DE-971

195
10-2-74

UCRL-51615

HYBRID-II, A TWO-DIMENSIONAL MULTISPECIES FOKKER-PLANCK COMPUTER CODE

Arthur A. Mirin

July 26, 1974

Prepared for U.S. Atomic Energy Commission under contract No. W-7405-Eng-48

MASTER

LAWRENCE LIVERMORE LABORATORY

University of California/Livermore

Printed in the United States of America
Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy $___*; Microfiche $1.45

*Pages	NTIS Selling Price
1-50	$4.00
51-150	$5.45
151-325	$7.60
326-500	$10.60
501-1000	$13.60

TID-4500, UC-32
Mathematics and
Computers

LAWRENCE LIVERMORE LABORATORY
University of California/Livermore, California/94550

UCRL-51615

HYBRID-II, A TWO-DIMENSIONAL MULTISPECIES FOKKER-PLANCK COMPUTER CODE

Arthur A. Mirin

MS. date: July 26, 1974

MASTER

Contents

HYBRID-II, A TWO-DIMENSIONAL MULTISPECIES FOKKER-PLANCK COMPUTER CODE

Abstract

The computer program HYBRID-II solves the two-dimensional Fokker-Planck equation for an arbitrary number of ion species plus electrons, with the assumption that the electron distribution is isotropic. The ion equations are advanced using a second-order fully implicit split difference scheme. An implicit algorithm is used to advance the electron equation.

The ion velocity and angular finite difference meshes usually consist of 45 and 20 points, respectively. The electron velocity mesh, which is an extension of the ion velocity mesh, usually consists of 201 points. The field length for a three-species calculation on the above mesh is approximately 230000_8, and the running time is 5 to 10 minutes on the CDC 7600, depending, of course, on the number of time steps necessary.

The code operates without disk files or tapes. The necessary input consists merely of data cards. Both CRT plots and either HSP or microfiche printed data make up the output.

Introduction

This paper summarizes the basic equations of the HYBRID-II code and the methods used to solve them. A detailed description of the program along with a listing is included.

I. Basic Equations

We let $f_a(v, \theta)$ be the distribution function for ion species "a," and $f_e(v)$ the electron distribution function. Here, v is the velocity magnitude and θ is the angle between $\vec{v}$ and the magnetic axis. We assume $f_a(v, \theta) = f_a\left(v, \pi - \theta\right)$. The normalized velocity x is defined by

$$v = \tilde{v}x, \tag{1.1}$$

where $\tilde{v}$ is an input constant, and the normalized distribution functions are given by

$$F_a = \frac{\tilde{v}^3}{K_a} f_a, \tag{1.2}$$

where K_a is a constant.

The equation for ion species "a" is then

$$\frac{\partial F_a}{\partial t} = \frac{K_a \Gamma_a}{\tilde{v}^3} \left(\frac{1}{x^2} \frac{\partial \overline{G}_a}{\partial x} + \frac{1}{x^2 \sin\theta} \frac{\partial \overline{H}_a}{\partial \theta}\right) + S_a(x, \theta), \tag{1.3}$$

where S_a is the source term, and

$$\overline{G}_a = \overline{A}_a F_a + \overline{B}_a \frac{\partial F_a}{\partial x} + \overline{C}_a \frac{\partial F_a}{\partial \theta}, \tag{1.4}$$

$$\overline{H}_a = \overline{D}_a F_a + \overline{E}_a \frac{\partial F_a}{\partial x} + \overline{F}_a \frac{\partial F_a}{\partial \theta}. \tag{1.5}$$

The coefficients $\overline{A}_a$, $\overline{B}_a$, $\overline{C}_a$, $\overline{D}_a$, $\overline{E}_a$, and $\overline{F}_a$ satisfy

$$\overline{A}_a = \frac{x^2}{2} \frac{\partial^3 G_a}{\partial x^3} + x \frac{\partial^2 G_a}{\partial x^2} - \frac{\partial G_a}{\partial x} - \frac{1}{x} \frac{\partial^2 G_a}{\partial \theta^2} + \frac{1}{2} \frac{\partial^3 G_a}{\partial x \partial \theta^2} - \frac{\cot \theta}{x} \frac{\partial G_a}{\partial \theta}$$

$$+ \frac{\cot \theta}{2} \frac{\partial^2 G_a}{\partial x \partial \theta} - x^2 \frac{\partial H_a}{\partial x}, \tag{1.6}$$

$$\overline{B}_a = \frac{x^2}{2} \frac{\partial^2 G_a}{\partial x^2}, \tag{1.7}$$

$$\overline{C}_a = -\frac{1}{2x} \frac{\partial G_a}{\partial \theta} + \frac{1}{2} \frac{\partial^2 G_a}{\partial x \partial \theta}. \tag{1.8}$$

$$\overline{D}_a = \frac{\sin \theta}{2x^2} \frac{\partial^3 G_a}{\partial \theta^3} + \frac{\sin \theta}{2} \frac{\partial^3 G_a}{\partial x^2 \partial \theta} + \frac{\sin \theta}{x} \frac{\partial^2 G_a}{\partial x \partial \theta}$$

$$- \frac{1}{2x^2 \sin \theta} \frac{\partial G_a}{\partial \theta} + \frac{\cos \theta}{2x^2} \frac{\partial^2 G_a}{\partial \theta^2} - \sin \theta \frac{\partial H_a}{\partial \theta}, \tag{1.9}$$

$$\overline{E}_a = \sin \theta \, \overline{C}_a, \tag{1.10}$$

$$\overline{F}_a = \frac{\sin \theta}{2x^2} \frac{\partial^2 G_a}{\partial \theta^2} + \frac{\sin \theta}{2x} \frac{\partial G_a}{\partial x}, \tag{1.11}$$

where

$$G_a = \sum_b \ln \Lambda_{ab} \frac{Z_b^2}{Z_a^2} \frac{K_b}{K_a} g_b, \tag{1.12}$$

$$H_a = \sum_b \ln \Lambda_{ab} \frac{Z_b^2}{Z_a^2} \left(1 + \frac{m_a}{m_b}\right) \frac{K_b}{K_a} h_b, \tag{1.13}$$

and

$$\nabla^4 g_b = -8\pi\ F_b, \tag{1.14}$$

$$\nabla^2 h_b = -4\pi\ F_b. \tag{1.15}$$

Here, Z_a is the atomic number for species "a," m_a is the mass, and

$$\Gamma_a = 4\pi\ Z_a^4\ e^4/m_a^2\ . \tag{1.16}$$

The Coulomb logarithm is defined by

$$\ln \Lambda_{ab} = \ln\left[\left(\frac{m_a m_b}{m_a + m_b}\right)\frac{2\ \alpha\ \lambda_D c}{e^2}\ \sup_{a,\,b}\sqrt{\frac{2E}{m}}\right] - 1/2, \tag{1.17}$$

where

$$\alpha = 1/137, \tag{1.18}$$

$$\lambda_D = \sqrt{\frac{E_e}{6\pi n_e e^2}}. \tag{1.19}$$

The electron equation is

$$\frac{\partial F_e}{\partial t} = \frac{K_e \Gamma_e}{\tilde{v}^3}\left(\frac{1}{x^2}\ \frac{\partial \overline{G}_e}{\partial x} - \frac{\overline{F}_e\ \lambda\ F_e}{x^2}\right) + S_e. \tag{1.20}$$

where the θ derivatives in $\overline{G}_e$ and $\overline{F}_e$ are set to 0, and $\sin\theta = 1$ in $\overline{F}_e$. The quantity λ is defined as

$$\lambda = 1/\log_{10} R_{eff}. \tag{1.21}$$

where

$$R_{eff} = R/\left(1 - x_c^2/x^2\right) \tag{1.22}$$

if $x \geq x_c$, and $R_{eff} = \infty$ if $x \leq x_c$. Here, R is the mirror ratio (if applicable) and x_c is given by

$$e\phi = \frac{1}{2} m_e \tilde{v}^2 x_c^2, \qquad (1.23)$$

where ϕ is the ambipolar potential. For nonmirror devices, $\lambda = 0$. It is assumed that G_b and H_b, when used in the electron equation coefficients, depend only on the isotropic part of F_b.

2. Computation of Rosenbluth Potentials

The quantities g_b and h_b, defined in Eqs. (1.14) and (1.15), are Rosenbluth potentials. Changing notation, we let,

$$\nabla^2 h = -4\pi f, \qquad (2.1)$$

$$\nabla^4 g = -8\pi f, \qquad (2.2)$$

$$f(x, \theta) = \sum_{j=0}^{\infty} V_j(x) P_j(\cos\theta), \qquad (2.3)$$

$$g(x, \theta) = \sum_{j=0}^{\infty} \hat{G}_j(x) P_j(\cos\theta), \qquad (2.4)$$

$$h(x, \theta) = \sum_{j=0}^{\infty} \hat{H}_j(x) P_j(\cos\theta), \qquad (2.5)$$

where P_j are the Legendre polynomials.

We define the functionals

$$M_j(f)(x) = \int_x^{\infty} f(y)\, y^{1-j} dy, \qquad (2.6)$$

$$N_j(f)(x) = \int_0^{x} f(y)\, y^{2+j} dy, \qquad (2.7)$$

$$R_j(f)(x) = \int_x^{\infty} f(y)\, y^{3-j} dy, \qquad (2.8)$$

$$E_j(f)(x) = \int_0^{x} f(y)\, y^{4+j} dy. \qquad (2.9)$$

Then,

$$\hat{H}_j = \frac{4\pi}{2j+1}\left[x^{-j-1} N_j(V_j) + x^j M_j(V_j)\right], \tag{2.10}$$

$$\frac{\partial \hat{H}_j}{\partial x} = \frac{4\pi}{2j+1}\left[j x^{j-1} M_j(V_j) - (j+1) x^{-j-2} N_j(V_j)\right], \tag{2.11}$$

$$\hat{G}_j = \frac{4\pi}{2j+1}\left\{\frac{1}{2j+3}\left[x^{-j-1} E_j(V_j) + x^{j+2} M_j(V_j)\right]\right.$$

$$\left. - \frac{1}{2j-1}\left[x^{1-j} N_j(V_j) + x^j R_j(V_j)\right]\right\}, \tag{2.12}$$

$$\frac{\partial \hat{G}_j}{\partial x} = \frac{4\pi}{2j+1}\left\{\frac{1}{2j+3}\left[(j+2)x^{j+1} M_j(V_j) - (j+1)x^{-j-2} E_j(V_j)\right]\right.$$

$$\left. - \frac{1}{2j-1}\left[j x^{j-1} R_j(V_j) - (j-1)x^{-j} N_j(V_j)\right]\right\}, \tag{2.13}$$

$$\frac{\partial^2 \hat{G}_j}{\partial x^2} = \frac{4\pi}{2j+1}\left\{\frac{(j+1)(j+2)}{2j+3}\left[x^{-j-3} E_j(V_j) + x^j M_j(V_j)\right]\right.$$

$$\left. - \frac{j(j-1)}{2j-1}\left[x^{-j-1} N_j(V_j) + x^{j-2} R_j(V_j)\right]\right\}, \tag{2.14}$$

$$\frac{\partial^3 \hat{G}_j}{\partial x^3} = \frac{4\pi}{2j+1}\left\{\frac{1}{2j+3}\left[j(j+1)(j+2)x^{j-1} M_j(V_j) - (j+1)(j+2)(j+3)x^{-j-4} E_j(V_j)\right]\right.$$

$$\left. - \frac{1}{2j-1}\left[j(j-1)(j-2)x^{-j-3} R_j(V_j) - (j+1)(j)(j-1)x^{-j-2} N_j(V_j)\right]\right\}. \tag{2.15}$$

The Legendre polynomials satisfy

$$j\,P_j(\mu) - (2j-1)\,\mu\,P_{j-1}(\mu) + (j-1)\,P_{j-2}(\mu) = 0, \tag{2.16}$$

$$P_0(\mu) = 1, \tag{2.17}$$

$$P_1(\mu) = \mu, \tag{2.18}$$

$$P_j^{(n)}(\mu) = (2j-1)\, P_{j-1}^{n-1}(\mu) + P_{j-2}^{(n)}(\mu), \qquad (2.19)$$

where $\mu = \cos\theta$. Combining Eqs. (2.6) through (2.19), one can compute the necessary derivatives of g_b and h_b with respect to x and θ.

3. Boundary Conditions

For the "nonmirror" configuration, the ion boundary conditions are

$$F_a(\infty, \theta) = 0, \qquad (3.1)$$

$$\frac{\partial F_a}{\partial x}(0, \theta) = 0, \qquad (3.2)$$

$$\frac{\partial F_a}{\partial x}(x, 0) = 0, \qquad (3.3)$$

$$\frac{\partial F_a}{\partial x}(x, \pi/2) = 0. \qquad (3.4)$$

Given a mirror ratio R, we define θ_{LC} by

$$\sin^2\theta_{LC} = 1/R. \qquad (3.5)$$

We let

$$x_{ca}^2 = e\phi/\tfrac{1}{2} m_a \tilde{v}^2, \qquad (3.6)$$

and

$$\sin^2\theta_L = 1 \quad \text{if} \quad x^2 \le \frac{x_{ca}^2 Z_a}{R-1} \qquad (3.7)$$

$$= \frac{1 + \dfrac{x_{ca}^2}{x^2} Z_a}{R} \quad \text{if} \quad x^2 \ge \frac{x_{ca}^2 Z_a}{R-1}. \qquad (3.8)$$

We see that θ_L is a function of x. Conditions (3.2) and (3.3) are replaced by the requirement that $F_a(x, \theta_L(x)) = 0$; i.e., F_a is 0 on the loss-cone boundary.

The electron boundary conditions are

$$F_e(\infty) = 0, \quad (3.9)$$

$$\frac{\partial F_e}{\partial x}(0) = 0. \quad (3.10)$$

4. Construction of Finite Difference Mesh

The ion velocity mesh is represented as $\{x_j\}_{j=1}^{J}$, where $x_1 = 0$, $x_J = x_{max}$, and $x_{J'} = x'_{max}$ = ion x maximum. We let $h_1 = x'_{max}/(J' - 1)$, and $x_2 = f_1 h_1$, $0 \le f_1 \le 1$ (f_1 is input), and then compute r_1 satisfying

$$x_{j+1} - x_j = r_1 (x_j - x_{j-1}) \quad (4.1)$$

for $2 \le j < J'$. We then define

$$x_{J'+1} = 2x_{J'} - x_{J'-1} \quad (4.2)$$

and compute r_2 satisfying (4.1) for

$$J' + 1 \le j < J.$$

Thus we see that the electron velocity mesh is an extension of the ion velocity mesh, and the double geometricity enables us to represent low energy ions and high energy electrons on the same mesh.

The ion angular mesh is represented as $\{\theta_i\}_{i=1}^{I}$, where $\theta_1 = 0$ or θ_{LC} (depending on whether or not there is a mirror), and $\theta_I = \pi/2$. An extra point θ_{I+1} is added for reflection purposes.

5. Finite Differencing

Let

$$\tilde{t}_a = \tilde{v}^3/K_a \Gamma_a \quad (5.1)$$

and

$$\tau_a = t/\tilde{t}_a. \quad (5.2)$$

Then Eq. (1.3) may be written as

$$\frac{\partial F_a}{\partial \tau_a} = \frac{1}{x^2}\frac{\partial \overline{G}_a}{\partial x} + \frac{1}{x^2 \sin\theta}\frac{\partial H_a}{\partial \theta} + \tilde{t}_a S_a. \tag{5.3}$$

The term $\partial\overline{G}_a/\partial x$ is differenced in space (dropping the subscript "a") as

$$\frac{\partial \overline{G}_a}{\partial x} \approx \frac{\overline{A}_{i,j+1} F_{i,j+1} - \overline{A}_{i,j-1} F_{i,j-1}}{2\Delta x_j} + \frac{\left[\frac{\overline{B}_{i,j+\frac{1}{2}}(F_{i,j+1} - F_{i,j})}{\Delta x_{j+\frac{1}{2}}} - \frac{B_{i,j-\frac{1}{2}}(F_{i,j} - F_{i,j-1})}{\Delta x_{j-\frac{1}{2}}}\right]}{\Delta x_j}$$

$$+ \frac{1}{2\Delta x_j}\left[\overline{C}_{i,j+1}\frac{(F_{i+1,j+1} - F_{i-1,j+1})}{2\Delta\theta_i} - \overline{C}_{i,j-1}\frac{(F_{i+1,j-1} - F_{i-1,j-1})}{2\Delta\theta_i}\right], \tag{5.4}$$

where

$$\overline{B}_{i,j\pm\frac{1}{2}} = \frac{1}{2}(\overline{B}_{i,j} + \overline{B}_{i,j\pm 1}), \tag{5.5}$$

$$\Delta x_{j+\frac{1}{2}} = x_j - x_{j-1}, \tag{5.6}$$

$$\Delta x_j = \frac{1}{2}(\Delta x_{j-\frac{1}{2}} + \Delta x_{j+\frac{1}{2}}). \tag{5.7}$$

The $\partial\overline{H}_a/\partial\theta$ term, $\Delta x_{j-\frac{1}{2}}$, $\Delta\theta_j$, and $\Delta\theta_{j\pm\frac{1}{2}}$ are defined analogously.

In advancing from cycle n to cycle n + 1, the coefficients and the source are explicitly evaluated at cycle n. Splitting is used:

$$\frac{F_a^{(n+1)'} - F_a^n}{\Delta\tau_a} = \frac{1}{x_j^2}\frac{\partial \overline{G}_a}{\partial x} + \frac{1}{2}\tilde{t}_a S_a, \tag{5.8}$$

$$\frac{\partial \overline{G}_a}{\partial x} = \frac{\partial}{\partial x}\left[\overline{A}_a F_a^{(n+1)'} + \overline{B}_a \frac{\partial F_a^{(n+1)'}}{\partial x} + \overline{C}_a \frac{\partial F_a^n}{\partial \theta}\right], \tag{5.9}$$

$$\frac{F_a^{(n+1)} - F_a^{(n+1)'}}{\Delta\tau_a} = \frac{1}{x_j^2 \sin\theta_i}\frac{\partial \overline{H}_a}{\partial \theta} + \frac{1}{2}\tilde{t}_a S_a, \tag{5.10}$$

$$\frac{\partial \overline{H}_a}{\partial \theta} = \frac{\partial}{\partial \theta}\left[\overline{D}_a F_a^{(n+1)} + \overline{E}_a \frac{\partial F_a^{(n+1)'}}{\partial x} + \overline{F}_a \frac{\partial F_a^{(n+1)}}{\partial \theta}\right]; \tag{5.11}$$

i. e., all terms except those of mixed second derivative type are implicit.

The solution of Eqs. (5.8)-(5.11) involves sweeping the difference mesh along x = constant lines and θ = constant lines. The procedure is adequately described in UCRL-71662.

In the event of a mirror configuration and nonzero ambipolar potential, the finite differencing stops at the loss boundary. The loss boundary is represented by approximating $\theta_L(x)$ (see Sec. 3) by the nearest mesh point. An inverse function $x_L(\theta)$ is defined, also approximated to the nearest mesh point.

The electron equation (1.20) is also differenced implicitly. The technique is merely a restriction of the ion equation integrator to one dimension.

6. Source Terms

The source term can be represented as

$$S_a(x, \theta) = \sum_{\ell} \frac{1}{K_a} J_a^{\ell} \tilde{S}_a^{\ell}(x, \theta), \tag{6.1}$$

where J_a^{ℓ} is a current, and $\tilde{S}_a^{\ell}$ is a shape function of density 1, satisfying

$$\tilde{S}_a^{\ell}(x, \theta) = a' e^{-b'(x-c')^2} e^{-d'(\cos\theta - e')^2}. \tag{6.2}$$

The constants a' - e' are allowed to vary for each source and for each species. Clearly $d' = 0$ for electron sources. The current J_a^{ℓ} is of the form

$$J_a^{\ell} = A_a^{\ell} + B_a^{\ell}\sqrt{n_a} + C_a^{\ell} n_a, \tag{6.3}$$

where n_a is the density.

7. Initial Distributions

The initial value of $F_a(x, \theta)$ is given by

$$F_a(x, \theta) = a' e^{-b'(x-c')^2} e^{-d'(\cos\theta - e')^2} \tag{7.1}$$

where c' corresponds to the initial energy, and a' is computed so that F_a has the correct initial density n_a. The constant K_a is strongly related to a'. Clearly, $d' = 0$ for electrons.

The density n_a is defined by

$$n_a = 4\pi K_a \int_0^{\pi/2} \int_0^\infty F_a(x, \theta) x^2 \sin\theta \, dx \, d\theta, \tag{7.2}$$

and the energy E_a by

$$E_a = \frac{1}{2} m_a \tilde{v}^2 \langle x^2 \rangle_a, \tag{7.3}$$

where

$$\langle x^2 \rangle_a = \frac{\int_0^{\pi/2} \int_0^\infty F_a(x, \theta) x^4 \sin\theta \, dx \, d\theta}{\int_0^{\pi/2} \int_0^\infty F_a(x, \theta) x^2 \sin\theta \, dx \, d\theta}. \tag{7.4}$$

8. Potential as a Function of Magnetic Field

For a mirror system, we compute $\phi(\psi)$, where $1 \le \psi \le R$, and $\phi(1) = 0$, $\phi(R) = \overline{\Phi}$ (denoted earlier as ambipolar potential). We let

$$n(\psi, \phi) = \sum_{ions} Z_a n_a(\psi, \phi), \tag{8.1}$$

$$n_a(\psi, \phi) = 4\pi K_a \int_0^\infty x\sqrt{x^2 + x_{ca}^2} \, dx \int_{\frac{x}{\sqrt{x^2+x_{ca}^2}}\sqrt{\frac{\psi}{R}}}^{\min\left\{1, \frac{x}{\sqrt{x^2+x_{ca}^2}}\sqrt{\psi}\right\}} \frac{s \, ds}{\sqrt{1-s^2}} F_a(x, \theta), \tag{8.2}$$

where

$$x_{ca}^2 = Z_a e\phi / \tfrac{1}{2} m_a \tilde{v}^2 \tag{8.3}$$

and

$$\mu^2 = \cos^2\phi = 1 - \frac{s^2(x^2 + x_{ca}^2)}{\psi x^2}. \tag{8.4}$$

Also,

$$n_e(\psi, \phi) = 4\pi K_e \left[\int_{x_{ce}}^{\bar{x}_{ce}} x^2\, dx\, F_e(x) \sqrt{1 - x_{ce}^2/x^2} \right.$$

$$\left. + \int_{\bar{x}_{ce}}^{\infty} x^2\, dx\, F_e(x) \sqrt{1 - \frac{x_{ce}^2}{x^2} + \frac{\psi}{R}\left(\frac{\bar{x}_{ce}^2}{x^2} - 1\right)} \right], \tag{8.5}$$

where

$$x_{ce}^2 = e\phi / \tfrac{1}{2} m_e \tilde{v}^2 \tag{8.6}$$

and

$$\bar{x}_{ce}^2 = e\bar{\phi} / \tfrac{1}{2} m_e \tilde{v}^2. \tag{8.7}$$

Given ψ, the idea is to find ϕ satisfying

$$n_i(\psi, \phi) = n_e(\psi, \phi), \tag{8.8}$$

and we let $\phi = \phi^*(\psi)$.

Pressure terms are defined similarly. We let

$$P_{\perp,\parallel}(\psi) = \sum_{\text{ions}} P_{\perp,\parallel}^{a}(\psi) + P_{\perp,\parallel}^{e}(\psi). \tag{8.9}$$

Here, $P_{\perp}^{a}(\psi)$ is defined like $n_a(\psi, \phi^*(\psi))$, but a factor $\frac{1}{2} m_a \tilde{v}^2 \psi x^2(1-\mu^2)$ is added to the integrand for the perpendicular case, and a factor $m_a \tilde{v}^2 \left[x^2(1-\psi(1-\mu^2)) + \phi^* Z_a / \frac{1}{2} m_a \tilde{v}^2\right]$ for the parallel case. The electron term $P_{\perp,\parallel}^{e}(\psi)$ drops the second term in (8.5), and a factor $\frac{1}{3} m_e \tilde{v}^2 \left(x^2 - e\phi^* / \frac{1}{2} m_e \tilde{v}^2\right)$ is added to the integrand in the first term.

A few other diagnostic quantities are computed, such as

$$K_m = \inf \left[\frac{-\frac{\partial}{\partial \psi}(\psi^2)}{\frac{\partial}{\partial \psi}(P_{\perp}(\psi))} \right]_{P_{\perp}(\psi) \leq 0}, \tag{8.10}$$

$$L_m = \inf \left[\frac{\frac{\partial}{\partial \psi}(\psi^2)}{\frac{\partial}{\partial \psi}(P_{\parallel}(\psi))} \right]_{P_{\parallel}(\psi) \geq 0}, \tag{8.11}$$

$$\psi_{vac}(\psi) = \sqrt{\psi^2 + K_m P_{\perp}(\psi)}, \tag{8.12}$$

$$R_{vac} = R/\psi_{vac}(1), \tag{8.13}$$

and

$$\beta_{max} = K_m P_{\perp}(1)/(1 + K_m P_{\perp}(1)). \tag{8.14}$$

9. Charge Neutrality Requirements

For mirror systems, a self-consistent ambipolar potential at the mirror, ϕ is computed at each time step by requiring charge neutrality. We let

$$n_+(t) = \sum_{ions} n_a(t) Z_a, \tag{9.1}$$

$$n_-(t) = n_e(t). \tag{9.2}$$

Suppose we're advancing the ion and electron equations from t to t + Δt. Let

$$d_+ = n_+(t + \Delta t) - n_+(t), \tag{9.3}$$

$$d_- = n_-(t + \Delta t) - n_-(t). \tag{9.4}$$

If $n_+(t) > n_-(t)$ and $d_+ \geq d_-$, we raise ϕ by an amount $\Delta\phi$ and readvance the distributions from t to t + Δt. If $n_+(t) < n_-(t)$ and $d_+ \leq d_-$, we lower ϕ by an amount $\Delta\phi$ and readvance the distributions from t to t + Δt. This procedure is repeated until the above test is negative. If the neutrality requirements are not met after a certain number of tries, the calculation is stopped.

10. Numerical Integration

We define c_j and d_i satisfying

$$\int_{x_1}^{x_J} f(x_j)\, dx_j \approx \sum_{j=1}^{J} c_j f(x_j), \tag{10.1}$$

$$\int_{\theta_1}^{\pi-\theta_1} f(\theta_i)\, d\theta_i \approx \sum_{i=1}^{I} d_i f(\theta_i). \tag{10.2}$$

Note that in Eq. (10.2) θ_i ranges from θ_1 to $\pi/2$, and θ_1 is either 0 or θ_{LC}.

For two-dimensional integration,

$$\iint f(x, \theta)\, dx d\theta \approx \sum_{i,j} e_{i,j} f(x_j, \theta_i), \tag{10.3}$$

where

$$e_{i,j} = 2\pi\, x_j^2 \sin \theta_i\, c_j\, d_i. \tag{10.4}$$

Uneven Simpson integration is used. That is, if $x_{j\pm1} = x_j \pm h_\pm$,

$$\int_{x_{j-1}}^{x_{j+1}} f(x)\, dx \approx a_1 f(x_{j-1}) + a_2 f(x_j) + a_3 f(x_{j+1}), \tag{10.5}$$

where

$$a_1 = \frac{2h_-^3 - h_+^3 + 3h_-^2 h_+}{6h_-(h_+ + h_-)}, \tag{10.6}$$

$$a_2 = \frac{h_+^3 + h_-^3 + 3h_+ h_- (h_+ + h_-)}{6h_+ h_-}, \tag{10.7}$$

$$a_3 = \frac{2h_+^3 - h_-^3 + 3h_+^2 h_-}{6h_+ (h_+ + h_-)}. \tag{10.8}$$

11. Reaction Cross Section

The reaction cross section between two species is defined as

$$\langle \sigma v \rangle_{ab} = \frac{\int_0^\infty ds \int_0^s u^2\, du \int_0^u I_1(s,t) \sigma_{ab}(E)\, dt}{\int_0^\infty ds \int_0^s u du \int_0^u I_1(s,t)\, dt}, \tag{11.1}$$

where

$$I_1(s,t) = (s^2 - t^2) \left[f_a\left(\frac{s+t}{2}\right) f_b\left(\frac{s-t}{2}\right) + f_a\left(\frac{s-t}{2}\right) f_b\left(\frac{s+t}{2}\right) \right], \tag{11.2a}$$

$$E = \frac{\frac{1}{2} m_a t^2}{1.602 \times 10^{-9}}. \tag{11.2b}$$

An approximation to (11.1), where a = deuterium (D) and b = tritium (T), assuming there exists a tritium distribution of the same shape and energy as the deuterium distribution, is

$$\langle \sigma v \rangle_{DT} = \frac{\sqrt{\frac{5}{3}} \int_0^\infty v^3 \ \sigma_{DT}(E)\, f_D(v)\, dv}{\int_0^\infty v^2\, f_D(v)\, dv}, \tag{11.3}$$

where

$$E = \frac{\frac{5}{6} m_D v^2}{1.602 \times 10^{-9}}. \tag{11.4}$$

The quantity $\sigma_{DT}(E)$ is given by

$$10^{24} \times \sigma = 2.25 \times 10^4 \exp(-44.4/\sqrt{E})/E \text{ if } E < 20, \tag{11.5a}$$

$$10^{24} \times \sigma = 4.62 \times 10^{-4} E^2 - 0.137 \text{ if } 20 \le E < 30, \quad (11.5b)$$

$$10^{24} \times \sigma = 6.7 \times 10^{-4} E^2 - 0.32 \text{ if } 30 \le E < 50, \quad (11.5c)$$

$$10^{24} \times \sigma = 8.55 \times 10^{-2} E - 2.93 \text{ if } 50 \le E < 87, \quad (11.5d)$$

$$10^{24} \times \sigma = \frac{1.98 \times 10^4}{(E - 107)^2 + 4000} \text{ if } 87 \le E < 107, \quad (11.5e)$$

$$10^{24} \times \sigma = \frac{3.56 \times 10^4}{(E - 107)^2 + 7200} \text{ if } 107 \le E < 137, \quad (11.5f)$$

$$10^{24} \times \sigma = 8.7 - 0.0314\, E \text{ if } 137 \le E < 185, \quad (11.5g)$$

$$10^{24} \times \sigma = 7.05 \times 10^3 / E^{3/2} \text{ if } 185 \le E < 400, \quad (11.5h)$$

$$10^{24} \times \sigma = 255/(E - 110) \text{ if } 400 \le E. \quad (11.5i)$$

The quantities $f_a(v)$ appearing in Eqs. (11.1) and 11.3) are defined as

$$f_a(v) = \frac{1}{2} \int_0^{\pi} f_a(v, \theta) \sin \theta \, d\theta. \quad (11.6)$$

12. Role of Subroutines

The code is controlled from MAIN. Subroutine INITIAL sets up initial data; subroutine TLOOP controls beam turnoff and time-step changes; subroutine RESTART controls the code restart procedure; subroutine TDSTORE stores data for time-dependent plots; subroutine GETR computes mesh ratios for geometric meshes; subroutine GAMMAIN computes $\ln \Lambda_{ab}$; subroutine SOURCEE computes the source terms; subroutine FINIT computes the initial distribution functions; subroutine XINIT computes the meshes; subroutine EEPRINT controls printed output; subroutine EEPLOT controls plotted output; subroutine BOUNDARY computes the ion loss boundary; subroutine POTSHAPE computes $\phi(\psi)$; subroutine DENSINPT computes $n(\psi)$, $p_\perp(\psi)$, $P_\parallel(\psi)$; subroutine XSWEEP advances the difference equations; subroutine

COEF computes the coefficients; subroutine AMBI controls the ϕ computation; subroutine CHOPDST sets the ion distributions to 0 outside the loss boundary; subroutine GNANDE computes the densities and energies; subroutines QCALC, EVALS, and GETFG compute $<\sigma v>_{ab}$; subroutine FINDSIG computes σ (E); and subroutine SIGV1D computes $<\sigma v>_{DT}$ using the one-dimensional integral approximation.

13. Table of Parameters and Variables in Common

Variable	Function
IY	Number of θ mesh points.
JX	Number of electron x mesh points.
JXI	Number of ion x mesh points.
NEQ	Number of ion species.
NONCH	Number of simultaneously stored values in arrays for time-dependent plots.
MX	Number of terms in Legendre polynomial expansions.
LZ	Number of ψ mesh points.
NB	Number of ϕ mesh points.
NPTS	Number of mesh points in Chebychev quadrature in $\phi(\psi)$ calculation.
NSOR	Number of Gaussian sources per species.
KNSET	Number of problems to be run.
ND(2)	Indicates FR80 or HSP output.
XMAX	x_{max}.
XMAXI	x'_{max}.
ERATIO	R.
FGMMESH	f_1 (see Sec. 4).
NSTOP	Number of time steps (cycles).
NPRINT	Number of cycles between printouts.
NPLOT	Number of cycles between plots.
NRSTRT	Cycle number for code restart.
NCHEC	Number of cycles between time-dependent sampling of data for plots.
MIMAX	Maximum number of attempts to satisfy charge neutrality requirements.
NSL	*Zero for mirror problem, 1 otherwise.*
DTR	Δt.
VNORM	$\tilde{v}$.
DT1, DT2, DT3	Future values of Δt.
NSET1, 2, 3	Cycles for changing Δt.

Variable	Function
POTENT	Ambipolar potential ϕ.
DPOT	$\Delta\phi/\phi$.
TINJSTOP	Time of beam turnoff.
PLTANG	Angle for 3-D plots.
BETA Ø	Maximum allowable β_{max} (Sec. 8).
EMASS(K)	m_a.
ANUMB(K)	Z_a.
REDEN(K)	n_a.
ENIT(K)	Initial E_a.
XNIT(K)	b' (Sec. 7).
YNIT(K)	d' (Sec. 7).
VNIT(K)	e' (Sec. 7).
ESOR(KS, K)	Source energy.
XSOR(KS, K)	b' (Sec. 6).
YSOR(KS, K)	d' (Sec. 6).
USOR(KS, K)	e' (Sec. 6).
ASOR (KS, K)	A_a^ℓ (Sec. 6).
BSOR(KS, K)	B_a^ℓ (Sec. 6).
CSOR(KS, K)	C_a^ℓ (Sec. 6).
PPOT (L)	$\phi^*(\psi)$ array.
BMAG(L)	ψ array.
DNSIE (L)	$n(\psi)$ array.
PPERP(L)	$P_\perp(\psi)$ array.
PPAR(L)	$P_\parallel(\psi)$ array.
PSVAC(L)	$\psi_{vac}(\psi)$ array.
PI	$\pi = 3.1415926536$.
CHARGE	$e = 4.803 \times 10^{10}$.
TIME	t.
N	Cycle number.
ERGTKEV	1.602×10^{-9}.
EXT (3, 4)	Extrapolation constants for boundary values.
KBOUND	Flag for error in BOUNDARY.
KGETGFK	Equals zero before COEF is entered.
KAMBI	Flag for charge neutrality requirements not satisfied.
KGNANDE	Flag for zero density.
BRATIO1	1/R.
KSINGUL	Flag for geometric mesh error.
JO(I, K)	$x_L(\theta)$.
IO(I, K)	$\theta_L(x)$.
JMIN(K)	Minimum $x_L(\theta)$.

Variable	Function
IMIN(K)	Minimum θ_L (x).
TNORM(K)	$\tilde{t}_a$.
RATIOM(K, L)	m_a/m_b.
ANORMK(K)	K_a.
ENERGY(K)	E_a.
GAM2(K, L)	Constant part of $\ln \Lambda_{ab}$.
GAMMA(K, L)	$\ln \Lambda_{ab}$.
RATK(K, L)	K_a/K_b.
EIONS(K)	$\frac{1}{2} m_a \tilde{v}^2/1.602 \times 10^9$.
RATIOZ2 (K, L)	Z_a^2/Z_b^2
CINT(I)	θ integration constants.
CINT(J)	x integration constants.
SPHERI(I, J)	x-θ integration constants.
XI(J)	$1/x$.
X(J)	x.
X2I(J)	$1/x^2$.
X3I(J)	$1/x^3$.
XSQ(J)	x^2.
DXM5(J)	$\Delta x_{j-\frac{1}{2}}$.
DX(J)	Δx_j.
Y(I)	θ.
DXI(J)	$1/\Delta x_j$.
DYM5(I)	$\Delta\theta_{i-\frac{1}{2}}$.
DY(I)	$\Delta\theta_i$.
DYI(I)	$1/\Delta\theta_i$.
CTNY(I)	$\cot\theta$.
COSS(I)	$\cos\theta$.
SINN(I)	$\sin\theta$.
DXP5(J)	$\Delta x_{j+\frac{1}{2}}$.
DYP5(I)	$\Delta\theta_{i+\frac{1}{2}}$.
XM(J, K)	Powers of x.
COG(M, 15)	Coefficients for Rosenbluth potentials.
PLEG(M, I)	P_j (Legendre polynomials)
DPLEG(M, I)	P_j'.
DDPLEG(M, I)	P_j''.
DDDPLEG(M, I)	P_j'''
EMQF(J, M)	$M_j(f)$.
ENQF(J, M)	$N_j(f)$
ERQF(J, M)	$R_j(f)$
EEQF(J, M)	$E_j(f)$.
PTIME(N)	Time array.

Variable	Function
PDENS(N, K)	Time-dependent density array.
PENGY(N, K)	Time-dependent energy array.
PPOTENT(N)	Time-dependent potential array.
FE(J)	$F_e(x)$.
SORE(J)	$S_e(x)$.
AELL(J)	$\bar{A}_e$.
BELL(J)	$\bar{B}_e$.
CELL(J)	Part of $\bar{C}_e$ independent of λ.
F(I, J, K)	$F_a(x, \theta)$.
SOURCE (I, J, K)	$S_a(x, \theta)$.
CAL - CFL(I, J, K)	$\bar{A}_a(x, \theta) - \bar{F}_a(x, \theta)$.

14. Program Listing

```
UNCLASSIFIED                                    XEROX                  08150

BOX U16




BBBBBB    OOOOOOO   X     X          U     U      1        6666
 B    B   O     O    X   X           U     U     11       6
 B    B   O     O     X X            U     U      1      6
 BBBBB    O     O      X             U     U      1      666666
 B    B   O     O     X X            U     U      1      6     6
 B    B   O     O    X   X           U     U      1      6     6
BBBBBB    OOOOOOO   X     X           UUUUU      111      66666




U     U   N     N    CCCCC    L
U     U   NN    N   C     C   L
U     U   N N   N   C         L
U     U   N  N  N   C         L
U     U   N   N N   C         L
U     U   N    NN   C     C   L
 UUUUU    N     N    CCCCC    LLLLLLL




X     X   EEEEEEE   RRRRRR    OOOOOOO   X     X
 X   X    E         R     R   O     O    X   X
  X X     E         R     R   O     O     X X
   X      EEEE      RRRRRR    O     O      X
  X X     E         R  R      O     O     X X
 X   X    E         R   R     O     O    X   X
X     X   EEEEEEE   R    R    OOOOOOO   X     X
```

```
      PROGRAM DUMFG(TAPE2,TAPE6,TAPE3=TAPE6,PRINT)
      CLICHE COMM
      PARAMETER(IY=20,JX=201)
      PARAMETER(JXI=45)
      PARAMETER(NEQ=2)
      PARAMETER(MX=4)
      PARAMETER(MMX=2*MX,MXP3M=-MMX-3,MXP4=MMX+4)
      PARAMETER(NONCH=200)
      PARAMETER(IYP1=IY+1,IYP2=IY+2,JXP1=JX+1)
      PARAMETER(IYJX=IY+JX)
      PARAMETER(IYP1JX=IYP1*JXI)
      PARAMETER(LZ=25,NB=25,NPTS=200)
      PARAMETER(NSOR=1)
      COMMON/MLZU/PPOT(LZ),BMAG(LZ),PS1(LZ),PS2(LZ),DNSIE(LZ),
     C PPERP(LZ),PPAR(LZ),PSVAC(LZ),DNSI(NB),DNSE(NB),DDIFF(NB),PHI(NB),
     C VINT(JXI),VINTPE(JXI),VINTPA(JXI),VENT(JX),UMMU(IY),COSYM(NPTS)
      COMMON/MCON1/PI,TWOPI,FOURPI,PIO2,CHARGE,TIME,DTR,DT1,D)  DT3,
     C NSET1,NSET2,NSET3,N,NPRINT,NPLOT,NSTOP,NRSTRT,NCH,NCHEC,
     C IYM1,IYM2,IYM3,JXM1,JXM2,JXM3,JXIM,JXIM2,JXIM3,ERGTKEV,
     C POTENT,DPOT,XMAX,XMAX1,EXT(3,4),KBOUND,KGETGFK,KAMBI,
     C KGNANDE,MIMAX,BRATIO,BRATIOI,PLTANG,KSINGUL,VNORM
     C ,KNPROBS,NSL,TINJSTOP,BETA0
      COMMON/MCON2/JO(IY,NEQ),IO(JX,NEQ),JMIN(NEQ),IMIN(NEQ),
     C TNORM((0,NEQ)),RATIOM((0,NEQ),(0,NEQ)),EMASS((0,NEQ)),
     C ANUMB((0,NEQ)),ANORMK((0,NEQ)),DENS((0,NEQ)),REDEN((0,NEQ)),
     C ENERGY((0,NEQ)),ENGY((0,NEQ)),DENSTOR((0,NEQ)),
     C ESOR(NSOR,(0,NEQ)),XSOR(NSOR,(0,NEQ)),YSOR(NSOR,(0,NEQ)),
     C USOR(NSOR,(0,NEQ)),ASOR(NSOR,(0,NEQ)),BSOR(NSOR,(0,NEQ)),
     C CSOR(NSOR,(0,NEQ)),ENIT((0,NEQ)),XNIT((0,NEQ)),
     C YNIT((0,NEQ)),VNIT((0,NEQ)),GAM2((0,NEQ),(0,NEQ)),
     C GAMMA((0,NEQ),(0,NEQ)),RATK((0,NEQ),(0,NEQ)),EIONS((0,NEQ)),
     C RATIOZ2((0,NEQ),(0,NEQ))
      COMMON/MCON3/CYNT(IY),CINT(JX),SPHER1(IY,JX),XI(JX),TEMPG(JX),
     C X(JX),X2I(JX),X3I(JX),XSQ(JX),DXM5(JXP1),DX(JX),Y(IYP1),DXI(JX),
     C DYM5(IYP2),DY(IY),DYI(IY),CTNY(IY),COSS(IY),SINN(IY),
     C EP(IYJX),FP(IYJX)
      DIMENSION DXP5(JX),DYP5(IYP1)
      EQUIVALENCE(DXP5(1),DXM5(2))
      EQUIVALENCE(DYP5(1),DYM5(2))
      COMMON/MCON4/XM(JX,(MXP3M,MXP4)),COG((0,MX),15),PLEG((0,MMX),IY),
     C DPLEG((0,MMX),IY),DDPLEG((0,MMX),IY),DDDPLEG((0,MMX),IY),
     C EMQF(JX,(0,MX)),ENQF(JX,(0,MX)),ERQF(JX,(0,MX)),EEQF(JX,(0,MX))
      COMMON/MTEMP/FT(IYP1,JXI),FN(IYP1,JXI),FO(IYP1,JXI),SO(IYP1,JXI),
     C CA(IYP1,JXI),CB(IYP1,JXI),CC(IYP1,JXI),CD(IYP1,JXI),
     C CE(IYP1,JXI),CF(IYP1,JXI)
      COMMON/MPLOT/PTIME(NONCH),PDENS(NONCH,(0,NEQ)),
     C PENGY(NONCH,(0,NEQ)),PPOTENT(NONCH)
      COMMON/MILEC/FE(JX),FE1(JX),SORE(JX),AEL(JX),BEL(JX),CCEL(JX),
     C AELL(JX),BELL(JX),CELL(JX)
      LCM (MLCMC)
      COMMON/MLCMC/F(IYP1,JXI,NEQ),F1(IYP1,JXI,NEQ),F2(IYP1,JXI,NEQ),
     C SOURCE(IYP1,JXI,NEQ),CAL(IYP1,JXI,NEQ),CBL(IYP1,JXI,NEQ),
     C CCL(IYP1,JXI,NEQ),CDL(IYP1,JXI,NEQ),CEL(IYP1,JXI,NEQ),
     C CFL(IYP1,JXI,NEQ)
      ENDCLICHE
      USE COMM
      CALL CRTID(2HAM,1,1) $ CALL FRAME
      CALL ASSIGN(2,0,7RINTIONS,0)
      READ (2,2) KNSET
    2 FORMAT(I5)
```

```
      DIMENSION ND(5) $ ND(1)=0 $ ND(3)=6 $ ND(4)=0
      READ (2,46) ND(2)
   46 FORMAT(A10)
      ND(5)=1 $ CALL HSP(ND)
      DO 1 KNPROBS=1,KNSET
      KGNANDE=0 $ KBOUND=0 $ KGETGFK=0 $ KAMBI=0
      KSINGUL=0
      WRITE (3,3) KNPROBS
    3 FORMAT(1H1,27HBEGINNING OF PROBLEM NUMBER,I5,//)
      CALL INITIAL
      IF (KSINGUL .EQ. 1) GO TO 1
      CALL EEPRINT
      CALL EEPLOT
      IF (KBOUND .EQ. 1) GO TO 1
      IF (KGNANDE .EQ. 1) GO TO 1
      IF (NSTOP .EQ. 0) GO TO 1
   11 N=N+1
      CALL TLOOP
      CALL GAMMAIN
      CALL SOURCEE
      CALL COEF
      CALL AMBI
      IF (KGNANDE .EQ. 1 .OR. KAMBI .EQ. 1 .OR. KBOUND .EQ. 1) GO TO 19
      CALL TDSTORE
      IF (N .GT. NPRINT*(N/NPRINT)) GO TO 12
      CALL EEPRINT
   12 IF (N .GT. NPLOT*(N/NPLOT)) GO TO 13
      CALL EEPLOT
   13 IF (N .LT. NSTOP) GO TO 11
   19 CONTINUE
      CALL EEPRINT
      CALL EEPLOT
      CALL POTSHAPE
      CALL DENSINPT
      CALL QCALC
    1 CONTINUE
      CALL PLOTE
      CALL QUIT(1)
      END
      SUBROUTINE INITIAL
      USE COMM
      N=0
      NCH=0
      TIME=0.
      READ (2,1) XMAX,XMAXI,BRATIO
    1 FORMAT(5E16.6)
    2 FORMAT(16I5)
      READ (2,2) NSTOP,NPRINT,NPLOT,NRSTRT
      READ (2,2) NCHEC,MIMAX
      READ (2,2) NSL
      READ (2,1) DTR,VNORM
      READ (2,1) DT1,DT2,DT3
      READ (2,2) NSET1,NSET2,NSET3
      READ (2,1) POTENT,DPOT
      READ (2,1) TINJSTOP
      READ (2,1) PLTANG
      READ (2,1) BETA0
      DO 3 K=0,NEQ
      READ (2,1) EMASS(K),ANUMB(K),REDEN(K)
      READ (2,1) ENIT(K),XNIT(K),YNIT(K),VNIT(K)
```

```
      DO 41 KS=1,NSOR
      READ (2,1) ESOR(KS,K),XSOR(KS,K),YSOR(KS,K),USOR(KS,K)
      READ (2,1) ASOR(KS,K),BSOR(KS,K),CSOR(KS,K)
 41   CONTINUE
  3   CONTINUE
      WRITE (3,11) XMAX,XMAXI,BRATIO
 11   FORMAT(5HXMAX=,E16.6,5X,6HXMAXI=,E16.6,5X,7HBRATIO=,E16.6)
      WRITE (3,12) NSTOP,NPRINT,NPLOT,NRSTRT
 12   FORMAT(6HNSTOP=,I5,5X,7HNPRINT=,I5,5X,6HNPLOT=,I5,5X,7HNRSTRT=,I5)
      WRITE (3,7) NCHEC,MIMAX
  7   FORMAT(6HNCHEC=,I5,5X,6HMIMAX=,I5)
      WRITE (3,39) NSL
 39   FORMAT(4HNSL=,I5)
      WRITE (3,13) DTR,VNORM
 13   FORMAT(4HDTR=,E16.6,5X,6HVNORM=,E16.6)
      WRITE (3,14) DT1,DT2,DT3,NSET1,NSET2,NSET3
 14   FORMAT(4HDT1=,E16.6,5X,4HDT2=,E16.6,5X,4HDT3=,E16.6,/,6HNSET1=,
     C I5,5X,6HNSET2=,I5,5X,6HNSET3=,I5)
      WRITE (3,15) POTENT,DPOT
 15   FORMAT(21HAMBIPOLAR POTENTIAL =,E16.6,5X,5HDPOT=,E16.6)
      WRITE (3,64) TINJSTOP
 64   FORMAT(9HTINJSTOP=,E15.6)
      WRITE (3,24) PLTANG
 24   FORMAT(7HPLTANG=,E16.6)
      WRITE (3,31) BETA0
 31   FORMAT(6HBETA0=,E16.6)
      DO 4 K=0,NEQ
      WRITE (3,16) K,EMASS(K),ANUMB(K),REDEN(K)
 16   FORMAT(//,2HK=,I5,//,5HMASS=,E16.6,5X,2HZ=,E16.6,5X,8HDENSITY=,
     C E16.6)
      WRITE (3,17) ENIT(K),XNIT(K),YNIT(K),VNIT(K)
 17   FORMAT(5HENIT=,E16.6,5X,5HXNIT=,E16.6,5X,5HYNIT=,E16.6,5X,
     C 5HVNIT=,E16.6)
      DO 42 KS=1,NSOR
      WRITE (3,43) KS
 43   FORMAT(/,10HSOURCE NO ,I5,/)
      WRITE (3,18) ESOR(KS,K),XSOR(KS,K),YSOR(KS,K),USOR(KS,K)
 18   FORMAT(5HESOR=,E16.6,5X,5HXSOR=,E16.6,5X,5HYSOR=,E16.6,5X,
     C 5HUSOR=,E16.6)
      WRITE (3,19) ASOR(KS,K),BSOR(KS,K),CSOR(KS,K)
 19   FORMAT(5HASOR=,E16.6,5X,5HBSOR=,E16.6,5X,5HCSOR=,E16.6)
 42   CONTINUE
  4   CONTINUE
      PI=3.1415926536 $ PIO2=.5*PI $ TWOPI=PI+PI $ FOURPI=TWOPI+TWOPI
      CHARGE=4.803E-10 $ ERGTKEV=1.602E-9
      DO 21 K=0,NEQ
      EIONS(K)=.5*EMASS(K)*VNORM**2/ERGTKEV
      DO 21 I=0,NEQ
      RATIOZ2(I,K)=(ANUMB(I)/ANUMB(K))**2
 21   RATIOM(I,K)=EMASS(I)/EMASS(K)
      CALL XINIT
      IF (KSINGUL .EQ. 1) RETURN
      CALL FINIT
      CALL BOUNDARY
      IF (KBOUND .EQ. 1) RETURN
      CALL CHPDST
      CALL GNANDE
      IF (KGNANDE .EQ. 1) RETURN
      DO 23 K=0,NEQ
      GAM1=FOURPI*(ANUMB(K)*CHARGE)**4/EMASS(K)**2
```

```
 23 TNORM(K)=VNORM**3/(ANORMK(K)*GAM1)
    DO 301 I=0,NEQ $ DO 301 K=0,NEQ
    RATK(I,K)=ANORMK(I)/ANORMK(K)
301 CONTINUE
    RETURN
    END
    SUBROUTINE TLOOP
    USE COMM
    IF (N .EQ. NRSTRT) GO TO 31
    IF (SENSE SWITCH 1) 31,32
 31 CALL RESTART
 32 CONTINUE
    IF (ABSF(TIME-TINJSTOP) .GT. .5*DTR) GO TO 21
    DO 51 KS=1,NSOR
    ASOR(KS,0)=0. $ BSOR(KS,0)=0.
    DO 22 K=1,NEQ
    ASOR(KS,K)=0. $ BSOR(KS,K)=0. $ CSOR(KS,K)=0.
 22 CONTINUE
 51 CONTINUE
    CALL POTSHAPE
    CALL DENSINPT
    CALL QCALC
 21 CONTINUE
    IF (N .EQ. NSET1) DTR=DT1
    IF (N .EQ. NSET2) DTR=DT2
    IF (N .EQ. NSET3) DTR=DT3
    TIME=TIME+DTR
    RETURN
    END
    SUBROUTINE RESTART
    USE COMM
    CALL PLOTE
    CALL ASSIGN(2,0,7RINTIONS,-1)
    CALL OFFMON
    CALL ASSIGN(2,0,7RINTIONS,0)
    READ (2,2) NSTOP,NPRINT,NPLOT,NRSTRT
  2 FORMAT(16I5)
    READ (2,2) NSET1,NSET2,NSET3
    READ (2,1) DT1,DT2,DT3
  1 FORMAT(5E16.6)
    DIMENSION ND(5) $ ND(1)=0 $ ND(3)=6 $ ND(4)=0
    READ (2,3) ND(2)
  3 FORMAT(A10)
    ND(5)=1 $ CALL HSP(ND)
    CALL CRTID(2HAM,1,1)
    WRITE (3,4) NSTOP,NPRINT,NPLOT,NRSTRT,NSET1,NSET2,NSET3,DT1,
   C DT2,DT3
  4 FORMAT(10HNEW VALUES,//,6HNSTOP=,I5,5X,7HNPRINT=,I5,5X,6HNPLOT=,
   C I5,5X,7HNRSTRT=,I5,/,6HNSET1=,I5,5X,6HNSET2=,I5,5X,6HNSET3=,I5,
   C /,4HDT1=,E16.6,5X,4HDT2=,E16.6,5X,4HDT3=,E16.6,/)
    RETURN
    END
    SUBROUTINE TDSTORE
    USE COMM
    IF (N .GT. NCHEC*(N/NCHEC)) RETURN
    IF (NCH .EQ. NONCH) NCH=0
    NCH=NCH+1
    PTIME(NCH)=TIME
    DO 1 K=0,NEQ
    PDENS(NCH,K)=REDEN(K) $ PENGY(NCH,K)=ENERGY(K)
```

```
    1 CONTINUE
      PPOTENT(NCH)=POTENT
      RETURN
      END
      SUBROUTINE GETR(JX,XMAX,H,R,KSINGUL)
      JXM1=JX-1 $ KMAX=100
      KSINGUL=0
      TESTH=XMAX/FLOATF(JXM1)
      R=1.
      IF (H .GT. (1.-1.E-14)*TESTH) RETURN
      A1=XMAX/H $ A2=1.-A1 $ A3=1./FLOATF(JXM1)
      K=0 $ GUESS=100.
    1 K=K+1
      IF (GUESS .LT. 0.) GUESS=-GUESS
      GUESS1=A1*GUESS**A3+A2
      IF (ABSF(GUESS-GUESS1) .LT. 1.E-12) GO TO 2
      IF (K .EQ. KMAX) GO TO 3
      GUESS=GUESS1 $ GO TO 1
    3 WRITE (3,4) GUESS1
    4 FORMAT(36HTOO MANY GUESSES REQUIRED - R**JXM1=,E16.6)
      KSINGUL=1 $ RETURN
    2 R=GUESS1**A3
      WRITE (3,5) R
    5 FORMAT(/,4HRAM=,E20.14,/)
      RETURN
      END
      SUBROUTINE GAMMAIN
      USE COMM
      IF (N .GT. 1) GO TO 1
      ALP=1./137. $ R0=2.82E-13 $ CLIGHT=3.E+10
      GACON1=2.*ALP/(SQRTF(3.*PI)*R0*CLIGHT)
      GACON2=GACON1/(3.7943E-4*EMASS(0))
      DO 2 I=0,NEQ $ DO 2 K=0,NEQ
      GAM2(I,K)=EMASS(I)*EMASS(K)*GACON2/(EMASS(I)+EMASS(K))
    2 CONTINUE
    1 CONTINUE
      TEMPE=1.E+3*ENERGY(0) $ DENSE=REDEN(0)
      DEBY=SQRTF(TEMPE/DENSE)
      DO 3 I=0,NEQ $ DO 3 K=0,NEQ
      SI=ENERGY(I)/EMASS(I) $ SK=ENERGY(K)/EMASS(K)
      SF=SI $ IF (SK .GT. SI) SF=SK
      VIKF=SQRTF(SF*ERGTKEV)
      GAM3=GAM2(I,K)*DEBY*VIKF
      GAMMA(I,K)=LOGF(GAM3)-.5
    3 CONTINUE
      RETURN
      END
      SUBROUTINE SOURCEE
      USE COMM
      DO 1 K=1,NEQ
      DO 31 I=1,IYP1 $ DO 31 J=1,JX1
   31 FT(I,J)=0.
      DO 32 KS=1,NSOR
      XOD2=ESOR(KS,K)/EIONS(K) $ XOD=SQRTF(XOD2)
      XUSE=XSOR(KS,K)
      IF (ESOR(KS,K) .EQ. 0.) XUSE=1.5*EIONS(K)/XSOR(KS,K)
      DO 2 J=1,JX1
      FACX=-XUSE*(X(J)-XOD)**2
      DO 3 I=1,IY
      FACY=-YSOR(KS,K)*(COSS(I)-USOR(KS,K))**2
```

```
      SO(I,J)=EXPF(FACX+FACY)
    3 CONTINUE
    2 CONTINUE
      S=0. $ DO 4 I=1,IY $ DO 4 J=1,JX1
    4 S=S+SO(I,J)*SPHERI(I,J)
      CUR=ASOR(KS,K)+BSOR(KS,K)*SQRTF(REDEN(K))+CSOR(KS,K)*REDEN(K)
      FACT=CUR/(S*ANORMK(K))
      DO 5 J=1,JX1
      DO 6 I=1,IY
    6 SO(I,J)=SO(I,J)*FACT
    5 SO(IYP1,J)=SO(IYM1,J)
      DO 33 I=1,IYP1 $ DO 33 J=1,JX1
   33 FT(I,J)=FT(I,J)+SO(I,J)
   32 CONTINUE
      CALL WRITELCM(FT,SOURCE(1,1,K),IYP1JX)
    1 CONTINUE
      K=0
      DO 34 J=1,JX
   34 FEI(J)=0.
      DO 35 KS=1,NSOR
      XOD2=ESOR(KS,K)/EIONS(K) $ XOD=SQRTF(XOD2)
      XUSE=XSOR(KS,K)
      IF (ESOR(KS,K) .EQ. 0.) XUSE=1.5*EIONS(K)/XSOR(KS,K)
      DO 12 J=1,JX
      FACX=-XUSE*(X(J)-XOD)**2
   12 SORE(J)=EXPF(FACX)
      S=0. $ DO 14 J=1,JX
   14 S=S+SORE(J)*XSQ(J)*CINT(J)
      S=FOURPI*S
      CUR=ASOR(KS,K)+BSOR(KS,K)*SQRTF(REDEN(K))+CSOR(KS,K)*REDEN(K)
      FACT=CUR/(S*ANORMK(K))
      DO 15 J=1,JX
   15 SORE(J)=FACT*SORE(J)
      DO 36 J=1,JX
   36 FEI(J)=FEI(J)+SORE(J)
   35 CONTINUE
      DO 37 J=1,JX
   37 SORE(J)=FEI(J)
      RETURN
      END
      SUBROUTINE FINIT
      USE COMM
      DO 1 K=1,NEQ
      XOD2=ENIT(K)/EIONS(K)
      XOD=SQRTF(XOD2)
      IF (ENIT(K) .EQ. 0.) XNIT(K)=1.5*EIONS(K)/XNIT(K)
      DO 2 J=1,JX1
      FACX=-XNIT(K)*(X(J)-XOD)**2
      DO 3 I=1,IY
      FACY=-YNIT(K)*(COSS(I)-VNIT(K))**2
      FN(I,J)=EXPF(FACX+FACY)
      IF (J .EQ. JX1) FN(I,J)=0.
    3 CONTINUE
    2 FN(IYP1,J)=FN(IYM1,J)
      CALL WRITELCM(FN,F1(1,1,K),IYP1JX)
    1 CALL WRITELCM(FN,F(1,1,K),IYP1JX)
      K=0
      XOD2=ENIT(K)/EIONS(K) $ XOD=SQRTF(XOD2)
      IF (ENIT(K)  EQ. 0) XNIT(K)=1.5*EIONS(K)/XNIT(K)
      DO 12 J=1,JX
```

```
      FACX=-XNIT(K)*(X(J)-XOO)**2
   12 FE(J)=EXPF(FACX)
      RETURN
      END
      SUBROUTINE XINIT
      USE COMM
      IYM1=IY-1 $ IYM2=IY-2 $ JXM1=JX-1 $ JXM2=JX-2
      IYM3=IY-3 $ JXM3=JX-3
      JXIM=JXI-1 $ JXIM2=JXI-2 $ JXIM3=JXI-3
      BRATIOI=1./BRATIO
      THETAL=ACOSF(SQRTF(1.-BRATIOI))
      IF (NSL .EQ. 0) THETAL=0.
      Y(1)=THETAL $ Y(IY)=PIO2
      HY2=(PIO2-THETAL)/FLOATF(IYM1)
      DO 7 I=2,IYM1
    7 Y(I)=Y(I-1)+HY2
      Y(IYP1)=2.*Y(IY)-Y(IYM1)
      X(1)=0. $ X(JX)=XMAX
      X(JXI)=XMAXI $ HX=XMAXI/JXIM
      DO 8 J=2,JXIM
    8 X(J)=X(J-1)+HX
      IF (JXI .GE. JXM1) GO TO 19
      X(JXI+1)=X(JXI)+HX
      IF (JXI .EQ. JXM2) GO TO 19
      JUSE=JX-JXI+1 $ XUSE=XMAX-XMAXI
      CALL GETR(JUSE,XUSE,HX,RAM,KSINGUL)
      IF (KSINGUL .EQ. 1) RETURN
      DO 2 J=JXI+2,JXM1
    2 X(J)=X(J-1)+RAM*(X(J-1)-X(J-2))
   19 CONTINUE
      WRITE (3,71) ((J,X(J)),J=1,JX)
   71 FORMAT(//,6HX MESH,/,(I5,E16.6,I5,E16.6,I5,E16.6,I5,E16.6))
      WRITE (3,72) ((I,Y(I)),I=1,IYP1)
   72 FORMAT(//,6HY MESH,/,(I5,E16.6,I5,E16.6,I5,E16.6,I5,E16.6))
      DO 3 J=1,JXM1
    3 DXP5(J)=X(J+1)-X(J)
      DXP5(JX)=0. $ DXM5(1)=0.
      DO 4 J=1,JX
      DX(J)=.5*(DXP5(J)+DXM5(J))
      XSQ(J)=X(J)**2
      DXI(J)=1./DX(J)
      IF (J .EQ. 1) GO TO 4
      XI(J)=1./X(J) $ X2I(J)=XI(J)**2 $ X3I(J)=XI(J)*X2I(J
    4 CONTINUE
      DO 5 I=1,IY
    5 DYP5(I)=Y(I+1)-Y(I)
      DYP5(IYP1)=0. $ DYM5(1)=0.
      DO 6 I=1,IY
      DY(I)=.5*(DYM5(I)+DYP5(I))
      DYI(I)=1./DY(I)
      COSS(I)=COSF(Y(I)) $ SINN(I)=SINF(Y(I))
      IF (NSL .EQ. 0 .AND. I .EQ. 1) GO TO 6
      CTNY(I)=COSS(I)/SINN(I)
    6 CONTINUE
      COSS(IY)=0. $ CTNY(IY)=0.
      D2=DXP5(2) $ B2=DXM5(2)
      CINT(1)=(2.*B2**3-D2**3+3.*B2**2*D2)/(6.*B2*(D2+B2))
      DO 13 J=2,JXM1,2
      F0=DXP5(J) $ B0=DXM5(J)
   13 CINT(J)=(F0**3+B0**3+3.*F0*B0*(F0+B0))/(6.*F0*B0)
```

```
      DO 14 J=3,JXM2,2
      D1=DXP5(J-1) $ B1=DXM5(J-1) $ D2=DXP5(J+1) $ B2=DXM5(J+1)
      CINT(J)=(2.*D1**3-B1**3+3.*D1**2*B1)/(6.*D1*(D1+B1))+
     C (2.*B2**3-D2**3+3.*B2**2*D2)/(6.*B2*(D2+B2))
   14 CONTINUE
      D1=DXP5(JXM1) $ B1=DXM5(JXM1)
      CINT(JX)=(2.*D1**3-B1**3+3.*D1**2*B1)/(6.*D1*(D1+B1))
      D2=DYP5(2) $ B2=DYM5(2)
      CYNT(1)=(2.*B2**3-D2**3+3.*B2**2*D2)/(6.*B2*(D2+B2))
      DO 15 I=2,IY,2
      F0=DYP5(I) $ B0=DYM5(I)
   15 CYNT(I)=(F0**3+B0**3+3.*F0*B0*(F0+B0))/(6.*F0*B0)
      DO 16 I=3,IYM1,2
      D1=DYP5(I-1) $ B1=DYM5(I-1) $ D2=DYP5(I+1) $ B2=DYM5(I+1)
      CYNT(I)=(2.*D1**3-B1**3+3.*D1**2*B1)/(6.*D1*(D1+B1))+
     C (2.*B2**3-D2**3+3.*B2**2*D2)/(6.*B2*(D2+B2))
   16 CONTINUE
      DO 17 I=1,IYM1
   17 CYNT(I)=2.*CYNT(I)
      DO 18 I=1,IY $ DO 18 J=1,JX
   18 SPHERI(I,J)=TWOPI*XSQ(J)*SINN(I)*CINT(J)*CYNT(I)
      EXT(1,1)=(Y(1)-Y(3))/(Y(2)-Y(3))
      EXT(2,1)=(Y(1)-Y(2))/(Y(3)-Y(2))
      EXT(3,1)=0.
      EXT(1,2)=(Y(IY)-Y(IYM2))/(Y(IYM1)-Y(IYM2))
      EXT(2,2)=(Y(IY)-Y(IYM1))/(Y(IYM2)-Y(IYM1))
      EXT(3,2)=0.
      EXT(1,3)=(X(1)-X(3))/(X(2)-X(3))
      EXT(2,3)=(X(1)-X(2))/(X(3)-X(2))
      EXT(3,3)=0.
      EXT(1,4)=(X(JXI)-X(JXIM2))/(X(JXIM)-X(JXIM2))
      EXT(2,4)=(X(JXI)-X(JXIM))/(X(JXIM2)-X(JXIM))
      EXT(3,4)=0.
      HP=(BRATIO-1.)/FLOATF(LZ-1) $ BMAG(1)=1. $ DO 74 L=2,LZ
   74 BMAG(L)=BMAG(L-1)+HP
      DO 73 L=1,LZ
      PS1(L)=SQRTF(BMAG(L)) $ PS2(L)=SQRTF(BMAG(L)/BRATIO)
   73 CONTINUE
      RETURN
      END
      SUBROUTINE EEPRINT
      USE COMM
      WRITE (3,1) N,TIME,POTENT
    1 FORMAT(/,2HN=,I5,5X,5HTIME=,E16.6,5X,10HPOTENTIAL=,E16.6,/)
      DO 2 K=0,NEQ
      WRITE (3,6)
    6 FORMAT(1H1)
      WRITE (3,3) K,REDEN(K),ENERGY(K)
    3 FORMAT(2HK=,I5,5X,8HDENSITY=,E16.6,5X,7HENERGY=,E16.6,//)
      IF (K .EQ. 0) GO TO 7
      CALL READLCM(FN,F(1,1,K),IYP1JX)
      DO 4 J=1,JXI
      WRITE (3,5) J,X(J),(FN(I,J),I=1,IYP1)
    5 FORMAT(/,2HJ=,I5,2X,2HX=,E16.6,2X,2HF=,//,(6E16.6))
    4 CONTINUE
      GO TO 2
    7 CONTINUE
      WRITE (3,8) ((J,X(J),FE(J)),J=1,JX)
    8 FORMAT(/,(2HJ=,I5,2X,2HX=,E16.6,2X,3HFE=,E16.6))
    2 CONTINUE
```

```
      WRITE (3,6)
      RETURN
      END
      SUBROUTINE EEPLOT
      USE COMM
      DO 1 K=0,NEQ
      IF (K .EQ. 0) GO TO 21
      CALL READLCM(FN,F(1,1,K),IYP1JX)
      CALL SETCH(1.,20.,0,0,1,1)
      WRITE (100,5) K,TIME
    5 FORMAT(11HSPECIES NO ,I5,5X,5HTIME=,E16.6)
      PTANG=PLTANG
      DO 41 J=1,JX1
      TEMPG(J)=X(J) $ DO 41 I=1,IYP1
   41 FT(I,J)=FN(I,J)
      IF (PLTANG .LE. 180.) GO TO 42
      PTANG=PLTANG-180.
      DO 43 J=1,JX1
      JJ=JX1+1-J $ TEMPG(J)=X(JJ) $ DO 43 I=1,IYP1
   43 FT(I,J)=FN(I,JJ)
   42 CONTINUE
      CALL PLOT3D(PTANG,TEMPG,Y,FT,1.,1.,1.,JX1,IY,IYP1)
      GO TO 22
   21 CONTINUE
      CALL SETCH(1.,20.,0,0,1,1)
      WRITE (100,5) K,TIME
      CALL CARTMM(JX,DMIN,DMAX,FE,1)
      CALL MAPS(0.,XMAX,0.,DMAX,.2,.999,.2,.999)
      CALL TRACE(X,FE,JX)
      CALL FRAME
   22 CONTINUE
      IF (N .EQ. 0) GO TO 1
      CALL CARTMM(NCH,DMIN,DMAX,PDENS(1,K),1)
      CALL CARTMM(NCH,EMIN,EMAX,PENGY(1,K),1)
      IF (DMAX .LE. DMIN+1.E-90) DMAX=1.1*DMAX
      IF (EMAX .LE. EMIN+1.E-90) EMAX=1.1*EMAX
      CALL MAPS(PTIME(1),PTIME(NCH),DMIN,DMAX,.2,.999,.65,.999)
      CALL TRACE(PTIME,PDENS(1,K),NCH)
      CALL MAPS(PTIME(1),PTIME(NCH),EMIN,EMAX,.2,.999,.15,.5)
      CALL TRACE(PTIME,PENGY(1,K),NCH)
      CALL SETCH(17.,3.,1,0,1,0)
      WRITE (100,6) REDEN(K),ENERGY(K),TIME
    6 FORMAT(8HDENSITY=,E16.6,5X,7HENERGY=,E16.6,//,5HTIME=,E16.6)
      CALL SETCH(1.,17.,1,0,1,1)
      WRITE (100,7) K
    7 FORMAT(2HK=,I3,2X,6HENERGY,30X,7HDENSITY)
      CALL FRAME
    1 CONTINUE
      IF (N .EQ. 0) RETURN
      CALL CARTMM(NCH,DMIN,DMAX,PPOTENT,1)
      IF (DMAX .LE. DMIN+1.E-90) DMAX=1.1*DMAX
      CALL MAPS(PTIME(1),PTIME(NCH),DMIN,DMAX,.2,.999,.15,.5)
      CALL TRACE(PTIME,PPOTENT,NCH)
      CALL SETCH(17.,3.,1,0,1,0)
      WRITE (100,10) TIME
   10 FORMAT(20HAMBIPOLAR POTENTIAL ,//,5HTIME=,E16.6)
      CALL FRAME
      RETURN
      END
      SUBROUTINE BOUNDARY
```

```
      USE COMM
      DO 1 K=1,NEQ
      XC2=ANUMB(K)*POTENT/EIONS(K)
      XMIN=SQRTF(XC2/(BRATIO-1.))
      DO 2 J=1,JXIM
      IF (NSL .EQ. 0 .AND. J .EQ. 1) GO TO 3
      IF (XMIN .LE. X(J)+.5*DXP5(J)) GO TO 3
    2 CONTINUE
      WRITE (3,4) K,POTENT,XC2,XMIN
    4 FORMAT(21HBOUNDARY FAILURE - K=,I5,5X,7HPOTENT=,E16.6,5X,
     C 4HXC2=,E16.6,5X,5HXMIN=,E16.6)
      KBOUND=1 $ RETURN
    3 JMIN(K)=J
      DO 24 J=1,JMIN(K)
   24 IO(J,K)=IY
      JMP1=JMIN(K)+1
      DO 5 J=JMP1,JX1
      PFAC=(1.+XC2/XSQ(J))*BRATIOI
      TLC=ASINF(SQRTF(PFAC))
      IF (NSL .EQ. 0) TLC=0.
      I=1 $ IF (TLC .LE. Y(I)+.5*DYP5(I)) GO TO 6
      DO 7 I=2,IY
      IF (TLC .GT. Y(I)-.5*DYM5(I) .AND. TLC .LE. Y(I)+.5*DYP5(I))
     C GO TO 6
    7 CONTINUE
      WRITE (3,4) K,POTENT,XC2,XMIN
      WRITE (3,8) TLC
    8 FORMAT(4HTLC=,E16.6)
      KBOUND=1 $ RETURN
    6 IO(J,K)=I
    5 CONTINUE
      IF (NSL .EQ. 0) IO(1,K)=1
      IMIN(K)=IO(JX1,K)
      IF (IMIN(K) .LT. IY) GO TO 31
      WRITE (3,4) K,POTENT,XC2,XMIN
      WRITE (3,32) IMIN(K)
   32 FORMAT(5HIMIN=,I5)
      KBOUND=1 $ RETURN
   31 CONTINUE
      IMP1=IMIN(K)+1
      DO 9 I=IMIN(K),IY
      JO(I,K)=0
      DO 10 J=JMIN(K),JX1
      IF (IO(J,K) .EQ. I) JO(I,K)=J
   10 CONTINUE
    9 CONTINUE
      IF (NSL .EQ. 0) JO(1,K)=1
      IF (NSL .EQ. 0) JO(IY,K)=1
      DO 11 II=IMP1,IYM1
      I=IMP1+IYM1-II
      IF (JO(I,K) .EQ. 0) JO(I,K)=JO(I+1,K)
   11 CONTINUE
      DO 12 I=1,IMIN(K)-1
   12 JO(I,K)=JX1
    1 CONTINUE
      IF (N .GT. 1) RETURN
      DO 71 K=1,NEQ
      WRITE (3,72) K,IMIN(K),((J,IO(J,K)),J=1,JX1)
   72 FORMAT(//,2HK=,I5,5X,5HIMIN=,I5,//,(2HJ=,I5,5X,3HIO=,I5))
      WRITE (3,73) JMIN(K),((I,JO(I,K)),I=1,IY)
```

```
 73 FORMAT(/,5HJMIN=,I5,//,(2HI=,I5,5X,3HJO=,I5))
 71 CONTINUE
    RETURN
    END
    SUBROUTINE POTSHAPE
    USE COMM
    IF (NSL .EQ. 0) RETURN
    NBM1=NB-1
    BETN=PI/FLOATF(NPTS) $ BETN2=.5*BETN
    LZM1=LZ-1
    DO 97 IN=1,NPTS
    ARGN=FLOATF(2*IN-1)*BETN2
    COSYN(IN)=COSF(ARGN)
 97 CONTINUE
    DO 1 I=1,IY
  1 UMMU(I)=COSS(IYP1-I)
    PPOT(1)=0. $ PPOT(LZ)=POTENT
    PHM=POTENT/EIONS(0) $ CUT=SQRTF(PHM)
    S=0. $ DO 102 KK=1,NEQ
102 S=S+ANUMB(KK)*REDEN(KK)
    T=0. $ DO 321 J=1,JX
    FACT=1. $ IF (X(J) .LE. CUT) GO TO 321
    FACT=SQRTF(1.-BRATIO1*(1.-PHM*X21(J)))
321 T=T+FACT*FE(J)*XSQ(J)*CINT(J)
    CHRAT=S/(FOURPI*ANORMK(0)*T)
    PHI(1)=0. $ HP=POTENT/FLOATF(NBM1)
    DO 2 K=2,NB
  2 PHI(K)=PHI(K-1)+HP
    KLIM=1
    DO 73 L=2,LZM1
    DO 3 K=KLIM,NB
    DNSI(K)=0. $ DNSE(K)=0.
    PS=BMAG(L)
    DO 101 KK=1,NEQ
    CALL READLCM(FN,F(1,1,KK),IYP1JX)
    PH=PHI(K)*ANUMB(KX)/EIONS(KK)
    DO 4 J=1,JX1
    VINT(J)=0. $ IF (J .EQ. 1) GO TO 4
    S=0. $ DO 5 IN=1,NPTS
    SJN=COSYN(IN)
    RATSW=X(J)/SQRTF(XSQ(J)+PH)
    UULIM=RATSW*PS1(L) $ ULLIM=RATSW*PS2(L)
    IF (SJN .GT. UULIM .OR. SJN .LT. ULLIM) GO TO 5
    WJN=SJN/RATSW $ UMU2=1.-WJN**2/PS
    UMU=SQRTF(UMU2+1.E-13) $ DO 7 II=1,IYM1
    IF (UMU .LT. UMMU(II) .OR. UMU .GE. UMMU(II+1)) GO TO 7
    IUSE=II $ GO TO 8
  7 CONTINUE
    IUSE=IYM1
  8 W1=(UMU-UMMU(IUSE))/(UMMU(IUSE+1)-UMMU(IUSE))
    W0=1.-W1
    IVSE=IYP1-IUSE
    FIONS=W0*FN(IVSE,J)+W1*FN(IVSE-1,J)
    S=S+FIONS*SJN
  5 CONTINUE
    VINT(J)=BETN*S
  4 CONTINUE
    S=0. $ DO 10 J=1,JX1
    T=X(J)*SQRTF(XSQ(J)+PH)
 10 S=S+T*VINT(J)*CINT(J)
```

```
      DNSI(K)=DNSI(K)+S*ANUMB(KK)
  101 CONTINUE
      PH=PHI(K)/EIONS(0)
      DO 21 JJ=1,JXM1
      IF (CUT .LT. X(JJ) .OR. CUT .GE. X(JJ+1)) GO TO 21
      JP=JJ $ GO TO 22
   21 CONTINUE
      CUT=XMAX $ JP=JX
   22 JP1=JP+1
      CUTL=SQRTF(PH)
      DO 23 JJ=1,JXM1
      IF (CUTL .LT. X(JJ) .OR. CUTL .GE. X(JJ+1)) GO TO 23
      JQ=JJ $ GO TO 24
   23 CONTINUE
      CUTL=XMAX $ JQ=JX
   24 JQ1=JQ+1
      DO 27 J=1,JQ
   27 VENT(J)=0.
      DO 25 J=JQ1,JP
   25 VENT(J)=SQRTF(1.-PH*X21(J))
      DO 26 J=JP1,JX
   26 VENT(J)=SQRTF(1.-PS/BRATIO+(PS/BRATIO*PHM-PH)*X21(J))
      S=0. $ DO 13 J=1,JX
      T=XSQ(J)*FE(J)*VENT(J)
   13 S=S+T*CINT(J)
      DNSE(K)=S*CHRAT
    3 CONTINUE
      IF (TIME .GT. 0.) GO TO 173
      WRITE (3,174) L,((K,DNSI(K),DNSE(K)),K=KLIM,NB)
  174 FORMAT(/,2HL=,I5,/,(2HK=,I5,2X,5HDNSI=,E16.6,2X,5HDNSE=,E16.6))
  173 CONTINUE
      DO 31 K=KLIM,NB
   31 DDIFF(K)=DNSI(K)-DNSE(K)
      DO 33 K=KLIM,NBM1
      PROD=DDIFF(K)*DDIFF(K+1)
      IF (PROD .LE. 0.) GO TO 34
   33 CONTINUE
      S=ABSF(DDIFF(KLIM)) $ KHU=KLIM
      DO 133 K=KLIM+1,NB
      T=ABSF(DDIFF(K)) $ IF (S .LE. T) GO TO 133
      S=T $ KHU=K
  133 CONTINUE
      PPOT(L)=PHI(KHU)
      KLIM=KHU
      GO TO 73
   34 CONTINUE
      DENOM=DDIFF(K)-DDIFF(K+1)
      ENUM=PHI(K+1)-PHI(K)
      IF (ABSF(DENOM) .LT. 1.E-90) DENOM=1.E+90
      PPOT(L)=PHI(K)+DDIFF(K)*ENUM/DENOM
      KLIM=K
   73 CONTINUE
      IF (TIME .GT. 0.) RETURN
      WRITE (3,175) ((L,BMAG(L),PPOT(L)),L=1,LZ)
  175 FORMAT(2HL=,I5,2X,4HPSI=,E16.6,2X,5HPPOT=,E16.6)
      RETURN
      END
      SUBROUTINE DENSINPT
      USE COMM
      IF (NSL .EQ. 0) RETURN
```

```
      BETN=PI/FLOATF(NPTS) $ BETN2=.5*BETN
      LZMI=LZ-1
      PHM=POTENT/EIONS(0) $ CUT=SQRTF(PHM)
      DO 73 L=1,LZMI
      DNSIE(L)=0. $ PPERP(L)=0. $ PPAR(L)=0.
      PS=BMAG(L)
      DO 101 KK=1,NEQ
      CALL READLCM(FN,F(1,1,KK),IYP1JX)
      PH=PPOT(L)*ANUMB(KK)/EIONS(KK)
      DO 4 J=1,JXI
      VINT(J)=0. $ VINTPE(J)=0. $ VINTPA(J)=0.
      IF (J .EQ. 1) GO TO 4
      S=0. $ SE=0. $ SA=0.
      DO 5 IN=1,NPTS
      SJN=COSYN(IN) $ RATSW=X(J)/SQRTF(XSQ(J)+PH)
      UULIM=RATSW*PS1(L) $ ULLIM=RATSW*PS2(L)
      IF (SJN .GT. UULIM .OR. SJN .LT. ULLIM) GO TO 5
      WJN=SJN/RATSW $ UMU2=1.-WJN**2/PS
      UMU=SQRTF(UMU2+1.E-13) $ DO 7 II=1,IYMI
      IF (UMU .LT. UMMU(II) .OR. UMU .GE. UMMU(II+1)) GO TO 7
      IUSE=II $ GO TO 8
    7 CONTINUE
      IUSE=IYMI
    8 W1=(UMU-UMMU(IUSE))/(UMMU(IUSE+1)-UMMU(IUSE))
      W0=1.-W1
      IVSE=IYP1-IUSE
      FIONS=W0*FN(IVSE,J)+W1*FN(IVSE-1,J)
      T=FIONS*SJN
      TE=T*EIONS(KK)*ERGTKEV*PS*XSQ(J)*(1.-UMU**2)
      TA=2.*T*EIONS(KK)*ERGTKEV*(XSQ(J)*(1.-PS*(1.-UMU**2))+PH)
      S=S+T $ SE=SE+TE $ SA=SA+TA
    5 CONTINUE
      VINT(J)=BETN*S $ VINTPE(J)=BETN*SE $ VINTPA(J)=BETN*SA
    4 CONTINUE
      S=0. $ SE=0. $ SA=0.
      DO 10 J=1,JXI
      T0=X(J)*SQRTF(XSQ(J)+PH)
      T=T0*VINT(J) $ TE=T0*VINTPE(J) $ TA=T0*VINTPA(J)
      S=S+T*CINT(J) $ SE=SE+TE*CINT(J) $ SA=SA+TA*CINT(J)
   10 CONTINUE
      DNSIE(L)=DNSIE(L)+ANUMB(KK)*S*FOURPI*ANORMK(KK)
      PPERP(L)=PPERP(L)+SE*FOURPI*ANORMK(KK)
      PPAR(L)=PPAR(L)+SA*FOURPI*ANORMK(KK)
  101 CONTINUE
      PH=PPOT(L)/EIONS(0)
      DO 21 JJ=1,JXMI
      IF (CUT .LT. X(JJ) .OR. CUT .GE. X(JJ+1)) GO TO 21
      JP=JJ $ GO TO 22
   21 CONTINUE
      CUT=XMAX $ JP=JX
   22 JP1=JP+1
      CUTL=SQRTF(PH) $ DO 23 JJ=1,JXMI
      IF (CUTL .LT. X(JJ) .OR. CUTL .GE. X(JJ+1)) GO TO 23
      JQ=JJ $ GO TO 24
   23 CONTINUE
      CUTL=XMAX $ JQ=JX
   24 JQ1=JQ+1
      DO 27 J=1,JQ
   27 VENT(J)=0.
      DO 25 J=JQ1,JP
```

```
      VENT(J)=SQRTF(1.-PH*X2I(J))
      FACI=EMASS(0)*VNORM**2/3.*(XSQ(J)-PH)
      VENT(J)=FACI*VENT(J)
   25 CONTINUE
      DO 26 J=JP1,JX
   26 VENT(J)=0.
      S=0. $ DO 13 J=1,JX
      T=XSQ(J)*FE(J)*VENT(J)
   13 S=S+T*CINT(J)
      PPERP(L)=PPERP(L)+S*FOURPI*ANORMK(0)
      PPAR(L)=PPAR(L)+S*FOURPI*ANORMK(0)
   73 CONTINUE
      DNSIE(LZ)=0. $ PPERP(LZ)=0. $ PPAR(LZ)=0.
      WRITE (3,41)((L,BMAG(L),PPOT(L),DNSIE(L),PPERP(L),PPAR(L)),L=1,LZ)
   41 FORMAT(/,(2HL=,I5,2X,4HPSI=,E16.6,2X,4HPOT=,E16.6,2X,5HDENS=,
     C E16.6,/,10X,6HPPERP=,E16.6,2X,5HPPAR=,E16.6))
      CALL FRAME
      CALL CARTMM(LZ,DMIN,DMAX,PPOT,1)
      CALL CARTMM(LZ,EMIN,EMAX,DNSIE,1)
      CALL MAPS(1.,BRATIO,DMIN,DMAX,.2,.999,.15,.5)
      CALL TRACE(BMAG,PPOT,LZ)
      CALL MAPS(1.,BRATIO,EMIN,EMAX,.2,.999,.65,.999)
      CALL TRACE(BMAG,DNSIE,LZ)
      CALL SETCH(17.,1.,1,0,1,0)
      WRITE (100,81) TIME
   81 FORMAT(5HTIME=,E16.6,9H  SECONDS)
      CALL SETCH(1.,17.,1,0,1,1)
      WRITE (100,82)
   82 FORMAT(14HN OF PSI - TOP)
      CALL SETCH(2.,17.,1,0,1,1)
      WRITE (100,83)
   83 FORMAT(19HPHI OF PSI - BOTTOM)
      CALL FRAME
      CALL CARTMM(LZ,DMIN,DMAX,PPERP,1)
      CALL CARTMM(LZ,EMIN,EMAX,PPAR,1)
      CALL MAPS(1.,BRATIO,DMIN,DMAX,.2,.999,.15,.5)
      CALL TRACE(BMAG,PPERP,LZ)
      CALL MAPS(1.,BRATIO,EMIN,EMAX,.2,.999,.65,.999)
      CALL TRACE(BMAG,PPAR,LZ)
      CALL SETCH(17.,1.,1,0,1,0)
      WRITE (100,81) TIME
      CALL SETCH(1.,17.,1,0,1,1)
      WRITE (100,84)
   84 FORMAT(17HPPAR OF PSI - TOP)
      CALL SETCH(2.,17.,1,0,1,1)
      WRITE (100,85)
   85 FORMAT(21HPPERP OF PSI - BOTTOM)
      CALL FRAME
      EKM=1.E+90 $ DO 201 L=1,LZM1
      DENOM=PPERP(L)-PPERP(L+1)
      IF (DENOM .LE. 0.) GO TO 201
      EK=(BMAG(L+1)**2-BMAG(L)**2)/DENOM
      IF (EKM .GT. EK) EKM=EK
  201 CONTINUE
      ELM=1.E+90 $ DO 209 L=1,LZM1
      DENOM=PPAR(L+1)-PPAR(L) $ IF (DENOM .LE. 0.) GO TO 209
      EK=(BMAG(L+1)**2-BMAG(L)**2)/DENOM
      IF (ELM .GT. EK) ELM=EK
  209 CONTINUE
      BETMAX=EKM*PPERP(1)/(1.+EKM*PPERP(1))
```

```
      BETO=BETAO $ IF (BETO .GT. BETMAX) BETO=BETMAX
      DO 202 L=1,LZ
  202 PSVAC(L)=SQRTF(BMAG(L)**2+BETO*PPERP(L)/(PPERP(1)*(1.-BETO)))
      RVAC=BRATIO*SQRTF(1.-BETO)
      EKTP=PPERP(1)/DNSIE(1) $ EKTPK=EKTP/ERGTKEV
      CALL CARTMM(LZ,DMIN,DMAX,PSVAC,1)
      CALL MAPS(1.,BRATIO,DMIN,DMAX,.2,.999,.5,.999)
      CALL TRACE(BMAG,PSVAC,LZ)
      CALL SETCH(1.,17.,1,0,1,1)
      WRITE (100,203)
  203 FORMAT(13HPSIVAC OF PSI)
      CALL SETCH(17.,7.,1,0,1,0)
      WRITE (100,204) EKM,ELM,RVAC,BETMAX,BETAO,EKTP,EKTPK
  204 FORMAT(4HEKM=,E16.6,2X,4HELM=,E16.6,/,5HRVAC=,E16.6,/,7HBETMAX=,
     C E16.6,2X,6HBETAO=,E16.6,/,7HKTPERP=,E16.6,9H  ERGS = ,E16.6,
     C 5H  KEV)
      CALL SETCH(17.,1.,1,0,1,0)
      WRITE (100,81) TIME
      CALL FRAME
      WRITE (3,205) EKM,ELM,RVAC,BETMAX,BETAO,EKTP,EKTPK,
     C ((L,BMAG(L),PSVAC(L)),L=1,LZ)
  205 FORMAT(/,4HEKM=,E16.6,2X,4HELM=,E16.6,2X,5HRVAC=,E16.6,2X,
     C 7HBETMAX=,E16.6,/,6HBETAO=,E16.6,5X,7HKTPERP=,E16.6,9H  ERGS = ,
     C E16.6,5H  KEV,//,7HPSIVAC=,//,(I5,2E16.6))
      RETURN
      END
      SUBROUTINE XSWEEP
      USE COMM
      DO 100 K=0,NEQ
      DT=DTR/TNORM(K)
      RBGN=1./DT
      IF (K .EQ. 0) GO TO 101
      CALL READLCM(CA,CAL(1,1,K),IYP1JX)
      CALL READLCM(CB,CBL(1,1,K),IYP1JX)
      CALL READLCM(CC,CCL(1,1,K),IYP1JX)
      CALL READLCM(CD,CDL(1,1,K),IYP1JX)
      CALL READLCM(CE,CEL(1,1,K),IYP1JX)
      CALL READLCM(CF,CFL(1,1,K),IYP1JX)
      CALL READLCM(FT,F(1,1,K),IYP1JX)
      CALL READLCM(FO,F1(1,1,K),IYP1JX)
      CALL READLCM(SO,SOURCE(1,1,K),IYP1JX)
      DO 1 I=,,IYP1 $ DO 1 J=1,JX1
      FO(I,J)=FT(I,J)
    1 FN(I,J)=0.
      IMP1=IMIN(K)+1
      DO 2 I=IMP1,IY
      JS=JO(I,K) $ JSP=JS+1
      EP(JS)=0. $ FF(JS)=0.
      IF (NSL .EQ. 1) GO TO 71
      SUMC=CB(I,1)+CB(I,2) $ DENOM=SUMC-DXM5(2)*CA(I,1)
      EP(1)=(SUMC+CA(1,2)*DXM5(2))/DENOM
   71 CONTINUE
      DO 3 J=JSP,JX1M
      TXSD=2.*XSQ(J)*DX(J)
      ALP=(CA(I,J+1)+(CB(I,J)+CB(I,J+1))/DXP5(J))/TXSD
      GAM=(-CA(I,J-1)+(CB(I,J-1)+CB(I,J))/DXM5(J))/TXSD
      BET=RBGN+((CB(I,J+1)+CB(I,J))/DXP5(J)+(CB(I,J)+CB(I,J-1))/
     C DXM5(J))/TXSD
      TXSDT=2.*TXSD*DY(I)
      DEL=RBGN*FT(I,J)+(CC(I,J+1)*(FO(I+1,J+1)-FO(I-1,J+1))-
```

```
   C CC(I,J-1)*(FO(I+1,J-1)-FO(I-1,J-1)))/TXSDT
     DEL=DEL+.5*TNORM(K)*SO(I,J)
     DEN=BET-GAM*EP(J-1)
     EP(J)=ALP/DEN $ FP(J)=(DEL+GAM*FP(J-1))/DEN
   3 CONTINUE
     FN(I,JXI)=0.
     DO 4 JJ=JS,JXIM
     J=JS+JXIM-JJ
   4 FN(I,J)=EP(J)*FN(I,J+1)+FP(J)
   2 CONTINUE
     DO 31 J=1,JXI
     IF (NSL .EQ. 0) FN(1,J)=FN(2,J)
  31 FN(IYP1,J)=FN(IYM1,J)
     DO 5 I=1,IYP1 $ DO 5 J=1,JXI
     IF (FN(I,J) .LT. 0.) FN(I,J)=0.
     FO(I,J)=FN(I,J)
     FT(I,J)=FN(I,J) $ FN(I,J)=0.
   5 CONTINUE
     JMP1=JMIN(K)+1
     DO 6 JJ=JMP1,JXIM
     J=JMP1+JXIM-JJ
     IS=IO(J,K) $ ISP=IS+1
     EP(IS)=0. $ FP(IS)=0.
     IF (NSL .EQ. 1) GO TO 72
     SUMC=CF(1,J)+CF(2,J) $ DENOM=SUMC-DYM5(2)*CD(1,J)
     EP(1)=(SUMC+CD(2,J)*DYM5(2))/DENOM
  72 CONTINUE
     DO 7 I=ISP,IY
     TXSD=2.*XSQ(J)*SINN(I)*DY(I)
     ALP=(CD(I+1,J)+(CF(I,J)+CF(I+1,J))/DYP5(I))/TXSD
     GAM=(-CD(I-1,J)+(CF(I-1,J)+CF(I,J))/DYM5(I))/TXSD
     BET=RBGN+((CF(I+1,J)+CF(I,J))/DYP5(I)+(CF(I-1,J)+CF(I,J))/
    C DYM5(I))/TXSD
     TXSDT=2.*DX(J)*TXSD
     DEL=RBGN*FT(I,J)+(CE(I+1,J)*(FO(I+1,J+1)-FO(I+1,J-1))-
    C CE(I-1,J)*(FO(I-1,J+1)-FO(I-1,J-1)))/TXSDT
     DEL=DEL+.5*TNORM(K)*SO(I,J)
     DEN=BET-GAM*EP(I-1)
     EP(I)=ALP/DEN $ FP(I)=(DEL+GAM*FP(I-1))/DEN
   7 CONTINUE
     FN(IYP1,J)=(EP(IYM1)*FP(IY)+FP(IYM1))/(1.-EP(IYM1)*EP(IY))
     DO 8 II=IS,IY
     I=IS+IY-II
     FN(I,J)=EP(I)*FN(I+1,J)+FP(I)
   8 CONTINUE
   6 CONTINUE
     IF (NSL .EQ. 1) GO TO 73
     DO 74 I=1,IYP1
  74 FN(I,1)=FN(IY,2)
  73 CONTINUE
     CALL WRITELCM(FN,F(1,1,K),IYP1JX)
     GO TO 100
 101 CONTINUE
     XCN2=POTENT/EIONS(0) $ XCN=SQRTF(XCN2)
     DO 82 J=1,JX
     CCEL(J)=0. $ IF (X(J) .LE. XCN) GO TO 82
     IF (NSL .EQ. 0) GO TO 82
     REFF=BRATIO/(1.-XCN2*X2I(J))
     PROB=2.3025851/LOGF(REFF)
     CCEL(J)=CELL(J)*PROB
```

```
 82 CONTINUE
    SUMC=BELL(1)+BELL(2) $ DENOM=SUMC-DXM5(2)*AELL(1)
    EP(1)=(SUMC+AELL(2)*DXM5(2))/DENOM $ FP(1)=0.
    DO 13 J=2,JXM1
    TXSD=2.*XSQ(J)*DX(J)
    ALP=(AELL(J+1)+(BELL(J)+BELL(J+1))/DXP5(J))/TXSD
    GAM=(-AELL(J-1)+(BELL(J-1)+BELL(J))/DXM5(J))/TXSD
    BET=RBGN+((BELL(J+1)+BELL(J))/DXP5(J)+(BELL(J)+BELL(J-1))/DXM5(J))
   C /TXSD+CCEL(J)*X2I(J)
    DEL=RBGN*FE(J)+TNORM(K)*SORF(J)
    DEN=BET-GAM*EP(J-1)
    EP(J)=ALP/DEN $ FP(J)=(DEL+GAM*FP(J-1))/DEN
 13 CONTINUE
    FE(JX)=0. $ DO 14 JJ=1,JXM1
    J=JX-JJ
 14 FE(J)=EP(J)*FE(J+1)+FP(J)
100 CONTINUE
    RETURN
    END
    SUBROUTINE COEF
    USE COMM
    DIMENSION UM(IY)
    EQUIVALENCE(UM,COSS)
    IF (KGETGFK .EQ. 1) GO TO 10
    DO 20 M=MXP3M,MXP4
    DO 1 J=2,JX
  1 XM(J,M)=X(J)**M
 20 XM(1,M)=0.
    DO 4 I=1,IY
    PLEG(0,I)=1. $ DPLEG(0,I)=0.
    DDPLEG(0,I)=0. $ DDDPLEG(0,I)=0.
    IF (MX .EQ. 0) GO TO 4
    PLEG(1,I)=UM(I) $ DPLEG(1,I)=1.
    DDPLEG(1,I)=0. $ DDDPLEG(1,I)=0.
    DO 5 M=2,MMX
    PLEG(M,I)=((2*M-1)*UM(I)*PLEG(M-1,I)-(M-1)*PLEG(M-2,I))/M
    DPLEG(M,I)=(2*M-1)*PLEG(M-1,I)+DPLEG(M-2,I)
    DDPLEG(M,I)=(2*M-1)*DPLEG(M-1,I)+DDPLEG(M-2,I)
    DDDPLEG(M,I)=(2*M-1)*DDPLEG(M-1,I)+DDDPLEG(M-2,I)
  5 CONTINUE
  4 CONTINUE
    DO 7 MM=0,MX
    M=2*MM
    FAC=FOURPI/(2*M+1)
    COG(MM,1)=FAC/(2*M+3)
    COG(MM,2)=FAC/(2*M-1)
    COG(MM,3)=(M+2)*COG(MM,1)
    COG(MM,4)=(M+1)*COG(MM,1)
    COG(MM,5)=M*COG(MM,2)
    COG(MM,6)=(M-1)*COG(MM,2)
    COG(MM,7)=(M+1)*COG(MM,3)
    COG(MM,8)=(M-1)*COG(MM,5)
    COG(MM,9)=M*COG(MM,7)
    COG(MM,10)=(M+3)*COG(MM,7)
    COG(MM,11)=(M-2)*COG(MM,8)
    COG(MM,12)=(M+1)*COG(MM,8)
    COG(MM,13)=FAC*M
    COG(MM,14)=FAC*(M+1)
    COG(MM,15)=FAC
  7 CONTINUE
```

```
      KGETGFK=1
   10 CONTINUE
      DO 91 I=1,IYP1 $ DO 91 J=1,JX1
   91 SO(I,J)=0.
      DO 92 K=1,NEQ
      CALL WRITELCM(SO,CAL(1,1,K),IYP1JX)
      CALL WRITELCM(SO,CBL(1,1,K),IYP1JX)
      CALL WRITELCM(SO,CCL(1,1,K),IYP1JX)
      CALL WRITELCM(SO,CDL(1,1,K),IYP1JX)
      CALL WRITELCM(SO,CEL(1,1,K),IYP1JX)
      CALL WRITELCM(SO,CFL(1,1,K),IYP1JX)
   92 CONTINUE
      DO 61 J=1,JX
      AELL(J)=0. $ BELL(J)=0. $ CELL(J)=0.
      AEL(J)=0. $ BEL(J)=0. $ CCEL(J)=0.
   61 CONTINUE
      DO 93 K=0,NEQ
      IF (K .NE. 0) CALL READLCM(FT,F(1,1,K),IYP1JX)
      DO 14 MM=0,MX
      M=2*MM
      IF (K .EQ. 0) GO TO 401
      DO 12 J=1,JX1
      S=0. $ DO 11 I=1,IY
   11 S=S+CYNT(I)*FT(I,J)*PLEG(M,I)*SINN(I)
   12 TEMPG(J)=.5*FLOATF(2*M+1)*S
      DO 400 J=JX1+1,JX
  400 TEMPG(J)=0.
      GO TO 402
  401 DO 403 J=1,JX
  403 TEMPG(J)=0.
      IF (M .GT. 0) GO TO 402
      DO 404 J=1,JX
  404 TEMPG(J)=FE(J)
  402 CONTINUE
      IX1=1-M $ IX2=2+M $ IX3=3-M $ IX4=4+M
      IX5=3-IX1 $ IX6=3-IX2 $ IX7=3-IX3 $ IX8=3-IX4
      SM=0. $ SN=0. $ SR=0. $ SE=0.
      EMQF(JX,MM)=SM $ ENQF(1,MM)=SN $ ERQF(JX,MM)=SR $ EEQF(1,MM)=SE
      DO 13 J=2,JX
      JJ=JX+1-J
      TM=.5*DXP5(JJ)*(XM(JJ,IX1)*TEMPG(JJ)+XM(JJ+1,IX1)*TEMPG(JJ+1))
      TN=.5*DXM5(J)*(XM(J-1,IX2)*TEMPG(J-1)+XM(J,IX2)*TEMPG(J))
      TR=.5*DXP5(JJ)*(XM(JJ,IX3)*TEMPG(JJ)+XM(JJ+1,IX3)*TEMPG(JJ+1))
      TE=.5*DXM5(J)*(XM(J-1,IX4)*TEMPG(J-1)+XM(J,IX4)*TEMPG(J))
      SM=SM+TM $ SN=SN+TN $ SR=SR+TR $ SE=SE+TE
      EMQF(JJ,MM)=SM $ ENQF(J,MM)=SN $ ERQF(JJ,MM)=SR $ EEQF(J,MM)=SE
   13 CONTINUE
      DO 15 J=1,JX
      EMQF(J,MM)=EMQF(J,MM)*XM(J,IX5)
      ENQF(J,MM)=ENQF(J,MM)*XM(J,IX6)
      ERQF(J,MM)=ERQF(J,MM)*XM(J,IX7)
      EEQF(J,MM)=EEQF(J,MM)*XM(J,IX8)
   15 CONTINUE
   14 CONTINUE
      IF (K .EQ. 0) GO TO 411
      DO 16 J=2,JX1M $ DO 16 I=2,IY
      S=0. $ SX=0. $ SY=0. $ SXX=0. $ SXY=0. $ SYY=0.
      SXXX=0. $ SXXY=0. $ SXYY=0. $ SYYY=0. $ SH=0. $ SHX=0. $ SHY=0.
      DO 17 M=0,MX
      MM=2*M
```

```
      T=COG(M,1)*(EEQF(J,M)+EMQF(J,M))-COG(M,2)*(ENQF(J,M)+ERQF(J,M))
      TX=XI(J)*(COG(M,3)*EMQF(J,M)-COG(M,4)*EEQF(J,M)-
     C COG(M,5)*ERQF(J,M)+COG(M.5)*ENQF(J,M))
      TXX=X2I(J)*(COG(M,7)*(EEQF(J,M)+EMQF(J,M))-
     C COG(M,8)*(ENQF(J,M)+ERQF(J,M)))
      TXXX=X3I(J)*(COG(M,9)*EMQF(J,M)-COG(M,10)*EEQF(J,M)-
     C COG(M,11)*ERQF(J,M)+COG(M,12)*ENQF(J,M))
      TH=X2I(J)*COG(M,15)*(ENQF(J,M)+EMQF(J,M))
      THX=X3I(J)*(COG(M,13)*EMQF(J,M)-COG(M,14)*ENQF(J,M))
      S=S+T*PLEG(MM,I) $ SX=SX+TX*PLEG(MM,I)
      SXX=SXX+TXX*PLEG(MM,I) $ SY=SY+T*DPLEG(MM,I)
      SXY=SXY+TX*DPLEG(MM,I) $ SYY=SYY+T*DDPLEG(MM,I)
      SXXX=SXXX+TXXX*PLEG(MM,I) $ SXXY=SXXY+TXX*DPLEG(MM,I)
      SXYY=SXYY+TX*DDPLEG(MM,I) $ SYYY=SYYY+T*DDDPLEG(MM,I)
      SH=SH+TH*PLEG(MM,I) $ SHX=SHX+THX*PLEG(MM,I)
      SHY=SHY+TH*DPLEG(MM,I)
   17 CONTINUE
      GX=SX $ GY=-SINN(I)*SY $ GXX=SXX $ GXY=-SINN(I)*SXY
      GYY=-UM(I)*SY+SINN(I)**2*SYY
      GXXX=SXXX $ GXXY=-SINN(I)*SXXY
      GXYY=-UM(I)*SXY+SINN(I)**2*SXYY
      GYYY=SINN(I)*(SY+3.*UM(I)*SYY-SINN(I)**2*SYYY)
      HX=SHX $ HY=-SINN(I)*SHY
      CA(I,J)=.5*GXXX*XSQ(J)+X(J)*GXX-GX+.5*GXYY-XI(J)*(GYY+CTNY(I)*GY)
     C +.5*CTNY(I)*GXY
      FN(I,J)=-HX*XSQ(J)
      CB(I,J)=.5*XSQ(J)*GXX $ CC(I,J)=.5*(GXY-XI(J)*GY)
      CD(I,J)=.5*X2I(J)*(SINN(I)*GYYY-GY/SINN(I)+COSS(I)*GYY)+
     C SINN(I)*(.5*GXXY+XI(J)*GXY)
      FO(I,J)=-SINN(I)*HY $ CE(I,J)=SINN(I)*CC(I,J)
      CF(I,J)=.5*XI(J)*SINN(I)*(XI(J)*GYY+GX)
   16 CONTINUE
      M=0
      DO 63 J=2,JX
      TX=XI(J)*(COG(M,3)*EMQF(J,M)-COG(M,4)*CEQF(J,M)-
     C COG(M,5)*ERQF(J,M)+COG(M,6)*ENQF(J,M))
      TXX=X2I(J)*(COG(M,7)*(EEQF(J,M)+EMQF(J,M))-
     C COG(M,8)*(ENQF(J,M)+ERQF(J,M)))
      TXXX=X3I(J)*(COG(M,9)*EMQF(J,M)-COG(M,10)*EEQF(J,M)-
     C COG(M,11)*ERQF(J,M)+COG(M,12)*ENQF(J,M))
      THX=X3I(J)*(COG(M,13)*EMQF(J,M)-COG(M,14)*ENQF(J,M))
      AEL(J)=.5*TXXX*XSQ(J)+X(J)*TXX-TX-THX*XSQ(J)*(1.+RATIOM(0,K))
      BEL(J)=.5*XSQ(J)*TXX $ CCEL(J)=.5*XI(J)*TX
   63 CONTINUE
      GO TO 412
  411 CONTINUE
      M=0
      DO 413 J=2,JX
      TX=XI(J)*(COG(M,3)*EMQF(J,M)-COG(M,4)*EEQF(J,M)-
     C COG(M,5)*ERQF(J,M)+COG(M,6)*ENQF(J,M))
      TXX=X2I(J)*(COG(M,7)*(EEQF(J,M)+EMQF(J,M))-
     C COG(M,8)*(ENQF(J,M)+ERQF(J,M)))
      TXXX=X3I(J)*(COG(M,9)*EMQF(J,M)-COG(M,10)*EEQF(J,M)-
     C COG(M,11)*ERQF(J,M)+COG(M,12)*ENQF(J,M))
      THX=X3I(J)*(COG(M,13)*EMQF(J,M)-COG(M,14)*ENQF(J,M))
      IF (J .GT. JXIM) GO TO 415
      CA(IY,J)=.5*TXXX*XSQ(J)+X(J)*TXX-TX
      FN(IY,J)=-THX*XSQ(J) $ CB(IY,J)=.5*XSQ(J)*TXX
      CC(IY,J)=0. $ CD(IY,J)=0. $ CE(IY,J)=0.
      FO(IY,J)=0.
```

```
      CF(IY,J)=.5*XI(J)*TX
      DO 414 I=2,IYMI
      CA(I,J)=CA(IY,J) $ FN(I,J)=FN(IY,J) $ CB(I,J)=CB(IY,J)
      CC(I,J)=CC(IY,J) $ CD(I,J)=CD(IY,J) $ CE(I,J)=CE(IY,J)
      FO(I,J)=FO(IY,J) $ CF(I,J)=SINN(I)*CF(IY,J)
  414 CONTINUE
  415 CONTINUE
      AEL(J)=.5*TXXX*XSQ(J)+X(J)*TXX-TX-THX*XSQ(J)*(1.+RATIOM(0,K))
      BEL(J)=.5*XSQ(J)*TXX $ CCEL(J)=.5*XI(J)*TX
  413 CONTINUE
  412 CONTINUE
      DO 21 KK=1,NEQ
      ANRI=GAMMA(KK,K)*RATIOZ2(K,KK)*RATK(K,KK)
      ANR2=ANRI*(1.+RATIOM(KK,K))
      CALL READLCM(SO,CAL(1,1,KK),IYPIJX)
      DO 22 I=2,IY $ DO 22 J=2,JXIM
   22 SO(I,J)=SO(I,J)+ANRI*CA(I,J)+ANR2*FN(I,J)
      CALL WRITELCM(SO,CAL(1,1,KK),IYPIJX)
      CALL READLCM(SO,CBL(1,1,KK),IYPIJX)
      DO 23 I=2,IY $ DO 23 J=2,JXIM
   23 SO(I,J)=SO(I,J)+ANRI*CB(I,J)
      CALL WRITELCM(SO,CBL(1,1,KK),IYPIJX)
      CALL READLCM(SO,CCL(1,1,KK),IYPIJX)
      DO 24 I=2,IY $ DO 24 J=2,JXIM
   24 SO(I,J)=SO(I,J)+ANRI*CC(I,J)
      CALL WRITELCM(SO,CCL(1,1,KK),IYPIJX)
      CALL READLCM(SO,CDL(1,1,KK),IYPIJX)
      DO 25 I=2,IY $ DO 25 J=2,JXIM
   25 SO(I,J)=SO(I,J)+ANRI*CD(I,J)+ANR2*FO(I,J)
      CALL WRITELCM(SO,CDL(1,1,KK),IYPIJX)
      CALL READLCM(SO,CEL(1,1,KK),IYPIJX)
      DO 26 I=2,IY $ DO 26 J=2,JXIM
   26 SO(I,J)=SO(I,J)+ANRI*CE(I,J)
      CALL WRITELCM(SO,CEL(1,1,KK),IYPIJX)
      CALL READLCM(SO,CFL(1,1,KK),IYPIJX)
      DO 27 I=2,IY $ DO 27 J=2,JXIM
   27 SO(I,J)=SO(I,J)+ANRI*CF(I,J)
      CALL WRITELCM(SO,CFL(1,1,KK),IYPIJX)
   21 CONTINUE
      KK=0 $ ANRI=GAMMA(KK,K)*RATIOZ2(K,KK)*RATK(K,KK)
      DO 421 J=2,JX
      AELL(J)=AELL(J)+AEL(J)*ANRI
      BELL(J)=BELL(J)+BEL(J)*ANRI
      CELL(J)=CELL(J)+CCEL(J)*ANRI
  421 CONTINUE
   93 CONTINUE
      DO 111 K=1,NEQ
      CALL READLCM(CA,CAL(1,1,K),IYPIJX)
      CALL READLCM(CB,CBL(1,1,K),IYPIJX)
      CALL READLCM(CC,CCL(1,1,K),IYPIJX)
      CALL READLCM(CD,CDL(1,1,K),IYPIJX)
      CALL READLCM(CE,CEL(1,1,K),IYPIJX)
      CALL READLCM(CF,CFL(1,1,K),IYPIJX)
      DO 113 I=2,IY
      CA(I,1)=EXT(1,3)*CA(I,2)+EXT(2,3)*CA(I,3)+EXT(3,3)*CA(I,4)
      CB(I,1)=EXT(1,3)*CB(I,2)+EXT(2,3)*CB(I,3)+EXT(3,3)*CB(I,4)
      CC(I,1)=EXT(1,3)*CC(I,2)+EXT(2,3)*CC(I,3)+EXT(3,3)*CC(I,4)
      CD(I,1)=EXT(1,3)*CD(I,2)+EXT(2,3)*CD(I,3)+EXT(3,3)*CD(I,4)
      CE(I,1)=EXT(1,3)*CE(I,2)+EXT(2,3)*CE(I,3)+EXT(3,3)*CE(I,4)
      CF(I,1)=EXT(1,3)*CF(I,2)+EXT(2,3)*CF(I,3)+EXT(3,3)*CF(I,4)
```

```
      CA(I,JXI)=EXT(1,4)*CA(I,JXIM)+EXT(2,4) CA(I,JXIM2)+
     C EXT(3,4)*CA(I,JXIM3)
      CB(I,JXI)=EXT(1,4)*CB(I,JXIM)+EXT(2,4)*CB(I,JXIM2)+
     C EXT(3,4)*CB(I,JXIM3)
      CC(I,JXI)=EXT(1,4)*CC(I,JXIM)+EXT(2,4)*CC(I,JXIM2)+
     C EXT(3,4)*CC(I,JXIM3)
      CD(I,JXI)=EXT(1,4)*CD(I,JXIM)+EXT(2,4)*CD(I,JXIM2)+
     C EXT(3,4)*CD(I,JXIM3)
      CE(I,JXI)=EXT(1,4)*CE(I,JXIM)+EXT(2,4)*CE(I,JXIM2)+
     C EXT(3,4)*CE(I,JXIM3)
      CF(I,JXI)=EXT(1,4)*CF(I,JXIM)+EXT(2,4)*CF(I,JXIM2)+
     C EXT(3,4)*CF(I,JXIM3)
 113  CONTINUE
      DO 112 J=2,JXIM
      CA(1,J)=EXT(1,1)*CA(2,J)+EXT(2,1)*CA(3,J)+EXT(3,1)*CA(4,J)
      CB(1,J)=EXT(1,1)*CB(2,J)+EXT(2,1)*CB(3,J)+EXT(3,1)*CB(4,J)
      CC(1,J)=EXT(1,1)*CC(2,J)+EXT(2,1)*CC(3,J)+EXT(3,1)*CC(4,J)
      CD(1,J)=EXT(1,1)*CD(2,J)+EXT(2,1)*CD(3,J)+EXT(3,1)*CD(4,J)
      CE(1,J)=EXT(1,1)*CE(2,J)+EXT(2,1)*CE(3,J)+EXT(3,1)*CE(4,J)
      CF(1,J)=EXT(1,1)*CF(2,J)+EXT(2,1 *CF(3,J)+EXT(3,1)*CF(4,J)
      CA(IYP1,J)=CA(IYM1,J) $ CB(IYP1,J)=CB(IYM1,J)
      CC(IYP1,J)=-CC(IYM1,J) $ CD(IYP1,J)=-CD(IYM1,J)
      CE(IYP1,J)=-CE(IYM1,J) $ CF(IYP1,J)=CF(IYM1,J)
 112  CONTINUE
      CALL WRITELCM(CA,CAL(1,1,K),IYP1JX)
      CALL WRITELCM(CB,CBL(1,1,K),IYP1JX)
      CALL WRITELCM(CC,CCL(1,1,K),IYP1JX)
      CALL WRITELCM(CD,CDL(1,1,K),IYP1JX)
      CALL WRITELCM(CE,CEL(1,1,K),IYP1JX)
      CALL WRITELCM(CF,CFL(1,1,K),IYP1JX)
 111  CONTINUE
      AELL(1)=AELL(2)+DXM5(2)/DXP5(2)*(AELL(2)-AELL(3))
      BELL(1)=BELL(2)+DXM5(2)/DXP5(2)*(BELL(2)-BELL(3))
      CELL(1)=CELL(2)+DXM5(2)/DXP5(2)*(CELL(2)-CELL(3))
      AELL(JX)=AELL(JXM1)+DXP5(JXM1)/DXM5(JXM1)*(AELL(JXM1)-AELL(JXM2))
      BELL(JX)=BELL(JXM1)+DXP5(JXM1)/DXM5(JXM1)*(BELL(JXM1)-BELL(JXM2))
      CELL(JX)=CELL(JXM1)+DXP5(JXM1)/DXM5(JXM1)*(CELL(JXM1)-CELL(JXM2))
      RETURN
      END
      SUBROUTINE AMBI
      USE COMM
      MI=0
      DO 1 K=0,NEQ
      DENSTOR(K)=REDEN(K)
      IF (K .EQ. 0) GO TO 1
      CALL READLCM(FT,F(1,1,K),IYP1JX)
      CALL WRITELCM(FT,F2(1,1,K),IYP1JX)
    1 CONTINUE
      DO 71 J=1,JX
   71 FE1(J)=FE(J)
      S=0. $ DO 2 K=1,NEQ
      S=S+REDEN(K)*ANUMB(K)
    2 CONTINUE
      QSPLUS=S $ QSMINUS=REDEN(0)
   10 MI=MI+1
      CALL XSWEEP
      CALL GNANDE
      IF (KGNANDE .EQ. 1) RETURN
      S=0. $ DO 3 K=1,NEQ
      S=S+REDEN(K)*ANUMB(K)
```

```
    3 CONTINUE
      QPLUS=S $ QMINUS=REDEN(0)
      D1=QPLUS-QSPLUS $ D2=QMINUS-QSMINUS
      WRITE (3,465) N,MI,QSPLUS,QPLUS,D1,QSMINUS,QMINUS,D2,POTENT
  465 FORMAT(2HN=,I5,2X,3HMI=,I5,2X,8HOLD NI =,E16.6,5X,8HNEW NI =,
     C E16.6,5X,5HDIFF=,E16.6,/,19X,8HOLD NE =,E16.6,5X,8HNEW NE =,
     C E16.6,5X,5HDIFF=,E16.6,5X,7HPOTENT=,E16.6)
      IF (QSMINUS .EQ. QSPLUS .OR. N .EQ. 1 .OR. MIMAX .EQ. 0) GO TO 160
      IF (NSL .EQ. 0) GO TO 160
      IF (MI .GE. MIMAX) GO TO 460
      IF (QSPLUS .LT. QSMINUS .AND. D1 .GT. D2) GO TO 160
      IF (QSPLUS .GT. QSMINUS .AND. D1 .LT. D2) GO TO 160
      IF (QSPLUS .LT. QSMINUS) POTENT=(1.-DPOT)*POTENT
      IF (QSPLUS .GT. QSMINUS) POTENT=(1.+DPOT)*POTENT
      DO 4 K=1,NEQ
      CALL READLCM(FT,F2(1,1,K),IYPIJX)
      CALL WRITELCM(FT,F(1,1,K),IYPIJX)
    4 CONTINUE
      DO 72 J=1,JX
   72 FE(J)=FE1(J)
      CALL BOUNDARY $ IF (KBOUND .EQ. 1) RETURN
      CALL CHPDST
      CALL GNANDE
      IF (KGNANDE .EQ. 1) RETURN
      DO 5 K=1,NEQ
      CALL READLCM(FT,F(1,1,K),IYPIJX)
      FACT=DENSTOR(K)/REDEN(K)
      DO 6 I=1,IYP1 $ DO 6 J=1,JX1
    6 FT(I,J)=FACT*FT(I,J)
      REDEN(K)=DENSTOR(K)
      CALL WRITELCM(FT,F(1,1,K),IYPIJX)
      CALL WRITELCM(FT,F2(1,1,K),IYPIJX)
    5 CONTINUE
      GO TO 10
  160 CONTINUE
      DO 31 K=1,NEQ
      CALL READLCM(FT,F2(1,1,K),IYPIJX)
      CALL WRITELCM(FT,F1(1,1,K),IYPIJX)
   31 CONTINUE
      RETURN
  460 KAMBI=1
      WRITE (3,461)
  461 FORMAT(26HAMBI POT REQ NOT SATISFIED)
      RETURN
      END
      SUBROUTINE CHPDST
      USE COMM
      IF (NSL .EQ. 0) RETURN
      DO 1 K=1,NEQ
      CALL READLCM(FT,F(1,1,K),IYPIJX)
      DO 4 J=1,JX1
      DO 5 I=1,IO(J,K)
    5 FT(I,J)=0.
    4 FT(IYP1,J)=FT(IYM1,J)
      CALL WRITELCM(FT,F(1,1,K),IYPIJX)
    1 CONTINUE
      RETURN
      END
      SUBROUTINE GNANDE
      USE COMM
```

```
      DO 1 K=1,NEQ
      CALL READLCM(FT,F(1,1,K),IYP1JX)
      GN=0. $ EN=0.
      DO 7 J=1,JX1 $ DO 2 I=1,IY
      IF (FT(I,J) .LT. 0.) FT(I,J)=0.
      TERM=FT(I,J)*SPHERI(I,J)
      GN=GN+TERM $ EN=EN+TERM*XSQ(J)
    2 CONTINUE
      FT(IYP1,J)=FT(IYM1,J)
    7 CONTINUE
      CALL WRITELCM(FT,F(1,1,K),IYP1JX)
      IF (GN .EQ. 0.) GO TO 3
      ENGY(K)=EN/GN $ ENERGY(K)=ENGY(K)*EIONS(K)
      IF (N .GT. 0) GO TO 13
      GNI=1./GN $ DO 11 J=1,JX1 $ DO 11 I=1,IYP1
   11 FT(I,J)=FT(I,J)*GNI
      GN=1. $ CALL WRITELCM(FT,F(1,1,K),IYP1JX)
      CALL WRITELCM(FT,FI(1,1,K),IYP1JX)
   13 CONTINUE
      DENS(K)=GN
      IF (N .EQ. 0) GO TO 4
      REDEN(K)=ANORMK(K)*DENS(K) $ GO TO 1
    4 ANORMK(K)=REDEN(K)/DENS(K)
      GO TO 1
    3 WRITE (3,5) K $ KGNANDE=1
    5 FORMAT(17HZERO DENSITY - K=,I5)
    1 CONTINUE
      GN=0. $ EN=0. $ K=0
      DO 21 J=1,JX
      IF (FE(J) .LT. 0.) FE(J)=0.
      TERM=FE(J)*CINT(J)*XSQ(J)
      GN=GN+TERM $ EN=EN+TERM*XSQ(J)
   21 CONTINUE
      IF (GN .EQ. 0.) GO TO 22
      ENGY(K)=EN/GN $ ENERGY(K)=ENGY(K)*EIONS(K)
      GN=FOURPI*GN
      IF (N .GT. 0) GO TO 23
      GNI=1./GN $ DO 31 J=1,JX
   31 FE(J)=FE(J)*GNI
      GN=1.
   23 CONTINUE
      DENS(K)=GN
      IF (N .EQ. 0) GO TO 24
      REDEN(K)=ANORMK(K)*DENS(K) $ RETURN
   24 ANORMK(K)=REDEN(K)/DENS(K)
      RETURN
   22 WRITE (3,5) K $ KGNANDE=1
      RETURN
      END
      SUBROUTINE QCALC
      USE COMM
      COMMON/MQCAL/LT,TEMPGT(JX1)
      DIMENSION SIGMA(JX1)
      DIMENSION ANSWER(2)
      IF (NEQ .LT. 2) RETURN
      S1=ABSF(1.-EMASS(1)/3.3433E-24) $ S2=ABSF(1.-ANUMB(1))
      S3=ABSF(1.-EMASS(2)/5.0150E-24) $ S4=ABSF(1.-ANUMB(2))
      IF (S1+S2+S3+S4 .GT. 1.E-10) RETURN
      CALL READLCM(FT,F(1,1,1),IYP1JX)
      CALL READLCM(FN,F(1,1,2),IYP1JX)
```

```
      DO 1 J=1,JXI
      T=0.
      S=0.
      DO 2 I=1,IY
      T=T+CYNT(I)*SINN(I)*FN(I,J)
    2 S=S+CYNT(I)*SINN(I)*FT(I,J)
      TEMPGT(J)=T
    1 TEMPG(J)=S
      WRITE (3,631) ((JL,TEMPG(JL),TEMPGT(JL)),JL=1,JXI)
  631 FORMAT(I5,2E16.6)
      UMU=EIONS(1)
      DO 30 LT=1,2
      IF (LT .EQ. 1) GO TO 31
      CALL FINDSIG(UMU,SIGMA)
      GO TO 32
   31 CONTINUE
      DO 33 J=1,JXI
   33 SIGMA(J)=1.
   32 CONTINUE
      QANS=0.
      DO 20 J=1,JXI
   20 QANS=QANS+CINT(J)*EVALS(J,SIGMA)
      ANSWER(LT)=QANS
   30 CONTINUE
      SIGV=0. $ IF (ANSWER(1) .EQ. 0.) GO TO 7
      SIGV=VNORM*ANSWER(2)/ANSWER(1)
    7 CONTINUE
      EDT=2.24E+4
      ENUM=REDEN(1)*REDEN(2)*SIGV*EDT
      DENOM=0. $ DO 18 K=1,2 $ DO 18 KS=1,NSOR
      SENGY=ESOR(KS,K)
      IF (SENGY .EQ. 0.) SENGY=1.5*EIONS(K)/XSOR(KS,K)
      CUR=ASOR(KS,K)+BSOR(KS,K)*SQRTF(REDEN(K))+CSOR(KS,K)*REDEN(K)
   18 DENOM=DENOM+CUR*SENGY
      QDT=0. $ IF (DENOM .EQ. 0.) GO TO 19
      QDT=ENUM/DENOM
   19 CONTINUE
      WRITE (3,21) SIGV,QDT
   21 FORMAT(/,5HSIGV=,E16.6,5X,2HQ=,E16.6)
      RETURN
      END
      FUNCTION EVALS(J,SIGMA)
      USE COMM
      COMMON/MQCAL/LT,TEMPGT(JXI)
      DIMENSION SIGMA(1)
      DIMENSION STFG(JXI),UPART(JXI)
      DO 1 KK=1,J
    1 STFG(KK)=(XSQ(J)-XSQ(KK))*GETFG(J,KK)
      UPART(1)=0.
      DO 2 KK=2,J
      ADDIT=.5*DXM5(KK)*(STFG(KK-1)+STFG(KK))
      UPART(KK)=UPART(KK-1)+ADDIT
    2 CONTINUE
      SUM=0.
      DO 4 KK=2,J
      ARGM=X(KK-1)**LT*SIGMA(KK-1)*UPART(KK-1)
      ARGP=X(KK)**LT*SIGMA(KK)*UPART(KK)
      SUM=SUM+.5*DXM5(KK)*(ARGM+ARGP)
    4 CONTINUE
      EVALS=SUM
```

```
      RETURN
      END
      FUNCTION GETFG(J,KK)
      USE COMM
      COMMON/MQCAL/LT,TEMPGT(JXI)
      ARG1=.5*(X(J)+X(KK))
      ARG2=.5*(X(J)-X(KK))
      J1=0 $ J2=0
      DO 1 JM=1,JXIM
      IF (X(JM) .LE. ARG1 .AND. ARG1 .LT. X(JM+1)) J1=JM
      IF (X(JM) .LE. ARG2 .AND. ARG2 .LT. X(JM+1)) J2=JM
    1 CONTINUE
      IF (ARG1 .GE. X(JXI)) J1=JXIM
      IF (ARG2 .GE. X(JXI)) J2=JXIM
      FRAC1=(ARG1-X(J1))/DXP5(J1)
      FRAC2=(ARG2-X(J2))/DXP5(J2)
      ANS1=(1.-FRAC1)*TEMPG(J1)+FRAC1*TEMPG(J1+1)
      ANS2=(1.-FRAC2)*TEMPGT(J2)+FRAC2*TEMPGT(J2+1)
      ANS3=(1.-FRAC2)*TEMPG(J2)+FRAC2*TEMPG(J2+1)
      ANS4=(1.-FRAC1)*TEMPGT(J1)+FRAC1*TEMPGT(J1+1)
      GETFG=ANS1*ANS2+ANS3*ANS4
      RETURN
      END
      SUBROUTINE FINDSIG(UMU,SIGMA)
      USE COMM
      DIMENSION SIGMA(1)
      SIGMA(1)=0.
      DO 1 J=2,JXI
      EPS=UMU*XSQ(J)
      IF (EPS .GE. 20.) GO TO 41
      TEMPQ=2.25E+4*EXPF(-44.4/SQRTF(EPS))/EPS $ GO TO 40
   41 IF (EPS .GE. 30.) GO TO 42
      TEMPQ=4.62E-4*EPS*EPS-0.137 $ GO TO 40
   42 IF (EPS .GE. 50.) GO TO 43
      TEMPQ=6.70E-4*EPS*EPS-0.320 $ GO TO 40
   43 IF (EPS .GE. 87.) GO TO 44
      TEMPQ=8.55E-02*EPS-2.93 $ GO TO 40
   44 IF (EPS .GE. 107.) GO TO 45
      TEMPQ=1.98E+4/((EPS-107.)**2+4000.) $ GO TO 40
   45 IF (EPS .GE. 137.) GO TO 46
      TEMPQ=3.56E+4/((EPS-107.)**2+7200.) $ GO TO 40
   46 IF (EPS .GE. 185.) GO TO 47
      TEMPQ=8.70-0.0314*EPS $ GO TO 40
   47 IF (EPS .GE. 400.) GO TO 48
      TEMPQ=7.050E+3/(EPS*SQRTF(EPS)) $ GO TO 40
   48 TEMPQ=255./(EPS-110.)
   40 CONTINUE
      SIGMA(J)=1.E-24*TEMPQ
    1 CONTINUE
      RETURN
      END
      SUBROUTINE SIGVID
      USE COMM
      DIMENSION SIGMA(JXI)
      IF (NEQ .LT. 2) RETURN
      S1=ABSF(1.-EMASS(1)/3.3433E-24) $ S2=ABSF(1.-ANUMB(1))
      S3=ABSF(1.-EMASS(2)/5.0150E-24) $ S4=ABSF(1.-ANUMB(2))
      IF (S1+S2+S3+S4 .GT. 1.E-10) RETURN
      CALL READLCM(FT,F(1,1,1),IYPIJV
      DO 1 J=1,JXI
```

```
      S=0. $ DO 2 I=1,IY
    2 S=S+CYNT(I)*SINN(I)*FT(I,J)
    1 TEMPG(J)=S
      UM2=5./3. $ UM=SQRTF(UM2)
      S=0. $ DO 3 J=1,JXI
    3 S=S+CINT(J)*XSQ(J)*TEMPG(J)
      DENS=S $ FACTOR=UM*VNORM/DENS
      UMU=UM2*EIONS(1)
      CALL FINDSIG(UMU,SIGMA)
      S=0. $ DO 6 J=1,JXI
    6 S=S+CINT(J)*SIGMA(J)*TEMPG(J)*X(J)*XSQ(J)
      SIGVH=FACTOR*S
      RETURN
      END
```