251
4-28-81
88
②
FEBRUARY 1981H. 2567
PPPL-1753
UC-20f.g

MASTER 2-3890

NEW TECHNIQUES FOR CALCULATING HEAT
AND PARTICLE SOURCE RATES DUE TO
NEUTRAL-BEAM INJECTION IN
AXISYMMETRIC TOKAMAKS

BY

R. J. GOLDSTON, D. C. MCCUNE,
H. H. TOWNER, S. L. DAVIS,
R. J. HAWRYLUK, G. L. SCHMIDT

PLASMA PHYSICS
LABORATORY

PRINCETON UNIVERSITY
PRINCETON, NEW JERSEY

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

This work was supported by the U.S. Department of Energy
Contract No. DE-AC02-76-CHO 3073. Reproduction, transla-
tion, publication, use and disposal, in whole or in part,
by or for the United States government is permitted.

R. J. Goldston, D. C. McCune, H. H. Towner
S. L. Davis, R. J. Hawryluk, G. L. Schmidt

ABSTRACT

DISCLAIMER

[illegible]

I. INTRODUCTION

In order to determine the electron thermal diffusivity $[\chi_e(r,t)]$, the ion thermal diffusivity $[\chi_i(r,t)]$ and the particle diffusivity $[D(r,t)]$ from time- and space-resolved measurements of plasma parameters during neutral beam injection in tokamaks, one needs to know accurately the beam power delivered from the injected particles to the bulk ions, the power delivered to the bulk electrons, and the ion and electron particle source rates as functions both of space and time. In ohmically heated tokamak discharges, the current density, $j(r)$, and the concomitant safety factor, $q(r)$, as well as $\vec{j}(r) \cdot \vec{E}(r)$ are only known qualitatively, due to uncertainties in the electrical conductivity and in $E_\phi(r)$, the toroidal electric field. As a result it is difficult to determine χ_e accurately, or to calculate the neoclassical χ_i (which is proportional to q^2) for comparison with experiment. In addition, because of the direct link between the poloidal magnetic field (B_p) and the ohmic heating power density, it is intrinsically impossible to determine the parametric dependence of χ_e on B_p in tokamak experiments without auxiliary heating. With neutral beam injection, by contrast, it is possible at least in principle to determine the input power profiles with reasonable accuracy, and the plasma heating can certainly be varied independently of the confining fields. In addition, since the particle source rate is peaked near the plasma center, neutral beam injection provides much clearer experimental conditions for the study of particle transport than does the usual recycling source, which is strongly peaked to the outside. There is, as a result, a high premium on knowing the particle source term accurately, in addition to the heating rate. When n_b/n_e is not small, however, the calculation of the energy and particle sources due to injection is especially complex, and Monte Carlo

techniques must be used in order to include all of the effects which are known to be important. In this paper we describe the methods used in the transport analysis code TRANSP (Ref. 1) to calculate beam deposition (Section II), fast ion orbits (Section III), charged particle collisions (Section IV), charge exchange transport of beam particles (Section V), and, finally, beam driven currents and momentum transfer (Section VI). In light of the fact that neutral beam injection experiments are costly, and the data available for analysis are rare, we feel that as complete a model as possible should be used to analyze transport in these cases. Nonetheless there is real value in quick, approximate treatments, and in the concluding Section VII we point out a number of corners that might be cut, suggest a few ways to do this, and also point out which effects, in our opinion, may not be neglected in present day experiments.

II. BEAM DEPOSITION

A. Constant Census

In order to provide control over statistical fluctuations in the calculated source rates, it is desirable to maintain a chosen "constant census" of trajectories being followed by the fast ion orbit code. We find, for example, that runs with ~ 1000 Monte Carlo particles give reasonably smooth and reproducible energy and particle source rates. In order to maintain this "constant census" of test particles, whenever new beam particles are to be injected, first old ones must be deleted. Thus each time the code calculates beam deposition it first computes the actual physical number of particles to be introduced, given by

$$N_{inj} = \int_t^{t + \Delta t} [I_b(E_0) + I_b(E_0/2) + I_b(E_0/3)] dt ,$$

where Δt is the beam code time step, typically 5 msec for PLT, and I_b is the input neutral beam current at the full, half, and one-third energies. N_{inj} is to be compared to the number of physical fast ions circulating in the plasma, N_{circ} , and the code prepares to inject a number of new Monte Carlo particles given by

$$N_{new}^* = \frac{N_{cc}^* N_{inj}}{N_{inj} + N_{circ}} ,$$

where the star superscript designates Monte Carlo particles, as opposed to physical ones. N_{cc}^* is the desired constant census number, in our case $N_{cc}^* = 1000$. To make space for the new particles we must eliminate a number of Monte Carlo particles using Russian roulette. The number to delete, N_{Rr}^* , is given by

$$N_{Rr}^* = N_{new}^* + N^* - N_{cc}^* ,$$

where N^* is the current number of circulating Monte Carlo particles. The fraction to be randomly deleted, N_{Rr}^*/N^* , is eliminated in each radial zone separately in order to avoid creating fluctuations in the beam density. Each Monte Carlo particle has a weight, W , which gives the number of physical particles it represents. After the Russian roulette, the total physical energy that has been deleted from the circulating ions is replaced by proportionally increasing the weight of each of the remaining particles. We note that this technique tends to minimize Russian roulette during the steady-

state phase of injection, since at this time typically about the same number of beam ions are being injected as are thermalizing at $3/2 kT_i$, leaving the plasma on loss orbits or leaving by charge exchange. As a consequence of this, the Monte Carlo weights do not increase indefinitely during the slowing down process. If instead one allots Monte Carlo particles to new injected ions in proportion to their energy rather than in proportion to their number, W increases considerably during the lifetime of a particle. As a result fairly few Monte Carlo particles eventually must represent the entire thermalization rate of the slowed down beam ions, and this seemingly reasonable approach gives unacceptably poor statistics in the particle source terms.

B. Neutral Trajectories

1. Geometry

Once the code has determined the number of new Monte Carlo particles to inject, it selects trajectories for these particles. The finite-width neutral beam is taken to be composed of parallel rays traveling horizontally into the plasma from the ion source, while the intensity profile simulates the measured beam intensity profile at the plasma center. The code uses the Box-Mueller technique (Ref. 2) to sample starting points out of a Gaussian intensity distribution, and any trajectories which lie outside of a rectangular cut-off which represents the injection port, are rejected. The energies of the injected particles are randomly selected with probability proportional to the measured beam intensity at E_0 , $E_0/2$, and $E_0/3$.

To determine the deposition of the neutral particles, the usual Monte Carlo technique for simulating exponential attenuation is used. A random number, η , is selected such that $0 < \eta < 1$. The code then performs the integral

$$I = \int n_e \sigma_{\text{tot}} d\ell$$

along the particle trajectory, where σ_{tot} is the total beam stopping cross section described below. The integration is done most economically by stepping not in constant increments of $\delta\ell$ along the particle trajectory, but rather in constant increments, δR , of major radius. This allows rather large steps in $\delta\ell$ near the point where the neutral beam is tangent to $\vec{B}$, without any sacrifice in accuracy. At the point along the trajectory where $I = \ln(1/\eta)$, the beam neutral is taken to be ionized and is converted to a fast ion ready to be treated by the orbit code.

A final Russian roulette is applied before initial conditions for a fast ion are passed to the orbit code. Particles that are deposited near the center of the plasma are relatively more important for plasma heating and fueling than those that orbit at the periphery, but the shrinking volume of the radial zones near the plasma core tends to lower the number of Monte Carlo particles in the central zones. To counteract this tendency, Russian roulette is used to improve statistics near the center, at the expense of statistics towards the outside of the plasma. At the moment of deposition at radius r , a Monte Carlo particle represents W fast ions. After deposition, this weight, W , is to be increased by a factor $F = 1 + (W_a + 1)r/a$, where W_a is a constant, typically of order 10-50. A random number is chosen, however, $0 < \eta < 1$, and if $\eta > 1/F$ the particle is "Russian rouletted," i.e., eliminated from consideration. Thus near the plasma periphery particles are given weights increased by a factor of $\sim W_a$, but only $\sim 1/W_a$ of those that are initially deposited are followed by the orbit code. When W_a is chosen equal to the number of radial zones in the code (20 in the current version of the TRANSP

beam code) this redistribution of statistics just counterbalances the intrinsic bias in the statistics towards giving the outer zones with larger volumes more Monte Carlo particles. Any weight which is created (or destroyed) through the Monte Carlo noise associated with this Russian roulette step is eliminated by renormalizing all of the new deposited weights to give the proper total amount of injected physical energy.

2. Stopping Cross Sections

σ_{tot} , the total beam stopping cross section, has contributions from electron impact ionization (Ref. 3), from impact ionization collisions with hydrogenic species bulk ions (Ref. 3), from charge exchange with hydrogenic species bulk ions (Ref. 4), from total electron loss in collisions with impurities (Ref. 5), and finally from charge exchange and impact ionization with circulating beam ions. Reaction rates for collisions with electrons, $\langle\sigma v\rangle_{ei}$, and with impurities, $\langle\sigma v\rangle_{loss}$, are calculated in the usual way (Ref. 6). For charge changing collisions with hydrogenic species bulk ions, $\langle\sigma v\rangle_{ii}$ and $\langle\sigma v\rangle_{cx}$ are interpolated from a table of previously calculated values for a monoenergetic beam traveling in a Maxwellian background plasma. The TRANSP code allows for a mixture of hydrogenic species, and collision cross sections are computed separately for each.

Beam neutral collisions with circulating beam ions are somewhat more difficult to compute. The reaction rates for charge exchange and impact ionization between the circulating beam ions and the injected beam neutrals, which are required at a given time step, are calculated during the previous beam code time step. During the previous time step's calculation of beam deposition, the density profiles of injected primary neutrals at E_0 , $E_0/2$, and $E_0/3$ were calculated by taking contributions from each step along the trajectories of each injected Monte Carlo neutral. The result is given by

$$n_b^0 = \frac{1}{\Delta t V} \sum_j \frac{W_j \delta l_j}{v_j},$$

where V is the volume of the region under consideration, W_j is the weight of the j 'th Monte Carlo particle (the number of physical particles it represents), δl_j is its flight path within volume V , and v_j is its injection velocity.

During the subsequent orbiting and slowing down of the circulating fast ions (described below) the reaction rates for charge exchange and impact ionization between the fast ions and the primary beam neutrals are calculated. The relative angles between the injected neutrals and the orbiting ions are taken into account, and direct calculations of the beam-beam reaction rates for charge exchange $n_b^0 n_b^+ \langle \sigma v \rangle_{cx}$ and for impact ionization $n_b^0 n_b^+ \langle \sigma v \rangle_{ii}$ are performed by summing

$$n_b^0 n_b^+ \langle \sigma v \rangle = \frac{n_b^0}{\Delta t V} \sum_j \delta t_j W_j \sigma(v_{rel}) v_{rel}$$

along the ion trajectories. Since n_b^0 is known for each energy, the stopping power of the beam ions, $n_b^+ \langle \sigma v \rangle$, is now known as well and can be used in the subsequent beam deposition step as part of the total plasma stopping power. $n_e \sigma_{tot} v$ is then given by the sum of all the reaction rates discussed here. For high power neutral injection into high temperature, low density plasmas (Ref. 7), beam-beam collisions can constitute an important fraction of the total stopping power of the plasma (Fig. 1).

C. Particle Source Terms

The source rates of electrons and thermal neutral atoms within the plasma due to beam deposition could be calculated by taking into account only particle production during the ionization and charge exchange events which occur when Monte Carlo beam neutrals are converted into fast ions. This method, while it has the advantage that it conserves particles precisely (for example, only W neutral atoms can be generated by charge exchange between thermal ions and a Monte Carlo injected neutral with weight W), it has the disadvantage of failing to make use of all of the information available. Instead we use the technique described above for the calculation of the density of the primary beam neutrals, n_b^0 , to also calculate $n_b^0 n_i \langle \sigma \rangle_{cx}$, in order to generate the volume source of thermal neutrals as input to the neutral transport code. This neutrals calculation is itself a Monte Carlo code which has evolved from the AURORA code (Ref. 8). In addition, the beam code calculates $n_b^0 n_i \langle \sigma \rangle_{ii}$ (impact ionization on hydrogenic bulk ions), $n_b^0 n_x \langle \sigma \rangle_{loss}$ (total electron loss in collisions with impurities), and $n_b^0 n_e \langle \sigma \rangle_{ei}$ (electron impact ionization), in order to generate the volume electron source due to injection. The creation of beam ions from fast neutrals is not considered a source of ions; rather the beam ions are considered an independent species until they slow down to $3/2 T_i$. When beam ions charge exchange with thermal neutrals, this constitutes a neutral particle sink (which is included in the neutral transport code) and also a source of thermal ions.

The calculation of the particle source terms is, in many cases, the aspect of the beam code which is most subject to Monte Carlo noise. This is because the particle source that is calculated comes largely from a single deposition step, while the heating rates are due to circulating fast ions that

have been deposited over a series of previous steps. Control over the number of neutral trajectories tracked into the plasma has therefore been provided in order to be able to improve the source term statistics without depositing more than the required number of Monte Carlo fast ions. Typically 700 neutral trajectories need to be followed each time step for adequate statistics, and if fewer are called for by the deposition code, the necessary additional trajectories are provided for use in the source term calculations.

III. FAST ION ORBITS

Once beam neutrals are ionized in the plasma, the guiding center orbit equations are integrated during the slowing down process to find where energy and thermalized ions are deposited in the plasma. The magnetic geometry currently employed in TRANSP consists of axisymmetric circular concentric flux surfaces, with B_θ assumed much less than B_ϕ , so that the gradient and curvature drifts in the poloidal field may be neglected. With these approximations the guiding center orbit equations become quite simple. Expressed in r, θ coordinates they are

$$\dot{r} = v_d \sin \theta$$

$$\dot{\theta} = \frac{v_\perp}{qR_0} + v_d \frac{\cos \theta}{r},$$

where

$$v_d = \frac{(\epsilon - \mu B/2)}{\Omega_0 R_0}$$

$$\epsilon = v^2 \quad \mu = v_\perp^2/B \quad \Omega_0 = e B_0/(mc) \quad ,$$

ϵ and μ are conserved quantities, so we calculate directly $v_{\perp} = \sqrt{\epsilon - \mu B}$. An equation for $\dot{v}_{\parallel}$ can also be derived

$$\dot{v}_{\parallel} = \frac{\mu B r \sin \theta}{2 q R^2}.$$

This equation, however, is only integrated in order to determine the sign of $v_{\parallel}$. Thus our orbit equations explicitly conserve ϵ and μ to the computer round off error. It is convenient to express the orbit equations in the r, θ "flux coordinates" described above, since these are natural to the problem being solved when $v_{\parallel} B_0/B_{\phi} \gg v_d$. We have experimented with using R, y coordinates in the poloidal plane, and we find that a time step of approximately half the size is required to obtain equal accuracy.

R, y coordinates have the advantage that they do not have the nonphysical singularity at the origin which makes $\dot{\theta}$ go to infinity for an orbit drifting vertically through $r = 0$. To avoid this problem, which can be important for low magnetic field and high beam energy experiments where v_d is not small, we have adopted a hybrid approach. In the region near the origin where $r < 3v_d \delta t$, the code solves the orbit equations in R, y coordinates, while elsewhere it employs r, θ "flux coordinates." This avoids the singularity at the origin, while preserving most of the speed and accuracy of the r, θ system.

The technique we use to integrate the drift equations is to perform a simple two-term Taylor expansion. We have not experimented extensively with other methods, but very high order techniques would be inappropriate, since the drift motion is interrupted many times during each orbit by Fokker-Planck collisions (see below). We find that to maintain roughly constant accuracy in integrating the equations as pitch angle and energy vary, it is convenient to impose a time step control:

$$\delta t \propto v^{-1} (1 + 3\mu B/\epsilon) .$$

For a velocity of 2×10^8 cm/sec, and $\mu = 0$ in PLT, adequate accuracy is obtained with $\delta t = 1.5 \times 10^{-7}$ sec. If δt is increased substantially above this value, the first symptom of numerical inaccuracy is that passing orbits tend to shrink down towards $r = 0$, in violation of Liouville's theorem. For scaling from machine to machine, we have found that δt should be varied approximately in proportion to the major radius of the device.

IV. CHARGED PARTICLE COLLISIONS

A. Fokker-Planck Equation

The appropriate Fokker-Planck operator for small angle binary collisions of test particles in a magnetized Maxwellian plasma, with $v_i < v < v_e$, is (Ref. 9, 10)

$$\left. \frac{\partial f}{\partial t} \right|_c = \frac{1}{\tau_s v^2} \frac{\partial}{\partial v} (v^3 + v_c^3) f + \frac{1}{\tau_s v^2} \frac{\partial}{\partial v} \left(\frac{v^2 T_e}{m_b} + \frac{v_c^3 T_i}{m_b v} \right) \frac{\partial f}{\partial v} + \frac{v_{ii}}{2} \frac{\partial}{\partial \zeta} (1 - \zeta^2) \frac{\partial}{\partial \zeta} f ,$$

$$\text{where } \tau_s = \frac{6.32 \times 10^8 A_b T_e^{3/2} \text{ (eV)}}{n_e Z_b^2 \ln \Lambda_e} \quad v_{ii} = \frac{\langle Z \rangle \bar{A}_1 v_c^3}{\tau_s [Z] A_b v^3}$$

$$v_c^3 = 1.51 \times 10^{20} T_e^{3/2} \text{ (eV)} [Z] / \bar{A}_1$$

$$\langle Z \rangle = \frac{\sum n_i Z_i^2 \ln \Lambda_i}{n_e \ln \Lambda_e} \quad \frac{[Z]}{\bar{A}_1} = \frac{\sum n_i Z_i^2 \ln \Lambda_i / A_i}{n_e \ln \Lambda_e}$$

$$\zeta = v_i/v.$$

In this formulation we have neglected terms of order $(T_e/E_b)^2$ and of order $\exp(-E_b/T_i)$. The terms proportional to v_c^3 represent collisions with thermal ions; the other terms represent collisions with electrons. Interactions between beam ions have been neglected in this linearization of the collision operator. The beam particles, however, do displace thermal ions, reducing n_i/n_e and so decreasing v_c^3 . In order to investigate the accuracy of this approximation, we have developed a simple isotropic Fokker-Planck code, based on solving the nonlinear integro-differential Fokker-Planck equation, essentially as described in Ref. (11). For a reasonable range of parameters, including $n_b \gtrsim n_i$, we find that the ion and electron heating rates as calculated with the full nonlinear equation are approximated to within 1% by using the linearized operator, but allowing the beam particles to displace thermal ions, as in the Monte Carlo code. Collisions between beam particles cause significantly enhanced energy diffusion of the beam species, but evidently little change in the net energy transfer.

We have not discussed so far the effect of the toroidal electric field. Neglecting neoclassical processes, the combined drift of the electron distribution and the toroidal electric field together give rise to an "effective electric field," E^* . For $Z_b = 1$, $E^* = E_\phi (1 - 1/\langle Z \rangle)$. If one assumes that the electrons in a tokamak have a neoclassical distribution function, then the electron banana trapping process also affects E^* . Calculations, however, are only available for the collisionless regime (Ref. 12, 13). For large aspect ratio one finds

$$E^* = E_\phi [1 - 1/\langle Z \rangle + 1.46 / \bar{e} (1 + 0.7/\langle Z \rangle)/\langle Z \rangle].$$

This gives rise to an acceleration of the fast ions

$$\frac{\partial v}{\partial t} = \frac{eE^* r}{m_b}$$

and a change in pitch angle:

$$\frac{\partial \zeta}{\partial t} = (1 - \zeta^2) \frac{eE^*}{m_b v}.$$

Both of these effects are implemented each time step by altering v and ζ by

$$\Delta v = \delta t \, eE^* r / m_b$$

$$\Delta \zeta = \delta t (1 - \zeta^2) eE^* / m_b v.$$

B. Monte Carlo Simulation Technique

The three terms in the Fokker-Planck operator given above represent drag, energy diffusion, and pitch angle scattering. Starting from a delta function source (the Monte Carlo particle with given v_0 , ζ_0), the operator causes a directed motion in velocity space towards lower v and smaller $|\zeta|$, as well as a spreading in both dimensions. In order to calculate these effects, we evaluate

$$\begin{aligned} \langle \partial v / \partial t \rangle &= \frac{\int (\partial f / \partial t) v^3 \, dv d\zeta}{\int f v^2 \, dv d\zeta} \\ \langle \partial v^2 / \partial t \rangle &= \frac{\int (\partial f / \partial t) v^4 \, dv d\zeta}{\int f v^2 \, dv d\zeta} \\ \langle \partial \zeta / \partial t \rangle &= \frac{\int \zeta (\partial f / \partial t) v^2 \, dv d\zeta}{\int f v^2 \, dv d\zeta} \end{aligned}$$

$$\langle \partial \zeta^2 / \partial t \rangle = \frac{\int \zeta^2 (\partial f / \partial t) v^2 dv d\zeta}{\int f v^2 dv d\zeta}.$$

Taking $f = \delta(v - v_0) \delta(\zeta - \zeta_0)$, we arrive at

$$\langle \partial v / \partial t \rangle = - \frac{v}{\tau_s} (1 - 2T_e / m_b v^2) - \frac{v^3}{v^2 \tau_s} (1 + T_i / m_b v^2)$$

$$\langle \partial \zeta / \partial t \rangle = - v_{ii} \zeta,$$

and the appropriate displacement in velocity space is given by

$$v' = v_0 - \delta t \left[\frac{v}{\tau_s} (1 - 2T_e / m_b v_0^2) + \frac{v^3}{v_0^2 \tau_s} (1 + T_i / m_b v_0^2) \right]$$

$$\zeta' = \zeta_0 - \delta t \zeta_0 v_{ii},$$

where v' and ζ' are the new values of v and ζ after a time step δt .

The diffusive spreading in velocity space is given by

$$\langle \Delta v^2 \rangle = \langle (v - v')^2 \rangle = \delta t (\langle \partial v^2 / \partial t \rangle - 2v \langle \partial v / \partial t \rangle)$$

and

$$\langle \Delta \zeta^2 \rangle = \langle (\zeta - \zeta')^2 \rangle = \delta t (\langle \partial \zeta^2 / \partial t \rangle - 2\zeta \langle \partial \zeta / \partial t \rangle),$$

where we have retained terms to order δt only. Evaluating these expressions, we arrive at

$$\langle \Delta v^2 \rangle = \frac{2\delta t}{\tau_s} \left(\frac{T_e}{m_b} + \frac{v^3}{v^3} \frac{T_i}{m_b} \right)$$

and

$$\langle \Delta \zeta^2 \rangle = \delta t \, v_{ii} (1 - \zeta_0^2) .$$

The spreading is implemented by using the Box-Mueller technique to select a new velocity and a new pitch angle from Gaussian distributions of width $\langle \Delta v^2 \rangle$ and $\langle \Delta \zeta^2 \rangle$. Note that if the speed diffusion is to be included, one must also include the order (T/E_b) correction terms to the drag rates calculated above, in order to be consistent to order (T/E_b) .

At each time step where v is altered, the beam particle energy loss to the ions, to the electrons, and to the toroidal electric field is evaluated. These quantities are stored as radial arrays for each beam code time step.

C. Acceleration Technique

Using the orbit equations described in Section III and the Monte Carlo Fokker-Planck operator described above, one could simply integrate along fast ion trajectories, applying the collision operator at each time step. We find, however, that even for well optimized coding the computer time required for this straightforward simulation technique is exorbitant, and some method to speed up the calculation is required. The problem lies in the fact that the time scale for a fast ion to complete a single orbit is typically, at most, some tens of microseconds, while the thermalization time is on the order of tens of milliseconds. Thus inspection of the particle orbits in a straightforward simulation calculation reveals that many nearly identical orbits are generated over and over again during the slowing down process. This is most strongly evident for passing particles, which have the shortest orbit times and are the least sensitive to small changes in pitch angle. For

particles close to the trapping boundary, however, the bounce time is much longer and the orbit topology is quite sensitive to small amounts of pitch angle scattering, so sequentially calculated orbits are not so nearly identical.

Neoclassical theory suggests that the radial transport rate of beam ions in the banana regime should be proportional to their collision frequency. Since the slowing down rate is also a collisional phenomenon, one should therefore be able to enhance the Fokker-Planck effects, to some degree, relative to the orbital motion of the fast ions, with little net change in the radial transport. The limitation on this speed up technique is that the collision rate must not be so strongly enhanced that fast ions are artificially transferred into the plateau regime, where transport processes no longer depend linearly on collision frequency. Indeed modern neoclassical theory (Ref. 14) shows that plateau effects extend well into the banana regime, and that $D_{\perp}$ does not become proportional to $v_{\perp i}$ until the plasma as a whole is extremely collisionless. This is because of the effect mentioned above: particles with pitch angles near the trapping boundary have long bounce times, and small collisions alter their orbits substantially; thus they enter into the banana regime at much higher energies than do more typical particles.

The danger associated with too much speed up can clearly be seen in the fast ion orbit code. In straight simulation calculations, or in cases with modest degrees of acceleration, fast ions which go from counter passing orbits to banana orbits always pass through trajectories very close to the World's Fattest Banana (Ref. 6), where the banana tips meet at the machine midplane. If collisional effects are sped up substantially near the trapping boundary, however, particles can go from passing to deeply trapped without tracing out W.F.B. orbits. This means that their trajectories are less likely to travel

outside of the plasma on loss orbits.

To avoid the resulting spurious improvement in fast ion confinement, and to attempt to provide equal accuracy for all orbits against errors associated with the speed up procedure, a variable acceleration scheme is used. Each time a Monte Carlo particle crosses the machine midplane (thereby completing a half orbit) the code calculates the expectation value of the particle's change in pitch angle, $\langle |\Delta\zeta| \rangle$, its actual relative change in velocity, $\Delta v/v$, and its probability of charge exchange, $\int n_0 \langle \sigma v \rangle_{cx} dt$, during the previous half orbit. In order to obtain the desired degree of accuracy and the desired speed of calculation, the code attempts to maintain the largest of these dimensionless quantities equal to a predetermined acceleration constant, G . It does this by varying the speed up which is applied uniformly to the Fokker-Planck, $\vec{E}$ -field, and charge exchange processes for each half orbit, based on the results of the previous half orbit. Thus if $\langle |\Delta\zeta| \rangle = 0.05$, which is greater than both $\Delta v/v$ and $\int n_0 \langle \sigma v \rangle_{cx} dt$, and if $G = 0.1$, the code will increase the speed up for the next half orbit by a factor of 2. Conversely if $\langle |\Delta\zeta| \rangle$ were greater than G , the speed up would be reduced. For PLT parameters, with $G = 0.05$ the acceleration factor can be of order 100 for passing particles, while it automatically reduces to $\sim 1-2$ for particles near the trapping boundary. A nonphysical "deceleration" that would be associated with acceleration by less than unity is not permitted. Thus the loss regions for W.F.B.'s are adequately taken into account, and residual plateau effects on slower injected ions are accurately treated. Results from runs with $G = 0.05$ are in every respect indistinguishable from those of straight simulation runs, with the exception that execution times are reduced by more than an order of magnitude. For $G = 0.1$ in PLT the heating profiles are surprisingly close to the straight simulation runs, but the loss rates are somewhat reduced

since the W.F.B. orbits are not so accurately traced out.

Finally, it is not necessary to perform a Fokker-Planck operation at each orbit time step. Since it is desirable to simulate the continuous velocity space diffusion that occurs physically, many collision events are required per orbit, but this number of collisions is not necessarily identical to the number of integration time steps needed for accurate solution of the drift equations. Execution time can be significantly reduced, with no change in results, by simply making use of a skip factor, S , which is set equal to ~ 3 . At each orbit time step a random number η is selected, $0 < \eta \leq 1$, and if $\eta > 1/S$ no collision is calculated. If $\eta \leq 1/S$, a Fokker-Planck collision is computed, but δt for the calculation is enhanced by S .

V. CHARGE EXCHANGE TRANSPORT OF BEAM PARTICLES

Monte Carlo techniques are especially well suited to treating the problem of charge exchange transport of beam particles in tokamaks. In the usual approach to solving the Fokker-Planck equation by numerical integration (Ref. 10), one includes a charge exchange collision operator

$$\left. \frac{\partial f}{\partial t} \right|_{cx} = -f/\tau_{cx}(v) .$$

This accounts for depletion of the fast ion distribution function by charge exchange, but it does not take into consideration the possibility that the second generation beam neutrals which are created in these collisions may be reionized in the plasma. In the Monte Carlo approach the reionization probability can be treated naturally by continuing to follow beam particles through all charge changing collisions, until they either thermalize or escape the discharge.

A. Initial Charge Exchange

Beam ions circulating in the plasma may charge exchange with thermal neutrals originating at the walls or limiter, with volume source thermal neutrals originating from beam deposition and from recombination, with first generation "primary" beam neutrals coming directly from the injectors, and finally with "secondary" beam neutrals that themselves arise from charge exchange of circulating beam ions. The contribution to $n_o \langle \sigma v \rangle_{cx}$ due to collisions with thermal wall source and limiter source neutrals, as well as due to collisions with thermal volume source neutrals, is calculated assuming a local Maxwellian neutral distribution function, with $T_o = 2/3 \langle E_o \rangle$. At the same time local values of $n_o n_b^+ \langle \sigma v \rangle_{ii}$ and $n_o n_b^+ \langle \sigma v \rangle_{cx}$ are also computed, by integrating along the fast ion orbits. These are used to determine the loss rate for thermal neutrals due to collisions with beam ions, which is needed for the thermal neutral transport code.

The contribution to $n_o \langle \sigma v \rangle_{cx}$ due to collisions with primary beam neutrals takes into account the angle between the circulating beam ion velocity vector and the velocity vector of the beam neutral. Since a full six dimensional beam neutral distribution is not retained, the neutral velocity is assumed to be in the direction of the central ray of the injector, and the beam neutrals are assumed to be localized poloidally on each flux surface in the area where the finite width beam intersects the surface. Separate beam neutral density profiles for each injection energy are retained, however, so that the relative velocity between the injected neutrals and the circulating beam ions can be reasonably well calculated. As discussed in Section II, radial profiles of $n_b^o n_b^+ \langle \sigma v \rangle_{cx}$ and $n_b^o n_b^+ \langle \sigma v \rangle_{ii}$ for each energy are calculated by integrating along the ion trajectories for use in the beam deposition calculation. Since the volume source thermal neutrals arising from

beam deposition tend to be localized on the large major radius side of the plasma where the primary beam neutrals are located, the code assumes the same poloidal localization for these neutrals as for the beam neutrals.

The final contribution to $n_b^0 \langle \sigma v \rangle_{cx}$ arising from collisions with secondary beam neutrals is difficult to evaluate precisely. A full calculation of this process would take into account the distribution of secondary neutrals as a function of energy, aiming direction, radius, and poloidal angle within the plasma; this would be both costly and highly subject to Monte Carlo noise. No simple approximation to the direction and poloidal distribution is available, as in the case of the primary beam neutrals. As an alternative, therefore, the code keeps track only of the radial profile of the secondary beam neutral density, and of the profile of the average energy of the beam neutrals. To estimate the charge exchange reaction rate the relative energy is taken to be the sum of the beam ion energy and the average energy of the secondary beam neutrals. No anisotropy nor poloidal variation is assumed for the secondary neutrals.

B. Monte Carlo Simulation Technique

Particles which circulate at the outside of the plasma in the high neutral density region, have large weights (i.e., represent many physical particles) due to the initial Russian roulette process used in the deposition calculation (see Section II). As a result, if these particles were straightforwardly charge exchanged and reabsorbed in the central region of the plasma they would cause a substantial amount of statistical noise, unless their weight had been reduced by being split into particles with comparable weight to those already circulating in the central region. Also relatively few Monte Carlo particles are used to calculate the major portion of the

charge exchange loss. Therefore, for the purpose of calculating charge exchange, a given Monte Carlo beam ion is taken to represent $N_{cx} = W/\langle W \rangle_0$, "potential" Monte Carlo beam neutrals, where W is the weight of the given Monte Carlo ion and $\langle W \rangle_0$ is the average weight of particles in the central region of the plasma. Then the probability of one of these N_{cx} particles suffering a charge exchange reaction during time δt is $N_{cx} n_0 \langle \sigma v \rangle_{cx} \delta t$. The code therefore performs the integral $J = \int N_{cx} n_0 \langle \sigma v \rangle_{cx} dt$ along the fast ion orbit, until $J = \ln(1/\eta)$ where η is again a random number such that $0 < \eta \leq 1$. At this point the beam ion weight is reduced by W/N_{cx} , and a secondary neutral of weight W/N_{cx} is generated. The velocity vector of the secondary neutral is determined from the pitch angle of the original beam ion. Since a random gyrophase was already selected for the calculation of $N_{cx} n_0 \langle \sigma v \rangle_{cx}$ due to collisions with primary beam neutrals, this same phase angle is used for the velocity vector of the secondary neutral. The neutral is then tracked across the plasma and either escapes or is reionized.

C. Fast Neutral Transport

As a beam neutral travels in the plasma, the code calculates an integral $K = \int n_e \langle \sigma v \rangle_{tot} dt$ along the neutral trajectory and when (and if) K reaches $\ln(1/\eta)$ (where η is a new random number, $0 < \eta \leq 1$), the particle is reionized. In addition the code keeps track of the time the particle spends in each zone in order to find the secondary beam neutral density and average energy. It also finds the ion, electron, and thermal neutral source rate profiles as was done for the primary beam neutrals. The deposition cross sections for the secondary beam neutrals are calculated in the same manner as those for the primary beam neutrals, with the exception that beam neutral-beam ion collisions are calculated on the basis of the average beam ion

density and energy, analogously to the way that the charge exchange of circulating beam ions with secondary beam neutrals is calculated.

When a secondary beam neutral is redeposited in the plasma, a Russian roulette is performed which is complementary to the splitting that was done at its creation. In order to avoid following particles with weights much lower than the average at the deposition zone radius, r , the newly created beam ion (which typically has a relatively low weight for particles at zone radius r due to the initial splitting process) is given a chance of survival $C = W/\langle W \rangle_r$, where W is the Monte Carlo weight of the secondary beam neutral, and $\langle W \rangle_r$ is the average weight of particles circulating in the zone at radius r . If the particle survives this selection process, its weight is then increased by a factor of $1/C$. In the unusual case where $C > 1$, no Russian roulette is employed nor is the weight of the particle changed. In this manner the radial profile of the Monte Carlo weight of the beam ions is approximately preserved, making the most effective use of the particles being followed in each zone.

Figure 1 shows an example of a beam particle undergoing many of the processes described in Sections II through V. The particle is initially deposited as a counter passing ion and through pitch angle scattering enters a banana orbit and then scatters back to counter passing. It undergoes a number of charge exchange events, in each case being reionized within the plasma, and finally thermalizes at $3/2 T_i$. The parameters used in this calculation are typical of PLT neutral injection experiments, with the exception that the neutral density is enhanced by a factor of 10 in order to show a number of examples of charge exchange transport.

VI. BEAM DRIVEN CURRENTS AND MOMENTUM TRANSFER

The momentum balance between the circulating beam ions, the bulk ions, and the bulk electrons drifting in the toroidal electric field gives rise both to the effective electric field E^* (see Section IV) which acts on the beam ions, and to a net beam driven toroidal current. Including neoclassical effects, for large aspect ratio and $Z_b = 1$, this current is given by

$$j_{bd} = j_b \{ 1 - 1/\langle Z \rangle + 1.46 \sqrt{\epsilon} \{ 1 + 0.7/\langle Z \rangle \} / \langle Z \rangle \} ,$$

where

$$j_b = n_b e v_{||b} .$$

This calculation is only valid in the banana regime. For a situation where one assumes that banana trapping does not give rise to a neoclassical electron distribution function, the same expression for j_{bd} is valid, but with $\epsilon = 0$. Effects associated with bulk plasma rotation are neglected. The net j_{bd} can be calculated easily, since the power transferred to the fast ions from the ohmic heating circuit (as calculated in Section IV) is simply $j_{bd} E_\phi$.

The collisional momentum transfer to the bulk ions and electrons is straightforward to calculate, although the present version of TRANSP does not solve the momentum balance equation, so this feature is only implemented in standalone versions of the beam code. Drag and energy diffusion terms give rise to a transfer of angular momentum

$$\dot{p} = m_b \zeta (\Delta v) R$$

while pitch angle scattering causes a transfer

$$\dot{p} = m_b v(\Delta\zeta)R \quad .$$

The momentum gained from (lost to) the toroidal electric field during co- (counter-) injection would have been otherwise exchanged between the ohmic heating circuit and the bulk electrons, so this must be counted as a loss (gain) of momentum from (to) the bulk electrons.

The beam ions also transfer momentum to the bulk plasma through their radial motion. Co- (counter-) injected ions tend to lose part of their mechanical toroidal angular momentum by moving inward (outward) across the poloidal magnetic field. To preserve quasi-neutrality the bulk plasma sets up a counterbalancing radial current and, through the resulting $j_r \times B_\theta$ force, picks up the momentum lost from the beam ions. The code calculates the beam j_r by evaluating the net deposition of charge due to the beam particles and making use of the relation

$$\vec{\nabla} \cdot \vec{j} = \partial \rho / \partial t \quad .$$

VII. CONCLUSION

To analyze 500 msec of a PLT discharge including 150 msec of neutral beam injection using the TRANSP transport analysis code, typically requires ~15 hours of CPU time on a VAX-11 computer. Of this time ~10 hours (equivalent to ~20 minutes of CRAY-1 time) are devoted to the Monte Carlo beam code; the remainder of the time is used by the Monte Carlo thermal neutrals package, and the parts of the TRANSP code which solve the magnetic diffusion equation, the ion and electron energy balances, and the particle balance.

Clearly it would be desirable to develop a faster beam treatment, even if it were somewhat less accurate, which could be mated with a simplified neutrals package to produce a transport analysis code to be used for preliminary study of data, parameter surveys, and other applications where large numbers of runs are required, but the highest degree of accuracy is not necessary.

While the Monte Carlo techniques employed in TRANSP permit the inclusion of a great deal of detailed physics, it is the Monte Carlo approach itself which gives rise to the long running time. Thus it seems unlikely that omitting some of the physics treated in the present code but retaining the Monte Carlo approach would result in a substantial improvement. The alternative technique which is commonly used is to numerically solve the drift kinetic equation for the fast ions on each of a number of magnetic surfaces. In order to include as much orbit physics as possible, the small banana width bounce averaged Fokker-Planck equation (Ref. 15) should form the basis of the calculation, but with a loss region (Ref. 16) taken into account for counter and perpendicular injection.

In such a calculation much of the physics handled in the Monte Carlo calculation could be included. Orbit effects would be treated in a rough manner by averaging over first orbit excursions and including loss regions. Beam deposition through collisions with circulating beam ions could be taken into account by integrating $\langle gv \rangle$ over the calculated fast ion distribution. The only effect that seems impossible to include, even in an approximate manner, without Monte Carlo techniques is the reabsorption of charge exchanged fast neutrals. Thus it might prove necessary to retain a Monte Carlo subroutine to calculate the transport of beam particles by charge exchange. A simple reduction in the loss rate, modeling a "typical" reabsorption factor, fails to reflect the actual transport of beam particles across the plasma

column. A diffusion model would be inappropriate because of the anisotropy of the beam distribution and because $\lambda_{mfp} \approx a$ in neutral beam injection experiments. We speculate that a hybrid beam treatment including collisions with charged particles in a bounce averaged Fokker-Planck equation, but handling fast neutral transport with a Monte Carlo calculation, might be able to include most of the effects treated in the full Monte Carlo code with much less use of computer time. On the other hand, for situations where the fast ion banana width is large and E_b is of order E_c , proper treatment of the orbit physics is necessary.

In conclusion, we have presented here a description of the Monte Carlo techniques employed in the neutral beam section of the TRANSP tokamak transport analysis code. Features of this beam code that are unique and appropriate to intensely beam heated tokamaks are: 1. the inclusion of beam-beam collisions in the deposition process, 2. a full treatment of thermal neutral sources and sinks for use in a parallel running neutral transport code, 3. detailed tokamak orbit physics, including a speedup technique which accurately calculates World's Fattest Banana orbits as well as the transition to the plateau regime at low energies, 4. a Monte Carlo Fokker-Planck operator accurate to order T/E_b , which includes energy diffusion and electric field effects, and 5. a complete treatment of charge exchange transport of fast ions. While this code requires a considerable amount of computer time, it accurately treats most of the known important processes which affect neutral beam injection experiments in axisymmetric tokamaks.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Douglass Post, Dr. David Mikkelsen, and Dr. Graham Lister for many illuminating discussions on Monte Carlo techniques and their application to tokamak and neutral beam physics problems. This work was supported by the U.S. Department of Energy under contract No. DE-AC02-76-CHO-3073.

REFERENCES

- (1) R. J. Hawryluk, "An Empirical Approach to Tokamak Transport," in Course on Physics of Plasmas Close to Thermonuclear Conditions, Varenna, Italy 1979.
- (2) J. M. Hammersley and D. C. Handscomb, Monte Carlo Methods, Halsted Press, New York 1975, 3a.
- (3) R. L. Freeman and E. M. Jones, Culham Laboratory CLM-R 137 (1974).
- (4) A. C. Riviere, Nucl. Fusion 11 (1971) 363.
- (5) R. E. Olson et al., Phys. Rev. Lett. 41 (1978) 163.
- (6) T. L. Stix, Plasma Physics 14 (1972) 367.
- (7) H. P. Eubank et al., Proc. of Seventh Int. Conf. on Plasma Phys. and Conf. Nucl. Fus. Res., Innsbruck (1978).
- (8) M. H. Hughes and D. E. Post, J. Comp. Phys. 28 (1978) 43.
- (9) M. N. Rosenbluth, W. MacDonald, and D. Judd, Phys. Rev. 107 (1957) 1.
- (10) J. A. Cordey and W. G. F. Core, Phys. Fluids 17 (1974) 1626.
- (11) J. Killen and K. D. Marx, Methods of Computational Physics, Vol. 9 (Ed. B. Alder), Academic Press, New York (1970) 422.
- (12) J. W. Connor and J. G. Cordey, Nucl. Fusion 14 (1974) 185.
- (13) D. F. H. Stuart and J. G. Cordey, "Beam Induced Currents in Toroidal Plasmas of Arbitrary Aspect Ratio," unpublished manuscript (1980).
- (14) F. L. Hinton and M. N. Rosenbluth, Phys. Fluids 16 (1973) 836.
- (15) J. G. Cordey, Nucl. Fusion 16 (1976) 499.
- (16) J. A. Rome et al., Nucl. Fusion 16 (1976) 55.

FIGURE CAPTIONS

Fig. 1 Deposition rate of beam ions due to:

- a. Impact ionization on electrons, ions, and impurities plus charge exchange with impurities.
- b. Charge Exchange with D^+ background ions.
- c. Charge Exchange with H^+ background ions.
- d. Beam-Beam Charge Exchange.
- e. Beam-Beam Impact Ionization.

This is a case of 2.1 MW D^0 injection into an initially H^+ plasma. The line average electron density rose from 1 to $2 \times 10^{13} \text{ cm}^{-3}$, and the ion temperature rose to 5.5 keV.

Fig. 2 A Monte Carlo beam particle slowing down, pitch angle scattering, charge exchanging, and finally thermalizing in a PLT plasma. The neutral density has been enhanced by $\times 10$ to provide a number of examples of charge exchange transport.

Fig. 1. (PPPL-809092)

MONTE CARLO ORBIT