
<td>SF1=CSF1/(1.5G/1.5P+KI)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0355</td>
<td>CSF2 = CSF2 + SF2*DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0356</td>
<td>IF (MODINT.KI.EQ.01) .AND. (INT_.NE.01)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0357</td>
<td>SF3=CSF3*DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0358</td>
<td>WGHIT=SF1/SF3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0359</td>
<td>SN=SN*EPK                  IF (NTP.GT.420) ANSI=2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0360</td>
<td>NS=ANSW=0.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0361</td>
<td>IF (SN.EQ.0), ANSIW= NS+SN +0.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0362</td>
<td>SWF=SWF*ANSW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0363</td>
<td>NS=NSW +0.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0364</td>
<td>SF=SFME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0365</td>
<td>NNEW=WNMES +0.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0366</td>
<td>NNEW=WNEWS +0.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0367</td>
<td>WSFL=WSF<em>VLE+KVALUE</em>DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0368</td>
<td>CETEP=ETEP + ETAS*DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0369</td>
<td>ETAS=ETAS + ETAS*DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0370</td>
<td>CNS=CNS+SN+SN-DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0371</td>
<td>CNES=CNES+WNES+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0372</td>
<td>CHAS=CHAS+WNES+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0373</td>
<td>CHMET=CHMET+WNMET+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0374</td>
<td>CNEX=CNEX +WNEX+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0375</td>
<td>CESAVG=CESAVG + ESAVG*DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0376</td>
<td>CEGAS=CAS+FAAS+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0377</td>
<td>CROSS=CROSS+(UNIEN-WNESI)<em>DAVS/(1.5</em>SF3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0378</td>
<td>WVF=WMVLE+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0379</td>
<td>NSAVG=CNES+SN*DISG +0.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0380</td>
<td>NIZAVG=CNES+SN*DISG +10.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0381</td>
<td>ANIZVG=CN12*DISG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0382</td>
<td>ANSVG=CN12*DISG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0383</td>
<td>AETEP=ETEP*DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0384</td>
<td>AETAS=ETAS*DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0385</td>
<td>IF (LANIVG .NE.01) WVLVE=1.0+ANSVG-AETEP-AETAS/ANIZVG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0386</td>
<td>ALOSS=ALOSS<em>SN</em>DISG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0387</td>
<td>ANES=CNES+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0388</td>
<td>ANAS=CNAS+DAVS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0389</td>
<td>AESAVG=CESAVG*DISG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0390</td>
<td>ANMET=CNMET+DISG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0391</td>
<td>ANEX=CNEX+DISG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0392</td>
<td>ANSVG=CN12*DISG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0393</td>
<td>ANSVG=ANES+ANAS-ANIZVG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0394</td>
<td>IF (LAN .LE.00), ANSIW= SWF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0395</td>
<td>IF (WNIN .NE.00), WNVALUE = (SN*EO+WGHIT-ETEP-BTAS1/WNIN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0396</td>
<td>IF (LANIVG .NE.01) WVLVE=1.0+ANSVG-AETEP-AETAS/ANIZVG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0397</td>
<td>IF (NS.EQ.1) TMS=0.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1.33 FORTRAN IV G LEVEL
0398 NET=NET+NAS
0399 CALL SOURCE(MS,NTP)
0400 IF (NET.GT.01) CALL FILLOL(NTP,NET,NAS)
0401 90 NINT=NINT+NET
0402 IF (NUP .GT.01) CALL SEPROC(NTP, K3)
0403 IF (NONS.GT.01) CALL RRPROC(NTP, K3)
0404 NAST=NAST+NW
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/NSTP
0406 ANTPS=DSAVS/NTP5
0407 NSTP=NSTP
0408 DO 16 1,JZ
0409 CNREZIMI= CNREZI(M) + NREZI(M)*DAVS
0410 CEZIMI= CEZI(M) + EZI(M)*DAVS
0411 DN 1,JZ
0412 FNREZINI= NREZIMI/ANTPD
0413 SUMNE=0.0
0414 DO 18 M
0415 SUMNE= SUMNE + NEI(M)
0416 TNEI(M)= TNEI(M) + NEI(M)/EDEIMI
0417 193 PNEI(M)= PNEI(M)*ANTPD/EDEIMI
0418 EAVS=0.0
0419 DO 20 M
0420 EAVS= EAVS + E(M)*PNEI(M) + EDEIMI*DAVS
0421 1755 CNEIMI= TNEI(M)*NSTP
0422 CNION=CNWUPPD2.0*AIKY/SQRT(ANTI+1)
0423 ANION=0.0
0424 ANIOZ=0.0
0425 ANIO2=0.0
0426 DO 20 M
0427 ANIO2= ANIO2 + PMIEG(M)*CNEI(M)*EDEIMI
0428 ANION= ANION + PMIEG(M)*CNEI(M)*EDEIMI
0429 20 CONTINUE
0430 CNION=ANION+CNION
0431 DNIO2=ANI02*CNION
0432 CJO=ALOSS
0433 IF (ALOSS.LT.0.05) CJO=0.0
0434 IF (ABS(CJO).LT.10.0+50) CJO=10.0+50
0435 TNIO2=ABS(50+CJO)/DNIO2
0436 TNIO2=SQR(TNIO2)
0437 TNION=SQR(TNION)
0438 CNIO2=CNIO2+TNIO2*DAVS
0439 ANIOS=CNIO2*DI30
0440 WF=1.0
0441 IF (IANES+ANAS+NEI .LT. 100) WF=(ANIZVG+ANSVG)/IANES+ANAS
0442 IF (NODINT,NPM,NEI) GO TO 1790
0443 NION=ANIOS
0445 IF (WF.GT.0.005) AND (.WF.LT. 200.) NION=ANIOS*WF
0446 1790 CONTINUE
0447 DAVT=DAVEND
FORTRAN IV G LEVEL 18

MAIN DATE = 05/31/02
0448  ALPHA=MN1EK(NTPW/DAV1)
0449  ANEA=MNO10, 3333/DEN
0450  1789 FORMAT('0',ANION+,E13.4,3X,*TMION*,E13.4,3X,
0451    X*ALPHA*,F12.5,3X,*LEAK*,F13.4,3X/)
0452  WRITE(6,1789) ANION, DNO2, TMION, ALPHA, ALOSS
0453  171 FORMAT('0',*GAS SOURCE THICKNESS',F4.4,3X,'CM',
0454    *DEX',F11.4,3X,'CM',*DELT
0455    X+F11.4,3X)
0456  WRITE(6,171) ED, POPD, D, DED, DEDP
0457  180 FORMAT(° NUMBER OF TIME CYCLES',16,3X,'TIME ',E12.5,' SEC',
0458    13X,'GENERATED SOURCE PART*',16,3X,'MGTE # OF SOURCE PART*',1)
0459  115 IF (NE(M),EQ,0) GO TO 281
0460  WRITE(6,115) DAV1, DAVG, EAVG, EAVG, EAVG
0461  1805 FORMAT('0',*DAVG =',F12.5,3X,*DGEN =',F12.5,3X,*OVERALL EAVG ='
0462    E10,5,3X,*MGTE CHECK =',F10,5,3X)
0463  WRITE(6,101)
0464  181 FORMAT('0',1X,'NUMBER DISTRIBUTION',2X,'FLUX DENSITY',4X,'AVG ENERGY
0465    IE ENERGY',3X,'CUMULATED # DIST',3X,'ENERGY INTERVAL',2X,'CUM FLUX
0466    EDEN',4F,12.4,4X,'*DEN -DEV',4X)
0467  IF (NE(M),LE,0) DEAVG=0.0
0468  FNE=MNE(M)
0469  FLX1=FNE(M)*YFE
0470  IF (NE(M),EQ,0) GO TO 281
0471  IF (NE(M),EQ,0) IF (DAVG,EAVG,0.0)
0472  X/GAVG=GAVG*FNE(M)*5.0/EAVG
0473  IF (IDEAVG,EAVG,0.0)
0474  FLX2=CN11*Flx
0475  IF (IDEAVG,EAVG,0.0)
0476  IF (IDEAVG,EAVG,0.0)
0477  IF (IDEAVG,EAVG,0.0)
0478  IF (NE(M),EQ,0) GO TO 281
0479  IF (NE(M),EQ,0) IF (IDEAVG,EAVG,0.0)
0480  IF (NE(M),EQ,0) IF (IDEAVG,EAVG,0.0)
0481  IF (NE(M),EQ,0) IF (IDEAVG,EAVG,0.0)
0482  281 WRITE(6,187) NE(M), E12.5,5X,E12.4,3X,E12.5,4X,E12.5,4X,E12
0483    M,4X,E12.4)
0484  WRITE(6,188) NTNP, NS1N, NTPW, DET, DAVG, NCOL
0485  183 FORMAT('0',*TOTAL # OF PARTICLES PROCESSED',15,5X,'CUMULAT
0486    IE # OF PARTICLES',110,3X,'MGTE TOTAL # OF PARTICLES',*
0487    1,12,8,4X,OLD DAVG =',F12.5,4X,'# OF COLLISIONS',15,15
0488  WRITE(6,181)
0489  187 FORMAT('0',1X,'ZONE NUMBER',5X,'AVG ENERGY',5X,'NUMBER OF
0490    IF ELECTRONS',3X,'CUMULATED # DEN',16,3X,'CUM AVG ENERGY',3X
0491    1ZONE # DENSITY',1)
0492  DD23. M=kZ
0493  ANZ+CNN1(N)A=1050+1S
0494  PC1=CV1M+CNN1M
0495  282 WRITE(6,184) M, E12.5, N12M, ANZ, PN1Z1(M)
FORTRAN IV G LEVEL

0485 13A FORMAT(' ',5X,15,5X,PI.16,8X,PI.16,5X,PI.16,7X, 

0486 3E12.5)

0487 WRITE(6,185) NION,WHEN,WHN,ANTYP,WHN,WHN,WHN,WHN,WHN,WHN,

0488 XANAS,MTOP,K1,K2,MF,BF

0489 185 FORMAT(' NUMBER OF POSITIVE IONS=',1E16.6,' NUMBER OF TONIAX

0490 LIONS OCCURRED=',F9.3,5X,'ACTUAL #=',15,3X,'CUMULATIVE AVG=',F10.3,

0491 X/,' # OF ESCAPED PARTICLES=',F9.3,5X,'ACTUAL #=',15,3X,'CUMULATIVE

0492 X AVG=',F9.2,15,3X,'CUMULATIVE X AVG=',F9.2,15,3X,'TOTAL # OF

0493 PARTICLES IN THE NEW CYCLE=',15,3X,'KINT=12,3X,'KRN=12,'12,

0494 4X,' BALANCE FACTOR WF=',F10.5,3X,'DF2=+',F10.5,/)

0488 WRITE(6,188) ED1,NPM,NO,KELC

0489 188 FORMAT(' ENERGY DIVIDING LINE FOR NA PLAY =',15,4X,'EVERY=',12,

0490 X,'TH CYCLE CHANGE N=+',12,15X,' TIME INTERVAL MULTIPLY FACTOR=',F9.4,

0491 X3X,'ELASTIC XSEC / BY=',12,15X)

0492 WRITE(6,189) HSO,ANSV,THOR,HSE,WNAV,WHN,WHN,WHN,WHN,WHN,

0493 XHGT

0494 189 FORMAT(' SOURCES PARTICLES GENERATED OLD =+',110,15X,' SOURCES

0495 PARTICLES (WEIGHTED) IN THE NEW CYCLE =+',110,3X,' CUMUL.

0496 XIVE AVG=',F10.3,'X=',F10.3,15X,' # OF PARTICLES KILLED OFF IN NA

0497 PROC='+',110,15X,' # OF SUPER ELASTIC COLLISION=',F10.4,'/',

0498 15X,' # OF EXCITATIONS TO META STABLE LEVEL=',

0499 X,F10.3,3X,'CUMULATIVE AVG=',F9.3,'1

0500 //,' # OF EXCITATIONS TO OTHER LEVELS=',F10.4,'X=',F10.3,15X,' CUMUL.

0501 XIVE AVG=',F10.3,'X=',F10.3,15X,' # OF PARTICLES CROSS OVER TO

0502 HIGH ENERGY REGION NWP=',110,3X,'OVERALL WT=',E12.4,15X)

0493 WRITE(6,190) EF,EDP,POPO,FSF1,FSF2,FSF3

0494 190 FORMAT(' THE SOURCE RATE=',E11.3,' PER SEC=',4X,'W-VALUE=',

0495 E12.4,3X,'W-VALUE=',E12.4,3X,'CUM AVG WV1=',F10.3,4X,'CUM AVG

0496 WV2+',F10.3,4X,'AVERAGE ENERGY OF ESCAPED PARTICLES=',F12.3,

0497 X,'CUMULATIVE AVG=',F11.4)

0498 CKFNE=SUMNE+ANTPO

0499 WRITE(6,191) CKFNE,CESUM,NION,NTR

0500 191 FORMAT(' THE SN # CHECK =+',F12.4,3X,' THE CN # CHECK=',F12.4,15X,

0501 4X,'CALCULATED N= USE F=',E14.6,3X,'CALC N= USE CN =+',E14.6,3X,'AR

0502 XXF,CN=+',E14.6,15X)

0503 IF (MODINT.51,NE.01.OR.(MODINT.101,EQ.01)) GO TO 3

0504 WRITE(9) SAVG,DANG,THE,NWH,NT,MT,MTP,NION,DFEG,NSM,SMF,E,U

0505 WRITE(9) 2,MT,NWH,FSF3

0506 IF (MODINT.101,NE.01) GO TO 4

0507 WRITE(10) SAVG,DANG,THE,NWH,NT,MT,MTP,NION,DFEG,NSM,SMF,E,U

0508 WRITE(10) 2,MT,NWH,FSF3
5099  ENDFILE IO
5100  REWIND IO
5101  4 CONTINUE
5102  IF (MODINT.NCUM1.NE.0) GO TO 6666
5113  NSTP=0
5114  NSTP=0
5115  TAVS=0.0
5116  NAST=0
5117  NINT=0
5118  WAVE=0.0
5119  CN512=0.0
5120  CNSS=0.0
5121  CNMET=0.0
5122  CNE=0.0
5123  CESAVG=0.0
5124  CEASG=0.0
5125  CSF2=0.0
5126  CSEX=0.0
5127  CNI2=0.0
5128  CSF1=0.0
5129  DO 54 M=1,JZ
5130  CNR(M)=0.0
5131  94 CEZ(M)=0.0
5132  DO 55 M=1,JZ
5133  TNE(M)=0.0
5134  CSN(M)=0.0
5135  CNE=0.0
5136  CNS=0.0
5137  DTAVS=0.0
5138  CETE=0.0
5139  CETAS=0.0
5140  IF (MODINT.101.NE.0) GO TO 6666
5141  6666 CONTINUE
5142  NT= NT + 1
5143  DO 56 M=1,500
5144  DZ(JM)=0.0
5145  IES(JM)=0
5146  IAK=JM
5147  56 IAS(JM)=0
5148  DO 51 NM=1,50
5149  WSENM=0.0
5150  51 ISEM(JM)=0.0
5151  1!
5152  C. REINITIALIZE THE OUTPUT QUANTITIES
5153  DO 52 NM=1,JE
5154  FCN(M)=0.0
5155  52 NE(M)= 0.0
5156  DO 53 NM=1,JE
5157  NREZ(M)= 0.0
5158  53 NEZ(M)= 0.0
5159  DO 59 NM=1,JE
5160  59 NEZ(M)= 0.0
FORTRAN IV  O  LEVEL 18

MAIN

DATE = 04/12/76  05/31/02

0561  NIZIN=0
0562  NOIN=0
0563  NIN=0
0564  NES=0
0565  NSE=0
0566  NCOL=0
0567  NAS=0
0568  NASW=0
0569  DXT=0.0
0570  NNET=0
0571  NEE=0
0572  WNES=0.0
0573  WNAS=0.0
0574  WNESX=0.0
0575  WNES=0.0
0576  WNESX=0.0
0577  WNES=0.0
0578  ETA5=0.0
0579  ETAP=0.0
0580  ETOT=0.0
0581  NEK=0
0582  NUP=0
0583  NONIN=0
0584  NONX=0
0585  INV=NINV+1
0586  CALL RNINZ1NINV
0587  WRITE(6,721)
0588  IF (INT(LE,NINI)) GO TO 7777
0589  STOP
0590  DEBUG SUBCHR
0591  END
FUNCTION INDEXE(E)
COMMON /AREA/, CONST, DELT, ET, DEL, DELT, D12, D13, D14
IF (E.LT.0.08) GO TO 11
IF (E.GT.0.159 .1599) GO TO 13
INDEXE = 5*(ALOG10I)+CONST +0.01*D12
RETURN
11 INDEXE = 1 + END
12 RETURN
13 INDEXE = 36 + (ALOG10I)*CONST + 0.09799073
RETURN
END
IF (NI(1).LT.1) GO TO 300
J=NI(1)+IE-1
DO 301 J=IE, JI, 2
R=TRANS(I)
RZ=1.0-R
E(1J)= ERD(IJ) + EDE(IJ)*R
E(1J)= FRO(IJ) + ENE(IJ)*R
WT(J)=WT(I)
WT(J+1)=WT(I)
EAG=EAG + E(1J) *E(J+1)
IF(MUD(JJ-1),P.EQ.0, AND,(J,GE,J)) EAG=EAG=E(J+1)
CONTINUE
IE=IE+NI(I)
300 CONTINUE
NSO=NSO
EAVG=EAVG/NSO
IF (E(J).LE.EDII) WM I-K = MTII
IF (E(J).GT.EDII ) MTI=MNZ*WTII
DO 70 IF M 1 1.LE.O.CI E11)-E(J)-11
FTM=FTM + E(M)*WTII
70 CONTINUE
EAVG=ETM/NTMZ
WRITE(6,507) EAVG, EAVG
QD=QD+0.25
76 I=1
77 RZ=TRANS(I)
78 RU=TRANS(I)
79 Z(I)=0*RZ
80 UI(I)=2.0*RZ-1.0
81 I= I+1
82 Z(I)=0*RZ
83 UI(I)=2.0*RZ
84 I= I+1
85 Z(I)=0*(1.0-RZ)
86 UI(I)=2.0*RZ-1.0
87 I= I+1
88 Z(I)=0*(1.0-RZ)
89 UI(I)=1.0-2*RZ
90 I= I+1
91 IF((IZ-1).GE.0) GO TO 101
92 IF ((IZ,LT,1)) GO TO 105
93 RZ=TRANS(I)
94 RU=TRANS(I)
95 IF((IZ-1).EQ.2) GO TO 102
96 IF((IZ-.LT.1).EQ.1) GO TO 103
97 IF((IZ -1).EQ.0) GO TO 104
105 RETURN
SUBROUTINE SOURCE(NS,NTP)

COMMON D, E(S100), I(S100), U(S100), AS(500), IRR(500), NRR, IES(4000)
 X, WT(100), SWF
COMMON /AREA/ C1, MUO, E1, EIS, SF, AREA, PDPO, P1, EQ, NION, IE, IL, NASW1,
 INSEW, AKT, AKT1, AM1, AMCl, EDI, PDMUO, PD28, K1, K2, K3, NM2, ED2, ED3

REAL MUO, NION

C SOURCE PARTICLE PARAMETERS GENERATING PROCEDURE
C II: THE INITIAL INDEX VALUE FOR THE SOURCE PARTICLE

I=NTP+1
DO 100 K=1, NS, 2
E(1)=0
RZ=RAN3Z(0)
RU=RAN3Z(0)
Z(1)=D*RZ
U(I)=RU+RU-1.0
WT(I)=MN2/SWF
IF ((NS-K).LE.3) GO TO 100
E(1)=EC
Z(1)=RZ*0
U(I)=U(I)
WT(I)=MN2/SWF
I=I+2
100 CONTINUE
NTP=NTP+NS
105 RETURN
END
SUBROUTINE FILHOLE(NTP, NES, NAS)

COMMON D, E(51001), Z(51001), U(51001), IAS(5001), IRR(5001), NRR, IES(6001)

X, WT(51001), SWF

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)

U(J)=U(K)

WT(J)=WT(K)

K=K-1

100 K=NTP-NES

105 IF (NAS.LE.0) RETURN

K=NTP

DO 300 I=1, NAS

J=IAS(I)

E(J)=E(K)

Z(J)=Z(K)

U(J)=U(K)

WT(J)=WT(K)

300 K=K-1

NTP=NTP-NAS

200 RETURN

END
SUBROUTINE RRPROC(NTP,K3)

COMMON D, E(5100), Z(5100), U(5100), IAS(500), IRR(850), NKR, IES(600)
X, WT(5100), SMF

COMMON /A6/ ESE(50), NUP, NDNR, NDNM, 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

300 RETURN

END
FUNCTION ALDA(E)

COMMON /AREA2/ CL, MUO, EI, EIS, SF, ANEM, PDPO, PI, EO, NION, JE, JZ, NASW1,
INSE, EKT, AIXT, AMI, AMCL, ED1, PDUN, PD6, K1, K2, K3, MW2, EO2, ED3

COMMON /AREA3/ XSEL, XELC, XRCB, XMET, XEXC, XION, XTOL, XELC

REAL MUO, NION

WELC = 1.0 / KELC

IF (E.LE.5.) XELC = 23.0 * PDPO

IF (E.GT.5. AND. (E.LE.19.8)) XELC = 23.0 * PDPO * SQRT(15. / E)

IF (E.GT.19.8) XELC = EXP(-0.02 * E) * 16.76 * PDPO

XELC = XELC * WELC

IF (E.LE.19.8) GO TO 5001

4001 AE1 = 0.25 * (E - 19.8) ** 2

XMET = 0.109 * (E - 19.8) / AE1 ** 0.0311

XMET = XMET * PDPO

E19 = E - 21.4

AE2 = 1850. + E19 ** 2

XEXC = 68.80 * E19 * PDPO / AE2

IF (E.LT.21.4) XEXC = 0.0

AE3 = 4900. + E24 ** 2

XION = 172. + E24 / AE3

XION = XION * PDPO

IF (E.LT.24.46) XION = 0.0

XTOL = XMET + XELC + XION + XMET

XRCB = 0.0

XMET = 0.0

XTOL = XTOL + XION

6001 ALDA = 1.0 / XTOL

RETURN

END
SUBROUTINE COLPRO1, D, X, W, H, S

COMMON D, E(S100), z(S100), u(S100), t(E3(S50)), j(S100), h(E3(S50))

COMMON /AREA1/ NIZN, NOZM, NAS, DIZN(S100), NSE, NHE, NEXC, ETAS

COMMON /AREA2/ CT, MT, EI1, E15, SF, APAA, PPQ, P0, EQ, E10, IE1, JO, J2, JASM

COMMON /AREA3/ SEL, XELC, XRCB, XMET, XEXC, XION, XTOL, KELC

COMMON /AREA43/ IS, KU, E, EM, S, F, NEAM, PI, EGT, NMA, NMA

COMMON /AREA5/ ESE, S0, NUP, NDAR, W, SE

DIMENSION PX(4)

REAL MU0, %, NION

INTEGER KIND OP COLLISION

REAL MOD, NION

C DECIDE WHICH KIND OF COLLISION

C, ENERGY REGION 1 19.03EV TO 1 KEV

DO 45 IF(AIP.GT.PK(X1, XI) IP=IX41

IF (AIP.LE.10.80 RETURN

C INELASTIC COLLISION PROCESS EXCITE TO META STABLE STATE

C EXCITATION TO METHASTABLE LEVEL

C, EXCITATION TO OTHER LEVELS

C IONIZATION PROCESS PRODUCE SECONDARY ELECTRONS

END
C IONIZATION PRODUCED PARTICLES ALWAYS PUT TO THE END OF RESIDUES

C045 DISSOCIATION = OGEN
C046 R1=RAN3(Z10)
C047 U1=I+P1-1.0
C048 NNTP =NTP +NIZN
C049 E(NNTP)= E(I)+0.1
C050 Z(NNTP)=Z(I)
C051 UNNTP)=U(I)
C052 E(I)= E(I)+0.9
C053 WTNTP)= WTI(I)
C054 IF ( (E(NNTP) ,GT.E01) OR ( I(I) *1.111*E(I).LT.FB1)) GO TO 4018
C055 NON=NON+1
C056 NN=NN+1
C057 W(T(NNTP)=W(T(I)
C058 IF (NN==0) =NTP+NIZN
C059 CONTINUE
C060 RETURN

C ENERGY REGION 2 LOW ENERGY RANGE DOWN TO THERMAL REGION
C061 5000 P00= XELC*XTI
C062 PSEL= XSEL*XTI
C063 PRCB= XRCB*XTI
C064 PX1= PRCB
C065 PX2= PSEL +PRCB
C066 IP=1
C067 GO TO 55
C068 55 IF(I(I).GT.PX111) IP=IP+1
C069 GO TO (5013,5015,5011,50)

C ELASTIC COLLISION PROCESS
C070 5011 IF (ELE.I.EQ.0.0) GO TO 50125
C071 IF (LINUS.T.EQ.0.1) OR (E(I).GT.E12.51) GO TO .0 50124
C072 50115 R=RAN3(Z10)
C073 R2=RAN3(Z10)
C074 E1=ED2*RE
C075 E2=ED2*RE
C076 FLX=PM3(Z1)
C077 FR2=PM3(E2)
C078 IF ( (E.R.E2) RS=RE
C079 IF ( (E.R.E2) RS=R2
C080 IF ( (E.R.E2) OR. (R2.LE.FR1)) GO TO 5012
C081 GO TO 50115
C082 50122 ENN=RS*ED2
C083 50124 IF (LINUS.T.EQ.0.1) OR (RE.E2.5)) ENN=0.0
C084 R1=RAN3(Z10)
C085 U1=2.0*R1-1.0
C086 U=SQRT((1.-U1**2)*(1.-U1)**2))
C087 U=2.*RE-1.0
C088 COST=UM(I) +SIGNUM.US
C089 DELE=C1+(E(I)-ENN)*1.0-COST*ELE
C090 E(I)= E(I) -DELE
C091 U1=U1
C092 50125 IF (ELE.I.EQ.0.0) U11= 2.0*RAN3(Z10) -1.0
C093 RETURN

C RECOMBINATION PROCESS
FORTRAN IV G LEVEL 18

0094 5013 NAS = NAS + 1
0095 IAS = NAS + 1
0096 WNAS = WNAS + WT(I)
0097 IF (E(I).GT.0.0) ETAS = ETAS + E(I)*MT(I)
0098 RETURN2

C SUPERELASTIC COLLISION PROCESS

0100 5015 E(I) = E(I) + EIS
0101 WNSE = WNSE + WT(I)
0102 NSE = NSE + 1
0103 RP = RAN3(101)
0104 UI(I) = RP * RP - 1.0
0105 RETURN
0106 END
FORTRAN IV  G  LEVEL 19

MAIN

```
0033 C1=2.*3.010649/4.003873
0034 ETV=SORT12,U+1.02037/4.1081*1.06*08
0035 VTV= 1.0/(ETV*ETV)
0036 CALL UNHER1(*OFF*)
0037 PTV6=PVPU+25.0
0038 MUC=(1.PF<91)/ETV
0039 CHU0=1.0*09
0040 CHUPV=CMU1*PDV
0041 IF (PDV<GT,1.0*1) CMUPV=CMU0*10.0
0042 PDNU=CMUPV/ETV
0043 CLU5='r'
0044 NU=0
0045 NWN=0
0046 NNP=0
0047 IM=0
0048 NK=0
0049 MA=0
0050 NES=0
0051 NSE=0
0052 NQIN=0
0053 NI=0
0054 NIV=0
0055 NKU=0
0056 NINV=0999999
0057 NME=0
0058 NEX=0
0059 ETA5=0.0
0060 ETEP=0.0
0061 NIP5=0
0062 NW=0
0063 NIP=0
0064 DTAS=0.0
0065 CSFZ=0.0
0066 CNO2=0.0
0067 CSE1=0.0
0068 CETAS=0.0
0069 CETE=0.0
0070 NCOL=0
0071 WNM1=0.0
0072 WNSA=0.0
0073 WNXS=0.0
0074 WNS=0.0
0075 WN=0.0
0076 WNZ=0.0
0077 ND=0.0
0078 TE1=JE
0079 DE=3.56E+16*PDQ.
0080 DETZ=D/JZ
0081 DINV= 1.0/DE
0082 CZSW=1.0/ALOG12.01
0083 ED2=0.467543
0084 ED3=19.0
0085 PH=ED2*24.0
```
FORTRAN IV G LEVEL

MAIN

DATE = 72123

01/28/29

0086 AM1 = 2, * (AKT/SORTPI*AKT)
0087 CALL EM3INT (4930491211)
0088 SO = POP*SR1, 0E+16
0089 AKT = CMUPD / (AKT*PI)
0090 DE = 0, 0
0091 DE2 = 3, 55
0092 DE3 = 0, 5
0093 DE4 = 1, 0, 0
0094 UI2 = 1, / DE2
0095 UI2 = 1, / DE3
0096 AM1 = SORT12.0E+2, 71R3 * AKT * ED2
0097 DO 11 I = 6, 36
0098 ERL = (1-3)*DE2 - 4, 0
0099 EMV = ERL - 1, 1685
0100 FROD = 2, 0**ERL
0101 FG = (1-1) = 7, 0**EMV
0102 IF (FG(1-1) = (2, 0**DE2 - 1, 0)
0103 JE1 = JE1
0104 DO 12 I = 37, JE1
0105 FRL = 7, 005 + (1-16)*DE3
0106 EMV = ERL - 1, 1405
0107 FROD = 2, 0**ERL
0108 FG = (1-1) = 2, 0**EMV
0109 12 FDE = ED1(11) = ED1(11)*12, 0**DE3 - 1, 0
0110 DO 10 I = 1, 5
0111 FDE(1) = 0, 016
0112 EG(1) = ED1(11) = 0, 008
0113 13 FROD = ED1(11) = 11
0114 FDE(3) = 29, 45
0115 EG(4) = 1737, 682
0116 IF (INTAPE.NE.1) GO TO 2
0117 READ(11) EAVG, DAVG, THE, NTW, NT, NTP, NION, DDEG, NSW, SWP, E, U
0118 READ(11) Z, WT, ANIOS, SF3
0119 NS = NSW + K2
0120 REST = 10
0121 NT = NT + 1
0122 2 CONTINUE
0123 IF (INTAPE.NE.0) GO TO 5
0124 READ(19) EAVG, DAVG, THE, NTW, NT, NTP, NION, DDEG, NSW, SWP, F, U
0125 READ(19) Z, WT, ANIOS, SF3
0126 NS = NSW + K2
0127 REST = 9
0128 XX = 11
0129 5 CONTINUE
0130 IF (INTAPE.NE.0) GO TO 122
0131 NION = SORT (50*10/AKITG)
0132 ANEAM = NION + 0, 53333/DEM
0133 CALL SOURCE (1TP, EAVG, NTW, PH, FID)
0134 AL = 1DAE (EAVG)
0135 DAVG = AAL
0136 DDEG = DAVG**NT/SORT (EAVG)
0137 SF2 = 1, 0
0138 IF (NION.NE.0) SF2 = NTM/HION.
0139      SN=SN+ALND*SFZ/(ETV*SORT(EAVG))
0140      NSM=NSM*0.5
0141      IF (SN,GT,ANSW) NSM=SN.
0142      SWF=NSM/SN
0143      SF=SWF
0145      ANOS=NCH
0146      CALL SOURCE(NS,NTP)
0147      SF1=SFZ
0148      SF3=SF2
0149      THE=0.0
0150      122 CONTINUE
0151
0152      C GENERATE SOURCE PARTICLES TO INITIALIZE THE PROGRAM
0153      ANEQ=NION=0.33333/DEH
0154      70 FORMAT(15*F1.7)
0155      WRITE(6,71) AITG, NION, NTP, AL, EAVG
0156      71 FORMAT(15*F1.7)
0157      IF (ETG,GT,1.0) EAVG=*, F14.6,3X, EAVG=*, F14.6,3X
0158      72 FORMAT(15*F1.7)
0159      WRITE(6,72) HPA, NSW
0160      762 FORMAT(15*F1.7)
0161      C WITHIN THE OUTER LOOP THE I TH PARTICLE HISTORY
0162      C GEOMETRIC ROUTINE ENTERING HERE
0163      7727 IF (E(11).GT.0.1) .AND. (.1.GT.NTP) DCEN= DIKN(NOWN+1)
0164      7776 IF (E(11).GT.0.1) .AND. (.1.GT.NTP) DIKN= DIKN(NOWN+1)
0165      NFLEG=0
0166      UIW=UIU(L)*IUSN
0167      LG2=0
0168      ETAGM=0
0169      30000 IF (E(11).GT.0.1) ETAGM=1.0
0170      30001 IF (E(11).GT.0.1) ETAGM=1.0
0171      JINT=(I11)=DIZV+1
0172      1032 IF (NFTG+NZD) ETAGM=0.0
0173      1033 IF (NFTG+NZD) ETAGM=1.0
0174      IF (NFLEG .EQ. 1) GO TO 809
0175      IF (NFTG+NZD) ETAGM=(DCOL*DCOL+CF1) .EQ. 0.0
0176      ETOT=ETOT+DCT(D13/11)
0177      ETAGM=ETAGM+1.0
0178      IF (NFLEG .EQ. 1) GO TO 809
0179      IF (NFTG+NZD) ETAGM=0.0
0180      DCEN=DCEN
0181      1032 IF (NFTG+NZD) ETAGM=0.0
0182      DCEN=DCEN
0183      1032 IF (NFTG+NZD) ETAGM=0.0
0184      809 NFLEG=1
0185      800 IF (DCEN.LT.NDCOL) GO TO 900
0186      900
FORTAN IV LEVEL 1B  MAIN  DATE = 72/125  01/28/79

0240   WNSE = WNSE/KELC
0241   NSE = WNSE * 0.9
0242   DD 191  NSE*,JE
0243   DD 191  NSE*,JE
0244   EE(M) = EE(M) + EKZ(M,N)
0245   191  NE(M) = NE(M) + NEZ(M,N)
0246   IF ((K3 = SQ1) .OR. (HUP20SQ1)) GO TO 24
0247   M3 = K3/JJ
0248   IF (N3.LT.11) M3 = 1
0249   N3 = K3 - 1
0250   IF (N3.LT.11) JZ = N3
0251   IF ((N3.GE.11) .OR. N3 = N7) GO TO 22
0252   M4 = 23  - 1/NUP
0253   K = NOFKE(ESE(N))
0254   EE(K) = EE(K) + (K3-1) + ESE(N) + WSE(N)
0255   NE(K) = NE(K) + (K3-1) + WSE(N)
0256   IF (N3.GE.JJ) JZ = JZ
0257   DD 21  L41,JJ
0258   IF (K3.LT.JJ) JZ = JZ
0259   NWZ = KJ + ESE(N)
0260   NWZ = KJ + ESE(N)
0261   IF (NK3.LT.JJ) JZ = N3
0262   M4 = 1
0263   IF (N3.GT.JJ) GO TO 22
0264   CONTINUE
0265   CONTINUE
0266   M4 = 192  - 1/JJ
0267   DD 191  N3*,JE
0268   WFF2(P) = WFF2(M) + NEZ(M,N)
0269   1913  E(1) = E(1) + F7(J,J,M)
0270   CONTINUE
0271   NTMI = NT*NIJN - NSE+NAS
0272   NTPH = P, J
0273   NTPH = P, J
0274   DD 14  NP = JF
0275   +TCL = -TCL +60(N)
0276   14  NT = NT*NEK + NE(M)
0277   ETUL + ETCL
0278   NTP = NTP
0279   NAVG = NAVG*4
0280   DAVS = DAVS + DAVS1000.1**NTMP
0281   NTP = NTP + DAVS
0282   DIAG1 = 2/DAVS
0283   NTP = 100/NTMP
0284   NTP = NTP + NTMP + CT.5
0285   DAVS = DAVS + DAVS
0286   NTP = NTP + NTMP + DAVS
0287   NTP = NTP
0288   EWAVG + ETCL + NTP
0289   M304 = NAS
0290   M304 = NAS + M304 + CT.5
0291   NTP = NTP + NIH
0292   NTP = NTP
<table>
<thead>
<tr>
<th>FORTRAN IV LEVEL</th>
<th>MAIN</th>
<th>DATE = 72125</th>
<th>01/28/29</th>
</tr>
</thead>
<tbody>
<tr>
<td>0293</td>
<td>$DAVG + DTOT / 0.9 / NTP$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0294</td>
<td>$EAVG * ETOT / NTP$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0295</td>
<td>$DEG * DEG / ETV$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0296</td>
<td>$TMF = TMF + DEG$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0297</td>
<td>$TMS = TMS + DEG$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>C IF SUU NGE R TICLES S0(NS)&lt; NES THEN ELIMINATE AS MANY RESIDUE PARTICLES AT TH</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0298</td>
<td>$NTP = NTP + NMS + NAS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0299</td>
<td>$DCEINIT = ALODA = EAVG$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0300</td>
<td>$NT = NTPH + 5$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0301</td>
<td>$DDEG = DAVG + ND / SQRT(EAVG)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0302</td>
<td>$SF = 1.0$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0303</td>
<td>$IF (ANIOS, NE, 0) SF2 = NTPS / ANIOS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0304</td>
<td>$EAVG = 0.0$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0305</td>
<td>$IF (WNES, NE, 0.1) EAVG = EYEP / WNES$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0306</td>
<td>$CASFVG = 0.0$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0307</td>
<td>$IF (WNAS, NE, 0.1) EASAVG = ETAS / WNAS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0308</td>
<td>$SF = SF / W$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0309</td>
<td>$IF (WNES + WNAS, GT, 0.0), RET = (WNES + SN) / (WNES + WNAS)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0310</td>
<td>$CN55 = CN55 * SN / DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0311</td>
<td>$CSF2 = CSF2 + (ETEP * ETAS + 40 * WNAS) / TMS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0312</td>
<td>$IF (MOD(INT, 51) = 0.0) AND (INT, NE, 0) SFI = CSF2 / 150 + E0 + 0.1$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0313</td>
<td>$CSF1 = CSF1 + SF2 * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0314</td>
<td>$IF (INT, NE, 01.0) AND (INT, NE, 01.0, (SF3, LT, SF1 + 0.11)) SF3 = ASORT (SF1, SF3)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0315</td>
<td>$WGHT = SF1 / SF3$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0316</td>
<td>$SN = SJ + DEG * SF3 * WGHT / ETV$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0317</td>
<td>$NSW = ANSW + 0.5$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0318</td>
<td>$IF (SN, GT, ANSW) NSM = SN + 0.5$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0319</td>
<td>$SF = NSV / 34$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0320</td>
<td>$NS = NSM * K3$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0321</td>
<td>$CSF = SF$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0322</td>
<td>$RF = RF / RF + RF + 0.5$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0323</td>
<td>$RF = RF / RF + 0.5$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0324</td>
<td>$WSVE = WSVE + WVALUE * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0325</td>
<td>$CETEP = CETEP + FTEP * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0326</td>
<td>$CETAS = CETAS + ETAS * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0327</td>
<td>$CN55 = CN55 * WNES * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0328</td>
<td>$CN55 = CN55 * WNES * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0329</td>
<td>$CN55 = CN55 * WNES * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0330</td>
<td>$CN55 = CN55 * WNES * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0331</td>
<td>$CN55 = CN55 * WNES * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0332</td>
<td>$CN55 = CN55 * WNES * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0333</td>
<td>$CN55 = CN55 * WNES * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0334</td>
<td>$CESAVG = CESAVG + EAVG * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0335</td>
<td>$CEAVG = CEAVG + EAVG * DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0336</td>
<td>$CSLSS = CLSS * WNES * DAVS / (TMS * SF3 * WGHT)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0337</td>
<td>$WVLF = WVLF / D15G$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0338</td>
<td>$ALISS = CLSS / D15G$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0339</td>
<td>$ANISVG = CN55 / D15G$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0340</td>
<td>$ANISVG = CN55 / D15G$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0341</td>
<td>$CETEP = CETEP / DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0342</td>
<td>$AETAS + CETAS / DAVS$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0343</td>
<td>$IF (ANIZVG, NE, 0) WVL2 = (EO * ANISVG - AEETEP - AETAS) / ANIZVG$.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FORTRAN IV GLEVEL 10

0344 ANES=CNFS/DTAVS
0345 ANAS=CNFS/DTAVS
0346 ANS=CNFS/DTAVS
0347 ANS=CNFS/DTAVS
0348 ANMET=CNHT/DTAVS
0349 ANEX=CNEX/DTAVS
0350 ANS+ANER+ANAS-ANIZVG
0351 IF (ANS.LT.0.0) ANS=MSI/SNF
0352 IF (ANS.LT.0.0) ANS=MSI/SNF
0353 IF (WNP.NE.0.1) WNP=ANGEO-ETEP-ETAS1/MNIZN
0354 IF (NS.LE.1) TMS=0.0
0355 NET=NEKS/NAS
0356 CALL SOURCE(NS,NTP)
0357 IF (NET.GT.0.0) CALL FILMDNTP,MES, NAS)
0358 QD.1NIT=QD.INT+NIE6
0359 IF (WNP.GT.0.0) CALL SEPROCINTP, NS1
0360 IF (VNOX.GT.0.0) CALL AREPROCINTP, NS2
0361 NAST=NAST+NWS
0362 C INTRODUCHE SOURCE PARTICLES FOR THE NEXT CYCLE
0363 C EVERY FIFTH CYCLE RENORMALIZE NEON AND OUTPUT QUANTITIES ARE PRINTED
0364 C RENORMALIZATION PROCESS OF NION
0365 ANSTP=1.0/NSTP
0366 ANTPS=1.0/NSTP
0367 CNREZ*M=CNREZ*M+NREZ(M)*DAVS
0368 16 CEZ(M)=CEZ(M)+EZ(M)*DAVS
0369 DO IN=1.0 NS
0370 PNREZ(M)=NREZ(M)*ANTP
0371 16 EZ(M)=FT(M)/NREZ(M)
0372 SUMN=0.0
0373 SUMN=SUMN+NE(M)
0374 TNE(M)=TNE(M)+NE(M)/EDE(M)
0375 193 FNE(M)=NE(M)/ANTP/EDE(M)
0376 EAV*0.0
0377 DO IN=1.0 NS
0378 EAV3=EV(M)=FNE(M)*EDE(M)+EAV3
0379 1755 CNH=M=ANSTP
0380 CNHI=CNH+P,2,A1KT/SORT(AKT+1)
0381 ANIOV=0.0
0382 ANIO2=0.0
0383 DO IN=20 NS
0384 ANIO2=ANIOZ+FNE(M)*CNH(M)*EDE(M)
0385 ANION=ANION+FNE(M)*CNH(M)*EDE(M)
0386 20 CONTINUE
0387 DNION=ANION+CNH(M)
0388 DNION=ANION+CNH(M)
0389 CDNL=CDNLSS
0390 IF (ALCSS.LT.0.0) CDL=0.0
0391 IF (ABS(CJD1.GT.1.0)) CDL=1.0+50
0392 TDON=ANION+CJD1/DNION
0393 .ANION=ABS(50+CJD1/DNION

01/28/20
FORTRAN IV MAIN

MAIN (TNIO2=SORT(TNIO2)

DATE = 72125

0394

0395

0396

0397

0398

0399

0400

0401

0402

0403

0404

0405

0406

0407

0408

0409

0410

0411

0412

0413

0414

0415

0416

0417

0418

0419

0420

0421

0422

0423

0424

0425

0426

0427

0428

0429

0430

0431

0432

0433

0434

156
1  CALCULATED N+ USE PN+ E14.6,3X, CALC N+ USE CN+ E14.6,3X, 'AR

XIF C(=1:14.6,71
0457 IF (.NOT.(MODINT,5).NE.0) OR (.NOT.(MODINT,10).EQ.0)) GO TO 3
0458 WRITE(9) EAVG, DAVG, THE, NTW, NT, NTP, NIDN, DODEG, NSW, SWF, E, U
0459 WRITE(9) Z, WT, ANIOD, SP
0460 ENDFILE 9
0461 REWIND 9
0462 3 CONTINUE
0463 IF (.NOT.(MODINT,10).NE.0) GO TO 4
0464 WRITE(10) EAVG, DAVG, THE, NTW, NT, NTP, NIDN, DODEG, NSW, SWF, E, U
0465 WRITE(10) Z, WT, ANIOD, SP
0466 ENDFILE 10
0467 REWIND 10
0468 4 CONTINUE
0469 IF (.NOT.(NZIP,NZIP).NE.0) GO TO 4666
0470 MISP=0
0471 NTP=0
0472 C5AVS=C,E
0473 NAST=0
0474 NTWRT=0
0475 M=(I.ECGR
0476 USP=0.0
0477 C5F1=0.0
0478 CNIOZ=0.0
0479 CN51=0.0
0480 C5N9S=0.0
0481 C5N4T=0.0
0482 C5NEX=0.0
0483 C5AVG=0.0
0484 CEASG=0.0
0485 DO 54 ND=1,JE
0486 CNRE2(M)=0.0
0487 54 CE/M(M)=.0
0488 DU 55 ND=1,JE
0489 THE(M)=0.0
0489 TNZ(M)=0.0
0489 IF (.NOT.(MODINT,101).NE.0) GO TO 6666
0490 6666 CONTINUE
0491 IF (.NOT.(NZIP,AS I%$).NE.0) GO TO 6666
0492 M= E14.6,3X, 'AR
0493 NTP=0
0494 NTP=0
0494 NTP=0
0495 C5N9S=0.0
0496 C5N4T=0.0
0497 6666 CONTINUE
0498 M= E14.6,3X, 'AR
0499 DIZ(M)=0.0
0500 IRR(M)=0.0
0501 IRR(M)=0.0
0502 TES(M)=0.0
0503 LAD(M)=0.0
0504 DO 71 ND=1,500
0505 M= E14.6,3X, 'AR
0506 71 ESE1(M)=0.0
0507 E=1
C. REINITIALIZE THE OUTPUT QUANTITIES

0508       DO 52 M=1, JE
0509       RE(M)=0.
0510   52 NE(M)= 0.0
0511       DO 53 N=1, JL
0512       NLZ(N)=0.0
0513       EZ(M)=0.0
0514       DO 53 N=1, JE
0515       RZ(N)=0.0
0516       RZ(N)=0.0
0517       NZN=0
0518       NZN=0
0519       NZN=0
0520       NES=0
0521       NES=0
0522       MCUL=0
0523       NAC=0
0524       D0T=0.0
0525       NET=0
0526       NEXCE=0
0527       WNAS=0.0
0528       WNAS=0.0
0529       WNES=0.0
0530       WNES=0.0
0531       WNEt=0.0
0532       WNEt=0.0
0533       ETAS=0.0
0534       FTEP=0.0
0535       TUT=0.0
0536       NUP=0
0537       NOPE=0
0538       NOPE=0
0539       INV=INV+1000
0540       CALL INV(INV)
0541       WPITC(J,721)
0542       IF INT.ENT.3 == GO TO 7777
0543       IF 2000 STOP
0544       DEBUG SUBCHK
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.