



**R-MHD: An Adaptive-Grid
Radiation-Magnetohydrodynamics
Computer Code**

Osman Yasar and Gregory A. Moses

January 1991

UWFDM-845

***FUSION TECHNOLOGY INSTITUTE
UNIVERSITY OF WISCONSIN
MADISON WISCONSIN***

**R-MHD: An Adaptive-Grid
Radiation-Magnetohydrodynamics Computer
Code**

Osman Yasar and Gregory A. Moses

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

<http://fti.neep.wisc.edu>

January 1991

UWFDM-845

Abstract

A one-dimensional adaptive-grid finite differencing computer code is presented for radiation-magnetohydrodynamics (R-MHD) simulations of fusion plasmas. The governing equations consist of the continuity, momentum, and energy equations of a non-relativistic, radiative and magnetized fluid plus field-intensity equations of radiation and magnetic fields. The equations are solved on an adaptive grid that dynamically moves the points at every timestep to follow steep gradients in physical quantities, such as temperature, mass density, pressure and momentum. The fluid equations are solved explicitly and the timestep control is based on the CFL condition for fluid flow. The radiation and magnetic field equations are solved implicitly because the timestep set by fluid flow is several orders of magnitude larger than the timescale for radiation and magnetic fields.

PROGRAM SUMMARY

Title of program: R-MHD

Catalogue number:

Program obtainable from: CPC Program Library, Queen's University of Belfast, N. Ireland

Computer: CRAY X-MP and Y-MP: *Installation:* San Diego Supercomputer Center, San Diego, CA 92186-9784.

Operating System: UNIX, UNICOS (CRAY Operating System).

Programming language used: FORTRAN 77

Number of bits in a word: 64

Peripherals used: line printer, disk files, tektronix terminal.

No. of lines in combined program and test deck: 6020

Keywords: Adaptive grid, equidistribution principle, nuclear fusion, plasmas, inertial confinement fusion, radiation hydrodynamics, magnetohydrodynamics.

Nature of physical problem

Radiation processes and radiation transport play an important role in inertial confinement fusion (ICF) plasmas. The dynamics of target implosion, and of the plasma channels can be strongly affected by radiative transfer. The coupling between the plasma, radiation and magnetic fields in light ion beam transport channels must be accurately determined to predict and interpret the outcome of ICF experiments.

Method of solution

The problem involves continuity, momentum and energy equations for the plasma, and field intensity equations for radiation and magnetic fields. These governing equations are solved on an adaptive grid system where mesh points follow dynamically the steep gradients to provide better resolution. The adaptive mesh generation is based on the equidistribution

principle and an explicit procedure is followed to update the mesh distribution in time. The plasma equations are also solved explicitly using donor-cell spatial differencing. The timestep control is done through the CFL condition and a constraint on the time-rate change of plasma energy. Since the timestep is chosen to be on the order of the plasma timescale, the radiation and magnetic field equations are solved implicitly considering the fact that they would have required a smaller timestep if they were solved explicitly. The radiation transfer equation describes how photons travel in the medium, thus it requires angular dependent solution of the specific intensity. The multigroup radiative transfer equation is solved for discrete angles and energy groups which makes the solution a detailed one but a necessary one because any approximation (e.g., diffusion) to a shortcut solution might have sacrificed the accuracy of the results. The discrete ordinate S_N method with many number of discrete angles would be very close to the real answer. Since the equation is also discretized for the photon energy groups, there is a need for groupwise opacity data which is found from an atomic physics code [1] published earlier. The equation of state data for plasma equations is also obtained from this code.

Restrictions on the complexity of the problem

Our radiation-magnetohydrodynamics (R-MHD) code is one-dimensional, assuming symmetry in all other directions. The plasma is considered in the magnetohydrodynamics (MHD) frame, thus it is assumed non-relativistic and it involves low frequencies. Although the code was originally designed for inertial confinement fusion plasmas, it can be applied to any compressible flow. When applied to plasmas, it should be noted that ions and electrons are assumed to have the same temperature. Regarding the numerics, the physical problem is solved on an adaptive grid system that is generated through an explicit procedure. The resolution via mesh refinement could be as low as a factor of $10^{-2} - 10^{-4}$ of the initial uniform mesh spacing. The discrete ordinate method used to solve the radiation field intensity is based on S_6 , although one can try to modify this for a higher number of angles.

Typical running time

The CPU time largely depends on the simulation time and the number of photon energy groups. For a run with 20 energy groups and a simulation time of $2 \mu\text{s}$, the code takes 3 – 5 minutes on CRAY Y-MP. This time may go up to 30 – 40 minutes for 200 energy groups. The hydrodynamic part does not seem to be a problem timewise, but the detailed solution of the radiation field is, when used with a large number of groups.

Unusual features of the problem

The R-MHD code involves a great deal of physics including the mesh adaptivity scheme, fluid dynamics, particle transport and radiation-magnetohydrodynamics. It is written in FORTRAN 77 and best handled in a UNIX environment with make utilities.

References

1. J. J. MacFarlane, *Comput. Phys. Commun.* 56 (1989) 259-278.

LONG WRITE-UP

1. Introduction

Radiation processes and radiation transport play an important role in inertial confinement fusion (ICF). The dynamics of target implosions can be strongly affected by radiative transfer. The *indirect drive* approach for target design purposely converts the driver beam energy in the form of laser light or ion beams to x-rays and uses these x-rays to drive the implosion. A background gas in the target chamber is important for transporting light ion beams from the final driver element to the target. Light ion beams with their high charge to mass ratio and high current densities, cannot propagate over large distances in a vacuum without significant beam divergence. One solution to this [1], is to fill the standoff region with a gas and strike a discharge along the path from the diode to the target. This gas is preconditioned by passing a current through it to form a long magnetized plasma between the ion diode and the target to confine the ion beams. Radiative transfer can play an important role during the formation of this z-discharge plasma channel. It certainly plays a role as an ion beam passes through and heats the plasma channel. The gas in the target chamber also interacts with the exploding target, creating a so-called microfireball. These hot and dense plasmas emit and reabsorb radiation at soft x-ray energies (1-1000 eV). The radiation represents a significant energy transfer mechanism in the plasma, thus the coupling between the plasma and the radiation field must be accurately determined to predict and interpret the outcome of ICF experiments.

Previous studies involved modeling these target chamber problems on a lagrangian grid system with the radiative transfer represented by a diffusion equation [2, 3, 4]. The plasma hydrodynamics in both ion beam transport channels and ICF target explosions has characteristics of steep gradients and shock structures that need to be handled with

a robust computational grid system. The advancements made in this paper and in Ref. [5] are solving the governing equations on an adaptive grid system and representing the radiative transfer without making the diffusion approximation. In the following sections, we describe our adaptive-grid radiation-magnetohydrodynamics (R-MHD) computer code that satisfies these needs.

2. Radiation-Magnetohydrodynamics Equations

We analyze only the radial motion of the plasma and thus assume a one dimensional model with symmetry in all other dimensions. The equations are presented for curvilinear as well as cartesian coordinates. The problem of ion beam transport channels is in cylindrical geometry whereas the target explosion problem is studied in spherical geometry. The radiation hydrodynamics equations are written in the laboratory frame and then transferred to the adaptive grid frame. They are solved along with a set of grid equations that describes how the grid system evolves in time. An explicit scheme is used for grid generation to prevent implicit coupling between the physical equations and the grid system. A conservative differencing scheme based on the control volume approach is chosen to retain the conservative nature of the governing equations. The numerical method to discretize the equations is a first-order upwind differencing scheme (donor cell). The dissipative characteristics of the upwind differencing are minimized with grid adaptation.

The governing equations for a nonrelativistic fluid in the frame of radiation magnetohydrodynamics (R-MHD) are described as [6, 7]

$$\begin{aligned}
 \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) &= 0 \\
 \frac{\partial}{\partial t} (\rho \mathbf{u} + \frac{1}{c^2} \mathbf{F}) + \nabla p + \nabla \cdot (\rho \mathbf{u} \mathbf{u} + \vec{\mathbf{P}}) &= \frac{\mathbf{J} \times \mathbf{B}}{c} \\
 \frac{\partial}{\partial t} (e_p + e_R) + \nabla \cdot (\mathbf{q} + (e_p + p) \mathbf{u} + \mathbf{F}) &= \mathbf{J} \cdot \mathbf{E}
 \end{aligned} \tag{1}$$

where e_p and e_R are energy density for plasma and radiation, \mathbf{F} and $\vec{\mathbf{P}}$ are the radiation flux and pressure tensor. Also $\mathbf{J} \cdot \mathbf{E}$ is the Joule heating term and is equal to $\mathbf{E}' \cdot \mathbf{J}'$, the rate of Joulean dissipation in fluid frame, plus $\mathbf{u} \cdot (\frac{\mathbf{J} \times \mathbf{B}}{c})$, the rate at which the force $\frac{\mathbf{J} \times \mathbf{B}}{c}$ does work. A relation is needed between the pressure, density and temperature to close the system. This relation can be found using the equation of state

$$p = (1 + \bar{Z})nkT$$

where \bar{Z} is the average charge state, n is number density and T is temperature.

The state of the radiation field and the magnetic field are found through the radiative transfer equation, the Maxwell's equations and Ohm's law respectively. The radiative transfer equation is a mathematical statement of the conservation of photons and is given in the following form [8]

$$\left(\frac{1}{c} \frac{\partial}{\partial t} + \hat{\Omega} \cdot \nabla\right) I(\mathbf{r}, t, \hat{\Omega}, \nu) = \eta(\mathbf{r}, t, \hat{\Omega}, \nu) - \chi(\mathbf{r}, t, \hat{\Omega}, \nu) I(\mathbf{r}, t, \hat{\Omega}, \nu) \quad (2)$$

where I is specific intensity, η and χ are called emissivity and extinction coefficients, and $\hat{\Omega}$ is the directional unit vector. The space-time evolution of the magnetic field in the MHD approximation is given as

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{u} \times \mathbf{B}) - \nabla \times \left(\frac{c^2}{4\pi\sigma} \nabla \times \mathbf{B}\right) \quad (3)$$

where σ is the electrical conductivity of the plasma. The equation describes how the magnetic field lines are convected and diffused in the non-relativistic and low-frequency plasma fluid.

In cylindrical coordinates, particularly for the z-pinch plasma problem, the radial dependence of an azimuthal magnetic field interacting with an axial current becomes

$$\frac{1}{c} \frac{\partial B}{\partial t} - \frac{\partial}{\partial r} \left(\frac{\eta c}{4\pi} \frac{1}{r} \frac{\partial}{\partial r} (rB)\right) + \frac{1}{c} \frac{\partial}{\partial r} (uB) = -\frac{\partial}{\partial r} (\eta J_b) . \quad (4)$$

Here J_b is the ion beam current density flowing in the channel, and η is the plasma resistivity. Also the multigroup photon conservation equation in one-dimensional (radial) cylindrical coordinates becomes

$$\frac{1}{c} \frac{\partial}{\partial t} I_g(r, t, \hat{\Omega}) + \frac{\mu}{r} \frac{\partial}{\partial r} (r I_g(r, t, \hat{\Omega})) - \frac{1}{r} \frac{\partial}{\partial \omega} (\zeta I_g(r, t, \hat{\Omega})) = \eta_g - \bar{\chi}_g I_g(r, t, \hat{\Omega}) \quad (5)$$

where we have replaced the streaming term [9] in the radiative transfer equation by

$$\hat{\Omega} \cdot \nabla I_g = \frac{\mu}{r} \frac{\partial}{\partial r} (r I_g) - \frac{1}{r} \frac{\partial}{\partial \omega} (\zeta I_g). \quad (6)$$

Cylindrical coordinates are complicated by the fact that even in one spatial dimension two angular variables, ζ and ω , are needed to describe the angular dependency of specific intensity I . ω is the angle between \hat{r} and $\hat{\Omega}$, and μ and ζ are the components of $\hat{\Omega}$ in \hat{r} and $\hat{\theta}$ directions. The angular discretization is done with a level-symmetric quadrature set described in Ref. [10].

The group constants η_g and $\bar{\chi}_g$ are given as

$$\eta_g = \int_{\nu_{g-1}}^{\nu_g} \eta_\nu d\nu$$

and

$$\bar{\chi}_g = \frac{\int_{\nu_{g-1}}^{\nu_g} \chi d\nu}{\int_{\nu_{g-1}}^{\nu_g} d\nu} \quad (7)$$

assuming that one has enough groups so that χ is nearly constant for each group. The radiation and material properties on the RHS of equation (5) are being measured in the inertial (lab) frame and they are assumed to be isotropic. Actually, there is a need to account for the velocity-dependence of these terms but this velocity-dependence is on the order of $O(v/c)$ and the authors believe the velocity-induced frequency shifts are ignorable for the problems of interest. The alternative is to use radiation/material properties evaluated in the comoving frame [8].

3. Equations on Adaptive Grid System

The grid generation is based on the equidistribution principle [11, 12]

$$\Delta x \cdot W(x) = constant$$

which states that at every timestep in the simulations the mesh points are arranged such that this product remains constant. Here, Δx is the cell size and W is a cell-averaged weight function. This principle leads to an explicit grid generation technique that has some advantages over an implicit one. It is easy to apply and easy to control such a mesh distribution. Mesh smoothness is simple and robust. In the present work, the smoothing is done on the weight function W , which indeed results in a smooth grid distribution.

Governing partial differential equations for fluid systems are generally described using a laboratory frame of reference. They can be transformed to a more general reference frame that reduces to the eulerian and lagrangian frames when the reference velocity is zero or equal to the local fluid velocity. In order to transfer the governing equations from (x, t) to such a moving frame (adaptive system; ξ, τ), we use a variable transformation as $\tau = t, \xi = \xi(x, t)$, thus $\frac{\partial}{\partial t} = \frac{\partial}{\partial \tau} - x_\tau \frac{\partial}{\partial x}$ and $\frac{\partial}{\partial x} = \frac{1}{x_\xi} \frac{\partial}{\partial \xi}$, or in another representation $[]_t = []_\tau - x_\tau []_x, []_x = \frac{1}{x_\xi} []_\xi$.

Transforming the R-MHD equations to the adaptive frame, we reach the following one-dimensional adaptive fluid equations.

Continuity

$$[x_\xi x^\delta \rho]_\tau + [x^\delta \rho(u - x_\tau)]_\xi = 0 ,$$

Momentum

$$[x_\xi x^\delta (\rho u + c^{-2} F)]_\tau + [x^\delta \rho u(u - x_\tau)]_\xi - [x^\delta x_\tau c^{-2} F]_\xi + x^\delta [p]_\xi + [x^\delta P]_\xi = x^\delta x_\xi F_m ,$$

Energy

$$[x_\xi x^\delta (e_p + e_R)]_\tau + [x^\delta e_p (u - x_\tau)]_\xi - [x^\delta x_\tau e_R]_\xi + [x^\delta (pu + q + F)]_\xi = x^\delta x_\xi E_m \quad (8)$$

where $\delta = 0, 1, 2$ for *planar, cylindrical and spherical* coordinates. Here F_m and E_m are the electromagnetic force and energy for the chosen coordinate system. Also x_ξ and x_τ are the mesh metric (jacobian) and speed, and we must estimate these mesh quantities before any attempt to solve the transformed equations. One also has to determine what the weight function should be. Many forms of weight function can be postulated. One that proves to be both simple and robust is the following

$$W = 1 + \alpha_a |A_x| + \beta_a |A_{xx}| + \alpha_b |B_x| + \beta_b |B_{xx}| \quad (9)$$

where A and B are some normalized physical quantity such as *velocity, pressure, mass, density, momentum density* or *temperature*. Also $A_x, A_{xx}, B_x,$ and B_{xx} are the first and second derivatives of A and B with respect to the spatial coordinate x .

It is important to note that the identity

$$[x^\delta x_\xi]_\tau = [x^\delta x_\tau]_\xi$$

must be preserved in the difference equations in order for them to preserve the conservative form. That is, when differenced equations are solved on the discretized coordinates, care should be taken to provide this identity relation which also introduces a formula to calculate the grid speed while the grid points move from x^n to x^{n+1} . Under the control volume approach, the *identity* equation is discretized as follows

$$\frac{(x^\delta x_\xi)_i^{n+1} - (x^\delta x_\xi)_i^n}{\Delta \tau} = (x^\delta x_\tau)_{i+\frac{1}{2}}^n - (x^\delta x_\tau)_{i-\frac{1}{2}}^n$$

where indices i and $i \mp \frac{1}{2}$ represent the i *th* cell's center and edge values. ξ is chosen to be the successive number of mesh points. Therefore $\Delta \xi$ is equal to 1. For this difference

equation to conserve the involved quantities, the cell edge velocities should be given as follows

$$(x_\tau)_{i\mp\frac{1}{2}}^n = \frac{(x^\delta)_i^{n+1} x_{i\mp\frac{1}{2}}^{n+1} - x_{i\mp\frac{1}{2}}^n (x^\delta)_i^n}{(x^\delta)_{i\mp\frac{1}{2}}^n \Delta\tau}.$$

4. Conservation of Energy

The total energy of the whole system is considered to be a combination of plasma, radiation and electromagnetic energies whose time dependent evolutions are given as the following:

Plasma Energy:

$$\begin{aligned} \frac{\partial}{\partial t} e_p + \nabla \cdot (\mathbf{q} + (e_p + p)\mathbf{u}) &= \mathbf{J}_p \cdot \mathbf{E} + S_{col} - \\ \int_0^\infty d\nu \int_{4\pi} d\Omega [\eta(\mathbf{r}, t, \hat{\Omega}, \nu) + \chi(\mathbf{r}, t, \hat{\Omega}, \nu) I(\mathbf{r}, t, \hat{\Omega}, \nu)] & \end{aligned} \quad (10)$$

Radiative Energy:

$$\frac{\partial}{\partial t} e_R + \nabla \cdot \mathbf{F} = \int_0^\infty d\nu \int_{4\pi} d\Omega [\eta(\mathbf{r}, t, \hat{\Omega}, \nu) - \chi(\mathbf{r}, t, \hat{\Omega}, \nu) I(\mathbf{r}, t, \hat{\Omega}, \nu)] \quad (11)$$

Electromagnetic Energy:

$$\frac{\partial}{\partial t} e_{em} + \nabla \cdot \mathbf{S} = -\mathbf{J}_{net} \cdot \mathbf{E} \quad (12)$$

The total energy change rate is then

$$\frac{\partial}{\partial t} e_t + \nabla \cdot (\mathbf{q} + (e_p + p)\mathbf{u} + \mathbf{F} + \mathbf{S}) = \mathbf{J}_p \cdot \mathbf{E} + S_{col} - \mathbf{J}_{net} \cdot \mathbf{E} + S_{col} \quad (13)$$

where $\mathbf{S} = \frac{c}{4\pi}(\mathbf{E} \times \mathbf{B})$ is the electromagnetic energy flux. Also here $\mathbf{J}_{net} = \mathbf{J}_p + \mathbf{J}_{other}$.

These equations are not necessarily solved at each time step except for the plasma energy equation. Radiation energy and electromagnetic energies are controlled by the radiative transfer and magnetic diffusion equations. These energy equations, however, provide a means to check the solution of those that are solved through the adaptive mesh technique. Based on the solutions we calculate what the system energy is, then we also calculate the energies through the energy equations; actually we solve them too to a first degree approximation. Thus we compare the solution of our complex equations with rather simple equations to check for consistency. The simulations done so far have shown energy conservations of 95 – 98 %.

Conservation of Magnetic Flux and Current:

In addition to the energy conservation equations, the magnetic flux conservation check also provides an indication of how accurately the magnetic diffusion is being solved. To get an expression for a numerical check on magnetic flux we will integrate Eq. (4) over space and time. That is,

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

This indeed is **Faraday's Law** and becomes

$$\frac{1}{c} [\phi(t) - \phi(0)] = \int_0^t [E_l(r_{max}, t) - E_l(0, t)] \quad (14)$$

where $\phi = \int_0^{r_{max}} dr' B(r', t)$, $E_l = E_f - \frac{1}{c} uB$. In finite difference form,

$$\frac{1}{c} (\phi^{n+1} - \phi^0) = \varepsilon^{n+1}$$

where $\varepsilon^{n+1} = \varepsilon^n + \Delta t^n (E_{imax}^{n+1} - E_{i=1}^{n+1})$.

The current conservation check is a comparison between the discharge current I_d and the current calculated by integrating the net current density J_{net} . In other words, the

current conservation check compares the discharge current I_d^n , as obtained either from the current equation solver or the user-specified current, and the following one that is calculated through the current density J_i^n as

$$I_d^n = 2\pi \int_0^{r^{max}} J_{net} r \, dr \cong 2\pi \sum_{i=1}^{imax} J_i^n r_i^n \Delta r_i^n. \quad (15)$$

5. The Equation of State and Opacity Tables

The solution of governing equations in previous sections must be supported with some tabular data for average charge state, specific internal energy, and emission and extinction coefficients; all being dependent on the temperature and the density of the plasma. The tables are generated by the atomic physics computer code IONMIX [13] that takes into account both LTE and non-LTE conditions. A logarithmic scale is used to tabulate the temperature and density. The tabular values for specific energy as a function of temperature and density, $\epsilon = f(\rho, T)$, are converted to a table for temperature as a function of the specific internal energy and density $T = f(\rho, \epsilon)$. This is because the scheme solves an energy equation for the plasma rather than a temperature equation as is done in many hydrodynamics codes. Thus, knowing the total energy and the hydro energy one can deduce the specific internal energy that is used to predict the plasma temperature.

6. Time Step Control

Among the governing equations, continuity, momentum and energy equations are solved with an explicit scheme whereas the magnetic diffusion and radiative transfer are solved with an implicit scheme. The time step, Δt , for the simulations is calculated as

$$\Delta t = \min(\Delta t_{CFL}, \Delta t_{jh}, \Delta t_{bc}, \Delta t_{re}, \Delta t_{ra})$$

where the CFL condition follows as

$$CFL = (|u| + a) \frac{\Delta t_{CFL}}{\Delta x} \leq 1$$

and $\Delta t_{jh}, \Delta t_{bc}, \Delta t_{re}, \Delta t_{ra}$ are calculated from a constraint on the allowed change in plasma energy (e_p) due to joule heating, beam-collisional energy, radiation emission and absorption for a given time step. That is, we assume

$$\Delta t_i \Delta \dot{e}_{p,i} = 0.1 e_p$$

where i represents the components above.

The differencing formulation describes the solution at a mesh point in terms of the solutions at neighboring points. Hence, in order for the solutions to be stable and error-free (as much as possible), we do not want any information to move more than one cell at a time step. This is a stability condition for the explicitly solved continuity, momentum and energy equations. For most of the simulations Δt is on the order of nanoseconds and this is small compared to the timescale of hydromotion but large compared to the magnetic diffusion and radiative transfer timescales. An attempt to work with smaller time steps would certainly make the simulations more costly and lengthy. Therefore we have chosen to solve magnetic and radiation equations with an implicit scheme that does not have any time constraint.

7. Subroutines

ADAPT Performs the adaptive mesh logic to obtain the new mesh locations, x_k^{n+1} , based on the Dwyer equidistribution method.

BFIELD Calculates the magnetic field through the magnetic field diffusion equation. It

uses an implicit time differencing along with a three point spatial differencing, and thus it solves a tridiagonal system. Also found here are the current density and the electric fields.

BOUND Sets up the boundary conditions at every time step.

COORD Updates the coordinate control variables depending on the geometry of the problem.

FUNCT Defines the weighting functions for the grid adaption.

INIT Initializes the primitive solution variables at the initial time.

KAPPA Finds the thermal conductivity which is a function of the temperature and some other plasma conditions.

LLAM Calculates the Coulomb logarithm $Log\Lambda$.

LOOK Does table lookup for one-variable dependent table.

LOOKV1 Does table lookup for two-variable tables, especially average charge table.

LOOKV2 Table lookup for the opacities. Again it is a three-variable table depending on temperature, density and energy group.

MAIN Main program that controls the calculation. Reads in the input, defines the constants, parameters, advances the solution for the next time step, makes a conservation check, and dumps the output.

MESHIN Initializes the adaptive mesh to the initial conditions and it is done to prevent rapid mesh movement for the first few time steps.

PDESOL Solves a single time step of hyperbolic partial differential equations (compressible flow problem) using the first order upwind explicit scheme.

OMEGA Defines the angular variable discretization for the S_N method. It uses a level-quadrature method and discretizes 2 angles for the cylindrical case and one angle for spherical cases.

RADFLD Solves the radiative transfer equation by the discrete ordinate method. It uses an implicit time differencing and a diamond spatial differencing method.

RCOLL Calculates the collisional energy loss of ions which is a function of the temperature, density and some other plasma and beam properties.

RESIST Calculates the electrical resistivity.

RESTRT Restarts the problem from a previous run.

TABLE Table lookup and conversion for EOS table.

UPWIND Performs the actual solution of the differenced equations using the first order upwind differencing method.

WEIGHT Computes the adaptive weight integrals based on Dwyer's adaptive method.

Among these subroutines **pdesol**, **upwind**, **adapt**, **weight**, **radfld**, **bfield** are most crucial ones for job execution. The flow diagram for a typical run follows the structure in Figure 1. After initialization of the grid system, the problem runs from the main program periodically calling subroutine **pdesol** which in itself calls the mesh updaters (**adapt**, **coord**, **bound**) and the solvers such as **upwind**, **radfld**, **bfield**.

8. Common Blocks

Table 1. COMMON/MESH/

VARIABLE	DESCRIPTION
xn(k)	grid locations at t_n in x dimension (<i>cm</i>)
yn(k)	grid locations at t_n in y dimension (<i>cm</i>)
xnp1(k)	grid locations at t_{n+1} (<i>cm</i>)
ynp1(k)	grid locations at t_{n+1} (<i>cm</i>)
psie($k \pm 1/2$)	geometric factor ($1,r,r^2$ for slab, cylinder and sphere)
psic(k)	geometric factor ($1,r,r^2$ for slab, cylinder and sphere)
dxn(k)	grid size at t_n (<i>cm</i>)
dxnp1(k)	grid size at t_{n+1} (<i>cm</i>)
dxdt($k \pm 1/2$)	grid velocity (<i>cm/sec</i>)

Table 2. COMMON/ADAPTX/

VARIABLE	DESCRIPTION
alpha(1:2)	used in $W = 1 + \alpha[A + B]_x + \beta[A + B]_{xx}$
beta(1:2)	used in $W = 1 + \alpha[A + B]_x + \beta[A + B]_{xx}$
ra(1:2)	R_α , fraction of points assigned to follow U_x
rb(1:2)	R_β , fraction of points assigned to follow U_{xx}
dxmin	average minimum mesh spacing desired (<i>cm</i>)
xint	$\int_0^L W dx'$
xint1a	$\int_0^L A_x dx'$
xint1b	$\int_0^L A_{xx} dx'$
xint2a	$\int_0^L B_x dx'$
xint2b	$\int_0^L B_{xx} dx'$

Table 3. COMMON/MISC/

VARIABLE	DESCRIPTION
cycle	current number of cycles
delt	current time step (<i>sec</i>)
cfl	CFL number (stability)
time	current simulation time (<i>sec</i>)
iadapt	adaption flag (0,1:off,on)
icoord	coordinate flag (1,2,3:slab,cylinder,sphere)
iloop	number of cycles for initial iteration
xl	domain length (<i>cm</i>)
xmax,xmin	maximum and minimum lengths (<i>cm</i>)
ireg	number of meshes
imx	length of subscript over mesh points ($ireg + 2$)
im1,im2	$imx - 1$, and $imx - 2$
kpta,kptb,kptd	$2, imx - 1, imx$
ivar1,ivar2	misc boundary conditions

Table 4. COMMON/BC/

VARIABLE	DESCRIPTION
bc(k)	boundary conditions
ibcr,ibcl	right and left bc
tbcr,tbcl	right and left temperature bc
ubcr,ubcl	right and left velocity bc

Table 5. COMMON/BEAM/

VARIABLE	DESCRIPTION
bcur(k)	beam current density ($statA/cm^2$)
bdens(k)	beam particle density ($particles/cm^3$)
scoll(k)	beam collisional energy loss rate ($erg/sec \cdot cm^3$)
btime,puls	beam shooting time and its duration (sec)
bcmx	beam peak current density ($statA/cm^2$)
bmass,vbeam	beam mass and velocity ($g, cm/sec$)
bchrg, achrg	beam and background electron charge
alphms	background electron mass (g)
poynt,emjoul	poynting vector and joule heating terms ($erg/sec \cdot cm^2, erg/sec \cdot cm^3$)
pljoul,emeng	plasma joule heating and em energy terms ($erg/sec \cdot cm^3$)

Table 6. COMMON/TIMEN/

VARIABLE	DESCRIPTION
u1(k)	volumetric mass (<i>g</i>)
u2(k)	volumetric momentum (<i>g · cm/sec</i>)
u5(k)	volumetric energy (<i>erg</i>)
u6(k)	volumetric radiation energy (<i>erg</i>)

Table 7. COMMON/TERMS/

VARIABLE	DESCRIPTION (<i>erg/sec</i>)
term11(k)	work-energy term
term22(k)	artificial viscosity-energy term
term33(k)	joule heating-energy term
term44(k)	heat conduction-energy term
term55(k)	radiation convection-energy term
term66(k)	radiation heat flux-energy term
term77(k)	collisional-energy term
term88(k)	radiation emission-energy term
term99(k)	radiation absorption-energy term

Table 8. COMMON/TABL/

VARIABLE	DESCRIPTION
rhotb(1:10)	logarithmic mass density vector for EOS table $\log(g/cm^3)$
tmptb(1:20)	logarithmic temperature vector for EOS table $\log(eV)$
arhot,brhot	initial values and increment for rhotb()
attb,bttb	initial values and increment for tmptb()
etable(10,20)	specific internal energy array, $f(t, \rho)$, (<i>joule</i>)
zch(10,20)	average charge state array, $f(t, \rho)$

Table 9. COMMON/LOOK1/

VARIABLE	DESCRIPTION
tm(k)	initial temperature (<i>eV</i>)
xnpos(k)	positions for initial temperature (<i>cm</i>)
axnpos,bxnpos	initial values and increment for xnpos() (<i>cm</i>)
ntable	table size

Table 10. COMMON/OPA/

VARIABLE	DESCRIPTION
engrup(ν)	energy groups for photons
orgp(ν, T, ρ)	extinction coefficient (opacity), (cm^{-1})
opgpe((ν, T, ρ))	emission coefficient (opacity), ($erg/cm^3 \cdot sec \cdot Hz \cdot sr$)

Table 11. COMMON/PRMVAR/

VARIABLE	DESCRIPTION
rho(k)	mass density (g/cm^3)
u(k)	average fluid element velocity (cm/sec)
e(k)	energy density (erg/cm^3)
p(k)	pressure density (erg/cm^3)
t(k)	temperature ($^{\circ}K$)
b(k)	magnetic field ($gauss$)
cur(k)	net current density ($statA/cm^2$)
pcur(k)	plasma current density ($statA/cm^2$)
bcur(k)	beam current density ($statA/cm^2$)
elecl(k)	laboratory frame electric field ($statvolt/cm$)
felec(k)	fluid frame electric field ($statvolt/cm$)
radint(k),ordint(k)	total radiation intensity at times t_n, t_{n+1} ($erg/cm^2 \cdot sec$)
radeng(k),ordeng(k)	radiation energy density at times t_n, t_{n+1} (erg/cm^3)
radpre(k),ordpre(k)	radiation pressure at times t_n, t_{n+1} (erg/cm^3)
radfrc(k),ordfrc(k)	radiation force at times t_n, t_{n+1} ($erg/cm^2 \cdot sec$)
radfrq(ν, μ, k)	specific radiation intensity at t_n ($erg/cm^2 \cdot sec \cdot Hz \cdot sr$)
ordrad(ν, μ, k)	specific radiation intensity at t_{n+1} ($erg/cm^2 \cdot sec \cdot Hz \cdot sr$)

9. Input

The input is provided in two text files: *prminp* (primary input data) and *tabinp* (tabular data for EOS and opacity tables). Between these two, *tabinp* need not change unless the gas type is different, or the opacity group structure is changed. The primary input, however, reflects the entries of how many grid points to use, or what fraction of points to assign for adapting a certain function. The adaptive function is set in the code (in subroutine *funct*) and this is where it will need changing. The format of the input file *prminp* is given below (all formatted with `format(e11.0)`):

head: One line of text describing the run.

tinit: Initial time to start with (*sec*).

rstrt: Restart flag (0-no, 1-yes) if run depends on a previous run (*sec*).

icoord: Coordinate flag: 1-planar, 2-cylindrical, 3-spherical.

ibcl: Left boundary condition, 0-no, 1-yes.

ibcr: Right boundary condition, 0-no, 1-yes.

ibct: Top boundary condition, 0-no, 1-yes.

ibtb: Bottom boundary condition, 0-no, 1-yes.

tbcl: Left temperature boundary condition, 0-no, 1-yes.

tbcr: Right temperature boundary condition, 0-no, 1-yes.

tbct: Top temperature boundary condition, 0-no, 1-yes.

tbtb: Bottom temperature boundary condition, 0-no, 1-yes.

gamma: Ratio of specific heats.

xmui: Absolute viscosity ($g/cm \cdot s$).

xkapi: Thermal conductivity ($erg/cm \cdot K \cdot s$).

eamb: Ambient temperature (K).

wtmol: Atomic or molecular weight (a.m.u.).

rgas: Universal gas constant, 8.328×10^7 (*erg/g · mole · K*).

ireg: Number of grid points in *x* direction.

xmin: Minimum *x* value.

xmax: Maximum *x* value.

jreg1: Number of grid points in *y*-region 1.

dy1: Δy in *y*-region 1.

jreg2: Number of grid points in *y*-region 2.

dy2: Δy in *y*-region 2.

jreg3: Number of grid points in *y*-region 3.

dy3: Δy in *y*-region 3.

iadapt: Adaptation flag: 0-no, 1-yes.

iloop: Initialization loop to prevent rapid mesh movement.

ra(1): Fraction of points to assign to the 1st derivative of function 1.

rb(1): Fraction of points to assign to the 2nd derivative of function 1.

ra(2): Fraction of points to assign to the 1st derivative of function 2.

rb(2): Fraction of points to assign to the 2nd derivative of function 2. All four normalized to 1.

dxmin: Average minimum mesh spacing.

dtrst: Time step for dumping information for restarting purposes.

dtmpra: Time step for dumping information for various quantities.

dtmprb: Time step for dumping information for various quantities.

dtmprc: Time step for dumping information for various quantities.

dtmprd: Time step for dumping information for various quantities.

ictrlb: Control flags for which quantities to write out: file b.

ictrlc: Control flags for which quantities to write out: file c.

delt: Initial time step (*sec*).

cfl: CFL number.

con(3): Artificial viscosity coefficient.

ivar1: Miscellaneous boundary condition.

ivar2: Miscellaneous boundary condition.

bvar3: Pressure ratio for shock tube problem (not used for z-pinch).

bvar4: Pressure on left side of shock tube (*dynes/cm²*).

bvar5: Temperature on left side of shock tube (*K*).

dtmin: Minimum time step (*sec*).

idelt: Number of time steps for initialization.

con(50): Smoothing coefficient – *k* em th cell.

con(51): Smoothing coefficient – *k* – 1, *k* + 1 em th cells.

con(52): Smoothing coefficient – *k* – 2, *k* + 2 em th cells.

con(7): Maximum number of weight iterations.

con(8): Absolute error on mesh initialization.

con(9): Adaptation error residual.

axnpos: Step value for x in representing initial T as $T(x)$.

bxnpos: Initial value for x in representing initial T as $T(x)$.

ntable: Table size for representing initial T as $T(x)$.

rhoin: Initial number density (*particles/cm³*).

curin1: Discharge current density amplitude (*statA/cm²*) - first pulse.

tper1: Time period for the first pulse (*sec*).

curin2: Discharge current density amplitude (*statA*) - main pulse.

tper2: Time period for the main pulse (*sec*).

trelax: Delay time between pulses (*sec*).

epsilon: Criteria for adaptation integrals.

btime: Beam shooting time (*sec*).

puls: Beam duration time (*sec*).

bcmax: Beam current density amplitude (*statA/cm²*).

frac: Relates to the beam flow radius.

bmass: Beam particle mass (*g*).

vbeam: Beam particle velocity (*cm/s*).

bchrg: Beam particles charge.

achrg: Background particles (electrons) charge.

alphms: Background particles mass (*g*).

TABINP input file

attb: Increment in temperature (logarithmic) for the average charge state and EOS tables.

bttb: Initial temperature (*eV*) for the average charge state and EOS tables.

arhot: Increment in density (logarithmic) for the average charge state and EOS tables.

brhot: Initial density (logarithmic) for the average charge state and EOS tables.

head0: Heading for charge state table (one line).

zch(10,20): Average charge table vs. temperature and density.

head1: Heading for EOS table.

etable(10,20): Specific internal energy table as $\epsilon = f(T, \rho)$, (*Joule*) = $f(\log(eV), \log(g \cdot cm^{-3}))$.

head2: Heading for photon energy groups.

engrp(20): Photon group energies.

head3: Heading for group extinction coefficients.

orgp(20,10,20): Multigroup extinction coefficients (*cm⁻¹*) as $f(\nu, T, \rho)$ with format 4(e12.6).

head4: Heading for group emission coefficients.

opgpe: Multigroup emission coefficients ($erg\ cm^{-3}s^{-1}Hz^{-1}sr^{-1}$) as $f(\nu, T, \rho)$ with format 4(e12.6).

10. Output and Other Auxiliary Files

There are five output files; four intended for plotting and one for restarting purposes. Those files are **gdmpa**, **gdmpb**, **gdmpc**, **gdmpd**. Among these, only **gdmpc** is actively used now. The time step for dumping information to these files is in the **prminp** input file, and it is currently set up to dump only to **gdmpc**. There is a mask array for the eleven physical quantities that can be written out to **gdmpc**. If it is 1, then it attempts to write it out. Those physical functions are set up to be: velocity, magnetic field, pressure, density, mesh spacing, temperature, plasma current, channel current, radiation energy density, emission, extinction. The order in the mask array is reversed, that is the rightmost element belongs to velocity.

There are plotting routines on the CRAY that take this data and make plots for various times. This plot package consists of two routines called **plt.f** and **subplt.f**, each written in FORTRAN to take advantage of the DISSPLA and GKSNCAR graphics libraries on the CRAY. When fed with the **gdmpc** input file, the plot program creates a file called **gmeta** and this file eventually can be displayed through a tektronix terminal (window) by a **gplot** command as “gplot -stek gmeta”.

Table 12. FORTRAN files

MAIN.f	initial.f	omega_cyl.f	restart.f
adapt.f	kappa.f	omega_sph.f	sbcmp.f
bfield.f	llam.f	pdesol.f	subcvm.f
boundary.f	look.f	radfld_cyl.f	table.f
coord.f	lookv1.f	radfld_sph.f	upwind.f
dtstep.f	lookv2.f	rcoll.f	weight.f
funct.f	meshini.f	resist.f	

11. How to Compile: Make

The source program consists of FORTRAN programs (.f) each holding a subroutine, and also header files (.h) that hold common blocks for those subroutines. Running problems in different coordinates (especially spherical and cylindrical) requires different subroutines to solve the radiative transfer equation, thus the idea of modularity was used to retain the code's generality. Table 12 lists the FORTRAN files. Among those, radfld_cyl.f (and radfld_cyl.h) or radfld_sph.f (and radfld_sph.h) has to be changed to radfld.f (and radfld.h) depending on the problem to run. This also has to be done with omega_cyl.f and omega_sph.f (no need for changing anything to omega.h since it already exists and it is the same for both coordinate systems).

Because the program is mainly developed for UNIX environments, it should be easy to compile it and run it under any UNIX environment varying from workstations to CRAY Y-MP.

12. Examples

The following two examples are typical runs for z-pinch plasma channels used in the design of light ion beam fusion reactors. The problem is in cylindrical coordinates and the simulation is done for the radial motion of an argon plasma that interacts with the magnetic and radiation fields in the channel. Initial conditions assume a uniform density of 3.5×10^{17} particles/cm³, and a gaussian temperature profile with 0.3 cm half-width resulting from a laser preionization in the channel. The plasma is subject to a discharge current shown in Figure 2. Most of the input parameters are given in the file “prminp” except that the radiative properties of the plasma are given in “tabinp”. Following section 12, you will find the “prminp” file for such a problem. The second input file (tabinp) will not be given because of its size. Also given there are some portions of the output file (gdmprc) and some plots found from that file. The opacity tables in “tabinp” are found by running an atomic physics code IONMIX [13].

12.1. Example 1

The first run is an MHD simulation of the plasma. Thus, the radiation field is artificially turned off. The subroutine call to *radfld* in *pdesol* is simply commented. Without the magnetic field, the simulations of a fluid expanding in the *r*-direction is in the form of a blast-wave. This motion is due to the gaussian temperature (thus pressure) profile whose gradient sets the gas into the radial motion. With a magnetic field in the azimuthal direction, the plasma is subject to a magnetic force radially inward. This magnetic force works against the gas pressure to reverse the radial motion. Figure 3 shows the mass density of the plasma for various times. The information on this plot and others presented here is taken from output file gdmprc. Since the discharge current increases in time, the plasma does not feel the magnetic pressure until the second pulse comes along around $t = 1.5 \mu\text{sec}$. Prior to

this time, the plasma is expanding leaving behind a rarefaction region. The presence of a stronger field pushes the plasma back toward the center creating a situation not desired for beam propagation since a high density in the center means a high energy loss for the beam through plasma-beam collisions. The temperature profiles, Figure 4, show the typical shock heating behind the shock region. The temperature is elevated by one order of magnitude due to the ohmic heating caused by the discharge current. The magnetic field in Figure 5 displays that this cylindrical plasma column behaves like a coaxial cable with a conductive inner region surrounded by a less conducting layer. The temperature drop around $r = 0.5$ cm causes a drop in the conductivity and the linearly increasing magnetic field up to that point starts to drop as $\propto 1/r$. The input files presented here should be able to reproduce the MHD results if user turns off the radiation by commenting “call radfld” in the “pdesol.f” subroutine.

12.2. Example 2

The second example is to add the radiation field to the previous example. The purpose of doing such a run is to see how the presence of radiation will affect the plasma hydrodynamics. The hot plasma in the center radiates the most and this radiation gets reabsorbed at the cool region if the mean free path of photons is smaller than the channel radius. It turned out that there are significant radiative effects when the plasma interacts with radiation in these conditions. The emission of radiation first of all flattens the temperature profile and causes a large drop due to emission. Secondly, the cold plasma reabsorbs the radiation coming from the center, and this absorption causes a premature expansion in the channel which ends up lowering the magnetic field peak value. When compared to previous plots, Figures 6, 7, 8 and 9 illustrate how this is happening: the channel radius (where magnetic field peaks) changes from 0.6 cm to 0.8 cm, the magnetic field drops from 23

kG to 19 kG. For plasma channels, these two parameters (the peak magnetic field and its corresponding channel radius) are important design issues. A certain magnetic field has to be achieved at a certain channel radius (typically 0.5 cm) in order for the beam to transport the required energy to the target efficiently. One has to account properly for the radiation effects since those effects simply degrade the quality of the channel. An overestimation or an underestimation could create a significant flaw in the channel design. The use of the radiative transfer equation, and of a delicate grid system should all contribute to the quality of the results one would get through use of the R-MHD code.

Acknowledgement

Support for this work was supplied through a grant from the NSF sponsored San Diego Supercomputer Center.

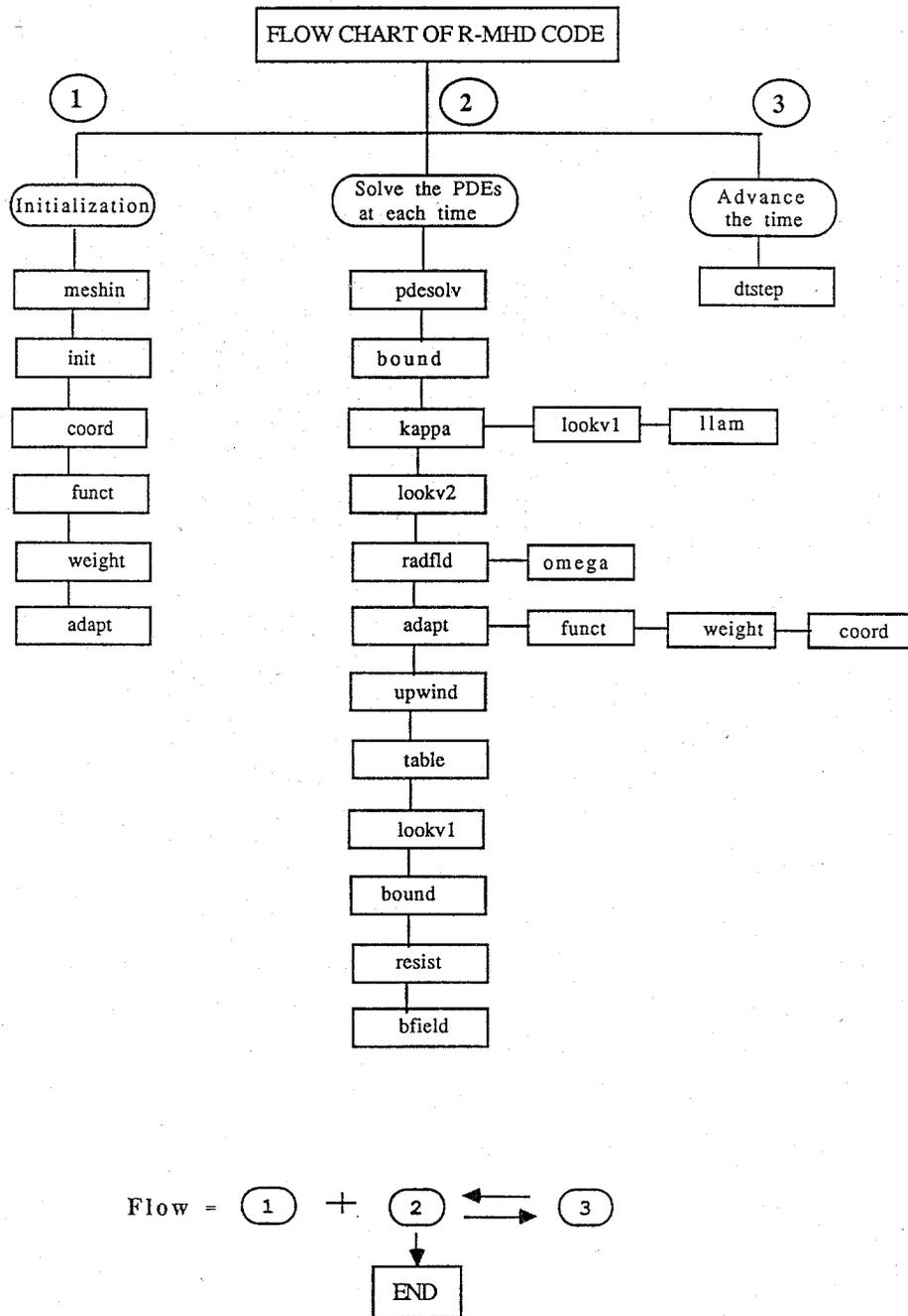


Figure 1. Flow diagram for R-MHD.

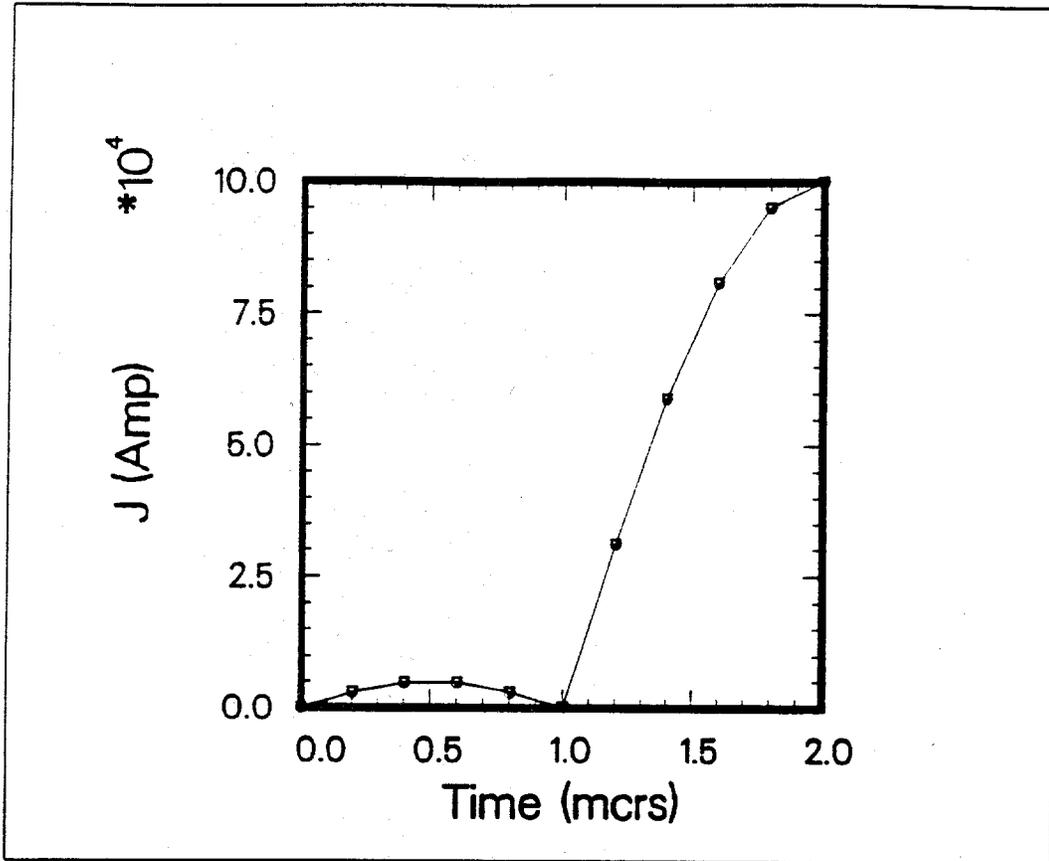


Figure 2. Double pulse discharge current history for the formation of plasma channels.

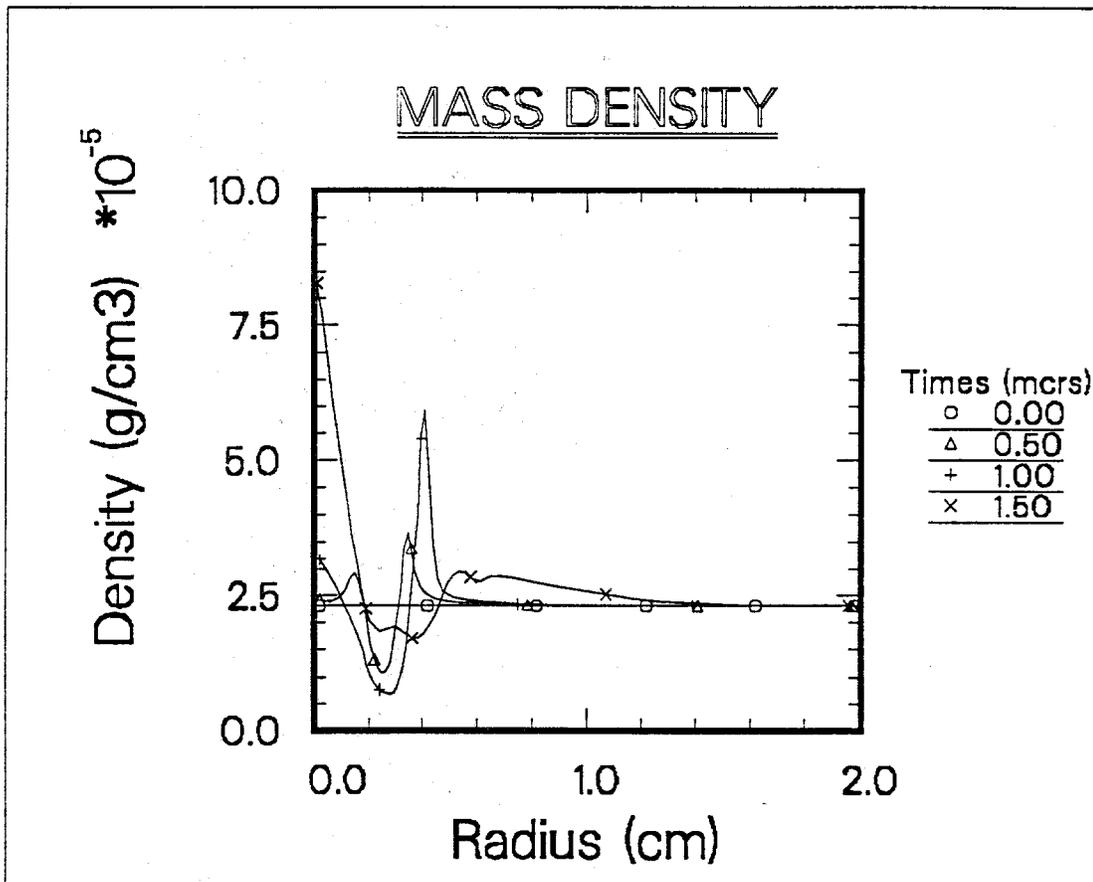


Figure 3. Mass density profiles for an argon plasma channel (MHD case).

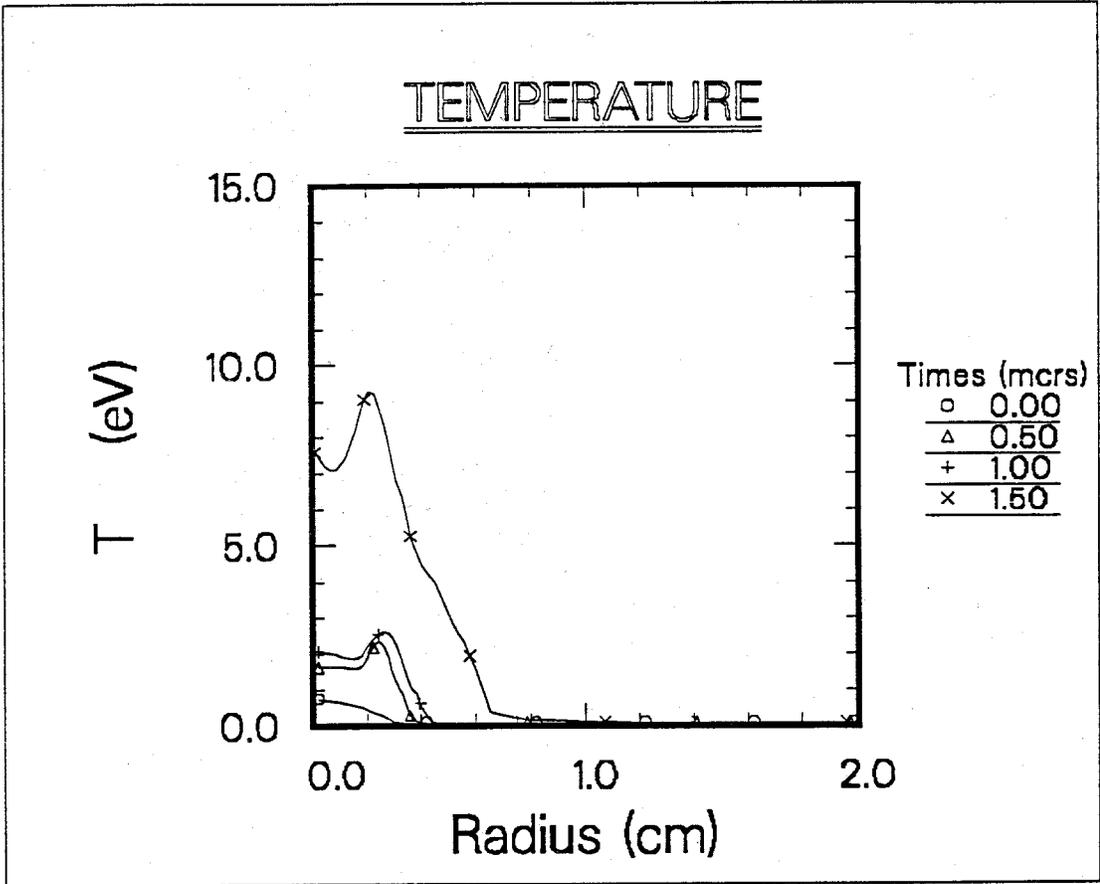


Figure 4. Temperature profiles for an argon plasma channel (MHD case).

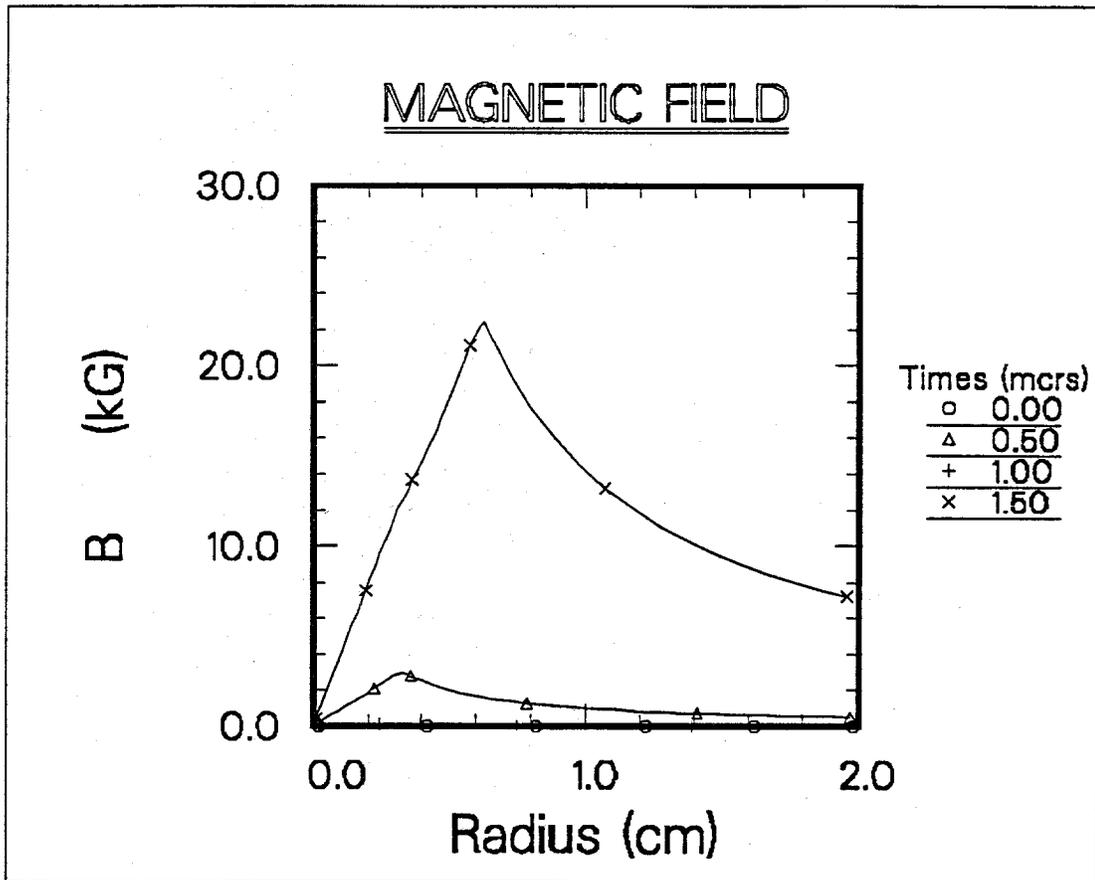


Figure 5. Magnetic field profiles for an argon plasma channel (MHD case).

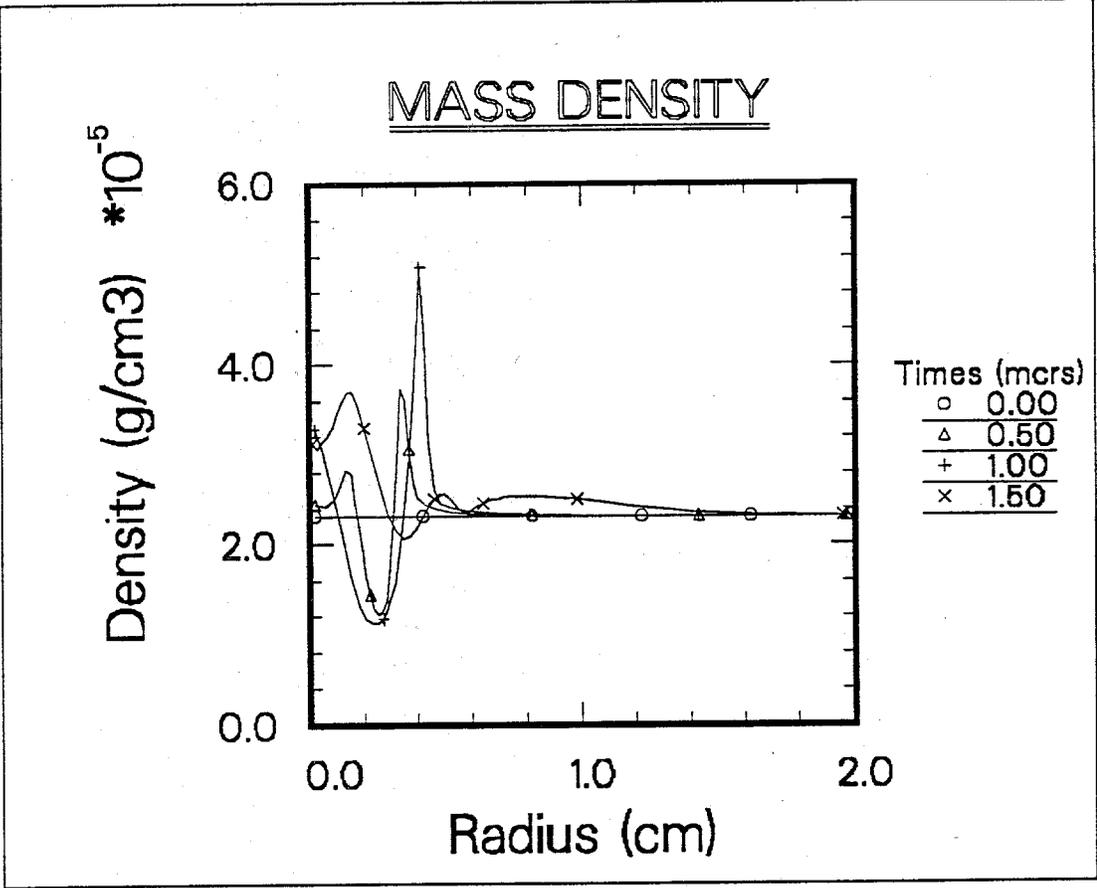


Figure 6. Mass density profiles for an argon plasma channel (R-MHD case).

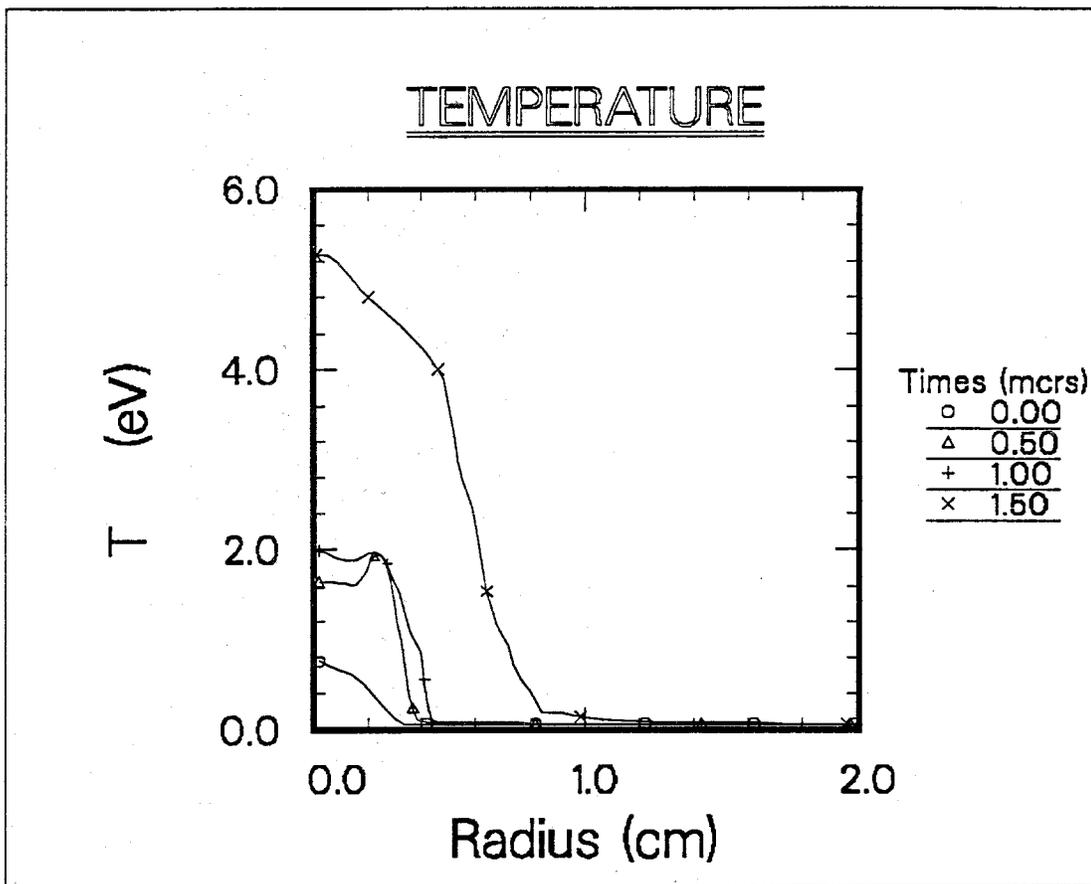


Figure 7. Temperature profiles for an argon plasma channel (R-MHD case).

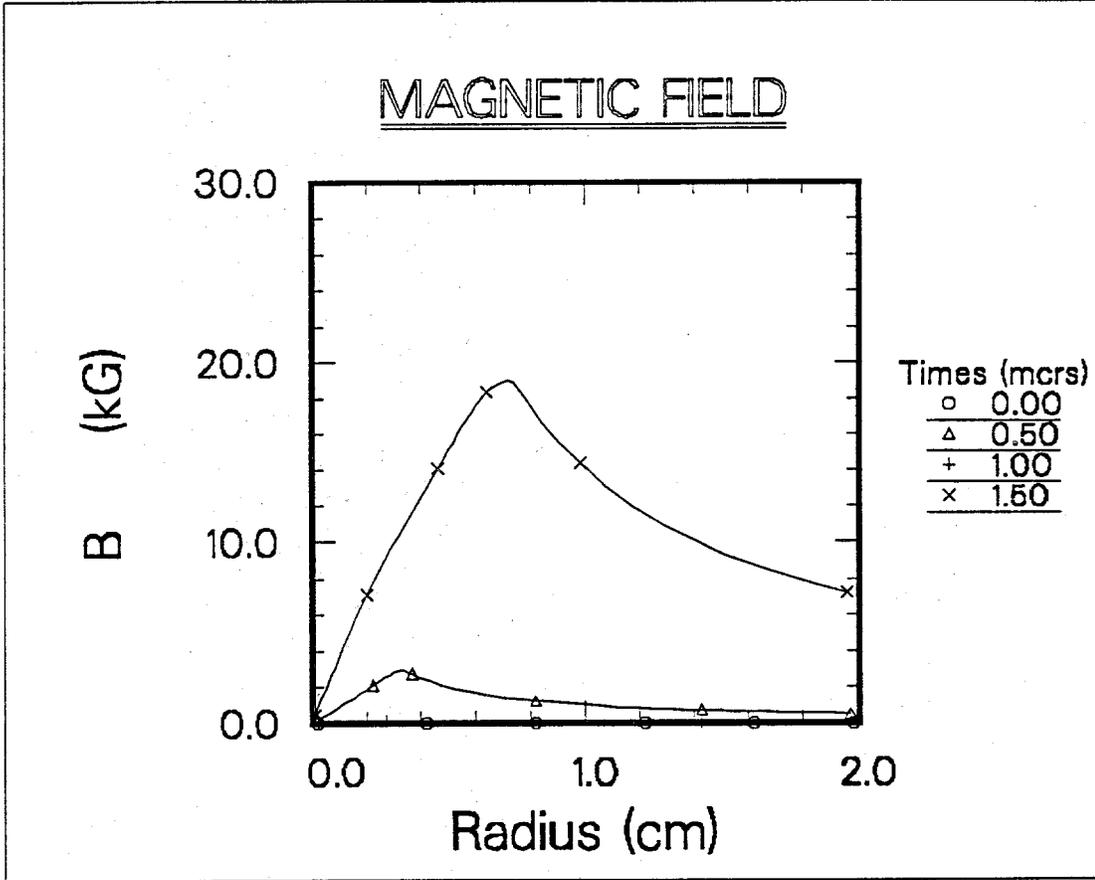


Figure 8. Magnetic field profiles for an argon plasma channel (R-MHD case).

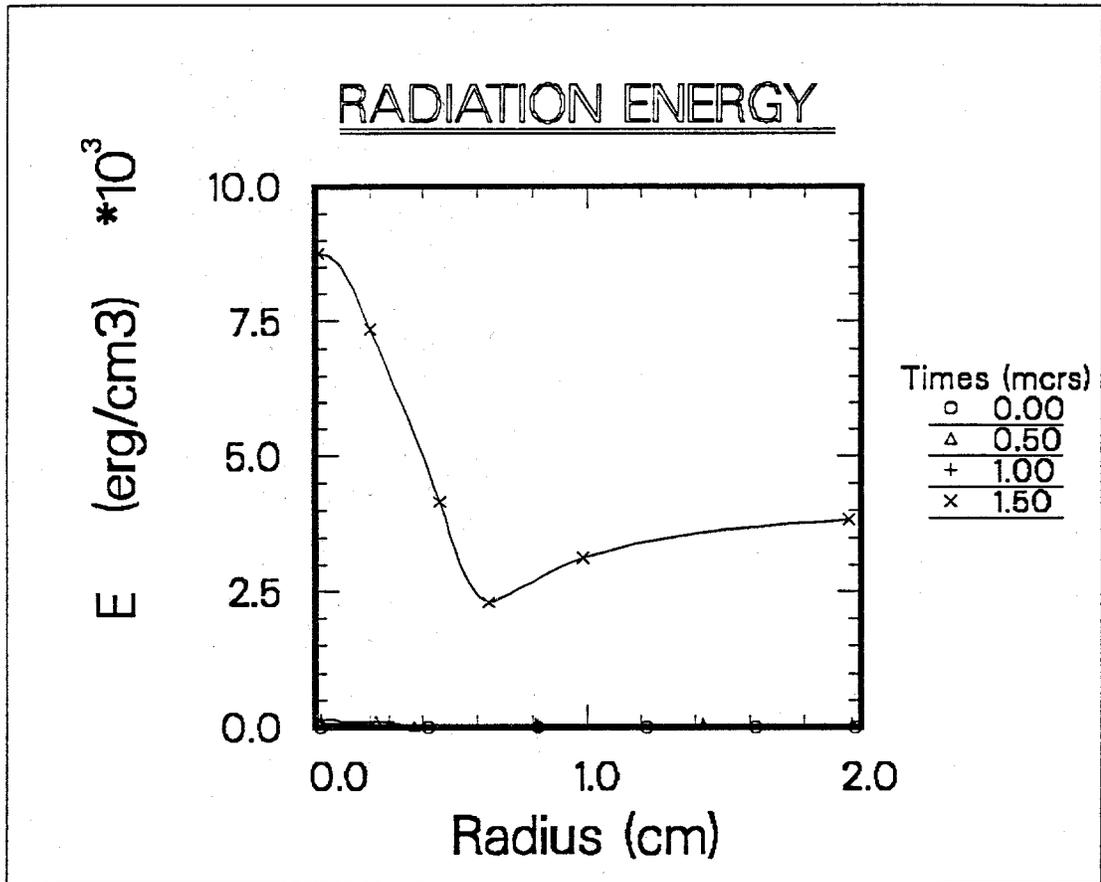


Figure 9. Radiation energy profiles for an argon plasma channel.

References

- [1] G. Yonas, Bull. Am. Phys. Soc. 21 (1972) 1102.
- [2] R.R. Peterson, “CONRAD - A Combined Hydrodynamics Condensation/Vaporization Computer Code,” University of Wisconsin Fusion Technology Institute Report UWFDm - 670, April 1986.
- [3] J.J. Watrous, G.A. Moses, R.R. Peterson, “Z-PINCH– A Multifrequency Radiative Transfer Magnetohydrodynamics Computer Code,” University of Wisconsin Fusion Engineering Program Report UWFDm-584, June 1984.
- [4] R.R. Peterson and G.A. Moses, “MIXERG - An Equation of State and Opacity Computer Code,” *Computer Physics Communications* **28**, 405 (1983).
- [5] O. Yasar, G. Moses, *J. of Comp. Phys.*, in press.
- [6] G.C. Pomraning, *The Equations of Radiation Hydrodynamics*, Pergamon Press, 1973.
- [7] W.F. Hughes, F.J. Young, *The Electromagnetodynamics of Fluids*, John Wiley, N.Y. 1966.
- [8] D. Mihalas and B.W. Mihalas, *Foundations of Radiation Hydrodynamics*, Oxford University Press, 1984.
- [9] E.E. Lewis, W.F. Miller, Jr., *Computational Methods of Neutron Transport*, Wiley-Interscience 1984.
- [10] O. Yasar, “A Computational Model for Z-Pinch Plasma Channels,” Ph.D. Thesis, University of Wisconsin-Madison, December 1989.
- [11] J.F. Thompson, Z.U.A. Warsi, C.W. Mastin, *Numerical Grid Generation*, North-Holland, 1985.

- [12] T.J. Bartel, “Adaptive Mesh Techniques for Unsteady Multi-Gradient Problems in Computational Fluid Dynamics,” Ph.D. Thesis, Univeristy of Wisconsin Fusion Engineering Program Report UWFDM-720, Jan. 1987.
- [13] J.J. MacFarlane, “IONMIX- A Code for Computing the Equation of State and Radiative Properties of LTE and Non-LTE Plasmas,” *Computer Physics Communication* 56 (1989) 259-278.

TEST INPUT

```

1-D R-MHD calc of Z-pinch channels with n-50 (argon)
0.0000e-00 INITIAL TIME
1.5000e-06 END TIME
0.00 restart
2.0 1-CARTESIAN 2-CYLINDRICAL 3-SPHERICAL
1.0 FLUID BOUNDARY CONDITONS IBCL
1.0 IBCR
1.0 IBCT
1.0 IBTB
0.0 TEMPERATURE BOUNDARY CONDITION TBCL
0.0 TBCR
0.0 TBCT
0.0 TBCB
1.41 GAMMA--RATIO OF SPECIFIC HEATS
288.60E-06 ABSOLUTE VISCOSITY MU (gm/cm-s)
2.0e+05 THERMAL CONDUCTIVITY (erg/cm-K-s)
0.0 AMBIENT TEMPERATURE
40.00 GAS MOLECULAR WEIGHT
8.3280e+07 UNIVERSAL GAS CONSTANT (erg/(gm-mole K))
50 NUMBER OF CELLS IN X-REGION 1
0.0 MIN X VALUE
2. MAX X VALUE
1.00 NUMBER OF CELLS IN Y-REGION 1
1.00 DELY FOR Y-REGION 1
00.0 NY2
00.0 DY2
00.0 NY3
00.0 DY3
2.0 ADAPTION CONTROL FLAG
100. INITIALIZATION ADATION LOOPS
0.30 FIRST DERIVATIVE OF ADAPTION FUNCTION 1
0.1 SECOND DERIVATIVE OF ADAPTION FUNCTION 1
0.30 FIRST DERIVATIVE OF ADAPTION FUNCTION 2
0.1 SECOND DERIVATIVE OF ADAPTION FUNCTION 2
0.020 AVERAGE MINIMUM DELTA-X VALUE
99999. DELTA-t (sec) PER FILE RESTART
1.0000e-04 FILE GDMPA
50000. FILE GDMPB
0.5000e-06 FILE GDMPC (CURRENTLY USED)
1.0000e-04 FILE GDMPD
0000000. DUMP FILE B TYPE CONTROL MASK (Psi,Tg,Ts,Rho,P,V,U)
00010111011. DUMP FILE C TYPE CONTROL MASK (F,J,I,X,gv,T,et,rho,p,bf,u)
1.0000E-08 TIME STEP (SEC)
1.0e-1 CFL LIMIT
1.0 ARTIFICIAL VISCOSITY COEFF b1
0.0 IVAR1 - MISC BOUNDARY VALUES
0.0 IVAR2

```

```

03.0      pressure ratio (for shock-tube problem)- not used for z-pinch
10.000E+06 left PRESSURE (dynes/cm2) -- shock tube.
300.      temperature (K) --shock tube.
1.0000E-09 MINIMUM TIME STEP (s)
00.0      bvar7
5.0       initial constatnt delta t
0.5       smoothing coeff -- k
0.1666           -- k-1, k+1
0.083            -- k-2, k+2
75.0       max number of weight iterations
.001       absolute error on mesh initialization
.001       adapti error residual
1.1e-2     axnpos - slope for Temp(x) on x-axis
0.0        bxnpos - init value "
51         ntable - no. of tabular values for Temp(x).
3.5e17     rhoIn - initial particle density
0.5e04     current density amplitude for 1st pulse (stat.Amp/cm2)
02.e-6     time period for 1st pulse (sec)
1.0e05     current density amp. for 2nd pulse (stat.Amp/cm2)
4.0e-6     time period for 2nd pulse (sec)
0.0e-6     time between pulses
1.0e-6     epsilon
1.8e-5     the time of ion-beam shooting(sec)
50.0e-9    time duration of the beam(sec)
9.0e13     beam cur. dens. amp. (statA/cm2)
0.3        some fraction
1.8e-24    beam mass(gr)
2.0e8      beam speed(cm/sec)
1.0        beam charge(Z)
1.0        background charge(Z)
9.11e-28   background mass(gr)
7.85e-1    7.71e-1    7.57e-1    7.43e-1    7.30e-1
7.16e-1    7.03e-1    6.90e-1    6.78e-1    6.65e-1
6.51e-1    6.36e-1    6.18e-1    5.99e-1    5.77e-1
5.53e-1    5.26e-1    4.98e-1    4.66e-1    4.33e-1
3.97e-1    3.60e-1    3.21e-1    2.82e-1    2.43e-1
2.04e-1    1.68e-1    1.33e-1    1.03e-1    7.60e-2
7.00e-2    7.00e-2    7.00e-2    7.00e-2    7.00e-2
7.00e-2

```


0.178E+01	0.100E-29	0.000E+00	0.232E-04	0.400E-01	0.700E-01	0.000E+00
0.182E+01	0.100E-29	0.000E+00	0.232E-04	0.400E-01	0.700E-01	0.000E+00
0.186E+01	0.100E-29	0.000E+00	0.232E-04	0.400E-01	0.700E-01	0.000E+00
0.190E+01	0.100E-29	0.000E+00	0.232E-04	0.400E-01	0.700E-01	0.000E+00
0.194E+01	0.100E-29	0.000E+00	0.232E-04	0.400E-01	0.700E-01	0.000E+00
0.198E+01	0.100E-29	0.000E+00	0.232E-04	0.400E-01	0.700E-01	0.000E+00
0.14994E-05	50					
0.114E+09	0.979E+08	0.302E-03	0.746E-06	0.744E-06	0.212E+15	0.212E+15
0.934E-03	0.320E+08	0.000E+00	0.000E+00			
0.212E+15	0.000E+00	0.596E+08	0.000E+00	0.000E+00		
0.119E-01	-0.316E+05	0.450E+03	0.313E-04	0.238E-01	0.527E+01	0.875E+04
0.328E-01	-0.772E+05	0.131E+04	0.307E-04	0.180E-01	0.528E+01	0.874E+04
0.507E-01	-0.139E+06	0.195E+04	0.316E-04	0.177E-01	0.527E+01	0.871E+04
0.683E-01	-0.187E+06	0.259E+04	0.320E-04	0.175E-01	0.524E+01	0.864E+04
0.858E-01	-0.241E+06	0.323E+04	0.332E-04	0.175E-01	0.520E+01	0.855E+04
0.103E+00	-0.291E+06	0.385E+04	0.346E-04	0.175E-01	0.515E+01	0.842E+04
0.121E+00	-0.334E+06	0.447E+04	0.359E-04	0.176E-01	0.509E+01	0.827E+04
0.139E+00	-0.370E+06	0.507E+04	0.369E-04	0.179E-01	0.503E+01	0.810E+04
0.157E+00	-0.402E+06	0.569E+04	0.371E-04	0.190E-01	0.496E+01	0.789E+04
0.178E+00	-0.437E+06	0.636E+04	0.359E-04	0.222E-01	0.488E+01	0.764E+04
0.203E+00	-0.484E+06	0.714E+04	0.330E-04	0.280E-01	0.480E+01	0.735E+04
0.232E+00	-0.534E+06	0.802E+04	0.295E-04	0.313E-01	0.472E+01	0.702E+04
0.263E+00	-0.577E+06	0.890E+04	0.262E-04	0.306E-01	0.465E+01	0.666E+04
0.293E+00	-0.605E+06	0.970E+04	0.233E-04	0.290E-01	0.457E+01	0.631E+04
0.322E+00	-0.607E+06	0.104E+05	0.214E-04	0.279E-01	0.449E+01	0.598E+04
0.349E+00	-0.576E+06	0.111E+05	0.207E-04	0.268E-01	0.441E+01	0.565E+04
0.375E+00	-0.519E+06	0.117E+05	0.209E-04	0.252E-01	0.433E+01	0.534E+04
0.399E+00	-0.447E+06	0.124E+05	0.219E-04	0.236E-01	0.426E+01	0.504E+04
0.422E+00	-0.375E+06	0.130E+05	0.231E-04	0.222E-01	0.418E+01	0.474E+04
0.444E+00	-0.311E+06	0.135E+05	0.243E-04	0.208E-01	0.410E+01	0.444E+04
0.464E+00	-0.259E+06	0.141E+05	0.251E-04	0.192E-01	0.401E+01	0.416E+04
0.482E+00	-0.220E+06	0.146E+05	0.255E-04	0.181E-01	0.390E+01	0.387E+04
0.500E+00	-0.195E+06	0.151E+05	0.256E-04	0.177E-01	0.364E+01	0.355E+04
0.518E+00	-0.185E+06	0.156E+05	0.251E-04	0.175E-01	0.330E+01	0.325E+04
0.535E+00	-0.181E+06	0.161E+05	0.245E-04	0.175E-01	0.299E+01	0.302E+04
0.553E+00	-0.172E+06	0.165E+05	0.239E-04	0.176E-01	0.277E+01	0.284E+04
0.570E+00	-0.153E+06	0.169E+05	0.236E-04	0.176E-01	0.261E+01	0.267E+04
0.588E+00	-0.127E+06	0.173E+05	0.237E-04	0.176E-01	0.246E+01	0.253E+04
0.606E+00	-0.101E+06	0.177E+05	0.240E-04	0.175E-01	0.222E+01	0.243E+04
0.623E+00	-0.843E+05	0.181E+05	0.244E-04	0.176E-01	0.182E+01	0.234E+04
0.641E+00	-0.721E+05	0.184E+05	0.246E-04	0.179E-01	0.154E+01	0.231E+04
0.659E+00	-0.567E+05	0.186E+05	0.248E-04	0.190E-01	0.133E+01	0.233E+04
0.679E+00	-0.387E+05	0.188E+05	0.250E-04	0.208E-01	0.117E+01	0.237E+04
0.700E+00	-0.191E+05	0.189E+05	0.251E-04	0.217E-01	0.105E+01	0.243E+04
0.721E+00	0.631E+03	0.190E+05	0.253E-04	0.202E-01	0.942E+00	0.248E+04
0.742E+00	0.124E+05	0.189E+05	0.254E-04	0.217E-01	0.718E+00	0.254E+04
0.767E+00	0.152E+05	0.184E+05	0.254E-04	0.285E-01	0.557E+00	0.261E+04
0.801E+00	0.143E+05	0.176E+05	0.254E-04	0.376E-01	0.417E+00	0.269E+04
0.844E+00	0.125E+05	0.167E+05	0.253E-04	0.494E-01	0.184E+00	0.281E+04
0.904E+00	0.107E+05	0.156E+05	0.252E-04	0.707E-01	0.190E+00	0.297E+04
0.984E+00	0.966E+04	0.144E+05	0.250E-04	0.882E-01	0.137E+00	0.313E+04
0.108E+01	0.789E+04	0.131E+05	0.247E-04	0.101E+00	0.106E+00	0.327E+04
0.118E+01	0.586E+04	0.119E+05	0.243E-04	0.107E+00	0.911E-01	0.340E+04
0.129E+01	0.388E+04	0.109E+05	0.240E-04	0.110E+00	0.828E-01	0.350E+04
0.140E+01	0.223E+04	0.101E+05	0.237E-04	0.112E+00	0.764E-01	0.358E+04
0.151E+01	0.108E+04	0.933E+04	0.234E-04	0.113E+00	0.726E-01	0.365E+04
0.163E+01	0.426E+03	0.868E+04	0.233E-04	0.114E+00	0.710E-01	0.370E+04
0.174E+01	0.140E+03	0.811E+04	0.233E-04	0.115E+00	0.704E-01	0.375E+04
0.186E+01	0.788E+02	0.761E+04	0.232E-04	0.115E+00	0.702E-01	0.379E+04
0.196E+01	0.942E+02	0.722E+04	0.232E-04	0.850E-01	0.702E-01	0.383E+04