Fluid-Structure Interaction Simulations in Liquid-Lead

Simulations of the General Fusion Magnetized Target Fusion Reactor Concept

by

James Gregson

B.Eng., Dalhousie University, 2005

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

The Faculty of Graduate Studies

(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

August 2009

Abstract

An Eulerian compressible flow solver suitable for simulating liquid-lead flows involving fluid-structure interaction, cavitation and free surfaces was developed and applied to investigation of a magnetized target fusion reactor concept. The numerical methods used and results of common test cases are presented. Simulations were then performed to assess the smoothing properties of interacting mechanically generated shocks in liquid lead, as well as the early-time collapse behavior of cavities due to free surface reflection of such shocks. An empirical formula to estimate shock smoothness based on the shock smoothing results is presented, and issues related to shock driven cavity collapse in liquid liner magnetized target fusion reactors are presented and discussed.

Table of Contents

Appendices

List of Figures

Acknowledgements

Many thanks to my supervisor, Dr. Ollivier-Gooch and to the General Fusion team.

Chapter 1

Introduction and Background

1.1 Acoustically Driven Magnetized Target Fusion Reactor

The work presented in this thesis was carried out under a collaboration between General Fusion Ltd. and the Advanced Numerical Simulation Lab (ANSLab) at the University of British Columbia under an Industrial Postgraduate Scholarship from NSERC and MITACS. General Fusion Ltd. is (at the time of writing) a startup company with the aim of commercializing nuclear fusion as a method of power generation. To achieve this, an approach known as Magnetized Target Fusion (MTF) was chosen. MTF involves compressing plasma to the pressures and temperatures required to initiate a fusion reaction by confining it inside a collapsing solid or liquid liner. The choice of liner material and method of collapsing it varies — for example a solid liner may be crushed using explosives or electro-magnetically — however in the approach chosen by General Fusion, a liquid-lead liner will be compressed by a mechanically generated acoustic wave.

The design of the reactor is conceptually simple. A sphere of liquid lead approximately 3 m in diameter is enclosed in a steel reactor vessel and spun about the vertical axis at high angular velocity. The centripetal forces on the lead cause it to flow outwards in the radial direction, resulting in an evacuated cylindrical vortex along the axis of rotation. The vortex is similar to that formed by a draining sink. The evacuated vortex allows plasma injectors located at the poles of the sphere to inject a magnetized

plasma toroid into the cavity from either end at high velocity. When the two plasmas collide and merge in the center of the device their combined kinetic energy is converted to internal energy, causing an increase in temperature of the plasma.
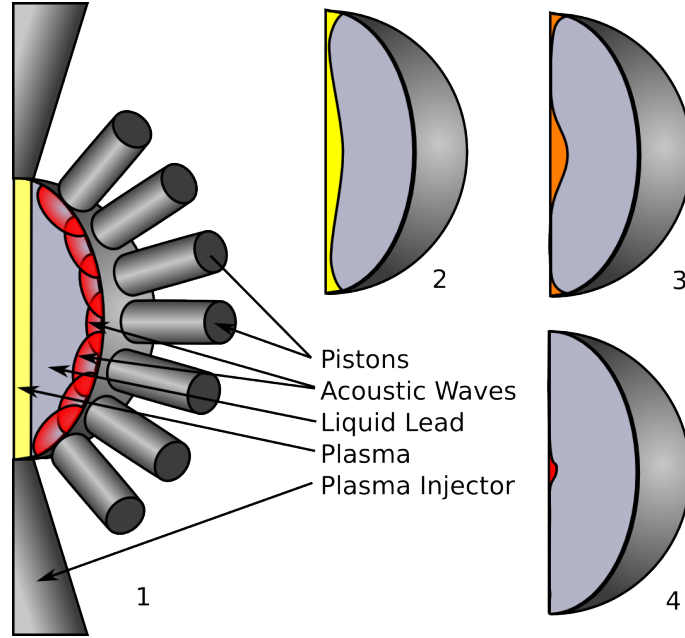
Figure 1.1: Depiction of General Fusion MTF reactor. 1. Reactor with pistons and plasma injectors showing acoustic waves from pistons. 2. Cavity beginning to close, pistons, plasma injectors and acoustic waves ommited for clarity. 3. Cavity at mid-compression, plasma density and temperature increasing. 4. Cavity at full compression, fusion burn ongoing

As the plasma is being injected, approximately 200 steam-driven pistons arranged around the outside surface of the reactor vessel (in a pattern similar to the dimples on a golf ball) ballistically impact the liquid lead liner. The pistons, 100 kg each, impact at 100 m/s and collectively have 100 MJ of energy. The impact lasts for approximately 100 μs, which gives an average rate of energy transfer over the impact time of 1 TW.

The impact of each piston forms an acoustic wave, which propagates

inwards in the radial direction from each piston face. While the individual waves propagate inwards, they interact with each other and form a single wave-front which sharpens into a strong shock due to geometric focusing. When the shock reaches and reflects from the lead-plasma interface, it imparts the lead at the interface with high radial velocity, causing the cavity to collapse rapidly. The collapse of the cavity is accelerated by geometric focusing, resulting in cavity wall velocities over 2 km/s at the end stages of collapse based on 1D simulations carried out by General Fusion prior to this project.

This thesis presents the numerical modeling that was performed to simulate the acoustic wave generation from piston impact, the propagation of the individual acoustic waves and the collapse of the plasma cavity. It focuses exclusively on the compressible hydrodynamic flows, leaving the issue of plasma dynamics and nuclear reactions to separate work that will be undertaken by General Fusion.

To perform this work, a custom inviscid compressible flow solver was developed which includes fluid-structure interaction capabilities, an equation of state for liquid lead and the development of a custom cavitation model. This thesis describes the numerics of that solver and discusses some of the simulation results obtained using it.

1.2 Flow Regimes and Modeling Considerations

There are several important flows in the concept MTF reactor. However, this work focuses only on the compressible aspects. Consequently, issues such as the vortex formation and cycling lead through the reactor to generate steam are left to follow up work by General Fusion.

Probably the most significant feature of the flow in the concept reactor is that it involves a compressible liquid. With flow velocities exceeding 2 km/s and pressures reaching 400 GPa, compressibility is unavoidable. Compressible, inviscid flows are governed by the Euler equations, and generally compressible flows in liquids are somewhat more complex to model than similar flows in gases due to equation of state complexity and the possibility

of cavitation occurring.

Cavitation adds a sharp non-linearity to the fluid constitutive relationship, and as such can present modeling challenges. More generally, obtaining an equation of state (the need for which is discussed in the following section) that expresses the liquid lead pressure in terms of its density and internal energy (or temperature) over the range of states expected to occur is difficult. An equation of state was found to model the lead, however its regime of applicability did not extend to cavitation. Since cavitation is one of the main features of the flows in the concept reactor, the lead equation of state was augmented with a new cavitation model.

Also present is fluid-structure interaction. Since the reactor uses mechanical pistons to generate the acoustic waves that crush the plasma cavity, it was necessary to include some capability for a structure to impose boundary conditions upon the lead liner. Cavitation complicates matters here too: it is important that the fluid near the structure be able to cavitate due to structural motion to get accurate results and because of this an uncommon test case was included to ensure solver accuracy.

Finally the plasma itself poses an issue since it is not actually governed by the Euler equations, falling instead into a class of flows referred to as Magneto-Hydrodynamics (MHD). Solvers capable of solving the MHD equations in domains with moving boundaries seem to be uncommon, although the developers of several codes such as NIMROD [20] and HIFI have mentioned that it should be possible with some development effort. Unfortunately simply becoming familiar enough with these solvers to operate them correctly can take considerable time to say nothing of the time required to add features to them. Given the time constraints on this project, it was decided that extending these solvers was not practical.

The plasma was consequently approximated (crudely) as a compressible fluid. Initially the plasma has extremely low density, on the order of 0.5 g/m^3 combined with temperatures over 1 million K. Due to this, the speed of sound in the plasma is extremely high, which poses problems for numerical methods that model shock propagation. Since it is shocks that drive the reactor, using such numerical methods was necessary and considerable

further approximation had to be made to the plasma state. As a result of these approximations, results obtained using the solver developed in Chapter 2 are only valid when the plasma pressure and density are essentially zero compared to that of the liquid lead. In spite of this, the results hold over a large portion of the cavity collapse and are adequate for exploring some of the issues that arise.

1.3 The Euler Equations

The Euler equations govern compressible inviscid flow. In two dimensions the conservative form of the equations in Cartesian coordinates are:

$$\frac{\partial U}{\partial t} = -\left(\frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y}\right) + S \tag{1.1}$$

where

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix} \tag{1.2}$$

$$F_x = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho u v \\ u(\rho E + P) \end{pmatrix} \tag{1.3}$$

$$F_y = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + P \\ v(\rho E + P) \end{pmatrix} \tag{1.4}$$

$$E = \quad e_i + \rho(u^2 + v^2)/2 \tag{1.5}$$

Here U is a vector of conserved variables, S is a vector of source terms and F_x, F_y are the flux vectors in the x and y coordinate directions. ρ is

the fluid density, $[u, v]^T$ is the fluid velocity vector, P is the pressure and E is the fluid total energy per unit mass, comprised of internal energy e_i and kinetic energy. The system of Equations (1.1-1.4) is not closed; there are four equations and five unknowns. To close the system, an additional equation must be supplied that links fluid pressure to one or both of fluid density and fluid internal energy. This additional equation is referred to as an equation of state since it links the thermodynamic state variables P, ρ and T, and serves as a means of introducing material-dependent behavior.

Integrating Equations (1.1) over a control volume gives the integral form of the Euler equations

$$\int_V \frac{\partial U}{\partial t} dV = -\int_V \left(\frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} \right) dV + \int_V S dV \qquad (1.6)$$

From the integral form of the equations, the divergence theorem may be used to convert the volume integrals for the flux terms in Equation (1.6) into surface integrals

$$\int_V \frac{\partial U}{\partial t} dV = -\int_A \left(F_x, \quad F_y \right) \cdot \left(\begin{array}{c} n_x \\ n_y \end{array} \right) dA + \int_V S dV \qquad (1.7)$$

which can in turn be rewritten

$$\int_V \frac{\partial U}{\partial t} dV = -\int_A F_n dA + \int_V S dV \qquad (1.8)$$

where F_n is defined as:

$$F_n = \left(\begin{array}{c} \rho q \\ \rho u q + P n_x \\ \rho v q + P n_y \\ (\rho E + P) q \end{array} \right) \qquad (1.9)$$

with $q = u n_x + v n_y$ and $[n_x, n_y]^T$ being the outward facing normal vector for the control-volume. At this point it is important to note that this form of the Euler equations only represents average values of density, momentum-density and energy-density within a control volume, not values at individual

points. This initially seems like a change for the worse, but actually forms the basis for the finite-volume method.

The source term S in Equation (1.1) and (1.8) can represent a variety of physical processes such as acceleration due to gravity or heat and mass exchange, but in this work it is used to transform the Euler equations from cartesian coordinate frames into axisymmetric and spherical coordinate systems without modifying the other terms in the equations. By setting S as in Equation (1.10), axisymmetric and spherical coordinate systems may be obtained simply by varying the parameter β.

$$S = \begin{pmatrix} -\beta\rho u/r \\ -\beta\rho u^2/r \\ -\beta\rho uv/r \\ -\beta u(\rho E + P)/r \end{pmatrix} \tag{1.10}$$

Setting β equal to one gives the Euler equations in an axisymmetric coordinate system about the y-axis, while setting β equal to two and all v velocities to zero gives a spherical coordinate system. Although not overly useful from an analytic standpoint, the ability to choose between coordinate systems on the basis of a single parameter is advantageous when constructing numerical methods, since effort does not have to expended developing custom numerical schemes for each individual coordinate system.

1.4 Riemann Problem in Compressible Flow

The flux function F_n of Equation (1.9) from Section 1.3 provides a mechanism for mass, momentum and energy to enter and leave a control volume. Changes to the contents of the control volume occur from the integral of the flux over the control-volume surface and the production terms S defined in Equation (1.10) integrated over the control-volume interior. However, the way that F_n is evaluated has a large effect on the stability of numerical methods for the Euler equations; for a code to be stable it must respect the physics of the Euler equations. In order to do this, the flux must be evaluated in a manner consistent with the solution to a compressible flow

Riemann problem.

A Riemann problem consists of two different states U_L and U_R initially separated by an interface which is removed at $t = 0$. U_L represents the state inside the control volume, and U_R the state outside, either determined by boundary conditions or by an adjacent control volume. At $t > 0$ when the interface is removed, three waves form with speeds S_L, S_M and S_R, where the wave moving at S_M is a contact wave (or material interface) and the other two waves can be either shocks or rarefactions depending on the initial left and right states. If the +x axis is mapped to the outward facing normal of the control volume with $x = 0$ corresponding to the control volume surface, the Riemann problem can be considered in 1D. In this case there are three characteristic waves that travel at $u - a$, u and $u + a$ (where a is the speed of sound) that correspond to the S_L, S_M and S_R waves. One possible wave configuration is shown in the space-time plot of Figure 1.2.
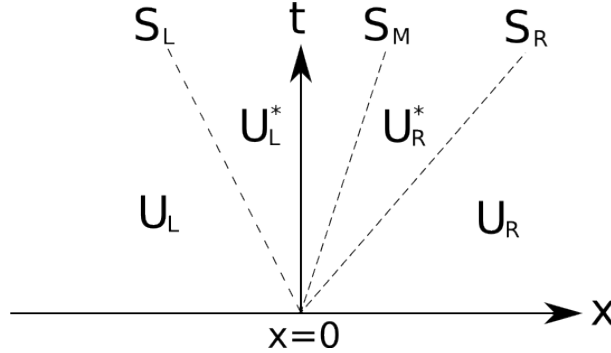
Figure 1.2: 1D Wave diagram of the Riemann problem showing the three waves and four states that result.

The S_L, S_M and S_R waves in the Riemann problem divide the solution into four regions of constant state. These are the initial left and right states denoted by U_L and U_R and two intermediate states, U_L^* and U_R^* which are either shocked or rarified versions of the U_L and U_R states, depending on the initial conditions. The U_L^* and U_R^* states are separated from the corresponding U_L and U_R states by the S_L and S_R waves and from each other by the S_M contact wave. The flux through the control volume surface depends

on which of these four states spans the interface, which is determined by the speeds S_L, S_M and S_R.
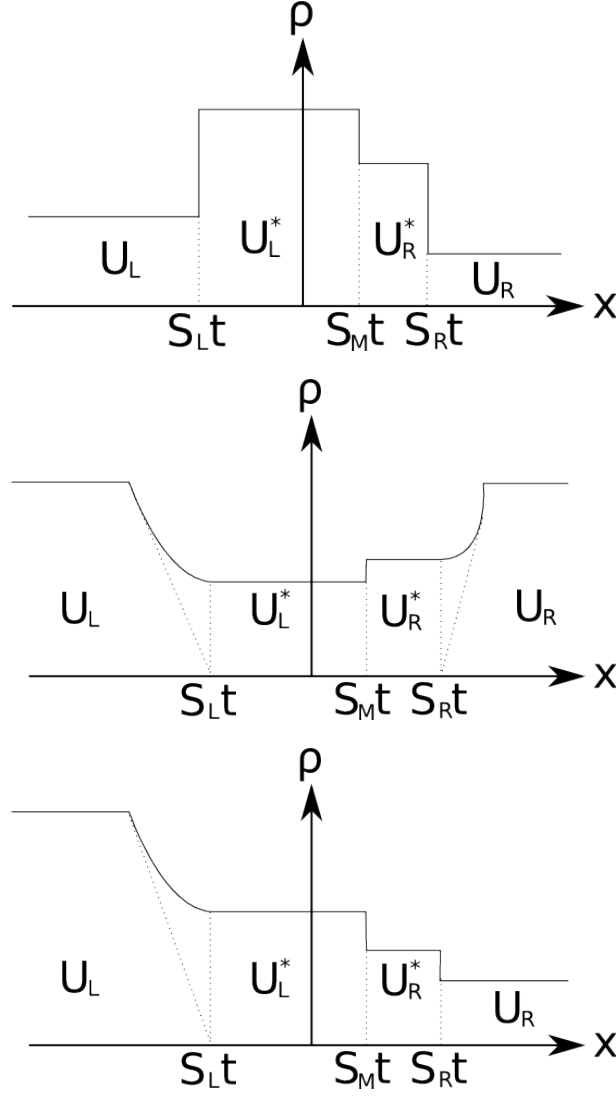
Figure 1.3: Wave type cases for the Riemann problem, top to bottom: two shocks, two rarefactions and one shock one rarefaction.

Physically there are three configurations of waves that may occur when vacuum is excluded: a contact wave with two shocks, a contact wave with

two rarefactions and a contact wave with a shock and a rarefaction. Density plots of these cases are shown in Figure 1.3. In all three cases the three wave speeds, S_L, S_M and S_R are needed to determine if the flow is subsonic or supersonic. If the flow is supersonic then only the state U_L or U_R that is upwind of $x = 0$ is needed to determine the flux. If the flow is subsonic, then the corresponding upwind U_L^* or U_R^* state is also required.

A Riemann solver is an algorithm that is used to calculate the flux through the control volume surface by determining the speeds of the three waves and the U_L^* and U_R^* states. Riemann solvers may be exact and solve the Riemann problem analytically, or approximate and obtain a linearized solution. Generally, approximate Riemann solvers produce suitably accurate results with much less computational time than exact Riemann solvers [21], and so are more often used in practice. In this work, the HLLC [22] approximate Riemann solver was used with wave speeds calculated as in Batten [2], which is discussed in Section 2.3.

The remainder of this thesis is organized as follows. Chapter 2 builds upon the Euler equations and Riemann problem presented in Chapter 1 to develop the numerical methods required to solve transient compressible inviscid flow problems, including a simple new cavitation model. Chapter 3 then presents results obtained using the solver, including test and validation problems and finally results obtained by applying the solver to two flow cases present in the General Fusion reactor concept. Chapter 4 concludes the thesis with discussion of some of the limitations of the solver presented in this work, along with suggestions for follow up work to extend the results presented in Chapter 3.

Chapter 2

Numerical Methods

This Chapter is presents the numerical methods used to simulate the transient compressible flows described in Section 1.2, namely the generation and propagation of acoustic and shock waves in liquid lead and the collapse of the plasma cavity. These numerical methods solve the time-dependent Euler equations from Section 1.3, and were developed using object-oriented C++.

2.1 The Finite-Volume Method

The unstructured finite-volume method (FVM) for the compressible Euler equations starts from the integral Euler equations (1.8) in Section 1.3 with flux F_n as defined in Equation (1.9) and source terms S defined in Equation (1.10). From this form the volume integrals may be simplified since the control volume form of the equation only tracks average values of density, momentum and energy, giving the continuum FVM as

$$\frac{\partial U}{\partial t} = -\frac{1}{V} \int_A F_n dA + S \tag{2.1}$$

This form of the FVM is not directly useful since analytic solutions for non-trivial cases are generally intractable. Consequently a discrete form is used which solves Equation (2.1) on a collection of domain-filling and non-overlapping control volumes (cells). Fluxes can then be evaluated at the interfaces (faces) between cells, and used to determine the time-derivative of the cell averages of the conserved variables (ρ, ρu, ρv and ρE) within the cell. During the flux evaluation step, the fluxes as evaluated from one side of a face must be of equal magnitude and opposite sign to those evaluated from the opposing side. If this condition is not met, the method will not

conserve the total mass, momentum and energy. This occurs because the faces have no volume in which to store discrepancies nor a method to return those discrepancies to the domain. Consequently any differences in flux as calculated in one cell compared to the opposing cell constitute aphysical losses or gains.

A common choice for cell geometry is to use triangles or quadrilaterals (in two space dimensions), since it is generally possible to both fill arbitrary domains with either while maintaining geometric restrictions on the shapes of the quadrilaterals/triangles that are necessary to obtain adequate solutions. The solver used in this work was developed to work with both triangles and quadrilaterals. Consequently all cells are polygonal with linesegments for faces. With the cell boundary defined to be piecewise linear, the integral over the surface of the cell becomes the sum of the integrals of the normal flux, F_n, over each face.

$$\frac{\partial U}{\partial t} = -\frac{1}{V} \sum_{i=1}^{N_{Faces}} \left(\int_{Face_i} F_{ni} dA_i \right) + S \tag{2.2}$$

The integral over each face is then computed via a Gauss quadrature rule of appropriate order for the order of the FVM. Throughout this work a second-order FVM was used which requires a single point quadrature rule with the Gauss point located at the center of each face. This results in the final semi-discrete form of the FVM for CV j as

$$\frac{\partial U_j}{\partial t} = -\frac{1}{V_j} \sum_{i=1}^{N_{Faces}} (F_{ni} A_i)_j + S_j \tag{2.3}$$

Equation (2.3) was implemented as a set of functions which evaluate the flux-integral and source terms and returns the solution time-derivative given an initial solution state and user-defined flux and source term functions.

2.2 Source Term Discretization

As described in Section 1.3, use of the geometric source terms defined in Equation (1.10 allows the transformation from Cartesian coordinates to axisymmetric or spherical coordinates. The source terms are straightforward to discretize to second-order accuracy and are evaluated using the control-volume averages at the cell centroids. i.e. for cell j

$$S_j = \begin{pmatrix} -\beta \rho_j u_j / r_j \\ -\beta \rho_j u_j^2 / r_j \\ -\beta \rho u_j v_j / r_j \\ -\beta u_j (\rho_j E_j + P_j) / r_j \end{pmatrix} \tag{2.4}$$

2.3 The HLLC Flux Solver

There were several reasons for choosing the HLLC solver over other solvers such as Roe's scheme [15] or AUSM+ [10]. Roe's scheme, which diagonalizes the equations of a primitive variable form of the Euler flux unfortunately couples the equation of state into the flux function itself. This means that new versions of Roe's scheme must be derived for every equation of state that will be used. This is awkward for practical use. In contrast, the AUSM flux solver does not couple the equation of state into the flux solver. It does however result in severe oscillations when used with liquids. HLLC has neither of these disadvantages, performing well for liquid and multi-material calculations [9] while not coupling the equation of state into the solver.

The HLLC Riemann solver defines the pressures and velocities of U_L^* and U_R^* to be equal to P^* and u^* respectively. This requires the contact wave speed S_M to also equal u^*, which maintains kinematic boundary conditions of continuous pressure and normal velocity on the contact surface.

The HLLC solver makes the further approximation that the Rankine-Hugoniot conditions may be applied across both the S_L and S_R waves. While this is a valid assumption when both waves are shocks, it is invalid for rarefactions where the fluid undergoes isentropic expansion, i.e. there is no discontinuous change in state. However in practice the solver has been

found to be adequate for most cases in spite of this inconsistency.

Once the Riemann solver has constructed an approximate solution, the flux through the face may be calculated by constructing the flux-vector F_n using the one or both of the two states that are upwind of the face. The flow is supersonic if either $S_L > 0$ or $S_R < 0$, in which case the flux should be constructed using the U_L or U_R states respectively. If the flow is subsonic and $S_M \geq 0$, the flow moves from the left to right and U_L and U_L^* should be used to create the flux. Otherwise the U_R and U_R^* state should be used. These conditions are listed in Equation (2.5).

$$
F_n = \begin{cases}
F_L, & S_L \geq 0 \\
F_R, & S_R < 0 \\
F_L^*, & S_L < 0 \text{ and } S_M \geq 0 \\
F_R^*, & S_R > 0 \text{ and } S_M < 0
\end{cases}
\tag{2.5}
$$

The first two cases of Equation (2.5) are respectively supersonic flow from the left and right, and the corresponding flux vectors are simply the analytic flux vector from Equation (1.9), e.g.:

$$
F_L = \begin{pmatrix}
\rho_L q_L \\
\rho_L u_L q_L + P_L n_x \\
\rho_L v_L q_L + P_L n_y \\
(\rho_L E_L + P_L) q_L
\end{pmatrix}
\tag{2.6}
$$

$$
F_R = \begin{pmatrix}
\rho_R q_R \\
\rho_R u_R q_R + P_R n_x \\
\rho_R v_R q_R + P_R n_y \\
(\rho_R E_R + P_R) q_L
\end{pmatrix}
\tag{2.7}
$$

For the subsonic cases, the fluxes F_L^* and F_R^* are found by applying the Rankine-Hugoniot conditions across the S_L and S_R waves respectively, giving

$$F_L^* = F_L + S_L(U_L^* - U_L) \qquad (2.8)$$

$$F_R^* = F_R + S_R(U_R^* - U_R) \qquad (2.9)$$

where the HLLC flux solver approximates U_L^* and U_R^* as

$$U_L^* = \frac{1}{S_L - S_M} \begin{pmatrix} \rho_L(S_L - q_L) \\ (S_L - q_L)\rho_L u_L + (P^* - P_L)n_x \\ (S_L - q_L)\rho_L v_L + (P^* - P_L)n_y \\ (S_L - q_L)\rho_L E_L - P_L q_L + P^* S_M \end{pmatrix} \qquad (2.10)$$

$$U_R^* = \frac{1}{S_R - S_M} \begin{pmatrix} \rho_R(S_R - q_R) \\ (S_R - q_R)\rho_R u_R + (P^* - P_R)n_x \\ (S_R - q_R)\rho_R v_R + (P^* - P_R)n_y \\ (S_R - q_R)\rho_R E_R - P_R q_R + P^* S_M \end{pmatrix} \qquad (2.11)$$

This gives a complete approximate solution to the Riemann problem provided the wave speeds S_L, S_M and S_R are known as well as the pressure P^*. HLLC calculates these as

$$S_L = min(q_L - a_L, q - a) \qquad (2.12)$$

$$S_R = max(q_R + a_R, q + a) \qquad (2.13)$$

$$S_M = \frac{\rho_R q_R(S_R - q_R) - \rho_L q_L(S_L - q_L) + P_L - P_R}{\rho_R(S_R - q_R) - \rho_L(S_L - q_L)} \qquad (2.14)$$

$$P^* = \rho_L(q_L - S_L)(q_L - S_M) + P_L$$

$$= \rho_R(q_R - S_R)(q_R - S_M) + P_R \qquad (2.15)$$

where

$$q_L = u_L n_x + v_L n_y \tag{2.16}$$

$$q_R = u_R n_x + v_R n_y \tag{2.17}$$

$$q = (q_L + q_R)/2 \tag{2.18}$$

$$a = (a_L + a_R)/2 \tag{2.19}$$

The HLLC flux solver algorithm can be summarized as follows. Compute the wave-speeds S_L and S_R. If the flow is supersonic from the left or right as determined by Equation (2.5), build the flux vector using Equation (2.6) or (2.7) respectively. Otherwise, compute the contact wave-speed S_M and pressure in the star region P^* using Equation (2.14) and (2.15). Based on the sign of S_M construct U_L^* or U_R^* using Equation (2.10) or (2.11) and the appropriate flux F_L or F_R from Equation (2.6) or (2.7). Finally compute F_L^* using F_L and U_L^* or F_R^* using F_R and U_R^* from Equation (2.8) or (2.9).

2.4 Solution Reconstruction and Limiting

In order for the finite volume method to function, fluxes between cells must be calculated at the faces. These fluxes are then added and subtracted from the neighboring cells such that flux out of one cell is flux into its neighbor. Fluxes at faces are calculated using the HLLC approximate Riemann solver (described in Section 2.3), which takes as input the thermodynamic state of the the cells adjacent to that face. However the thermodynamic states stored by the solver are control volume averages of solution variables and are not located at the interfaces between cells. Consequently the issue arises of what state to provide to the HLLC solver, given that the state as represented by the control volume averages is unlikely to be the state that should exist at the interfaces between cells.

$$\Phi_{1^{st}}(x_{GP}, y_{GP}) = \Phi_R \tag{2.20}$$

The simplest choice is to use the control volume averages unchanged

as in Equation (2.20), where $[x_{GP}, y_{GP}]^T$ is the point where the value of the solution variable Φ is desired and Φ_R is identical to the control volume average Φ_0 of the cell being reconstructed. This is a first order approach since the state within a cell is assumed constant and tends to causes excessive diffusion and smearing of sharp features in the solution. Large amounts of diffusion in the solution are unacceptable since the major features of interest in this work are shocks and material interfaces, which are for all intents and purposes physically discontinuous. An example of a first order reconstruction is shown in Figure 2.1.

Increasing the order of the method reduces the numerical smearing and diffusion of the solution, but does so at the cost of increased computational time and memory use. This work uses a second order method which is the most prevalent approach for transient unstructured simulations of compressible flows. Second-order methods are popular because they yield good results with reasonable computational resources while remaining straightforward to implement.
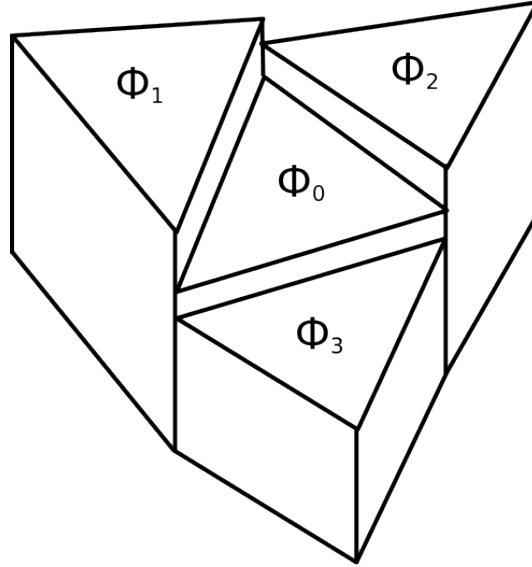

Figure 2.1: Example of first order reconstruction, values of Φ are constant within cells.

The second order method requires approximating each solution variable within a cell as a plane as in Figure 2.2 rather than as constant, i.e.

$$\Phi_{2^{nd}}(x_{GP}, y_{GP}) = \Phi_R + (x_{GP} - x_0)\frac{\partial \Phi}{\partial x} + (y_{GP} - y_0)\frac{\partial \Phi}{\partial y} \qquad (2.21)$$
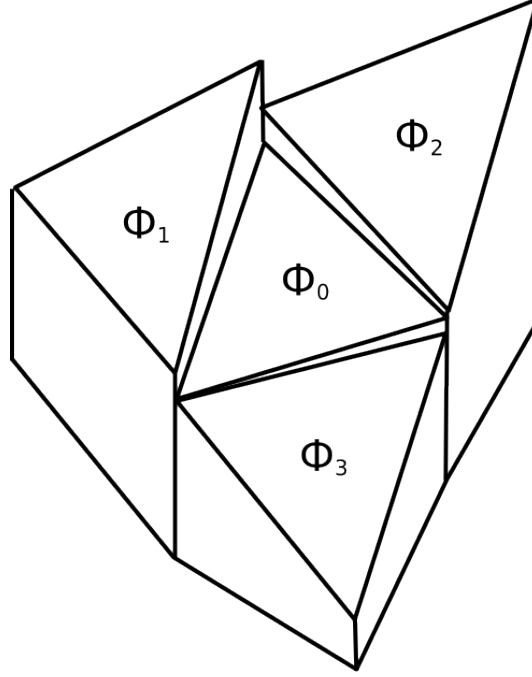
Figure 2.2: Example of second order reconstruction, values of Φ are planar within cells.

where $[x_0, y_0]^T$ is the control volume centroid. Although Equation (2.21) provides a second-order reconstruction of the field variable Φ, it also has the unfortunate property that it does not preserve the monotonicity of Φ. This means that it introduces nonphysical overshoots and undershoots around sharp features, an example of which is shown in Figure 2.3. Preserving monotonicity is essential for numerical methods to remain stable and makes sense intuitively; if overshoots or undershoots are introduced at sharp fea-

tures, then those features become sharper, which causes more (and more serious) overshoots and undershoots to occur.
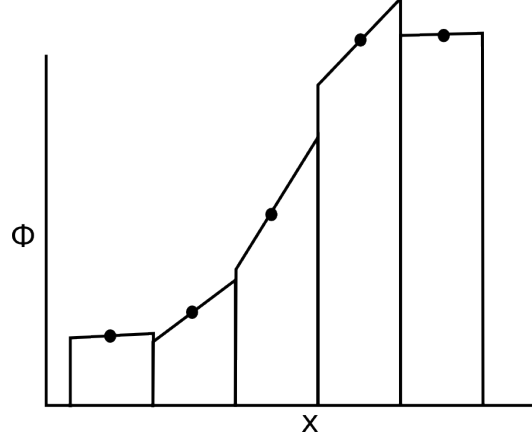
Figure 2.3: An unlimited second-order reconstruction showing overshoots.

To preserve monotonicity a limiter is introduced. The limiter is simply a scaling factor for the gradient that is adjusted dynamically for each cell and each field variable to constrain the reconstructed value to a range of admissible values. This range is simply the minimum and maximum of the control volume averages at the current cell and its neighbors; constraining the reconstructed value to this range prevents new extrema from being introduced. This is a simple change to introduce into the reconstruction, Equation (2.21) simply becomes Equation (2.22). A version of Figure 2.3 with limiting applied is shown in Figure 2.4.

$$\Phi_{2^{nd}}(x_{GP}, y_{GP}) = \Phi_R + \Psi \left((x_{GP} - x_0)\frac{\partial \Phi}{\partial x} + (y_{GP} - y_0)\frac{\partial \Phi}{\partial y} \right) \qquad (2.22)$$

where Ψ is a real number in the range $[0, 1]$. Ψ can be thought of as an interpolation weight that applies the second-order correction to the reconstruction according to the smoothness of the underlying field Φ. Regions where the solution is smooth will have $\Psi = 1$, while regions near shocks or
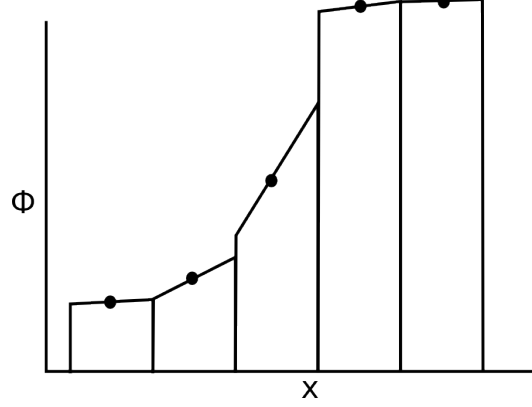
Figure 2.4: With limiting, the second-order reconstruction does not have overshoots

material interfaces have $\Psi = 0$. This has the effect of reducing the order of the numerical method to first order near discontinuities in accordance with Godunov's theorem [6].

Leaving the actual computation of $\frac{\partial \Phi}{\partial x}$, $\frac{\partial \Phi}{\partial y}$ and Ψ to later in this section, the reconstructed state at the interface between cells can then be computed for each cell as follows

$$\rho_{GP} = \rho_{2nd}(x_{GP}, y_{GP}) \tag{2.23}$$

$$u_{GP} = u_{2nd}(x_{GP}, y_{GP}) \tag{2.24}$$

$$P_{GP} = P_{2nd}(x_{GP}, y_{GP}) \tag{2.25}$$

$$e_{i_{GP}} = EOS.InternalEnergy(\rho_{GP}, P_{GP}) \tag{2.26}$$

where ρ_{2nd}, u_{2nd} and P_{2nd} are reconstructions in the form of Equation (2.22) and $EOS.InternalEnergy(\rho, P)$ is a function of the equation of state that returns the internal energy of a material for a given density and pressure. These values can then be used to compute the conserved variables $(\rho u)_{GP}$ and $(\rho E)_{GP}$ needed by the HLLC solver.

The reason for reconstructing the cell state from the variables ρ_{GP}, u_{GP}

and P_{GP} rather than the components of U is that as long as ρ and P are positive, the reconstructed values of e_i and a will also be positive and bounded by the control volume averages adjacent to the cell. If the conservative variables were reconstructed directly, there is no guarantee that the pressure at the Gauss points would be positive and monotone, since the conservative variable ρE is defined in terms of density, velocity and internal energy and unfortunate combinations of the three could yield non-physical negative pressures and undefined speed of sound.

With a method established for reconstructing the solution at Gauss points based on the adjacent control volume averages of solution variables, solution variable gradients and limiter values, there remains the issue of how to determine the gradients and limiter values. To compute gradients there are essentially two methods, least-squares and Green-Gauss. The Green-Gauss approach makes use of the divergence theorem to compute the gradient using the surface integral of the variable over the cell boundary as in Equations (2.29) and (2.30).

$$
\int_V \frac{\partial \Phi}{\partial x} dV = \int_A \Phi n_x dA \tag{2.27}
$$

$$
\int_V \frac{\partial \Phi}{\partial y} dV = \int_A \Phi n_y dA \tag{2.28}
$$

or in discrete form with

$$
\frac{\partial \Phi}{\partial x} = \frac{1}{V} \sum_{i=1}^{N_{Faces}} \Phi_i A_i n_{x_i} \tag{2.29}
$$

$$
\frac{\partial \Phi}{\partial y} = \frac{1}{V} \sum_{i=1}^{N_{Faces}} \Phi_i A_i n_{y_i} \tag{2.30}
$$

where the subscript i indicates the value at the Gauss point of face i, and Φ_i is approximated as the average of the adjacent control volume averages.

An alternative to the Green-Gauss technique is to use a least squares

method to generate a reconstruction of the form of Equation (2.21). Given the cell centroids and control-volume averages of the solution to be reconstructed, this reconstruction can be obtained to second order accuracy as the solution to

$$
\begin{pmatrix}
1 & x_0 & y_0 \\
1 & x_1 & y_1 \\
1 & x_2 & y_2 \\
1 & \vdots & \\
1 & x_n & y_n
\end{pmatrix}
\begin{pmatrix}
\Phi_R \\
\frac{\partial \Phi}{\partial x} \\
\frac{\partial \Phi}{\partial y}
\end{pmatrix}
=
\begin{pmatrix}
\Phi_0 \\
\Phi_1 \\
\Phi_2 \\
\vdots \\
\Phi_n
\end{pmatrix}
\tag{2.31}
$$

where the subscript indicates the control volume, zero for the control volume being reconstructed and non-zero for all other control volumes in the reconstruction stencil. Equation (2.31) has one problem however, since the value of Φ_R obtained may not be the control-volume average Φ_0. This can be remedied by subtracting the first Equation of the system from all subsequent Equations, to obtain

$$
\Phi_R = \Phi_0
$$

$$
\begin{pmatrix}
x_1 - x_0 & y_1 - y_0 \\
x_2 - x_0 & y_2 - y_0 \\
\vdots & \\
x_n - x_0 & y_n - y_0
\end{pmatrix}
\begin{pmatrix}
\frac{\partial \Phi}{\partial x} \\
\frac{\partial \Phi}{\partial y}
\end{pmatrix}
=
\begin{pmatrix}
\Phi_1 - \Phi_0 \\
\Phi_2 - \Phi_0 \\
\vdots \\
\Phi_n - \Phi_0
\end{pmatrix}
\tag{2.32}
$$

Equation (2.32) centers the reconstruction at the control-volume centroid and sets the constant term of the reconstruction to be the control-volume average. For a second order reconstruction this is adequate to ensure that the integral of the reconstructed variable over the control volume is equal to the control-volume average of that variable. It also has the added advantage of removing a degree of freedom from the system, which makes Equation (2.32) more efficient to solve numerically than Equation (2.31). The system (2.32) is then solved using either singular value or QR decomposition to obtain the reconstruction terms $\frac{\partial \Phi}{\partial x}$ and $\frac{\partial \Phi}{\partial y}$.

One thing to note is that just forming the least-squares system in Equation (2.32) is of comparable expense to computing the gradients using the Green-Gauss method. As a result of this the least squares approach is considerably more expensive. This can be offset for stationary meshes by storing the (SVD or QR) decomposition of Equation (2.32) at each cell since it only depends on the mesh geometry. Then each time the reconstruction needs to be updated the new Gauss point values of the solution variables can be found via a solve of the factored system. However for meshes with large numbers of cells and large reconstruction stencils the storage of the decompositions can be a significant fraction of the total memory used.
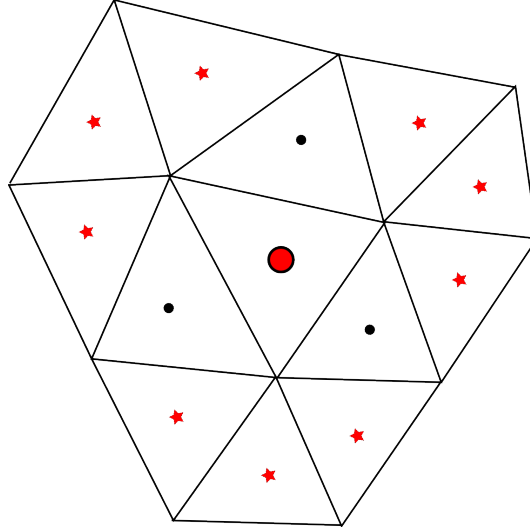
Figure 2.5: Stencil used during solution reconstruction of center cell (large dot). Green-Gauss: adjacent cells (dots). Least-Squares: Cells sharing vertex (stars and dots).

The least-squares approach will generally give more accurate gradients [13] than the Green-Gauss method, particularly when large stencils are used such as the one-ring shown in Figure 2.5. Since the Green-Gauss method operates over the surface of the control volume, using a larger stencil is difficult if not impossible. For the time-dependent shock simulations on

isotropic meshes in this work where material properties dictate large numbers of control-volumes and time-steps it was found that the minor improvements in gradient reconstruction from using the least-squares method were not worth the additional expense in either memory or runtime. Consequently the Green-Gauss procedure was used.

Once the reconstruction parameters Φ, $\frac{\partial \Phi}{\partial x}$ and $\frac{\partial \Phi}{\partial y}$ calculated, the limiter value Ψ can be found. The Barth–Jesperson [1] procedure is followed where the value of Ψ is chosen such that the reconstructed values of the solution variable at the face Gauss points falls within the range $[\Phi_{min}, \Phi_{max}]$, where Φ_{min} and Φ_{max} are the minimum and maximum of control-volume averages at the reconstructed cell and its immediate neighbors, i.e.

$$\Phi_{min} \leq \Phi_R + \Psi \left((x_{GP} - x_0)\frac{\partial \Phi}{\partial x} + (y_{GP} - y_0)\frac{\partial \Phi}{\partial y} \right) \leq \Phi_{max} \qquad (2.33)$$

Ψ is further restricted to lie within the range $[0, 1]$. Although this method does not restrict the value of Φ_{GP} for a given face to fall within the range specified by the adjacent cells, this procedure has been shown to be effective at reconstructing solutions on unstructured meshes without causing spurious overshoots.

2.5 Equations of State

As mentioned in Section 1.3 the Euler Equations in 2D consist of four conservation equations (density, energy and two components of momentum) in five unknowns (ρ, u, v, e_i and P). In order to solve the system, an additional equation of state (EOS) must be added that links density and/or internal energy to pressure. It is the equation of state alone that differentiates between whether water, air or liquid-lead is being simulated; all the other parts of the solver remain the same.

The majority of this work involved using liquid-lead as a working fluid, and consequently it was necessary to find an appropriate equation of state for liquid lead. However, while developing the solver it was also necessary

to validate the various components, and few test cases exist in the open literature for shocks in liquid lead. Consequently two other Equations of state were implemented to ensure the accuracy of the solver on standard test cases. These EOSes were the ideal gas EOS for gases and the Tait EOS for liquids.

2.5.1 Ideal-Gas Equation of State

The ideal gas EOS can be used to represent a variety of gases up to moderately high pressures and temperatures. It is one of the simplest EOSes and has the form

$$P = (\gamma - 1)\rho e_i \tag{2.34}$$

where γ is the adiabatic index. The speed of sound can be found by assuming isentropic compression and expansion and taking the square root of the partial derivative of pressure with respect to density, giving

$$a = \left(\frac{\partial P}{\partial \rho}\right)^{\frac{1}{2}} = \left(\frac{\gamma P}{\rho}\right)^{\frac{1}{2}} \tag{2.35}$$

2.5.2 Tait Equation of State

The Tait equation of state can be used to model liquids, and is frequently used to model water in underwater-explosion simulations [9], [5]. It has the form

$$P = B\left(\left(\frac{\rho}{\rho_0}\right)^n - 1\right) + A \tag{2.36}$$

where for water the constants $\rho_0 = 1000.0 \ kg/m^3$, $A = 101325.0Pa$, $B = 331.0MPa$ and $n = 7.15$. The speed of sound is calculated as

$$a = \left(\frac{\partial P}{\partial \rho}\right)^{\frac{1}{2}} = \left(\frac{nB\left(\frac{\rho}{\rho_0}\right)^n}{\rho}\right)^{\frac{1}{2}} \tag{2.37}$$

The Tait EOS differs from the Ideal-Gas EOS in one important aspect. Since it involves only pressure and density but not energy, the Tait EOS is barotropic. This closes the Euler equations but decouples the energy equation from the mass and momentum equations, so that simulations may be carried out with only the mass and momentum equations. However there is no harm in also solving the energy equation, which is done throughout this work.

2.5.3 Robinson's Lead Equation of State

To model liquid lead it was necessary to obtain an equation of state which was valid from vacuum up to approximately 4 MBar. For this extreme pressure range there were very few options. The best choice was due to Robinson [14], who developed a EOS based on classical thermodynamics which was fit to the results of gas-gun experiments where projectiles were fired at high velocity into lead targets. With known projectile velocities and corresponding measured shock velocities, he obtained points on the particle-velocity shock-velocity Hugoniot curve. Robinson used the Hugoniot curve data points, the Rankine-Hugoniot equations and an assumption of a linear shock-velocity to particle-velocity relationship to fit model parameters to an analytic expression for the Helmholtz free energy. That curve was integrated numerically and the integrated expression for free energy forms a semi-analytic equation of state. Pressure at a given density may then be obtained from the equation of state by numerically differentiating the Helmholtz free energy curve with respect to specific volume. This was done for 100,000 points spanning the range of densities from $\rho = 10644.0$ to $\rho = 31932.0$, yielding the tabulated pressure-density curve $F(\rho)$ used in Equation (2.39). More information on the formulation and assumptions of the Robinson equation of state can be found in [14].

$$P = F(\rho) \tag{2.38}$$

$$a = \left(\frac{F(\rho + \epsilon) - F(\rho - \epsilon)}{2\epsilon} \right)^{\frac{1}{2}} \tag{2.39}$$

$$\epsilon \ll 1$$

This equation of state is also barotropic, since pressure does not depend on internal energy. Like the Tait EOS, the energy equation could also be omitted when solving compressible flow problems using this EOS, however, it was left in for consistency.

2.5.4 Cavitation

One issue that arises when modeling liquids (including lead) that does not occur with gases is the possibility of cavitation. Cavitation occurs when the pressure of fluid drops below the cavitation pressure, resulting in a portion of the liquid to change phase into a vapor. When this occurs the properties of the fluid change drastically; even small amounts of vapor results in a dramatic drop in liquid bulk-modulus and speed of sound. The Tait equation of state for liquid water and Robinson equation of state for liquid lead do not account for this occuring; they can actually allow large nonphysical negative values of pressure.

To prevent these large negative pressures and to capture the effects of cavitation, cavitation models are commonly used. The simplest of these is a pressure cut-off which restricts pressure to either the cavitation pressure of the liquid or zero. Cut-off models have been used for some time particularly in acoustics, but can artificially blur the boundaries of cavitated regions in hydrodynamics simulations [11].

So-called one-fluid models have been developed to provide a more accurate alternative to the cutoff model. One-fluid models attempt to model the variation in pressure and speed of sound with density directly using a single fluid without resorting to full multiphase methods such as Saurel's [16] which represent the liquid and vapor phases distinctly and model the

transitions between them. One-fluid models are computationally more efficient than multiphase approaches and provide improved results compared to the cutoff model.

For this work, multiphase approaches were rejected early on for several reasons, the first being the more than doubling of computational and memory cost to solve two distinct fluid phases and the transfer of mass, momentum and energy between them. The second reason was the difficulty of determining closure coefficients for the phase exchange in liquid lead. In order for multiphase methods to function, these coefficients must be specified and the key advantage of multiphase methods (in the author's opinion) is their ability to model the physical processes involved in cavitation rather than the effects. Without accurate closure coefficients the models have much less of an advantage over the alternatives. As a result of this it was decided to use a one-fluid model.

Two one-fluid models were investigated: Schmidt's model [17] and Liu's isentropic model [11]. Both of these models are based around the relations

$$\frac{\partial P}{\partial \rho} = a^2 \tag{2.40}$$

$$a = \left[(\rho_L(1-\alpha) + \rho_G\alpha) \left(\frac{1-\alpha}{\rho_L a_L^2} + \frac{\alpha}{\rho_G a_G^2} \right) \right]^{-\frac{1}{2}} \tag{2.41}$$

where the subscripts L and G denote liquid and gas respectively, α is the volume fraction of gas and $[\rho_G, a_G]$ and $[\rho_L, a_L]$ are liquid/gas densities and speeds of sound, assumed either constant for Schmidt's model or to follow isentropic compression and expansion in Liu's model. As is pointed out by Liu [11], Schmidt's model suffers from requiring an abnormally small timestep to remain stable, as well as requiring non-physical values for the cavitation pressure. In contrast, implementing Liu's model proved more difficult than expected, and a complete, functioning implementation was not achieved.

Lacking an existing one-fluid model that was numerically efficient and straightforward to implement, a hybrid one-fluid, cutoff model was developed that avoided the timestep restrictions of Schmidt's model without the

complexity of Liu's model. The hybrid model assumes (incorrectly) that the liquid-vapour mixture pressure can be expressed as a linear function of the vapor volume fraction. The speed of sound is calculated by assuming (as in Schmidt's model) that the liquid and vapor densities and speeds of sound are known and constant. The hybrid model is summarized in Equations (2.42-2.44) and was used for all simulations presented in this thesis.

$$\rho = \alpha\rho_G + (1-\alpha)\rho_L \tag{2.42}$$

$$P = (1-\alpha)P_{Cav} \tag{2.43}$$

$$a = \left[(\rho_L(1-\alpha) + \rho_G\alpha)\left(\frac{1-\alpha}{\rho_L a_L^2} + \frac{\alpha}{\rho_G a_G^2}\right)\right]^{-\frac{1}{2}} \tag{2.44}$$

P_{Cav} is the pressure at which the fluid begins to cavitate. The hybrid model is essentially a cutoff model for pressure and a one-fluid model for soundspeed, except that monotonically increasing pressure with density assures that the Euler equations remain hyperbolic if not thermodynamically consistent (since $\frac{\partial P}{\partial \rho} \neq a^2$). Difficulties with degeneration of the Euler equations were noted in Fedkiw [5] where a centered discretization was used for cavitated regions. Unlike in Fedkiw, centered discretization of cavitated regions is not required for the hybrid model to remain stable.

For this work, thermodynamic consistency, while nice, is not essential; the lowest peak pressure in the domain during piston impact is approximately 2 GPa, while the potential error in pressure is on the order of 100 KPa, more than four orders of magnitude lower. This difference grows to six orders of magnitude at later times. It is highly unlikely that the lead equation of state is fit to 100 KPa accuracy at even modest volumetric strains, so focusing on this comparatively minor source of error did not seem a good use of effort.

The new hybrid model does not have the timestep restrictions of Schmidt's model but is able to model the dramatic decrease in mixture speed of sound with cavitation. Results of a benchmark case from Saurel [16] show it to be sufficiently accurate (see Section 3.1.3). To summarize the hybrid model

uses the assumption of fixed liquid and gas states in the cavitated fluid which uniquely determines the volume fraction of each and the soundspeed of the mixture. Unlike the Schmidt and Liu one-fluid models (both of which require $\frac{\partial P}{\partial \rho} = a^2$), the hybrid model instead uses a cut-off approximation for determining pressure, avoiding timestep restrictions and implementation complexity. Due to the high range of pressure seen in the simulations compared to the range of pressure where fluid is cavitated, this cut-off pressure approximation does not introduce significant errors into the simulations. As a result, the hybrid model is efficient and accurate for the flows encountered in this project.

2.5.5 Plasma Equation of State

Originally it was hoped that simulations of the full reactor could be carried out. These would be somewhat crude approximations since the plasma is governed by the Magneto-Hydrodynamics (MHD) equations rather than the Euler equations. Regardless it was thought that a basic model of the plasma as a compressible, low density and high temperature gas would provide some insight into the flow during the later stages of collapse. Knowledge of the late stages of collapse is important, since it has the potential to identify the causes and methods of mitigating hydrodynamic instabilities which could quench the plasma before the fusion burn begins.

Unfortunately, although numerical methods exist to handle mixtures of materials, e.g. [9], the gas state required to approximate the plasma has a sufficiently high speed of sound that the simulations quickly become intractable. Using the ideal gas equation of state with representative plasma values just after injection $\gamma = 5/3$, $\rho = 4.59(10^{-4})kg/m^3$ and $T = 1.1(10^6)K$ results in a speed of sound of around $78km/s$. This is more than 45 times the speed of sound in the lead, which in turn implies a 45 times decrease in the maximum stable timestep for a given mesh. The timestep limit can be overcome through the use of implicit time advance, however that comes at the price of much higher cost per timestep and would still result in a considerable increase in runtime, to say nothing of the added difficulty implementing an

efficient implicit multi-material solver for plasma and cavitated liquid lead.

As a result of these issues, it was decided to approximate the plasma at the early stages of collapse as a vacuum, or more accurately, a highly rarified liquid lead with density of the order of $1kg/m^3$. This is not as bad an approximation as it may seem at first, since early in the collapse process the plasma density and pressure are very small, $\rho = 4.59(10)^{-4}kg/m^3$ and $P = 1.68MPa$, while the corresponding density and pressure in the lead are $10644.0kg/m^3$ and $2.0GPa$. At these density and pressure ratios the plasma may as well not exist for its influence on the flow of liquid lead, and the use of rarified lead to represent the plasma means a single phase solver with no mixture model may be used.

As the plasma is compressed this approximation begins to break down and eventually become patently false when the pressure in the plasma is sufficient to slow and finally reverse the flow of the lead. However this occurs relatively late in the collapse process when the radius of the cavity is approximately 3-4 cm compared with the initial 20 cm, so a reasonable indication of the evolution of the cavity shape may be obtained before the breakdown of this assumption.

2.6 Time Integration

With equations of state for the materials and numerical methods in place to reconstruct the solution and calculate fluxes, the time-derivative of the solution variables may be found using Equation (2.3). All that remains to have a functional compressible flow solver is a method for advancing the solution in time.

For transient inviscid compressible flows where the evolution of shocks and material interfaces over time is desired, the most common form of time advance is explicit time-stepping. Explicit time-stepping takes the solution at a given time $t = t_0$ and expresses the solution at a later time $t = t_0 + \Delta t$ as a function of the solution(s) at time(s) $t < t_0 + \Delta t$. This means that the solutions required to step forward in time are already known, and the solution update $\frac{\partial U}{\partial t}$ can be evaluated directly rather than determined

implicitly as the solution to a system of (nonlinear) equations. The simplest form of explicit time-integration is the forward Euler method

$$U^{t+\Delta t} = U^t + \left(\frac{\partial U}{\partial t} \Delta t \right)^t \tag{2.45}$$

where the superscripts indicate the time at which each term exists. Writing this out as in Equation (2.3) we obtain the following for the update for a single cell j.

$$U_j^{t+\Delta t} = U_j^t + \Delta t \left(-\frac{1}{V_j} \sum_{i=1}^{N_{Faces}} (F_{ni} A_i) + S \right)_j^t \tag{2.46}$$

Since fluxes are evaluated using values at time t from cell j and its neighbors (and neighbor's neighbors for second-order Green-Gauss reconstruction), $U_j^{t+\Delta t}$ may be evaluated directly rather than obtained from matrix inversion. This is straightforward and efficient to compute, however for the scheme to remain stable the timestep Δt must be kept small. This has the effect of limiting the distance along characteristics that information can propagate in a single timestep to one cell length. If the timestep Δt is too large and allows information to propagate further than one cell per timestep, cells being to overfill and overempty aphysically, which results in severe oscillations of the solution and generally causes the solver to crash.

To limit the timestep a maximum signal propagation velocity is calculated for each cell. This velocity is simply the sum of the speed of sound and the velocity magnitude, corresponding (conceptually) to the $u + a$ characteristic from the solution to the Riemann problem. With this signal propagation velocity the maximum stable timestep for a control-volume j can be computed as the characteristic length of the control-volume L_j divided by the signal propagation velocity $|u_j| + a_j$, i.e.

$$(\Delta t_{max})_j = L_j / (|u_j| + a_j) \tag{2.47}$$

For axis-aligned 2D structured meshes, the characteristic length of the control-volume can be approximated as the minimum of the x and y spacings

of the mesh, but for unstructured meshes there are a number of possible definitions. For this work, the characteristic length was chosen as the minimum distance between face Gauss points for all faces on the cell. This approach is stable, can be precomputed and applies to flows in all orientations.

The maximum timestep used to advance the solution is then the minimum of all the maximum stable cell timesteps in the mesh multiplied by the *CFL* number [4] in the range [0,1].

$$\Delta t = CFL \cdot min((\Delta t_{max})_0, (\Delta t_{max})_1, ..., (\Delta t_{max})_n) \qquad (2.48)$$

The *CFL* number is included as an additional safety factor for stability during time-advance, and its value can be adjusted depending on the flow regime anticipated. In this work, a *CFL* number of 0.5 was used.

Like results from first-order spatial reconstruction, results from first-order timestepping procedures such as those in Equations (2.45) and (2.46) tend to be overly diffusive for practical use. For this work, instead of using the first order method from Equation (2.45), a second order Runge-Kutta method is used.

$$U^{t+\Delta t/2} \;\; = \;\; U^t + \frac{\Delta t}{2}\left(\frac{\partial U}{\partial t}\right)^t \qquad (2.49)$$

$$U^{t+\Delta t} \;\; = \;\; U^t + \Delta t \left(\frac{\partial U}{\partial t}\right)^{t+\Delta t/2} \qquad (2.50)$$

This method is still explicit, since the solution at $t + \Delta t/2$ and $t + \Delta t$ depend only on the (known) solutions at time t and $t+\Delta t/2$ respectively, but is second-order accurate, and consequently less diffusive than the forward Euler method. The timestep limiting procedure is unchanged by switching to this method of time integration, although the solver is somewhat more stable due to the larger stability region for the Runge-Kutta method compared to the forward Euler method.

2.7 Fluid-Structure Interaction

As described in Section 1.1, the magnetized target fusion reactor being designed by General Fusion generates an acoustic wave in a liquid lead liner by impacting it with an array of mechanical pistons. Physically this involves both solid and fluid dynamics, as the pistons must interact with the lead to transfer their energy. Coupled domain processes such as this fall into the category of Fluid-Structure Interaction (FSI).

Interactions between fluid and structural domains occur on the wetted surface, which is the surface that is common to both domains. Boundary conditions on the wetted surface can be simplified to continuity of pressure and velocity (if heat and mass transfer are ignored). Continuity of velocity requires that no mass may cross the wetted surface and thus be transferred to the opposing domain, while continuity of pressure is required since the wetted surface (as a surface) has no mass and so any pressure gradient across it implies infinite acceleration. Continuity of velocity and pressure can also be seen as necessary for the limit case of a structure with vanishing mass, thickness and strength recovering the boundary conditions for a material interface.

2.7.1 One-Way and Two-Way Coupling

In FSI scenarios it is possible to differentiate between one-way and two-way coupling. Problems with one-way coupling have the motion and/or pressure in either the fluid or structural domains prescribed. Examples of this would be prescribing the motion of a piston within an engine while carrying out CFD simulation of an engine cylinder as in Figure 2.7, or using a precomputed pressure-time history from a blast-wave and applying it to simulations of different structural configurations as in Figure 2.6.

These both involve FSI, but boundary conditions are only enforced within one domain using boundary data known in the opposing domain. This type of coupling occurs when the conditions in one domain are known or assumed, and the response of the opposing domain is desired. These problems can be decoupled; the solution for the dominant domain can be computed ahead of
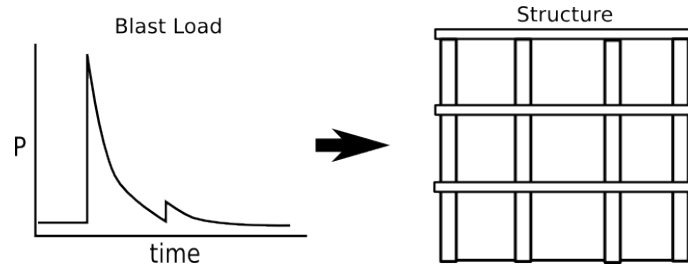
Figure 2.6: One-way coupling of a blast loading to a structure

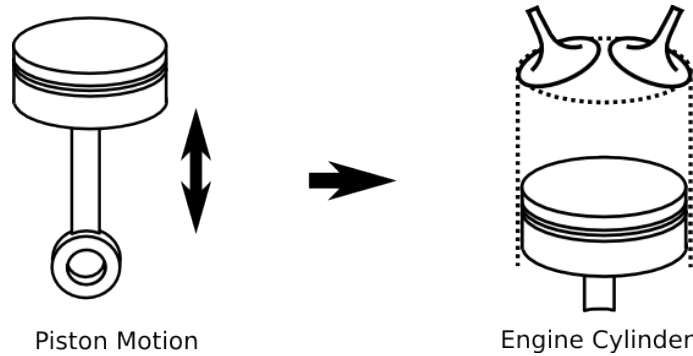time and applied to any number of opposing domains.

Figure 2.7: One-way coupling of piston motion to an engine cylinder

In contrast, two-way or full coupling involves boundary condition information passing back and forth between the domains, and is applied when either the response of the system as a whole is desired, or when the interactions are complex enough that the conditions in one domain cannot be assumed with any reasonable degree of accuracy. Since each domain requires boundary condition information from the other at all times, the simulations must be run in parallel and cannot be decoupled. This introduces a significant amount of complexity to FSI simulations, since the two domains must be able to resolve and swap boundary condition data and negotiate a timestep. An example of a situation that would require this type of coupling is a maneuvering aircraft, where motion of the control surfaces causes

changes in the fluid flow, which feed back to the structure through changes in pressure loads as shown in Figure 2.8.
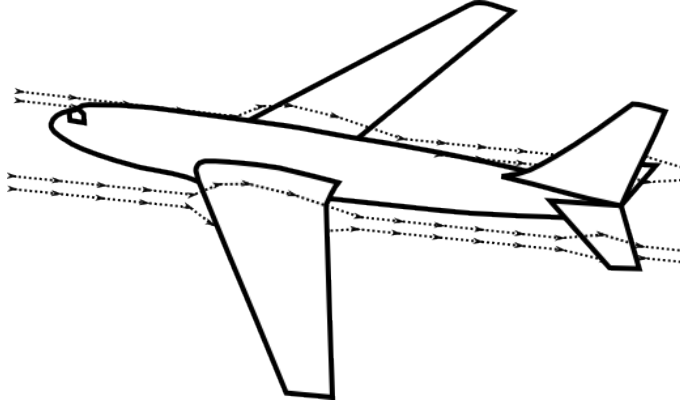

Figure 2.8: A maneuvering aircraft is an example of two-way coupling between structure and fluid

Solving two-way FSI simulations can be approached in several ways. One is to use a solver that applies to both domains and provide the constraints to implement the needed boundary conditions [3]. Although conceptually simple it is often impractical to write this type of solver, particularly when there are existing validated and standardized packages for each domain separately. A monolithic solver simplifies user interaction with the solver, and some commercial solvers implement this approach [12].

Another approach is to couple two single domain solvers through a multiphysics interface. For the solvers to communicate, both solvers must have a multiphysics interface, however this is becoming more and more common [12],[18]. Performing FSI simulations then only requires algorithms to access the data from either domain via its interface and apply it to the opposing domain as a boundary condition. Coupling two simpler, domain specific solvers is becoming a popular approach since it does not require a single monolithic solver that handles everything; instead small, focused solvers can be written which are coupled together dynamically as needed. This a schematic of this approach is shown in Figure 2.9.

Both one-way and two-way coupling mechanisms were implemented in the compressible flow solver, although only the one-way coupling was ultimately used. In the two-way coupling, communication with the structural solver LS/DYNA was to be performed via sockets using the *User Defined Loading* subroutine provided by LS/DYNA. The author had used this approach previously with good success [8],[7]; however time-constraints and shifting priorities during this project prevented the LS/DYNA portion of the two-way coupling from being completed.
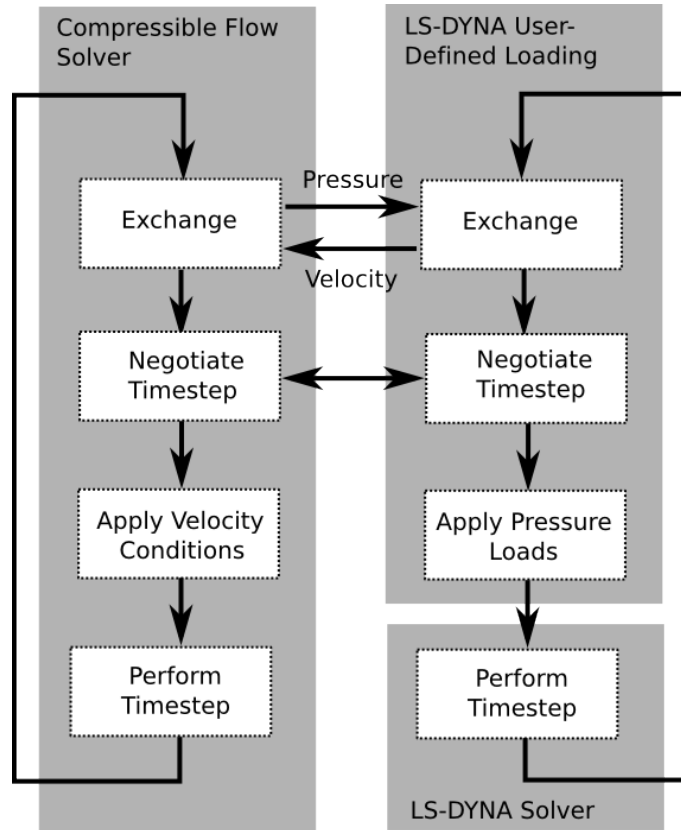

Figure 2.9: Schematic of coupling scheme with custom compressible flow solver and user-defined loading extension linked to LS-DYNA structural solver

2.7.2 Boundary Conditions

Across the wetted surface there is continuity of pressure and velocity but to implement fluid-structure interaction it is necessary to maintain boundary conditions that keep the two domains *connected*. For the two domains to actually be coupled, changes in boundary conditions from one domain must influence the solution in the opposing domain. If this is not the case, then the problem is not truly fully coupled, since the domain which is unaffected could just as well be used as a precomputed boundary condition for the other.

The issue arises of how to couple the domains such that each affects the other. One possibility is to use a penalty method so that each domain attempts to apply both pressure and velocity conditions to the other, and discrepancies between the two serve as negative feedback. In each timestep the constraints are then iterated over until the errors in satisfying them are are negligibly small. This type of approach is best suited towards monolithic approaches, where one solver can coordinate evaluating the constraints and iterating over them until convergence.

Another, more straightforward method is for each of the domains to apply one of the boundary conditions. If the problem is truly coupled, this will cause each domain to adjust over time to correct the boundary condition errors. For the fluid-structure interaction in this work, the structure was to be handled by a solver using a Lagrangian reference frame (LS/DYNA), while the fluid is solved in an Eulerian frame (solver described in the previous sections). Since the underlying definition of the structural problem is already in a Lagrangian frame, it is logical to use the structure to determine velocity constraints on the fluid. This requires that the pressure boundary condition come from the fluid. So the feedback mechanism by which each domain influences the other is as follows: pressure loads from the fluid are applied to the structure and induce structural motion, which in turn causes volumetric strains in the fluid, which is coupled back to the pressure through the fluid equation of state, which changes the loading on the structure.

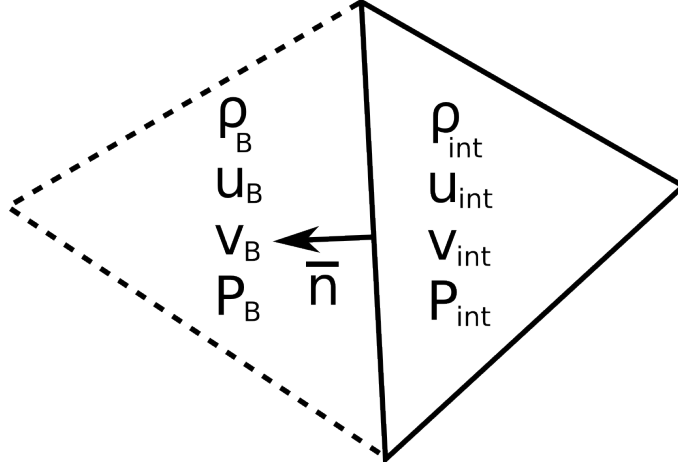Applying velocity boundary conditions for the fluid is straightforward.

Figure 2.10: Depiction of boundary cell and primitive variables required for FSI boundary condition. Since boundary conditions are enforced pointwise at Gauss points, the geometry of the boundary cell is actually not required.

Each face on the wetted surface is assigned a normal velocity, corresponding to the projection of the structural velocity at the face Gauss point location in the direction of the face normal. The state in the adjoining boundary cell is then set as follows

$$\rho_B = \rho_{int} \tag{2.51}$$

$$e_{iB} = e_{iint} \tag{2.52}$$

$$u_B = 2q_s n_x - u_{int} \tag{2.53}$$

$$v_B = 2q_s n_y - v_{int} \tag{2.54}$$

where subscript B indicates the boundary cell value, subscript int indicates the interior cell and q_s indicates the normal component of the structural velocity for that face. The boundary cell density ρ_B and internal energy e_{iB} are set to be the same as the interior values ρ_{int} and e_{iint}. This corresponds to a slip-wall condition in a moving reference frame, and allows energy and momentum to be transferred into the domain. Since density and internal

energy are duplicated in the boundary cell, it also maintains pressure equilibrium (for the fluid) across the wetted surface. Once the boundary cell values are set the solution variables can be reconstructed as usual.

Applying the pressure boundary condition on the structure is accomplished by integrating the fluid pressure over the surface of the structure. If the structure is composed of line-segment, triangular or quadrilateral elements, the fluid pressure can be integrated to second order accuracy by using Gauss points at the structural element centroids. The fluid pressure at the Gauss point location can be obtained by evaluating the reconstruction of the pressure field.

2.7.3   Small vs. Large Deformation

The discussion of FSI has so far avoided the issue of where to apply boundary conditions. For the structure this is not much of a problem since the boundary conditions should be applied on the wetted surface, which deforms with the structure. However the Eulerian fluid is more involved, since as the structure deforms, regions that were once fluid become structure and regions that were once structure become fluid. There are two approaches for dealing with the fluid, small-deformation methods and large-deformation methods.

Small-deformation methods assume that the effect on the fluid of structural deformation is small compared with the effect caused by the structural velocity. An example of this is a vibrating plate in a fluid where the plate vibrations cause pressure waves to be radiated into the fluid. In this case, the response of the fluid can be modeled well using a small-deformation method, since the effect on the fluid of the boundary geometry change is negligible. Small-deformation methods are convenient since the Eulerian mesh can remain fixed and unchanged throughout the duration of the calculation, and FSI can be implemented as a simple boundary conditions on fluid faces that are identified during the simulation initialization. Generally these methods are most applicable when the deformation of the boundary is small relative to the length scales in the domain (hence the term small-deformation).

In contrast, Large-deformation methods attempt to capture the change in the geometry of the boundary explicitly, under the assumption that those changes have a significant impact on the resulting fluid flow. Arbitrary Lagrange-Euler (ALE) methods [3], Chimera meshes and level-set methods [5] are all examples of large-deformation approaches. An example of a simulation requiring large-deformation methods would be flow through a valve, where the geometry of the fluid boundary has a large impact on the resulting flow patterns. Large-deformations methods are generally more complex than small-deformation methods, and there is no tried and true approach that works well for all situations. ALE approaches that deform the fluid mesh with the structure have difficulties with shearing motion and are prone to mesh tangling, Chimera methods only support rigid-body motion and must maintain boundary conditions between meshes while level-set methods add additional PDEs to the Euler equations and are prone to large conservation errors.

In the case of the pistons interacting with liquid lead, it was estimated by General Fusion that the pistons would move approximately 5 mm into the lead, from an initial radius of 1.5 m (or 0.33% of the reactor radius). Based on this estimate it was decided that a small-deformation method would be suitable for an initial investigation, although the possibility of a large-deformation ALE approach was not ruled out. Subsequent tests performed using the large-deformation code LS-DYNA showed the small deformation approach to be more than adequate, which is unsurprising given the results of the FSI benchmark presented in Section 3.1.4.

Chapter 3

Results and Discussion

This Chapter presents a selection of results obtained using the code described in Chapter 2. The results presented in Section 3.1 were performed to validate the numerical methods used in the solver, while those in Sections 3.2 and 3.3 examined the behavior of shock-propagation and cavity collapse in the General Fusion reactor concept. All results were obtained using second-order space and time methods, with the exception of those explicitly mentioned.

3.1 Test Cases and Validation

Before using the solver to examine shock propagation in the General Fusion reactor design, a number of test cases were performed. The test cases were structured to test the basic numerical scheme, and once those produced acceptable results, advanced to more specific cases such as cavitation and fluid-structure interaction.

3.1.1 Shock Tube

Shock tubes are one of the most common experimental apparati for compressible flow, and due to their close relationship with the Riemann problem for compressible flow, also serve as one of the fundamental tests for transient compressible flow solvers.

A shock tube problem is (generally) 1D, consisting of two quiescent states separated by a diaphragm which bursts at $t = 0$, resulting in the one shock, one contact and one rarefaction wave pattern from Figure 1.3 in Section 2.3. Shock tube problems test the flux-solver, time-integration and spatial

reconstruction schemes of a solver and have the added advantage of having an exact, analytical solution to compare with.

Here a slight variation of the Sod shock tube problem [19] is simulated, using ideal gas having $\gamma = 1.4$ a left state $\rho_L = 1.0$, $u_L = 0.0$ and $P_L = 1.0$ and a right state $\rho = 0.125$, $u = 0.0$ and $P = 0.125$. The results for density, pressure and velocity plotted as points with the exact solution shown as a solid line at $t = 0.2s$. The results compare very well with the exact solution and are shown in Figure 3.1.

For comparison purposes, the same problem was also simulated using first order time integration and spatial reconstruction. No other parameters were changed. A comparison of the second order density results from Figure 3.1 and the first order results are shown in Figure 3.2. This highlights why second-order (or higher) time-integration and spatial reconstruction is necessary; higher order methods produce substantially better solutions on a given mesh.
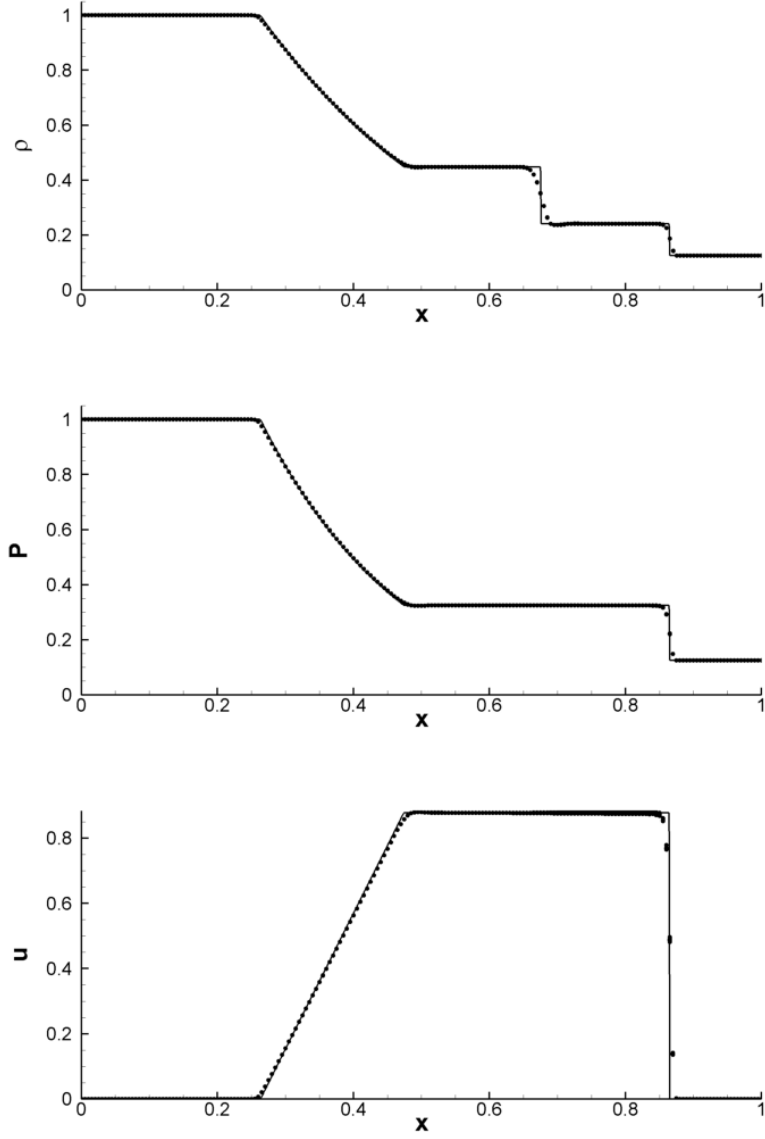
Figure 3.1: Comparison of numerical solution (scattered points) with exact solution (solid line) of the shock tube problem for density, pressure and velocity
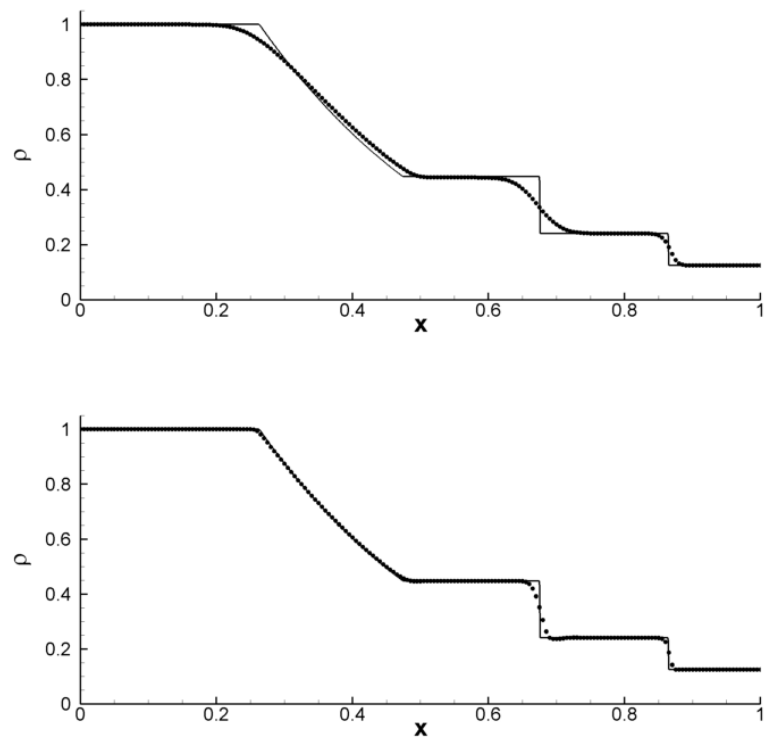
Figure 3.2: Comparison of density for shock tube problem using first-order time and space numerical methods (top) vs. second-order (bottom)

3.1.2 Geometric Source Terms

Although straightforward to implement, the geometric source terms play an important role in the solver since they can allow certain problems to be reduced in dimension. For example, when simulating cavity collapse, the 3D problem can be reduced (with some approximations) to a 2D axisymmetric problem which results in substantial savings in memory and computation time at a fixed mesh resolution, or conversely, significantly higher resolution at a fixed cell count.

To validate the geometric source terms, several tests were performed using the initial conditions of the shock tube test from Section 3.1.1. The first case tested the axisymmetric source terms by performing a simulation in 1D using the axisymmetric source terms and comparing the results to those obtained from a 2D simulation of the same axisymmetric problem (problem domain shown in Figure 3.3) but with the source terms turned off. Both the 1D and 2D meshes were generated using the same mesh-spacing, although the 2D mesh was generated using unstructured quadrilaterals.
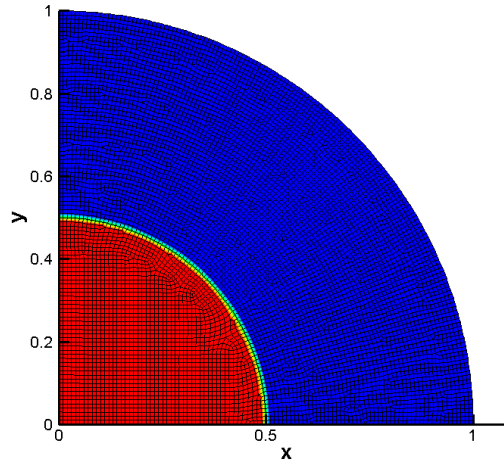

Figure 3.3: 2D Domain with contours of density used to test geometric source terms
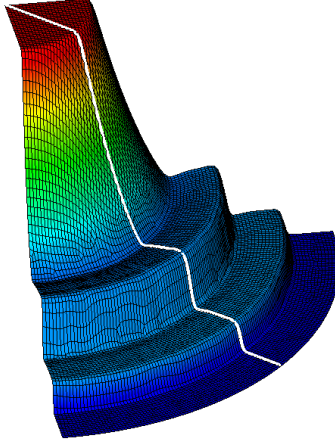
46

Figure 3.4: Comparison of 2D planar solution (contoured mesh) with 1D solution using axisymmetric source terms (superimposed line).

The results of this case are shown as an elevation plot in Figure 3.4, where the thick line shows the 1D results obtained using the axisymmetric source terms and the contoured mesh shows the results from the 2D simulation with no source terms. The results compare very well; Figure 3.5 shows a line plot of the same data in which the 1D results are shown as scattered points and all of the points from the 2D domain have been rotated to fall on the $R - \rho$ plane, where $R = (x^2 + y^2)^{1/2}$. Rotating the 2D points rather than simply taking a section provides an indication of the symmetry of the solution, thicker lines in the plot are caused by discrepancies in the solution at constant R. For this test these errors are minor, and it is easy to see that the 2D results agree very well with the results from the 1D simulation.

Once the axisymmetric source terms were validated, the spherical source terms were then checked using the same procedure, except that a 1D simulation with spherical source terms was checked against a 2D axisymmetric simulation using the same domain as in the previous case. Results are shown in Figure 3.6, with the thick superimposed line showing the results obtained
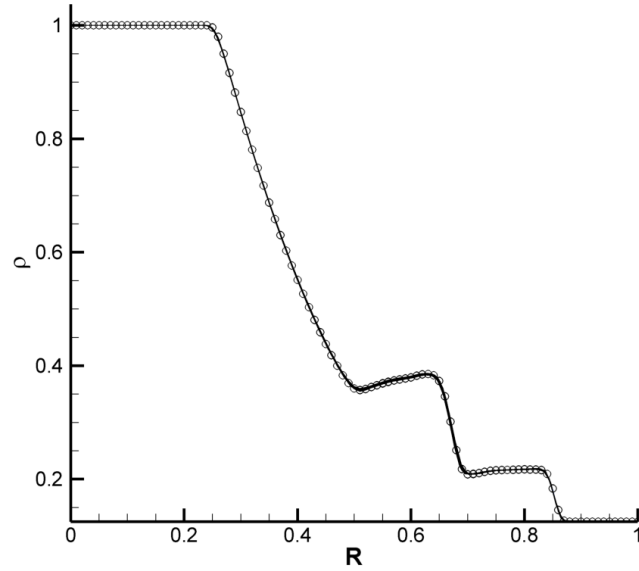
Figure 3.5: Comparison of 2D planar solution (solid line, $R = (x^2 + y^2)^{1/2}$) with 1D solution using axisymmetric source terms (scattered points).

from the 1D simulation using spherical source terms and the contoured mesh showing results from the 2D axisymmetric simulation. As before the results are nearly coincident (see Figure 3.7).
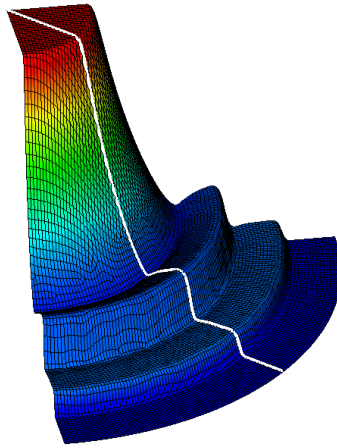
Figure 3.6: Comparison of 2D axisymmetric solution (contoured mesh) with 1D solution using spherical source terms (overset line).
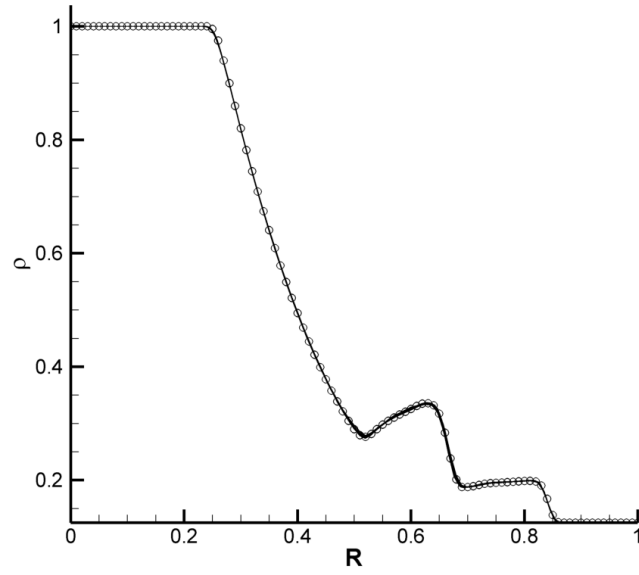
Figure 3.7: Comparison of 2D axisymmetric solution (solid line, $R = (x^2 + y^2)^{1/2}$) with 1D solution using spherical source terms (scattered points).

3.1.3 Cavitation

To test the hybrid cavitation model developed in Section 2.5.4, a test case from Saurel [16] was performed. In this case water (modeled here using the Tait equation of state) having left and right states of [ρ,u,P] of [1000 kg/m^3, -100 m/s, $10^5 Pa$] and [1000 kg/m^3, 100 m/s, $10^5 Pa$] initially is simulated for 0.2 ms in a 1m domain using 1000 cells. The results at 0.2 ms have a vapor bubble of low density at the location of the initial boundary between states, along with a transition from the left velocity to the right. Surrounding the vapor region is a low pressure cavitated region, which extends to the left and right.
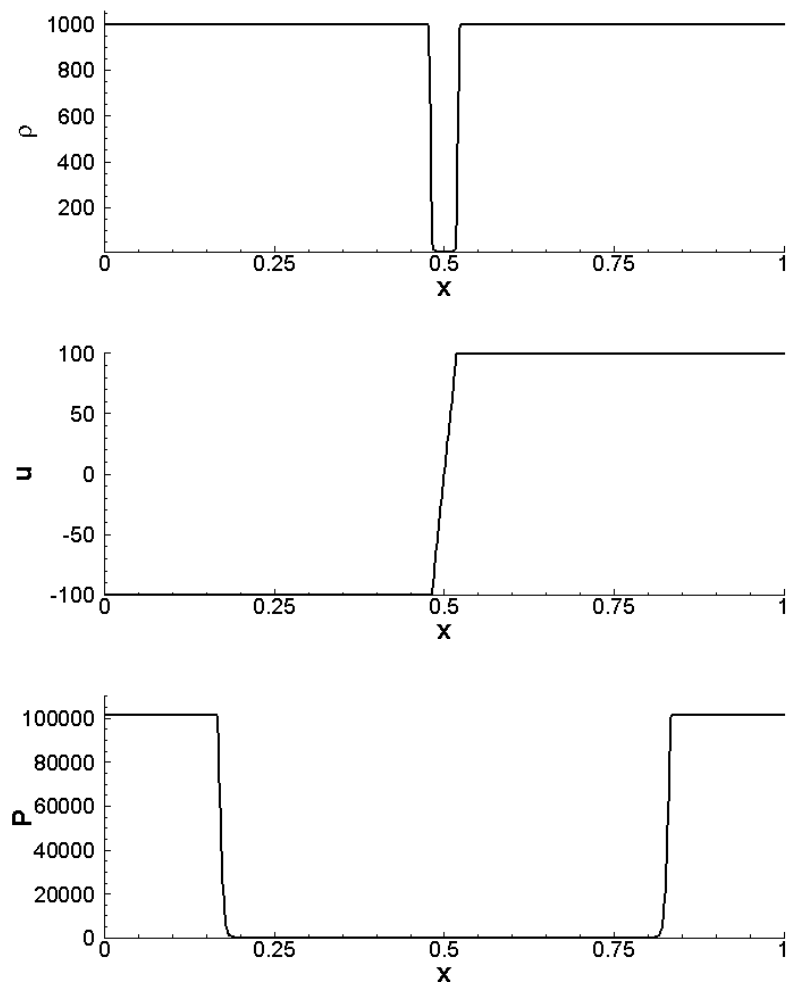
Figure 3.8: Cavitation benchmark results

These results compare reasonably well with Saurel particularly considering that different equations of state were used for modeling the water. One key difference is that our results are considerably sharper than theirs. We are not sure if this is due to using a second-order scheme while they (may have) used a first order method. Regardless, our method captures the bounds of the vapor region quite well along with the transition in velocity while slightly enlarging the low-pressure surrounding region compared to theirs.

3.1.4 Fluid-Structure Interaction

To test the one-way coupled fluid-structure interaction (FSI) boundary condition with prescribed structural motion, the similarity between the FSI boundary condition and a material interface can be exploited to change the existing tests into FSI benchmarks. As was described in Section 2.7, the structure at the limit of zero strength and stiffness should behave as a material interface. Consequently, existing tests such as the shock-tube can be adapted simply by replacing the material interface with a structure moving at the same velocity and omitting one half of the fluid simulation.

This was done with the cavitation benchmark and the shock tube. The results for the cavitation benchmark are shown in Figure 3.9, and show the results obtained by placing an FSI boundary (plotted as points) moving at 100 m/s to the right at the edge of a domain fill with water moving at 100 m/s to the left. The results of this should be similar to those obtained from the cavitation benchmark (plotted as a line), although not identical since in the benchmark the fluid velocity transitions between ± 100 m/s over a number of cells.
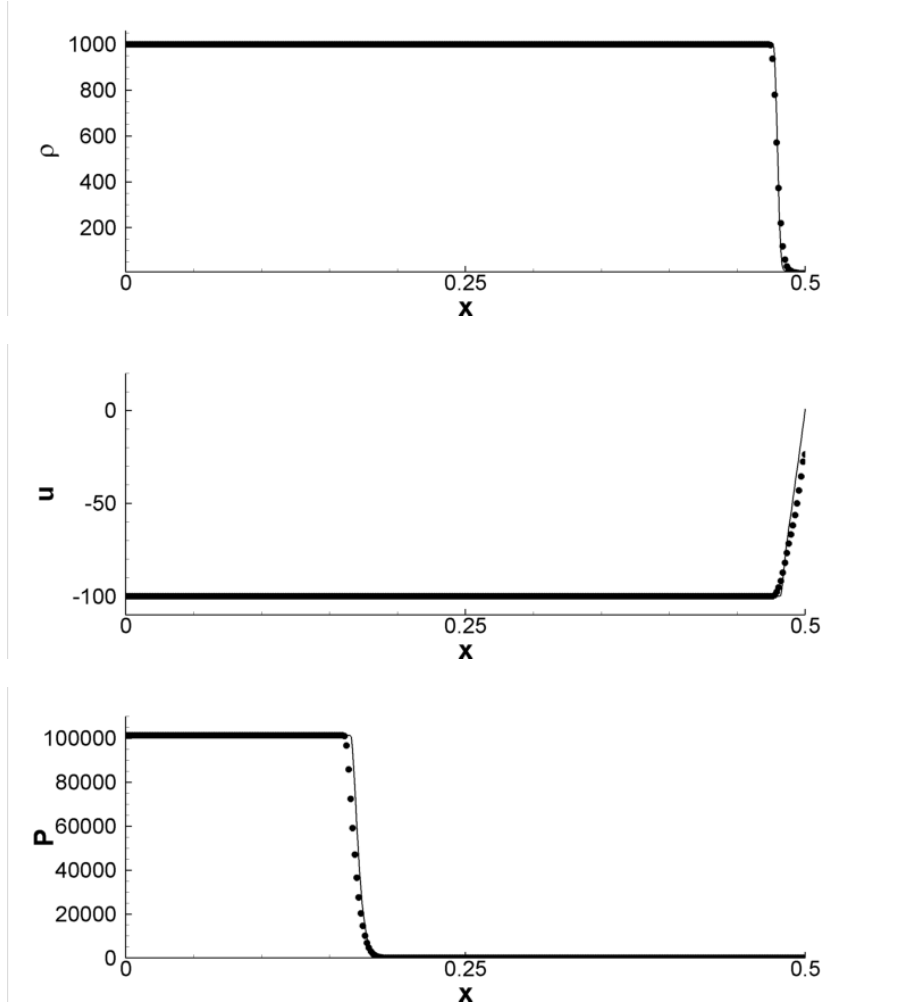
Figure 3.9: FSI benchmark results (dotted) compared with cavitation bench-mark results (solid)

Agreement between the two solutions is good; the low-density cavitated region is reproduced quite well, as is the pressure distribution. Ambiguity in the velocity field occurs, although the cause is not clear. Two possibilities are the transition for the velocity field for the cavitation benchmark and effects introduced by the non-conservative nature of the FSI boundary condition.

The shock tube test was also modified to serve as an FSI test. The modified version consisted of quiescent air with $rho = 0.125 kg/m^3$, $u = 0.0$ and $P = 0.125$, with an FSI boundary condition at the left edge of the domain. At $t = 0$ the velocity condition begins to move at a constant $0.87789 m/s$ to the right. This velocity was chosen since it is the S_M wave speed. The results to the right of the velocity condition should duplicate those from the shock tube test, except that the FSI test case does not have a moving contact surface, so the state to the left of the contact surface should be identical to the shocked right state. The results are shown below in Figure 3.10.
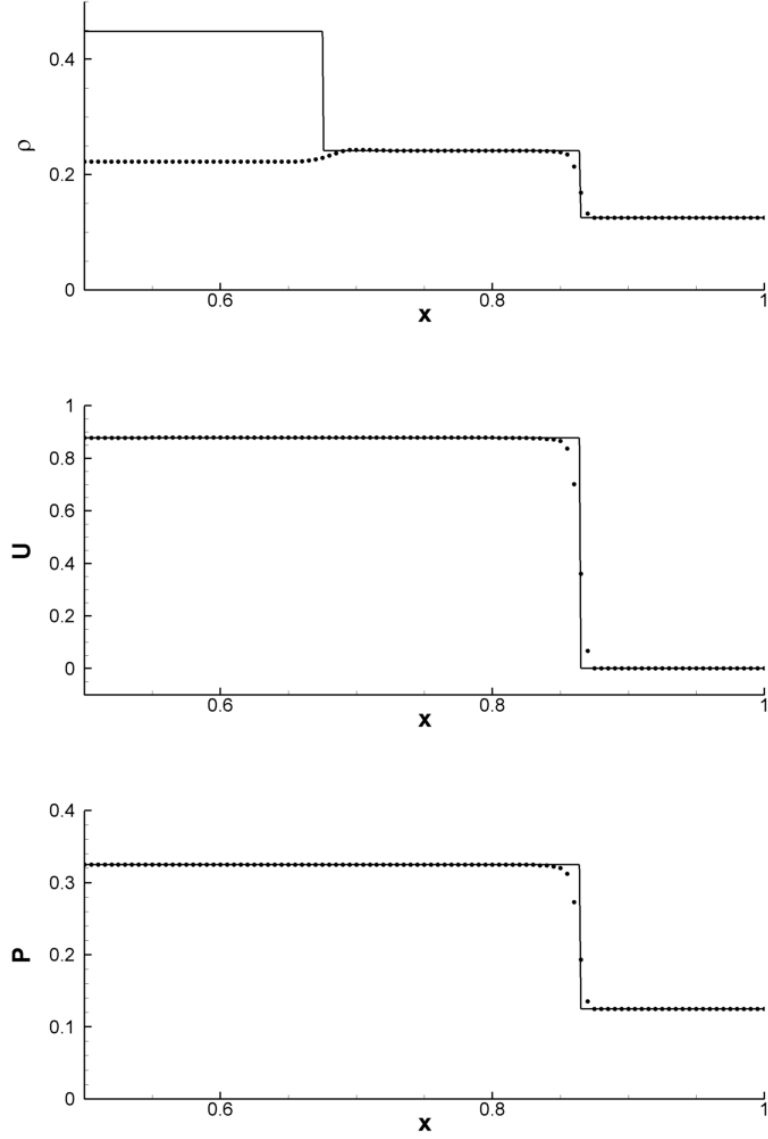
Figure 3.10: FSI benchmark results (dotted) compared with shock tube results (solid).

The results agree well with the exception of the drop in density for $x < 0.675$. The density here should be equal to the density in the region from $x = 0.7 - 0.85$, however it is underpredicted. The underprediction is due to the nature of the FSI boundary condition; fluid is set in motion by the boundary, but with no motion of the boundary or fluid inflow the density must drop. This is an artifact of the small deformation assumption and if a large-deformation method had been used no density error would occur, although the solution would not exist in the mispredicted region due to boundary motion. One thing to note is that although the boundary "moves" by approximately 1/3 of the domain for this benchmark, the resulting solution from the small deformation method does a good job of capturing the resulting shock. Errors are confined predominately to the region which should have been covered by boundary motion.

This suggests that the small deformation assumption is quite reasonable for the General Fusion reactor, where the 300 mm diameter pistons move only 5mm into the 1.5 m radius domain.

3.2 Shock Smoothing and Merging

In order to generate the acoustic pulse that crushes the plasma cavity, an array of pistons must impact the liquid lead liner. Each piston in the array contains two sub-pistons, an anvil piston and a hammer piston. These are depicted in Figure 3.11. The hammer piston is accelerated down an evacuated housing with high-pressure steam on the driving side. After traveling approximately one meter, it strikes the stationary anvil piston which is already in contact with the liquid lead liner. Energy from the hammer piston is transferred first to the anvil piston and then from the anvil piston to the liquid lead liner in a similar manner to billiards balls striking each other. As the anvil pistons interacts with the lead, a high pressure acoustic wave is formed on each piston face which propagates inwards; it is these waves which cause the collapse of the plasma cavity.

However, since the pistons create a number of distinct and separate waves, the shock which forms from the piston waves is rippled. These rip-
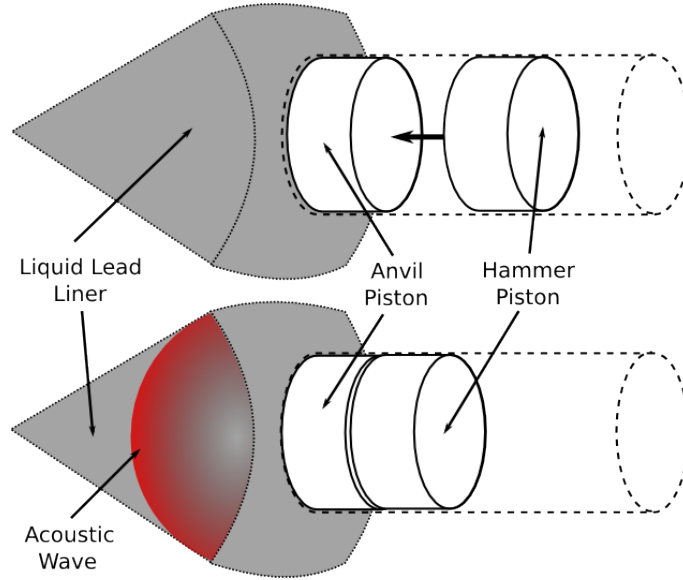
Figure 3.11: Depiction of hammer and anvil piston arrangement, reactor and piston housing omitted for clarity.

ples have the potential to act as perturbations from which hydrodynamic instabilities may grow; Richtmeyer-Meshkov instabilities as the shock reflects from the lead-plasma interface and Rayleigh-Taylor instabilities during the later stages of plasma compression. It was consequently desirable to gain a better understanding of how the distinct waves from each piston merge, smooth and interact with each other to form a coherent front, and how that front behaves as it propagates in order to be able to estimate the smoothness of the resulting front which could later be used to estimate instability growth rates.

This process of smoothing and merging occurs because as each piston generates an acoustic pulse, refraction occurs at the piston edges. These refracted waves have lower energy than the initially planar front generated at the piston face and propagate perpendicular to the direction of piston motion. Since the pistons are closely spaced, the refracted waves quickly begin to interact (see Figure 3.12), forming Mach stems and triple points.
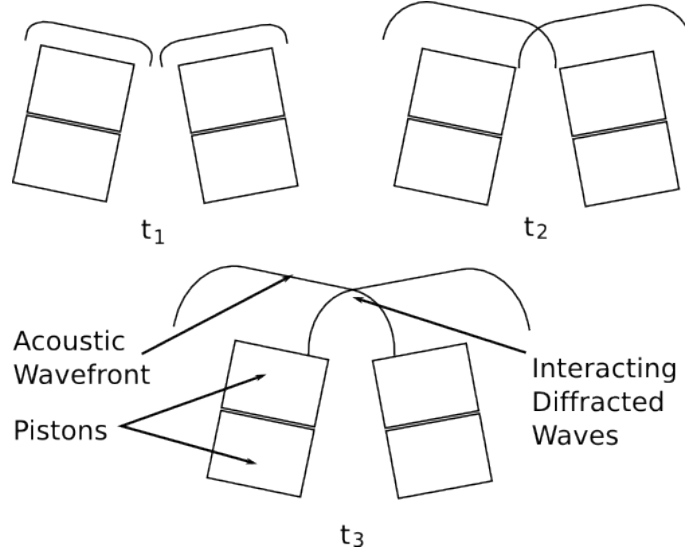
Figure 3.12: Depiction of diffracted wave interactions for adjacent pistons at three times, t_1, t_2 and t_3.

The Mach stems and triple points propagate faster than the main wavefront created by the piston face due to the increase in pressure that occurs during the wave interactions, and can catch up to and pass the planar front. As this process occurs, new Mach stems and triple points begin to form behind the main wavefront and the process repeats in a leapfrog fashion. Simulations results showing this process are shown in Figure 3.13.

To better understand this process, a series of simulations were carried out to assess how piston-generated shocks at varying Mach numbers smoothed over time, and which factors most influenced the degree of smoothing. The simulations consist of a 10×1 rectangular domain. On the far right edge of the domain a piston was simulated using a one-way fluid-structure interaction boundary condition with a step function in velocity at time $t = 0$. The piston was sized to be a fraction of the domain height; the remaining fraction denoted the gap-width Φ, and all boundaries excluding the piston were set to slip-walls. Using symmetry planes through the piston and gap allowed only half of a single piston and half of a single gap to be simulated.

(a) Wave interactions form new shock

(b) Triple point and Mach stem begin to form

(c) Triple point moves away from symmetry plane

(d) New shock begins to overtake initial shock

(e) New shock overtakes initial shock
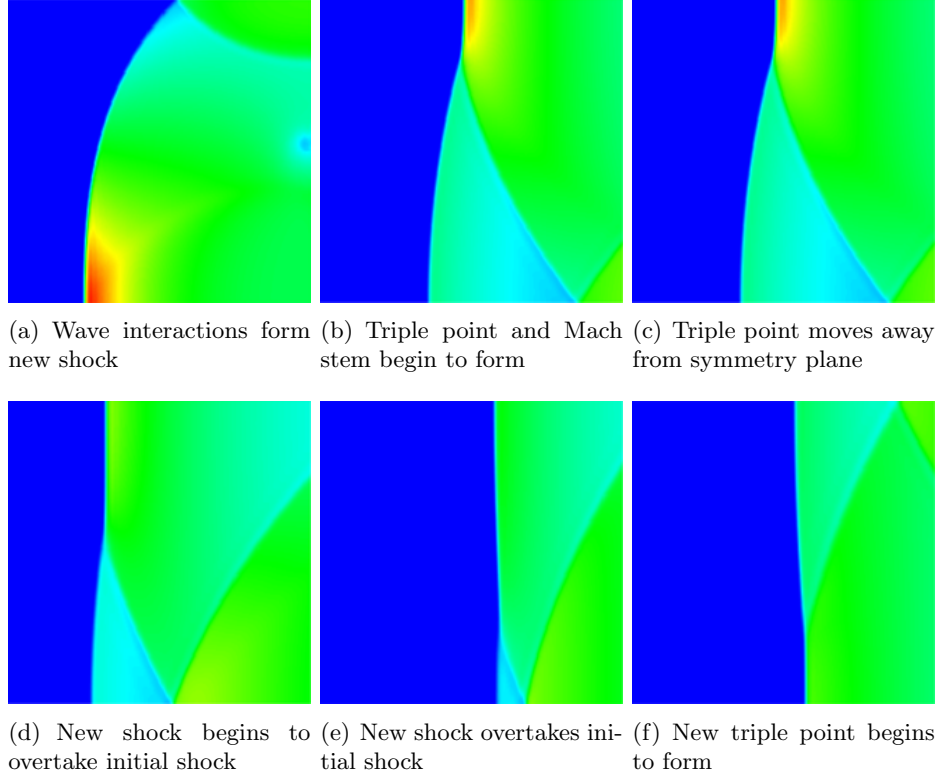
(f) New triple point begins to form

Figure 3.13: Sample results illustrating shock smoothing process

A sketch of the domain is shown in Figure 3.14.

To quantitatively assess the wave smoothness, a metric Δ was chosen in which the smoothness of the front was the maximum distance between the leading and trailing points of the shock front. Results for the smoothness were obtained automatically from the output of each simulation through the use of time-of-arrival gauge arrays along the top and bottom edges of the domain. The gauge positions and arrival times allowed piecewise-linear position-time curves for the shock at the domain top and bottom to be obtained. The difference between these curves when sampled at the same time gives Δ and the average value of the two curves gives the distance from the piston D. The gauge arrays had a gauge every $\frac{1}{4}$ units; this was found
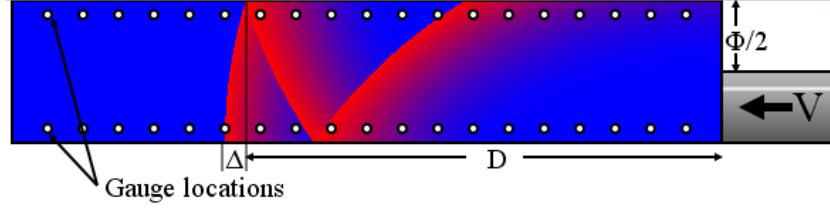
Figure 3.14: Illustration of the (shortened) simulation domain. Boundary condition on the piston is the velocity boundary condition described above, all other boundaries are slip-wall conditions. Gauge array locations are shown as white circles along top and bottom of domain. D, Φ and Δ are measured in units of piston spacing(twice the domain height)

to be sufficient to resolve the smoothing process. To determine the shock arrival time, gauges used a threshold heuristic in which the shock arrival time was the first time that the measured pressure (less ambient) exceeded 10% of the expected pressure at the piston face (determined by solving the Rankine-Hugoniot Equations). Although more sophisticated methods could be used this was found to be reliable for all shock Mach numbers at all values of D and it produced results that agreed well with manual sampling while still being consistent.

Two series of simulations were performed. The first compared the effect of gap-width Φ at constant Mach number, while the second compared varying Mach number at constant gap-width. Results from simulations performed at Mach 1.5, 1.25 and 1.125 are shown in Figures 3.15-3.17.
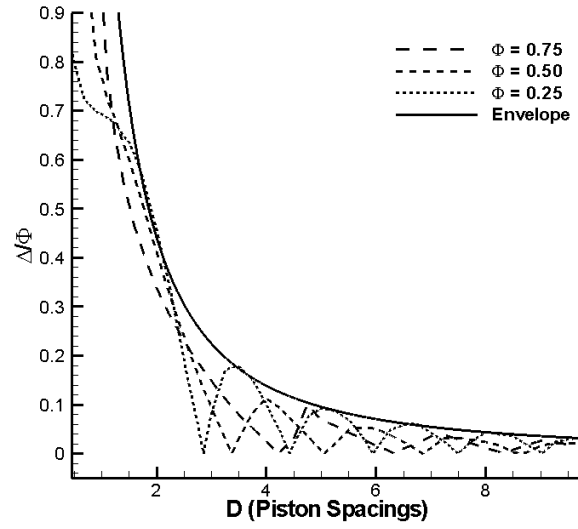
Figure 3.15: Variation of Δ/Φ for a Mach 1.5 shock as a function of distance from pistons (D) and gap width (Φ), both measured in units of piston spacing. The equation for the envelope is $\Delta/\Phi = 1.4D^{-5/3}$

Figure 3.16: Variation of Δ/Φ for a Mach 1.25 shock as a function of distance from pistons (D) and gap width (Φ), both measured in units of piston spacing. The equation for the envelope is $\Delta/\Phi = 1.6D^{-5/3}$

Figure 3.17: Variation of Δ/Φ for a Mach 1.125 shock as a function of distance from pistons (D) and gap width (Φ), both measured in units of piston spacing. The equation for the envelope is $\Delta/\Phi = 1.8D^{-5/3}$
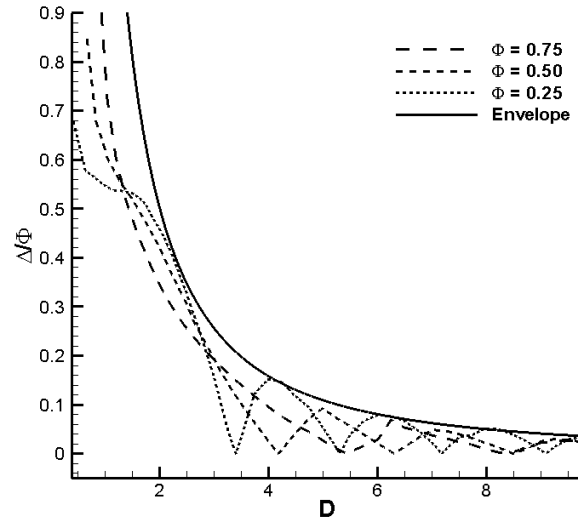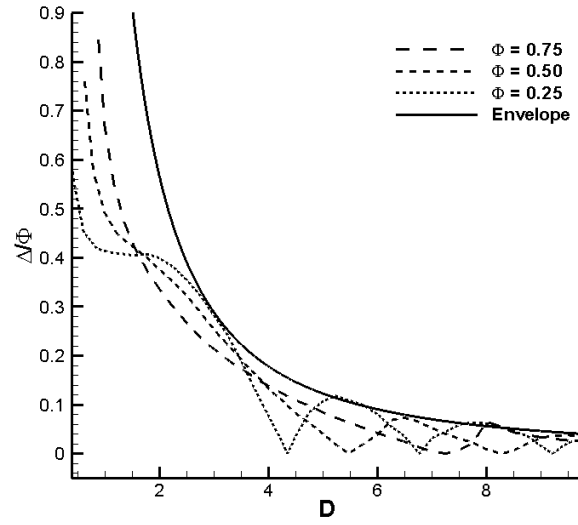
Figures 3.15-3.17 show that the peak values of Δ when normalized by the gap-width Φ decay with largely similar envelopes for the range of gap-widths studied. There is a trend towards smaller-gap widths having higher Δ/Φ (particularly for higher Mach numbers), although this is relatively minor when compared with the global trend. One possible explanation is that cases with larger gaps have correspondingly smaller pistons; consequently the initial front has less energy and also has to propagate further (decaying more) before interactions with neighboring waves may occur. The subsequent interaction between less energetic waves intuitively seems likely to produce more sluggish reactions, an observation that seems to be borne out by simulations performed at constant Φ but varying Mach number presented below.

Another trend in Figures 3.15-3.17 is increasing sharpness of waves with increasing gap-width. A gap-width of 0.25 results in quite round peaks, compared to sharp peaks with a gap-width of 0.75. The reason for this is unclear.

Envelopes were fit to the peaks of the data in Figures 3.15-3.17 in order to obtain approximate maximum values for Δ as a function of Φ and D. A form of $\Delta/\Phi = AD^{-5/3}$ was found to work well, with A being determined by the Mach number of the shock. Mach numbers of 1.5, 1.25 and 1.125 have A values of 1.4, 1.6 and 1.8 respectively. It should be noted that these envelopes may significantly overpredict Δ for $D < 2$. This quantifies the observed trend towards increased smoothing with increasing Mach number although how long this increase continues is unknown. However, practically generating shocks mechanically at Mach numbers greater than 1.5 without destroying the machine that creates them (as is the focus of this work) seems challenging. For example, a steel piston impacting liquid lead to produce an approximately Mach 1.1 shock would see a compressive stress of approximately 2 GPa, well above the yield stress of most steels.

Convergence of the results was checked at $\Phi = 0.5$ for a Mach 1.5 shock and is shown in Figure 3.18. The strongest shock was chosen since previous work in underwater explosions had indicated that stronger shocks tended to require higher resolutions meshes. This shows that the results are reasonably
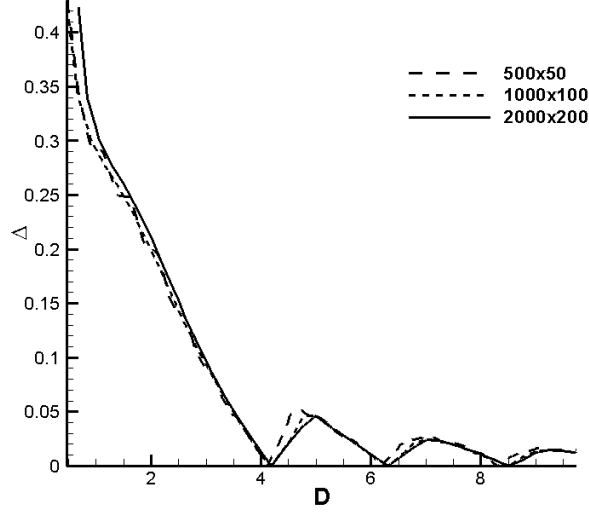
Figure 3.18: Convergence of results for three different mesh resolutions.

well converged for mesh resolution of 2000x200 particularly for $D > 3$, which is the region that has more practical importance for this work.

Figures 3.19-3.21 show the effect of varying the Mach number (M=1.125, 1.25, 1.5) at constant gap widths ($\Phi = 0.25, 0.5, 0.75$). The envelopes for these plots are the same ($\Delta/\Phi = 1.8D^{-5/3}$) as those for the Mach 1.125 shocks presented in Figure 3.19.
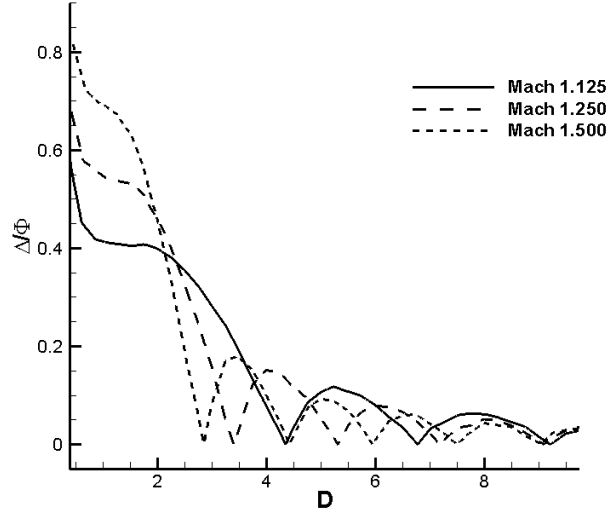
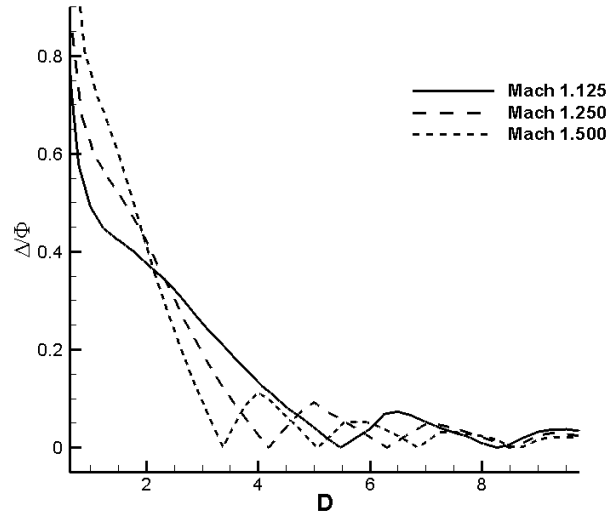Figure 3.19: Δ/Φ at $\Phi = 0.25$ for M=1.125, 1.250, 1.500

Figure 3.20: Δ/Φ at $\Phi = 0.5$ for M=1.125, 1.250, 1.500
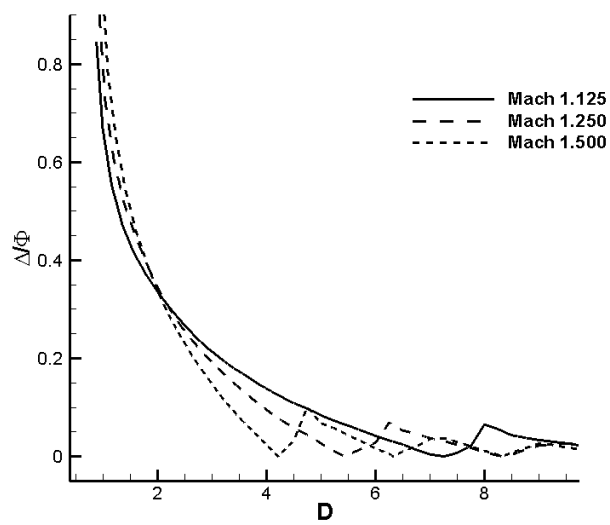
Figure 3.21: Δ/Φ at $\Phi = 0.75$ for M=1.125, 1.250, 1.500

For Φ=0.25 and 0.5, increasing Mach number increases the smoothing of the wave, as would be expected from the A coefficients of the smoothing envelopes. For Φ=0.75, the effect is less clear due to fewer peaks in the results, although it does appear to be the trend as well. As mentioned in the results for varied gap-widths, results with large gap widths tend to have sharper peaks than those with small gaps across all Mach numbers studied.

Overall the factor that was found to most affect front smoothness was the gap-width between pistons as a fraction of piston spacing. One interesting effect was that domains with smaller gap-widths values resulted in rounder Δ/Φ oscillations than domains with larger Φ values, although the overall envelopes were similar.

To assess qualitatively how smoothing depends on material properties, a series of simulations were performed using water and air as the working fluids for $\Phi = 0.5$. The results from these are shown in Figures 3.22 and 3.23. Qualitatively the results for water are quite similar to those from lead. This was expected since liquid lead can be modeled reasonably well using the Tait equation of state that was used for the water. The envelope for both is also similar with only the leading factor changing from 1.6 to 1.75, although it should be noted that there was considerable leeway in obtaining this fit and it should not be extrapolated beyond the data obtained.

The same series performed for air shows that there is little to no additional smoothing as Mach number increases. One possible explanation for this is that temperature increases during wave interactions have a significant impact on the resulting pressure for air, while for liquid lead and water the equations of state are barotropic so kinetic energy of the flow that is converted into internal energy of the fluid is essentially lost to the system. The smoothing envelope was found to be $\Delta/\Phi = 1.3D^{-1.4}$.

Using the envelopes for lead, water and air at $D = 10$ and $\Phi = 0.5$, the Δ values are 3.44%, 3.77% and 5.18% of the gap-width. Considering the differences in the materials these changes are quite small, particularly between the liquids. Consequently it can be concluded that the dependence of smoothing on material properties is relatively weak. Looking at just lead, the changes in the envelopes with Mach number are on the order of 0.8%
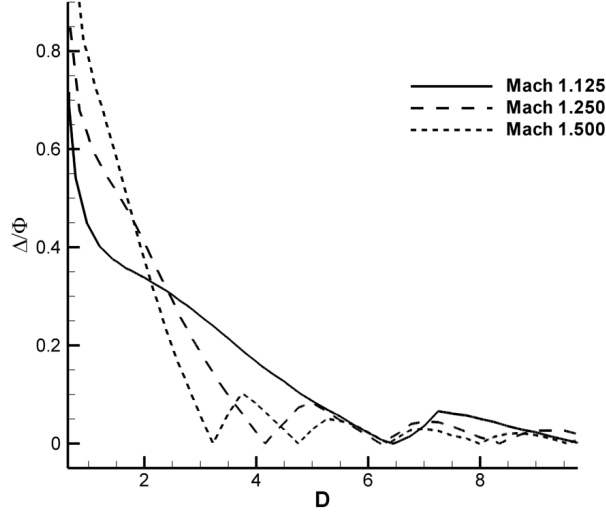
Figure 3.22: Δ/Φ at $\Phi = 0.5$ for water at M=1.125, 1.250, 1.500. The envelope is $\Delta/\Phi = 1.75D^{-5/3}$.

of the gap width between Mach 1.125 and Mach 1.5. The dependence on Mach number is therefore also reasonably weak, and the primary determining factor in wave smoothness is simply the gap-width.

A number of envelopes bounding the smoothness of the shocks in lead under various conditions were established from the simulation results. Of these, a conservative estimate of front smoothness for Mach numbers up to Mach 1.5 and gap-widths from 25-75% of the piston spacing is $\Delta = 1.8\Phi D^{-5/3}$, where Δ is the distance between leading and trailing points on the front, D is the piston spacing and Φ is the gap-width as a fraction of D.

It should be noted that significantly smoother fronts than the above estimates may also be obtained at a given D due to the oscillatory nature of the front by choosing the Mach number and gap-width such that the leading and trailing points of the shock are even. For instance, for $\Delta/\Phi = 0.75$ and $M = 1.5$ as shown in Figure 3.21 above, Δ is near zero when D is close to

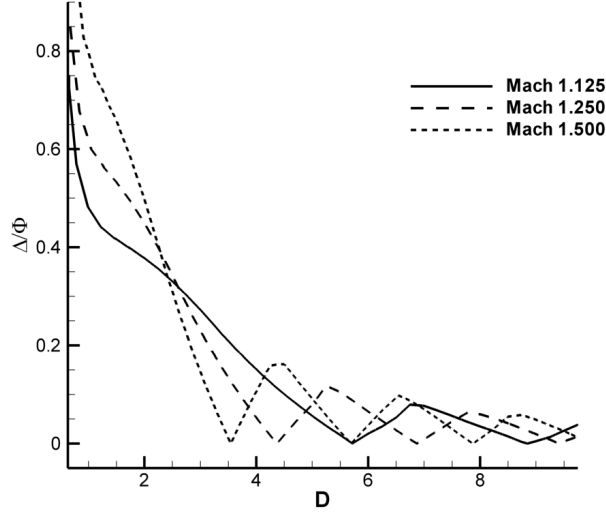Figure 3.23: Δ/Φ at $\Phi = 0.5$ for air at M=1.125, 1.250, 1.500. The envelope is $\Delta/\Phi = 1.3D^{-1.4}$.

4.2, 6.4 and 8.4. These points occur where a lagging front catches the leading front. This is dependent on material, geometry and shock Mach number, so must be determined through simulation for the desired configuration subject to any mechanical or hydrodynamic constraints that affect the workings of the desired system.

3.3 Cavity Collapse

One of the major issues with the design of the General Fusion MTF reactor is obtaining a good cavity collapse, where good implies optimizing fusion yield as a function of capital and operating costs of the reactor. One of the key components of the fusion yield is the volumetric compression of the plasma; the more compressed the plasma is the better. Another issue is the shape of the cavity, where it is assumed that spherical or approximately spherical, cavities are preferable to oddly shaped cavities. This is based on a

number of observations, strangely shaped cavities provide more opportunity for hydrodynamic instabilities to develop, have larger surface areas for heat and mass transfer as lead at the lead-plasma interface vaporizes and could disrupt the magnetic field that helps to confine the plasma.

One dimensional hydrodynamic calculations carried out prior to this work suggest that volumetric compressions of 1000X will result in sufficient fusion yield to power the reactor and produce power at a cost competitive to current energy production methods. Based on these results and other factors, the design of the reactor would be such that at 20 cm radius spherical plasma cavity would see 1000X volumetric compression, resulting in a 2 cm cavity at peak compression. However for there to be a path for the plasma to enter the reactor, the cavity must be initially open at either end. Immediately following injection of plasma, the collapse process must begin, since the plasma is expected to cool in times on the order of tenths of milliseconds. One of the key design issues with the reactor is how to close the ends of the initially cylindrical vortex to obtain an approximately spherical cavity which can then be compressed by a factor of ten in the radial direction.

The success of this process depends strongly on the early time hydrodynamics of how the cavity ends are closed, and how the resulting cavity is formed. For much of this process, the flow can be approximated as a straightforward compressible lead flow with highly rarified lead representing the plasma. This type of flow can be handled by the solver developed and tested in the preceding sections, however actually finding a solution, that is a good collapse, is daunting. There are a large number of design variables — initial cavity shape, piston positions, piston powers, reactor shape — and to add to the difficulty, the simulations take on the order of an hour to complete. Furthermore the objective function is difficult to define precisely, since ultimately an optimizer will need a single number that encodes cavity shape, collapse velocity and the effect of surface perturbations for each combination of design variables. In addition to that, early optimization of the collapse could be wasted time if the results of MHD simulations do not align with the assumptions regarding cavity shape described above.

This section presents what amounts to dipping a toe in the water of this optimization problem. Significant effort will be required to obtain an optimal collapse, with no small part of this effort spent determining valid ranges for the design variables. These ranges depend on many factors, some physical constraints such as the materials and construction of the reactor while others are economic constraints such as the cost of manufacturing the reactor. All of these factors must then be included in a model that trades off fusion gain for reactor capital and operating cost, which would ideally include the best known physics for the plasma behavior.

Simulations were performed with varied piston impact timings for fixed reactor geometry, cavity geometry and piston velocity time-history. The simulations were carried out in an axisymmetric coordinate system with quarter symmetry and six pistons equally spaced along the wall of the liner. It should be noted that using an axisymmetric coordinate frame implicitly defines the pistons as rings, rather than as individual pistons with cylindrical cross-sections. The effect of this assumption is unknown, but presumed small since the packing of the final pistons will actually be more uniform than that of the rings. The velocity time-history of the pistons was approximately a Gaussian function, with a peak value of velocity of 100 m/s and a width (corresponding to one standard deviation) of 80 microseconds.

Before carrying out the axisymmetric simulations, a one-dimensional resolution study was performed to determine the cell-size required to adequately resolve the collapse. The tests were run until the plasma pressure as calculated by assuming isentropic compression with the ideal gas equation of state became of comparable magnitude (30%) to that the liner, corresponding to a radius of approximately 4 cm. The pressure rise in the plasma happens very quickly past this point, and the approximation of the plasma as having negligible pressure and density begins to break down. The convergence results are shown in Figure 3.25.

The convergence test results show that the solution is grid-converged for meshes of greater than 400 cells, corresponding to edge lengths less than 3.75 mm. One thing to note is that although the peak pressure values for the 200 cell mesh are quite low compared with the higher resolutions, the
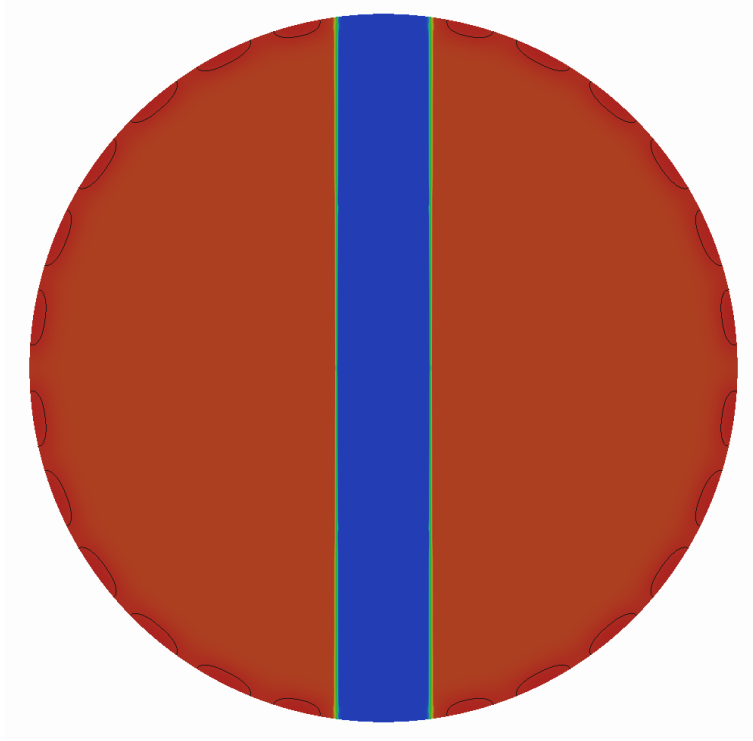
Figure 3.24: Domain used in collapse simulations mirrored about x and y axes. Line contours of pressure show approximate piston locations.

location and size of the cavitated region (region of zero pressure to the right of the shock) and the material interface (shock location) are reasonably well predicted (given the pressure disparity). The 200 and 400 cell meshes corresponds to edge lengths of 7.5 mm and 3.75 mm respectively.

The 2D axisymmetric collapse meshes were generated with an edge length of 5.5 mm in the region near the collapse, and approximately 15-20 mm elsewhere. This could be done due to the smooth function used for the piston velocity. Since the piston velocities are smooth, the resulting pressure wave is as well for quite some time, only sharpening into a shock well after it has passed a radius of 0.5 m. This allows much coarser resolutions while the wave is smooth, and yields significant savings in cell count
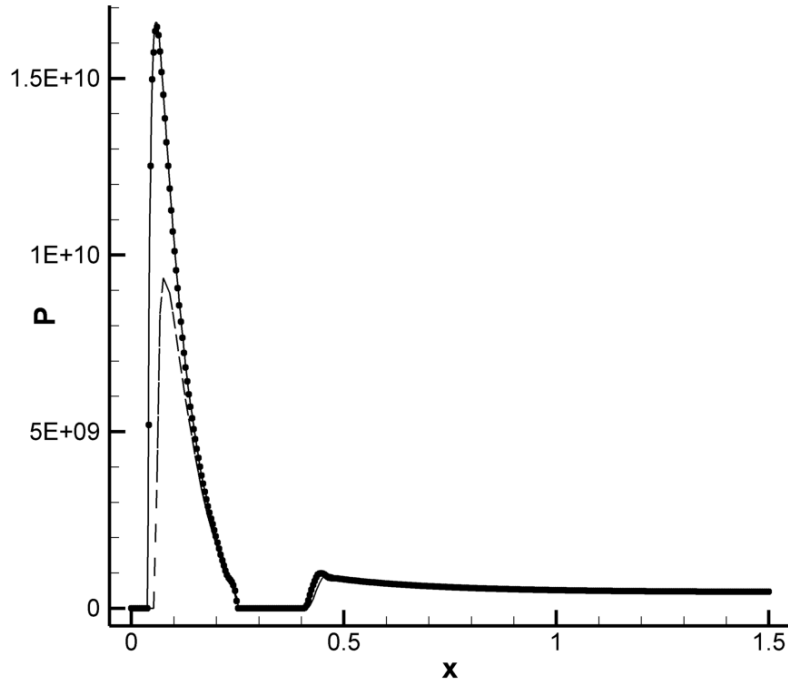
Figure 3.25: Convergence results of 200 cell (dotted line), 400 cell (solid line) and 800 cell (circles) meshes

and simulation run time while still maintaining acceptable accuracy.

The first simulation carried out involved no variation of piston timing. This was performed to see what collapse geometry would result if all pistons impacted the liner simultaneously. The results were less than ideal: rather than forming a single spherical cavity, the lead liner split the cavity in two separate regions. These results are shown over time in Figure 3.26.
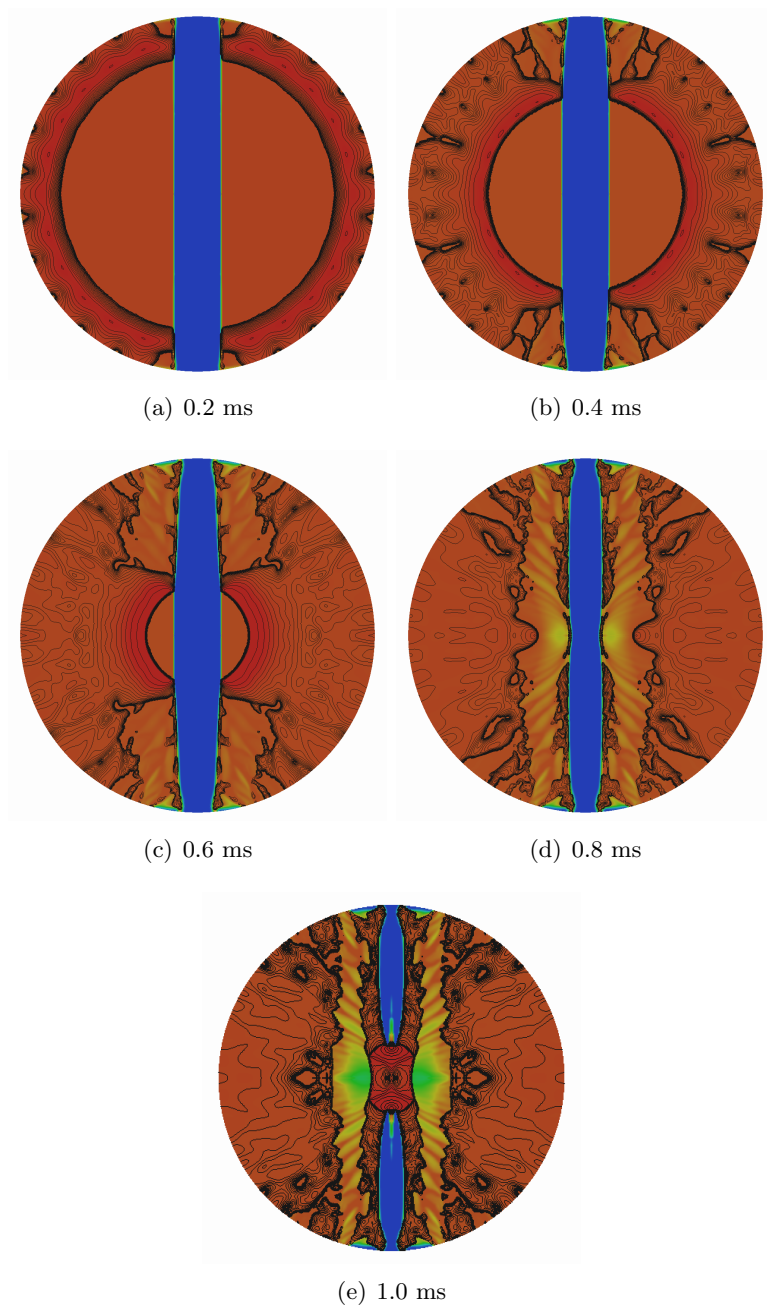
(a) 0.2 ms

(b) 0.4 ms

(c) 0.6 ms

(d) 0.8 ms

(e) 1.0 ms

Figure 3.26: Collapse in 0.2 ms increments with no piston delay

Figure 3.26 is plotted with line contours of pressure and flood contours of density. Regions without line contours or that are blue/green/yellow indicate cavitation. The cavitation pattern observed in Figure 3.26 differs significantly from that in the spherical case shown in Figure 3.25 which suggests that a spherical shell of liquid should delaminate. Instead, all along the cavity wall a layer of lead separates from the bulk of the liner as the pressure wave passes. Initially this layer is cavitated, however radial focusing quickly closes the cavitation and keeps the pressure above atmospheric. Collapse of the cavity wall is quite slow near the poles of the reactor where the pressure wave has no opportunity to focus and passes tangentially along the free-surface, but rapid along the equator where the shock has focused considerably and reflects normally from the lead-plasma interface. Although the rapid collapse at the equator is desirable, closure of the ends along the axis of symmetry must be more rapid in order to create a single, well shaped cavity.

Also highly undesirable is the formation of high speed liquid lead jets, as can be seen Figure 3.26e as narrow vertical features emanating from the center of the reactor. These jets have velocities of up to 6 km/s and diameters of 4-6 cm giving them considerable kinetic energy. Consequently it will be important to the continued operation of the reactor that any jets in the final design not be directed to the reactor housing or plasma injectors.

Two other simulations were performed with the timings of piston impact between adjacent pistons varied by $\pm$ 20 microseconds. A positive delay indicates that pistons near the poles impact the liner first while a negative delay is the opposite. With six pistons and 20 μs delay between adjacent pistons, this would mean that the equatorial pistons would impact 100 μs after the pole pistons. Results from the 20 μs delay case are shown in Figure 3.27.

(a) 0.2 ms

(b) 0.4 ms

(c) 0.6 ms

(d) 0.8 ms

(e) 1.0 ms
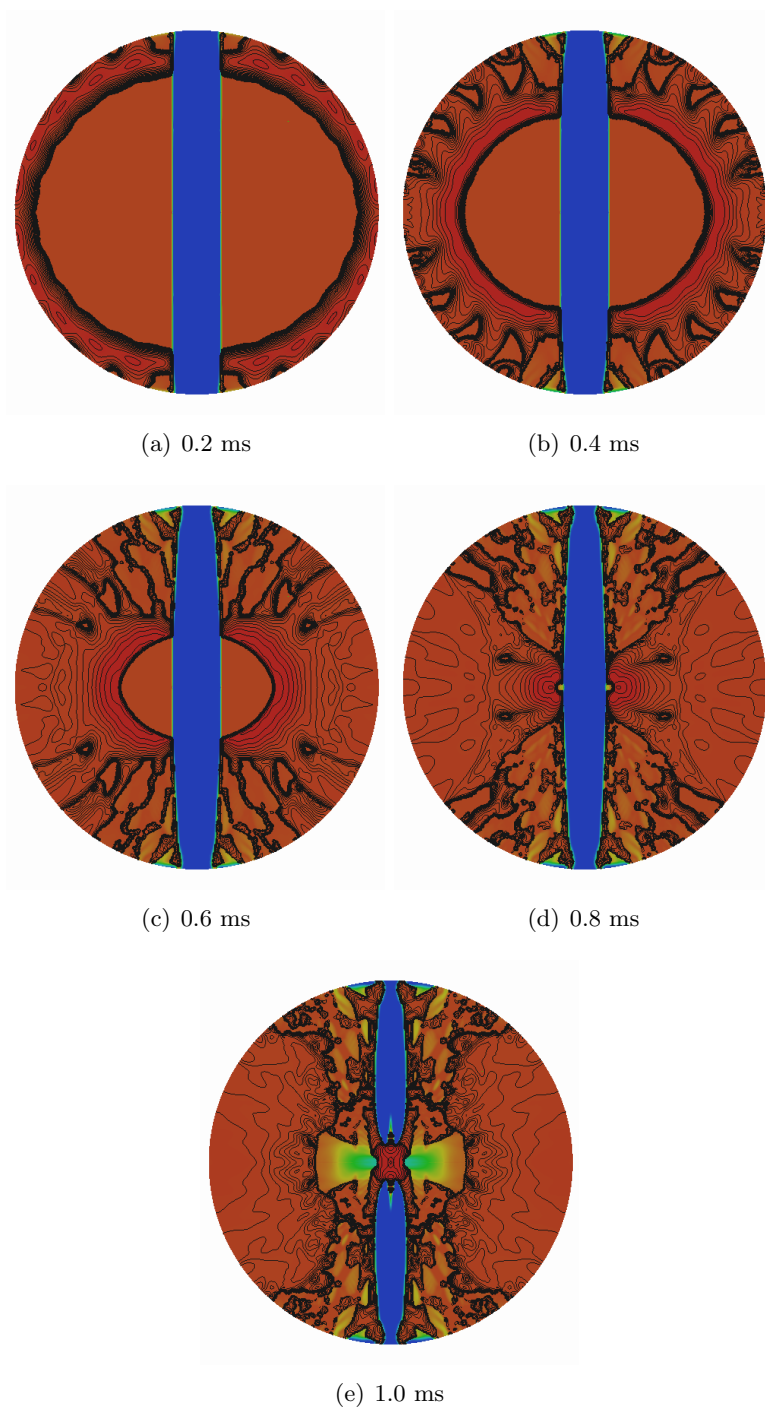
Figure 3.27: Collapse in 0.2 ms increments with 20 μs delay

It was thought that a positive delay would help to correct the closing behavior since for the zero delay case the cavity closed fastest at the equator, and a positive delay would give the portions of the cavity near the poles a head start. This was not the case, as it turns out a positive delay exacerbated the problem.

The reasons for this become apparent when looking at the pressure contour lines. Positive delay mean that the pole pistons fire first, and the equatorial pistons last. Due to this, the shock front that was nearly spherical in the zero delay case becomes squashed along the axis of rotational symmetry, the result being that the front has higher curvature at the equator. Higher curvature at the equator results in a greater level of focusing and higher pressures when the shock finally impinges upon the lead-plasma interface. The reflection then imparts an even higher velocity to the liquid lead at the equator before, sufficient to more than offset the delay in the shock reaching the free surface. Consequently the cavity closes even more quickly at the equator than in the zero delay case.

With this counter-intuitive result, it seemed reasonable that firing the equatorial pistons before the pole pistons would reverse the effect and show an improvement in the closing behavior. This would reduce the focusing near the equator and consequently lower the flyoff velocity of the lead-plasma interface, consequently making the collapse more even. Firing the equatorial pistons first did indeed improve the collapse, and the results are shown in Figure 3.28.

With the delay of the pole pistons, the shock becomes stretched rather than squashed along the axis of rotational symmetry. This reduces the focusing of the shock near the equatorial plane and as a result of that, reduces the flyoff velocity of the lead. This results in a section of the liner approximately 1m in height reaching the centerline at approximately the same time.

(a) 0.2 ms

(b) 0.4 ms
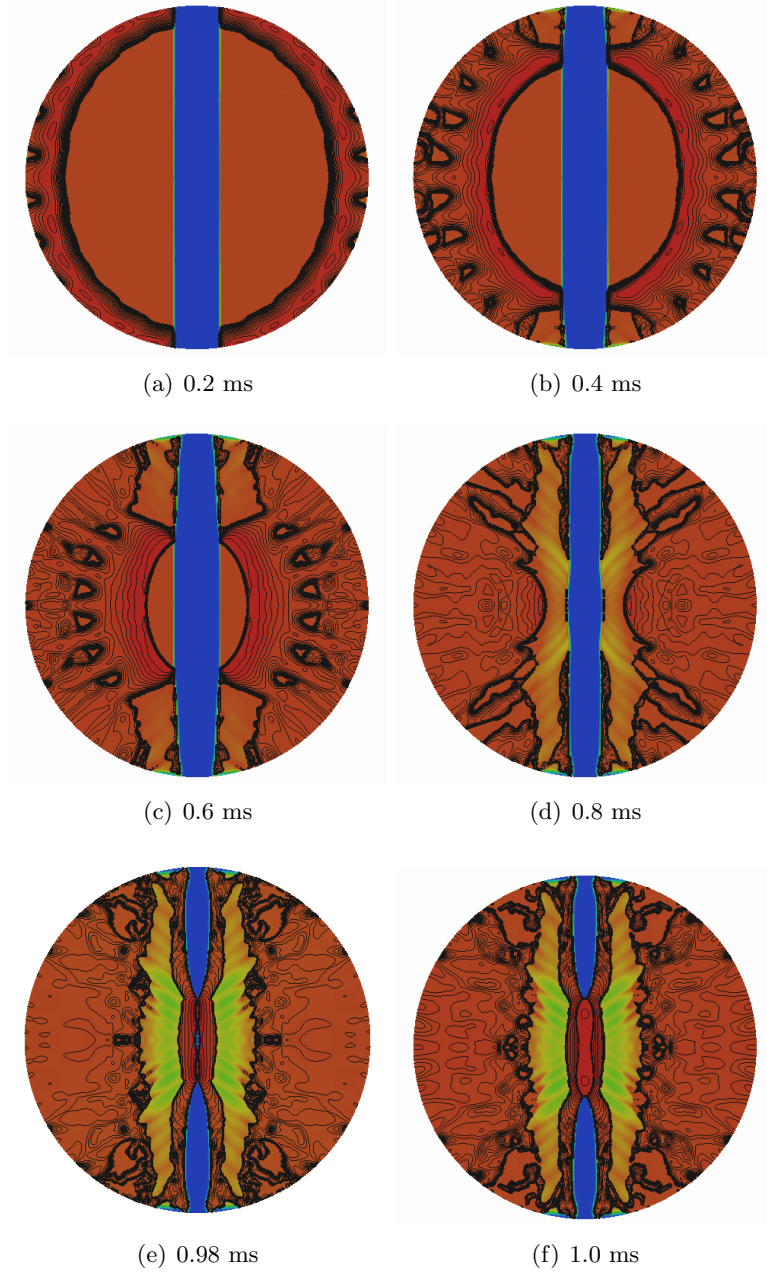
(c) 0.6 ms

(d) 0.8 ms

(e) 0.98 ms

(f) 1.0 ms

Figure 3.28: Collapse with -20 μs delay between pistons, subfigure e shows small cavity produced as a result of piston delay

Although firing the equatorial pistons 20 μs early results in a significant improvement in the collapse behavior, it does not produce a spherical cavity, and although not shown, still produces high velocity jets of lead directed out of the reactor. The piston timings could be adjusted further, perhaps to 40 μs or even higher, however what the effect of this would be on the smoothing of the shock is unclear. It is important that the individual piston pressure waves form a single, moderately smooth wave to reduce the likelihood of Richtmyer-Meshkov instabilities, and the simulations presented here are of too low resolution to resolve instability growth.

Another, more practical concern, is to make sure that the piston delay leaves a path open for the plasma to be injected into. With the closure of the cavity taking from 100 to 200 μs from the time the shock reaches the plasma interface at the equatorial plane, it is likely that the plasma will have to be injected well after the pistons fire in order for the plasma to remain hot enough that a fusion reaction may occur. Consequently piston placement and power may need to be adjust to prevent the cavity from closing off near the poles too quickly (as is evident in all of the cases shown) for the plasma to be squeezed through.

In spite of not obtaining a good collapse, there is reassuring evidence that such a collapse *can* be obtained. It seems reasonable to assume that variables such as piston power and reactor shape will have at least as large an influence on the collapse behavior as the piston timings. That would leave considerable leeway in adjusting and optimizing parameters to obtain a suitable collapse. The process of determining those parameters will, however, be involved and require more time and computational resources than have been applied to the problem in this work.

Chapter 4

Conclusions and Future Work

A two-dimensional solver for the compressible Euler equations was developed that allows simulations of liquid lead, water and air including cavitation and fluid structure interaction. The background theory and numerical methods used were presented, and benchmark results for relevant test cases with analytical and existing numerical results were found to be in good agreement. The developed solver was applied to an investigation of the smoothing behavior of interacting mechanically generated shocks and used to develop an empirical formula to estimate the shock smoothness. Finally the solver was used to simulate the early-time collapse behavior of plasma cavities in a magnetized target fusion reactor. These results were found to differ significantly from one-dimensional simulations performed before this project by General Fusion in that the collapse is not spherical and high velocity jets form. These discoveries highlight the need for detailed numerical modeling of such designs to determine their efficacy.

There is a considerable amount of work remaining in assessing this reactor design. Simulations including the plasma (or some approximation thereof) are needed to capture the late-time collapse behavior of the cavity. Some promise exists in approximating the plasma with a higher-density, lower temperature ideal gas state than the state the plasma actually exists at. Such a state could have a speed of sound more than ten times lower than the actual plasma state while still exhibiting similar compression behavior. This would alleviate issues related to stable time-step size in explicit time advance and only require the introduction of a pressure equilibrium mixture

model to handle ideal gas and liquid mixtures. Such methods are commonplace and have been applied with good success to similar problems such as underwater explosions [9]. A prototype one-dimensional first order solver for spherical coordinate systems has been developed by the author that shows promise in this regard.

There is also a need to extend the shock smoothing simulations to three-dimensional, axisymmetric and spherical coordinate systems. Although it is expected that these system will show higher levels of smoothing than the two-dimensional planar system examined in this work, the actual level of smoothing is unknown. Furthermore, high resolution simulations of interacting mechanically generated shocks reflecting from free-surfaces should be performed to assess how shock non-uniformity affects the growth of Richtmyer-Meshkov instabilities.

Bibliography

[1] T.J. Barth and D.C. Jesperson. The design and application of upwind schemes on unstructured meshes. *AIAA*, 89(0366), 1989.

[2] P. Batten, N. Clarke, and D. M. Causon. On the choice of wavespeeds for the HLLC Riemann solver. *Journal of Scientific Computing*, 18, 1997.

[3] David J. Benson. Computational methods in lagrangian and eulerian hydrocodes. *Comput. Methods Appl. Mech. Eng.*, 99(2-3):235–394, 1992.

[4] R. Courant, K. Friedrichs, and H. Lewy. On the partial difference equations of mathematical physics. *Mathematische Annalen*, 100, 1928.

[5] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152(2):457–492, 1999.

[6] S. K. Godunov. *Difference Methods for Shock Waves*. PhD thesis, Moscow State University, 1954.

[7] J. Gregson, T. Dunbar, and J. J. Lee. Simulation of structural failure from contact underwater explosions. In *78th Shock and Vibration Symposium Proceedings*, 2007.

[8] J. Gregson, R. Link, J. J. Lee, and M. Smith. Coupled simulation of the response of targets to close proximity underwater explosions in two and three dimensions. In *77th Shock and Vibration Symposium Proceedings*, 2006.

[9] R. Link, R. C. Ripley, M. Norwood, T. Josey, L. Donahue, and J. Slater. Analysis of the loading and response of flat plate targets subjected to close-proximity underwater explosions. In *74th Shock and Vibration Symposium Proceedings*, 2003.

[10] M. S. Liou. A sequel to ausm: Ausm+. *J. Comput. Phys.*, 129(2):364–382, 1996.

[11] T. G. Liu, B. C. Khoo, and W. F. Xie. Isentropic one-fluid modelling of unsteady cavitating flow. *Journal of Computational Physics*, 201, 2004.

[12] LSTC. *LS/DYNA Theory Manual*, 1998.

[13] D. J. Mavripilis. Revisiting the least-squares procedure for gradient reconstruction on unstructured meshes. Technical Report CR-2003-212683, NASA, 2003.

[14] C. M. Robinson. A multi-phase equation of state for solid and liquid lead. In *Shock Compression of Condensed Matter*. American Physical Society, 2004.

[15] P. L. Roe. Approximate riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.*, 135(2):250–258, 1997.

[16] R. Saurel and O. Lemetayer. A multiphase model for compressible flows with interfaces, shocks, detonation waves and cavitation. *Journal of Fluid Mechanics*, 431, 2001.

[17] D. P. Schmidt, C. J. Rutland, and M. L. Corradini. A fully compressible, two-dimensional model of small, high-speed, cavitating nozzles. *Atomization Sprays*, 9, 1999.

[18] SIMULIA. *ABAQUS Theory Manual*.

[19] G. A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J. Comput. Phys.*, 27, 1978.

[20] C.R. Sovinec, A.H. Glasser, T.A. Gianakon, D.C. Barnes, R.A. Nebel, S.E. Kruger, S.J. Plimpton, A. Tarditi, M.S. Chu, and the NIM-ROD Team. Nonlinear magnetohydrodynamics with high-order finite elements. *J. Comp. Phys.*, 195:355, 2004.

[21] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics - A practical introduction - 2nd edition*. Springer, Berlin, 1999.

[22] E. F. Toro, M. Spruce, and W. Speares. Restoration of the contact surface in the HLL Riemann solver. *Shock Waves*, 4, 1994.