Z-PINCH - A Multifrequency Radiative Transfer Magnetohydrodynamics Computer Code

John J. Watrous, Gregory A. Moses and Robert R. Peterson

June 1984
(revised June 1985)

UWFDM-584

FUSION TECHNOLOGY INSTITUTE

UNIVERSITY OF WISCONSIN

MADISON WISCONSIN

Z-PINCH - A Multifrequency Radiative Transfer Magnetohydrodynamics Computer Code

John J. Watrous, Gregory A. Moses and Robert R. Peterson

Fusion Technology Institute
University of Wisconsin
1500 Engineering Drive
Madison, WI 53706

http://fti.neep.wisc.edu

June 1984 (revised June 1985)

UWFDM-584

ZPINCH - A MULTIFREQUENCY RADIATIVE TRANSFER MAGNETOHYDRODYNAMICS COMPUTER CODE

John J. Watrous

Gregory A. Moses

Robert R. Peterson

Fusion Engineering Program
Department of Nuclear Engineering
University of Wisconsin-Madison
Madison, Wisconsin 53706 USA

June 1984

(Revised June 1985)

(Revised June 1986)

(Revised February 1987)

UWFDM-584

Nature of the Physical Problem

Electrical discharge channels have been proposed as a means of propagating intense light ion beams in inertial confinement fusion [1,2]. The efficiency of beam propagation depends on the structure of the channel. The ZPINCH code simulates both the formation of the channel and the hydrodynamic response of the channel to the passage of the beam.

Method of Solution

The behavior of the plasma is calculated using a radiation-magnetohydrodynamics model. The radiation hydrodynamics phase of the calculations is based on the MF-FIRE code [3]. The plasma is modeled as a single temperature fluid. Radiative transfer is treated in the multifrequency diffusion approximation. Tabulated equations of state and multifrequency mean Planck and Rosseland opacities are computed using the MIXERG code [4]. The hydrodynamic equations are modified by including the contribution to the momentum and energy of the plasma from electric and magnetic fields.

The electric and magnetic fields are calculated from the MHD approximation to Maxwell's curl equations and a simple form of Ohm's Law. These fields are driven by a capacitor bank and voltage source. By treating the channel as a distributed circuit element, rather than as a system of lumped circuit elements [5], an exact equation describing the current in the channel is obtained.

Restrictions on the Complexity of the Problem

The ZPINCH code assumes one-dimensional symmetry in cylindrical geometry, i.e. all quantities vary only with time and radius. The plasma can be divided into a maximum of 50 Lagrangian zones. Up to 20 frequency groups can be used for the radiative transfer calculation.

The external circuit driving the discharge is assumed to consist of a series connection of a resistor, an inductor, a voltage source, and a capacitor. All of the circuit component values are allowed to vary in time with a time dependence for each given by a nine-parameter function.

Unusual Features of the Program

The ZPINCH code is written in FORTRAN 77 with two exceptions: (1) NAMELIST input and (2) the manner in which the COMMON blocks are used. NAMELIST input is a Cray FORTRAN feature which allows free-format input to variables in any order. To assign a value to a variable, write an equality, VARIABLE_NAME = VALUE, in the NAMELIST input file. On computer systems without this feature, the input subroutine will have to be rewritten.

The COMMON blocks are listed only at the beginning of the program where they are equated to "COMDECKS" for the UPDATE pre-processor. UPDATE expands a "source deck" into a "compile deck" by substitution of the lines in a COMDECK for each occurrence of the appropriate "CALL" pre-processor command. Use of COMDECKS and CALL commands in this way abbreviates the listing of a program which uses the same COMMON blocks in several subroutines. A CALL command occupies just one line, whereas a COMMON block may occupy several lines. This feature is similar to the INCLUDE statement or other commands which are available on most computer systems.

References

[1] J.R. Freeman, L. Baker, D.L. Cook, "Plasma Channels for Intense-Light-Ion-Beam Reactors," Nucl. Fusion 22, 383 (1982).

[2] P.F. Ottinger, D. Mosher, S.A. Goldstein, "Microstability of a Focused Ion Beam Propagating Through a z-pinch Plasma," Phys. Fluids 22, 332 (1979).

[3] G.A. Moses, R.R. Peterson, T.J. McCarville, "MF-FIRE - A Multifrequency Radiative Transfer Hydrodynamics Code," University of Wisconsin Fusion Engineering Program Report UWFDM-458 (1983).

[4] R.R. Peterson and G.A. Moses, "MIXERG - An Equation of State and Opacity Computer Code," Comp. Phys. Comm. 28, 405 (1983).

[5] D.G. Colombant, M. Lambe, H.W. Bloomberg, "WHYRAC, A New Modular One-Dimensional Exploding Wire Code," NRL Memorandum Report 3726 (Feb. 1978).

TABLE OF CONTENTS

		PAGE
1.	Introduction	1
2.	The Equation of Motion	3
3.	The Magnetic Diffusion Equation and Current Equation	7
	3.1 Magnetic Diffusion Equation	7
	3.2 The Current Equation	17
4.	The Energy Equations	29
	4.1 Two-Temperature Option	29
	4.2 Multifrequency Option	42
5.	The Equation of State and Opacity Tables	47
6.	The Conservation Equations	49
	6.1 The Current and Magnetic Flux Conservation Checks	49
	6.2 Energy Conservation in the Circuit	51
	6.2 The Energy Conservation Check	52
7.	The Time Step Control	58
8.	The Subroutines and Their Functions	60
9.	Input/Output Units and Storage Requirements	63
10.	The Common Blocks	64
11.	Input	84
	11.1 Initial Conditions and Control Parameters	84
	11.2 Ion Beam Parameters	84
	11.3 Circuit Parameters	97
	11.4 Input Data Files	101
12.	Options	104
13.	Sample Problem	105
Acknowledgement		106
References		107

1. Introduction

The ZPINCH code was developed to simulate the formation and evolution of z-discharge plasma channels. The channel is modeled as a load in the discharge circuit. The current is driven either by a capacitor or by a time-dependent voltage source. Although the most physically consistent treatment is to include the plasma channel load in the discharge circuit in a self-consistent manner, the user has the options of decoupling the plasma load from the circuit and of specifying an arbitrary time-dependent current. The driving circuit, in addition to the capacitor and voltage source, consists of a resistor and an inductor. The values of these circuit elements are allowed to be time-dependent in a manner described below.

The discharge current provides a time-dependent boundary condition for the magnetic field in the channel. The magnetic field is found by solving the magnetic diffusion equation (MDE), a combination of Ampere's, Faraday's, and Ohm's Laws in which the displacement current is neglected. The MDE couples to the equations of plasma energy and momentum conservation through the electrical resistivity and a convective term. The electrical resistivity is a combination of a fully-ionized gas term and a weakly-ionized gas term as described below.

The discharge plasma is modeled with a single temperature coupled to a multigroup treatment of radiant energy transfer. From one to twenty groups may be used to describe the radiation field. A separate atomic physics code must be used to supply the equation of state, ionic charge state, and multi-frequency Planck and Rosseland opacities in tabular form (the MIXERG code [1], is normally used in our applications).

The motion of the discharge plasma is obtained by solving the equation of momentum conservation which includes the plasma pressure, a radiation pressure term, the J × B force, and an artificial viscosity term that helps the numerical algorithm to treat shocks.

It is assumed that there is no axial nor azimuthal dependence so that all quantities depend only on time and radius. The MHD equations are solved on a Lagrangian mesh. Up to 50 Lagrangian zones are allowed.

Each timestep in an execution is completed by an evaluation of a number of conservation equations. The equation of conservation of energy for the entire circuit is evaluated as a check on the solution of the current equation and the self-consistent treatment of the discharge load. The plasma energy equation is integrated in time and space to check that the transfer of energy from one region of the discharge plasma to another is proceeding in a consistent manner. The MDE is also integrated over space and time to check conservation of magnetic flux. A variation of the MDE is used to check the conservation of the electrical energy that the driving circuit provides to the electric and magnetic fields in the plasma.

The effect of the passage of an ion beam may also be modeled with the ZPINCH code. An ion beam current density profile may be specified by the user. This data is used in the code to calculate collisional heating and the production of the induced return current. The code does not include any beam propagation physics other than these effects on the discharge.

The ZPINCH code is based on the MF-FIRE code, a fireball simulator [2]. The algorithm for solving the MDE is derived from the MAGPIE code [10], a z-pinch simulator. The method of treating the discharge as a load in the dis-

charge circuit was inspired by the methods used in the WHYRAC exploding wire code [Ref. 5 of preceding section].

The units in the ZPINCH code are

length	- cm	pressure	- J/cm^3 = MPa
time	- second	charge	- esu
mass	- gram	magnetic field	- Gauss
speed	- cm/s	electric field	- volts/cm
energy	- Joule	current	- amperes
temperature	- eV	electrical resistivity	- second

All input and output quantities are given in these units. At various places in the solution of the MDE, all field quantities are expressed in terms of the cgs-esu system of units (magnetic field in Gauss, electric field in statvolts/cm, current in statamperes). Some of the optional debug output gives quantities in terms of these units, but all normal and post-processing output is given in terms of the units given above.

2. The Equation of Motion

The equations solved by the ZPINCH code are written in the Lagrangian coordinate system, i.e. the equations describe a point that moves with the local fluid velocity. The advantage of this coordinate system is that the mass flux is zero, so the conservation equations are simplified considerably. The code can automatically choose a suitable Lagrangian mesh from dimensions input by the user if desired.

Figure 1 illustrates the index system used to denote spatial boundaries. The Lagrangian mass of each zone, $m_{o_{j-1/2}}$, is defined by integrating

$$dm_o = \rho(r)\, r\, dr \qquad (1)$$

from boundary j-1 to j, where ρ is the mass density and r is the spatial coordinate. The Lagrangian mass is a constant for each zone, so it is a convenient replacement for the product $\rho(r)r\,dr$ when writing the conservation equations in finite difference form. The average Lagrangian mass of two zones, m_{o_j}, will appear in the finite difference form of the equation of motion, and is defined as

$$m_{o_j} = \frac{\left(m_{o_{j+1/2}} + m_{o_{j-1/2}}\right)}{2} , \tag{2}$$

and

$$\Delta m_{o_j} = m_{o_{j+1/2}} - m_{o_{j-1/2}} .$$

In Lagrangian coordinates, the equation of motion is

$$\frac{\partial u}{\partial t} = - V \frac{\partial}{\partial r} (P + q) - \frac{V}{c} J_p B , \tag{3}$$

where: V is the specific volume of the plasma,

J_p is the net current density,

u is the radial velocity of the plasma,

B is the magnetic field,

P is the sum of the plasma and radiation pressure,

c is the speed of light, and

q is the artificial viscosity [5].

The explicit, finite difference form of Eq. (3) that is solved by the ZPINCH code is

$$\frac{u_j^{n+1/2} - u_j^{n-1/2}}{\Delta t^n} = - \frac{(r_j^n)^{\delta-1}[\Delta P_j^n + \Delta q_j^{n-1/2}]}{\Delta m_{o_j}} - \frac{1}{r_j^n \Delta m_{o_j}} \left(\frac{(r_{j+1/2}^n B_{j+1/2}^n)^2 - (r_{j-1/2}^n B_{j-1/2}^n)^2}{8\pi}\right) + \frac{1}{c}\left[\frac{V_j^n + V_{j-1}^n}{2}\right][J_{Bj}^{n-1/2} \; B_j^n] \; . \tag{4}$$

The superscript n is the time index. The terms in brackets are defined as

$$\Delta P_j^n = P_{j+1/2}^n - P_{j-1/2}^n \qquad \text{and} \qquad \Delta q_j^{n-1/2} = q_{j+1/2}^{n-1/2} - q_{j-1/2}^{n-1/2} \; . \tag{5}$$

The artificial viscosity is a function of the zone specific volume, so to make Eq. (4) explicit, Δq_j is evaluated at $t^{n-1/2}$. The artificial viscosity used is

$$\begin{aligned} q_{j-1/2}^{n-1/2} &= 0 && \text{for } \dot{V}_{j-1/2}^{n-1/2} > 0 \; , \\ &= \frac{2(u_j^{n-1/2} - u_{j-1}^{n-1/2})^2}{V_{j-1/2}^{n-1/2}} && \text{for } \dot{V}_{j-1/2}^{n-1/2} < 0 \; . \end{aligned} \tag{6}$$

The quantity $\dot{V}$ is the time rate of change of the specific volume.

The plasma pressure, P_p, is computed from the perfect gas law,

$$P_{P_{j\pm1/2}}^n = 1.602 \times 10^{-19} \; (1 + Z_{j\pm1/2}^n) * n_{P_{j\pm1/2}}^n * T_{P_{j\pm1/2}}^n \; , \tag{7}$$

where: Z is the charge state of the plasma,

n_p is the number density of plasma atoms,

T_P is the plasma temperature.

The radiation pressure, P_R, is computed from the radiation energy density, E_R, by

$$P^n_{R_{j\pm1/2}} = \frac{1}{3} (E_R)^n_{j\pm1/2} \ , \tag{8}$$

where the radiation energy density has been assumed to be isotropic. Although in some instances the radiation field may not be isotropic, the radiation pressure is very small compared to the plasma pressure for the temperature and densities of interest here, so the assumption of an isotropic radiation field does not affect the plasma motion.

After solving Eq. (4) for $u_j^{n+1/2}$, the new radii are computed from

$$r_j^{n+1} = r_j^n + u_j^{n+1/2} \, \Delta t^{n+1/2} \ . \tag{9}$$

New specific volumes and other quantities are then computed in preparation for the next time step. In going from Eq. (3) to Eq. (4), the J × B force was rewritten using

$$\begin{aligned} \frac{1}{c} \hat{r} \cdot \overrightarrow{J_P \times B} &= \frac{1}{c} \hat{r} \cdot [(\vec{J}_{NET} - \vec{J}_{BEAM}) \times \vec{B}] = \frac{1}{c} J_{BEAM} B - \hat{r} \cdot \vec{B} \times (\frac{1}{4\pi} \overrightarrow{\nabla \times B}) \\ &= \frac{1}{c} J_{BEAM} B - \frac{1}{4\pi} B \frac{1}{r} \frac{\partial}{\partial r} (rB) = \frac{1}{c} J_{BEAM} B - \frac{1}{r^2} \frac{\partial}{\partial r} (\frac{r^2 B^2}{8\pi}) \\ &= \frac{1}{c} J_{BEAM} B - \frac{1}{V} \frac{1}{r} \frac{\partial}{\partial m} (\frac{r^2 B^2}{8\pi}) \ . \end{aligned} \tag{10}$$

This term represents the magnetic pressure force that causes the channel to pinch.

The user has the option of either fixing the velocity in the outermost zone at zero, or allowing it to be calculated from the equation of motion. The first choice corresponds to a wall-confined z-pinch, the second corresponds to a z-pinch in an infinite medium.

Subroutines

The equation of motion is solved in HYDRO. HYDRO calls QUE, which computes the artificial viscosity term, and NUMDEN, which computes the new number densities. The magnetic pressure term is computed in MAGNET. The plasma pressure term is computed in EOS.

3. The Magnetic Diffusion Equation and Current Equation

3.1 Magnetic Diffusion

The equation governing the magnetic field is obtained by combining Ampere's Law

$$\vec{\nabla \times B} = \frac{4\pi}{c} \vec{J}_{NET} ,$$

Faraday's Law

$$\vec{\nabla \times E_\ell} = - \frac{1}{c} \frac{\partial \vec{B}}{\partial t} ,$$

and Ohm's Law

$$\eta \vec{J}_p = \vec{E}_f$$

where $\vec{J}_{NET} = \vec{J}_{BEAM} + \vec{J}_{RETURN} + \vec{J}_D$, and $\vec{E}_f = \vec{E}_\ell - \frac{1}{c} \vec{u \times B}$. $\vec{J}_{BEAM}$ is the beam current density, $\vec{J}_{RETURN}$ is the induced return current density, and $\vec{J}_D$ is the discharge current density. $\vec{J}_p$ is the plasma conduction current, $\vec{J}_p = \vec{J}_{RETURN} + \vec{J}_D$. For $\vec{B} = B(r,t)\hat{\theta}$, $\vec{E} = E(r,t)\hat{z}$, and $\vec{J}_{NET} = J_{NET}(r,t)\hat{z}$, the result is

$$\frac{1}{c}\frac{\partial B}{\partial t} - \frac{\partial}{\partial r}\frac{\eta c}{4\pi}\frac{1}{r}\frac{\partial}{\partial r} rB + \frac{1}{c}\frac{\partial}{\partial r} uB = -\frac{\partial}{\partial r}(\eta J_{BEAM}) \ . \tag{11}$$

By using the Lagrangian mass, Eq. (1), and a number of identities, Eq. (11) is cast into a form more amenable to finite differencing in the Lagrangian frame [4].

First, the convective term in Eq. (11) is split to rewrite the time derivative as a convective time derivative:

$$\frac{\partial B}{\partial t} + \frac{\partial}{\partial r} uB = \frac{\partial B}{\partial t} + u\frac{\partial B}{\partial r} + B\frac{\partial u}{\partial r} = \frac{DB}{Dt} + B\frac{\partial u}{\partial r}$$

where $$\frac{D}{Dt} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial r} = \frac{\partial}{\partial t}\Big|_{\text{constant m}} \ .$$

Then, using $\frac{\partial}{\partial r} = \frac{r}{V}\frac{\partial}{\partial m}$, Eq. (11) becomes

$$\frac{1}{c}\frac{DB}{Dt} - \frac{r}{V}\frac{\partial}{\partial m}\frac{\eta c}{4\pi}\frac{1}{r}\frac{\partial}{\partial r} rB - \frac{1}{c}\frac{r}{V} B\frac{\partial u}{\partial m} = -\frac{\partial}{\partial r}(\eta J_{BEAM}) \ . \tag{12}$$

By using a new dependent variable

$$G(r,t) \equiv rB(r,t) \ ,$$

and using the identity

$$\frac{DB}{Dt} + B\frac{r}{V}\frac{\partial u}{\partial m} = \frac{1}{r}\frac{DG}{Dt} + G\frac{r}{V}\frac{\partial}{\partial m}\left(\frac{u}{r}\right) \ ,$$

Eq. (12) may be rewritten as

$$\frac{1}{c}\frac{1}{r}\frac{DG}{Dt} + \frac{1}{c} G \frac{r}{V}\frac{\partial}{\partial m}\left(\frac{u}{r}\right) - \frac{r}{V}\frac{\partial}{\partial m}\frac{\eta c}{4\pi}\frac{1}{r}\frac{\partial G}{\partial r} = -\frac{\partial}{\partial r}\left(\eta J_{BEAM}\right) . \tag{13}$$

Multiplying through by V/r and using V/r = dr/dm, Eq. (13) becomes

$$\frac{1}{c}\frac{1}{r}\frac{dr}{dm}\frac{DG}{Dt} - \frac{\partial}{\partial m}\frac{\eta c}{4\pi}\frac{1}{r}\frac{\partial G}{\partial r} + \frac{1}{c} G \frac{\partial}{\partial m}\left(\frac{u}{r}\right) = -\frac{dr}{dm}\frac{\partial}{\partial r}\left(\eta J_{BEAM}\right) . \tag{14}$$

This is the form of the magnetic diffusion equation that will be treated with finite difference techniques. G is calculated at the center of each zone, while u and r are known at the zone boundaries (see Section 2). After multiplying Eq. (14) by $\Delta m_{j-1/2}$, the finite difference form of the equation is

$$\begin{aligned}\frac{1}{c}\left(\frac{\Delta r}{r}\right)^{n+1/2}_{j-1/2}\left(\frac{G^{n+1}_{j-1/2} - G^{n}_{j-1/2}}{\Delta t^{n+1/2}}\right) - \left[\left(\frac{\eta c}{4\pi}\frac{1}{r}\frac{\partial G}{\partial r}\right)^{n+1/2}_{j} - \left(\frac{\eta c}{4\pi}\frac{1}{r}\frac{\partial G}{\partial r}\right)^{n+1/2}_{j-1}\right] \\ + \frac{1}{c} G^{n+1/2}_{j-1/2}\left(\frac{U^{n+1/2}_{j}}{r^{n+1/2}_{j}} - \frac{U^{n+1/2}_{j-1}}{r^{n+1/2}_{j-1}}\right) = -\left(\left(\eta J_{BEAM}\right)^{n+1/2}_{j} - \left(\eta J_{BEAM}\right)^{n+1/2}_{j-1}\right)\end{aligned} \tag{15}$$

Each of the two terms in square brackets are calculated as

$$\begin{aligned}\left(\frac{\eta c}{4\pi}\frac{1}{r}\frac{\partial G}{\partial r}\right)^{n+1/2}_{j} &= \left(\frac{\eta c}{4\pi}\frac{1}{r\Delta r}\right)^{n+1/2}_{j}\left[\frac{1}{2}\left(G^{n+1}_{j+1/2} + G^{n}_{j+1/2}\right) - \frac{1}{2}\left(G^{n+1}_{j-1/2} + G^{n}_{j-1/2}\right)\right] \\ &= \frac{c}{4\pi}\left[\frac{2}{r^{n+1/2}_{j}}\left(\frac{\Delta r^{n+1/2}_{j+1/2}}{-\eta^{n+1/2}_{j+1/2}} + \frac{\Delta r^{n+1/2}_{j-1/2}}{+\eta^{n+1/2}_{j-1/2}}\right)^{-1}\right]\left[\frac{1}{2}\left(G^{n+1}_{j+1/2} + G^{n}_{j+1/2}\right)\right. \\ &\quad \left. - \frac{1}{2}\left(G^{n+1}_{j-1/2} + G^{n}_{j-1/2}\right)\right] .\end{aligned} \tag{16}$$

The harmonic mean in the resistivity term is used to insure a meaningful

averaging across zones of widely varying resistivity. The convective term in Eq. (15) is approximated as

$$\frac{1}{c} G_{j-1/2}^{n+1/2} \left(\frac{u_j^{n+1/2}}{r_j^{n+1/2}} - \frac{u_{j-1}^{n+1/2}}{r_{j-1}^{n+1/2}}\right) = \frac{1}{c} \left(\frac{G_{j-1/2}^{n+1} + G_{j-1/2}^{n}}{2}\right)\left[\frac{r_{j-1}^{n+1/2} u_j^{n+1/2} - r_j^{n+1/2} u_{j-1}^{n+1/2}}{r_j^{n+1/2} r_{j-1}^{n+1/2}}\right] \tag{17}$$

$$\cong \frac{1}{c} \left(\frac{G_{j-1/2}^{n+1} + G_{j-1/2}^{n}}{2}\right)\left[\frac{r_{j-1}^{n+1/2} u_j^{n+1/2} - r_j^{n+1/2} u_{j-1}^{n+1/2}}{r_{j-1/2}^{2}}\right]$$

for $\Delta r_j \ll r_j$.

Defining the coefficients

$$a_j^{n+1/2} \equiv \frac{c}{4\pi} \left[\frac{2}{r_j^{n+1/2}} \left(\frac{\Delta r_{j+1/2}^{n+1/2}}{-\eta_{j+1/2}^{n+1/2}} + \frac{\Delta r_{j-1/2}^{n+1/2}}{+\eta_{j-1/2}^{n+1/2}}\right)^{-1}\right] \tag{18}$$

$$\alpha_{j-1/2}^{n+1/2} \equiv \frac{1}{c\Delta t^{n+1/2}} \left(\frac{\Delta r}{r}\right)_{j-1/2}^{n+1/2} \tag{19}$$

$$\beta_{j-1/2}^{n+1/2} \equiv \frac{1}{2c} \left(\frac{r_{j-1}^{n+1/2} u_j^{n+1/2} - r_j^{n+1/2} u_{j-1}^{n+1/2}}{(r_{j-1/2}^{n+1/2})^2}\right) , \tag{20}$$

$$\gamma_{j-1/2}^{n+1/2} \equiv (\eta J_{BEAM})_j^{n+1/2} - (\eta J_{BEAM})_{j-1}^{n+1/2} ,$$

Eq. (15) may be rewritten as

$$\alpha_{j-1/2}^{n+1/2}\,(G_{j-1/2}^{n+1} - G_{j-1/2}^{n}) = \frac{1}{2}\{a_j^{n+1/2}[(G_{j+1/2}^{n+1} + G_{j+1/2}^{n}) - (G_{j-1/2}^{n+1} + G_{j-1/2}^{n})] \tag{21}$$

$$- a_{j-1}^{n+1/2}[(G_{j-1/2}^{n+1} + G_{j-1/2}^{n}) - (G_{j-3/2}^{n+1} + G_{j-3/2}^{n})]\}$$

$$- \beta_{j-1/2}^{n+1/2}\,(G_{j-1/2}^{n+1} + G_{j-1/2}^{n}) - \gamma_{j-1/2}^{n+1/2}\,.$$

Equation (21) is then cast into the standard algorithm [4]

$$\Delta G_j^{n+1} = E_{j-1/2}^{n+1/2}\,G_{j+1/2}^{n+1} - F_{j-1/2}^{n+1/2}$$

where $\Delta G_j^{n+1} \equiv G_{j+1/2}^{n+1} - G_{j-1/2}^{n+1}$.

Define the new coefficients

$$A_{j-1/2}^{n+1/2} \equiv \frac{1}{2}\,a_j^{n+1/2}$$

$$B_{j-1/2}^{n+1/2} \equiv \alpha_{j-1/2}^{n+1/2} + \beta_{j-1/2}^{n+1/2}$$

$$C_{j-1/2}^{n+1/2} \equiv \frac{1}{2}\,a_{j-1}^{n+1/2}$$

$$D_{j-1/2}^{n+1/2} \equiv G_{j-1/2}^{n}\,(\alpha_{j-1/2}^{n+1/2} - \beta_{j-1/2}^{n+1/2}) + A_{j-1/2}^{n+1/2}\,\Delta G_j^{n} - C_{j-1/2}^{n+1/2}\,\Delta G_{j-1}^{n} - \gamma_{j-1/2}^{n+1/2}\,.$$

With these, Eq. (21) may be written as

$$B_{j-1/2}^{n+1/2}\,G_{j-1/2}^{n+1} + C_{j-1/2}^{n+1/2}\,\Delta G_{j-1}^{n+1} - A_{j-1/2}^{n+1/2}\,\Delta G_j^{n+1} = D_{j-1/2}^{n+1/2}\,. \tag{22}$$

By assuming $\Delta G_{j-1}^{n+1/2} = E_{j-3/2}^{n+1/2}\,G_{j-1/2}^{n+1} - F_{j-3/2}^{n+1}$ and using

$$G^{n+1}_{j-1/2} = G^{n+1}_{j+1/2} - \Delta G^{n+1}_{j} \ ,$$

Eq. (22) becomes

$$B^{n+1/2}_{j-1/2} \left[G^{n+1}_{j+1/2} - \Delta G^{n+1}_{j}\right] + C^{n+1/2}_{j-1/2} \left\{E^{n+1/2}_{j-3/2} \left(G^{n+1}_{j+1/2} - \Delta G^{n+1}_{j}\right) - F^{n+1/2}_{j-3/2}\right\}$$

$$- A^{n+1/2}_{j-1/2} \Delta G^{n+1}_{j} = D^{n+1/2}_{j-1/2} \ . \tag{23}$$

Solving Eq. (23) for ΔG^{n+1}_{j} gives

$$\Delta G^{n+1}_{j} = E^{n+1/2}_{j-1/2} G^{n+1}_{j+1/2} - F^{n+1/2}_{j-1/2} \qquad \text{for } 1 \leq j < \text{JMAX} \tag{24}$$

where $$E^{n+1/2}_{j-1/2} = \frac{B_{j-1/2} + C_{j-1/2} E_{j-3/2}}{A_{j-1/2} + B_{j-1/2} + C_{j-1/2} E_{j-3/2}} \qquad 2 \leq j \leq \text{JMAX}$$

and $$F^{n+1/2}_{j-1/2} = \frac{C_{j-1/2} F_{j-3/2} + D_{j-1/2}}{A_{j-1/2} + B_{j-1/2} + C_{j-1/2} E_{j-3/2}} \qquad 2 \leq j \leq \text{JMAX}$$

(all at time level n+1/2). First, an outward sweep is performed to calculate the A, B, C, D, E, and F coefficients. The magnetic field is then constructed from Eq. (24) by starting at the outer zone with the boundary condition

$$G^{n+1}_{jmax-1/2} = \frac{2}{c} I^{n+1}_{D}$$

where I^{n+1}_{D} is the discharge current in the channel. An inward sweep on Eq. (24) yields all the new values of $G^{n+1}_{j-1/2}$. When an ion beam is included in the

run, the net current in the channel ranges from $I_{NET} = I_D$ to $I_{NET} = I_D + I_{BEAM}$. However, the appropriate boundary condition is always $G^{n+1}_{jmax-1/2} = \frac{2}{c} I_D^{n+1}$. This is due to the neglect of the displacement current in Ampère's Law. Suppose that I_{NET} were used in the boundary condition rather than I_D. Without the capacitive coupling represented by the displacement current, the code can interpret current only as diffusive conduction current. The result is that the current assigned in the boundary condition starts as a skin current flowing at $r = r_{max}$ and diffuses inward. However, the source term $(-\frac{\partial}{\partial r} nJ_{BEAM})$ in the magnetic diffusion equation still drives a return current that tends to neutralize the beam's current density. The code will then calculate that the net current is I_{NET}, the current that was used for the boundary condition, plus whatever fraction of the beam current that was not neutralized. The consequence is that the code continually calculates more current than is actually there. In other words, the use of I_{NET} in the boundary condition, rather than I_D, results in an artificial production of current.

The outer sweep to calculate the new E and F coefficients requires the values of $E_{1/2}$ and $F_{1/2}$, which are <u>a priori</u> unknown. The choice of values for these coefficients reflects a choice on the axial boundary condition for $G = rB$. From Ampère's Law, the gradient of G is

$$\frac{\partial G}{\partial r} = \frac{2}{c} r J(r,t)$$

and this should approach zero as $r \to 0$ for physically meaningful current densities. The choice of $E_{1/2} = F_{1/2} = 0$ reflects this zero-gradient boundary condition in the numerical solution. However, this axial boundary condition

does not uniquely choose the desired solution to Eq. (24). The steady-state solutions to Eq. (14) with the boundary condition $\partial G/\partial r = 0$ at $r = 0$ for the case of no fluid velocity and constant resistivity are

$$G(r) \propto r^2$$

and

$$G(r) = \text{constant} .$$

The zero-gradient condition is automatically satisfied by both and does not select the former, which is the one that corresponds to a nonsingular current density.

The proper solution is picked by forcing G to zero in the innermost zone. Numerically, this is done by assigning a very large value, e.g. 10^{10} to $\alpha_{1/2}^{n+1/2}$ (Eq. 19). The effect of this is to make the inner zone appear to be a nearly perfect conductor. Since the magnetic field cannot diffuse into a perfect conductor, the initial value of G assigned to the inner zone changes very little over the course of the problem.

The electrical resistivity presently used in ZPINCH is from classical transport theory and includes both the effects of Coulomb collisions and electron-atom collisions. The expression used is [5]

$$\eta = \eta_W + \eta_F \tag{25}$$

where $$\eta_F \equiv (5.799 \times 10^{-15})\, Z(T_p)\, \ln \Lambda\, T_p^{-3/2} = C_F Z \ln \Lambda\, T_p^{-3/2}$$

and $$\eta_W \equiv \frac{m_e^{1/2}}{e^2} \frac{\sigma_{ea}(T_p)}{\alpha(T_p)} T_p^{1/2} = C_W \frac{\sigma_{ea}}{\alpha} T_p^{1/2} .$$

σ_{ea} is the electron-atom collision cross section, Z is the average ionic charge state, and α is the degree of ionization of the plasma. When the degree of ionization falls below 0.001, the behavior of η is controlled by η_W. For α greater than 0.1, η_F is the dominant term in η. In finite difference form, the resistivity is

$$\pm\eta_j^{n+1/2} = C_F \frac{Z_{j\pm1/2}^{n\pm1/2} \ln \Lambda_{j\pm1/2}^{n+1/2}}{T_{p_j}^{n+1/2} \sqrt{T_{p_{j\pm1/2}}^{n+1/2}}} + C_W \frac{\sigma_{ea_j}^{n+1/2}}{\alpha_{j\pm1/2}^{n+1/2}} \sqrt{T_{p_{j\pm1/2}}^{n+1/2}} \quad . \tag{26}$$

The user should be made aware that the resistivity is extremely dependent on the temperature-dependent charge state. For most gases, the charge state changes by several orders of magnitude in a narrow (~ 500-1000°K) temperature range somewhere below 1 eV. Since the evolution of the discharge depends strongly on the rapid change in resistivity in this narrow temperature range, the user should make certain that his atomic physics data gives a temperature dependence for the charge state that conforms to known experimental or theoretical data. References [5], [14], and [15] have been of use to the authors in confirming the charge state and electrical resistivity of argon and N_2.

The electron-atom cross-section currently used in the code is a constant value of 1.263×10^{-16} cm^2 that was chosen to be representative of the low-temperature cross-sections of many gases. Reference [5] provides more complete data, giving graphs of the temperature-dependent cross-section for several gases. However, the exact value used for the cross-section has very little effect of the evolution of the discharge.

Once the magnetic field has been found, the current density and electric field are calculated from it through the use of Ampère's Law and Ohm's Law. First, the net current density J_{NET} is calculated using Ampère's Law:

$$J_{NET} = \frac{c}{4\pi} \frac{1}{r} \frac{\partial}{\partial r} (rB) \ .$$

Using the coefficient $a_j^{n+1/2}$ (Eq. 18) along with the definition $G = rB$, the finite difference form of Ampère's Law may be written as

$$J_{NET_j}^{n+1/2} = \frac{a_j^{n+1/2}}{\left[\frac{2}{\frac{1}{-\eta_{j+1/2}^{n+1/2}} + \frac{1}{+\eta_{j-1/2}^{n+1/2}}}\right]} \left[\frac{\Delta G_j^{n+1} + \Delta G_j^n}{2}\right] \ . \tag{27}$$

This is then used to calculate the electric field via Ohm's Law:

$${}_fE_j^{n+1/2} = \left[\frac{2}{\frac{1}{-\eta_{j+1/2}^{n+1/2}} + \frac{1}{+\eta_{j-1/2}^{n+1/2}}}\right] \left[J_{NET_j}^{n+1/2} - J_{BEAM_j}^{n+1/2}\right] \ . \tag{28}$$

The subscript f denotes that this is the electric field in a reference frame moving with the fluid. The electric field in the lab reference frame is then calculated via

$${}_\ell E_j^{n+1/2} = {}_fE_j^{n+1/2} - \frac{1}{c} u_j^{n+1/2} B_j^{n+1/2} \ . \tag{29}$$

At times, particularly when an ion beam is included in the simulation, it is useful to examine the difference between ${}_\ell E$ and ${}_fE$. The difference, $\frac{1}{c} uB$, is an electric field induced by the expansion or pinching of the magnetized plasma.

Subroutines

The magnetic diffusion equation is solved in MAGNET. MAGNET also computes the electric field, current density, Joule heating term, and magnetic pressure. ABCMAG, called by MAGNET, computes the coefficients A, B, C, D, E, and F. MAGCOF, called by ABCMAG, computes the coeffieicnts a, α, β, and γ. RESIST, called by ABCMAG, computes the plasma electrical resistivity. SIGMA, called by RESIST, computes the electron-atom collision cross-section.

3.2 The Current Equation

Many computer simulations of z-pinch configurations have used a current equation in which the channel is modeled by lumped circuit parameters. This allows the voltage drop along the channel to be calculated as

$$V_{ch} = IR_{ch} + \dot{I}L_{ch} \tag{30}$$

where R_{ch} and L_{ch} are the lumped resistance and inductance characterizing the channel. Alternatively from elementary considerations [6], one finds that the channel voltage drop may be expressed as

$$V_{ch} = \ell\eta(0,t)\ J_p(0,t) + \frac{1}{c}\frac{d\Phi}{dt} \tag{31}$$

where ℓ is the channel length and Φ is the magnetic flux in the channel. For the case of no fluid motion, the solution to the magnetic diffusion equation is of the form

$$B(r,t) = \int_0^t I(t')\ f(r,t-t')\ dt' \ . \tag{32}$$

Using this to compute

$$J(r,t) = \frac{c}{4\pi}\frac{1}{r}\frac{\partial}{\partial r} rB$$

and

$$\Phi(t) = \int_0^{r_{max}} B(r',t)\, dr' ,$$

one may see that an expression of the form of Eq. (30) cannot be obtained. One must use Eq. (31) to calculate the voltage drop in the channel. Only when the timescale chracterizing the changing discharge current is much longer than the resistive diffusion timescale is the decomposition into lumped parameters valid.

However, it is possible to express V_{ch} in a form that comes close to this sort of decomposition. To do so, let it first be assumed that the channel is contained in a gas-filled can as shown in Fig. 2. The side and left end of the can are perfect conductors and are held at ground. The right end of the can is a perfect resistor. It is between the channels point of contact with the right end and ground that the driving voltage is applied. The plasma channel lies along the axis of the cylindrical can. r_w is the radial distance between the axis and the path of the return current. Let $r = a$ be a stationary point that lies somewhere between the return current path and the outer edge of the hot plasma. By application of first Ohm's then Faraday's Laws, Eq. (31) may be rewritten as

$$V_{ch} = \ell E(a,t) + \frac{\ell}{c}\frac{\partial}{\partial t}\int_0^{r_w} B(r,t)\, dr .$$

Between $r = a$ and $r = r_w$, no current flows so that V_{ch} becomes

$$V_{ch} = \ell E(a,t) + \dot{I}(t) \frac{2\ell}{c^2} \ln \left(\frac{r_w}{a}\right) .$$

Thus, the channel voltage drop has been decomposed into an electric field term and an $L\dot{I}$ term.

In the algorithm that solves the current equation, it is the rate of change of V_{ch} that is required. Differentiating the last expression above with respect to time and using Faraday's Law to replace the electric field term with an axial electric field and the rate of change of magnetic flux gives

$$\frac{dV_{ch}}{dt} = \frac{d}{dt}\left[\ell E(0,t) + \frac{\ell}{c}\frac{\partial}{\partial t}\int_0^a B(r,t)\, dr\right] + \ddot{I}\frac{2\ell}{c^2}\ln\left(\frac{r_w}{a}\right)$$

$$= \ell \frac{\partial E(0,t)}{\partial t} + \frac{\ell}{c}\frac{\partial^2}{\partial t^2}\int_0^a B(r,t)\, dr + \ddot{I}\frac{2\ell}{c^2}\ln\left(\frac{r_w}{a}\right)$$

If we define $L_{d,plas}$, nominally the plasma inductance, as

$$L_{d,plas} \equiv \frac{\ell}{c\dot{I}(t)}\frac{\partial}{\partial t}\int_0^a B(r,t)\, dr ,$$

then the rate of change of channel voltage may be expressed as

$$\frac{dV_{ch}}{dt} = \ell \frac{\partial E(0,t)}{\partial t} + \dot{L}_{d,plas}\dot{I} + (L_{d,plas} + L_{d,vac})\ddot{I}$$

where $$L_{d,vac} \equiv \frac{2\ell}{c^2}\ln\left(\frac{r_w}{a}\right) .$$

This is the expression for dV_{ch}/dt that is used in solving the current

equation. By defining L_p in this manner, the rate of change of channel voltage may formally be decomposed into an electric field term and inductance-like terms. The reader should carefully note that this definition of L_p may not necessarily be the same as one would make in standard lumped-circuit theory.

The current equation is derived using Kirchhoff's Law for voltage drops around a series circuit consisting of the channel, a voltage supply V_{appl}, a capacitor C, a resistor R, and an inductor L, as shown in Fig. 2. The voltage supply, resistor, and inductor may vary with time. After setting the sum of voltage drops around the circuit to zero and differentiating once in time, the current equation becomes

$$L_o(t)\ddot{I} + (2\dot{L}_o(t) + R_o(t))\dot{I} + (\dot{R}_o(t) + \frac{1}{C})I = \dot{V}_{appl} - \dot{V}_{ch} . \qquad (33)$$

Regarding Eq. (33) as an inhomogeneous ODE, a standard 4th order Runge-Kutta algorithm is used to obtain the solution. Equation (33) is rewritten as

$$\ddot{I} + p(t)\dot{I} + q(t)I = s(t) \qquad (34)$$

where

$$p(t) \equiv \frac{2\dot{L}_o(t) + R_o + \dot{L}_{d,plas}}{L_o + L_{d,plas} + L_{d,vac}}$$

$$q(t) \equiv \frac{\ddot{L}_o + \dot{R}_o + 1/C}{L_o + L_{d,plas} + L_{d,vac}}$$

$$s(t) \equiv \frac{\dot{V}_{appl} - \ell \frac{\partial E(0,t)}{\partial t}}{L_o + L_{d,plas} + L_{d,vac}}$$

Equation (34) is then written as a pair of first order ODE's:

$$\dot{X}_I \equiv X_{II} = f_I(X_I, X_{II}; t)$$

$$\dot{X}_{II} = -p(t)X_{II} - qX_I + s(t) \equiv f_{II}(X_I, X_{II}; t) \tag{35}$$

where $X_I = I(t)$ and $X_{II} = \dot{I}(t)$.

Letting the index j replace I and II and denoting the time step with index i, the solving algorithm may be written as

$$X_j^{i+1} = X_j^i + \Delta t^{i+1/2} \left[\frac{1}{6} K_{j1}^i + \frac{1}{3} K_{j2}^i + \frac{1}{3} K_{j3}^i + \frac{1}{6} K_{j4}\right]$$

where $$K_{j1}^i \equiv f_j(X_I^i, X_{II}^j; t^i) ,$$

$$K_{j2}^i \equiv f_j(X_I^{i*}, X_{II}^{i*}; t^i + \frac{1}{2}\Delta t^{i+1/2}) ,$$

with $$X_j^{i*} \equiv X_j^i + \frac{1}{2}\Delta t^{i+1/2} K_{j1}^i ,$$

$$K_{j3}^i \equiv f_j(\overline{X}_I^i, \overline{X}_{II}^i; t^i + \frac{1}{2}\Delta t^{i+1/2}) ,$$

with $$\overline{X}_j^i \equiv X_j^i + \frac{1}{2}\Delta t^{i+1/2} K_{j2}^i ,$$

and $$K_{j4}^i \equiv f_j(X_j^{i\dagger}, X_{II}^{i\dagger}; t^i + \Delta t^{i+1/2}) ,$$

with $$X_j^{i\dagger} \equiv X_j^i + \Delta t^{i+1/2} K_{j3}^i .$$

Specializing the coefficients to the particular form of the f_j,

$$K_{I,1} = X_{II}^i$$

$$K_{I,2} = X_{II}^i + \frac{1}{2}\Delta t^{i+1/2} K_{I,1}$$

$$K_{I,3} = X_{II}^i + \frac{1}{2}\Delta t^{i+1/2} K_{I,2}$$

$$K_{I,4} = X_{II}^i + \Delta t^{i+1/2} K_{I,3}$$

$$K_{II,1} = -p^i X_{II}^i - q^i X_I^i + s^i$$

$$K_{II,2} = -p^{i+1/2}(X_{II}^i + \frac{1}{2}\Delta t^{i+1/2} K_{II,1}) - q^{i+1/2}(X_I^i + \frac{1}{2}\Delta t^{i+1/2} K_{II,1}) + s^{i+1/2}$$

$$K_{II,3} = -p^{i+1/2}(X_{II}^i + \frac{1}{2}\Delta t^{i+1/2} K_{II,2}) - q^{i+1/2}(X_I^i + \frac{1}{2}\Delta t^{i+1/2} K_{II,2}) + s^{i+1/2}$$

$$K_{II,4} = -p^{i+1}(X_{II}^i + \Delta t^{i+1/2} K_{II,3}) - q^{i+1}(X_I^i + \Delta t^{i+1/2} K_{II,3}) + s^{i+1} .$$

There are three options for obtaining the discharge current. The first and most physically meaningful option is to include the discharge voltage drop in a self-consistent manner in the solution of the current equation. The second option is to ignore the channel voltage drop and solve the current equation for an LRC circuit with driving term. The third option is simply to specify the current as a function of time. The third option requires that the user prepare a table containing the values of times and current at those times as described in Section 11.3.

The second option allows the user to see what sort of current a particular LRC circuit will drive, then assess the influence of the discharge voltage drop by making an identical run, but with option 1 taken. An attractive feature of this second option is that it often allows the code to use relatively large timesteps. This can cause problems in the circuit equation, however, if $I/\dot{I}$ or $\dot{I}/\ddot{I}$ are small compared to Δt, which can happen if the current passes through zero or through an extremum. To avoid such problems and to avoid placing a constraint on the timestep, the option 2 current equation solver is equipped with its own timing scheme. It subdivides the hydro timestep according to

$$\Delta t^{n}_{current} = \frac{\Delta t^{n+1/2}}{MIN\{m,\ MAX[\frac{\Delta t^{n+1/2}}{|I^{n}/\dot{I}^{n}|}\ ,\ \frac{\delta t^{n+1/2}}{|\dot{I}^{n}/\ddot{I}^{n}|}]\}} \tag{36}$$

where MAX and MIN denote the integer parts of the maximum and minimum values of their arguments, respectively. m is a user input parameter that gives the upper limit of current-timesteps to be used per hydro-timestep.

The first option, that of including the discharge voltage drop in the circuit equation, is the most difficult to handle computationally. The problem is that the rate of change of the discharge current at time t^{n+1}, $\dot{I}^{n+1}$, cannot be solved for until the rate of change of the discharge voltage drop, $\dot{V}^{n+1}_{ch}$, is known. However, $\dot{V}^{n+1}_{ch}$ cannot be evaluated until $\dot{I}^{n+1}$ is known. The solution to this dilemma is to use an iterative procedure. The procedure consists of making an initial guess of $\dot{V}^{n+1}_{ch}$, solving for $\dot{I}^{n+1}$, then using that value to make an improved guess of $\dot{V}^{n+1}_{ch}$. This process is repeated until a convergence criterion has been satisfied.

The iterative procedure that has been developed to calculate values of $L_{d,plas}^{n+1}$, $\dot{L}_{d,plas}^{n+1}$, and $\dot{E}^{n+1}$ that are consistent with the value of $\dot{I}^{n+1}$ is described in detail in the following outline. The procedure consists of three nearly identical phases which differ only in how certain extrapolations are performed. The first phase is a zeroth order guess of the values of $L_{d,plas}$, $\dot{L}_{d,plas}$, and $\dot{E}$ at time t^{n+1} and a subsequent zeroth order guess of $\dot{I}^{n+1}$ (steps I through IV). The second phase sets up and uses a more sophisticated framework for the extrapolations that are needed for calculating $L_{d,plas}^{n+1}$, $\dot{L}_{d,plas}^{n+1}$, and $\dot{E}^{n+1}$ (steps V through IX). The third phase continues with the iterations using the framework set up by the second phase. The maximum number of iterations and the convergence level are set by the user. Subscripts on the left denote quantities associated with the indicated iteration. Quantities without a left subscript are quantities whose values are accepted as the true value at that time and will not be changed by iterations. Quantities in parentheses denote the result of a calculation.

Iterative Procedure Outlined

I. Make zeroth order estimate for $L_{d,plas}^{n+1}$ $({}_{o}L_{d,plas}^{n+1})$

   A. Advance current from I^{n} to I^{n+1} using RK algorithm (I^{n+1})

   B. Advance MDE using I^{n+1} as boundary datum $(B^{n+1}(r), E^{n+1}(r), \phi^{n+1})$

   C. Calculate $\phi^{n+1/2}$ by linear interpolation between t^{n} and t^{n+1}

   D. Use integrated form of MDE as an extrapolator to make zeroth order estimate of ${}_{o}\phi^{n+3/2}$:

$$ {}_{o}\phi^{n+3/2} = \phi^{n+1} + \frac{\Delta t^{n+1/2}}{2} \left[c\left(E^{n+1}(r_{max}) - E^{n+1}(0)\right) \right. $$

$$ \left. - \int_{0}^{r_{max}} B^{n+1}(r) \, \frac{\partial u^{n+1/2}}{\partial r} \, dr \right] $$

E. Calculate $_{o}L^{n+1}_{d,plas} = \frac{\ell}{c} \frac{(_{o}\phi^{n+3/2} - \phi^{n+1/2})}{\Delta t^{n+1/2}\ \dot{I}^{n}}$

II. Make zeroth order estimate of $\frac{\partial E^{n+1}}{\partial t}(0)$ $(_{o}\frac{\partial E^{n+1}}{\partial t}(0))$

A. Use linear extrapolation to obtain $_{o}E^{n+3/2}(0)$

B. Calculate $_{o}\frac{\partial E^{n+1}}{\partial t}(0) = \frac{_{o}E^{n+3/2}(0) - E^{n+1/2}(0)}{\Delta t^{n+1/2}}$

N.B. Steps A and B are equivalent to the approximation

$$_{o}\frac{\partial E^{n+1}}{\partial t}(0) = \frac{E^{n+1}(0) - E^{n}(0)}{\Delta t^{n+1/2}}$$

III. Make zeroth order estimate of $\dot{L}^{n+1}_{d,plas}$ $(_{o}\dot{L}^{n+1}_{d,plas})$

$$_{o}\dot{L}^{n+1}_{d,plas} = \dot{L}^{n}_{d,plas}$$

IV. Make zeroth order estimate of $\dot{I}^{n+1}$ $(_{o}\dot{I}^{n+1})$ (RK algorithm)

V. Make first order estimate for $L^{n+1}_{d,plas}$ $(_{1}L^{n+1}_{d,plas})$

A. Make first order estimate for I^{n+2} $(_{1}I^{n+2})$

B. Advance MDE using $_{1}I^{n+2}$ $(_{1}B^{n+2}(r),\ _{1}E^{n+2}(r),\ _{1}\phi^{n+2})$

C. Calculate $_{1}\phi^{n+3/2}$ by linear interpolation between $_{1}\phi^{n+2}$ and ϕ^{n+1}

D. Calculate $_{1}L^{n+1}_{d,plas} = \frac{\ell}{c}\left(\frac{_{1}\phi^{n+3/2} - \phi^{n+1/2}}{\Delta t^{n+1/2}}\right)/_{o}\dot{I}^{n+1}$

VI. Make first order estimate for $\frac{\partial E^{n+1}}{\partial t}(0)$ $(_{1}\frac{\partial E^{n+1}}{\partial t}(0))$

A. Use linear interpolation to get $_{1}E^{n+3/2}(0)$

B. Calculate $_1\frac{\partial E^{n+1}}{\partial t}(0) = \frac{{}_1E^{n+3/2}(0) - E^{n+1/2}(0)}{\Delta t^{n+1/2}}$

VII. Make first order estimate for $\dot{L}^{n+1}_{d,plas}$ $(_1\dot{L}^{n}_{d,plas})$

Approximate: $_1\dot{L}^{n+1}_{d,plas} = \frac{{}_1L^{n+1}_{d,plas} - L^{n}_{d,plas}}{\Delta t^{n+1/2}}$

VIII. Make first order estimate for $\dot{I}^{n+1}$ $(_1\dot{I}^{n+1})$ (RK algorithm)

IX. Check for convergence: is $\frac{{}_1\dot{I}^{n+1} - {}_o\dot{I}^{n+1}}{{}_o\dot{I}^{n+1}} < \varepsilon$?

X. Make mth estimate for $L^{n+1}_{d,plas}$ $(_mL^{n+1}_{d,plas})$

A. Make mth estimate for I^{n+2} $(_mI^{n+2})$

B. Obtain mth estimates for ϕ^{n+2} and $E^{n+2}(0)$ by scaling with the current:

$$_m(\ )^{n+2} = \left(\frac{{}_mI^{n+2}}{{}_{m-1}I^{n+2}}\right)\ {}_{m-1}(\ )^{n+2}$$

where ( ) stands for ϕ and $E(0)$.

C. Use linear interpolation to get $_m\phi^{n+3/2}$

D. Calculate $_mL^{n+1}_{d,plas} = \left(\frac{{}_m\phi^{n+3/2} - \phi^{n+1/2}}{\Delta t^{n+1/2}}\right)/{}_{m-1}\dot{I}^{n+1}$

XI. Make mth estimate for $\frac{\partial E^{n+1}}{\partial t}(0)$ $\left(_m\frac{\partial E^{n+1}}{\partial t}(0)\right)$

A. Use linear interpolation to get $_mE^{n+3/2}(0)$

B. Calculate $_m\frac{\partial E^{n+1}}{\partial t}(0) = \frac{{}_mE^{n+3/2}(0) - E^{n+1/2}(0)}{\Delta t^{n+1/2}}$

XII. Make mth estimate for $\dot{L}^{n+1}_{d,plas}$ ($_m\dot{L}^{n+1}_{d,plas}$)

Use the approxximation: $$_m\dot{L}^{n+1}_{d,plas} = \frac{_mL^{n+1}_{d,plas} - L^{n}_{d,plas}}{\Delta t^{n+1/2}}$$

XIII. Make mth estimate for $\dot{I}^{n+1}$ ($_m\dot{I}^{n+1}$) (RK algorithm)

XIV. Check for convergence: is $$\frac{_m\dot{I}^{n+1} - {}_{m-1}\dot{I}^{n+1}}{{}_{m-1}\dot{I}^{n+1}} < \varepsilon?$$

The initial conditions needed for solving the current equation are the charge on the capacitor, the values of the elements in the LRC circuit, and the initial values of the current and rate of change of current. The user specifies all but $\dot{I}(0)$ which is calculated from the other initial values in such a way as to insure that Kirchoff's Law of Voltages is satisfied at time t = 0. When the discharge voltage drop is included in the simulation, the calculation of $\dot{I}(0)$ requires the value of $V_{ch}(0)$, which is not known. In order to do the calculation, an iterative procedure similar to that described above is used. This iterative procedure alternates between making estimates of what the discharge voltage drop will be at one timestep in the future with a given $\dot{I}(0)$ and I(0), and calculating an $\dot{I}(0)$ consistent with that voltage drop. The user specifies the maximum number of iterations and the convergence criterion. The rate of convergence of the procedure depends on the initial timestep: the smaller the initial timestep, the fewer iterations will be required to satisfy a given convergence criterion.

There are two options for characterizing the driving circuit: it may either consist of a single LRC circuit with time-dependent circuit elements or it may consist of up to three different LRC series circuits with constant

circuit elements. The second option allows simulation of switching from one driving circuit to another. In this option, the change in R and L are not included in the circuit equation, Eq. (33). With either option R, L, and $\dot{v}_o$ are allowed to vary in time with time-dependence given by

$$f(t) = f_o + f_1 \exp[-(\frac{t - t_1}{\tau_1})^{n_1}] + f_2 \exp[-(\frac{t - t_2}{\tau_2})^{n_2}]$$

where f represents R, L or $\dot{v}_o$. The time-dependence of each of these circuit elements is thus defined by nine parameters, all of which are chosen by the user. To simulate switching from one constant-element LRC circuit to another, the user should choose large values of the exponent n_1 or n_2 (values of around 50 work well) and small values for the timescales τ_1 and τ_2 (values of 10^{-10} s work well). These choices cause f(t) to behave like a sum of step-functions with changes occurring at times t_1 and t_2. The option for changing C also exists. This option should be taken only when one is simulating switching from one LRC circuit to another since the effect of varying C is not taken into account in the circuit equation. In the code, C has a time-dependence like f(t) above, except that the values of the exponents and timescales are chosen to force C to behave like a sum of step-functions. A complete list of circuit element input parameters is given in Table 8.

Subroutines

MAGCUR runs the combined calculation of current and magnetic fields. MAGBC is called if the user-defined current option is taken. MAGBC reads a table containing values of current vs. time. CURRNT is called if the user has taken the option where the discharge load is ignored. CURRNT solves the current equation using a 4th order Runge-Kutta algorithm, and calls VOLORO to

evaluate the inductance, resistance, and applied voltage in the LRC circuit. If the user decides to include the discharge voltage drop, MAGCUR calls SCLOAD, which runs the iterative solution of the coupled current equation and magnetic diffusion equation. SCLOAD calls IRK to obtain the current via the Runge-Kutta algorithm. LDIS is called to compute the inductance of the discharge channel. IDOTRK computes the rate of change of the current and calls VOLORO to evaluate the inductance, resistance, and applied voltage in the LRC circuit. MAGNET and its subordinate subroutines are called to solve the magnetic diffusion equation.

4. The Energy Equations

4.1 Two-Temperature Option

In some problems, the plasma temperature may become high enough that thermal radiation is an important energy transport mechanism. The ZPINCH code uses flux limited diffusion to model radiation transport. The absorption and emission of thermal radiation are strongly temperature dependent, so the radiation diffusion equation is solved simultaneously with the plasma temperature equation. The equations solved by the ZPINCH code are

$$C_v \frac{\partial T_P}{\partial t} = \frac{\partial}{\partial m_o}\left(r^{\delta-1} \kappa_P \frac{\partial T_P}{\partial r}\right) - \frac{\partial P_P}{\partial T_P} \dot{V} T_P - q\dot{V} + \omega_R E_R - \omega_P T_P + \Psi + S \quad (37\text{-a})$$

$$V \frac{\partial E_R}{\partial t} = \frac{\partial}{\partial m_o}\left(r^{\delta-1} \kappa_R \frac{\partial E_R}{\partial r}\right) - \frac{4}{3} E_R \dot{V} \qquad - \omega_R E_R + \omega_P T_P \quad (37\text{-b})$$

where: C_v is the specific heat at constant volume,

κ_P is the plasma thermal conductivity,

κ_R is the radiation thermal conductivity,

ω_R is the radiation absorption coefficient,

ω_P is the radiation emission coefficient,

S is the rate that internal energy is added due to collisions between the beam ions and the discharge plasma electrons,

Ψ is the Joule heating term.

In writing Eq. (37-a), the thermodynamic identity [7]

$$\frac{\partial E_P}{\partial t} + P_P \dot{V} = C_v \frac{\partial T_P}{\partial t} + \frac{\partial P_P}{\partial T_P} \dot{V} T_P \tag{38}$$

was used to replace $\frac{\partial E_P}{\partial t}$ and $P_P \dot{V}$ with terms involving T_p. To simplify the notation in the finite difference equations that follow, the time index of quantities evaluated at $t^{n+1/2}$ will be omitted. In fully implicit finite difference form, Eqs. (37-a) and (37-b) are

$$C_{v_{j-1/2}} \frac{T_{P_{j-1/2}}^{n+1} - T_{P_{j-1/2}}^{n}}{\Delta t^{n+1/2}} = \frac{1}{\Delta m_{o_{j-1/2}}} \left[\frac{r_j^{\delta-1}}{\left(\frac{\Delta r}{\kappa_P}\right)_j} \left(T_{P_{j+1/2}}^{n+1} - T_{P_{j-1/2}}^{n+1} \right) \right.$$

$$\left. - \frac{r_{j-1}^{\delta-1}}{\left(\frac{\Delta r}{\kappa_P}\right)_{j-1}} \left(T_{P_{j-1/2}}^{n+1} - T_{P_{j-3/2}}^{n+1} \right) \right] - \left(\frac{\partial P_P}{\partial T_P}\right)_{j-1/2} \dot{V}_{j-1/2} T_{P_{j-1/2}}^{n+1} \tag{39-a}$$

$$- q_{j-1/2} \dot{V}_{j-1/2} + \omega_{R_{j-1/2}} E_{R_{j-1/2}}^{n+1} - \omega_{P_{j-1/2}} T_{P_{j-1/2}}^{n+1} + S_{j-1/2}^{n} + \Psi_{j-1/2}^{n}$$

and

$$v^{n+1/2}_{j-1/2} \frac{E^{n+1}_{R_{j-1/2}} - E^{n}_{R_{j-1/2}}}{\Delta t^{n+1/2}} = \frac{1}{\Delta m_{o_{j-1/2}}} \left[\frac{r^{\delta-1}_{j}}{\frac{\Delta E_{R_j}}{(\frac{\Delta r}{\kappa_R})_j + \frac{\Delta E_{R_j}}{F_{R_j}}}} (E^{n+1}_{R_{j+1/2}} - E^{n+1}_{R_{j-1/2}}) \right.$$

$$\left. - \frac{r^{\delta-1}_{j-1}}{\frac{\Delta E_{R_{j-1}}}{(\frac{\Delta r}{\kappa_R})_{j-1} + \frac{\Delta E_{R_{j-1}}}{F_{R_{j-1}}}}} (E^{n+1}_{R_{j-1/2}} - E^{n+1}_{R_{j-3/2}}) \right] - E^{n+1}_{R_{j-1/2}} \frac{4}{3} \dot{V}_{n-1/2} \qquad (39\text{-b})$$

$$- \omega_{R_{j-1/2}} E^{n+1}_{R_{j-1/2}} + \omega_{P_{j-1/2}} T^{n+1}_{P_{j-1/2}} \quad .$$

The denominators of the terms in square brackets represent the resistance per unit area to thermal and radiative diffusion between zone centers. For instance,

$$(\frac{\Delta r}{\kappa_P})_j = \frac{1}{2} (\frac{r_{j+1} - r_j}{\kappa^+_P} + \frac{r_j - r_{j-1}}{\kappa^-_P}) \quad , \qquad (40)$$

and so on for $(\frac{\Delta r}{\kappa_P})_{j-1}$, $(\frac{\Delta r}{\kappa_R})_j$, and $(\frac{\Delta r}{\kappa_R})_{j-1}$. Equations (39-a) and (39-b) can be written in matrix form as

$$\underline{\underline{\alpha}}_{j-1/2} (\underline{\theta}^{n+1}_{j-1/2} - \underline{\theta}^{n}_{j-1/2}) = \underline{\underline{a}}_j (\underline{\theta}^{n+1}_{j+1/2} - \underline{\theta}^{n+1}_{j-1/2}) - \underline{\underline{a}}_{j-1} (\underline{\theta}^{n+1}_{j-1/2} - \underline{\theta}^{n+1}_{j-3/2}) \qquad (41)$$

$$- \underline{\underline{\gamma}}_{j-1/2} \underline{\theta}^{n+1}_{j-1/2} - \underline{\underline{\omega}}_{j-1/2} \underline{\theta}^{n+1}_{j-1/2} + \underline{\beta}_{j-1/2}$$

where

$$\underline{\underline{\alpha}}_{j-1/2} = \begin{pmatrix} C_{v_{j-1/2}} & 0 \\ 0 & V_{j-1/2} \end{pmatrix} \frac{\Delta m_{o_{j-1/2}}}{\Delta t^{n+1/2}} ,$$

$$\underline{\underline{a}}_{j} = \begin{pmatrix} r_j^{\delta-1}/(\Delta r/\kappa_P)_j & 0 \\ 0 & r_j^{\delta-1}/((\Delta r/\kappa_R)_j + \Delta E_{R_j}/F_{R_j}) \end{pmatrix} ,$$

$$\underline{\underline{\gamma}}_{j-1/2} = \begin{pmatrix} (\partial P_P/\partial T_P)_{j-1/2} \dot{V}_{j-1/2} & 0 \\ 0 & \dot{4V}_{j-1/2}/3 \end{pmatrix} \Delta m_{o_{j-1/2}} ,$$

$$\underline{\underline{\omega}}_{j-1/2} = \begin{pmatrix} \omega_P & -\omega_R \\ -\omega_P & \omega_R \end{pmatrix}_{j-1/2} \Delta m_{o_{j-1/2}} ,$$

$$\underline{\beta}_{j-1/2} = \begin{pmatrix} -q_{j-1/2} \dot{V}_{j-1/2} + \Psi^n_{j-1/2} + S^n_{j-1/2} \\ 0 \end{pmatrix} \Delta m_{o_{j-1/2}} ,$$

and $$\underline{\theta}^{n+1}_{j-1/2} = \begin{pmatrix} T_{P}^{n+1}{}_{j-1/2} \\ E_{R}^{n+1}{}_{j-1/2} \end{pmatrix} .$$

A more compact matrix equation can be written by redefining the coefficients as follows:

$$\underline{\underline{A}}_{j-1/2} = \underline{\underline{a}}_j \quad ,$$

$$\underline{\underline{B}}_{j-1/2} = \underline{\underline{\alpha}}_{j-1/2} + \underline{\underline{a}}_j + \underline{\underline{a}}_{j-1} + \underline{\underline{\gamma}}_{j-1/2} + \underline{\underline{\omega}}_{j-1/2} \quad ,$$

$$\underline{\underline{C}}_{j-1/2} = \underline{\underline{a}}_{j-1} \quad ,$$

$$\underline{D}_{j-1/2} = \underline{\underline{\alpha}}_{j-1/2}\, \underline{\theta}^n_{j-1/2} + \underline{\beta}_{j-1/2} \quad .$$

With these redefinitions, Eq. (41) becomes

$$-\underline{\underline{A}}_{j-1/2}\, \underline{\theta}^{n+1}_{j+1/2} + \underline{\underline{B}}_{j-1/2}\, \underline{\theta}^{n+1}_{j-1/2} - \underline{\underline{C}}_{j-1/2}\, \underline{\theta}^{n+1}_{j-3/2} = \underline{D}_{j-1/2} \quad . \tag{42}$$

If JMAX is the number of zone boundaries, then Eq. (42) represents a JMAX by JMAX block tridiagonal matrix equation that has two by two matrices for elements. If the coefficients of Eq. (42) are evaluated at t^n, it can be solved by Gaussian elimination. Solutions can be shown to be of the form [8]

$$\begin{aligned} &\underline{\theta}^{n+1}_{j-1/2} = \underline{\underline{E}}_{j-1/2}\, \underline{\theta}^{n+1}_{j+1/2} + \underline{F}_{j-1/2} \quad , \quad \text{for } 1 \leq j < \text{JMAX} \\ &\underline{\theta}^{n+1}_{\text{JMAX}+1/2} = \text{BOUNDARY CONDITIONS} \quad , \quad \text{for } j = \text{JMAX} \quad . \end{aligned} \tag{43}$$

The $\underline{\underline{E}}$ matrix and $\underline{F}$ vector can be related to known quantities by decreasing the spatial index of Eq. (43) by one, and substituted into Eq. (42). One finds that

$$\underline{\underline{E}}_{j-1/2} = (\underline{\underline{B}}_{j-1/2} - \underline{\underline{C}}_{j-1/2} * \underline{\underline{E}}_{j-3/2})^{-1} * \underline{\underline{A}}_{j-1/2} \quad ,$$

and

$$\underline{F}_{j-1/2} = (\underline{\underline{B}}_{j-1/2} - \underline{\underline{C}}_{j-1/2} * \underline{\underline{E}}_{j-3/2})^{-1} * (\underline{D}_{j-1/2} + \underline{\underline{C}}_{j-1/2} * \underline{F}_{j-3/2}) \tag{44}$$

for $2 \leq j < JMAX$, and

$$\underline{\underline{E}}_{1/2} = (\underline{\underline{B}}_{1/2})^{-1} * \underline{\underline{A}}_{1/2} \quad ,$$

$$\underline{F}_{1/2} = (\underline{\underline{B}}_{1/2})^{-1} * \underline{D}_{1/2} \quad ,$$

for $j = 1$. To solve Eq. (43), a sweep is made from the innermost zone out to the outer zone to evaluate $\underline{\underline{E}}$ and $\underline{F}$, and then back to the center to evaluate the components of the $\underline{\theta}^{n+1}$ vector.

The expression for the thermal conductivity of the plasma, κ_p, that is used in the ZPINCH code is the theoretical expression developed for electrons' interacting with stationary ions [9]. The theoretical expression includes an experimentally determined constant to prevent κ_p from diverging as the average ionization state approaches zero. The expression is

$$\kappa_p = 20 \left(\frac{2}{\pi}\right)^{3/2} \frac{T_p^{5/2}}{\sqrt{m_e}\, e^4\, (Z + 4)\, \ln \Lambda} \quad , \tag{45}$$

where: e is the electron charge,

m_e is the electron mass,

$\ln \Lambda$ is the Coulomb logarithm.

To save computational effort, the Coulomb logarithm is computed from a curve fit that has an error of less than 10% for $\ln \Lambda$ greater than 5. In finite

difference form, the thermal conductivities are

$$\kappa^{\pm}_{P_j} = \frac{1.22 \times 10^2 \; T^2_{P_j} \; T^{1/2}_{P_{j\pm1/2}}}{(4 + Z_{j\pm1/2}) \; \ln \Lambda_{j\pm1/2}} \quad . \tag{46}$$

The T_P^2 terms are evaluated at the zone boundaries rather than the zone centers to enhance the numerical accuracy of the solution.

The Joule heating term, Ψ, is calculated using a finite-difference form that is meant to give both the best spatial centering and to satisfy the integral equation [10]

$$\int_{\substack{\text{channel} \\ \text{volume}}} \Psi \; dm = \int_{\substack{\text{channel} \\ \text{volume}}} JEr \; dr \; .$$

The expression used is

$$\Psi^{n+1/2}_{j-1/2} = \frac{1}{\Delta m_{j-1/2}} \frac{\Delta r^{n+1/2}_{j-1/2}}{2} \left[\frac{r^{n+1/2}_j \left({}_fE^{n+1/2}_j \; {}_\ell E^{n+1/2}_j \right)}{-\eta^{n+1/2}_{j+1/2}} + \frac{r^{n+1/2}_{j-1} \left({}_fE^{n+1/2}_{j-1} \; {}_\ell E^{n+1/2}_{j-1} \right)}{+\eta^{n+1/2}_{j-3/2}} \right] .$$

Note that the beam current density is not included in the Joule heating term.

The expression for the radiation conductivity that is used in the 2-T option is a frequency averaged value. If the radiation mean free path is much smaller than the gradients in the radiation energy density, then the frequency dependent radiation flux, $q_{R\nu}$, is given by [11]

$$q_{R\nu} = \frac{- \ell_\nu(T_P) c}{3} \frac{\partial E_{R\nu}}{\partial r} \quad , \tag{47}$$

where: ℓ_ν is the frequency dependent radiation mean free path,

$E_{R\nu}$ is the frequency dependent radiation energy density,

c is the speed of light.

The frequency averaged conductivity is obtained by integrating Eq. (47) over frequency. The frequency dependence of ℓ_ν is known from theoretical models of radiation absorption, but in general the frequency dependence of $E_{R\nu}$ is not known prior to solving the frequency dependent radiation transport equations. For the 2-T radiation diffusion model used in the ZPINCH code, there are two options for estimating $E_{R\nu}$. In the first option [12] the frequency dependence of $E_{R\nu}$ is assumed to be a dilute Planckian, that is

$$E_{R\nu} = \varepsilon V \frac{8\pi h\nu^3}{c^3} \frac{1}{\exp\left(\frac{h\nu}{T_R}\right) - 1} \quad , \tag{48}$$

where ε is a proportionality factor less than 1 and T_R is the radiation temperature. The radiation temperature is defined so as to reflect the temperature of the plasma that emitted the radiation occupying the point of interest. The radiation temperature at a point is evaluated by averaging the temperature of the transported radiation, the temperature of the emitted radiation, and the temperature of the radiation already present. In finite difference form, this average is

$$T^{n+1}_{R_{j+1/2}} = \frac{W_1 * T^{n+1/2}_{R_{j-1/2}} + W_2 * T^{n+1/2}_{R_{j+3/2}} + W_3 * T^{n+1/2}_{P_{j+1/2}} + W_4 * T^{n+1/2}_{R_{j+1/2}}}{W_1 + W_2 + W_3 + W_4} \qquad 1<j<JMAX \tag{49}$$

where the weighting functions are defined as

$$W_1 = \left(\frac{q_R r^{\delta-1} \Delta t}{\Delta m_o}\right)^{n+1/2}_{j-1/2} \qquad \text{if } q_{R_{j-1/2}} > 0$$

$$= 0 \qquad \text{if } q_{R_{j-1/2}} < 0 \tag{50}$$

$$W_2 = 0 \qquad \text{if } q_{R_{j+3/2}} > 0$$

$$= \left(\frac{q_R r^{\delta-1} \Delta t}{\Delta m_o}\right)^{n+1/2}_{j+3/2} \qquad \text{if } q_{R_{j+3/2}} < 0 \tag{51}$$

$$W_3 = (\omega_P T_P \Delta t)^{n+1/2}_{j+1/2} \tag{52}$$

$$W_4 = (E_R)^{n+1/2}_{j+1/2} \quad . \tag{53}$$

In the second option the radiation temperature is simply computed as

$$T_R = \left(\frac{E_R c}{4\sigma}\right)^{1/4} \quad ,$$

where σ is the Stefan-Boltzmann constant. The frequency averaged radiation flux across zone boundaries is represented by q_R in Eqs. (50) and (51), which after integrating Eq. (47) over frequency can be written as

$$q_R = - \frac{\ell(T_P, T_R) c}{3} \frac{\partial E_R}{\partial r} \quad , \tag{54}$$

where

$$\ell(T_P, T_R) = \frac{15}{4\pi^4} \int_0^\infty \ell_\nu(T_P) \frac{U^4 e^{-U}}{(1 - e^{-U})^2} \, dU \quad , \tag{55}$$

and

$$U(T_R) = \frac{h\nu}{T_R} \quad . \tag{56}$$

Equation (55) defines the Rosseland mean free path (including spontaneous emission [11]), and is a function of the plasma density, the local plasma temperature, and the local radiation temperature. From Eq. (54), the frequency averaged radiation conductivity can be written in finite difference form as

$$\kappa_{R_j}^{\pm} = 10^{10} \, V_{j\pm1/2} \, / \, \sigma_{R_{j\pm1/2}} \tag{57}$$

where $\sigma_{R_{j\pm1/2}}$ is the Rosseland opacity (cm^2/g).

If the Rosseland mean free path is larger than the spatial zoning, then radiation may stream from zone to zone without being absorbed. In this case the diffusion model overestimates the radiation flux, and must be modified with a flux limiter. This flux limiter has been included in Eq. (39-b) where it is referred to as F_{R_j} and $F_{R_{j+1}}$. The maximum radiation flux, cE_R, occurs when the radiation intensity of free streaming radiation approaches complete anisotropy. If the radiation intensity is completely isotropic, then the flux limit is $cE_R/4$. This latter expression is used in the ZPINCH code. In finite difference form, the flux limit is

$$\begin{aligned} F_j &= 3.75 \times 10^9 \left[(E_R)_{j+1/2}^{n+1/2} + (E_R)_{j-1/2}^{n+1/2}\right] && 1 \leq j < \text{JMAX} \\ F_{\text{JMAX}} &= 7.5 \times 10^9 \, (E_R)_{\text{JMAX}+1/2}^{n+1/2} && j = \text{JMAX} \quad . \end{aligned} \tag{58}$$

The expression for the absorption coefficient used in the ZPINCH code can be obtained by integrating the frequency dependent absorption rate over frequency. From the definition of the radiation opacity, the frequency dependent absorption rate is

$$\omega_{R\nu} E_{R\nu} = cE_{R\nu}\sigma_\nu(T_P) \quad . \tag{59}$$

Using Eq. (48) to integrate Eq. (59) over frequency results in (including spontaneous emission)

$$\omega_R E_R = c E_R \sigma_P(T_R, T_P) \tag{60}$$

where

$$\rho\sigma_P(T_P, T_R) = \frac{15}{\pi^4} \int_0^\infty \frac{U^3(T_R)\, dU}{\ell_\nu(T_P)(e^{U(T_R)} - 1)} \quad , \tag{61}$$

and

$$U(T_R) = \frac{h\nu}{T_R} \quad . \tag{62}$$

Equation (61) defines the nonequilibrium Planck opacity, which is the inverse of the frequency averaged distance that radiation with a temperature T_R will travel in a plasma at temperature T_P before being absorbed. The finite difference form of the absorption coefficient is

$$\omega_{R_{j-1/2}}^{n+1/2} = 3 \times 10^{10}\ \sigma_P(T_R, T_P)_{j-1/2}^{n+1/2} \quad . \tag{63}$$

The expression for the radiation emission coefficient that is used in the ZPINCH code is obtained by assuming that the plasma is in local thermodynamic equilibrium (LTE) [13]. The plasma is in LTE if the electrons and ions are in collisional equilibrium with each other. The radiation spectrum emitted by a plasma in LTE has a Planckian frequency distribution, because the electron-ion recombination processes are the same as those that occur in a blackbody. Then the frequency dependent emission rate can be written as

$$\omega_{P\nu} T_P = \frac{cV}{\ell_\nu(T_P)} \frac{8\pi h\nu^3}{c^3} \frac{1}{\exp\left(\frac{h\nu}{T_P}\right) - 1} \quad . \tag{64}$$

Averaging Eq. (64) over frequency yields

$$\omega_P T_P = 4\sigma T_P^4\ \sigma_P(T_P) \quad , \tag{65}$$

where

$$\rho\sigma_P(T_P) = \frac{15}{\pi^4} \int_0^\infty \frac{U^3(T_P)\, dU}{\ell_\nu(T_P)(e^{U(T_P)} - 1)} \quad , \tag{66}$$

and

$$U(T_P) = \frac{h\nu}{T_P} \quad . \tag{67}$$

Equation (65) defines the equilibrium Planck opacity, which is the inverse of the average distance that radiation at a temperature T_P will travel in a plasma at temperature T_P before being absorbed. The finite difference form of the emission rate is

$$\omega_{P\,j-1/2}^{n+1/2} = 4.12 \times 10^5 \left[T_P^3\, \sigma_P(T_P)\right]_{j-1/2}^{n+1/2} \quad . \tag{68}$$

The heating of the plasma due to collisions with beam ions is handled within the framework of plasma kinetic theory. The method is to calculate the energy loss rate of a single beam ion in each zone. This energy loss rate is then summed over all the beam ions in a single zone. This total energy loss rate is then added to the plasma energy. Thus, S in Eq. (39-a) is

$$S_{j-1/2}^n = n_{BEAM_{j-1/2}} (-\dot{\varepsilon}_{BEAM_{j-1/2}}) \left[\frac{(r_j^{n+1/2})^2 - (r_{j-1}^{n+1/2})^2}{\Delta m_{j-1/2}}\right]$$

where $n_{BEAM_{j-1/2}}$ is the number density of the beam in zone j and $\dot{\varepsilon}_{BEAM_{j-1/2}}$ is the energy loss rate of a beam ion in the j^{th} zone. From plasma kinetic theory, the energy loss rate of an ion due to interactions with background species α is

$$\dot{\varepsilon}^{\alpha}_{BEAM} = -\{2\,\frac{m_{BEAM}}{m_\alpha}\left[1 - \frac{2}{\pi}\left(\frac{v_{BEAM}}{v_\alpha}\right)\exp\left(-\frac{v^2_{BEAM}}{v^2_\alpha}\right)\right]\}\,\frac{4\pi n_\alpha e^4 Z^2_{BEAM} Z^2_\alpha}{m^2_{BEAM} v^3_{BEAM}}\,\ln\Lambda$$

where v_{BEAM} is the speed of the beam ion, v_α is the thermal speed of the α species, m_{BEAM} is the beam ion mass, m_α is the mass the α species particle, and n_α is the number density of the α species. Due to the ratio m_{BEAM}/m_α, the energy loss rate is the greatest for the interaction of beam ions with discharge electrons. It should be noted that the above form of $\dot{\varepsilon}_{BEAM}$ is valid only for beam ions with speeds much greater than the thermal speed of the α species, i.e. for

$$\left(\frac{v_{BEAM}}{v_\alpha}\right)^2 = \frac{\varepsilon_{BEAM}}{T_p}\,\frac{m_\alpha}{m_{BEAM}} >> 1\ .$$

For α = electrons, this requires

$$\frac{\varepsilon_{BEAM}}{T_p} >> \frac{m_{BEAM}}{m_\alpha}\ .$$

Subroutines

ENERGY solves the coupled plasma temperature/radiation temperature equations for both the 2-T and multifrequency options. When an ion beam is included in the simulation, ENERGY calls BEAMCD, ENADBM, and ENLSCO, which compute the beam current density, the collisional heating term, and the collisional energy loss coefficient, respectively. TEMPBC is called to give the boundary condition for the temperature equation. In the 2-T option, ABCDEF is called to compute the $\underline{\underline{A}}$, $\underline{\underline{B}}$, $\underline{\underline{C}}$, $\underline{\underline{D}}$, $\underline{\underline{E}}$, and $\underline{\underline{F}}$ matrices. ABCDEF calls MATRIX to

compute the $\underline{\underline{a}}$, $\underline{\underline{\alpha}}$, $\underline{\underline{\gamma}}$, and $\underline{\underline{\omega}}$ matrices for use in calculating $\underline{\underline{A}}$-$\underline{\underline{F}}$. OMEGA computes the radiation emission and absorption coefficients in the 2-T option. KAPPA calculates the plasma and radiation thermal conductivities and the radiation flux limit in the 2-T option. LLAM is called to compute log lambda.

4.2 Multifrequency Option

In the multifrequency option we have rewritten Eqs. (37-a) and (37-b) as

$$C_v \frac{\partial T_p}{\partial t} = \frac{\partial}{\partial m_o}\left(r^{\delta-1}\,\kappa_p \frac{\partial T_p}{\partial r}\right) - \frac{\partial P_p}{\partial T_p}\dot{V}T_p - q\dot{V} + A - J + \Psi + S \qquad (69\text{-a})$$

$$V \frac{\partial E_R^g}{\partial t} = \frac{\partial}{\partial m_o}\left(r^{\delta-1}\,\kappa_R^g \frac{\partial E_R^g}{\partial r}\right) - \frac{4}{3}E_R^g \dot{V} - c\sigma_p^g E_R^g + J^g \qquad g = 1,\ldots,G \qquad (69\text{-b})$$

where: C_v is the specific heat at constant volume,

κ_p is the plasma thermal conductivity,

κ_R^g is the radiation conductivity for frequency group g,

J^g is the rate of radiation emitted by the plasma into group g,

S is the rate of internal energy added to the plasma due to collisions between beam ions and discharge plasma electrons,

Ψ is the rate of internal energy added to the plasma by the Joule heating,

σ_p^g is the Planck opacity for group g.

$$E_R^g = \int_{h\nu_g}^{h\nu_{g+1}} dh\nu\; E_R(r,h\nu,t) \qquad (70)$$

$$A^g = c\sigma_p^g E_R^g \qquad (71)$$

$$J^g = \frac{8\pi k T_P^4}{c^2 h^3} \sigma_P^g \int_{x_g}^{x_{g+1}} dx \frac{x^3}{e^x - 1} \quad ; \quad x = \frac{h\nu}{T_P} \tag{72}$$

$$\kappa_R^g = \frac{cV}{3\sigma_R^g} \tag{73}$$

σ_R^g = Rosseland opacity for group g (cm^2/g)

$$A = \sum_{g=1}^{G} A^g \tag{74}$$

$$J = \sum_{g=1}^{G} J^g \quad . \tag{75}$$

This set of G+1 equations is not solved simultaneously as in the 2-T model. Instead the multigroup equations are first solved individually and the terms A and J are computed. These terms are then explicitly included in the plasma temperature diffusion equation which is solved next. This different treatment of the multifrequency equations in no way affects the way that the 2-T method is solved in ZPINCH. This leads to a very slight inefficiency in the number of subroutines but does not affect the execution time. The coding is kept very concise by doing this.

The multigroup equations are written in finite difference form as

$$\frac{E_{R_{j-1/2}}^{g,n+1} - E_{R_{j-1/2}}^{g,n}}{\Delta t^{n+1/2}} = \frac{1}{\Delta m_{o_{j-1/2}}} \left[\frac{r_j^{\delta-1}}{\left(\frac{\Delta r}{\kappa_R^g}\right)_j + \left(\frac{\Delta E_R^g}{F_R^g}\right)_j} \left(E_{R_{j+1/2}}^{g,n+1} - E_{R_{j-1/2}}^{g,n+1}\right) \right.$$

$$\left. - \frac{r_{j-1}^{\delta-1}}{\left(\frac{\Delta r}{\kappa_R^g}\right)_{j-1} + \left(\frac{\Delta E_R^g}{F_R^g}\right)_{j-1}} \left(E_{R_{j-1/2}}^{g,n+1} - E_{R_{j-3/2}}^{g,n+1}\right) \right] - E_{R_{j-1/2}}^{g,n+1} \frac{4}{3} \dot{V}_{n-1/2} \qquad (76)$$

$$- c\sigma_{P_{j-1/2}}^{g} E_R^{g,n+1} + J_R^{g,n+1}$$

for group g. This is reduced using the notation

$$\alpha_{j-1/2}\left(E_{R_{j-1/2}}^{g,n+1} - E_{R_{j-1/2}}^{g,n}\right) = a_j\left(E_{R_{j+1/2}}^{g,n+1} - E_{R_{j-1/2}}^{g,n+1}\right) - a_{j-1}\left(E_{R_{j-1/2}}^{g,n+1} - E_{R_{j-3/2}}^{g,n+1}\right)$$

$$- \gamma_{j-1/2} E_{R_{j-1/2}}^{g,n+1} - \omega_{j-1/2} E_{R_{j-1/2}}^{g,n+1} + \beta_{j-1/2} \qquad (77)$$

where: $\alpha_{j-1/2} = V_{j-1/2}\, \Delta m_{o_{j-1/2}} / \Delta t^{n-1/2}$

$$a_j = r^{\delta-1} / \left((\Delta r / \kappa_R^g)_j + \Delta E_{R_j}^g / F_{R_j}^g \right)$$

$$\gamma_{j-1/2} = \left(4 \dot{V}_{j-1/2} / 3\right) \Delta m_{o_{j-1/2}}$$

$$\omega_{j-1/2} = c\sigma_{P_{j-1/2}}^{g} \Delta m_{o_{j-1/2}}$$

$$\beta_{j-1/2} = J_{j-1/2}^{g} \Delta m_{o_{j-1/2}} \quad .$$

The coefficients α, a, γ, ω, and β should be evaluated at $t^{n+1/2}$. However, values at that time are not yet known so they are evaluated at t^{n}. These terms are regrouped in the familiar form

$$-A_{j-1/2}E_{R_{j+1/2}}^{g,n+1} + B_{j-1/2}E_{R_{j-1/2}}^{g,n+1} - C_{j-1/2}E_{R_{j-3/2}}^{g,n+1} = D_{j-1/2} \tag{78}$$

where: $A_{j-1/2} = a_j$

$B_{j-1/2} = \alpha_{j-1/2} + a_j + a_{j-1} + \gamma_{j-1/2} + \omega_{j-1/2}$

$C_{j-1/2} = a_{j-1}$

$D_{j-1/2} = \alpha_{j-1/2}E_{R_{j-1/2}}^{g,n} + \beta_{j-1/2}$.

We then express the solution as

$$E_{R_{j-1/2}}^{g,n+1} = EE_{j-1/2} * E_{R_{j+1/2}}^{g,n+1} + FF_{j-1/2} \qquad 1 \leq j \leq JMAX \tag{79}$$

$$E_{R_{JMAX+1/2}}^{n+1} = E_{R_{BC}} \qquad \text{Boundary Condition} \quad .$$

Then we can compute

$$EE_{j-1/2} = A_{j-1/2}/(B_{j-1/2} - C_{j-1/2} * EE_{j-3/2}) \tag{80}$$

$$FF_{j-1/2} = (D_{j-1/2} + C_{j-1/2} * FF_{j-3/2})/(B_{j-1/2} - C_{j-1/2} * EE_{j-3/2}) \quad (81)$$

for $2 < j \leq JMAX$ and

$$EE_{1/2} = A_{1/2}/B_{1/2} \quad (82)$$

$$FF_{1/2} = D_{1/2}/B_{1/2} \quad (83)$$

for $j = 1$.

Once the radiation specific energies have been computed, then the absorption is computed as

$$A^g_{j-1/2} = c\sigma^g_{P_{j-1/2}} E^{g,n+1}_{R_{j-1/2}} \quad (84)$$

$$A_{j-1/2} = \sum_{g=1}^{G} A^g_{j-1/2} \quad . \quad (85)$$

The source term in the plasma temperature equation is then computed as

$$\beta_{j-1/2} = [-q_{j-1/2}\dot{V}_{j-1/2} + \Psi_{j-1/2} + S_{j-1/2} + (A - J)_{j-1/2}] \Delta m_{o_{j-1/2}} \quad . \quad (86)$$

The single plasma temperature equation is solved using the same standard implicit finite difference technique described for the 2-T and multifrequency diffusion equations.

Subroutines

In the multifrequency treatment, ENERGY calls a series of subroutines that compute the plasma and radiation transport properties separately, rather

than simultaneously, as in the 2-T treatment. ABCPLS calculates the A, B, C, D, E, and F coefficients used in solving the plasma temperature equation. ABCPLS calls PLSCOF to compute the α, a, β, γ, and ω coefficients ABCPLS calls PCOND to calculate the plasma thermal conductivity. ENERGY calls RADTR and EMISSN to control the solution of the radiation energy equation and compute the emission coefficient, respectively. RADTR calls ABCRAD to compute the radiation energy equation's A, B, C, D, E, and F coefficients for each energy group. ABCRAD calls RADCOF to compute the a, α, β, γ, and ω coefficients used to compute A-F for each energy group. RADCOF calls RCOND to calculate the radiation thermal conductivity coefficient for each energy group.

5. The Equation of State and Opacity Tables

There are six material quantities that must be supplied in tabular form by the user of the ZPINCH code. These are

$Z(n_P,T_P)$	Charge State
$E_P(n_P,T_P)$	Specific Internal Energy
$\sigma_R(n_P,T_P,T_R)$	Rosseland Opacity
$\sigma_P(n_P,T_P,T_R)$	Planck Opacity
$\sigma_R^g(n_P,T_P)$	Multigroup Rosseland Opacity
$\sigma_P^g(n_P,T_P)$	Multigroup Planck Opacity

These tables are generated for ZPINCH by the MIXERG code [1]. Logarithmic interpolation is used to interpolate between points in the tables. For instance, the charge state is stored as log $Z(\log n_P,\log T_P)$. In what follows, the indices associated with the dependent variables are

$$K - \log T_R \ ,$$

$$L - \log T_P \ ,$$

$$M - \log n_P \ .$$

Points in the two-dimensional tables can be represented as a two-dimensional grid, as shown in Fig. 3. The indices with stars denote the location of a quantity located between points in the table, for instance $\log Z(L^*,M^*)$. To compute the desired quantity we first interpolate along the M axis:

$$\log Z(L,M^*) = \log Z(L,M) + \frac{\log Z(L,M+1) - \log Z(L,M)}{\log n_p(M+1) - \log n_p(M)} * (\log n_p - \log n_p(M)) \quad , \tag{87}$$

$$\log Z(L+1,M^*) = \log Z(L+1,M) + \frac{\log Z(L+1,M+1) - \log Z(L+1,M)}{\log n_p(M+1) - \log n_p(M)} * (\log n_p - \log n_p(M)) \quad , \tag{88}$$

where n_p is the number density corresponding to $\log Z(L^*,M^*)$. Now interpolating along the L axis,

$$\log Z(L^*,M^*) = \log (L,M^*) + \frac{\log Z(L+1,M^*) - \log Z(L,M^*)}{\log T_p(L+1) - \log T_p(L)} * (\log T_p - \log T_p(L)) \quad , \tag{89}$$

where T_p is the temperature corresponding to $\log Z(L^*,M^*)$.

The grids used to interpolate in the three-dimensional tables are shown in Fig. 4. First we interpolate for $\log \sigma(M,K^*,L^*)$ and $\log \sigma(M+1,K^*,L^*)$ in the manner prescribed above. Then interpolating in the third dimension,

$$\log \sigma(M^*,K^*,L^*) = \log \sigma(M,K^*,L^*)$$

$$+ \frac{\log \sigma(M+1,K^*,L^*) - \log \sigma(M,K^*,L^*)}{\log n_p(M+1) - \log n_p(M)} (\log n_p - \log n_p(M)) \quad . \tag{90}$$

If the plasma temperature computed by solving the energy equations is less than the lowest temperature in the equation of state tables, then the code automatically computes Z and E_p by interpolating between the bounds of the table and the values for a perfect un-ionized gas. This procedure preserves the consistency of the calculation at low temperatures. The number density, n_p, should never exceed the bounds of the tables, or inaccurate results will be obtained.

The multigroup tables are treated like G two-dimensional tables where G is the number of groups.

Subroutines

EOS controls the reading of the equation of state and opacity tables and computes the charge state, specific internal energy, Rosseland opacity, Plank opacity, and multigroup Rosseland and Planck opacities. EOS calls POINT to compute pointers used in reading the tables. EOS calls TABLE2 and TABLE3 to perform logrithmic interpolations in the two- and three-dimensional tables, respectively.

6. The Conservation Equations

At the end of each timestep, the ZPINCH code performs a number of conservation checks to diagnose how consistently the calculations are proceeding.

6.1 The Current and Magnetic Flux Conservation Checks

The current conservation check is a comparison between the discharge current and the current calculated by integrating the net current density. When

there is no ion beam, this serves to indicate whether the code is producing reasonably consistent results. In most cases, the discrepancy between the discharge current and the integrated current density is less than 10%. When an ion beam is included in the run, the current conservation check serves to indicate the degree to which the beam current is being neutralized. In any case, the current conservation check compares the discharge current $I_D^{n+1/2}$, as obtained from either the current equation solver or the user specified current, and

$$I_{NET}^{n+1/2} = 2\pi \int_0^{r_{max}} J_{NET}(r,t)\, r dr \cong 2\pi \sum_{j=1}^{J_{MAX}} J_{NET_{j-1/2}}^{n+1/2} V_{j-1/2}^{n+1/2} \Delta m_{j-1/2} .$$

The magnetic flux conservation check provides an indication of how accurately the magnetic diffusion equation is being solved. It is based on integrating Eq. (11) over space and time:

$$\int_0^{r_{max}} dr' \int_0^t dt' \frac{1}{c} \frac{\partial B}{\partial t} = \int_0^{r_{max}} dr' \int_0^t dt' \frac{\partial}{\partial r'} \left[\frac{nc}{4\pi} \frac{1}{r} \frac{\partial}{\partial r} (rB) - \frac{1}{c} uB - nJ_{BEAM}\right] .$$

Using

$$\phi(t) = \int_0^{r_{max}} dr'\, B(r',t) , \qquad E_f = E_\ell + \frac{1}{c} uB, \qquad \text{and } J_{TOT} = J_{BEAM} + J_p ,$$

this becomes

$$\frac{1}{c} [\phi(t) - \phi(0)] = \int_0^t dt' \, [E_\ell(r_{max},t) - E_\ell(0,t)] .$$

Or, in finite difference form,

$$\frac{1}{c}\left(\phi^{n+1} - \phi^{o}\right) = V^{n+1}$$

where $$V^{n+1} = V^{n} + \Delta t^{n+1/2}\left[E_{\ell_{jmax}}^{n+1/2} - E_{\ell_{j=1}}^{n+1/2}\right] .$$

For most problems the error in the flux check rarely exceeds 5%.

6.2 Energy Conservation in the Circuit

The flow of energy from the driving circuit into the discharge channel may be observed by solving an energy conservation equation for the entire circuit. This equation is obtained by multiplying the current equation by the current:

$$I(LI)^{\bullet} + RI^{2} + \frac{(Q(t) - Q(0))I}{C} = I(V_{appl} - V_{ch})$$

or

$$\frac{d}{dt}\left(\frac{1}{2} L_{c} I^{2}\right) + RI^{2} + \frac{d}{dt}\left(\frac{Q^{2}}{2C}\right) = IV_{appl} - IV_{ch} .$$

Each term in this equation is evaluated at the end of each cycle to show the power that is dissipated or stored in each element of the circuit at that time. By integrating this equation in time, the total energy dissipated or stored in each element may also be found:

$$\frac{1}{2} L_{o}(t)\, I^{2}(t) - \frac{1}{2} L(0)\, I^{2}(0) + \int_{0}^{t} R_{o}(t')\, I^{2}(t')\, dt' + \frac{Q^{2}(t)}{2C} - \frac{Q^{2}(0)}{2C}$$

$$= \int_{0}^{t} I(t')\left[V_{appl}(t') - V_{ch}(t')\right] dt'$$

or, in finite difference form

$$\frac{1}{2}\left\{L_o^{n+1}(I^{n+1})^2 - L_o^{o}I^{o2} + \frac{(Q^{n+1})^2}{C} - \frac{(Q^{o})^2}{C}\right\} = A^{n+1}$$

where

$$A^{n+1} = A^{n} + \Delta t^{n+1/2}\left\{-R_o^{n+1/2}(I^{n+1/2})^2 + I^{n+1/2}V_{appl}^{n+1/2} - I^{n+1/2}V_{ch}^{n+1/2}\right\} .$$

In addition to these power and energy checks, the voltage drops on each of the elements in the circuit are evaluated to check how well Kirchhoff's Law of Voltages is being followed.

6.3 The Energy Conservation Check

Energy can be stored in and exchanged between three media: the plasma, the radiation field, and the magnetic field. In the absence of a magnetic field, the flow of energy between plasma and radiation is described by the plasma and radiation energy equations. In the presence of a magnetic field, the plasma energy equation must be modified by the inclusion of a term describing the addition of energy from Joule heating and the work done in compressing magnetic field lines. In addition, the two equations must be augmented by an equation describing the energy in the magnetic field.

To obtain an energy conservation equation for the magnetic field, Faraday's Law is applied to an infinitesimal bit of magnetic flux $d\phi = B\ dr$

$$\left(\frac{1}{c} B\ dr\right)_t = dE_\ell \ . \tag{92}$$

This equation is then multiplied through by $\frac{c}{4\pi} rB$. The left hand side of the resulting equation becomes

$$\frac{r}{4\pi} B \left(B \frac{V}{r} dm\right)_t = \frac{rB}{4\pi} \left(B_t \frac{V}{r} + \frac{B}{r} V_t - \frac{BV}{r^2} u\right) dm$$

$$= \left(V \frac{BB_t}{4\pi} + \frac{B^2}{4\pi} V_t - \frac{B^2}{4\pi} \frac{u}{r} V\right) dm .$$

Defining $w \equiv B^2/8\pi$, this becomes

$$\frac{r}{4\pi} B \left(B \frac{V}{r} dm\right)_t = \left[(Vw)_t + V_t w - 2 \frac{u}{r} Vw\right] dm .$$

Dividing through by dm, the right hand side of the modified Faraday's Law becomes

$$\frac{c}{4\pi} rB \frac{dE_\ell}{dm} = \frac{c}{4\pi} \left[(rE_\ell B)_m - E_\ell (rB)_m\right]$$

$$= \frac{c}{4\pi} \left[(rE_\ell B)_m - V \frac{c}{4\pi} E_\ell \frac{1}{r} \frac{\partial}{\partial r} (rB)\right]$$

$$= \frac{c}{4\pi} \left[(rE_\ell B)_m - VE_\ell J_{NET}\right] .$$

Thus, the magnetic energy conservation equation is

$$(Vw)_t = -V_t w + \frac{2u}{r} Vw + \frac{c}{4\pi} (rE_\ell B)_m - VE_\ell J_{NET} . \qquad (93)$$

Using $J_{NET} = J_p + J_{BEAM}$, the last term on the right hand side may be rewritten in such a way as to exhibit the energy lost through Joule dissipation and the energy contributed due to the presence of the beam:

$$VE_\ell J_{NET} = V(\eta J_p)(J_p + J_{BEAM}) = V\left[(\eta J_p^2) + (\eta J_p) J_{BEAM}\right] .$$

The first term represents the rate at which the fields lose energy due to Joule dissipation. The second term represents a rate of addition of energy due to fields induced by the ion beam current density. The terms in the second set of parentheses represent a rate of addition of energy due to fields induced by the ion beam.

Thus, the magnetic energy conservation equation is

$$(Vw)_t = -V_t w + \frac{2u}{r} Vw + \frac{c}{4\pi} (rEB)_m - V[(\eta J_p^2 + \eta J_p J_{BEAM}] \ .$$

The left hand side gives the rate of change of energy stored in the magnetic field per gram of material. The first two terms on the right hand side are work terms. The first describes the work done by the magnetic field in compressing plasma. The second describes the work done by the magnetic field in convecting the plasma. The third term on the right hand side is the divergence of the Poynting vector: it describes the rate at which the driving circuit is pouring energy into the field. The last terms, as already described, represents Joule dissipation loss and beam-induced gain.

The energy equation for the plasma is obtained by multiplying the equation of motion by the velocity u and adding the resulting equation to the plasma temperature diffusion equation. Using

$$\frac{u}{r} \left(\frac{r^2 B^2}{8\pi}\right)_m = 2u \frac{V}{r} \omega + ru\omega_m \ ,$$

the result is

$$\left(e_p + \frac{1}{2} u^2\right)_t = -(rup)_m + \dot{Q}_{DP} + \dot{Q}_{PR} - \left[2 \frac{uV}{r} \omega + ru\omega_m\right] + \Psi \ . \qquad (95)$$

The left hand side describes the rate of change of energy in the plasma. The first term on the right is the work term for the plasma. The next two terms,

$$\dot{Q}_{DP} \equiv \frac{\partial}{\partial m} r\kappa_P \frac{\partial T_P}{\partial r}$$

and

$$\dot{Q}_{PR} \equiv \omega_R E_R - \omega_P T_P$$

are the heat flux and radiation-plasma energy exchange terms, respectively. The bracketted term describes the work done by the plasma on the magnetic field. The last term is the energy addition rate from Joule heating.

The three equations that give a complete account of the energy may be written as

$$\dot{E}_B = W_{B1} - (ru)_m E_B - \Psi + S_m \qquad (96\text{-a})$$

$$\dot{E}_P = -W_{B1} - ru(E_B)_m + \Psi - W_P + \dot{Q}_{DP} + \dot{Q}_{PR} + S_P \qquad (96\text{-b})$$

$$\dot{E}_R = - W_R + \dot{Q}_{DR} - \dot{Q}_{PR} + S_R \qquad (96\text{-c})$$

where

$E_B \equiv \frac{VB^2}{8\pi}$, energy in magnetic field per gram material

$W_{PB} \equiv 2u \frac{V}{r} E_B$, work done on magnetic field by convection of field lines

$S_m \equiv \frac{c}{4\pi} (rEB)_m$, Poynting vector divergence

$E_P \equiv e + \frac{u^2}{2}$, energy per gram in plasma

$W_P \equiv (rup)_m$, the plasma work term

$W_R = (ru)_m P_R$, the radiation work term

$S_R \equiv$ source of energy to radiation

$S_P \equiv$ source of energy to plasma due to beam ion/discharge electron collisions.

After integration over space and time, these equations take the form

B-Field: $$e_B^{n+1} = e_B^0 + W_{B1}^{n+1} - W_{B2}^{n+1} - J^{n+1} + S^{n+1} \tag{97}$$

Plasma: $$e_P^{n+1} + T^{n+1} = e_P^0 + T^0 + H_P^{n+1} + E_{RP}^{n+1} - F_P^{n+1} - W_P^{n+1} - G_R^{n+1} + J^{n+1} - W_{B3}^{n+1} - W_{B1}^{n+1} \tag{98}$$

Radiation: $$e_R^{n+1} = e_R^0 + H_R^{n+1} - E_{RP}^{n+1} - F_R^{n+1} - W_R^{n+1} + G_R^{n+1} \tag{99}$$

Total: $$e^{n+1} + T^{n+1} = e^0 + T^0 + H^{n+1} + S^{n+1} - F^{n+1} - W^{n+1} \; . \tag{100}$$

The definitions of each term in the total are

$e^{n+1} = e_P^{n+1} + e_R^{n+1} + e_B^{n+1}$ -- the total internal energy of the plasma, radiation, and magnetic field;

T^{n+1} -- total kinetic energy of the plasma;

$H^{n+1} = H_P^{n+1} + H_R^{n+1}$ -- the total source of energy to the plasma and radiation (other than Joule heating);

S^{n+1} -- source of energy to the magnetic field;

$F^{n+1} = F_R^{n+1} + F_P^{n+1}$ -- total energy conducted out of channel by plasma or radiation;

$W^{n+1} = W_{B1}^{n+1} + W_{B2}^{n+1} + W_P^{n+1} + W_R^{n+1}$ -- total work done on outer boundary by plasma, radiation, and magnetic field;

G_R^{n+1} -- total work exchanged between plasma and radiation;

J^{n+1} -- Joule heating exchanged between magnetic field and plasma.

Each of these terms are given in finite difference form as follows:

$$e_x^{n+1} = \sum_{j=1}^{JMAX} (E_x)_{j-1/2}^{n+1} \Delta m_{j-1/2} \tag{101}$$

$$T^{n+1} = \frac{1}{4} \Delta m_{jmax-1/2} (u_{jmax}^{n+1/2})^2 + \frac{1}{2} \sum_{j=1}^{JMAX} \Delta m_j (u_j^{n+1/2})^2 \tag{102}$$

$$H_x^{n+1} = H_x^n + \Delta t^{n+1/2} \sum_{j=1}^{JMAX} (S_x)_{j-1/2}^{n+1/2} \Delta m_{j-1/2} \tag{103}$$

$$E_{RP}^{n+1} = E_{RP}^n + \Delta t^{n+1/2} \sum_{j=1}^{JMAX} (\dot{Q}_{RP})_{j-1/2}^{n+1/2} \Delta m_{j-1/2} \tag{104}$$

$$G_R^{n+1} = G_R^n + \Delta t^{n+1/2} \sum_{j=1}^{JMAX} u_j^{n+1/2} r_j^{n+1/2} (P_{R_{j+1/2}}^{n+1/2} - P_{R_{j-1/2}}^{n+1/2}) \tag{105}$$

$$+ \Delta t^{n+1/2} u_{jmax}^{n+1/2} r_{jmax}^{n+1/2} \frac{1}{2} (P_{R_{jmax+1}}^{n+1/2} - P_{R_{jmax-1}}^{n+1/2})$$

$$W_x^{n+1} = W_x^n + \Delta t^{n+1/2}\, u_{jmax}^{n+1/2}\, r_{jmax}^{n+1/2}\, P_{xjmax}^{n+1/2} \qquad \text{for } x = p,\ r \tag{106}$$

$$F_P^{n+1} = F_P^n + \Delta t^{n+1/2} \left[\frac{r}{\left(\frac{\Delta r}{\kappa_P}\right)}\right]_{jmax}^{n+1/2} \left(T_{P\,jmax+1/2}^{n+1/2} - T_{P\,jmax-1/2}^{n+1/2}\right) \tag{107}$$

$$F_R^{n+1} = F_R^n + \Delta t^{n+1/2} \left[\frac{r}{\left(\frac{\Delta r}{\kappa_R}\right) + \left(\frac{\Delta E_R}{F_R}\right)}\right]_{jmax}^{n+1/2} \left(E_{R\,jmax+1/2}^{n+1/2} - E_{R\,jmax-1/2}^{n+1/2}\right) \tag{108}$$

$$W_{B3}^{n+1} = W_{B3}^n + \Delta t^{n+1/2} \left[r_{jmax}^{n+1/2}\, u_{jmax}^{n+1/2}\, \frac{1}{8\pi} \left(B_{jmax+1/2}^{n+1/2}\right)^2 - W_{B1}^{n+1/2}\right] \tag{109}$$

$$W_{B2}^{n+1} = W_{B2}^n + \Delta t^{n+1/2} \sum_{j=1}^{JMAX} \left(r_j^{n+1/2}\, u_j^{n+1/2} - r_{j-1}^{n+1/2}\, u_{j-1}^{n+1/2}\right) \frac{1}{8\pi} \left(B_{j-1/2}^{n+1/2}\right)^2 \tag{110}$$

$$W_{B1}^{n+1} = W_{B1}^n + \Delta t^{n+1/2} \sum_{j=1}^{JMAX} \left(u_j^{n+1/2} + u_{j-1}^{n+1/2}\right) \Delta r_{j-1/2}^{n+1/2}\, \frac{1}{8\pi} \left(B_{j-1/2}^{n+1/2}\right)^2 \tag{111}$$

$$S^{n+1} = S^n + \Delta t^{n+1/2}\, \frac{c}{4\pi} \left[(rE_\ell B)_{J_{MAX}}^{n+1/2} - (rE_\ell B)_{j=1}^{n+1/2}\right] . \tag{112}$$

Subroutines

All of the conservation checks are performed in ECHECK.

7. The Time Step Control

After each time step, the next time step is determined from a set of stability and accuracy constraints. The new time step is determined by

$$\Delta t^{n+3/2} = = \mathrm{Max}\left[\Delta t_{min}, \mathrm{Min}\left(\Delta t_{max}, \frac{K_1}{R_1^{n+1}}, \frac{K_2 \Delta t^{n+1/2}}{R_2^{n+1}}, \frac{K_3 \Delta t^{n+1/2}}{R_3^{n+1}}, \frac{K_4 \Delta t^{n+1/2}}{R_4^{n+1}}, \frac{K_5 \Delta t^{n+1/2}}{R_5^{n+1}}, \frac{K_6 \Delta t^{n+1/2}}{R_6}\right)\right] , \tag{113}$$

where: $$R_1^{n+1} = \text{Max}[(V_{j-1/2}^{n+1} P_{j-1/2}^{n+1})^{1/2}/\Delta r_{j-1/2}^{n+1/2}] \quad (114)$$

$$R_2^{n+1} = \text{Max}[(V_{j-1/2}^{n+1} - V_{j-1/2}^{n})/V_{j-1/2}^{n+1/2}] \quad (115)$$

$$R_3^{n+1} = \text{Max}[(E_{R_{j-1/2}}^{n+1} - E_{R_{j-1/2}}^{n})/E_{R_{j-1/2}}^{n+1/2}] \quad (116)$$

$$R_4^{n+1} = \text{Max}[(T_{P_{j-1/2}}^{n+1} - T_{P_{j-1/2}}^{n})/T_{P_{j-1/2}}^{n+1/2}] \quad (117)$$

$$R_5^{n+1} = \text{Max}[(B_{j-1/2}^{n+1} - B_{j-1/2}^{n})/B_{j-1/2}^{n+1/2}] \; . \quad (118)$$

$$R_6^{n+1} = |\dot{I}^{n+1/2}|/\left|\frac{\dot{I}^{n+1} - \dot{I}^{n}}{\Delta t^{n+1/2}}\right| \quad (119)$$

The maximum values of R_1, R_2, R_3, R_4, R_5, and R_6 are found by sweeping over the zones. The input parameters K_1, K_2, K_3, K_4, and K_5 determine the severity of each constraint. The default value for K_1, K_2, K_4, K_5, and K_6 is 0.05. The default value of K_3 is set to 1.0×10^{35}, which in effect removes the radiation energy as a time step constraint. The radiation diffusion equation is solved using a fully implicit differencing scheme and is stable for large time steps. The time step can of course be constrained using the change in radiation energy density by simply inputting a different value for K_3.

A typical simulation might involve turning on an ion beam or rapidly increasing the discharge current after a period of relatively gentle hydrodynamic relaxation. Because of the disparity in timescales between the previous "quite" period and the new, more violent driven period, the option exists to set a number of sentinel times to watch out for the new driven phase of the simulation. When any one of these sentinel times is reached, the timestep is

reduced to an appropriately small value input by the user. Without this safeguard, the only way to simulate the transition from a quiet phase to a violent one is to start with an uneconomically small timestep. Up to 10 different sentinel times and corresponding entry timesteps may be used.

8. The Subroutines and Their Functions

The ZPINCH code is written in FORTRAN 66, and can be run on any mainframe computer. It is written in a topdown modular style, as shown in Fig. 5. Each subroutine performs a specific function. These functions are briefly described below:

Subroutine Name	Function of Subroutine
ABCDEF -	computes the $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$, $\underline{E}$, and $\underline{F}$ matrices and vectors used to solve the energy transfer equations when using the 2-T option.
ABCMAG -	computes A, B, C, D, E, and F coefficients used to solve the magnetic diffusion equation.
ABCPLS -	computes A, B, C, D, E, and F coefficients used to solve the plasma temperature equation when using the multifrequency radiation option.
ABCRAD -	computes A, B, C, D, E, and F coefficients used to solve the radiation energy equation for a specified frequency group when using the multifrequency radiation option.
BEAMCD -	calculates beam current density in the Lagrangian mesh using the user-input beam current density profile.
CLEAR -	sets all common blocks to zero before the start of a calculation.
CURRNT -	solves the current equation using a 4th order Runge-Kutta algorithm for the case where the discharge voltage drop is ignored.
DUMP -	writes all common blocks on unit 2 at the end of a calculation.
ECHECK -	computes the integrals used in the energy conservation check and performs the flux and current conservation checks.

EMISSN - computes the frequency dependent radiation emission when using the multifrequency radiation option.

ENERGY - computes T_P, E_R, and then T_R.

ENADBM - calculates the rate at which the beam is adding energy to the plasma due to beam ion-discharge electron collisions.

ENLSCO - calculates the specific energy loss rate ($eV\text{-}s^{-1}/eV$) of the beam ions using the standard kinetic theory expression for the energy loss rate of an ion in a plasma.

EOS - computes the equation of state quantities.

FOROUT - writes formatted output to units 10 and 11 for postprocessing.

HYDRO - solves the equation of motion for the fluid velocity, new zone radii, Δr's, zone volumes, and specific volumes.

INITIA - reads namelist input and calls other initialization routines.

INIT1 - sets variable default values before reading input.

INIT2,3,3a,4,5 - computes initial conditions and writes a summary of the initial conditions to unit 6.

KAPPA - computes plasma and radiation thermal conductivity and the radiation flux limit, in the 2-T approximation.

LDIS - calculates the inductance of the discharge channel.

LLAM - computes log lambda.

MAGBC - assigns the magnetic boundary condition, i.e. the current using a user defined function of time.

MAGCOF - computes α, β, and a coefficients used to solve the magnetic diffusion equation.

MAGCUR - calls a combination of routines for obtaining the discharge current and solving the magnetic diffusion equation.

MAGNET - computes B, E, J, Joule heating, and magnetic pressure.

MAIN - calls other routines to form the loop for one time step.

MATRIX - computes $\underline{\underline{a}}$, $\underline{\underline{\alpha}}$, $\underline{\gamma}$, and $\underline{\underline{\omega}}$ matrices for use in the energy transfer calculation, in the 2-T approximation.

NUMDEN - computes number densities from the specific volume.

OMEGA - computes the radiation emission and absorption coefficients, in the 2-T approximation.

OUT,OUT3 - writes output to unit 6 at the end of specified time cycles.

PCOND - computes the plasma conductivity when using the multifrequency radiation option.

PLSCOF - computes α, γ, a, and β coefficients used to solve the plasma temperature equation when using the multifrequency radiation option.

POINT - finds pointers in the equation of state tables.

QUE - computes the artificial viscosity.

QUIT - wraps up the calculation at the end.

RADCOF - computes α, γ, ω, a, and β coefficients used to solve the radiation energy equation for a specified frequency group when using the multifrequency radiation option.

RADTR - controls the multifrequency radiation calculation by calling EMISSN and ABCRAD and then computes the total radiation energy density and radiation temperature when using the multifrequency radiation option.

RCOND - computes the radiation conductivity for a specified frequency group when using the multifrequency radiation option.

RESIST - calculates plasma electrical resistivity.

SHIFT - shifts values of variables at (n+1) to variables at (n) at the end of a time step.

SIGMA - computes temperature-dependent electron-atom collision cross section.

TABLE2 -
TABLE3
TABLEO
interpolates in the equation of state tables using the pointers.

TEMPBC - computes the plasma temperature and radiation specific energy boundary conditions.

TIMING - computes a new time step and determines whether the calculation is over.

UNREAD - reads in the common blocks from unit 4 at the beginning of a restarted calculation.

VOLORO - computes external circuit parameters V_0, L, and R as needed for solution of current equation.

WBIN - writes binary output to unit 8 for postprocessing.

XTRAP - uses a quadratic extrapolation algorithm to predict the value of a variable half a time step into the future.

ZONER - computes the Lagrangian zoning automatically.

9. Input/Output Units and Storage Requirements

The ZPINCH code uses seven different I/O units. These units are listed below along with their specific function.

Unit #	Function
2	ZPINCH writes all common blocks to this unit at the end of a calculation to allow a restart.
3	ZPINCH reads the equation of state tables from this unit.
4	ZPINCH reads the common blocks from this unit at the beginning of a restart calculation.
5	ZPINCH reads the namelist input from this unit.
6	ZPINCH writes lineprinter output to this unit.
8	ZPINCH writes binary output to this unit for postprocessing into plots.
10 11	ZPINCH writes formatted output to these units for postprocessing.
12	ZPINCH reads circuit parameter data or the user-specified discharge current from this unit.

When adding a variable to the common blocks, the block length (set in INITIA) must be changed so that DUMP and UNREAD will write and read the correct number of words for a restart. Notice that the lengths are measured in single words for the 64 bit Cray computer. This should be changed to double words if double precision is used on another computer. All of the variables should be

changed to double precision if a 32 bit word length computer is used. This must be changed to single words if single precision is used. All of the variables should be changed to single precision if a 64 bit word length computer is used.

10. The Common Blocks

For many of the variables, the second to the last letter indicates whether the variable is at a zone center or zone boundary, and the last letter denotes the time level. The suffixes are:

1 -- zone boundary

2 -- zone center

A -- t^{n+1}

B -- $t^{n+1/2}$

C -- t^{n}

D -- $t^{n-1/2}$

The letter R will appear in a variable name if the quantity is associated with the radiation field, and N if the quantity is associated with the plasma. Thus TR2B(J) is the radiation temperature in the center of zone j at time $t^{n+1/2}$, and U1D(J) is the fluid velocity on the zone j boundary at time $t^{n-1/2}$. The variables are grouped in common blocks so that a subroutine will find most of the variables that it needs in fewer than all of the blocks. All of the variables in the common blocks are listed along with their meaning and units. A * superscript denotes mandatory input variables, and a ** superscript denotes a variable with a default value.

Common Blocks

COMMON/TIME/

1)	TA	t^{n+1}	times (s)
2)	TB	$t^{n+1/2}$	
3)	TC	t^{n}	
4)	TD	$t^{n-1/2}$	
5)	DTB**	$\Delta t^{n+1/2}$	
6)	DTC	$\Delta t^{n} = (\Delta t^{n+1/2} + \Delta t^{n-1/2})/2$	
7)	DTD	$\Delta t^{n-1/2}$	
8)	DTE	Δt^{n-1}	
9)	DTF	$\Delta t^{n-3/2}$	
10)	DTG	Δt^{n-2}	
11)	DTH	$\Delta t^{n-5/2}$	
12)	DTI	Δt^{n-3}	
13)	DT	$\Delta t^{n+3/2}$, the new time step	
14)	TMAX*	Total time for the simulation	
15)	DTMIN**	Minimum allowed time step	
16)	DTMAX**	Maximum allowed time step	
17)	SNTNTL**	Sentinel times	
18)	DTNEW**	New time steps corresponding to the sentinel times	

COMMON/TEMPER/

1)	TN2A	$(T_p)^{n+1}_{j-1/2}$	Plasma temperatures (eV)
2)	TN2B	$(T_p)^{n+1/2}_{j-1/2}$	

3) TN2C* $(T_P)^n_{j-1/2}$

4) TN1B $(T_P)^{n+1/2}_j$

5) TNSR2B $\sqrt{(T_P)^{n+1/2}_{j-1/2}}$ $(eV)^{1/2}$

6) TR2A $(T_R)^{n+1}_{j-1/2}$ Radiation temperatures (eV)

7) TR2B $(T_R)^{n+1/2}_{j-1/2}$

8) TR2C* $(T_R)^n_{j-1/2}$

9) TR1B $(T_R)^{n+1/2}_j$

10) TBC** temperature boundary condition (eV)

COMMON/CNTROL/

1) CON** real constants used in ZPINCH

2) TGROW** maximum percentage that Δt can increase in one cycle

3) TEDIT** time at which output freq. switches from IO(1) to IO(11) (s)

4) GEOFAC a geometry factor; 1, 2π, 4π

5) TSCC** Courant condition time step control

6) TSCV** $\Delta V/V$ time step control

7) R1 worst case for Courant condition

8) R2 worst case for $\Delta V/V$

9) R3N worst case for $\Delta T_P/T_P$

10-17) T1-T8 temporary vectors to be used for any purpose within a subroutine

18) TIO elapsed time controlled output values (s)

19) TOUT elapsed time accumulator for elapsed time output (s)

20) IDELTA** 1 = cartesian 2 = cylindrical 3 = spherical

21) IDELM1 0 = cartesian 1 = cylindrical 2 = spherical

22) NCYCLE time cycle index

23) NMAX* maximum number of time steps

24) JMAX* maximum number of spatial zones

25) JMAXM1 JMAX-1

26) JMAXP1 JMAX+1 used for indexing

27) JMAXP2 JMAX+2

28) ISW** control switches

29) ILUNIT output units for flux quantities

30) JCOUR zone # of Courant condition worst case

31) JSPVOL zone # of $\Delta V/V$ worst case

32) JNTEMP zone # of $\Delta T_p/T_p$ worst case

33) INDEX a vector used for output indexing

34) IZONE zone # of worst case of Courant, $\Delta V/V$, $\Delta T_p/T_p$

35) ITYPE 1 = Courant 2 = $\Delta V/V$ 3 = $\Delta E_R/E_R$ 4 = $\Delta T_p/T_p$
5 = $\Delta B/B$ worst restriction

36) IITYPE 0 = physical -1 = min Δt 1 = max Δt

37) IEDIT** intermediate output cycle frequencies

38) IIZONE zone # of worst case if the Δt is Δt_{max} or Δt_{min}

39) ICOND principal time step constraint

40) ICOND2 secondary time step constraint if primary is Δt_{min} or Δt_{max}

41) NVMAX time step of maximum compression

42) IUNIT cm^2, radian-cm, steradian for δ = 1, 2, 3

43) JVMAX zone # of maximum compression

44) TSCTN** $\Delta T_p/T_p$ time step control

45) IO* primary output frequency vector for cycle number based output

46) IOBIN** output frequency of binary output

47) RADIUS** radius of outermost zone

48) RI** the radius of the region to be given constant Δm's

49) RO** not used

50) RO2 not used

51) NI** the number of zones in the inner constant mass region

52) RATIO mass ratio between successive zones in transition region

53) NFG** the number of frequency groups

54) R3R worst case for $\Delta E_R/E_R$

55) TSCTR* $\Delta E_R/E_R$ time step control

56) JRTEMP zone # of $\Delta E_R/E_R$ worst case

57) R3B worst case for $\Delta B/B$

58) TSCB* $\Delta B/B$ time step control

59) JBTEMP zone # of $\Delta B/B$ worst case

60) IQUIT an integer vector used for diagnosing the reason for termination of a run

COMMON/HYDROD/

1) U1D $u_j^{n-1/2}$ fluid velocity (cm/s)

2) U1B** $u_j^{n+1/2}$

3) DR2B $\Delta r_{j-1/2}^{n+1/2}$ zone widths (cm)

4) DR2A $\Delta r_{j-1/2}^{n+1}$

5) R1C r_j^n radius (cm)

6)	R1B	$r_j^{n+1/2}$	
7)	R1A	r_j^{n+1}	
8)	RS1C	$(r_j^n)^{\delta-1}$	
9)	RS1B	$(r_j^{n+1/2})^{\delta-1}$	
10)	RS1A	$(r_j^{n+1})^{\delta-1}$	
11)	PR2C	$(P_R)_{j-1/2}^n$	radiation pressure (J/cm^3)
12)	PR2B	$(P_R)_{j-1/2}^{n+1/2}$	
13)	PR2A	$(P_R)_{j-1/2}^{n+1}$	
14)	PN2C	$(P_p)_{j-1/2}^n$	gas pressure (J/cm^3)
15)	PN2B	$(P_p)_{j-1/2}^{n+1/2}$	
16)	PN2A	$(P_p)_{j-1/2}^{n+1}$	
17)	P2C	$P_{j-1/2}^n$	total pressure (J/cm^3)
18)	P2A	$P_{j-1/2}^{n+1}$	
19)	V2C	$V_{j-1/2}^n$	specific volume (cm^3/g)
20)	V2B	$V_{j-1/2}^{n+1/2}$	
21)	V2A	$V_{j-1/2}^{n+1}$	
22)	V0		initial specific volume

23)	COMPR		VO/V compression	
24)	VDOT2B	$\dot{V}^{n+1/2}_{j-1/2}$	time derivative of sp. volume (cm^3/g-s)	
25)	DMASS2	$\Delta m_{o_{j-1/2}}$	Lagrangian mass	δ=1 g/cm^2 δ=2 g/cm-radian δ=3 g/steradian
26)	DMASS1	$\Delta m_{o_j} = (\Delta m_{o_{j-1/2}} + \Delta m_{o_{j+1/2}})/2$		
27)	Q2B	$q^{n+1/2}_{j-1/2}$	artificial viscosity (J/cm^3)	
28)	VMAX		max compression	
29)	TAVMAX		time of max compression (s)	
30)	VOL2B	$V^{n+1/2}_{j-1/2}$	zone volume (cm^3)	
31)	VOL2A	$V^{n+1/2}_{j-1/2}$		
32)	PBC**		pressure boundary condition	

COMMON/ESCOM/

1)	ER2C	$E^{n}_{R_{j-1/2}}$	radiation energy density (J/cm^3)
2)	ENT2B	$(C_v)^{n+1/2}_{j-1/2}$	plasma specific heat (J/eV-g)
3)	ER2B	$E^{n+1/2}_{R_{j-1/2}}$	radiation energy density (J/cm^3)
4)	PNT2B	$(P_P)_T{}^{n+1/2}_{j-1/2}$	temperature derivative of gas pressure (J/cm^3-eV)
5)	ER2A	$(E_R)^{n+1}_{j-1/2}$	radiation energy density (J/cm^3)

6)	EN2A	$(E_p)^{n+1}_{j-1/2}$	plasma specific internal energy (J/g)
7)	DE2A	$(n_e)^{n+1}_{j-1/2}$	electron number density ($1/cm^3$)
8)	DN2A	$(n_p)^{n+1}_{j-1/2}$	ion number density
9)	DE2B**	$(n_e)^{n+1/2}_{j-1/2}$	electron number density
10)	DN2B*	$(n_p)^{n+1/2}_{j-1/2}$	ion number density
11)	ATW2B*	$A^{n+1/2}_{j-1/2}$	average ion atomic weight (amu)
12)	ZT2B	$\partial Z/\partial T^{n+1/2}_{j-1/2}$	temperature derivative of average charge (esu/eV)
13)	Z2B**	$Z^{n+1/2}_{j-1/2}$	average charge (esu)
14)	ZSQ2B	$(Z^{n+1/2}_{j-1/2})^2$	average squared charge $(esu)^2$
15)	VBC**		specific volume boundary condition (cm^3/g)
16)	AD	}	coefficients defining the grid for the equations of state
17)	AT		
18)	BD		
19)	BT		
20)	EBC	radiation energy density boundary condition	
21)	TN2AL	$\log (T^{n+1}_{P_{j-1/2}})$	
22)	TR2AL	$\log (T^{n+1}_{R_{j-1/2}})$	
23)	DN2AL	$\log (n^{n+1}_{P_{j-1/2}})$	

24) KEOS
25) LEOS
26) MEOS

} vectors used for indexing into the equation of state tables

27) EPSLON — a parameter that indicates how far out of equilibrium the radiation energy density is

28) SGMA1B — $\sigma_{ea_j}^{n+1/2}$, electron-atom collision cross section

29) RAD — 1/AD

30) RAT — 1/AT

31) RBD — 1/BD

32) RBT — 1/BT

33) XKEOS
34) XLEOS
35) XMEOS

vectors for indexing into the equation of state tables

COMMON/ESCOM1/

1) ZTAB — plasma charge state table

2) ENTAB — plasma specific internal energy table

3) RMFTAB — Planck opacity table

4) ROSTAB — Rosseland opacity table

5) HEADER — character description of EOS tables

COMMON/COEFF/

1) ROSS2B — $(\sigma_R)_{j-1/2}^{n+1/2}$ — Rosseland opacity (cm^2/g)

2) KANM1B — $(\kappa_p^-)_j^{n+1/2}$ — plasma thermal conductivity (J/cm-eV-s)

3) KANP1B — $(\kappa_p^+)_j^{n+1/2}$

4) KARM1B — $(\kappa_R^-)_j^{n+1/2}$ — radiation thermal conductivity ($cm^2 2/s$)

5)	KARP1B	$(\kappa_R^+)_j^{n+1/2}$	
6)	OMP2B	$(\omega_p)_{j-1/2}^{n+1/2}$	plasma emission coefficient (J/eV-g-s)
7)	OMR2B	$(\omega_R)_{j-1/2}^{n+1/2}$	plasma absorption coefficient (cm^3/s-g)
8)	RMFP2B	$(\sigma_p)_{j-1/2}^{n+1/2}$	Planck opacity (cm^2/g)
9)	RMFT2B	$(\sigma_p)_{j-1/2}^{n+1/2}$	Planck opacity for $T_p = T_R$ (cm^2/g)
10)	SND2B	$(S)_{j-1/2}^{n+1/2}$	arbitrary source to the plasma (J/g s)
11)	SHOK2B		shock heating (J/g/s)
12)	LAMN2B	$(\ell n\Lambda_{ei})_{j-1/2}^{n+1/2}$	Spitzer log Λ
13)	FLIM1B		radiation flux limit (J/cm^2 s)
14)	RFLU1B		diffusion flux (J/cm^2 s)

COMMON/COEFF1/

1)	BET12B	$(\beta_1)_{j-1/2}^{n+1/2}$	
			Beta Vector
2)	BET22B	$(\beta_2)_{j-1/2}^{n+1/2}$	
3)	AL112B	$(\alpha_{11})_{j-1/2}^{n+1/2}$	
			Diagonal Elements of Alpha Matrix
4)	AL222B	$(\alpha_{22})_{j-1/2}^{n+1/2}$	

5) OM112B $(\omega_{11})_{j-1/2}^{n+1/2}$

Diagonal Elements of Omega Matrix

6) OM222B $(\omega_{22})_{j-1/2}^{n+1/2}$

7) GM112B $(\gamma_{11})_{j-1/2}^{n+1/2}$

Diagonal Elements of Gamma Matrix

8) GM222B $(\gamma_{22})_{j-1/2}^{n+1/2}$

9) AA111B $(a_{11})_{j}^{n+1/2}$

Diagonal Elements of "a" Matrix

10) AA221B $(a_{22})_{j}^{n+1/2}$

11) OM122B $(\omega_{12})_{j-1/2}^{n+1/2}$

Off Diagonal Elements of Omega Matrix

12) OM212B $(\omega_{21})_{j-1/2}^{n+1/2}$

COMMON/COEFF2/

1) E11 (E_{11})

2) E12 (E_{12})

All Elements of the "E" Matrix

3) E21 (E_{21})

4) E22 (E_{22})

5) F1 (F_1)

Both Components of the "F" Vector

6) F2 (F_2)

7)	B11	(B_{11})	
8)	B12	(B_{12})	All Elements of the "B" Matrix
9)	B21	(B_{21})	
10)	B22	(B_{22})	
11)	D1	(D_1)	Both Elements of the "D" Vector
12)	D2	(D_2)	

COMMON/ECKCOM/

1)	T1A	$(T)_j^{n+1}$	kinetic energy of fluid (J/x)
2)	GGGE2A	$(G_e)_{j-1/2}^{n+1}$	radiation-plasma work (J/x)
3)	HHHR2B	$(H_R)_{j-1/2}^{n+1/2}$	radiation source (J/x)
4)	HHHN2B	$(H_P)_{j-1/2}^{n+1/2}$	plasma source (J/x)
5)	EEEC2A	$(E_c)_{j-1/2}^{n+1}$	radiation-plasma energy exchange (J/x)
6)	EEEERO	E_{R_o}	total initial radiation internal energy (J/x)
7)	EEEENO	E_{P_o}	total initial plasma internal energy (J/x)
8)	EEEEER	$(E_R)^{n+1}$	total radiation internal energy (J/x)
9)	EEEEEN	$(E_P)^{n+1}$	total plasma internal energy (J/x)
10)	TTTTTT	$(T)^{n+1}$	total fluid kinetic energy (J/x)

11)	HHHHHR	$(H_R)^{n+1}$	total radiation source (J/x)
12)	HHHHHN	$(H_P)^{n+1}$	total plasma source (J/x)
13)	EEEEEC	$(E_c)^{n+1}$	total radiation-plasma energy exchanged (J/x)
14)	GGGGGE	$(G_e)^{n+1}$	total radiation-plasma work (J/x)
15)	WWWWWR	$(W_R)^{n+1}$	total work done on radiation (J/x)
16)	WWWWWN	$(W_P)^{n+1}$	total work done on plasma (J/x)
17)	FFFFFR	$(F_R)^{n+1}$	total radiation heat lost across outer boundary (J/x)
18)	FFFFFN	$(F_P)^{n+1}$	total plasma heat lost across outer boundary (J/x)
19)	WWWWR	$(W_R)^{n+1}$	total work done on radiation on last cycle (J/x)
20)	WWWWN	$(W_P)^{n+1}$	total work done on plasma on last cycle (J/x)
21)	FFFFR	$(f_R)^{n+1}$	total radiation lost at outer bd. on last cycle (J/x)
22)	FFFFN	$(f_P)^{n+1}$	total plasma energy lost at outer bd. on last cycle (J/x)
23)	HHHHR	$(h_R)^{n+1}$	total radiation source on last cycle (J/x)
24)	HHHHN	$(h_P)^{n+1}$	total plasma source on last cycle (J/x)
25)	EEEEC	$(e_c)^{n+1}$	total radiation-plasma heat exchange on last cycle (J/x)
26)	GGGGE	$(g_e)^{n+1}$	total work to maintain one fluid on last cycle (J/x)
27)	ENLHS	left side of plasma energy balance equation (J/x)	

28) ETLHS left side of total energy balance equation (J/x)

29) ERRHS right side of radiation energy balance equation (J/x)

30) ENRHS right side of plasma energy balance equation (J/x)

31) ETRHS right side of total energy balance equation (J/x)

32) TTTTNO initial kinetic energy (J/x)

where δ=1 x = cm^2

δ=2 x = cm-radian

δ=3 x = steradian

COMMON/MAGSTR/

1) BR2A $(rB)^{n+1/2}_{j-1/2}$ (Gauss-cm)

2) B1A magnetic field B at j, n+1 (Gauss)

3) EF1B electric field in fluid's rest frame at j, n+1/2 (statvolts/cm)

4) B2C magnetic field at j-1/2, n (Gauss)

5) BR2C $(rB)^{n}_{j-1/2}$, (Gauss-cm)

6) B2A magnetic field at j-1/2, n+1 (Gauss)

7) EL1B electric field in lab rest frame at j, n+1/2 (statvolts/cm)

8) CDN1B net current density at j, n+1/2 (statamps/cm^2)

9) DBR1A $\Delta(rB)^{n+1}_{j}$ (Gauss-cm)

10) B1B magnetic field at j, n+1/2 (Gauss)

11) BRSQ2C $(r^2B^2/8\pi)^{n}_{j-1/2}$ (erg,cm)

12) BR2B $(rB)_{j-1/2}^{n+1/2}$ (Gauss-cm)

13) DBR1C $\Delta(rB)_j^n$ (Gauss-cm)

14) CDN2B current density at j-1/2, n+1/2 (statamps/cm^2)

15) PSI2B Joule heating term at j, n+1/2 (ergs/gm-radian-s)

16) B2B magnetic field at j-1/2, n+1/2 (Gauss)

COMMON/COEFF3/

1) A33
2) B33
3) C33
4) D33
5) E33
6) F33

A, B, C, D, E, and F coefficients used in solution of magnetic diffusion equation

7) AL332B $\alpha_{j-1/2}^{n+1/2}$
8) BET32B $\beta_{j-1/2}^{n+1/2}$
9) AA331B $a_j^{n+1/2}$

α, β, and a coefficients used in solution of magnetic diffusion equation

10) ETAM1B $-\eta_j^{n+1/2}$
11) ETAP1B $+\eta_j^{n+1/2}$

electrical resistivity (s)

12) GMBM2B $\gamma_{j-1/2}^{n+1/2}$ ion beam source term in magnetic diffusion equation

COMMON/MCKCOM/

1) FNEW F^{n+1} influx-conservation check (statvolts)

2) WW1A W_B^{n+1} in energy check (Joule/cm-radian)

3) FOLD — F^n in flux-conservation check (statvolts)

4) RECRNT — relative error in current conservation (%)

5) WW1C — W_{B1}^n in energy check (Joule/cm-radian)

6) VCHNL — channel voltage drop (volts)

7) EMF — addition term to update F^n to F^{n+1}

8) JJJADD — addition term to update J^n to J^{n+1}

9) WW1ADD — addition term to update W_{B1}^n to W_{B1}^{n+1}

10) VVVADD — addition term to update V^n to V^{n+1}

11) REFLUX — relative error in flux conservation (%)

12) JJJA — J^{n+1} in energy conservation (Joule/cm-radian)

13) WW2A — W_{B2}^{n+1} in energy conservation (Joule/cm-radian)

14) VVVA — S^{n+1} in energy conservation (Poynting vector term - Joule/cm-radian)

15) EMRHS — right hand side of magnetic energy equation

16) JJJC — J^n in energy conservation

17) WW2C — W_{B2}^n in energy conservation

18) VVVC — S^n in energy conservation

19) EMLHS — left hand side of magnetic energy equation

20) WW2ADD — addition term to update W_{2B}^n to W_{2B}^{n+1}

21) UUUO — energy in magnetic field at start of problem (Joule/cm-radian)

22) WW3A — W_{B3}^{n+1} in energy conservation check (Joule/cm-radian)

23) WW3C — W_{BC}^n in energy conservation check (Joule/cm-radian)

24) UUUA — energy in magnetic field at timestep n + 1 (Joule/cm-radian)

25) BMESRC — energy source to magnetic field due to fields induced by the ion beam (Joule/cm-radian)

26) WW3ADD — addition term to update W_{B3}^n to W_{B3}^{n+1}

27) LSINKO rate of energy loss to external inductor at start of problem (Watts)

28) LSINKI rate of energy loss to external inductor at time $t^{n+1/2}$ (Watts)

29) LSINKC total energy stored in external inductor at time t^{n+1} (Joules)

30) RSINKI rate of energy loss to external resistor at time $t^{n+1/2}$ (Watts)

31) RSINKC total energy dissipated through external resistor at time t^{n+1} (Joules)

32) ESINKI rate of energy going into electric field part at the channel voltage drop (Watts)

33) ESINKC total energy invested in electric field part of the channel voltage drop (Joules)

34) BSINKI rate of energy going into magnetic part of the channel voltage drop at time $t^{n+1/2}$ (Watts)

35) BSINKC total energy invested in magnetic part of channel voltage drop at time t^{n+1} (Joules)

36) VSRCI rate of energy supplied by external voltage source at time $t^{n+1/2}$ (Watts)

37) VSRCC total energy supplied by external voltage source at time t^{n+1} (Joules)

38) CSRC0 initial energy supplied by the capacitor (Joules)

39) CSRCI rate of energy supplied by capacitor at time $t^{n+1/2}$ (Watts)

40) CSRCC total energy supplied by capacitor at time t^{n+1} (Joules)

41) SMSNKI total energy going into inductor, resistor, and channel voltage drop at time $t^{n+1/2}$ (Watts)

42) SMSNKC total energy invested in inductor, resistor, and channel voltage drop at time t^{n+1} (Joules)

43) SMSRCI rate of energy supplied by external voltage source and capacitor at time $t^{n+1/2}$ (Watts)

44) SMSRCC total energy supplied by external voltage source and capacitor at time t^{n+1} (Joules)

COMMON/CIRCUT/

1)	IA	discharge current at time t^{n+1} (Amps)
2)	IB	discharge current at time $t^{n+1/2}$ (Amps)
3)	IC**	discharge current at time t^{n} (Amps)
4)	IDOTA	dI/dt at t^{n+1} (Amps/s)
5)	IDOTB	dI/dt at $t^{n+1/2}$ (Amps/s)
6)	IDOTC	dI/dt at t^{n} (Amps/s)
7)	VAPPLA	voltage from arbitrary source at t^{n+1} (volts)
8)	VAPPLB	voltage from arbitrary souce at $t^{n+1/2}$ (volts)
9)	VAPPLC	voltage from arbitrary source at t^{n} (volts)
10-12)	VODOT(A,B,C)	rate of change of voltage from arbitrary source at times t^{n+1}, $t^{n+1/2}$, t^{n} (volts/s)
13-15)	LEXT(A,B,C)	inductance in LRC circuit at times t^{n+1}, $t^{n+1/2}$, t^{n} (volts/s)
16-18)	LDOT(A,B,C)	rate of change of inductance in LRC circuit at times t^{n+1}, $t^{n+1/2}$, t^{n} (Henrys/s)
19-21)	LDOT2(A,B,C)	2nd time derivative of inductance in LRC circuit at times t^{n+1}, $t^{n+1/2}$, t^{n} (Henrys/s^2)
22-24)	REXT(A,B,C)	resistance in LRC circuit at time t^{n+1}, $t^{n+1/2}$, t^{n} (Ohms)
25-27)	RDOT(A,B,C)	rate of change of resistance inLRC circuit at times t^{n+1}, $t^{n+1/2}$, t^{n} (Ohms/s)
28)	CAP	capacitance in LRC circuit (Farads)
29)	CHARGE	charge on capacitor in LRC circuit at time $t^{n+1/2}$ (Coulombs)
30)	QZERO**	charge on capacitor in LRC circuit at initialization time (Coulombs)
31-33)	PL(A,B,C)	discharge inductance at times t^{n+1}, $t^{n+1/2}$, t^{n} (Henrys)

34-36)	EDOT(A,B,C)	rate of change of axial electric field evaluated at the channel axis at times t^{n+1}, $t^{n+1/2}$, t^{n} (volts/cm-s)
37-38)	EOUT(E,G)	not used
39-40)	FLUX(A,B,C)	magnetic flux per unit length in the discharge at times t^{n+1}, $t^{n+1/2}$, t^{n} (Gauss/cm)
41-42)	INET(B,D)	net current in the channel at times $t^{n+1/2}$ and $t^{n-1/2}$ (statamperes)
43-45)	PLDT(A,B,C)	rate of change of discharge inductance at times t^{n+1}, $t^{n+1/2}$, t^{n} (Henrys/s)
46)	FLUXE	not used
47-50)	EAXIS(A,B,C,E)	electric field on the channel's axis evaluated at times t^{n+1}, $t^{n+1/2}$, t^{n}, t^{n-1} (volts/cm)
51)	CP**	parameters for circuit elements (see Table 8b)
52)	ARG	used for calculating the arguments in the exponential functions used in thevariable circuit element expressions
53)	MSSGE	used for transmitting a message from the table containing the user-defined discharge current
54)	LTABLE	number of records in the table containing the user-defined discharge circuit
55)	CPARAM	an array used for storing the user-defined current table
56)	NTIMES	number of entries in CPARAM table

COMMON/MFRAD/

1)	ERFD2A	frequency dependent radiation specific energy at t^{n+1} (J/g)
2)	ERFD2C	frequency dependent radiation specific energy at t^{n} (J/g)
3)	SRFD2B	frequency dependent radiation emission term at $t^{n+1/2}$ (J/g s)
4)	SR2B	frequency dependent Rosseland opacity (cm^{-1})
5)	SP2B	frequency dependent Planck opacity (cm^{-1})

6) SER2B — frequency integrated radiation absorption (J/g s)

7) SRE2B — frequency integrated radiation emission term (J/g s)

8) HNU1 — boundaries of frequency groups (keV)

9) HNU2 — centers of frequency groups (keV)

10) RFDOUT — frequency dependent radiation energy flux at first wall on a given time cycle (J)

11) RFDINT — time integrated frequency dependent radiation energy flux at first wall up through a given time cycle (J)

COMMON/IBEAM/

1) JBMIB — $J_{BEAM,j}^{n+1/2}$ ion beam current density (A/cm^2)

2) SHAPE — shape factor for ion beam current density

3) BP** — parameters used for defining ion beam current density profile

4) DBM2B — $n_{BEAM_{j-1/2}}^{n+1/2}$ number density of ion beam (cm^{-3})

5) ELC2B — specific energy loss rate coefficient of beam ions ($eV\text{-}s^{-1}/eV$)

6) BMEL2B — rate at which beam loses energy to the plasma due to collisions with plasma electrons (J/s-cm-rad)

7) ELOS2B — rate at which beam loses energy to the plasma due to collisions with plasma electrons (J/s-cm-rad)

8) BMTIME** — the times that specify the beam current density time-dependence (s)

9) BMMASS** — mass of the beam ions divided by the proton mass

10) EBEAM** — energy of the beam ions (eV)

11) IPLAS — I_p plasma conduction current (A)

12) ZBEAM** — charge of beam ions in units of the proton charge

13) IBEAM** — I_{BEAM} total current of the total ion beam at maximum intensity (A)

14) BMDUR — duration of beam pulse (s)

15) VBEAM — velocity of beam ions (cm/s)

16) NORM — normalization factor used with shape function to calculate ion beam current density profile

11. Input

The problem to be solved by the ZPINCH code is defined by the initial conditions, material properties, driving circuit parameters, and for an ion beam problem, the beam parameters. This information, along with control parameter values, is input in two ways: namelist input and attached files. This section describes how all the necessary information is input.

11.1 Initial Conditions and Control Parameters

The ZPINCH code reads a namelist called INPUT that is assigned to I/O unit 5 in order to obtain the initial conditions and control parameters. Variables that must appear in this namelist are given in Table 1. Table 2 gives variables with default values; these variables need not appear in the namelist unless values different than the default values are desired. Table 3 gives the variables that must appear in the namelist if the automatic zoning option is taken (ISW(4) = 1). Table 4 lists the integer switches used to control the code. Table 5 lists the real constants used by the code that may be changed by input. Table 6 gives the intermediate output vector that gives the user the power to have any internally computed quantity output during execution. In Tables 1-7, real scalar variables are denoted with (R), real vector variables with (RV), integer scalar variables with (I), and integer vector variables with (IV).

11.2 Ion Beam Parameters

The ZPINCH code may be used to simulate the propagation of an ion beam in a discharge channel. The beam is defined by a number of input parameters and does not change as a result of its interaction with the discharge plasma. The beam makes its appearance in the MHD equations as a source of magnetic flux in

Table 1. Input Variables

Variable	Type	Default Value	Description
JMAX	(I)	---	Number of spatial zones $3 \leq JMAX \leq 52$
NMAX	(I)	---	Maximum number of time steps
TMAX	(R)	---	Maximum problem time (s)
IO	(IV)	---	Output cycle frequencies IO(1) -- hydrodynamics IO(2) -- energy IO(3) -- mfp's and # densities IO(4) -- short edit IO(5) -- multifrequency radiation IO(6) -- electromagnetics IO(11) -- same as IO(1)-(6) except after IO(12) time TEDIT (see TEDIT description) IO(13) IO(14) IO(15) IO(16)
DR2B	(RV)	---	Δr of each zone (cm) (DR2B is only input if automatic zoning is not used)
DN2B	(RV)	---	Plasma number density (cm^{-3})
TN2C	(RV)	---	Plasma temperature (eV)
TR2C	(RV)	---	Radiation temperature (eV)
ATW2B	(RV)	---	Atomic weight (amu)

Table 2. Optional Input Variables

Variable	Type	Default Value	Description
IDELTA	(I)	2	Geometry = 1 planar = 2 cylindrical = 3 spherical
DTB	(R)	10^{-12}	Initial time step (s)
DTMIN	(R)	10^{-1}*DTB	Minimum time step (s)
DTMAX	(R)	10^{-2}*TMAX	Maximum time step (s)
TSCC	(R)	5×10^{-2}	Time Step Controls - Courant
TSCV	(R)	5×10^{-2}	- $\Delta V/V$
TSCTR	(R)	1×10^{35}	- $\Delta E_R/E_R$
TSCTN	(R)	5×10^{-2}	- $\Delta T_p/T_p$
TSCB	(R)	5×10^{-2}	- $\Delta B/B$
TEDIT	(R)	-1	If TEDIT $\neq$ 0 then before time TEDIT IO(1)-(6) are used and after IEDIT IO(11)-(16) are used as output frequencies
IOBIN	(I)	-1	Binary output frequency written to unit 8 for postprocessing
TGROW	(R)	1.5	Time step is allowed to increase no more than TGROW*DTB on each successive cycle
TBC	(R)	2.5×10^{-2}	Temperature boundary condition (eV)
VBC	(R)	0.1	Specific volume boundary condition (cm^3/g)
PBC	(R)		Pressure boundary condition ($Joule/cm^3$)
U1B	(RV)	0	Initial velocity (cm/s)
IRS	(I)	0	Restart calculation flag = 0 Normal calculation = 1 Restarted calculation
ISW	(IV)	---	See Table 4 for definitions of these switches

Variable	Type	Default Value	Description
CON	(RV)	---	See Table 5 for the definitions of these numerical coefficients
IEDIT	(IV)	-1	See Table 6 for the definitions of these intermediate output frequencies
NFG	(I)	0	Number of frequency groups for a multi-frequency radiation calculation. If NFG=0 then the 2-T option is used. $0 \leq NFG \leq 20$.
B2C	(RV)	0	Initial magnetic field profile (Gauss)
QZERO	(R)	0	Initial charge on capacitor (esu)
CAP	(R)	0	Capacitance of external portion of circuit (farads)
TIO	(RV)	10^{-6}	Vector for elapsed time based output (s) TIO(1)-TIO(6) and TIO(11)-TIO(16) work analogous to IO(1)-IO(6) and IO(11)-IO(16)

Table 3. Input Variables for Automatic Zoning

Variable	Type	Default Value	Description
NI	(I)	---	Number of zones in the inner, constant mass region
RI	(R)	---	The radius of the inner region of constant mass zones (cm)
RATIO	(R)	---	Ratio of masses of successive zones outside of constant mass region

Table 4. Control Switches

ISW	Description	
1	= 0*	$T_R \sim E_R^{1/4}$
	= 1	$T_R \sim$ dilute Planckian
2	= 10*	number of constant time steps used at the beginning of a calculation
3	not used	
4	= 0*	user specifies zoning with DR2B
	= 1	automatic zoning (see Table 3)
5	not used	
6	= 0*	hydrodynamic motion is computed
	= 1	no hydro motion -- allows a pure temperature diffusion simulation
7	= 0*	magnetic diffusion equation is solved
	= 1	no magnetic field -- allows a pure gas dynamics simulation
8	not used	
9	not used	
10	= 1*	frequency of time step calculation
11	not used	
12	not used	
13	= 20/NFG	number of subgroups to divide frequency groups into when doing the integration
14	not used	
15	not used	
16	= 0*	velocity of outermost zone is calculated from equation of motion
	= 1	velocity in outermost zone is fixed at zero
17	not used	
18	not used	

19	= 1*	current is obtained from current equation
	= 0	user specifies current
20	= 0*	channel voltage drop is not included in the circuit equation
	= 1	channel voltage drop is included
21	= 0*	no ion beam
	= 1	an ion beam is included in the simulation
22	= 0	collisional heating is not taken into account in the plasma energy equation
	= 1*	collisional heating is taken into account
23	= 0	Joule heating is not taken into account in the plasma energy equation
	= 1*	Joule heating is taken into account
24	= 25*	maximum number of iterations allowed in the iterative procedure used to calculate the initial value of dI/dt
25	= 1*	maximum number of iterations allowed in the iterative procedure used for calculating self-consistent current and discharge voltage
26	= 0*	formatted output is written to units #10 and 11 for post-processing
	= 1	binary output is written to unit #8 for post-processing

*Denotes the default value.

Table 5. Real Constants

CON	Default	Description
1	1.2175×10^{2}	plasma thermal conductivity
2	1×10^{10}	radiation thermal conductivity
3	0.1	the percentage by which the radiation can be out of equilibrium before the nonequilibrium mean free path is used in the absorption term
4	1×10^{-30}	small term to avoid zero divide in flux limited radiation conduction term AA221B
5	0	if non-zero then it is used as a constant value of log Λ. Normally log Λ is computed.
6	1.37×10^{-5}	$4\sigma/c$
7	4.11×10^{5}	radiation emission term
8	3×10^{10}	radiation absorption term
9	1.602×10^{-19}	plasma pressure
10	3×10^{10}	radiation flux limit
11	10^{-11}	first timestep used once the ion beam has arrived
12	1.602×10^{-19}	plasma pressure derivative
13	0.69315	geometric factor for magnetic flux between the maximum simulation radius and the return current path
14	2.403×10^{-19}	plasma specific heat
15	2.403×10^{-19}	plasma specific internal energy
16	1.37×10^{-5}	radiation specific internal energy
17	0.0	up-stream average parameter
18	1.0	ion shock heating term
19	10^{-8}	timestep in circuit parameter table
20		$1/8\pi$
21	1.414	artificial viscosity coefficient

22	3×10^{10}	multifrequency radiation absorption term
23	6.334918×10^{4}	multifrequency radiation emission term
24	10^{10}	multifrequency radiation conductivity term
25	3×10^{10}	multifrequency radiation flux limit
26	1×10^{-20}	minimum allowable multifrequency radiation specific energy
27	2.3873×10^{9}	$c/4\pi$
28	$3. \times 10^{10}$	speed of light (cm/s); not used
29	3.3333×10^{-11}	$1/c$
30	10.	maximum number of current solver calculation cycles per global time step
31	5.7988×10^{-15}	coefficient on η_F, the fully ionized component to electrical resistivity
32	3.1109×10^{-1}	coefficient on η_W, the weakly ionized component to electrical resistivity
33	10^{-2}	minimum ratio of magnetic pressure to plasma pressure for which $\Delta B/B$ is to be considered as a time step constraint
34	300.	channel length (cm)
35	π	
36	4.8×10^{-10}	magnitude of electron charge in esu
37	9.79×10^{5}	a factor used for converting beam ion energy into speed
38	5.4420×10^{-4}	ratio of electron mass to proton mass
39	10^{-2}	convergence limit for the iterative procedure used to calculate the initial value of dI/dt
40	10^{-2}	convergence limit for the iterative procedure used to calculate self-consistent values of the discharge current anddischarge voltage drop
41	0	K_6 in timing scheme

Table 6. Intermediate Output Switches

IEDIT	Subroutine	Variables
1	ABCDEF	A11, A22, B11, B12, B21, B22, C11, C22, D1, D2, E11, E12, E21, E22, F1, F2
2	MATRIX	AL112B, AL222B
3	MATRIX	OM112B, OM122B, OM212B, OM222B
4	MATRIX	GM112B, GM222B
5	MATRIX	AA111B, AA221B
6	MATRIX	BET12B, BET22B
10	OMEGA	OMR2B, OMP2B
11	KAPPA	KARM1B, KARP1B, KANM1B, KANP1B, LAMN2B, FLIM1B
14	HYDRO	U1B, R1A, R1B, DR2A, DR2B, RS1A, RS1B, V2A, V2B, VDOT2B
15	QUE	Q2B
16	TEMPBC	T1(1) → T1(9), TR2A (JMAXP1), TN2A (JMAXP1)
17	RADTR	ER2A, ERFD2A
19	NUMDEN	DN2B, DE2B, DN2A, DE2A
20	EMISSN	SRE2B, SRFD2B
21	ABCRAD	A22, B22, C22, D2, E22, F2
22	RADCOF	AL222B
23	RADCOF	GM222B
24	RADCOF	AA221B
25	RADCOF	BET22B
26	RCOND	KARM1B, KARP1B, FLIM1B
27	PLSCOF	AL112B
28	PLSCOF	GM112B
29	PLSCOF	AA111B

30	PLSCOF	BET12B
31	PCOND	KANM1B, KANP1B, LAMN2B
32	ABCPLS	A11, B11, C11, D1, E11, F1
33	RADCOF	OM222B
40	ABCMAG	A33, B33, C33, D33, E33, F33
41	MAGCOF	AA331B, AL332B, BET32B
42	CURRENT	all coefficients used in algorithm for solving current equation and quantities used for determining internal timing parameters
43	ECOND	SGMA1B, TN1B, LAMN2B, 22B, fully and weakly ionized components of resistivity, ETAM1B, ETA1B
44	MAGNET	R1A, ETAP1B, ETAM1B, B2B, EF1B
50	VPLSMA	VPD, VPC, VPB, VPDOTC, VPDOTB, VPDOTA, and various timesteps used
70	VOLORO	complete diagnostics of the determination of circuit parameters as obtained by linear interpolation between entries in the circuit parameter table.
71	VOLORO	
72	MAGBC	complete diagnostics of the determination of user-specified discharge current as obtained by linear interpolation between entries in the circuit parameter table
73	XTRAP	complete diagnostics of 2nd order extrapolation
77		if set to a positive number, the tablecontaining the user-defined discharge current is output during initialization
80	SCLOAD	intermediate quantities in iterative procedure used for calculating self-consistent values of the discharge current and voltage
81	IRK	quantities used in the Runge-Kutta algorithm used to obtainI(t)
82	LDIS	quantities used to calculate discharge inductance
83	IDOTRK	quantities used in Runge-Kutta algorithm used to obtain dI/dt

84	LDISDT	quantities used to calculate rate of change of discharge inductance

Table 7. Input Variables for Ion Beam

Variable	Type	Default Value	Description
BMTIME	RV	---	the four times specifying the ion beam time-dependence (seconds)
BMMASS	R	---	mass of beam ion (proton masses)
EBEAM	R	---	energy of beam ion (eV)
ZBEAM	R	---	charge of beam ion (proton charge)

the magnetic diffusion equation and as an energy source in the plasma energy equation.

The default option in ZPINCH is for no beam. To include an ion beam in the simulation, the user must set ISW(21) to 1. In addition, the user must input the ion species (ratio of ion mass to proton mass), the ionic charge (ratio of ion charge to proton charge), the ion energy (eV), and a vector containing four times that are used to specify the time-dependence of the beam current. All beam parameter input is done via namelist INPUT described in Section 11.1. Table 7 lists the beam parameters that must appear in the namelist if the ion beam option is taken (ISW(21) = 1).

The time-dependence of the beam current consists of an upward ramp from zero to maximum current, a period of constant current, and a downward ramp from maximum current to zero current. This time-dependence can be completely characterized by four parameters that the user has the freedom to choose:

BMTIME(1) -- time at which upward ramp starts (s)

BMTIME(2) -- time at which constant current period begins (s)

BMTIME(3) -- time at which constant current period ends and downward ramp starts (s)

BMTIME(4) -- time at which the downard ramp ends (s)

11.3 Circuit Parameters

The user has the option of specifying the discharge current as a function of time (ISW(19) = 0) or of allowing it to be calculated from the circuit equation (ISW(19) = 1). In both cases, the circuit parameters ($I_D(t)$ or R(t), L(t), $\dot{V}_{Appl}(t)$ and C(t)) are obtained by reading the NAMELIST file attached to unit 5.

For the first option, the user may specify the discharge current as a function of time. The form of this function is a table of times and values and is assumed to be piecewise linear. The user supplies the number of table entries to the variable NTIMES and the table entries to the array variqable CPARAM. Table 8a shows the form of the table in the array CPARAM.

The subroutine MAGBC uses linear interpolation to determine values of the discharge current between table times. This requires that the table entries in CPARAM be in chronological order, with the entry for time = zero in CPARAM(1,1). While the program is running, if the problem time becomes larger than the largest time in the table, then MAGBC will return a value for the discharge current equal to the value corresponding to the largest table time. To avoid possible problems, the user should be sure that the problem time never exceeds the largest table time.

For the second case, the user may specify the resistance, the inductance, the rate of change of the driving voltage, and the capacitance as functions of time. These functions all have the same form, given by the equation

$$f(t) = a_1 + a_2 \exp\left[-\left(\frac{t - a_4}{a_5}\right)^{a_8}\right] + a_3 \exp\left[-\left(\frac{t - a_6}{a_7}\right)^{a_9}\right]$$

The constants a_1, a_2, etc., for each function are read (via NAMELIST) into the vector variable CP. Table 8b shows the correspondence between the vector elements and these constants for each of the circuit parameters.

The subroutine VOLORO evaluates these functions for time t to determine the values of the circuit parameters. If the values of the circuit parameters are to remain constant with time, the user should be sure to set the values of

Table 8a. Contents of the Discharge Current Table

NTIMES = number of table entries ($2 \leq$ NTIMES ≤ 500)

CPARAM(1,1) = t0 (should be 0.0)	CPARAM(1,2) = discharge current at t0
(2,1) = t1 (time in seconds)	(2,2) = current in Amps at t1
•	•
•	•
•	•
(NTIMES,1) = last table time	(NTIMES,2) = last table current

Table 8b. Contents of Circuit Parameter Table

CP(1)	= a_1	for external resistance
(2)	= a_2	
(3)	= a_3	
(4)	= a_4	
(5)	= a_5	
(6)	= a_6	
(7)	= a_7	
(8)	= a_8	
CP(10)	= a_1	for external inductance
(11)	= a_2	
(12)	= a_3	
(13)	= a_4	
(14)	= a_5	
(15)	= a_6	
(16)	= a_7	
(17)	= a_8	
(18)	= a_9	
CP(19)	= a_1	for external applied voltage
(20)	= a_2	
(21)	= a_3	
(22)	= a_4	
(23)	= a_5	
(24)	= a_6	
(25)	= a_7	
(26)	= a_8	
(27)	= a_9	
CP(28)	= a_1	for external capacitance
(29)	= a_2	
(30)	= a_3	
(31)	= a_4	
(32)	= a_5	
(33)	= a_6	
(34)	= a_7	
(35)	= a_8	
(36)	= a_9	

a_2 and a_3 to zero and the value of a_1 to the value of the circuit parameter. The rest of the constants should always be non-zero.

Note that any value of the circuit parameter table time parameter τ may be used as long as CON(19) has the same value. Since the array CPARAM has a maximum dimension of 1000, the maximum problem time that can be used before the code attempts to access locations beyond CPARAM (1000,j) is 999 τ.

11.4 Input Data Files

In addition to the NAMELIST input described in Section 11, ZPINCH requires equation of state data to be read in through unit 3. Therefore, before one executes ZPINCH, one must attach a file to unit 3 that has equation of state data in the proper form.

In general, the equation of state data is dependent on the plasma species and the range of densities and temperatures one is considering. The standard way of creating this data is to use the computer code MIXERG [1]. The data file created by MIXERG for the examples quoted in this paper is included on the ZPINCH source tape. The files created by MIXERG are not in the proper form for use in ZPINCH. The PREP code for converting the output of MIXERG for use in ZPINCH is included on the source tape. To execute this code, one must attach the file created by MIXERG to unit 11 and create a NAMELIST input file titled INIT which contains the approximate average number density for the ZPINCH problem, DENAV. PREP writes to a file attached to unit 10 which contains the equation of state input for ZPINCH. The formats for file 11 and for file 10 are given in Tables 9 and 10. The first line of file 10 is a heading that one may change by modifying the source code for PREP. This heading is printed out during the ZPINCH run and is usually used to identify the plasma species for which the equation of state information has been calculated. File 10 must then be copied into file 3 for use by ZPINCH.

Table 9. Data File Entering PREP

Data Group #	Data Group Length (Records)	Description	Format
1	1	IMAX = # of densities JPMAX = # of plasma temperatures JRMAX = # of radiation temperatures NGRP = # of radiation energy groups	4I4
2	(NGRP+1)/4*	energy group boundaries (eV)	4D12.6
3	IMAX.JPMAX/4*	charge state of plasma $(e)^{+}$	4D12.6
4	IMAX.JPMAX/4*	energy density of plasma $(J/g)^{+}$	4D12.6
5	IMAX.JPMAX/4*	heat capacity of plasma $(J/g\text{-}eV)^{+}$	4D12.6
6	IMAX.JPMAX/4*	temperature derivative of charge state $(e/eV)^{+}$	4D12.6
7	IMAX.JPMAX.JRMAX/4*	Rosseland opacity (whole spectrum) $(cm^2/g)^{++}$	4D12.6
8	IMAX.JPMAX.JRMAX/4*	Planck opacity (whole spectrum) $(cm^2/g)^{++}$	4D12.6
	.... If there is only 1 frequency group, the file stops here		
9	IMAX.JPMAX.NGRP/4*	Rosseland group opacity $(cm^2/g)^{+++}$	4D12.6
10	IMAX.JPMAX.NGRP/4*	Planck group opacity $(cm^2/g)^{+++}$	4D12.6

* round up to the next integer

\+ the plasma temperature changes most frequently, the plasma density least frequently

++ the radiation temperature changes most frequently, then the plasma temperature and then the plasma density least frequently

+++ the energy group changes most frequently, then the plasma temperature then the plasma density least frequently

Table 10. Data File Leaving PREP

Data Group #	Data Group Length (Records)	Description	Format
1	1	header	' ', 10A4
2	1	$\log_{10}$ of lowest plasma density (cm^{-3}), $\log_{10}$ of ratio between consecutive plasma densities (cm^{-3}), $\log_{10}$ of lowest plasma temperature (eV), $\log_{10}$ of ratio between consecutive plasma temperatures (eV), # of frequency groups	4D12.6, I12
3	10 JPMAX/4*	plasma charge state $(e)^{+}$	4D12.6
4	10 JPMAX/4*	plasma energy density $(J/g)^{+}$	4D12.6
5	10 JPMAX.JRMAX/4*	Rosseland opacity (whole spectrum) $(cm^2/g)^{++}$	4D12.6
6	10 JPMAX.JRMAX/4*	Planck opacity (whole spectrum) $(cm^2/g)^{++}$	4D12.6
	.... If there is only 1 frequency group, the file ends here		
8	(NGRP+1)/4*	frequency group boundaries (eV)	4D12.6
9	10 JPMAX.NGRP/4*	Rosseland group opacity $(cm^2/g)^{++}$	4D12.6
10	10 JPMAX.NGRP/4*	Planck group opacity (cm^2/g)	4D12.6

* round up to the next integer

\+ the plasma temperature changes most frequently, the plasma density least frequently

++ the radiation temperature changes most frequently, then the plasma temperature and then the plasma density least frequently

+++ the energy group changes most frequently, then the plasma temperature then the plasma density least frequently

A sample runstream, which shows by example how to create file 3, has also been included on the ZPINCH source tape.

12. Options

This section briefly describes the most important options that may be taken by selecting the appropriate control parameters. The nature of these options is to disable or enable various portions of the code. This gives the user the freedom to investigate various physical effects independently of others.

The most simple mode in which to run ZPINCH is to turn off the equation of motion (ISW(6) = 1) and the magnetic diffusion equation (ISW(7) = 1) and solve only the coupled plasma temperature and radiation energy equations. One may then either choose the single group (NFG = 0) or multigroup method (NFG $\leq$ 20) for solving the radiation energy equation. If the single group option is chosen, the radiation temperature may either be modeled as $T_R \sim E_R^{1/4}$ (ISW(1) = 0) or as a dilute Planckian (ISW(1) = 1).

The next level of complexity is to turn on the equation of motion (ISW(6) = 0). If this option is taken, the user may either allow the outer zone to move without constraint (ISW(16) = 0) or to fix the position of the outermost zone (ISW(16) = 1). The latter simulates a discharge contained in a can, the former simulates a discharge that is in an effectively infinite medium.

The next mode of operation is to enable the magnetic diffusion equation (ISW(7) = 0). If this option is taken, the user must also specify how the discharge current, which is used in the boundary condition for the magnetic diffusion equation, is to be determined. The simplest option is to specify the discharge current as a function of time (ISW(19) = 0). Otherwise, the

user may choose to have the discharge current determined from the circuit equation (ISW(19) = 1). If this option is taken, the user must give the parameters defining the driving circuit as described in Section 11. In addition, the user may choose not to include the discharge voltage drop as a part of the circuit equation (ISW(20) = 0). This allows the user to assess how much the discharge modifies the behavior of the current that is driven by any particular circuit configuration.

Finally, the user may choose to include an ion beam in the simulation (ISW(21) = 1). If this option is taken, a number of parameters must be input that define the beam as described in Section 11. In order to assess the relative importance of Joule heating and of collisional heating, options have been provided that allow the user to turn off either of these terms (ISW(22) = 0 and ISW(23) = 0, respectively) in the plasma energy equation.

13. Sample Problem

A sample calculation was performed to demonstrate the code's performance. The simulation was of an argon channel with an initially uniform mass density of 2.3246×10^{-5} gm/cm^3. The initial temperature profile was Gaussian with 0.8 eV maximum and 0.5 cm HWHM. The discharge current was driven by a LRC circuit consisting of a 7 Ω resistor, a 0.5 μH inductor, and a 0.54 μF capacitor. The capacitor initially stored 40 kJ of energy. Figure 7 gives the NAMELIST input for this run.

Figures 8-19 show results of the calculation. Figure 8 shows the discharge current building up to a maximum of 37 kA at 1 μs. Figure 9 shows the rate of change of the current. Figure 10 shows the discharge inductance. The brief increase in dI/dt at about 0.4 μs is caused by the sharp dip in the discharge inductance. Figure 11 shows the voltages in the circuit. The sharp

drop in the discharge inductance at about 0.4 μs is caused by a combination of radiative transfer and magnetic diffusion effects. Figures 12 and 13 show radial profiles of the plasma temperature. Just prior to 0.4 μs, the temperature begins to increase rapidly in the outer initially cool region due to a radiative transfer of energy from the hot inner region. Figures 14 and 15 show radial profiles of the electrical resistivity. As the temperature profile spreads, the conducting area of the channel increases. The resistivity plots show a very rapid increase in the outer conducting radius around 0.4 μs. Figures 16 and 17 show the magnetic field profiles. From the resistivity profile at 486.46 ns, the peak magnetic field would be expected to occur at about 0.75 cm. However, the magnetic field plots show that at that time, the peak field is actually at about 0.4 cm. The finite magnetic diffusion timescale causes the magnetic field profile to lag behind the resistivity profile. The result is an excess amount of magnetic flux in the channel that decays away on the magnetic diffusion timescale. The resistive decay of this excess flux is the cause for the sharp drop in the discharge inductance. Figures 18 and 19 show the mass density profiles. The plots indicate the formation of a blast wave at early times. At later times, when the temperature profile has spread out, much of the driving force for the blast was has been sapped. The result is that the density peak formed at early times expands at the local sound speed.

Acknowledgement

This work was supported in part by Los Alamos National Laboratory, Los Alamos, NM, Sandia National Laboratory, Albuquerque, NM, Lawrence Livermore National Laboratory, Livermore, CA, and Kernforschungszentrum Karlsruhe, Karlsruhe, FRG.

References

[1] R.R. Peterson and G.A. Moses, "MIXERG - An Equation of State and Opacity Computer Code," Comp. Phys. Comm. 28, 405 (1983).

[2] G.A. Moses, R.R. Peterson, T.J. McCarville, "MF-FIRE - A Multifrequency Radiative Transfer Hydrodynamics Code," University of Wisconsin Fusion Engineering Program Report UWFDM-458 (Dec. 1982).

[3] J. Von Neumann and R. Richtmyer, "A Method for the Numerical Calculation of Hydrodynamic Shocks," J. Appl. Phys. 21, 232 (1950).

[4] D.L. Chapin, "Theoretical Studies of Exploding Wire and Laser Heated Pinch Plasmas," Ph.D. Thesis, University of Michigan (1974).

[5] S.V. Dresvin (ed.), Physics and Technology of Low Temperature Plasmas, Iowa State University Press, Ames, IA (1977).

[6] W.M. Manheimer, "Energy Input into a Gas Enclosed Z-Pinch," Phys. of Fluids 17, 1767 (Sept. 1974).

[7] R. Kidder and W. Barnes, "WAZER - A One-Dimensional, Two-Temperature Hydrodynamic Code," UCRL-50583, Lawrence Radiation Laboratory, Livermore, California.

[8] R.D. Richtmyer and K.W. Morton, Difference Methods for Initial Value Problems, Interscience Publishers, New York, (1967) 200.

[9] L. Spitzer, Physics of Fully Ionized Gases, Second Edition, Interscience Publishers, New York, (1962) 144.

[10] R. Kidder, D. Steinberg, and A. Cecil, "A One-Dimensional Magnetohydrodynamics Code," UCRL-14931, Lawrence Radiation Laboratory, CA.

[11] Y.B. Zel'dovich and Y.P. Raizer, Physics of Shock Waves and Other High Temperature Hydrodynamic Phenomena, W.D. Hayes and P.F. Probstein, eds., Academic Press, New York, (1966) Vol. 1, Chapt. 2.

[12] T.J. McCarville, G.A. Moses, G.L. Kulcinski, and I.O. Bohachevsky, "Improvement of the Two Temperature Radiative Transfer Model," Nucl. Tech./Fusion 5, 5 (1984).

[13] G.C. Pomraning, The Equations of Radiation Hydrodynamics, Pergamon Press, New York, (1973).

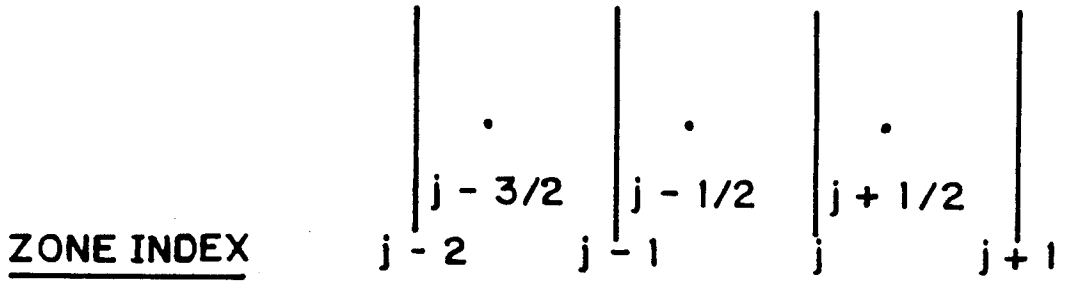


Figure 1 Index system used to denote spatial boundaries.

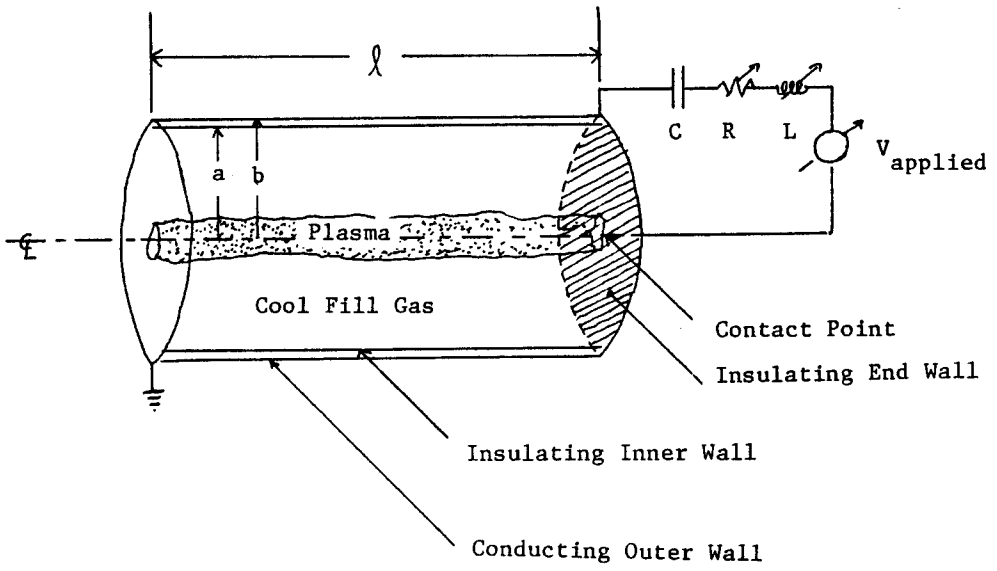


Figure 2 External circuit used to drive channel current.

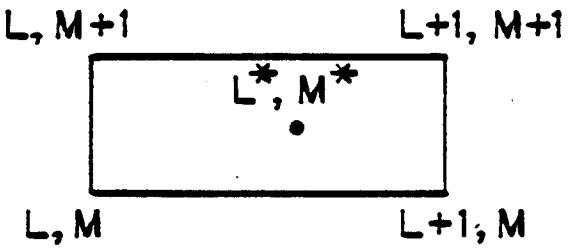


Figure 3 Indices used to interpolate in a two-dimensional grid.

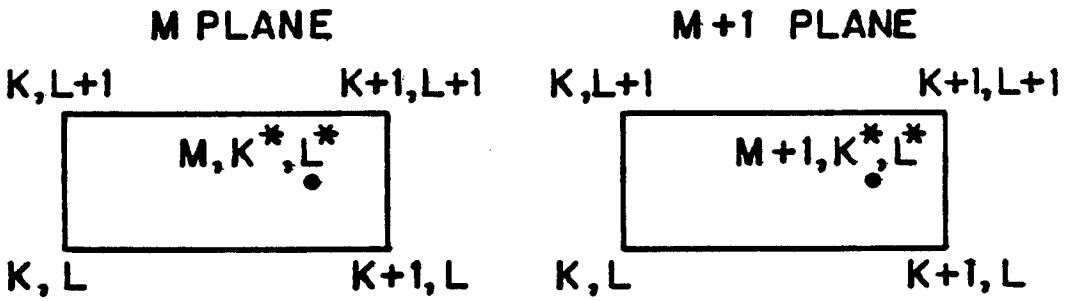
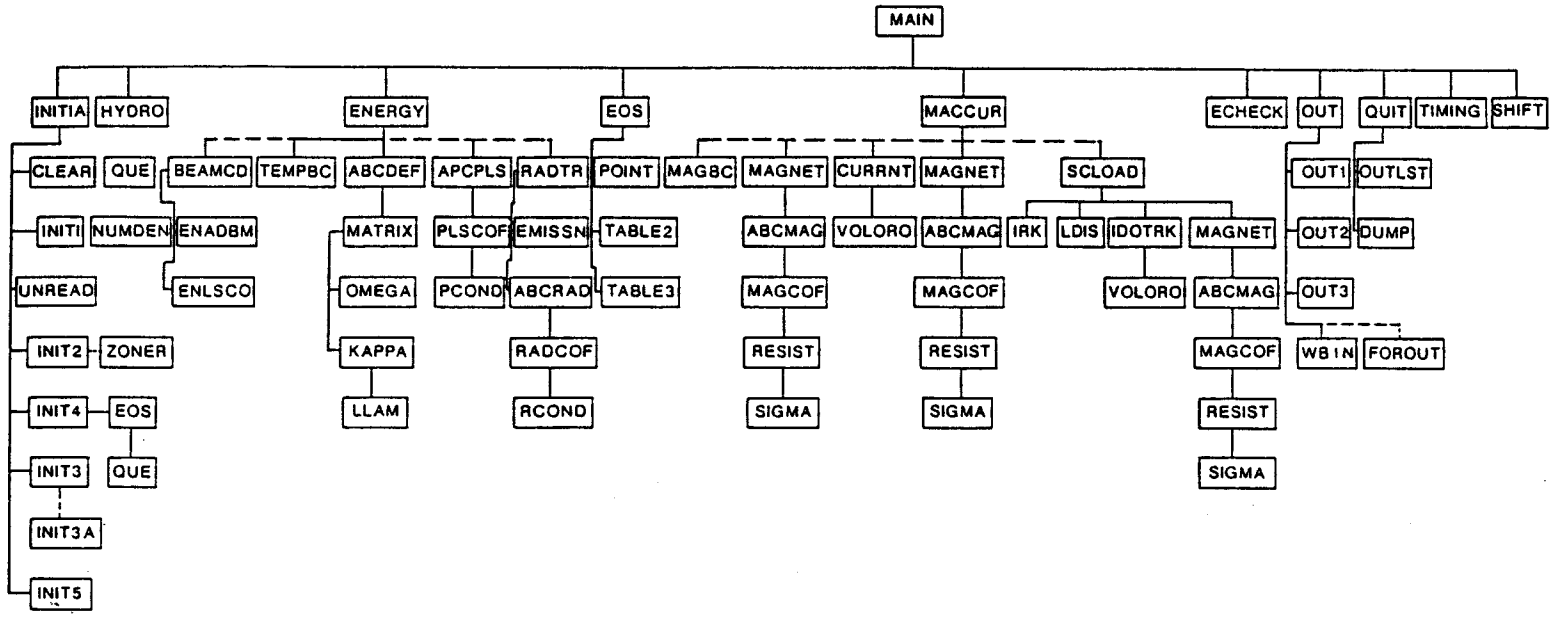


Figure 4 Indices used to interpolate in a three-dimensional grid.



ZPINCH STRUCTURE DIAGRAM

Figure 5 Structure chart of the Z-PINCH code.
Dotted lines indicate conditional routes.

```
      IMPLICIT REAL*4 (A-H,O-Z)
      DIMENSION CPARAM(1000), MSSGE(10)
C
      READ(5,90) (MSSGE(I), I = 1, 10)
   90 FORMAT(1X, 10A4)
C
      TIMRES  =  1.0E-08
C
C        WRITE THEM ONTO  UNIT 12
C
      WRITE(12,1000) (MSSGE(I), I = 1, 10)
 1000 FORMAT(1X,10A4)
C
      WRITE(12,2000) TIMRES
 2000 FORMAT(1X,1PE12.4)
C
C        NOW ASSIGN CIRCUIT PARAMETERS
C
      DO 102 I = 1, 1000
         T = TIMRES * (I - 1)  +  0.6E-06
         CPARAM(I) = T * (1.359E11) / EXP(T/1.E-06)
  102 CONTINUE
C
C        NOW WRITE ARRAY ONTO UNIT 12
C
      DO 300 I = 1,1000
         WRITE(12,3000)  CPARAM(I)
  300 CONTINUE
 3000 FORMAT(1X, 1PE12.4)
      STOP
      END
```

Figure 6. Listing of sample program to generate external circuit table.

```
&input
 isw(13) = 10,
iedit(80) =200, iedit(81) = 200, iedit(82) = 200, iedit(83) = 200,
iedit(84) = 200,
tgrow = 1.20d0,  tscb = 1.0d-02,
con(40) = 1.D-05, isw(25) = 10,
nmax =5000, nfg = 0, dtb = 2.0d-09, dtmin = 1.0d-09, dtmax = 5.0d-09,
tmax = 2.d-06,isw(2) =100,iobin =10,tedit = 2.0d-08,
io(1) =  1,io(2) =  1,io(3) =  1,io(4) = -1, io(5) =-1, io(6) =  1,io(7) =  1,
tio(1) =5.0D-08, tio(2) =5.0D-08, tio(3) =5.0D-08, tio(4) = -1.0d0,
tio(5) = -1.0d0, tio(6) =5.0D-08, tio(7) =5.0D-08,
io(11)=-1,io(12)=-1,io(13)=-1,io(14)=-1,io(15)=-1,io(16)=-1,io(17)=-1,
tio(11) =1.0D-07, tio(12) =1.0D-07, tio(13) =1.0D-07, tio(14) = -1.0d0,
tio(15) =-2.5d-07, tio(16) =1.0D-07, tio(17) =1.0D-07,
jmax = 50, atw2b = 50 * 40.0d0, pbc = 3.7792d-03, tbc = 0.07d0,
isw(21) = 0, isw(20) = 1, isw(19) = 1,   isw(24) = 25, con(39) = 1.0d-03,
con(11) = 10.0d0,    con(30) = 5,  con(41) = 1.0d-05,
isw(4) = 1, ni = 10, ri = 0.20d0, ratio = 1.19d0,
```

Fig. 7. Input file for sample problem (continued on next page)

```
tn2c = 7.85d-01, 7.71d-01, 7.57d-01, 7.43d-01, 7.30d-01,
       7.16d-01, 7.03d-01, 6.90d-01, 6.78d-01, 6.65d-01,
       6.51d-01, 6.36d-01, 6.18d-01, 5.99d-01, 5.77d-01,
       5.53d-01, 5.26d-01, 4.98d-01, 4.66d-01, 4.33d-01,
       3.97d-01, 3.60d-01, 3.21d-01, 2.82d-01, 2.43d-01,
       2.04d-01, 1.68d-01, 1.33d-01,1.03d-01, 7.60d-02, 20 * 0.07d0,
dn2b = 50 * 3.50d17,
tr2c = 7.85d-01, 7.71d-01, 7.57d-01, 7.43d-01, 7.30d-01,
       7.16d-01, 7.03d-01, 6.90d-01, 6.78d-01, 6.65d-01,
       6.51d-01, 6.36d-01, 6.18d-01, 5.99d-01, 5.77d-01,
       5.53d-01, 5.26d-01, 4.98d-01, 4.66d-01, 4.33d-01,
       3.97d-01, 3.d0d-01, 3.21d-01, 2.82d-01, 2.43d-01,
       2.04d-01, 1.68d-01, 1.33d-01,1.03d-01, 7.60d-02, 20 * 0.07d0,
  cp = 7.0d0, 0.d0, 0.d0, 0.d0, 1.d0, 0.d0, 1.d0, 1.d0, 1.d0,
       5.0D-07, 0.0D0, 0.d0, 0.0D0, 1.0D0, 0.d0, 1.d0, 50.d0, 1.d0,
       0.d01, 0.d0, 0.d0, 0.d0, 1.d0, 0.d0, 1.d0, 1.d0, 1.d0,
       0.54d-06, 0.d0, 0.d0, 0.d0, 0.d0,
  qzero = -0.207846097d0,
  &end
```

Fig. 7. Input file for sample problem (cont.)

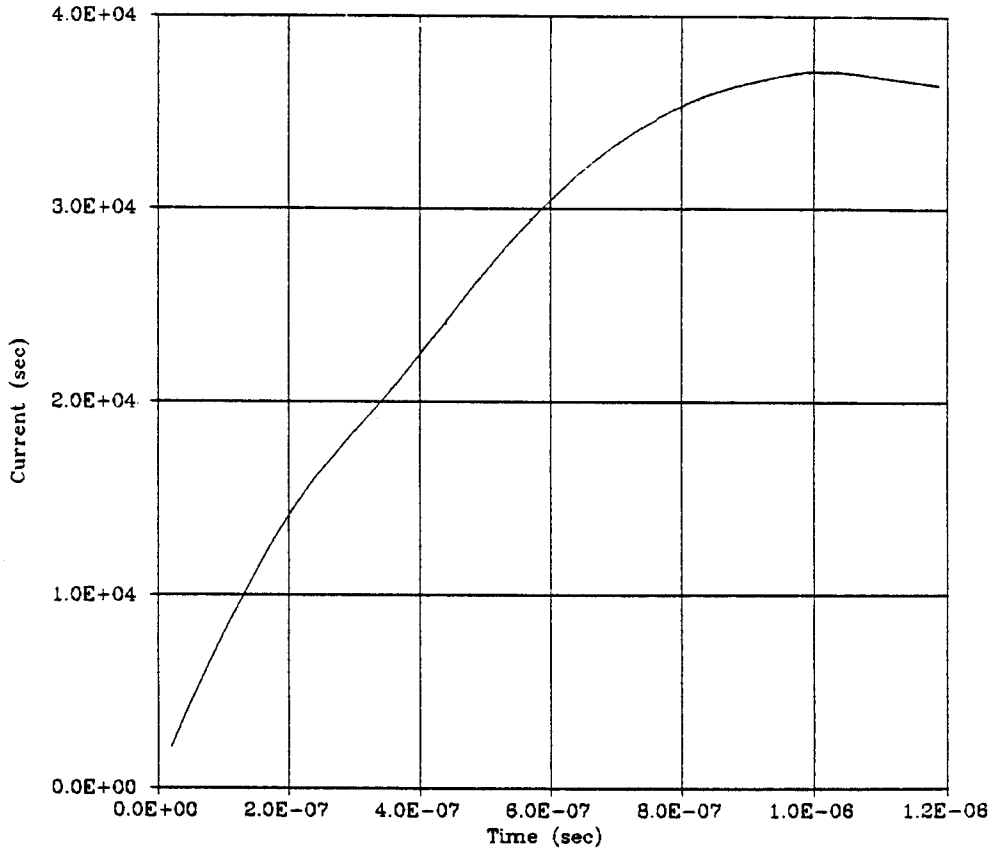


Figure 8. Discharge current for sample problem.

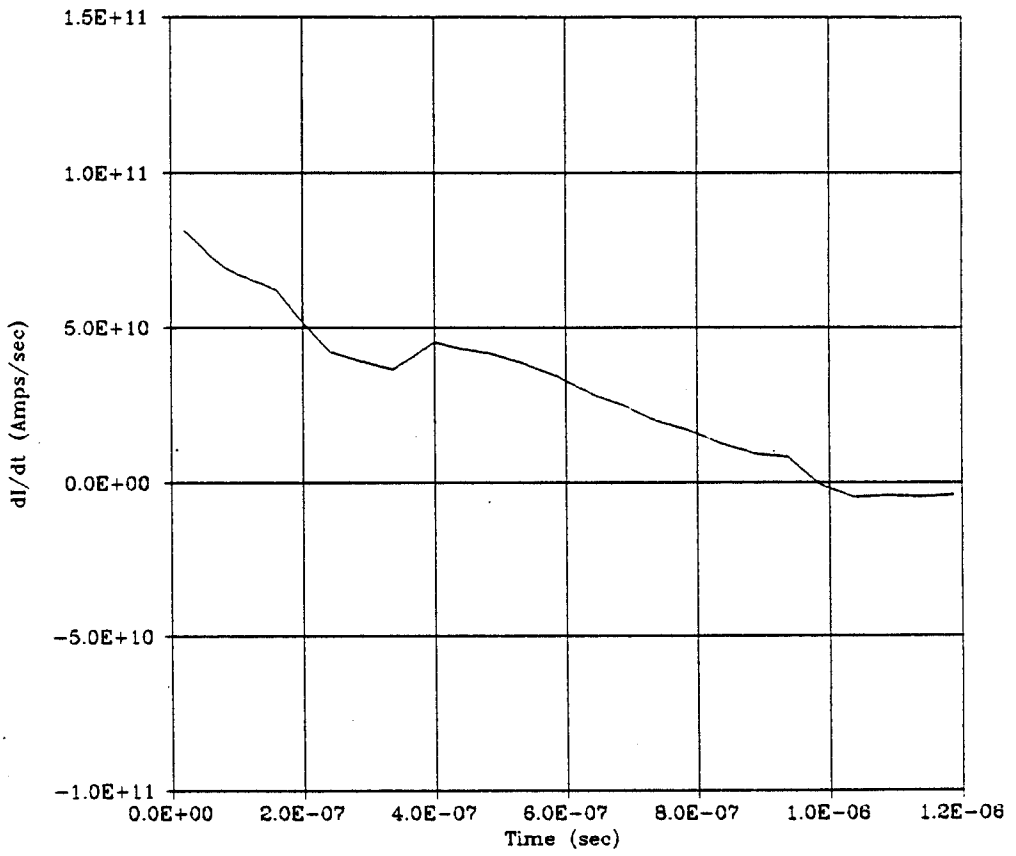


Figure 9. Rate of change of discharge current for sample problem.

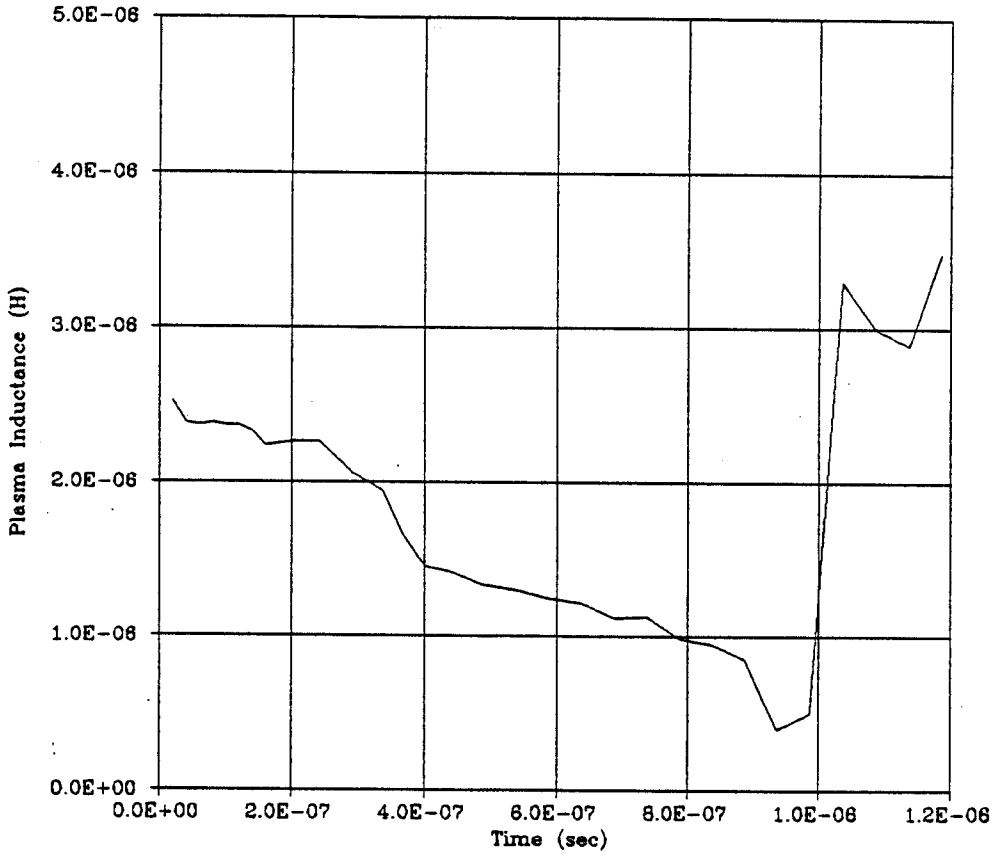


Figure 10. Discharge inductance for sample problem.

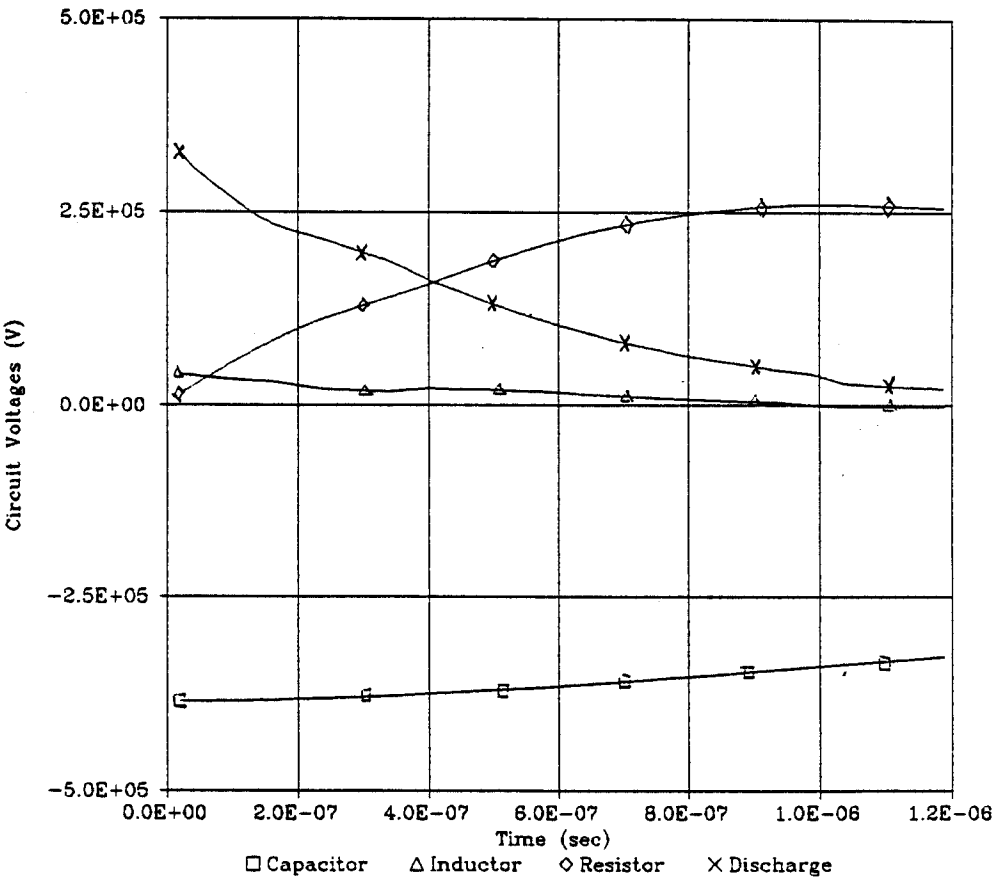


Figure 11. Circuit voltages for sample problem.

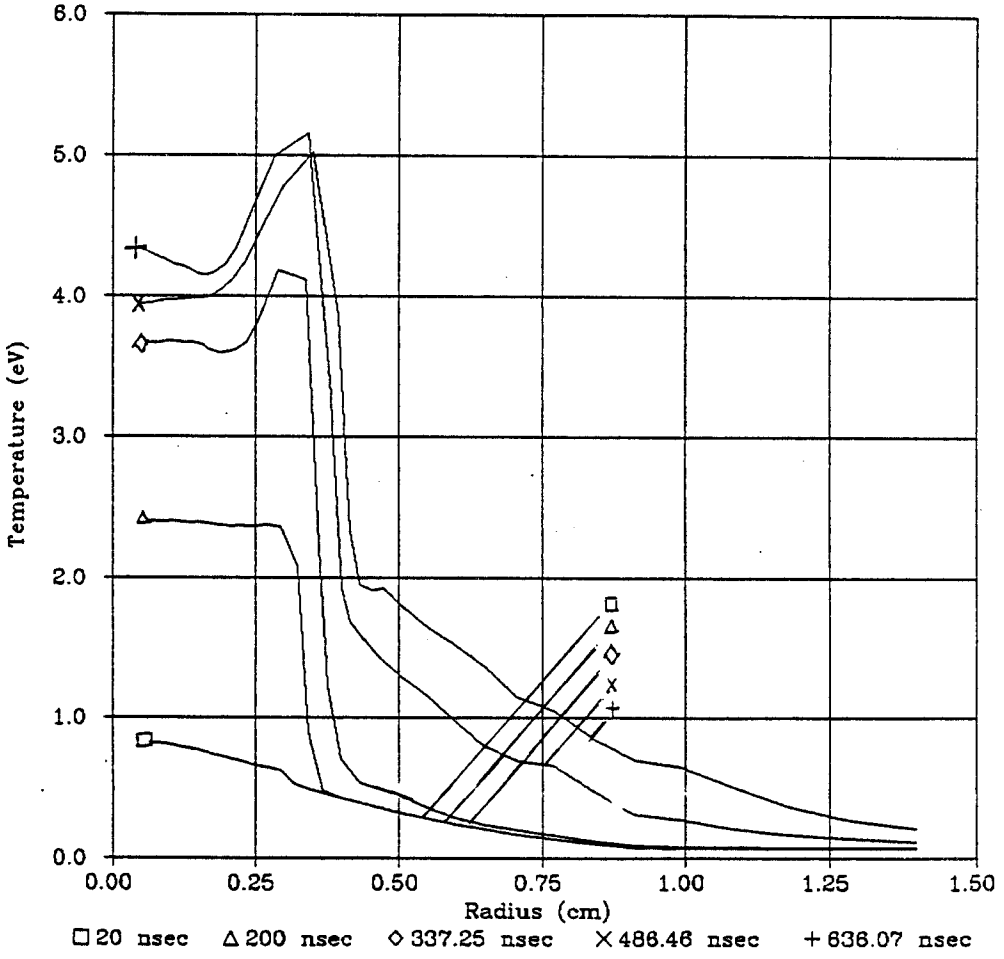


Figure 12. Plasma temperature profiles for sample problem.

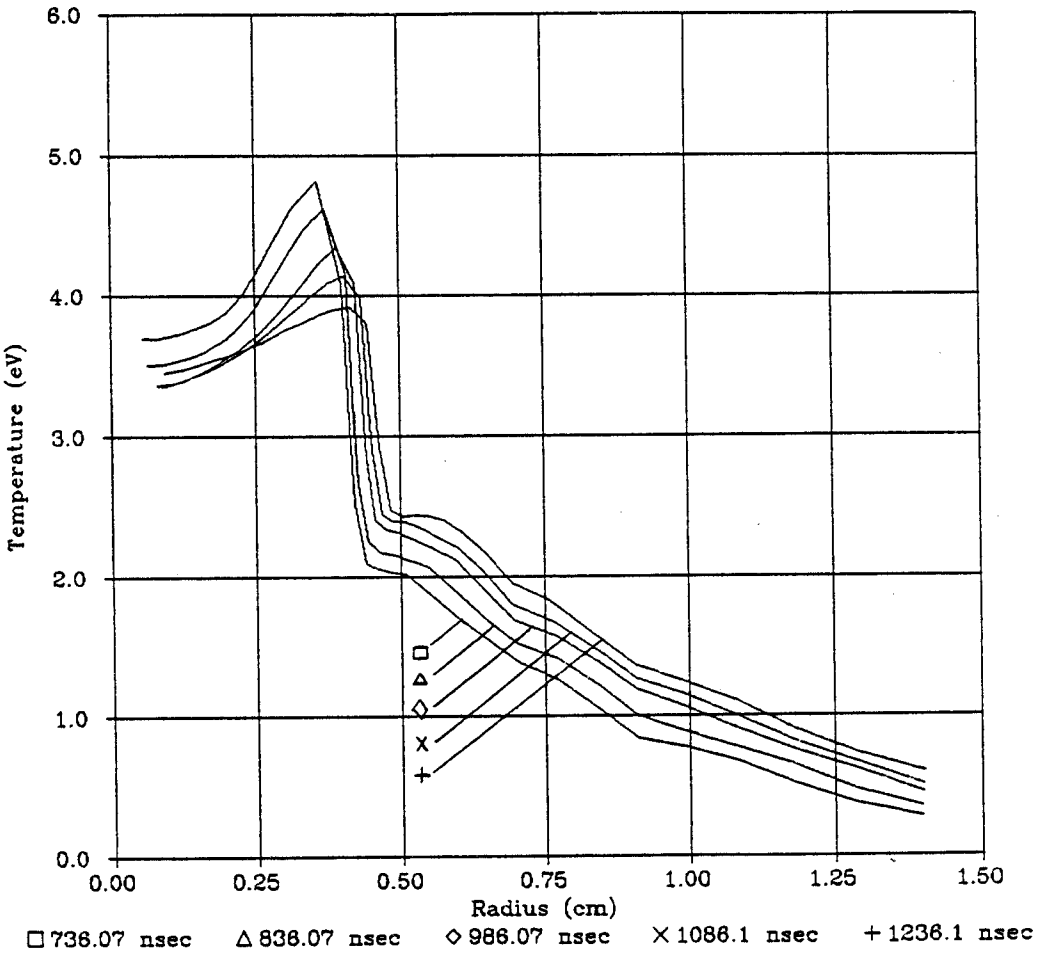


Figure 13. Plasma temperature profiles for sample problem.

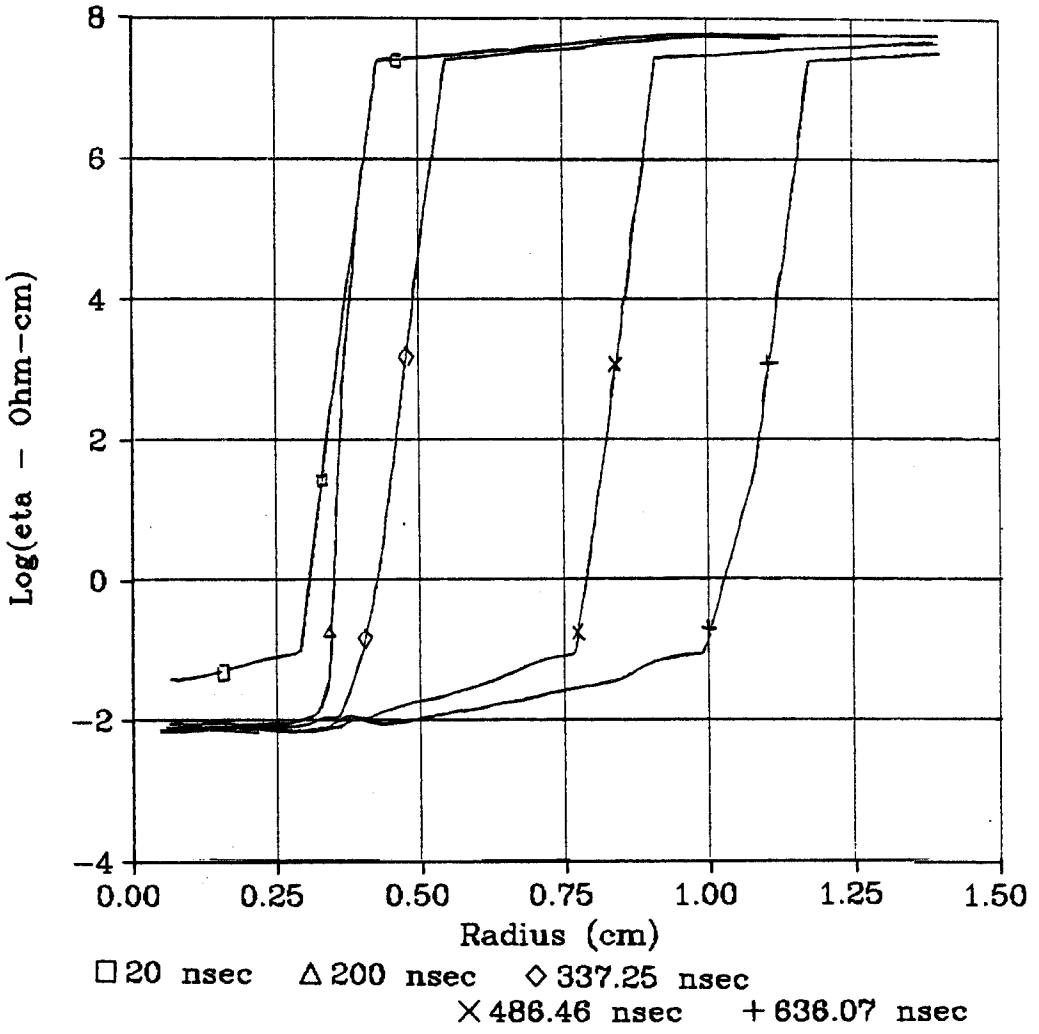


Figure 14. Electrical resistivity profiles for sample problem.

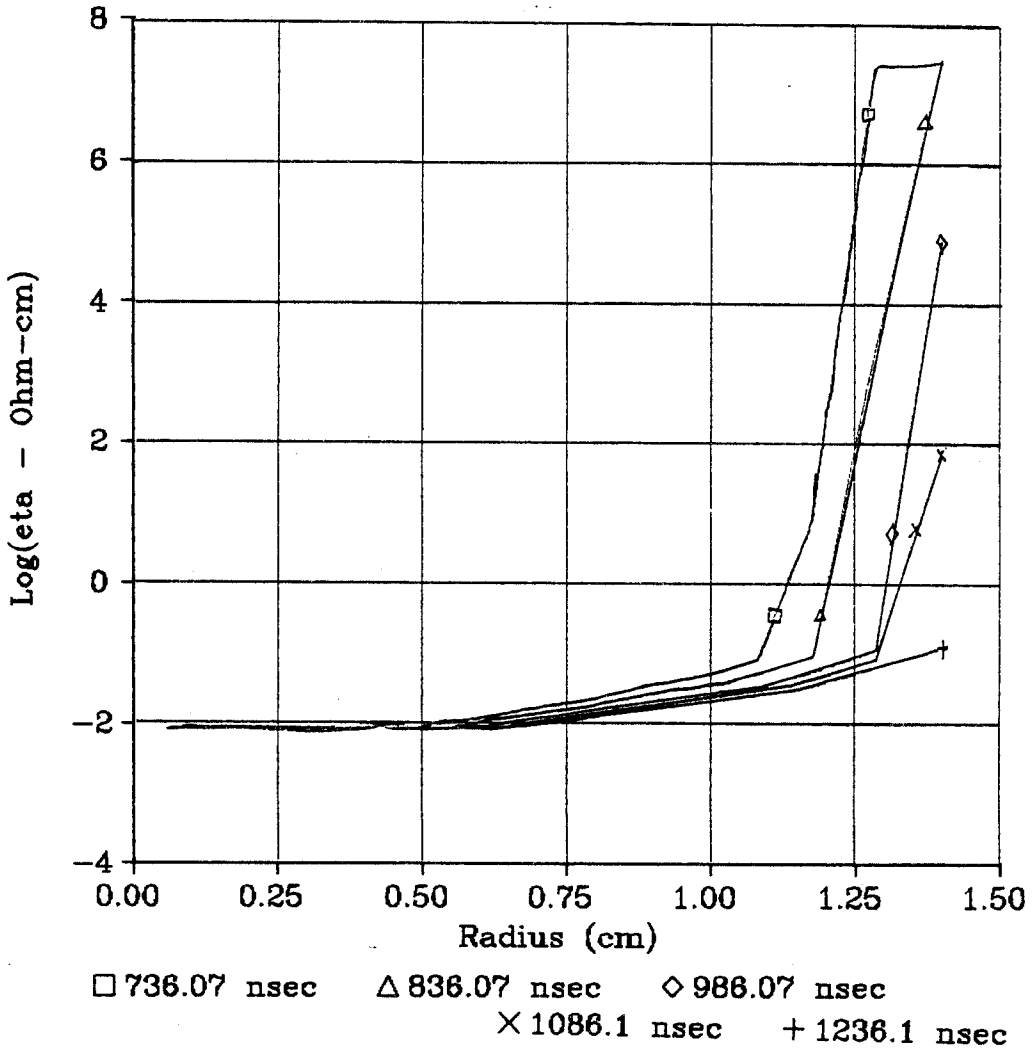


Figure 15. Electrical resistivity profiles for sample problem.

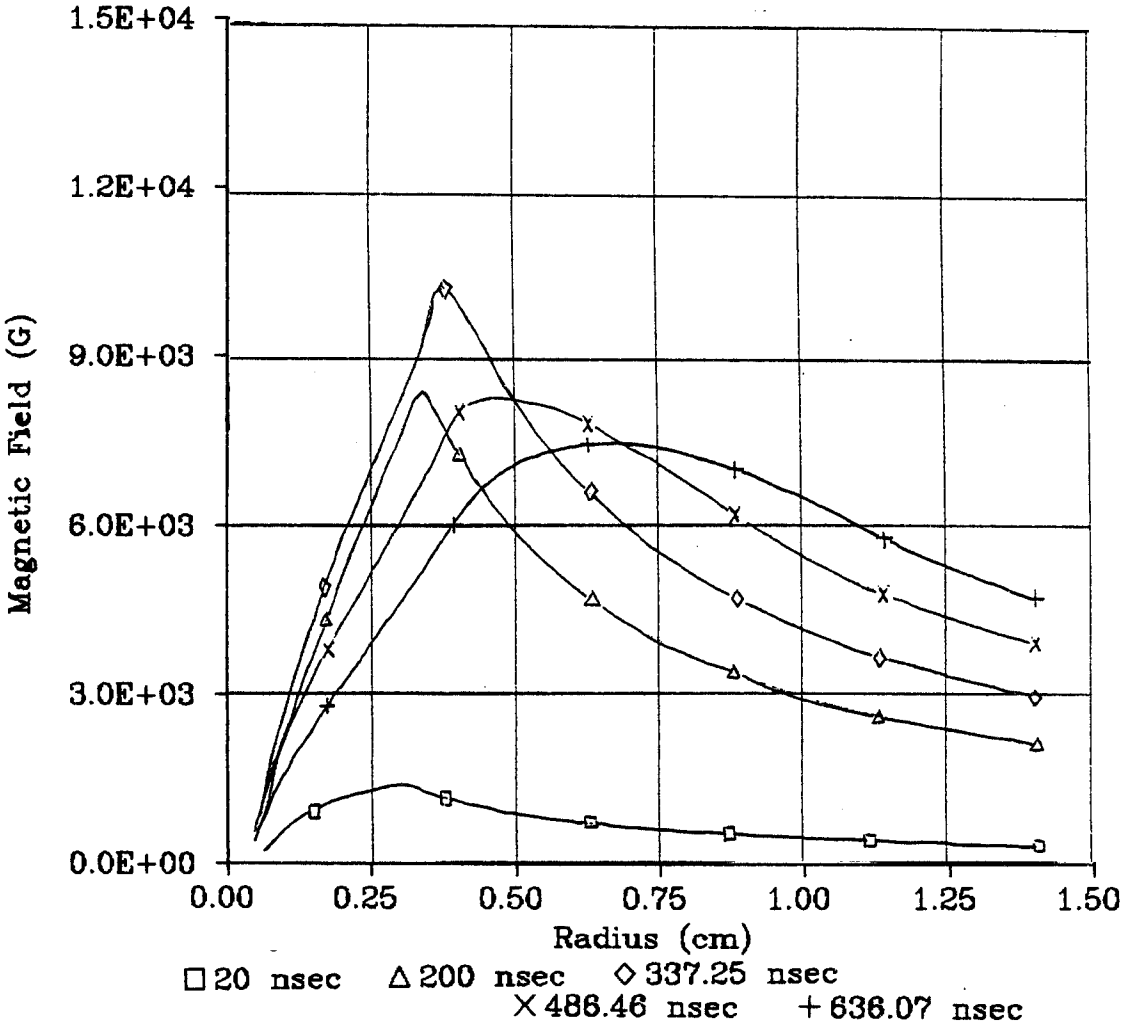


Figure 16. Magnetic field profiles for sample problem.

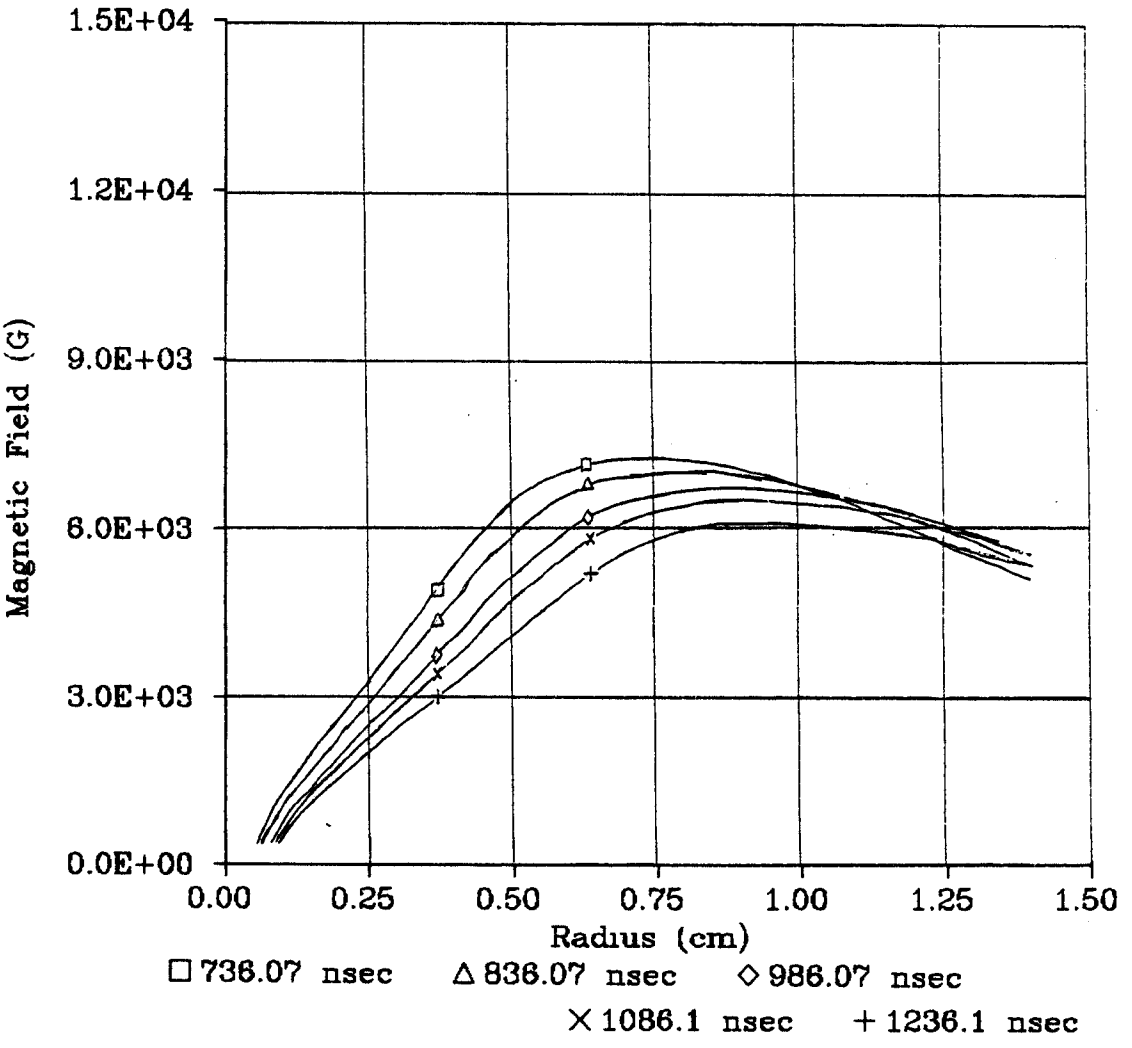


Figure 17. Magnetic field profiles for sample problem.

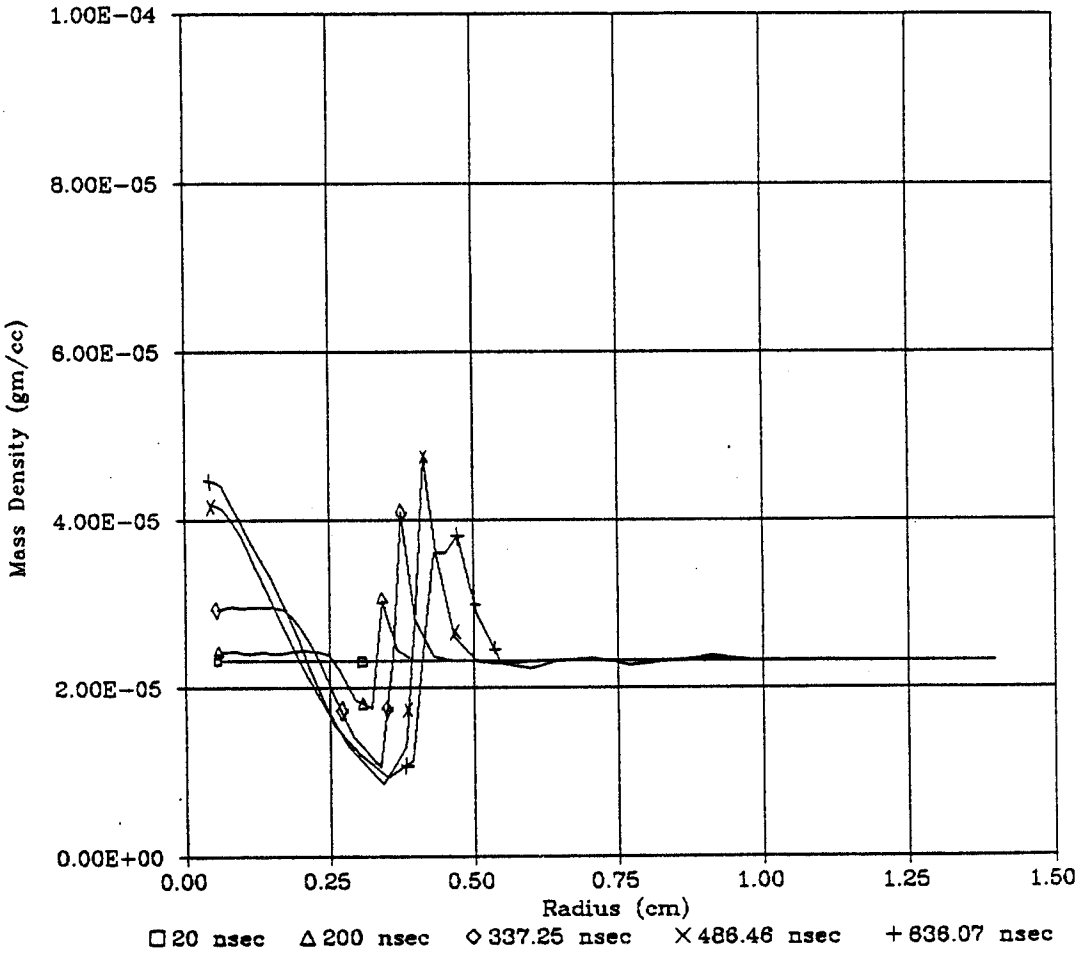


Figure 18. Mass density profiles for sample problem.

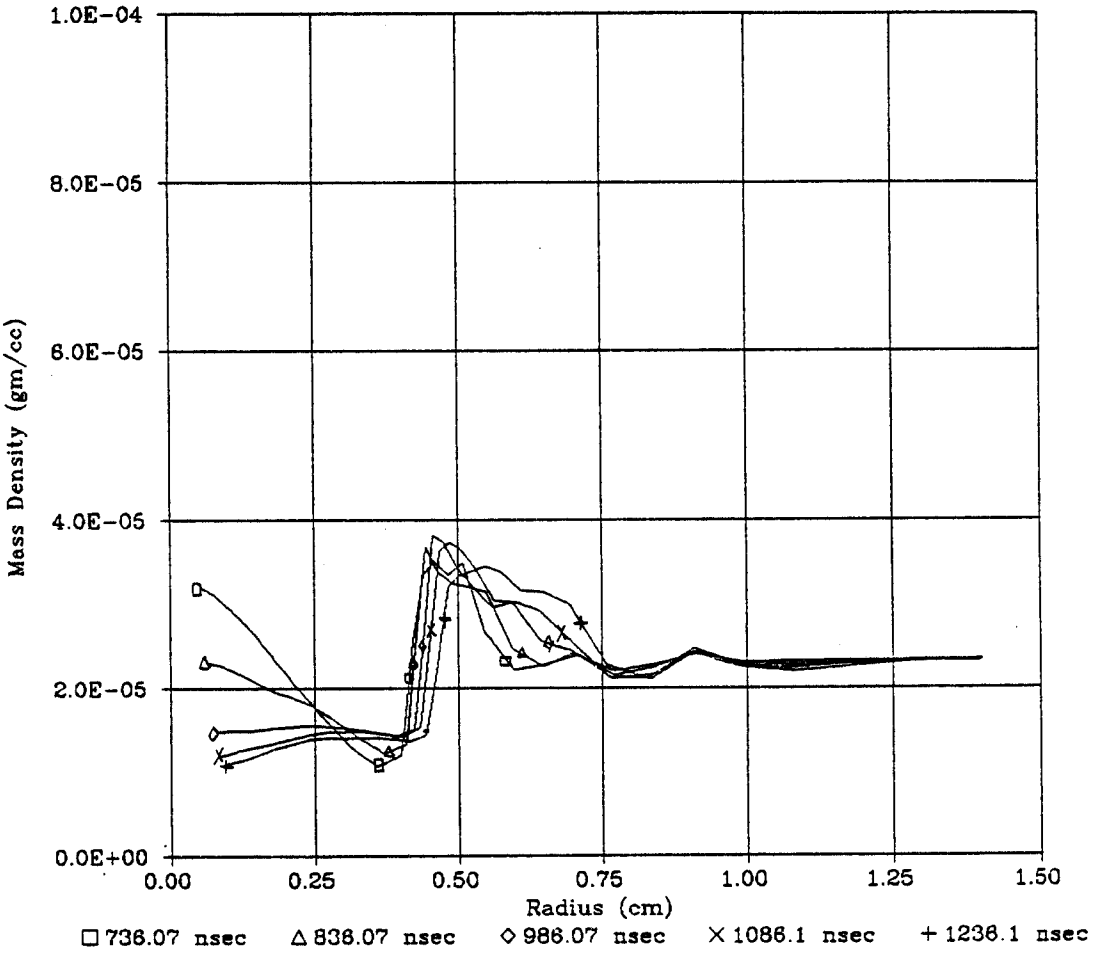


Figure 19. Mass density profiles for sample problem.