HIFAN 1865

DISH CODE

A deeply simplified hydrodynamic code for applications to warn dense matter

by
Richard More

Consultant for
Lawrence Berkeley National Laboratory (on behalf of U.S. HIFS-VNL)
1 Cyclotron Road, Berkeley, CA 94720
Accelerator Fusion Research Division

August 22, 2007

This work was supported by the Director, Office of Science, Office of Fusion Energy Sciences, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor The Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or The Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or The Regents of the University of California.

This work was supported by the Director, Office of Science, Office of Fusion Energy Sciences, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

22 August 2007

DISH CODE

A deeply simplified hydrodynamic code for applications to warm dense matter

Richard More, consultant,

LBNL, Berkeley, CA

1.) Introduction

2.) Outline of hydrodynamic calculation

3.) Hydrodynamic equations

4.) Equation of state

5.) Driving energy

6.) Zoning

7.) Omitted physics

8.) Tests and test problems

9.) Better calculations

References

1.) Introduction

DISH is a 1-dimensional (planar) Lagrangian hydrodynamic code intended for application to experiments on warm dense matter. The code is a simplified version of the DPC code written in the Data and Planning Center of the National Institute for Fusion Science in Toki, Japan. DPC was originally intended as a testbed for exploring equation of state and opacity models, but turned out to have a variety of applications. The DISH code is a "deeply simplified hydrodynamic" code, deliberately made as simple as possible. It is intended to be easy to understand, easy to use and easy to change.

1.1.) Warm Dense Matter

"Warm dense matter" means material at densities of .01 - 1 x solid density with temperatures of ~ 1000 K to ~ 20,000 K (i.e., 0.1 eV to perhaps 2 eV). WDM temperatures can be produced by laser energy deposition, ion or electron beam heating, pulsed electric current or x-ray heating. Warm dense matter occurs in nature in geophysical and astrophysical contexts. There are industrial applications in laser machining, explosive forming, surface modification by laser or ion-beam irradiation, EUV sources for lithography, initiation of explosives, exploding fuzes and arcs, high-temperature combustion, etc.

Warm dense matter research extends high-temperature physics and chemistry into a temperature range where pressures are so high that dense material cannot be confined. The experiments could not be performed without rapid heating and rapid diagnostics. Suitable diagnostics include electrical, optical and x-ray probes.

1.2.) WDM has high pressure

The pressure of hot matter can be estimated by the ideal-gas law:

$$p(Mbar) \approx 0.96 \frac{Z^*+1}{A} \rho(g/cm^3)\, T(eV) \qquad (1)$$

where Z^* = ion charge-state, A = atomic weight, ρ = mass density and T = temperature. Eq. (1) says that solid density matter has a pressure in the megabar range at T ~ 1 eV. Steel containers can hold pressures of tens of kilobars, but WDM will fracture most laboratory apparatus. The high pressure of warm dense matter leads to rapid hydrodynamic expansion, i.e., explosion.

The speed of hydrodynamic motion usually can be estimated from the sound speed. For thermal-equilibrium fluid, the sound speed is given by

$$c_s = \sqrt{\left(\frac{\partial p}{\partial \rho}\right)_S} \tag{2}$$

The quantity inside the square-root in Eq. (2) is the partial derivative of pressure with respect to density at constant entropy S. The sound speed can be estimated using the ideal-gas formula:

$$c_s \approx 1.0\ 10^6 \frac{cm}{\sec} \sqrt{\frac{Z^*+1}{A} kT(eV)} \tag{3}$$

It is instructive to compare Eq. (3) with text-book numbers for the real sound speed: air (20 C) = 3.4 10^4 cm/sec, water (20 C) = 1.48 10^5 cm/sec and aluminum = 6.4 10^5 cm/sec. The ideal-gas sound-speed from Eq. (3) is too small for condensed matter. (This fact is a warning that the ideal gas EOS is not good enough.)

1.3.) Hydrodynamic calculations answer certain questions

A typical experiment on WDM involves rapid heating, rapid diagnosis of the resulting state and explosive disassembly of the sample. Hydrodynamic calculations answer the question: How fast do these changes occur? The calculation tells how fast the heating and diagnostics must be. In most cases the expansion speed is a few times the sound speed.

Hydrodynamic calculations sharpen the question. The expansion is a continuous process and partly overlaps the heating. The outer surface of a target expands and cools before the interior can start to move. Certain diagnostics, such as visible light emitted by the target, report on the outer surface while x-ray probes would report on internal conditions. Hydrodynamic calculations give estimates of the space-time scale over which the desired high-density, high-temperature conditions persist.

Uncertainties in the hydrodynamic calculations are a consequence of the uncertain properties of matter at WDM conditions. There would be less reason to do experiments on WDM if the hydrodynamic calculations could reliably predict the behavior of rapidly-heated targets. For designing and interpreting experiments, the hydrodynamic calculations can be used in a sophisticated way: "high" and "low" calculations (high-pressure or high-temperature, etc.) can be used to bound the range of likely conditions.

1.4.) Even one-dimensional calculations can be useful

Various existing computer codes perform two- and three-dimensional hydrodynamic calculations. These codes represent large investments in code development and testing. For example, the laser-fusion code LASNEX is said to be the result of 100 man-years of code development.

Such elaborate 2-D or 3-D codes describe many physical processes, especially for high-temperature plasmas, but carry with them issues of access and clarity directly related to their complexity. It is obviously difficult to be certain what calculation is being done by a large code whose listing may be several thousand pages long - and unavailable.

Some typical phenomena predicted by 2- or 3-D codes:

(1.) In a diverging flow, the density drops more rapidly than in the corresponding planar expansion;
(2.) In a converging flow, the density rises more rapidly than in a corresponding planar compression;
(3.) Material can squirt ("jetting") when two surfaces collide at a small angle, and
(4.) Flows in two or three dimensions may contain angular momentum ("vorticity"), which does not occur in planar flow.

One-dimensional calculations omit these phenomena, but many of the simplest and cleanest experiments will be designed for planar motion. One-dimensional calculations can help us understand limited parts of an experiment (e.g., initial expansion perpendicular to a flat surface), even if a complete simulation requires 2-D or 3-D calculations. Even in experiments involving 2-D or 3-D motion, appropriate use of common sense can tell us what will be the 2-D or 3-D effect.

1.5.) WDM experiments involve uncertainties

Material properties are surprisingly uncertain in the WDM temperature range. For example, the liquid-vapor critical point is uncertain by about 50% for most metals. The EOS (pressure and energy) has a large uncertainty outside the well-studied conditions near the solid and liquid states.

Electronic properties such as electrical conductivity or optical constants (n, k), are poorly known and may have surprising behavior for special materials. There are reasons to expect "metal-insulator transitions" in the WDM range where the electrical conductivity may change by several orders of magnitude. Today it is not known whether these changes are sharp or gradual.

We can try to guess WDM material properties by interpolating between solid-liquid data and the relatively well-understood hot plasma. For certain materials this interpolation may give good results. For other materials, there is evidence that special properties occur in the WDM range. Gold, for example, is a metallic solid/liquid, and the high-temperature plasma has good conductivity, but gold vapor may have a semiconducting phase at intermediate temperatures ~ 1 eV. Glass (SiO_2) is a transparent covalent/ionic insulator at room temperature; doubtless glass becomes a conducting plasma at high temperatures, but it may be something quite special in the WDM range, if we can judge from laser absorption data.

1.6.) Heat conduction has only a weak influence

Thermal conduction and radiation heat conduction have important effects on the dynamics of high-temperature plasmas, but it appears they are less important in the WDM range. For this reason we do not include these processes in DISH.

Here we compare heat conduction with an estimate of hydrodynamic energy flow for aluminum at kT ~ 1 eV (we assume v ~ 3 c_s for free expansion):

$$\left[\tfrac{1}{2}\rho v^2\right]v \approx \tfrac{1}{2}(2.7\frac{g}{cm^3})\left(1.2\ 10^6\frac{cm}{\sec}\right)^3 \approx 2\ 10^{11}\frac{Watts}{cm^2} \tag{4}$$

We estimate the heat conduction using the room temperature thermal conductivity of metallic aluminum ($\kappa \sim 2.8\ 10^4$ Watts/cm-eV). The themal conductivity of metals is thought to be a weak function of temperature for temperatures up to several eV. Most materials have a thermal conductivity much smaller than aluminum.

If we heat a layer 1 micron thick to a temperature ~ 1 eV, we expect a temperature gradient $\nabla T \sim 10^4$ eV/cm. Then the heat current $q \sim \kappa\nabla T \sim 3\ 10^8$ W/cm^2. Whether this seems like a very large number or a moderately small number depends on the experiment considered. During one nanosecond, the energy carried by q is ~ 0.3 Joules/cm^2. We expect the "damage threshold" for significant heating of a surface is something like 1-2 Joule/cm^2, and as soon as hydrodynamic expansion begins both the gradients and heat currents are reduced.

At lower energy-density the flow is much slower and heat conduction may again be a dominant process for determining the temperature.

The black-body emission for T ~ 1 eV is ~ 10^5 Watts/cm^2. Since the real surface emission is usually only a fraction of this number, radiation energy flow can be neglected for many purposes.

Calculation of heat conduction adds to the delicacy and cost of numerical hydrodynamics and requires additional material properties (thermal conductivity and opacity). For these reasons we omit the thermal and radiation conduction from DISH. This omission would not be appropriate for higher temperatures or for processes on different space-time scales from those considered in the estimate above. The code user must take personal responsibility for the physical model he decides to apply.

The parent code DPC includes heat conduction and we left markers in DISH to show how it is called.

1.7.) Many reasons to be careful

We urge care and skepticism for calculations performed with DISH or any other hydrodynamic code. It is difficult to produce a complete list of potentially important corrections to the results, but at least the following points should be kept in mind:

1.) DISH assumes 1-D motion (planar motion).
2.) DISH neglects heat conduction.
3.) Energy deposition in DISH calculations is only as realistic as the user makes it.
4.) The EOS data supplied here is severely limited and may be entirely incorrect outside the density-temperature range described below. The ideal-gas and Van der Waals' EOS described here omit ionization, solid-state bonding effects and formation or dissociation of molecules.
5.) Several types of non-equilibrium might exist but are not considered by the code.

1.8.) An easy-to-use, easy-to-modify code

For applications within its scope, DISH should be easy to use and easy to change. It is compiled with FORTRAN77 and uses no system libraries or special functions. (Incidentally, the author makes changes directly in the code and recompiles it for every run. It is compiled using the -r8 switch to get a larger word size on a 32-bit computer.) As far as possible the variables and their units are labeled in the code.

To help learn to use the code, five simple test-cases are built into the code and by comparison of these the user can extend the code to more complicated situations. The test-cases are: ideal-gas shock-wave, ideal-gas rarefaction, 2-phase rarefaction, deposition-driven rarefaction and deposition-driven porous target. One switches between the cases with a parameter "icase" chosen early in the calculation. The first two cases can be checked with pocket-calculator evaluation of the exact solutions.

The author is not able to provide user consulting service: users who encounter problems must find their own solutions. Of course the author is happy to be informed if mistakes or ambiguities are found and will incorporate appropriate improvements in future versions of the code.

Summary: Key features (and limitations) of DISH:

Planar
Lagrangian
EOS is simple but limited
Transport and non-equilibrium phenomena are omitted
User-sets the time-step (and must verify it is not too large)
Problems are driven by initial condition, boundary conditions or by deposition
FORTRAN source listing is available

1.9.) UNITS of the main quantities

The basic code variables and their units are:

time t	[sec]
position x	[cm]
velocity v	[cm/sec]
mass density ρ	[gram/cm^3]
energy density E	[Joules/gram]
pressure p	[Joules/cm^3]
temperature T	[eV]
Entropy S	[Joules/gram-eV]

The code uses mixed cgs-MKS units, so a few code equations have conversion factors. In this report the temperature is often written "kT"; the Boltzmann constant k converts temperature units (usually Kelvin or Celsius degrees) into eV; however in the code units k = 1. Output files can be written using microns (10^{-4} cm) and nsec (10^{-9} sec) or even psec (10^{-12} sec). Pressure can be printed in Mbar (1 Mbar = 10^5 Joule/cm^3). The change of units occurs just before the write statement and only affects the printed data. As the user changes "case" he/she should check the output units and their labeling. The change of units from cm (code unit) to microns (output unit) is performed by a write statement "write ... x(i)/xun, ..." where the x-unit = "xun" is 10^{-4}.

If the user is not confident about recognizing and changing units, we recommend setting the parameters "XUN", "PUN", "TUN" to unity and appropriately modifying the write formats; then the code units listed above will be used.

The code variables generally exist in "old" and "new" versions. For example, the zone boundary positions are "xold" and "xnew". During each time step, the "old" data is used to generate "new" values. At the end of the time-step, the "new" values are copied into the "old" data locations to prepare for the following time-step.

The physical variables can be divided into "zone edge" and "zone center" quantities. The zone edges are points $x_j(t)$; usually x_0 is the left-most zone edge (nothing exists to the left of x_0). The material in zone j lies between x_j and x_{j+1}. The material properties are labeled by the left boundary point, so that ρ_j is the density of the fluid between x_j and x_{j+1}.

Zone edge variables	x_j, v_j
Zone center variables	ρ_j, E_j, T_j, p_j, S_j

In calculations with thermal conduction, the heat current q_j is a quantity of heat energy crossing the point x_j (per cm^2 per sec) and thus is a zone-edge variable. The heat deposition (~ div q) is a zone-center variable.

DISH uses a user-selected time-step Δt. If Δt is too small the calculation will be unnecessarily slow, but if it is too large the results may be inaccurate. A diagnostic parameter, the "Courant condition" is printed to help judge whether the time-step is satisfactory (see section 8). Most hydro-codes select their own time-steps for some optimum efficiency. This approach has the advantage of using a very small Δt at delicate times (initial start-up or at a sudden collision of two plasmas). It has the occasional disadvantage of leading to very slow calculations.

DISH's uniform and equal time-steps are convenient for preparing plots of the output. For modest calculations with only a few hundred zones the runs are fast enough that optimization seems unnecessary. This choice imposes on the user a responsibility to look at the output to examine the energy conservation and Courant condition to verify that Δt is satisfactory. If the number of zones is increased, for example, the time-step must be decreased.

The time-steps are controlled by two nested loops. This makes it easy to print output at specified times in the calculation. The user can change the number of steps together with the basic time-step Δt.

2.) Outline of hydrodynamic calculation

Set-up problem - set initial conditions

Outer time loop:

(print current status of calculation) ( ) = only every 10^3 - 10^4 time steps

(Audit energy changes) " " " " "

Inner time loop:

form artificial viscosity

find new zone velocities

find new zone positions, change densities

calculate energy change due to pdV work

calculate energy deposition

[calculate heat current, form thermal deposition] [inactive in DISH]

[calculation radiation flux, form radiation deposition] ["]

add up the energy changes

call EOS for pressure and temperature at new ρ, E

Update figures-of-merit (Courant, etc).

If step is accepted, replace old variables by "new" variables

End of inner time loop

Print various quantities

End of outer time loop

Final printing

3.) Hydrodynamic equations

3.1.) Euler-Lagrange equatons

To understand a simulation, we need to know what equations are being solved and what approximate methods are used. A big advantage of a "deeply simplified" code is that we can explain what it is doing. DISH aims to solve the Euler equations for the flow of an ideal fluid:

Mass conservation:
$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho v) \tag{5}$$

Acceleration:
$$\rho \frac{\partial v}{\partial t} = -\rho v \cdot \nabla v - \nabla p \tag{6}$$

Change of energy per gram:
$$d\varepsilon = T ds + \frac{p}{\rho^2} d\rho \tag{7}$$

Entropy equation:
$$\frac{\partial s}{\partial t} = -v \cdot \nabla s \tag{8}$$

In general, the fluid velocity v is a vector and the reader can test his understanding by inserting vector notation into the equations above. Here we consider planar flow, so the velocity v (and its gradient) are in the x-direction.

Energy conservation is a theorem, easily proven by combining equations (5-8) above:

$$\frac{\partial}{\partial t}\left(\rho\varepsilon+\frac{1}{2}\rho v^2\right) = -\nabla\cdot\left(\left[\varepsilon+\frac{v^2}{2}+\frac{p}{\rho}\right]\rho v\right) \tag{9}$$

The entropy equation and energy-conservation theorem must be modified if there is energy deposition, heat conduction or emission of radiation.

3.2.) Lagrangian solution of the Euler equations

Several methods can be used to solve the Euler equations. "Eulerian" methods use a fixed spatial grid. Lagrangian methods use a spatial grid that is carried with the flow, so the mass contained in each zone should remain constant. (Note - in some cases one might want to constrain the density to remain in the allowed range of the EOS data and if this is done the mass in some zones might be changed. The code does not do this at present.) There are more complicated or mixed methods. DISH uses the simplest Lagrangian method.

We explain the numerical method in two steps. First, we write the "Lagrangian" form of the Euler equations. Then we show the finite-difference scheme actually used by DISH. The Lagrangian form of the hydrodynamic equations is a regrouping of terms in the equations above. The Lagrangian equations contain the *convective* derivative, a combination of space and time derivatives. This derivative is zero for a quantity that is simply carried by the flow.

Convective derivative:
$$\frac{d}{dt} = \frac{\partial}{\partial t}+v\cdot\nabla \tag{10}$$

Mass conservation equation:
$$\frac{d\rho}{dt} = -\rho\ \nabla\cdot v \tag{11}$$

Acceleration equation
$$\rho\frac{dv}{dt} = -\ \nabla p \tag{12}$$

Adiabatic flow condition:
$$\rho\frac{ds}{dt} = 0 \tag{13}$$

In Lagrangian form, the energy conservation equation reads:

$$\frac{d}{dt}\left(\rho\varepsilon+\frac{1}{2}\rho v^2\right) = -\nabla\cdot(pv) - \left(\rho\varepsilon+\frac{1}{2}\rho v^2\right)\ \nabla\cdot v \tag{14}$$

The last two equations must be modified when the calculations include energy deposition or heat conduction.

3.3) Difference equations

There are several choices for a numerical difference scheme to approximately solve the fluid equations. A "fluid element" is a little bit of fluid. In the Lagrangian method, the location of a "fluid element" is a function $x(x_o, t)$ where x_o is the initial position of that bit of fluid. The initial coordinate is represented by a finite number of zones, so x_o is equivalent to the zone number j; thus we need to calculate $x_j(t)$. The difference equations used in DISH are:

Mass conservation:

$$\rho_j^{new} = \rho_j^{old} \frac{\left(x_{j+1}^{old} - x_j^{old}\right)}{\left(x_{j+1}^{new} - x_j^{new}\right)} \tag{15}$$

or, equivalently,

$$\rho_j^{new} = \frac{M_j}{\left(x_{j+1}^{new} - x_j^{new}\right)} \tag{16}$$

M_j = mass per zone is constant in time. (The planar calculation always refers to a unit area perpendicular to the flow direction, so M_j is really mass/cm^2).

The zone positions are advanced by

$$x_j^{new} = x_j^{old} + v_j^{new}\Delta t \tag{17}$$

and the velocities are advanced by the acceleration equation,

$$v_j^{new} = v_j^{old} + \left(\frac{p_j^{old} - p_{j-1}^{old}}{\frac{1}{2}\left[M_{j-1} + M_j\right]}\right)\Delta t \tag{18}$$

(The code also includes artificial viscosity, discussed below, which appears as an addition to the pressure p). These equations look like Newton's laws for "mass points" x_j, assigned a mass equal to half the masses of the adjacent zones. The force (per cm^2) is the pressure difference in the two adjacent zones. To avoid dividing by zero, the user should not assign zero mass to two adjacent zones when he sets up a calculation.

If there is no energy deposition or heat conduction, the flow is approximately adiabatic.

The numerical method is neither unique nor exact. It does not exactly conserve energy, but for good calculations (with appropriate time-steps and spatial zoning) the energy is conserved to about 1 %. The failure of energy conservation is a measure of numerical accuracy.

The first and last zones are treated by special formulas which the user can find (and control) in the subroutines DELTAX, DELTAV. There is no claim that these formulas are especially accurate; the first and last zones are unreliable enough that it is sometimes nicest to omit them from the plots.

3.4.) Artificial viscosity

The artificial viscosity is a numerical artifice used to prevent density oscillations behind shock-waves. It is an extra pressure, and in DISH it only occurs during compression. The formula used by DISH is Eq. (12.41) of Richtmyer and Morton:

$$p_j^{art} = 10^{-7} a_{AV} \rho_j^{old} \left(v_{j+1}^{old} - v_j^{old} \right)^2 \tag{19}$$

For expanding fluid, $(\Delta v/\Delta x)$ is positive and $p^{art} = 0$. The numerical factor 10^{-7} converts ergs/cm^3 into Joules/cm^3 (= code unit for pressure). The coefficient $a_{AV} = 1.5$ is a dimensionless viscosity parameter. (There are various forms for the artificial viscosity and some are active also during expansion.) The artificial viscous pressure is also included in the pdV work for the energy equation.

We invite the reader to experiment with the coefficient a_{AV} controlling the strength of artificial viscosity, especially in the first sample problem (ideal-gas shock wave) to see how it controls the density oscillations behind the shock front without degrading the accuracy of the calculation.

The artificial viscosity does not have the proper functional form to correspond to a differencing of the physical viscosity of the Navier-Stokes equations. For example, p^{art} does not scale to a physically sensible limit for infinitely small zones. Instead, the artificial viscosity is a numerical trick that gives a reasonable approximate treatment of shock-waves for a finite zone size.

4.) Equation of state

4.1.) The equation of state describes the flowing target material

After the code updates the density and energy/gram for each zone, an EOS call finds the corresponding pressure, temperature, entropy and sound speed. The equation of state supplies the thermodynamic properties of the target material. *The difference between mountain air and stainless steel only appears through their equations of state.*

Two EOS options are provided for DISH. One is an ideal-gas EOS and the other is a generic van der Waals' EOS including the liquid-vapor two-phase region. These

models have limited application. For anything more general or more accurate, additional subroutines and/or data sources would be required.

A few thermodynamic formulas are needed to deal with the equation of state. We quote these formulas in the form adopted for the DISH code. It should be underlined that these are "equilibrium" thermodynamic formulas: any one zone has a single temperature. The internal degrees of freedom corresponding to evaporation/condensation are assumed to have settled to their thermal-equilibrium states more rapidly than the hydrodynamic changes. The parent code DPC has provision for exploration of non-equilibrium equations of state but this is not a tool for users.

Most high-T plasma hydrodynamic codes use density and temperature as basic code variables. This is natural for hot plasmas where ionization, radiation and electron thermal conduction are naturally expressed as functions of temperature. For the lower temperatures of WDM, the temperature does not play such an essential role. In DISH, the basic thermodynamic variables are chosen to be density ρ and energy/gram ε.

Thermodynamic properties of any material are determined by the entropy per gram of material,

$$S = S(\rho, \varepsilon) \tag{20}$$

If we know this function, the other important thermodynamic properties are obtained by

$$\frac{1}{T} = \left(\frac{\partial S}{\partial \varepsilon}\right)_{\rho} \qquad \frac{p}{\rho^2 T} = \left(\frac{\partial S}{\partial \rho}\right)_{\varepsilon} \tag{21a,b}$$

In practise, the pressure and temperature are given directly as functions of ρ, ε.

4.2.) Ideal gas EOS

The ideal-gas law in DISH lets the user specify a fixed ion charge-state Z*. The ideal-gas equations are, in the usual text-book notation,

$$p = \frac{(Z^*+1)NkT}{V} \tag{22}$$

$$E = \frac{3}{2}\frac{(Z^*+1)NkT}{AM_p} \tag{23}$$

where N = number of atoms (or molecules) in volume V, A = atomic (or molecular) weight, Z* = fixed average ionization state and M_p is the atomic mass unit.

The ion charge-state Z* is a constant, independent of density and temperature, in this EOS. In fact, there exists a Thomas-Fermi estimate of Z*(ρ, T) which is even included as a subroutine of DISH, but the problem with simply inserting this into the ideal-gas equations is that the resulting EOS would omit the large contribution of ionization energy and would not be thermodynamically consistent. A proper EOS including ionization energy and ion interactions is more complicated. In the same way the state of molecular bonding is assumed not to change due to dissociation or other chemical changes. These limitations on the EOS imply limitations on the range of density-temperature conditions that can be studied.

4.3.) Van der Waals' law

The Van der Waal's gas law has a distinguished history as the simplest description of liquid-vapor phase equilibrium. It is not very accurate close to the critical point, and not very accurate in a global sense; however it has the same qualitative behavior as the "true" equation of state, at least as regards the liquid-vapor transition. The VdW formulas are simple and nicely adapted to hydrocode use. We hope that careful setting of the parameters (a, b) described below will enable the user to obtain reasonable results.

In the DISH code the Van der Waal's EOS is generated in a series of subroutines all beginning with the letter "V", e.g., VESETUP, VEOSINT, VEVALV, VUNSCALE, etc. These subroutines require data from a file "TPHDAT" which must be available. This file contains the liquid-vapor phase-boundary in units scaled to the critical density and temperature (a single table for all materials).

pressure
$$p = \frac{\rho kT}{AM_p(1-b\rho)} - a\rho^2 \tag{24}$$

Energy/mass
$$\varepsilon = \frac{3}{2}\frac{kT}{AM_p} - a\rho \tag{25}$$

Entropy/mass
$$S = \frac{k}{AM_p}\log\left(AM_p\frac{1-b\rho}{\rho\lambda^3}\right) \tag{26}$$

Thermal deBroglie wavelength
$$\lambda = \sqrt{\frac{h^2}{2\pi AM_p kT}} \tag{27}$$

Gibbs free energy
$$G = \varepsilon - TS + \frac{p}{\rho} \tag{29}$$

Critical point parameters:

Critical density, temperature $\rho_c = \frac{1}{3b}$ $kT_c = \frac{8a}{27b} AM_p$

Critical pressure, energy/mass $p_c = \frac{a}{27b^2}$ $\varepsilon_c = \frac{a}{9b}$

In actual use, the VdW EOS requires 3 inputs: the atomic mass A, the solid density ρ_{sol}, and an estimate of the critical temperature T_c. These inputs determine the constants a, b. The solid density is a little less than 1/b because the code includes a small thermal expansion (the pressure is zero at the solid density at room temperature).

Examining the formula for the pressure, one immediately sees the VdW EOS will give an unsatisfactory result at or above solid density.

4.4) Limits of the VdW EOS

The Van der Waals' EOS predicts an infinite pressure for solid density ~ 1/b, so it cannot calculate shock-waves or compression of solids.

This limitation is caused by the VdW EOS. With a suitable high-density EOS, DISH would have no trouble to calculate shock compression of solids. (Of course the code does not include low-pressure elastic-plastic phenomena.) Shock-waves in gases are good test-problems because there are analytic solutions, and DISH accurately satisfies the Hugoniot relations (equivalent to mass, momentum and energy conservation).

The VdW EOS also omits molecular dissociation, ionization and other more delicate phenomena.

4.5.) Maxwell construction for liquid-vapor equilibrium

The EOS described above is modified in the two-phase (liquid-vapor) region by the Maxwell construction described in textbooks of thermodynamics. It is then written as a function of density ρ and material internal energy ε, so the EOS subroutines return $T(\rho,\varepsilon)$, $p(\rho,\varepsilon)$, $S(\rho,\varepsilon)$ and the sound-speed $c_s(\rho,\varepsilon)$. This process is complicated enough to be left in the subroutines without complete description here. Suffice it to say that we tabulate the two-phase boundary in a general form (parameters scaled to the critical point).

A parameter KNOMAX enables one to evaluate the Van der Waals' formulas without the Maxwell construction (KNOMAX = 1). This option should predict strange behavior of fluids which have somehow entered the two-phase region. That strange behavior is interesting because many other hydrodynamic codes use EOS models without the Maxwell construction.

4.4.) What's missing from the EOS?

It is important to recognize what is missing from any EOS model. Our EOS models do not include ρ, T-dependent ionization! That means they are limited to temperatures well below the first ionization potential, which is typically in the range 5-10 eV. The EOS also omits formation or dissociation of molecules. The EOS does not include the compressibility of the solid. The VdW EOS gives infinite pressure at $\rho = 1/b$, which is only a little above the input solid density. The code cannot be used to calculate shock waves in solids, unless a high-density EOS is supplied. Under the VdW EOS a solid may expand too rapidly and the user may find it useful to set the input "solid density" a bit above the initial material density to reduce the initial expansion speed.

The EOS does not contain any contribution from free electrons, so non-equilibrium of electron and ion temperatures cannot be described by the code.

These phenomena are addressed by the more elaborate DPC code, which has several EOS models, including an elaborate EOS based on the Saha equation (with density corrections). The appropriate EOS for non-equilibrium ("non-LTE") matter is in fact an unsolved research problem, especially for high-Z plasmas at high densities.

4.7.) Other EOS data

The user can replace the EOS look-up subroutines in DISH by other EOS data. QEOS data is widely available but the author thinks it is not very accurate in the WDM temperature range. Since the EOS used by DISH has the format $p(\rho, \varepsilon)$, $T(\rho, \varepsilon)$, translation software may be needed before another EOS data-source could be used. If other data is used, the user should inspect the two-phase region to determine whether the Maxwell construction has been performed and what is the overall smoothness of the EOS data.

5.) Driving energy

5.1.) Rapid flow is a consequence of high energy density

The phrase "hydrodynamics is driven by ..." suggests that something is necessary to cause a cold solid to move. In fact, unless the initial pressure is exactly zero, and unless the numerical method is very precise, a target will gradually disassemble unless there is some restoring force holding it together. In DISH there is no material strength. The Van der Waals EOS has a negative pressure region just below solid density, but this is removed or concealed by the Maxwell construction. So we expect a slow hydrodynamic disassembly even with no driving force. This expansion is probably computer-dependent (roundoff errors or failure to have perfectly zero pressure in the initial state). The user can experiment to see how slow it will be on his computer. Interesting target simulations will move much more rapidly, at speeds $\sim 10^5$ cm/sec.

Target simulations can be driven in several ways. We describe three typical methods. Users may invent other ways.

5.2.) Initial pressure

Rapid and interesting hydrodynamic motion occurs when the target (or part of it) is given a high initial temperature producing a high initial pressure. This type of driving is easily implemented in the code. As the initial configuration is decided, an initial temperature "T_{INIT}" is used. Normally, T_{INIT} would be room temperature ~ .025 eV, but initial temperatures up to ~ 1 eV can be considered. The code evaluates the EOS at the initial density and temperature and prints the initial energy/gram and initial pressure.

One of the test cases is a rarefaction flow driven by initial pressure. If the Van der Waals' EOS is being used, the user must be careful to avoid the unrealistic high pressures (or negative pressures) the VdW formulas assign to solid-density or above.

5.3.) Wall motion

A shock wave can be driven by forcing one side of the calculation to move with a specified velocity. This condition is imposed in the subroutine DELTX, which forms new zone positions x_j^{new}, and in DELTAV which forms new velocities v_j^{new}. The sample problems show how to set the variables IRW, ILW, VRW, VLW which control the wall motion. (IRW = right wall switch; VRW = right wall velocity in cm/sec).

5.4.) Deposition

Hydrodynamic calculations can also be driven by energy deposition. The coding for this goes inside the time-loop, with preliminary specification of a time-dependent power and/or space-dependent deposition profiles. The user should find coding for uniform deposition in the basic version of the code; this can be modified as desired.

The deposited power can be made time-dependent by using the current value of the variable "TIME" to evaluate any function f(time). Space-dependence can depend upon the current density profile. For 1-D Lagrangian zoning, the integral $\int \rho \, dx$, taken from the left boundary to any zone, is independent of time even though the zone positions and densities vary.

The deposited energy is converted into Joules/gram by multiplying the power by dtstep (= time step) and dividing by the zone mass. An example is given in the code.

6.) Zoning and time-steps

Different physical situations are best modeled with different zoning (zone spacing). As a general rule, one should anticipate the expected time-evolution and try to obtain roughly equally-spaced zones at the most important time. For a rarefaction (surface release) flow this is obtained by using very fine zones near the original solid surface which expand and separate as the material releases.

The DISH code has several simple subroutines to generate zones. The reader can easily extend these.

ZONER1 forms equal-spaced zones. This is most appropriate for a shock-wave which makes a uniform compression.

ZONER2 makes zones which are tapered toward the left ("front") surface. The user can determine the ratio of zone thickness in adjacent zones (typically 0.98). For using this zoning, the right side of the target is usually not permitted to move. If the right side is also allowed to move freely, for a uniform initial temperature, the difference between left- and right-side expansion shows the effect of the different zoning.

ZONER3 makes tapered zones on both sides of the target, a symmetrical configuration. The zones are placed by a hyperbolic-tangent formula. We urge the user to study this carefully before using it because changes in the number of zones require adjustment of a parameter β in the subroutine. If the initial surface zones are very thin, a very small time-step will be necessary.

ZONEFILL puts material into the zones after they have been made. Please note it also calls the EOS to decide the initial pressure, energy density, entropy, etc.

A subroutine PORZONE produces a multilayer "porous" target, i.e., a number of layers of solid density material spaced by gaps with low-density material in between. This subroutine makes the zones and loads material in them, so it does not require (or want) a separate call to ZONEFILL.

At present the time-step DTIME is a user-defined constant. The user will have an idea how long he wishes to follow the target dynamics, and will find that 10^5 - 10^6 time-steps is normally acceptable, and can adjust the number of time steps and DTIME accordingly.

The value chosen for DTIME can be tested several ways. First, a Courant condition is printed (a summary over the whole calculation is printed close to the end of the output page). Second, the "Bow-tie" count is printed. In a one-dimensional calculation the bow-tie problem (nonmonotonic Lagrangian zones) can occur when the time-step is too large. Third, strange interface zones for colliding surfaces (e.g., in the interaction of layers representing porous material) will disappear when the time-step is reduced.

7.) Important physics is omitted from DISH

Electron heat conduction (omitted because of dt issues/instabilities)

Unequal electron and ion temperatures (requires sophisticated EOS)

Ionization (requires detailed EOS and/or atomic data)

Radiation (requires opacity data, is not very important for WDM)

Non-equilibrium atomic physics (requires large amounts of atomic data)

Probe-laser interaction (not needed for non-laser experiments)

High-density behavior (requires elaborate EOS)

Solid state, solid-liquid transition; fracture, anisotropic mechanical properties

Non-equilibrium evaporation and condensation (evaporation kinetics)

These physical phenomena are (or *sometimes* are) included in calculations performed with the parent code DPC, which was originally intended as a testbed for non-equilibrium EOS and/or opacity models. In most cases the basic theory of these phenomena is imperfect and it does not seem appropriate to distribute experimental theoretical modules at this time. As the research into these phenomena proceeds, it is likely that the theory and code packages will advance.

8.) Tests, test-problems and examples

How do we know that our hydrodynamic code gets the "right" answer? To be correct in a physical sense, it would have to include all the relevant processes and accurately mirror the geometry of the specific experiment being considered. Obviously this is a lot to ask.

To be correct in a mathematical sense, it should produce an accurate solution to the planar Euler equations for the simplified case being considered. Testing and checking can help verify this.

A first important check is energy conservation. The code does not assume energy conservation, so the energy audits test the accuracy of the calculation. At any point in the calculation, one should have

Initial energy + added energy = current energy

The initial energy is usually "material" energy, i.e., the material energy (Joules/gram) from the equation of state (multiplied by the number of grams/cm^2).

The initial energy is an arbitrary number because the EOS has an energy zero fixed at some arbitrary initial density-temperature condition. For this reason when we make a percentage error test, it's best to examine the ratio

(current energy - initial energy) / added energy

The added energy is accumulated from deposition and/or possible piston work from wall movement. The current energy is material energy plus kinetic energy of the flow.

The kinetic energy must be evaluated by an approximate formula because in general there is a gradient of velocity across each zone. In DISH there are four choices for the kinetic energy per zone:

$$K_1 = \frac{1}{2} M_j \left(\frac{v_j + v_{j+1}}{2} \right)^2$$

$$K_2 = \frac{1}{2} M_j \left[\frac{v_j^2 + v_{j+1}^2}{2} \right]$$

$$K_3 = \frac{1}{2} M_j \left[\frac{v_j^2 + v_j v_{j+1} + v_{j+1}^2}{3} \right]$$

and

$$K_4 = \frac{1}{2} \left[\frac{M_j}{2} v_j^2 + \frac{M_{j+1}}{2} v_{j+1}^2 \right]$$

These four expressions should give the same answer for a finely-zoned problem but are slightly different for typical calculations. The code evaluates all four formulas (variables ek1, ek2, ek3 and ek4 in subroutine "audit") and currently prints K_3 = ek3, which performs nicely in several test-cases. The equation for K_3 is obtained by averaging v^2 for an assumed linear variation of velocity from x_j to x_{j+1}. The user who has strong opinions about this question can easily change the preferred choice or introduce a more accurate formula. In the actual coding, a factor 10^{-7} converts ergs/zone into Joules/zone.

The Courant condition is a comparison of the time-step dt to the zone size dx. For each zone we form

$$f = \frac{c_s dt}{dx}$$

using the sound speed c_s and zone thickness dx. We then form (and print) the "global" maximum of this ratio over all zones. f should be a small number for a satisfactory calculation.

Another intrinsic numerical quality test is the maximum ratio $dx/\Delta x$, the maximum fractional change in zone thickness during one time-step. If this were too large we should be nervous about the first-order numerical differencing methods used in the code.

Exact solutions are available for a few hydrodynamic flows, and these are more delicate tests of the numerical hydrdynamics. For shock-wave(s) in an ideal gas we have the Hugoniot formulas and exact predictions for the discontinuities in density, temperature, etc. For planar release flows (rarefaction waves), we have an exact solution, the Riemann solution of the planar ideal-gas hydrodynamic equations. This is an excellent test of the code. Typically one finds 1-2 % accuracy for calculations with 100 zones.

9.) Better calculations

This document has repeatedly emphasized the important limitations of the DISH code. If one wants to do better, one has several choices. First, one can use a more comprehensive hydrodynamic simulation code, if one has access to it. Second, one can make one's own improvements to the DISH code.

The author's DPC code includes a lot of additional physics. Many pieces of this additional physics are exploratory or intended for a narrow or specific application; they are not (today) general user-friendly tools. The list is

a.) Electron thermal conduction and electrical conductivity (based on subroutines which supply approximate values for the material properties $\kappa(\rho, T)$ and $\sigma(\rho, T)$ and contain provision for adjusting constants to match existing handbook data).

b.) Laser interaction and ellipsometry. This is handled by a Maxwell-equation solver which calculates interaction of a plane-wave light-beam (s- or p-polarized) with a planar target. The light has an arbitrary angle of incidence. The absorbed light can be used for deposition and the reflected light is analyzed into its polarization components for interpretation of ellipsometry experiments. Of course this calculation requires a model for the AC dielectric function $\varepsilon(\rho, T, \omega)$ for the target material. To fit experimental data this dielectric function must be adjusted appropriately. (It is not known *a priori* except for cold matter or for very hot plasmas.)

c.) EOS data. The author has a variety of EOS models, described in many publications, each plausible in some range of density-temperature conditions. Among these the QEOS code has achieved wide-spread use but is not entirely satisfactory for WDM conditions. A Saha model which uses a great deal of experimental input data is especially satisfactory for low-density gas-phase material.

d.) Non-equilibrium evaporation and condensation. There is a special version of the DPC code which calculates non-equilibrium 2-phase kinetics based on classical

formulas for evaporation kinetics. This model needs experimental testing in the WDM region and also needs to be compared to calculations using 2-D and 3-D hydrodynamic codes.

e.) Non-equilibrium EOS. This subject, the original motivation, remains imperfectly understood and imperfectly treated, in DPC as in all other hydrodynamic codes.

f.) Radiation phenomena. The DPC code has, or sometimes has, calculations of the emission spectra of specific ions (e.g., hydrogen visible line-emission) and has had the ability to do radiation heat conduction using a multifrequency opacity calculated by semiclassical methods. For hot plasmas these processes play a well-known and important role. In the WDM regime it is not expected that radiation will be important except as a diagnostic, and it is important to note that the usual methods of radiative transfer are not suitable for visible-light emission from dense plasmas. (The Maxwell-solver described in point "b" above is more appropriate for this application.)

ACKNOWLEDGEMENT: The author is grateful to Alex Zystra, the first DISH user, for finding and helping to fix a few difficulties with a first user-version of the code. Alex was able to run the code on a new computer (MAC G5) within a few minutes of his first effort and checked the various test cases against analytic formulas (generally good agreement).

REFERENCES

1.) R. D. Richtmyer and R. W. Morton, **Difference Methods for Initial Value Problems**, second edition, Interscience Publishers, J. Wiley & Sons, New York, 1967.

2.) Ya. B. Zel'dovich and Yu. P. Raizer, **Physics of Shock Waves and High Temperature Hydrodynamic Phenomena**, Dover Publications, 2002.

DISH - SUBROUTINE LIST

MAIN pogram

ARTVISC = form artificial viscosity

AUDIT = Add up mass and energy over zones

AUDTZON = check zones for bowties, ratio of min/max zone width

DELTAV = change velocities by pressure gradient

DELTAX = change positions of zones

EOST = call for EOS for given density, temperature

EOSE = call for EOS for given density, energy/gram

NEWOLD = copy "new" variables into "old" locations

PDEV = form pdV work during time-step

ZONER1,2,... = make initial zones (1 = uniform, 2 = one-side taper,
3 = two-side taper, 4 = porous layer structure)

ZONEFILL = load uniform material into the zones

ZTF = TF ionization curve-fit subroutine Z*(ρ, T)

VESETUP = set up parameters for VdW EOS

VEOSINT = interpolation for given ρ, E, includes Maxwell construction

VEVALV = evaluate VdW formulas for given ρ, E

VUNSCALE = read and scale table for VdW two-phase boundary
"TPHDAT" and write the scaled table as "RHOTPLANE"

VTPBS = search for T for given ρ, E inside the two-phase region

VTPBI = interpolate data to an ρ, T point inside the two-phase region

VFITTP = power-series/exponential fit to low-T two-phase VdW data

VTVALV = evaluate VdW formulas for given ρ, T