Published in *Laser and Particle Beams* (1998), vol. 16, no. 3, pp. 405-430

Finite-element simulation code for high-power magnetohydrodynamics

By STANLEY HUMPHRIES, JR.* AND CARL EKDAHL**

* Field Precision, Albuquerque, NM 87192 USA
** Los Alamos National Laboratory, Los Alamos NM 87545 USA

(Received 16 December 1997; Accepted 30 March 1998)

We describe the mathematical basis and organization of Crunch, a 1Dl shock-hydrodynamics code to analyze pulsed power experiments at Los Alamos National Laboratory. The program uses finite-element methods that preserve stability during material collisions and shock convergence on axis. It handles coupled calculations of non-linear magnetic diffusion to simulate imploding liners. These calculations may be driven by multiple current waveforms or a self-consistent current variation derived from a pulsed-power generator model. Crunch incorporates elastic material contributions and calculates element break and melt points. The primary goal in program development was effective use by experimentalists. Crunch is controlled by a streamlined script language and runs on standard personal computers. An interactive graphical post-processor expedites analysis of results. To support the program we have assembled data resources in machine-independent format including Sesame equation-of-state tables, a material strength library and a library of temperature-dependent conductivities.

1. Introduction

This paper describes the theoretical basis and organization features of Crunch, a 1D finite-element code for material interactions at high temperature and pressure. The program simulates shock hydrodynamics in planar, cylindrical, or spherical systems. Equation-of-state data are determined from the Sesame Library (Bennet, *et.al*. 1978, Lyon & Johnson 1992) and elastic properties of materials are based on the Steinberg models (Steinberg 1996). Crunch handles coupled calculations with shocks driven by pulsed magnetic fields. The motivation for this function is investigation of imploding metal liners (Alikhanov & Konkashbaev 1973, Sherwood *et.al* 1980, Turchi *et.al* 1980, Chernychev *et.al* 1987, Parker 1993, Hockaday *et.al* 1998), a component of the High-energy-density Physics Program (Parsons *et.al* 1997, Ekdahl 1998) at Los Alamos National Laboratory (LANL). In this application, shown in figure 1, an intense pulsed axial current drives a cylindrical shell to a focus on axis to generate high temperatures and pressures in materials. In liner simulations Crunch can determine self-consistent drive current using an external circuit model for high-power generators. The program has potential application to a variety of pulsed-power devices including imploding wire arrays, magnetically accelerated

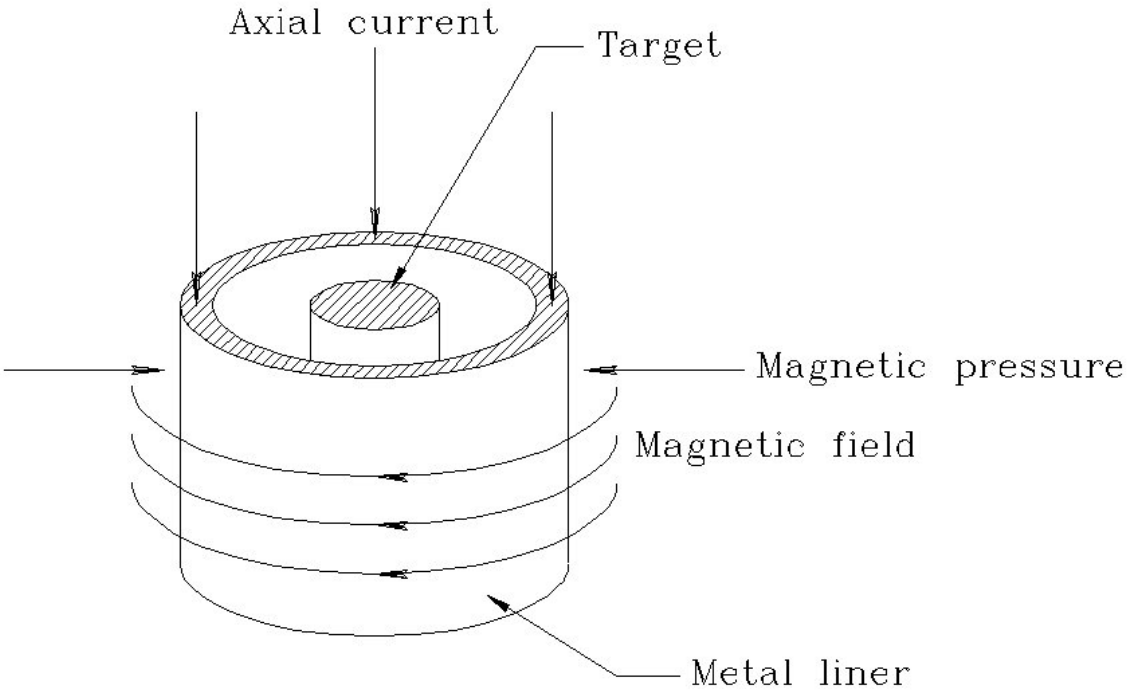


Figure 1. Imploding liner experimental geometry. A pulsed axial current creates a toroidal magnetic field which drives the liner radially inward. In some experiments, the liner collides with a target to create material states of high temperature and pressure.

projectiles, and compressive flux generators. The present version of the code does not include radiation contributions to pressure balance and heat conduction; therefore, the range of validity is limited to temperatures of 10^6 °K and below.

Although LANL has substantial resources for hydrodynamic simulations, there were several reasons to develop Crunch:

- The program incorporates features that support the range of experiments anticipated in the High-energy-density Physics Program.
- The finite-element approach exhibits good stability for collisions between objects and shock convergence on axis.
- Crunch can be used autonomously by experimental scientists. The physical basis of the program and the command set are fully documented.
- The program and supporting data are highly portable. The standard version runs on IBM-standard personal computers with support for common graphics and hardcopy devices.

Two principles guided development to ensure that code results would be independent of the code developer.

• All material data are accessible to the user. Information on the equation-of-state, electrical conductivity, and material strength are stored in external ASCII libraries. Material models can be corrected or updated without changing the program.
• Crunch has no adjustable internal parameters for material models. Results are absolute consequences of the information in the external libraries. Any extrapolation beyond the range of input data terminates a run.

The following sections document mathematical methods applied in Crunch and data resources we have assembled for personal computers. Section 2 covers the hydrodynamic difference equations with emphasis on cylindrical systems. The relationships follow from integral expressions of conservation of mass, momentum and energy over elements. Here, elements are volumes with a unique material identity that may move through the solution space. Section 3 describes equation-of-state information to complete the hydrodynamic equations. We have converted the Sesame library to a language- and machine-independent form. Strength models for elastic materials are discussed in Section 4. Section 5 covers numerical methods to calculate distributions of pulsed magnetic fields in a moving medium with temperature-dependent electrical conductivity. Section 6 reviews the conductivity library we assembled to cover the experimental range of interest for imploding-liner simulations.

In magnetic simulations with a single drive current Crunch includes a coupled circuit to represent pulsed power drivers. Section 7 describes the versatile model where all circuit components may have user-specified time variations. Section 8 reviews organizational aspects of Crunch, including user interfaces and options for multiple drive currents. Section 9 illustrates some capabilities of the code with two examples. The first, the Sedov blast wave problem, is a standard benchmark test for hydrodynamic codes. The second, an exploding wire simulation illustrates a coupled calculation with non-linear magnetic diffusion.

2. Integral hydrodynamic equations

Numerical hydrodynamic equations are difference representations of conservation of mass, momentum and energy. The Crunch equations are referenced to elements. Figure 2 shows element divisions for a one-dimensional simulation. Depending on symmetry, the slices represent thin plates, cylindrical shells, or spherical shells. The two sets of indices apply to elements and element boundaries. We shall denote boundary quantities with upper case letters and element quantities with lower case. For example, Element i with average radius r_i has boundaries at R_{i-1} and R_i. Table 1 lists symbols used in the paper. Elements retain their material identity as they move and change size during the calculation. The method is similar to Lagrangian finite-difference calculations (see, *e.g.*, Potter 1973). An inherent limitation of the approach is the difficulty of modeling processes like mixing and spallation. On the other hand, the element-centered approach has two advantages:

• Automatic zone refinement for compressional phenomena like shocks.
• Ability to model explosive processes where the solution volume size may change by

orders of magnitude.

Table 1
Symbols and units

Element quantities (Element *i* at time t^n)		
Symbol	Quantity	Units
r_i^n	Center of mass radius	m
m_i	Total mass	kg
ρ_i^n	Mass density	kg/m^3
p_i^n	Pressure	Pa
w_i^n	Artificial viscosity	Pa
j_i^n	Axial current density	A/m^2
u_i^n	Internal energy	J/kg
σ_i^n	Electrical conductivity	mhos/m
τ_i^n	Temperature	°K
ϵ_{rri}^n, $\epsilon_{\theta\theta i}^n$	Normal strains	(dimensionless)
s_{rri}^n, $s_{\theta\theta i}^n$	Normal stresses	Pa
Boundary quantities (Outside of Element *i* at time t^n)		
Symbol	Quantity	Units
R_i^n	Radius	m
$V_i^{n+½}$	Radial velocity	m/s
B_i^n	Toroidal magnetic field	tesla
Other quantities		
Symbol	Quantity	Units
C	Coefficient for artificial viscosity	(dimensionless)
Δt	Time step	s
E	Young's modulus	Pa
E_o	Young's modulus at ambient conditions	Pa

Table 1 Symbols and units		
ν	Poisson's ratio	(dimensionless)
η	Material compression ratio	(dimensionless)
Y	Yield stress	Pa
Y_o	Yield stress at Hugoniot elastic limit	Pa
A	Pressure dependence of Young's modulus	Pa^{-1}
B	Temperature dependence of Young's modulus	$°K^{-1}$
β	Work hardening parameter	(dimensionless)
N	Work hardening parameter	(dimensionless)
ϵ_i	Initial plastic strain	(dimensionless)
Y_{max}	Work hardening maximum stress	Pa
T_{m0}	Melt temperature at standard density	°K
γ_o	Initial value of Gruneisen's gamma	(dimensionless)
α	Coefficient of the volume dependence of γ	(dimensionless)
R_{min}	Inner boundary of magnetic region	m
R_{max}	Outer boundary of magnetic region	m
V_{max}	Velocity of outer boundary of magnetically-active region	m/s
I	Liner drive current	A
E_z	Axial electric field	V/m
$V_1,...,V_5$	Circuit model voltages	V
i_1, i_2	Circuit model currents	A
R_g, R_f, R_l	Circuit model resistances	Ω
L_g, L_f, L_l	Circuit model inductances	H
C_g	Circuit model capacitance	F
$\mathbf{C}^n$	Circuit vector at time t^n	---
R_c	Liner resistance	Ω

Table 1 Symbols and units		
L_c	Liner inductance	H
L_t	Inductance of power connections	H
h	Liner height	m
A, **B**	Circuit model component matrices	---

In this paper we shall concentrate on cylindrical systems. The extension to planar and spherical systems is straightforward. Conservation of mass implies that element masses do not change during the simulation. Consider an element with initial boundaries R_{oi-1} and R_{oi} and initial density ρ_{oi}. The mass is given by

$$m_i = \rho_{oi} \pi (R_{oi}^2 - R_{oi-1}^2). \quad \textbf{(1)}$$

The boundaries move in response to forces. The density at any time is related to the boundary positions by

$$\rho_i = \frac{m_i}{\pi (R_i^2 - R_{i-1}^2)} . \quad \textbf{(2)}$$

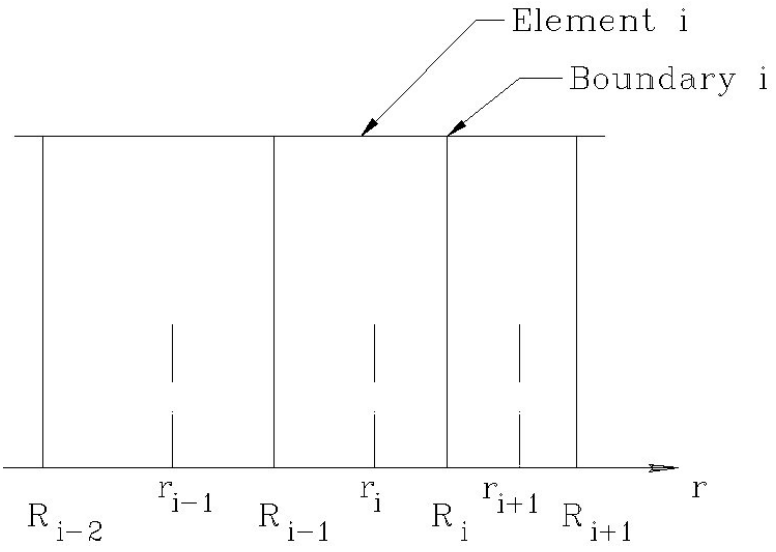


Figure 2. Index conventions for one-dimensional finite-element simulations. In cylindrical geometry, the elements are cylindrical shells.

Note that equations (1) and (2) do not incorporate approximations based on small element width. This feature avoids numerical problems when elements compress to cylindrical or spherical axes. Furthermore, the model allows the use of large elements. The average element radius corresponds to the center-of-mass coordinate. Assuming a uniform density, the average radius of a cylindrical element is related to the boundary radii by

$$r_i = \sqrt{\frac{R_i^2 + R_{i-1}^2}{2}} \ . \tag{3}$$

We express conservation of momentum as an equation of motion for element boundaries. The object is to find the boundary velocities

$$V_i = \frac{dR_i}{dt} \ . \tag{4}$$

The time rate-of-change of momentum at boundary *i* equals the time derivative of velocity times half the masses of adjacent elements,

$$\left(\frac{m_{i+1} + m_i}{2} \right) \frac{dV_i}{dt} \ . \tag{5}$$

The force on the boundary is the sum of forces from adjacent elements. Summing pressure and magnetic forces gives the equation of motion

$$\left(\frac{m_{i+1} + m_i}{2} \right) \frac{dV_i}{dt} = (-p_{i+1} - w_{i+1} + p_i + w_i)\ 2\pi R_i \ - \ B_i \left(\frac{j_{i+1} + j_i}{2} \right) \ \pi \ (r_{i+1}^2 - r_i^2) \ . \tag{6}$$

The new element quantities in equaton (6) are the pressure p_i, artificial viscosity w_i, and current density j_i. Current density flows in the axial direction for the geometry of Fig. 1. As explained in Sect. 5, the toroidal magnetic field B_i is a boundary quantity. The magnetic force equals the integral of the force density over half the volumes of adjacent elements. The pressure force equals the difference in pressure in the adjacent elements multiplied by the cylindrical area at the boundary. The artificial viscosity term damps non-physical oscillations at shock fronts. The physical rationale for artificial viscosity and its inclusion in the hydrodynamic equations are covered in Ref. 12. We used an adaptation of the von Neumann-Richtmyer form (Neumann & Richtmyer 1950, Richtmyer & Morton 1967) for finite-difference solutions,

$$w = -C\rho\Delta^2 \left| \frac{\partial v}{\partial x} \right| \frac{\partial v}{\partial x}. \tag{7}$$

In equation (7) *C* is an adjustable parameter to spread the shock over several mesh divisions, Δ is the mesh scale length, and $\partial v/\partial x$ is the spatial derivative of velocity.

Crunch advances hydrodynamic quantities using the standard time-centered leap-frog method[12]. The boundary velocities V_i are defined at half time steps and all other quantities apply at integral steps. Throughout this paper the superscript *n* denotes the time step, so that $t^{n+½} = t^n + \Delta t/2$. Replacing time derivatives in equation (6) with time-centered difference operators gives an equation to advance the boundary velocity,

$$V_i^{n+½} = V_i^{n-½} - \left(\frac{\Delta t}{m_{i+1}+m_i} \right) \times \left[(p_{i+1}^n + w_{i+1}^n - p_i^n - w_i^n)\ 4\pi R_i^n + B_i^n\ (j_{i+1}^n + j_i^n)\ \pi\ (r_{i+1}^{n\,2} + r_i^{n2}) \right] . \quad \textbf{(8)}$$

Equation (7) has the following difference representation,

$$w_i^n = -C\rho_i^n | V_i^{n-½} - V_{i-1}^{n-½} |\ (V_i^{n-½} - V_{i-1}^{n-½}) , \quad \textbf{(9)}$$

where *C* is a constant in the range 1-10. Given the modified velocities, the next step is to advance the boundary radii to the next integral time step,

$$R_i^{n+1} = R_i^n + V_i^{n+½}\Delta t. \quad \textbf{(10)}$$

New element densities and average radii can be determined from R_i^{n+1} using equations (2) and (3).

The internal energy u_i is an element property equal to the total energy of element *i* divided by m_i. At present Crunch does not model changes of u_i resulting from convection or radiation transport. Although simple to code, we omitted thermal conductivity contributions for three reasons:

- Thermal conduction in solids and liquids is negligible compared to energy transport by shocks.
- Thermal transport coefficients are not well known at high temperature and pressure.
- Energy transport in gases and plasmas is probably dominated by convection.

Under the limiting assumptions changes of internal energy in hydrodynamic calculations result from work performed by pressure, artificial viscosity and elastic stress. In magnetic problems ohmic heating also contributes. The work performed by pressure and artificial viscosity on element *i* in a time step is $-(p_i+w_i)\Delta V_i$, where ΔV_i is the change in element volume. The time rate of change of energy from resistive heating in an element with electrical conductivity σ_i equals j_i^2/σ_i times the element volume. With these contributions, the equation to advance internal energy

is

$$u_i^{n+1} = u_i^n + \frac{1}{m_i} \times$$

$$\left[-\frac{(p_i^{n+1}+p_i^n)}{2} \pi (R_i^{n+1\,2}-R_i^{n\,2}-R_{i-1}^{n+1\,2}+R_{i-1}^{n\,2}) + \frac{j_i \Delta t \pi (R_i^{n\,2}-R_{i-1}^{n\,2})}{\sigma_i} \right]. \quad \textbf{(11)}$$

The first term in brackets is a time-centered expression involving the advanced value of pressure. The pressure is estimated by the two-step process described in the next paragraph. It is not necessary to apply more complex time-centered expressions for the second term. In cases of interest the time scale for resistive heating is much longer than that for shock heating.

To close the set of equations we must find the new element pressures p_i^{n+1} corresponding to modified values of density and internal energy, ρ_i^{n+1} and u_i^{n+1}. The values are determined from the Sesame equation-of-state library discussed in the next section. The library consists of two-dimensional tables of pressure and internal energy as functions of density ρ and temperature τ: $p(\rho,\tau)$ and $u(\rho,\tau)$. With known values of density and internal energy the temperature τ can be determined with an inverse interpolation. We use a modified two-step method[12] to advance the pressure and preserve time-centering in equation (12). The advanced internal energy $u_i^{n+1'}$ is first estimated from equation (12) using only p_i^n. Sesame interpolations give estimates of the advanced temperature $\tau_i^{n+1'}$ and pressure $p_i^{n+1'}$. The quantity $(p_i^n+p_i^{n+1'}/)/2$ is then used in equation (12) to yield an improved value u_i^{n+1}. The interpolations are repeated to find τ_i^{n+1} and p_i^{n+1}.

3. Equation-of-state information

The Sesame Library (maintained by Los Alamos National Laboratory contains computed values of thermodynamic properties at high temperature and pressure for 131 materials. The standard format of the library is not convenient for personal computers; the data are contained in a single large binary file. It is necessary to apply a set of FORTRAN subroutines supplied by the laboratory to access information (Abdallah *et.al.*1980). We converted the Sesame data to a portable format that was both language- and machine-independent. We wrote programs to dissect the library and to create individual files ASCII for each material. The data are accessible with an editor, and it is easy to create new tables. With data compression the library fits on three 1.4 MB floppy disks.

The files in the PC Sesame Library have descriptive names of the form ALUM3713.SES. The first four characters of the prefix denote the type of material, while the next four give the LANL table numbers. Each material data file contains a standard header and four tables. The header provides information to load values into spreadsheets or programs. The first table contains the discrete values of density for tabulation of the pressure and internal energy; ρ_1, ρ_2, ... $\rho(N_\rho)$. The

second table lists temperature values: τ_1, τ_2, ... , $\tau_{N\tau}$. The third table gives corresponding pressure values with density indices as the outer loop: p(1,1), p(1,2), ..., p(1,N_τ), p(2,1), ... , p(i,j), ... , p(N_ρ,N_τ-1), p(N_ρ,N_T). The ordering differs from the standard Sesame arrangement and allows more efficient interpolation. The fourth table contains values of internal energy: u(1,1), u(1,2), ..., u(1,N_τ), u(2,1), ... , u(i,j), ... , u(N_ρ,N_τ-1), u(N_ρ,N_T). For backward compatibility, we maintained the practical units of the Los Alamos tables: ρ in gm/cm^3 (1000 kg/m^3), τ in °K, p in GPa (10^9 Pa), and u in MJ/kg (10^6 J/kg).

We also developed a structured FORTRAN software unit, OPEN_SES, that handles operations on the Sesame tables. All storage and manipulation of data is internal to the unit, with communication via subroutine and function calls. The functionality of OPEN_SES does not require data structure definition in the calling program. For example, the statement

```
CALL SesLoad(MatFile)
```

transfers material data from a Sesame file to memory storage controlled by the software unit. Here, the quantity *MatFile* is the 8-character PC Sesame file prefix. A subsequent call to the integer function `NCode()` returns the operation status. Related functions include `NTable()` (number of tables successfully loaded in memory) and `NMem()` (available memory bytes for table storage). The subroutine

```
SesConvFactors(RhoConv,TempConv,PressConv,UConv,VelConv)
```

sets up unit conversions for all input and output data. For example, the statement

```
CALL SesConvFactors(1.0E3,1.0,1.0E9,1.0E6.1.0)
```

changes hydrodynamic quantities to SI units. OPEN_SES assigns an integer material number *MatNo* to tables in the order they are loaded. The function to return the stored pressure value for density ρ_I and temperature τ_J has the name and pass parameters `PIJ(MatNo,I,J)`.

The critical routines for hydrodynamic codes are those that interpolate and invert quantities. The function `P(MatNo,Rho,Temp)` returns the pressure in the current units for material *MatNo* at density *Rho* and temperature *Temp*. OPEN_SES routines use polynomial interpolation with user-specified accuracy up to sixth order. Interpolation routines return an error code through `NCode()` for out-or-range values or interpolation errors. The inversion function `Temp(MatNo,Rho,Press)` gives temperature as a function of density and pressure using the following method. The routine identifies the pressure table row with density ρ_I closest to *Rho* and scans temperature columns for pressure entries that bracket *Press*. A one-dimensional interpolation gives the temperature $\tau(\rho_I$,Press). The procedure is repeated on nearby rows to build a table of values: ... $\tau(\rho_{I-1}$,Press), $\tau(\rho_I$,Press), $\tau(\rho_{I+1}$,Press),..., The number of entries depends on the interpolation order. A final interpolation gives temperature at the target pressure. The routine is fast enough so that it is unnecessary to build inverted tables T(ρ,p) in memory. Other OPEN_SES routines calculate sound speed and points on the shock Hugoniot curve (see, *e.g.*, Courant & Friedrichs 1991).

4. Material strength models

This section reviews elastic material models in Crunch. The *normal strain* in a medium is a dimensionless quantity that equals the change in the length of an element divided by its original length, while *normal stress* is the force per unit area that must be applied to a solid body to produce a strain. *Shear stresses* that give angular distortions of the element do not appear in 1D calculations. In cylindrical simulations, there are two possible normal strains, ϵ_{rr} and $\epsilon_{\theta\theta}$, caused by the two stresses s_{rr} and $s_{\theta\theta}$. Figure 3 illustrates the relationship between strains and stresses in a cylindrical system. As element moves only in the r direction but may have stresses and strains in θ because of radial compression or expansion.

We can identify two limiting cases where the system is unconstrained in z (*plane stress*) or clamped (*plane strain*). Following Fung (1965) and Huddlestone (1961), the stress-strain relationships for the two cases are

$$\begin{aligned} \epsilon_{rr} &= \frac{1}{E}\left(s_{rr} - \nu s_{\theta\theta} \right) , \\ \epsilon_{\theta\theta} &= \frac{1}{E}\left(-\nu s_{rr} + s_{\theta\theta} \right) . \end{aligned} \tag{12}$$

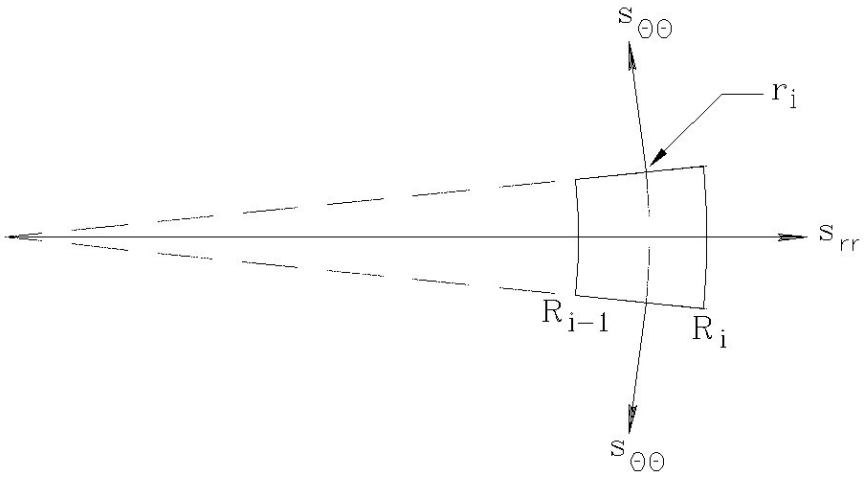
or

$$\begin{aligned} \epsilon_{rr} &= \frac{1}{E}\left[(1-\nu^2) s_{rr} - \nu(1+\nu) s_{\theta\theta} \right] , \\ \epsilon_{\theta\theta} &= \frac{1}{E}\left[-\nu(1+\nu) s_{rr} + (1-\nu^2) s_{\theta\theta} \right] . \end{aligned} \tag{13}$$

In equations (12) and (13), the quantity E is Young's modulus and υ is Poisson's ratio (which generally has a value near 0.25).

Concentrating on the constrained case, the relationships in equation (20) gives the stresses as

$$\begin{aligned} s_{rr} &= E\left[\frac{(1-\nu)\epsilon_{rr} + \nu\epsilon_{\theta\theta}}{(1+\nu)(1-2\nu)} \right] , \\ s_{\theta\theta} &= E\left[\frac{\nu\epsilon_{rr} + (1-\nu)\epsilon_{\theta\theta}}{(1+\nu)(1-2\nu)} \right] . \end{aligned} \tag{14}$$



Radial motion of an element results in an azimuthal strain. Noting that the strain equals the change

Figure 3. Stress on an element in a cylindrical system with applied in the *r* direction only, showing the origin of the hoop stress.

in length along θ divided by the original length and assuming continuity of the medium, the azimuthal strain in element *i* at time t^n is

$$\epsilon^n_{\theta\theta i} = \frac{r_i^n - r_i^o}{r_i^o} . \qquad \textbf{(15)}$$

where the quantity r_i^n is the average element radius [equation (3)] and the superscript *o* refers to quantities at the initial time. The radial strain is given by an expression similar to that of the planar case:

$$\epsilon^n_{rri} = \frac{[\ R_i^n - R_{i-1}^n\ ] - [\ R_i^o - R_{i-1}^o\ ]}{[\ R_i^o - R_{i-1}^o\ ]}. \qquad \textbf{(16)}$$

The quantities on the right-hand side of equation (16) are boundary radii. Inspection of figure 3 shows that azimuthal stress exerts a component of radial force, the *hoop stress*. For a given value of $s_{\theta\theta i}$, the radial force per unit area at the center-of-mass of element *i* from material expansion or compression is

$$-\frac{s^n_{\theta\theta i}\ [\ R_i^n - R_{i-1}^n\ ]}{r_i^n} , \qquad \textbf{(17)}$$

It is relatively easy to incorporate elastic forces in planar Crunch simulations. The stresses appear as extra contributions in the boundary equation of motion. For example, the force per unit area acting at boundary *i* is

$$p_i^n - p_{i+1}^n + w_i^n - w_{i+1}^n - s_{xxi}^n + s_{xxi+1}^n \qquad \textbf{(18)}$$

Equation (18) holds for positive values of strain (extension) because the fluid equation-of-state contributes no force. Equivalently, pressure entries in the Sesame tables are always positive. On the other hand, the tables contribute elastic forces in elements with negative strain (compression). In this case there is a redundancy if stresses are included. The following prescription holds for a planar numerical model: use $(p_i^n + w_i^n)$ for the force per area in an elastic element when $\epsilon_i^n < 0$ and use $(p_i^n + w_i^n - \sigma_i^n)$ when $\epsilon_i^n < 0$ if the element is unbroken. An element *breaks* if a positive value of s_i exceeds the yield stress or if the temperature reaches the melt value. The adjusted force is used in both the equation of motion and internal energy equation.

Material strength contributions to cylindrical problems are more involved. Consider radial expansion of a cylindrical element. The azimuthal strain and stress are positive, resulting in an inward radial hoop force. Conservation of volume gives a radial compression (negative strain); therefore, the direct radial force is given by equation-of-state contributions. The radial average of this force over the elements of a cylindrical shell is approximately zero, while the hoop stress gives a cumulative inward force. Next, suppose the cylindrical shell is pushed inward. There is compression in the azimuthal direction and expansion in the radial direction. The elastic forces resist the radial expansion, giving a net outward force associated with the azimuthal strain. An element breaks if the radial stress exceeds the yield value. Following these considerations, the following conventions are applied in Crunch. In unbroken elements only positive radial stresses are included in the equation of motion while the hoop force is included for positive and negative values of the azimuthal stress. Elastic forces are represented by an extra term in equation (6),

$$2(r_{i+1}+r_i)\left[-s_{rri}+s_{rri+1}-\frac{s_{\theta\theta i}\,[\,R_i-R_{i-1}\,]}{r_i}-\frac{s_{\theta\theta i+1}\,[\,R_{i+1}-R_i\,]}{r_{i+1}}\right]. \qquad \textbf{(19)}$$

The radial stress terms are included only if they have a positive value.

We used the Steinberg (1996) parametric model in Crunch for the elastic properties of materials. The following equation gives Young's modulus as a function of density, pressure and temperature:

$$E(\rho,p,\tau) = E_o\left[\,1 + A\,\frac{p}{\eta^{1/3}} - B\,(\tau-300)\,\right]. \qquad \textbf{(20)}$$

In equation (20) E_o (pascals) is the value of Young's modulus at ambient conditions, p (pascals) is the pressure, τ (°K) is the temperature, and η is the compression,

$$\eta = \frac{\rho}{\rho_o} \ . \tag{21}$$

The yield stress is proportional to the relative value of Young's modulus multiplied by a function of the element strain that represents work hardening:

$$Y = Y_o f(\epsilon) \frac{E}{E_o} \ . \tag{22}$$

The function is given by

$$Y_o f(\epsilon) = Y_o \left[1 + \beta (\epsilon + \epsilon_i)^n \right] \le Y_{max} \ . \tag{23}$$

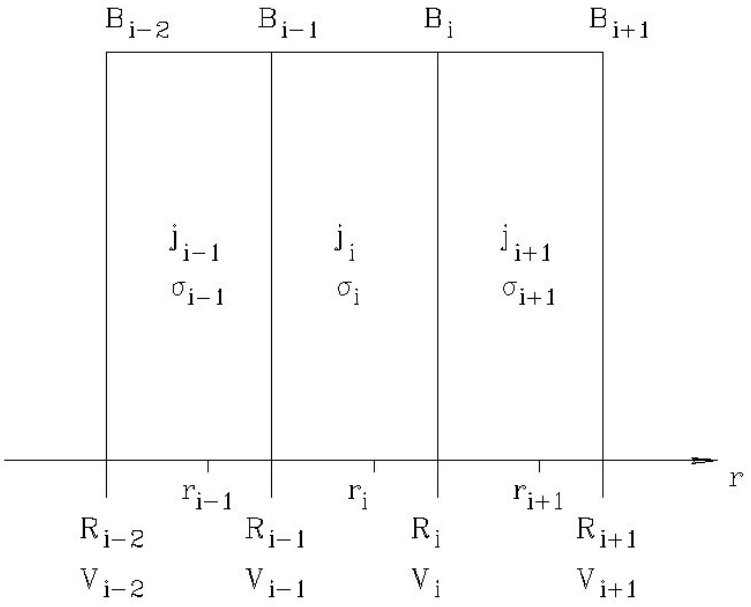
The symbols in equations (22) and (23) represent the following quantities: Y_o (pascals) is the yield strength at the Hugoniot elastic limit, Y_{max} (pascals) is the work hardening maximum, and β and n are work hardening parameters. Steinberg (1996) also gives an expression for the melt point of a material in compression

$$\tau_m(\eta) = \tau_{mo} \exp\left[2a \left(1 - \frac{1}{\eta} \right) \right] \eta^{2(\gamma_o - a - 0.333)} \ . \tag{24}$$

In equation (24) τ_{mo} is the melt temperature at ambient conditions, γ_o and a are model parameters, and η is given by equation (28). We have created an ASCII library that contains the material parameters in SI units.

5. Magnetic diffusion

This section reviews numerical methods in Crunch to treat magnetic diffusion in cylindrical systems like that of figure 1. The solution volume may contain layers of different materials including conductors and insulators. All layers participate in the hydrodynamic solution but only the conductors participate in the magnetic field solution. To begin, suppose that there is a single specified drive current $I(t)$ and that the conducting layers are contiguous. We shall call the set of conductors the *magnetically-active region*. The goal is to find the toroidal magnetic field distribution over the region as a function of time. Figure 4 shows index conventions for the magnetic field calculation. Element quantities include r_i^n, j_i^n (axial current density in A/m^2) and σ_i^n (electrical conductivity in mhos/m). The temperature-dependent element conductivity may change with time. Boundary quantities include R_i^n and $V_i^{n+½}$. The quantity R_{min} is the radius of the



inner

Figure 4. Index conventions for quantities in the cylindrical magnetic diffusion calculation.

edge of the magnetically active region and R_{max} is the radius of the outer edge. Because the element quantity j_z is proportional to the spatial derivative of toroidal magnetic field, we take B_i^n (tesla) as a boundary quantity. This association is consistent with the field boundary conditions:

$$B(R_{min},t) = 0,$$

$$B(R_{max},t) = \frac{\mu_o I(t)}{2\pi R_{max}} . \qquad (25)$$

The Crunch difference equations follow from the differential equation of magnetic diffusion in a moving medium (see, *e.g.*, Humphries 1997). Faraday's law gives the difference in axial electric field at radii r_o and r in the active region as,

$$E(r,t) = E(r_o,t) - \frac{\partial}{\partial t}\int_{r_o}^{r} B(r',t)dr' =$$

$$E(r_o,t) - \int_{r_o}^{r} dr' \frac{\partial B(r',t)}{\partial t} - B(r_o,t)\frac{\partial r_o}{\partial t} + B(r,t)\frac{\partial r}{\partial t} . \quad \textbf{(26)}$$

The other defining relationships are Ohm's law,

$$j(r,t) = \sigma(r,t)\, E(r,t), \quad \textbf{(27)}$$

and Ampere's law,

$$B(r,t) = \frac{\mu_o}{2\pi r}\int_{R_{min}}^{r} 2\pi r' dr' j(r',t) . \quad \textbf{(28)}$$

Taking derivatives and combining equations (24)-(28) gives,

$$\frac{\partial B}{\partial t} = \frac{1}{\sigma\mu_o}\frac{1}{r}\frac{\partial}{\partial r}(rB) - v\frac{\partial B}{\partial r} - B\frac{\partial v}{\partial r} . \quad \textbf{(29)}$$

The difference equations follow from equation (29). For spatial differencing we preserve the element viewpoint by treating material properties as element quantities. For time differencing we apply the method of Dufort and Frankiel (1953) exhibits numerical stability for any choice of time step. Substituting space and time-centered difference operators gives an equation to advance the field at a boundary point over a time interval t^{n+1}-t^{n-1} = 2Δt.

$$B_i^{n+1} = \frac{B_i^{n-1} + 2\Delta t/\mu_o(a+b-c-e)}{1+2\Delta t/\mu_o(d)} , \quad \textbf{(30)}$$

where

$$a = \frac{B_{i+1}^{n} R_{i+1}^{n} \; (r_{i+1}^{n} - r_{i}^{n})}{(R_{i+1}^{n} - R_{i}^{n}) \; r_{i+1}^{n} \sigma_{i+1}^{n}} \; ,$$

$$b = \frac{B_{i-1}^{n} R_{i-1}^{n} \; (r_{i+1}^{n} - r_{i}^{n})}{(R_{i}^{n} - R_{i-1}^{n}) \; r_{i}^{n} \sigma_{i}^{n}} \; ,$$

$$c = \frac{B_{i}^{n-1} R_{i}^{n-1}}{2(r_{i+1}^{n-1} - r_{i}^{n-1})} \left[\frac{1}{(R_{i+1}^{n-1} - R_{i}^{n-1}) \; r_{i+1}^{n-1} \sigma_{i+1}^{n-1}} + \frac{1}{(R_{i}^{n-1} - R_{i-1}^{n-1}) \; r_{i}^{n-1} \sigma_{i}^{n-1}} \right] , \qquad \textbf{(31)}$$

$$d = \frac{B_{i}^{n+1} R_{i}^{n+1}}{2(r_{i+1}^{n+1} - r_{i}^{n+1})} \left[\frac{1}{(R_{i+1}^{n+1} - R_{i}^{n+1}) \; r_{i+1}^{n+1} \sigma_{i+1}^{n+1}} + \frac{1}{(R_{i}^{n+1} - R_{i-1}^{n+1}) \; r_{i}^{n+1} \sigma_{i}^{n+1}} \right] ,$$

$$e = \left(\frac{V_{i}^{n+½} + V_{i}^{n-½}}{2} \right) \left(\frac{B_{i+1}^{n} - B_{i-1}^{n}}{R_{i+1}^{n} - R_{i-1}^{n}} \right) + \frac{B_{i}^{n} \; (V_{i+1}^{n+½} + V_{i+1}^{n-½} - V_{i-1}^{n+½} - V_{i-1}^{n-½})}{2 \; (R_{i+1}^{n} - R_{i-1}^{n})} \; .$$

Equation (30) is an explicit expression with second-order accuracy consistent with motion of the medium and variations of conductivity. The only penalty is that additional storage is required for B_{i}^{n-1}, R_{i}^{n-1}, $r_{,i}^{n-1}$ V_{i}^{n-1} and σ_{i}^{n-1}.

Crunch advances magnetic quantities after the hydrodynamic calculation. The program updates the boundary values $B(R_{min})$ and $B(R_{max})$ according to equation (25) and then determines field values at intermediate points from equation (30). The current density for elements in the active region follows from the relationship,

$$j_{i}^{n} = \frac{R_{i}^{n} B_{i}^{n} - R_{i-1}^{n} B_{i-1}^{n}}{\mu_{o} r_{i}^{n} \; (R_{i}^{n} - R_{i-1}^{n})} \; . \qquad \textbf{(32)}$$

In anticipation of High-energy-density Physics Program experiments with preheated or preaccelerated materials, Crunch has the capability to model multiple drive currents with arbitrary time variations. In this mode the user supplies up to five current waveform tables. The calculation applies to a contiguous magnetically-active region. Intervening insulators can be defined by assigning low values of conductivity. The waveform tables are arranged in radial order from inside to outside and associated with particular element boundaries. These boundaries are treated as points of fixed magnetic field, dividing the magnetically-active region in one or more solution segments. Each table gives the current of a segment. The total current enclosed within a fixed point is therefore the sum over inner tables. Voids can be included in the magnetic calculation. A void is a single-element layer between objects that are initially disconnected. The procedure is to set the void boundaries as fixed points associated with the same current table.

This ensures that: (1) the void carries no current, (2) the toroidal field varies as $1/r$ between the two sides and (3) the adjacent regions become electrically connected if the void collapses. Section 9 shows an example of a non-linear magnetic diffusion calculation with a void.

6. Temperature-dependent electrical conductivity

Imploding liner simulations require data on electrical conductivity over a broad range of material conditions, from the solid to plasma states. Following the method of Lindemuth (1985), we assembled material from several sources to create a conductivity library for common metals (aluminum, copper, gold, iron, lead, molybdenum, silver, titanium and tungsten) over the temperature range 300 °K to 10^8 °K. The present library is limited to materials near solid density, adequate for models of the acceleration phase of imploding liners. The low-temperature range (300 °K to 1000 °K) is covered by standard references. We adopted data from exploding wire experiments (Tucker & Toth 1975) for the range from 1000 °K through vaporization (~20,000 °K). Theoretical results (Lee & More 1984, Rinker 1988) were used above 100,000 °K. In the intermediate range of vaporized metal we connected the data with smooth interpolations.

Tucker and Toth (1975) reported the conductivity of metal wires subject to pulsed currents as a function of volume-averaged action. (Action is the time-integral of the square of the current density). The data cover the solid-liquid-vapor transition range that is poorly described by theory. Inference of the volume resistivity depends on the assumption that the wire cross-section does not change significantly during heating. The data are therefore invalid after the wire burst point. In the experiments changes of internal energy resulted almost entirely from ohmic heating. The changes of internal energy inferred from the action in Tucker and Toth (1975) were in good agreement with tabulated values for liquification and vaporization. We determined the temperature from the internal energy by Sesame table interpolations at solid density. Figure 5 shows resulting values of volume resistivity as a function of temperature for aluminum. The results are close to handbook values in the low-temperature range. The resistivity rises at melt and rapidly increases following vaporization. In the wire experiment with current constrained by inductance the ohmic heating rate escalated at the vaporization point rapidly leading to wire explosion.

We used the electron transport tables of Rinker (1988) maintained at LANL for high-temperature conductivity. Tabulated values for partially ionized material in the range $T > 10^5$ °K appeared reasonable. Values converged to those of Spitzer and Harm (1953) at the high end of the temperature range. The tables were invalid at lower temperatures, as shown in Fig. 6*a*. We took a pragmatic approach to fill in the missing data in the range 10^4 to 10^5 °K. The interpolation shown in Fig. 6*b* starts from the exploding wire burst point and follows a smooth transition to the high-temperature Rinker data. Data for the other metals followed similar behavior. Results were collected in an ASCII conductivity library.

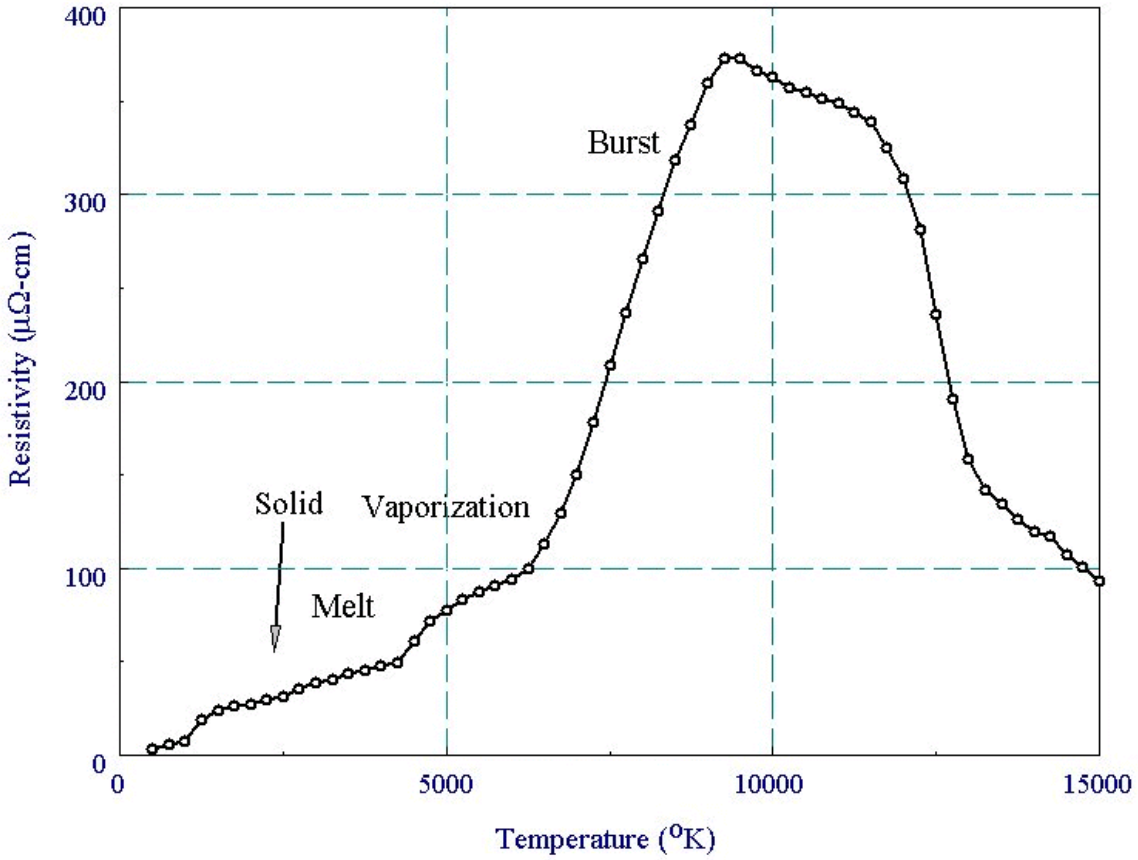


Figure 5. Volume resistivity as a function of temperature. Results derived from the exploding wire data of Tucker and Toth (1975).

7. Circuit model for pulsed-power drivers

Crunch includes a circuit model for self-consistent calculations of drive current in liner simulations [see, *e.g*., Struve *et.al*. (1988)]. Figure 7 shows the generic pulsed power circuit suggested by R. Reinovsky of LANL. The components are divided into three sets: generator, fuse and load. With the exception of the generator capacitance C_g, all components may have assigned time variations. With this feature, the model can address an array of pulsed power devices. For example, a time-dependent value for R_f enables simulations of capacitor banks with fuses. An inductive storage device is represented by assigning an infinite value to C_g, initiating a seed current in the first loop, and setting time-dependent values for L_g, R_g and R_f. This versatility requires that the numerical solution method gives physically reasonable results for large time-scale difference in different sections of the circuit. In this section, we shall first cover a robust method to advance voltages and currents and then discuss circuit contributions from the magnetohydrodynamic solution.

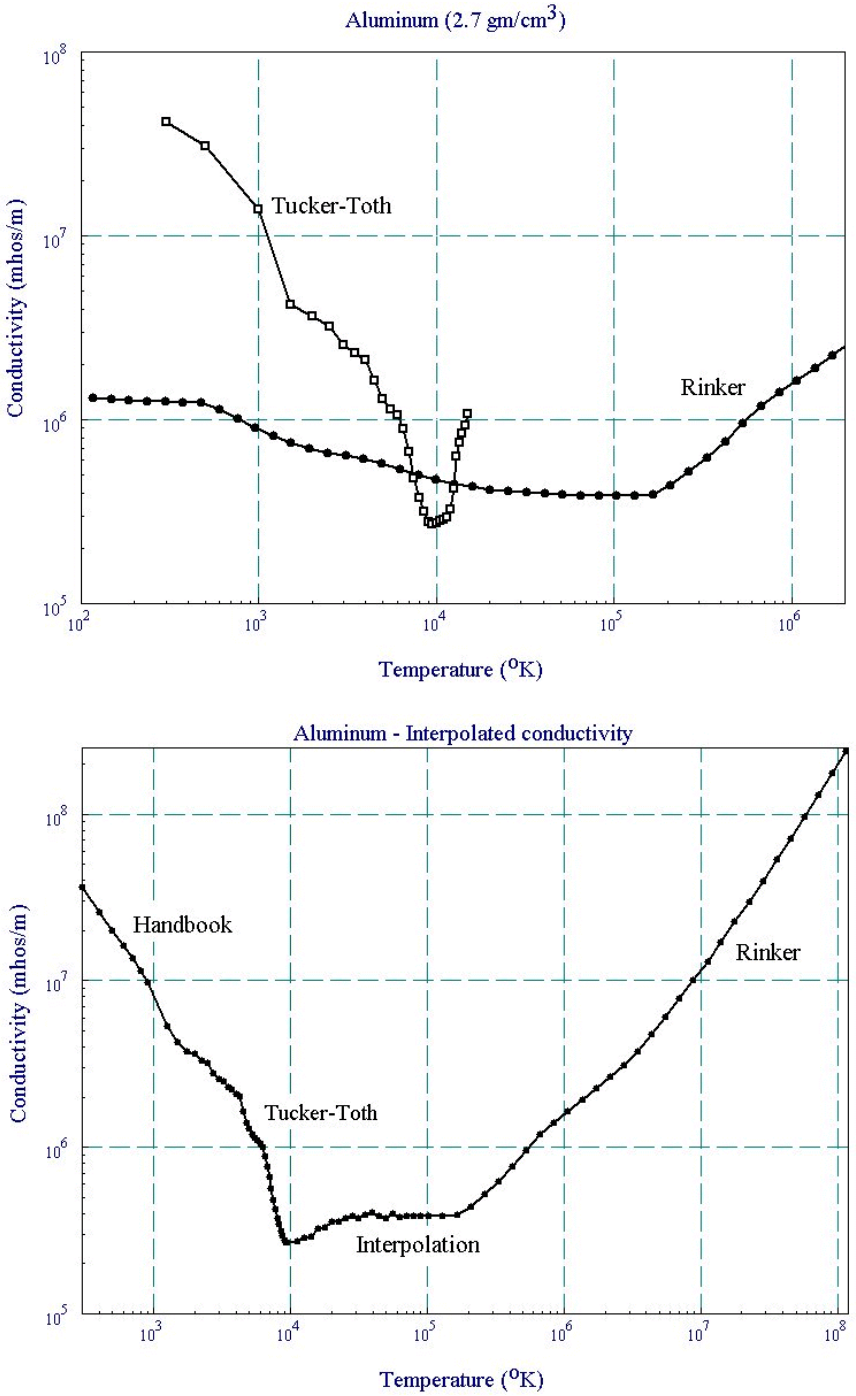


Figure 6. Aluminum electrical conductivity as a function of temperature at solid density. (a) Data sources. Open squares: results derived from exploding wire data of Tucker and Toth (1975). Filled circles: Results from the tables of Rinker (1988). (b) Final values used in the conductivity library including an interpolation at intermediate temperature.

Figure 7 shows seven circuit components. We can write a time-centered difference equation for each component. The resulting set advances the seven values of voltage and current over an interval Δt from time t^n to t^{n+1}.

$$
\begin{aligned}
\frac{V_1^n + V_1^{n+1} - V_5^n - V_5^{n+1}}{2} + \frac{i_1^n + i_1^{n+1}}{2} R_g^{n+\frac{1}{2}} &= 0, \\
\frac{V_4^n + V_4^{n+1} - V_1^n - V_1^{n+1}}{2} + \frac{i_1^{n+1} - i_1^n}{\Delta t} L_g^{n+\frac{1}{2}} &= 0, \\
\frac{V_2^n + V_2^{n+1}}{2} + \frac{i_2^n - i_2^{n+1} - i_1^n - i_1^{n+1}}{2} R_f^{n+\frac{1}{2}} &= 0, \\
\frac{V_4^n + V_4^{n+1} - V_2^n - V_2^{n+1}}{2} + \frac{i_2^{n+1} - i_2^n - i_1^{n+1} + i_1^n}{\Delta t} L_f^{n+\frac{1}{2}} &= 0, \\
\frac{V_3^n + V_3^{n+1}}{2} - \frac{i_2^{n+1} + i_2^n}{2} R_l^{n+\frac{1}{2}} &= 0, \\
\frac{V_4^n + V_4^{n+1} - V_3^n - V_3^{n+1}}{2} - \frac{i_2^{n+1} + i_2^n}{\Delta t} L_l^{n+\frac{1}{2}} &= 0, \\
\frac{V_5^{n+1} - V_5^n}{\Delta t} C_g + \frac{i_1^{n+1} + i_1^n}{2} &= 0.
\end{aligned}
\tag{33}
$$

The time-dependent circuit component values are set at the intermediate time step. We can express equation (40) succinctly in terms of a circuit vector

$$
\boldsymbol{C}^n = \begin{bmatrix} i_1^n \\ i_2^n \\ V_1^n \\ V_2^n \\ V_3^n \\ V_4^n \\ V_5^n \end{bmatrix}. \tag{34}
$$

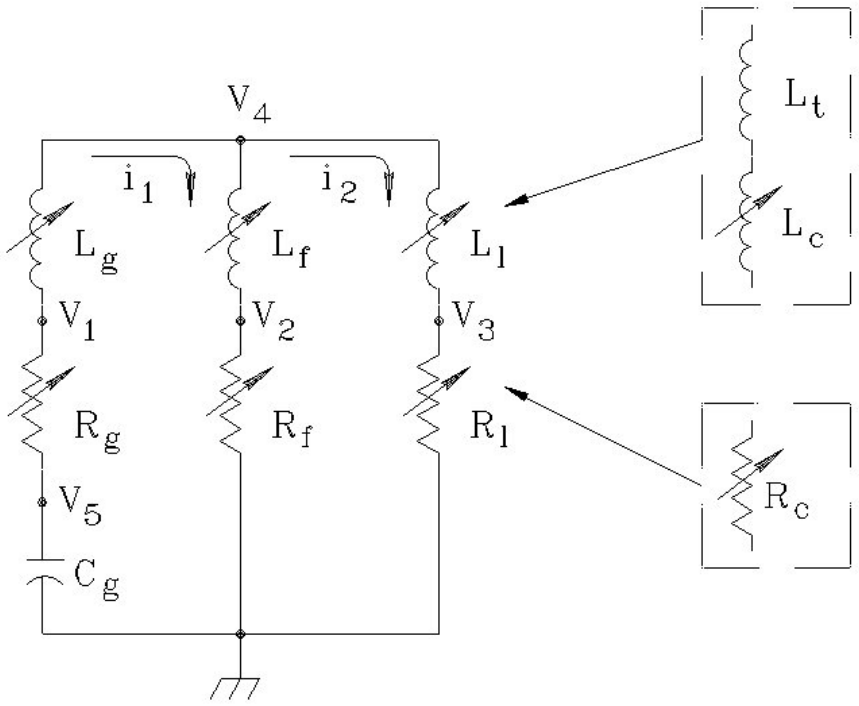


Figure 7. Circuit model for pulsed-power drivers.

After moving all quantities at t^{n+1} to the left-hand side and quantities at t^n on right-hand side, equation (33) becomes

$$\boldsymbol{A}^{n+½}\,\boldsymbol{C}^{n+1} = \boldsymbol{B}^{n+½}\,\boldsymbol{C}^{n}, \tag{35}$$

where the coefficient matrices are

$$\boldsymbol{A}^{n+½} = \begin{bmatrix} R_g^{n+½} & 0 & 1 & 0 & 0 & 0 & -1 \\ 2L_g^{n+½}/\Delta t & 0 & -1 & 0 & 0 & 1 & 0 \\ -R_f^{n+½} & R_f^{n+½} & 0 & 1 & 0 & 0 & 0 \\ -2L_f^{n+½}/\Delta t & 2L_f^{n+½}/\Delta t & 0 & -1 & 0 & 1 & 0 \\ 0 & -R_l^{n+½} & 0 & 0 & 1 & 0 & 0 \\ 0 & -2L_l^{n+½}/\Delta t & 0 & 0 & -1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 2C_g/\Delta t \end{bmatrix}. \tag{36}$$

and

$$\boldsymbol{B}^{n+½} = \begin{bmatrix} -R_g^{n+½} & 0 & -1 & 0 & 0 & 0 & -1 \\ 2L_g^{n+½}/\Delta t & 0 & 1 & 0 & 0 & -1 & 0 \\ R_f^{n+½} & -R_f^{n+½} & 0 & -1 & 0 & 0 & 0 \\ -2L_f^{n+½}/\Delta t & 2L_f^{n+½}/\Delta t & 0 & 1 & 0 & -1 & 0 \\ 0 & R_l^{n+½} & 0 & 0 & -1 & 0 & 0 \\ 0 & -2L_l^{n+½}/\Delta t & 0 & 0 & 1 & -1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 2C_g/\Delta t \end{bmatrix}. \quad \textbf{(37)}$$

We can rewrite equation (35) in a form to advances the circuit vector,

$$\boldsymbol{C}^{n+1} = Inv(\boldsymbol{A}^{n+½})\ \boldsymbol{B}^{n+½}\ \boldsymbol{C}^{n}. \quad \textbf{(38)}$$

The first step in Crunch circuit calculation is to set initial values for the circuit vector [equation (34)]. For a simple capacitor bank, the loop currents are zero and the charge voltage applies at V_1 and V_5. If circuit component values do not change with time, the matrix product Inv($\mathbf{A}^{n+½}$)$\mathbf{B}^{n+½}$ is evaluated only once If any of the components vary, the matrices must be updated at each time step.

The time variations of generator and fuse components follow user-supplied tables. These tables are ASCII files with multiple data lines containing a time and a component value. The following consideration applies in preparing tables. The voltage across a series combination of resistor and inductor is

$$V = iR + \frac{d}{dt}(Li) = i(R+\frac{dL}{dt}) + L\frac{di}{dt}. \quad \textbf{(39)}$$

Equation (39) is consistent with the form of equation (33) if the value of the resistor is taken as the effective value ($R + dL/dt$). The third column in Fig. 7 represents the pulsed power load, all components past the fuse. In Crunch the load inductance is divided into two parts, $L_l = L_t + L_c$. The constant quantity L_t represents the inductance of transmission lines and the experimental chamber volume outside the initial liner radius, $R_{max}{}^{o}$. The quantity L_c is the extra inductance resulting from the liner implosion. For a cylindrical liner of height h, the inductance equals

$$L_c^{\,n} = \frac{\mu_o h}{2\pi} \ln\left(\frac{R_{max}^{\,o}}{R_{max}^{\,n}}\right). \quad \textbf{(40)}$$

where R_{max}^{n} is the present liner outer radius. Note that equation (33) does not give the component value at the intermediate time step. This has little effect on the calculation because the time scale for changes in the drive current is much longer than the hydrodynamic time that determines Δt. Following equation (39), the load resistance has two components, $R_l = R_c + dL_c/dt$. The time derivative of liner inductance is related to the velocity of the outer boundary velocity by

$$\frac{dL_c}{dt} = - \frac{\mu_o h}{2\pi} \frac{V_{max}^{n}}{R_{max}^{n}} \quad . \tag{41}$$

The liner resistance (which includes the effects of magnetic field penetration) equals the voltage on the outside of the liner divided by the drive current (i_2 in figure 7):

$$R_c = \frac{j_I^n h}{\sigma_I^n i_2^n} \quad . \tag{42}$$

In equation (42), j_I^n and σ_I^n are the current density and electrical conductivity in the outer element.

8. Crunch organization

This section discusses how the methods discussed in the previous sections are implemented in Crunch. Although the CGraph post-processor has a full graphical-user-interface, the main Crunch code is controlled by script files. The text files consists of a series of lines with standard commands and parameters. There are several motivations to preserve this venerable technique.

- It is easier to modify an existing script with an editor than to walk through a complex menu structure.
- Script preparation automatically documents a run.
- Scripts can easily be incorporated in batch file procedures for extended run sequences.

To make the process as painless as possible, Crunch has a free-form line parser with detailed error checking. The parser accepts flexible number formats, comment lines, blank lines and indentations to create readable scripts. Commands may appear in any order; Crunch collects all input data initiating a run. Table 2 shows an example of a script file for an imploding liner run.

Table 2
Crunch script file example

```
ATLAS BENCHMARK
* Liner parameters:
*    Aluminum
*    Height: 0.04 m
*    Inner radius: 0.0450 m
*    Outer radius: 0.0465 m
* --- GENERAL CONTROL ---
  SETMODE: LINER
  GEOMETRY: CYLIN
  TMAX: 6.5E-6
  DT:    2.0E-9
  STEPCHANGE: 5.8E-6 0.1E-9
  STEPCHANGE: 6.0E-6 0.5E-9
  INTERPTYPE POLY
* --- MATERIAL PROPERTIES ---
  SESAME 1 ALUM3715
  MATCOND 1 TABLE ALUM
* --- LAYER PROPERTIES ---
  LAYER   1   1   0.0450  0.0465  50
  SETINIT 1 2.700E3 298.0
  VISCOSITY 1 10.0
* --- MAGNETIC PROPERTIES
  CGEN: 1250.0E-6
  LGEN: 10.0E-9
  RGEN: 2.0E-3
  LFUSE: 0.0
  RFUSE: 80.0
  LTLINE: 0.0
  INIT 0.0 0.0 240.0E3 0.0 0.0 0.0 240.0E3
  SETB 50 1
  CIRCFILE 2
* --- DIAGNOSTICS ---
  HISTORY 50
  DTIME: 1.0E-6
ENDFILE
```

Crunch handles 1D calculations in three geometries: planar, cylindrical and spherical. There are three operation modes: hydrodynamic, magnetic diffusion, and coupled magnetohydrodynamic. The magnetic modes are only definable for planar and cylindrical calculations. In the present version of the code, there are three ways to generate shocks:

- Applied pressure on the inner or outer boundary with a user-specified waveform.
- Assignment of initial velocities to elements.
- Coupled magnetic acceleration.

Setting up a system geometry is simply a matter of defining element boundaries and material identifies. In Crunch, elements are grouped into layers, contiguous sets with common properties. Layers are defined by the command

```
LAYER   1   1  0.0450  0.0465  50
```

The example sets up Layer 1, associates its properties with Material 1, assigns inner and outer boundaries of 0.0450 m and 0.0465 m, and divides it into 50 equal elements. The program handles up to 5000 elements in 200 layers. Layers may represent different materials or several layers could be used for variable mesh resolution in a single object. Single element layers can be used for objects that are approximately homogeneous over the problem time scale. An example is a compressed gas inside an imploding liner.

Material numbers are associated with stored tables for equation-of-state information, electrical conductivity, and strength parameters. Layers with the same material number share tables but may have different characteristics. Adjustable layer properties include the initial thermodynamic state (density and temperature with pressure and internal energy set from the Sesame tables), artificial viscosity coefficients, inclusion in the magnetically active region, application of elastic stresses, and initial velocity. There are two options for setting velocity in cylindrical and spherical problems: uniform or radially-weighted. In the latter case, velocities are assigned so that there is no initial compression ($\partial\rho/\partial t\big|_o = 0.0$) for all elements of the layer.

Many High-energy-density Physics Program experiments involve abrupt collisions between layers that are initially disconnected. For these calculations Crunch employs a special single-element *void* layer to perserve continuity of elements. The void model has a single parameter *DBounce* that can be set by the user. In the absence of a definition, the code uses a default value equal to the smallest initial element width. Voids have zero pressure when their width exceeds *DBounce*: $R_i^n - R_{i-1}^n > DBounce$. In this case, the void has no effect on the dynamics of adjacent elements. If the width drops to *DBounce*, the void is squashed. A squashed void is incompressible ($R_i^n = R_{i-1}^n + DBounce$) and has a pressure equal to the average of pressures in adjacent elements. For short time steps adjacent elements can adjust to the sudden change of forces at the void boundaries preserving numerical stability.

Crunch can make time-step adjustments to preserve numerical accuracy and stability under normal circumstances. In magnetohydrodynamic runs, the program uses the smaller of the hydrodynamic or magnetic time. To find the hydrodynamic time, Crunch searches for the minimum element width. This width is divided by either the element initial velocity or a generic sound speed of 1000 m/s. The time step equals the result divided by a hydrodynamic safety factor (default of 10). To find a magnetic time at the beginning of a calculation, the program searches the magnetically-active region for the minimum of the quantity.

$$\mu_o \sigma_i \, (R_i - R_{i-1})^2. \qquad \textbf{(43)}$$

The magnetic time step equals the value divided by the magnetic safety factor (default of 2). At later times when magnetic field values are available to estimate second spatial derivatives, Crunch searches for a minimum of the quantity

$$\mu_o B_i \, \frac{(r_{i+1} - r_i) \, [r_{i+1}\sigma_{i+1}(R_{i+1} - R_i) - r_i \sigma_i (R_i - R_{i-1})]}{R_{i+1}B_{i+1} - R_i B_i} \; . \qquad \textbf{(44)}$$

Automatic time step selection often fails in extreme circumstances like high-speed layer collisions or shock convergence on axis. In this case, the user can fine-tune the time step to pass through the danger zone. A typical run takes less than one minute on a personal computer, and two or three iterations are usually sufficient to achieve a successful solution. To help in this process the program records run variables and makes a graceful exit in the event of errors in intrinsic functions or floating point operations.

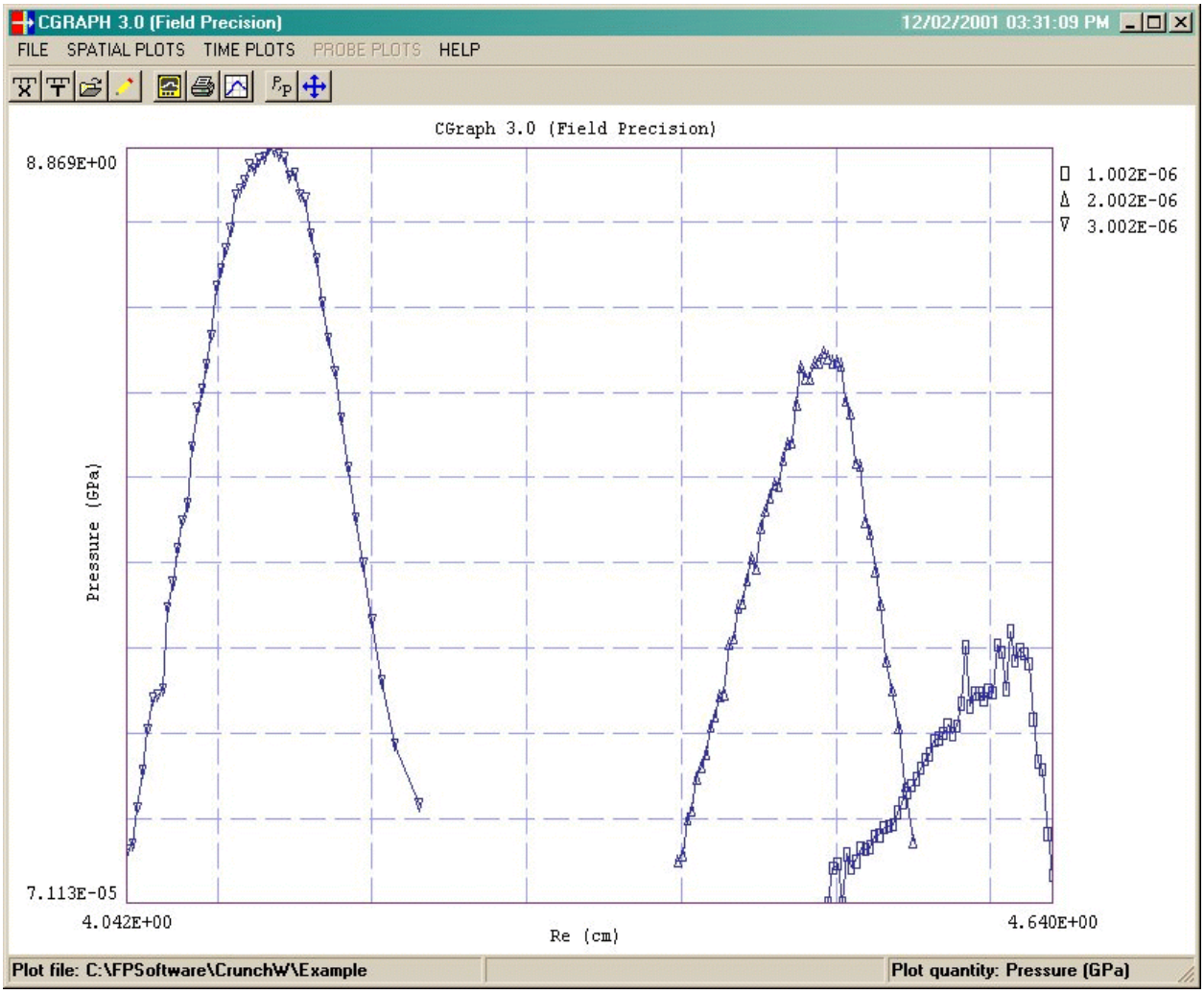
Crunch makes extensive use of external data tables in ASCII format. Previous sections covered complex tables that are generally not prepared by the user: the Sesame equation-of-state tables and the libraries STRENGTH.LIB and CONDUCT.LIB. These material tables are treated as absolute data; there is no way to adjust their values within the code. A second class of tables prescribes time-dependent quantities. These tables are generally prepared by the user to model specific experiments. Examples include waveforms for applied pressure, drive current, or pulsed power circuit components. Tables consists of several data lines, each containing two real numbers (time and tabulated quantity). Non-uniform time intervals and unsorted entry lines are allowed. The tables are processed with a free-form parser and may contain comment lines, indentations and a variety of real-number formats. The command to load a waveform table may include the two parameters *TMult* and *VMult* to scale entries as they are loaded. This feature makes it possible to perform parameteric scans without creating a large number of tables or to maintain a library of normalized waveforms.

To begin a run, Crunch reads commands from the control script and stores parameters. The program then checks the consistency of input data, opens output files and defines element dimensions. The next step is to set initial thermodynamic conditions for the elements. The program then initializes strength properties of elastic elements. The default is to include a positive strain that produces a stress that counteracts the initial element pressure force. This feature prevents spurious layer expansion during initial accelerations or drifts. For magnetic calculations, Crunch sets temperature-dependent element conductivities and determines the limits of the magnetically active region. The program assigns initial current densities if there is an initial bias current. For a calculation with a drive circuit Crunch finds values of the components at t = 0 and sets up the advancing matrices. To complete initialization, the program calculates a safe time step if there is no user specification.

In the main time loop, the first step is to check whether elastic elements have broken or melted and to remove them from the material strength calculation. The program then sums relevant element forces to advance the boundary velocities according to equation (6). The next steps are to advance boundary positions [equation (10)], calculate new element densities [equation (2)] and update artificial viscosity contributions [equation (9)]. Crunch finds changes of internal energy and new values for element pressure and temperature following the method described in Sect. 2. The following sequence of operations occurs for magnetic calculations. Temperature-dependent electrical conductivities are updated. If there is an external drive circuit Crunch updates the coefficient matrices **A** and **B** [equations (36) and (37)], calculates the product Inv(**A**)**B** with an LU decomposition, and advances the circuit vector following equation (38) to determine a new value for the drive current. Alternatively, the code interpolates user-supplied tables to find one or more drive currents at the half time step. The magnetic field values are advanced with equation (30) and then field arrays are shifted, preserving old values for the Dufort-Frankiel algorithm. Finally, new values of the element current density are calculated.

Crunch advances the time and checks whether an automatic step adjustment is required. The final activity in a cycle is to record information. The program supports several types of diagnostics. A data dump is a record of the properties of all elements at a specified time. Depending on the calculation mode, the dump may contain lists of hydrodynamic, elastic and magnetic quantities. Crunch has a flexible system to set dump times. The criteria, which act in combination, are: a uniform number of elapsed steps, a uniform time interval, or up to 25 user-specified times. Dump information can be recorded in two formats: a plot file in binary format passed to the CGraph program or an ASCII listing file. The listing file also contains results of several global calculations. Hydrodynamic listings include total kinetic energy, momentum, and change of internal energy. Magnetic listings include total stored magnetic energy and the time-integral of ohmic power dissipation. In runs with a drive circuit, Crunch makes files of $R_c(t)$ and $L_c(t)$ that can be ported to circuit simulation software. The program can make up to ten history files for specified elements. These files contain detailed records of the time-dependent hydrodynamic and magnetic quantities.

The CGraph program is a powerful analysis tool to create screen and hardcopy plots from Crunch output. Figure 8 illustrates the graphical user interface which accesses several menus. The functions of the *Main Menu* are to review data files, load runs, and transfer to submenus. The *Spatial Menu* controls plots of spatial variations of any hydrodynamic or magnetic quantity for a given data dump. The plots feature automatic grid selection, zoom capabilities, optional material or layer symbols, and superposition capability for any number of plots. The features apply to both screen and hardcopy plots. The *General Menu* creates plots similar to spatial plots except that any hydrodynamic or magnetic quantity can be assigned as the independent and dependent variables. The *Time Menu* creates plots of quantities in any element as a function of time. Here, CGraph scans the plot file extracting information on the specified element and quantity for all stored data dumps. The resolution is therefore determined by the number of dumps. These plots can be superimposed to show variations of hydrodynamic or magnetic quantity in two or more elements. The *History Menu* opens any probe file associated with the run and dislays any of the stored quantities as a function of time in a specified interval. Finally, the *Movie Menu* controls routines that display spatial plots for different data dumps sequentially to



simulate time animations.

Figure 8. Screen display of the CGraph postprocessor.

9. Examples

Crunch is used extensively by LANL researchers and undergoes continual testing. This section illustrates results with two simple calculations. More complex runs with liner acceleration, target collision and shock convergence are described in Humphries and Ekdahl (1996). The first example is a standard LANL hydrodynamic test (Clover 1993), the Sedov blast wave problem. A total energy U is deposited instantaneously in a small spherical volume at the center of a uniform gas distribution with zero temperature and density ρ_o. The properties of the resulting shock can

be determined for an ideal gas described by gamma law. Here, the pressure and internal energy are related to density and temperature by

$$u = C_v T,$$
$$p = (\gamma - 1)\rho C_v T. \qquad \textbf{(45)}$$

In equation (45), C_v is the specific heat at constant volume. We constructed a new Sesame table (IGAS0001.SES) taking $\gamma = 5/3$ and $C_v = 8640$ J/kg-°K (hydrogen). Zel'dovich and Raizer (1966) give the position of the shock front as a function of time after energy deposition as

$$r(t) = 1.15 \left(\frac{U}{\rho_o} \right)^{1/5} t^{2/5}. \qquad \textbf{(46)}$$

The density at the shock front equals $\rho_s = \rho_o (\gamma+1)/(\gamma-1) = 4\rho_o$. For $\rho_o = 0.1$ kg/m^3 and $U = 617$ J, Equation (46) predicts a shock front radius of 0.05 m at 5 μs. Our simulation used 90 spherical-shell elements distributed to a radius of 0.06 m. The initial energy was deposited in a sphere of radius 0.002 m composed of five elements. The spherical mass of 3.351 x 10^{-9} kG and initial temperature of 2.135 x 10^7 °K gives an internal energy of $u = 1.84 \times 10^{11}$ J/kG. Table 3 lists the input file for this problem. The interesting control statements are `DT` and `STEPCHANGE`. `DT` sets the short initial time step of 0.25 ns to resolve expansion of the small heated sphere. The `STEPCHANGE` function raises Δt to 5.0 ns at 0.5 μs, after the shock has expanded significantly. The `LAYER` commands set up two layers for the heated sphere and the quiescent gas, both associated with Material Table 1. The `DIAGNOSTIC` commands call for a single spatial dump at 5.0 μs and a detailed history at Element 70 (initial radius r = 0.0436 m) with a record at each time step. The plot of ρ(r) at 5.0 μs in Figure 9 is direct output from CGraph. The symbols show the location of element centroids. The advantage of the element-based approach is apparent - elements are concentrated near the shock front. The Crunch result is in excellent agreement with theory for the position of shock front and the magnitude of the density peak.

The second example illustrates a non-linear magnetic diffusion problem with multiple current segments. An exploding copper wire of radius 0.5 mm is located inside a copper return cylinder. The wire is defined as Layer 1. Layer 3 is the return conductor, a coaxial shell with inner diameter 1 mm and thickness 1 mm. The conductivities of the wire and return conductor are given by a table from CONDUCT.LIB. Both layers have 25 elements. Layer 2 is the single-element void separating the conductors. The following commands control the magnetic solution.

```
Current 1 CDAMP 1.0E-6 0.5E6
Current 2 CDAMP 1.0E-6 -0.5E6
SetB 25 1
SetB 26 1
SetB 51 2
```

Table 5. Crunch Script for the Sedov Blast Wave Problem

```
SEDOV BLAST WAVE
* --- RUN CONTROL ---
  SETMODE: Hydro
  GEOMETRY: Sphere
  INTERPTYPE Poly
  TMAX:  5.01E-6
  DT: 0.25E-9
  STEPCHANGE 0.5E-6 5.0E-9
* --- MATERIAL PROPERTIES ---
  SESAME 1 IGAS0099
* --- LAYERS ---
  LAYER  1  1  0.0000 0.002 5
  SETINIT 1  0.1000 2.135E7
  VISCOSITY 1 2.50
  LAYER  2  1  0.0020 0.060 90
  SETINIT 2  0.1000 0.00
  VISCOSITY 2 2.50
* --- DIAGNOSTICS ---
 MAKEDIAG 5.0E-6
 NHIST 1
 HISTORY 70
ENDFILE
```

The first command defines Current Table 1. Here CDAMP.CUR is a normalized crtically-damped pulse. The two adjustment factors give a peak current of 500 kA at 1.0 μs. The second command defines Current Table 2 as the negative of Table 1. The third command associates Table 1 with Boundary 25, the outer surface of the wire. The fourth command sets the same current at Boundary 26, the inside edge of the return conductor. These specifications removes the void from the magnetic calculations be ensuring that both surfaces have the same enclosed current. The final statement associates Table 2 with Boundary 51, the outer radius of the return conductor. The combined effect of statements 2 and 5 is to constrain the net system current to zero. Figure 11 shows distributions of current density (*a*) and temperature (*b*) at 0.8 μs. The distribution of j_z in the outer conductor differs from the exponential variation expected at low current because of reduced conductivity in the heated region. Figure 11*b* shows that the temperature on the inner surface is 1200°K. The diffusion process in the wire is highly non-linear. The current density in confined to a thin region at an inwardly propagating front by heating and reduced conductivity in the material behind. When the front reaches the axis, the simulation shows runaway heating and wire explosion.

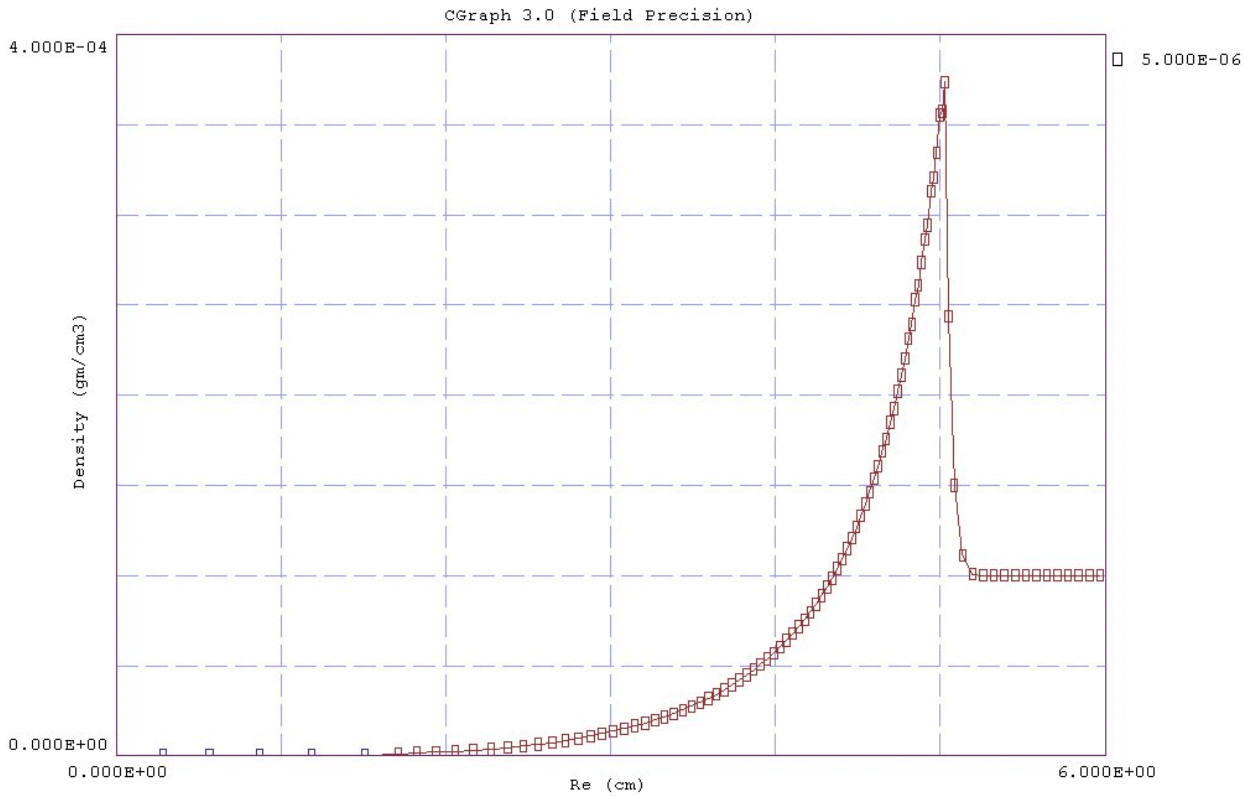


Figure 9. Simulation of the Sedov blast wave problem. CGraph plot of density as a function of position at 5 μs. Ideal gas with $\gamma = 5/3$. Initial density: 10^{-4} gm/cm^3. Initial energy of 617 J deposited in a sphere of radius 0.2 cm. 90 elements.

The Crunch code is in its second version. We anticipate future expansions to cover research requirements of the High-energy-density Physics Program. The next two areas of development are detonation models and radiation transport. The advantages of the element approach to shock hydrodynamics become more apparent in two-dimensional models. We have applied the conservation equations of Sect. 2 to conformal triangular meshes to create the Vogon code. This program handles planar and cylindrical two-dimensional systems. The hydrodynamic package for Vogon which uses Sesame table equation-of-state data has been completed and extensively tested. We are currently working on coupled magnetic acceleration for two-dimensional liner simulations.

Acknowledgements

We would like to thank Bard Bennett for preparing an ASCII version of the Sesame tables and Robert Reinovsky for suggesting and testing the pulsed-power circuit model. This work was supported under Los Alamos National Laboratory Subcontract C50430017-35.

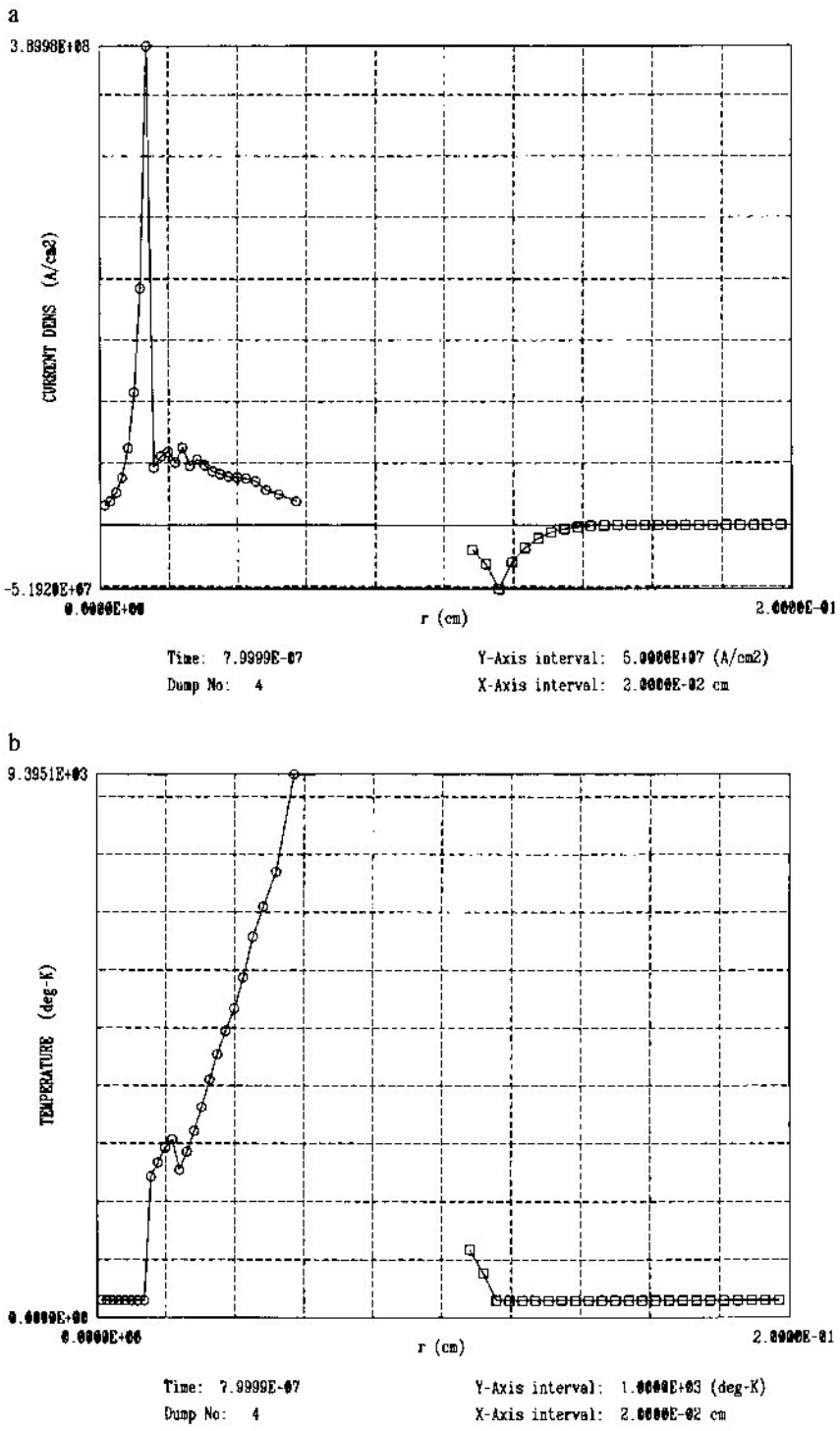


Figure 10. Simulation of an exploding wire, CGraph plots. Open circles: elements of the copper wire with initial radius 0.05 cm (Layer 1). Open squares: elements of the copper return conductor with initial inner radius 0.10 cm and outer radius 0.20 cm (Layer 3). The void element (Layer 2) is not plotted. (*a*)

Current density as a function of radius at 0.8 μs. (*b*) Temperature as a function of radius at 0.8 μs.

REFERENCES

ABDALLAH, J. *et al.* 1980 Los Alamos National Laboratory Report LA-8209.

ALIKHANOV, S.G. & KONKASHBAEV, I.K. 1973 Nucl. Fusion **14**, 3.

BENNET, B.I. *et al.* 1978 Los Alamos National Laboratory Report LA-7130.

CHERNYCHEV, VK. *et al.* 1987 *Megagauss Technology and Pulsed Power Applications*, C.M. Fowler *et al.*, eds. (Plenum Press, New York).

CLOVER, M.R. 1993 Los Alamos National Laboratory Report X6:MRC-93-126 (unpublished).

COURANT, R. & FRIEDRICHS, K.O. 1991 *Supersonic Flow and Shock Waves* (Springer-Verlag, New York), p. 138.

DUFORT, E.C. & FRANKIEL, S.P. 1953 Math. Tables and Other Aids to Comp. **7**, 135.

EKDAHL, C. & HUMPHRIES S. 1998 In *Seventh Int. Conf. on Megagauss Magnetic Field Generation and Related Topics* (Sarov, Russia) (in press).

FUND, C. 1965 *Foundations of Solid Mechanics* (Prentice-Hall, Englewood Cliffs, NJ).

HOCKADAY, M.P. *et al.* 1998 In *Proc. of 10th IEEE Int. Pulsed Power Conf.* (in press).

HUDDLESTON, J.V. 1961 *Introduction to Engineering Mechanics* (Addison-Wesley, Reading, MA).

HUMPHRIES, S. 1997 *Field Solutions on Computers* (CRC Press, Boca Raton, FL), Chap. 12.

HUMPHRIES, S. & EKDAHL, C. 1996 IEEE Trans. Plasma Sci. **24**, 1334.

LEE, Y.T. & MORE, R.M. 1984 Phys. Fluids **27**, 1273.

LINDEMUTH, I.A. 1985 J. Appl. Phys. **57**, 4447.

LYON, S.P & JOHNSON, J.D. (eds.) 1992 Los Alamos National Laboratory Report LA-UR-92-3407.

NEUMANN, J. & RICHTMYER, R.D. 1950 J. Appl. Phys. **21**, 232.

PARKER J. 1993 *A Primer on Liner Implosions* (Los Alamos National Laboratory, Los Alamos, NM) (unpublished).

PARSONS, W.M. et al. 1997 IEEE Trans. Plasma Sci. **25**, 205.

POTTER, D. 1973 *Computational Physics* (Wiley, New York), Chap. 9.

RICHTMYER, R.D. & MORTON, K.W. 1967 *Difference Methods for Initial-Value Problems, 2nd ed.* (Interscience, New York).

RINKER, G.A. 1988 Phys. Rev. A **37**, 1284.

SHERWOOD, A.R. *et al. 1980 Megagauss Physics and Technology*, P.J. Turchi, ed. (Plenum Press, New York), p. 391.

SPITZER, L. & HARM, R. 1953 Phys. Rev. **89**, 977.

STEINBERG, D.J. 1996 Lawrence Livermore Laboratory Report UCRL-MA-106439.

STRUVE, K.W. *et al.* 1998 in *Proc. of 1997 Pulsed Power Conf.* (in press).

TUCKER, T.J. & TOTH, R.P. 1975 Sandia National Laboratories Report SAND-75-0042.

TJRCHI, P.J. *et al. 1980 Megagauss Physics and Technology*, P.J. Turchi, ed. (Plenum Press, New York), p. 375.

ZEL'DOVICH, YA. B. & RAIZER, YU. P 1966 *Physics of Shock Waves and High-Temperature Hydrodynamic Phenomena* (Academic Press, New York), p. 93.