Contents

1 Introduction

1.1 Background and motivation

A detonation is a combustion-driven shock wave. Typically, a detonation will consist of an inert shock followed by a region of chemical reaction referred to as the reaction zone. Detonations have a wide variety of engineering applications, from obvious military uses to explosive welding, hard rock mining, and materials processing. Detonations can occur in a variety of materials, including gases (such as premixed hydrogen and oxygen), liquids, and solid explosives. Of particular interest in detonation problems is the motion of the detonation shock. Changes to the reaction zone may cause large variations in the strength and speed of the detonation front, so it can not be ignored in modeling detonations. For typical explosives, the reaction zone may be thousands of times smaller than the engineering scale. This multi-scaled nature of detonation can pose problems when trying to predict the motion of the detonation front.

For nearly a century, predictions of steady, unsupported, one-dimensional detonation velocities have been made, and these are in good agreement with experimental observations. This velocity is known as the Chapman–Jouget velocity, D_{CJ} [1], [2]. [1] For nearly as long, engineers have used the steady, one-dimensional results to predict the motion of unsteady, multi-dimensional detonation shock fronts. This assumption is equivalent to a Huygens' construction for the propagation of the detonation shock front, i.e. the detonation front propagates at a constant speed in a direction normal to itself. Although this model for detonation front motion is simple, it does not predict several aspects of multi-dimensional detonation flows. For example, detonation velocities have been observed to change by as much as 40% due to multi-dimensional effects [3]. Failure of detonation waves has also been observed experimentally. Other

[1] Numbers in square brackets [ ] denote entries in the References, beginning on page 101.

dynamics, such as pulsating and cellular detonations, will not be predicted by such a simple propagation rule.

Detonation shock dynamics (DSD) is an asymptotic theory whose key result is an intrinsic partial differential equation (PDE) for the dynamics of the detonation shock front. It will be demonstrated that DSD can predict several aspects of unsteady multi-dimensional detonations accurately. Once an appropriate relation for a particular explosive system is determined, the ability to predict the resulting initial–boundary-value problem for the evolution of the detonation shock front is needed. The resulting PDEs from DSD theory can usually only be integrated analytically for problems with special geometries, such as planar, cylindrical and spherical problems. Even then, it is not always possible to get a solution in closed form. Typical engineering applications involve very complicated boundaries, and the front can experience such topological changes as merging and burning out. The focus of this work will be the numerical solutions of these intrinsic DSD equations and verification of DSD theory by comparison with direct numerical simulation (DNS) of multi-dimensional unsteady detonation problems. Next, a brief overview of DSD is given.

1.2 Detonation shock dynamics (DSD)

A brief review of the types of intrinsic relations derived from DSD theory is given here. Details on how these relations are derived are given elsewhere [4], [5]. The resulting mathematical equation type, linear stability, and dynamics will be given for each relation in Chapters 2, 3, and 4. A review of DSD model boundary conditions will also be given in Chapter 2.

Detonation shock dynamics (DSD) is an asymptotic theory that describes the evolution of a multi-dimensional, curved, near-Chapman–Jouguet (CJ) detonation shock in terms of an intrinsic evolution equation for the shock surface. A complete

mathematical model of detonation [6] consists of the compressible Euler equations, an equation of state with a reaction progress variable, and a reaction-rate law. These equations admit a one-dimensional (1-D), steady traveling wave solution that corresponds to a detonation with a distributed, finite-width reaction zone. The structure calculation for this zone consists of a system of ordinary differential equations (ODEs) that contain a critical point within the zone. These equations, together with the shock conditions serve to define the normal speed of the detonation, D_{CJ}. The CJ detonation is the detonation whose speed corresponds to a sonic state at the end of the reaction zone. This steady solution was first calculated by Zeldovich, von Neumann and Doring, and is thus called the ZND solution.

The shock-evolution equations of DSD theory are derived from an asymptotic theory in which it is assumed that the curved shock has a large radius of curvature compared with the characteristic 1-D reaction-zone length, and that the important dynamic time scale is slow compared with the transit time for particles through the reaction zone [5], [7]. The intrinsic relation comes from the solution to a nonlinear eigenvalue problem.

In particular, three such intrinsic relations will be examined. The first is a relation between the normal detonation velocity, D_n, and the total shock front curvature, κ. The second is a relation between the time derivative of the normal detonation velocity, $\dot{D}_n$, the detonation velocity, D_n, and the total curvature, κ. The third is a relation between the second normal time derivative of normal detonation velocity, $\ddot{D}_n$, the first time derivative of the normal detonation velocity, $\dot{D}_n$, the normal detonation velocity, D_n, the time derivative of the total curvature, $\dot{\kappa}$, and the total curvature, κ. These relations are referred to as the $D_n - \kappa$, $\dot{D}_n - D_n - \kappa$ and the $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relations, respectively.

The simplest form of the intrinsic surface-evolution equation derived from DSD

theory is the $D_n - \kappa$ relation. The shock normal is chosen to point in the direction of the unreacted explosive and the curvature, κ, is defined to be positive when the shock is convex. Physically, positive curvature corresponds to a diverging detonation in which the shock is of convex shape; and D_n is below the plane CJ value, D_{CJ}, for $\kappa > 0$. When the curvature has the opposite sign, $\kappa < 0$, the shock has a concave shape and D_n lies above D_{CJ}. The resulting front dynamics are stable since a $D_n - \kappa$ relation is parabolic, as discussed in Chapter 2. The physical justification for modeling the shock dynamics in such a simple way is as follows.

In the streamwise direction, the reaction zone that supports the detonation resembles the classical ZND structure. Although the reaction zone is not strictly steady for multi-dimensional detonation, it continues to have the property that the shock is influenced only by the subsonic region between the sonic curve and the detonation shock curve. This insulation of the shock from the vast region that follows the reaction zone leads, in the limit of weak shock curvature (measured relative to the distance from the shock to the sonic curve), to the result that the normal detonation speed, D_n, is a function of the shock curvature, κ (under the assumption of sufficiently slow dynamics).

Although the shock is insulated from the far-field flow in the streamwise direction, the reaction zone provides a path by which disturbances can propagate in the direction transverse to the shock-normal direction. In particular, the disturbance generated at the edge of the explosive, where the high-pressure detonation products expand to low pressure, propagates through the reaction zone in the transverse direction, leading to a substantial decrease in the pressure of the reaction zone, even far from the edge. More than any other influence, these lateral rarefactions from the edge of the explosive control the speed and hence the shape and location of the detonation shock.

The $\dot{D}_n - D_n - \kappa$ relation, unlike the $D_n - \kappa$ relation, is hyperbolic, under appro-

priate conditions as discussed in Chapter 3. The relation is obtained when the detonation velocity varies far from D_{CJ}, and time dependence (i.e. $\dot{D}_n$) becomes of the order of the curvature in the asymptotic theory [4]. Since the dynamics are governed by a hyperbolic PDE, disturbances to the reaction zone propagate laterally at finite speeds. The same qualitative behavior is also true of the full Euler equations. This fact will become critical when disturbances (i.e. $\kappa, \dot{D}_n, D_n - D_{CJ}$) become large.

The $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation is also hyperbolic under appropriate conditions discussed in Chapter 4. It is derived in the limit of large activation energy and sufficiently slow dynamics and small curvatures [4]. It may generate very rich dynamics, including pulsating and cellular dynamics for various parameters in the equation of state and reaction-rate law. Further discussion of these dynamics is given in Chapter 4.

1.3 Level-set methods

Once a propagation rule has been determined, designers need the ability to predict the dynamics of the detonation front. In a typical application, the explosive will detonate at several different points, and the propagating fronts will evolve independent of one another, join, and interact with inert confining materials. The evolving detonation shock front can experience great topological complexity. Standard front-tracking methods applied to such problems are either very difficult to program, or are limited to special geometries and initial conditions. New techniques, known as level-set methods [8], avoid the problem associated with the topological complexity. The key idea is to represent the front as a contour of a higher-dimensional function. In doing so, the contour of this level-set function can experience topological changes, while the function itself remains continuous. Examples will be given below.

Osher and Sethian [8], discussed a novel and powerful imbedding concept that

has an underlying simplicity for the calculations of front motion. Specifically they considered the motion of a surface under the influence of a $D_n - \kappa$ relation. They pointed out some of the difficulties of attempting a numerical solution of surface dynamics that uses algorithms based on surface parameterizations. These difficulties include the corresponding loss of accuracy due to the bunching of nodes in regions where the front experiences a convergence, which results in a loss of stability of the method. And in regions of expansion, nodes diverge, and new nodes must be added to maintain stability. Rezoning is thus an essential feature of such methods. Furthermore, there is the logical complexity in the programming required to handle complex and perhaps unforeseen interactions, when sections of shock merge or break apart.

For a physical simulation that uses an underlying surface-parameterization method, a separate and independent description of the topology of each disparate segment of the shock surface must be carried along with all the rules that give the details for extinguishing old segments and creating new ones. A programmer who deals with the issues of trying to write reasonably robust code for engineering applications must confront a difficult task with these methods. These issues are especially important when the tracking algorithm is to be used as a subroutine in part of a larger application code that solves problems with great system complexity.

The level-set methods use instead a formulation where the surface of interest is imbedded in a field of one higher dimension in the physical space of the application. The surface of physical interest is found by taking a subset of the field, specifically a constant value of a field function which defines a level contour in two dimensions or a level surface in three dimensions. Thus for a 2-D application, the level curves are imbedded in a 2-D field, and for 3-D applications, the level surfaces are imbedded in a 3-D field. In particular, one solves for the dynamics of the level curves, $\psi =$

constant, where all the level curves obey the intrinsic relation of interest. The level curves of physical interest for the application are the ones that evolve from the initial configuration of the physical problem, where the level constant is used to identify the physically relevant surface, during its evolution. The curve or surface of interest, $\psi = 0$, is then the object of a contour search of the full field of $\psi(x, y, z, t)$.

Figure 1 shows a time snapshot of a representative 2-D level surface, $\psi(x, y, t)$, and its projection onto the x, y-plane. The imbedding relies on the contouring being uniquely defined, such that a single value of $\psi(x, y, t)$ is obtained for each point (x, y) at a given instant of time.

While it might seem that additional computation is required to represent a 2-D surface by a solving for a 3-D field, in fact the gain in logical simplicity leads to computations that are very efficient and accurate. These advantages easily override any perceived increase in computational cost.

1.4 Outline

In Chapter 2, the dynamics of a $D_n - \kappa$ relation are given. In particular, the level-set equation appropriate for a $D_n - \kappa$ relation is derived. It will be shown that the dynamics are given by parabolic evolution, and thus are stable. The numerical algorithm for implementation with arbitrarily complex 2-D boundaries is also presented. Examples of $D_n - \kappa$ relations and their solution via level sets will be given in Chapter 2.

The dynamics of a $\dot{D}_n - D_n - \kappa$ relation are discussed in Chapter 3. A new level-set formulation for a $\dot{D}_n - D_n - \kappa$ relation will be presented. Whitham's theory of geometrical shock dynamics (GSD) is shown to be a relation of the $\dot{D}_n - D_n - \kappa$ form. Preliminary numerical solutions to the $\dot{D}_n - D_n - \kappa$ level-set equations will be given. Examples from GSD will be presented, including a planar shock diffracting

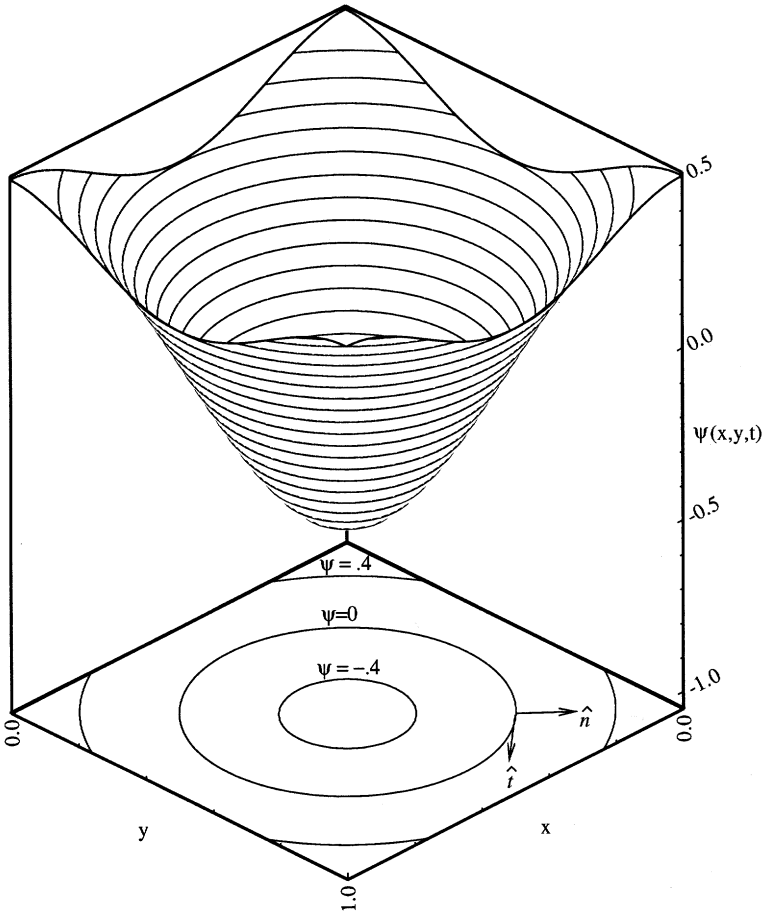

Figure 1: Schematic of level surface and the projection of level curves in the x, y-plane at an instant in time. Also shown are the normal and tangent to the level curve, $\psi = 0$.

over a circular cylinder, originally studied by Bryson and Gross [9].

In Chapter 4, the dynamics of a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation are discussed. The mathematical type and resulting linear stability analysis are also presented. Different regions of parameter space are shown to yield a variety of different dynamics, including 1-D pulsations, cellular dynamics and stable hyperbolic dynamics.

Chapter 5 discusses a numerical method for solving the reactive Euler equations, and presents three direct numerical simulations of detonations. An internal boundary method which allows for arbitrarily complex 2-D boundaries is presented. Also, comparisons between Huygens' construction, the $D_n - \kappa$ relation, and the $\dot{D}_n - D_n - \kappa$ relation are made for each of the three numerical simulations.

Chapter 6 will give conclusions and present future avenues for research in level-set methods, detonation physics, and related subjects.

2 Dynamics of a $D_n - \kappa$ relation

As mentioned in the introduction, detonation shock dynamics is the name given to a body of multi-dimensional theory that describes the dynamics of "near-Chapman–Jouguet" detonations. Its name follows from Whitham's theory of "geometrical shock dynamics," because of the similarity of the mathematical structure of the theories. The engineering application of DSD was originally set forth in two papers [10], [7]. The simplest result of DSD theory is that under suitable conditions, the detonation shock in the explosive propagates according to the simple formula

$$D_n = D_{CJ} - \alpha(\kappa), \tag{1}$$

where D_n is the normal velocity of the shock surface, D_{CJ} is the 1-D, steady, Chapman–Jouguet velocity for the explosive, and $\alpha(\kappa)$ is a function of curvature κ, that is a material property of the explosive. Figure 2 illustrates the sign of the curvature for a typical detonation shock. A sketch of a typical $D_n - \kappa$ relation for PBX9502 is shown in Figure 3.

2.1 Level-set formulation

Here the level-set method and its application and utility as a tool for computing the dynamics of propagating interfaces is explained. The numerical method used to solve the resulting PDE is also given.

First, notice that a surface (or the shock in DSD) is a subset with a dimension one lower than the space it travels in. The level-set method with applied boundary conditions solves for a field function $\psi(x, y, z, t)$ that depends on physical space and time, and the field identifies surfaces of constant values of ψ. The surface $\psi(x, y, z, t) = 0$ is typically identified with the surface of physical interest. Therefore, the computational task involves computing a field in space–time, and then exhibiting the surface

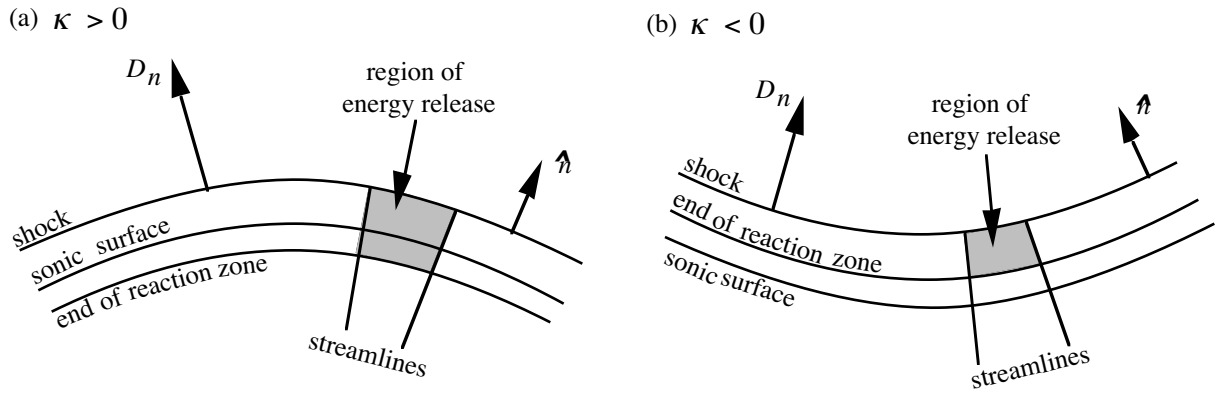

Figure 2: A snapshot of the x, y-plane, showing a diverging and a converging detonation. For a diverging detonation, the transverse dimension of the region of chemical-energy release is smaller than the dimension of the region of shock surface that it supports (the detonation speed falls below D_{CJ}). For a converging detonation the reverse is true and the detonation speed exceeds D_{CJ}.

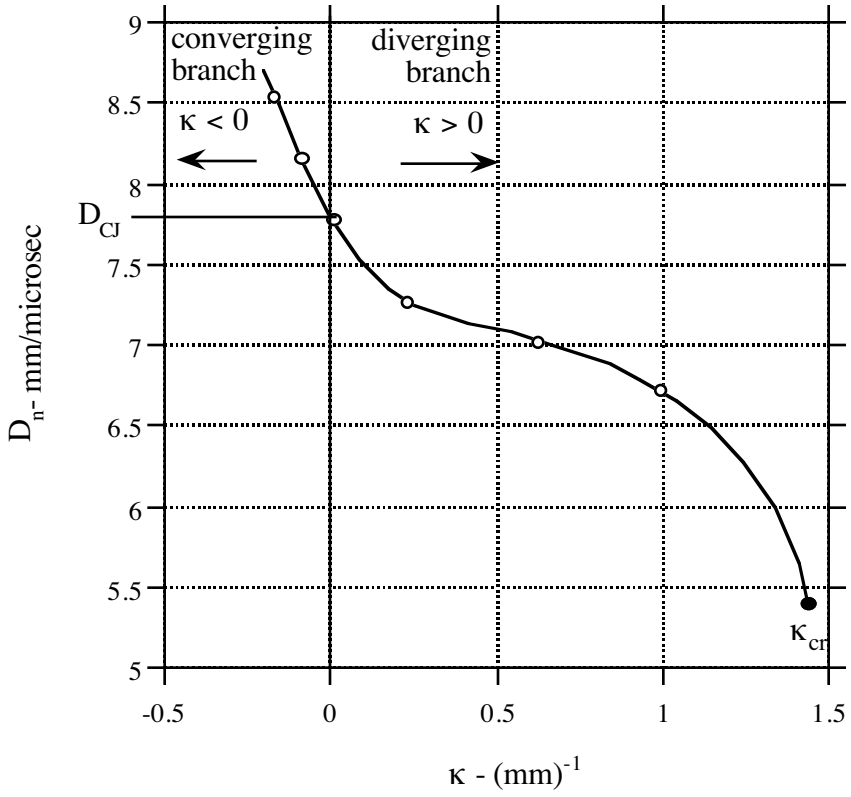

Figure 3: The $D_n - \kappa$ relation for a typical condensed phase explosive after Bdzil et al.'s calibration of PBX9502 [11].

of interest by searching for the special surface $\psi = 0$. Since a level curve is given by $\psi(x, y, z, t) =$ constant, it follows that its total derivative is zero, i.e.

$$\frac{\partial \psi}{\partial t} + \frac{\partial \psi}{\partial x}\frac{dx}{dt} + \frac{\partial \psi}{\partial y}\frac{dy}{dt} + \frac{\partial \psi}{\partial z}\frac{dz}{dt} = 0,$$

where the time derivatives, dx/dt and so on, are the components of the surface velocity $\vec{D}$, defined by that particular level curve. In coordinate-independent form the above equation is

$$\frac{\partial \psi}{\partial t} + \vec{\nabla}\psi \cdot \vec{D}(\kappa) = 0. \tag{2}$$

The outward surface normal, $\hat{n}$, is chosen to be positive in the direction of *outward* propagation. (In the physical application the detonation shock propagates from the burnt explosive towards the unburnt explosive and the positive normal points into the unburnt material.) In terms of the level-set function, the normal is given by $\hat{n} = \vec{\nabla}\psi/|\vec{\nabla}\psi|$. The total curvature satisfies the relation

$$\kappa \equiv \kappa_1 + \kappa_2 = \vec{\nabla} \cdot \hat{n}. \tag{3}$$

Using $\vec{D} \cdot \hat{n} = D_n$, and $\vec{\nabla}\psi \cdot \hat{n} = |\vec{\nabla}\psi|$ in (2), one obtains a Hamilton–Jacobi-like equation for the level-set function that is mainly used in the following discussions:

$$\frac{\partial \psi}{\partial t} + D_n(\kappa)|\vec{\nabla}\psi| = 0. \tag{4}$$

The curvature κ is simply related to the level-set field by using the definition of the curvature from (3) and by then carrying out the indicated differentiations. For example, for two dimensions and for Cartesian coordinates, the curvature is given by

$$\kappa = \frac{\psi_{xx}\psi_y^2 - 2\psi_{xy}\psi_x\psi_y + \psi_{yy}\psi_x^2}{(\psi_x^2 + \psi_y^2)^{3/2}}. \tag{5}$$

In summary, the shock (i.e. the surface of physical interest) is assigned the level $\psi = 0$, while the unburnt material has $\psi > 0$ and the burnt material has $\psi < 0$. A

unique way to specify ψ initially is to choose ψ equal to the signed minimum distance from the initial shock surface. Equation (4) is then a partial differential equation for the level-set function ψ, that is to be solved subject to its initial data.

The solution of the PDE with initial and boundary conditions generates the field $\psi(x, y, z, t)$, and the location of the shock is then simply found by search for the level surface $\psi = 0$. This is easily done by creating a table of arrival times of the shock across the computational grid. This is referred to as the burn table. Numerically generating a burn table is discussed in Section 2.8

2.2 Mathematical type and stability

A relation between the normal detonation velocity, D_n, and curvature, κ, is a parabolic evolution equation. This fact is most easily seen as follows. Suppose a linear $D_n - \kappa$ relation, $D_n = 1 - \alpha\kappa$, is given and further that the level-set function (and thus the front) may be written as

$$\psi(x, y, t) = y - y_s(x, t) = 0.$$

Then the normal is given by

$$\hat{n} = \frac{\vec{\nabla}\psi}{|\vec{\nabla}\psi|} = \frac{-y_{s,x}}{(1 + y_{s,x}^2)^{1/2}}\hat{\imath} + \frac{1}{(1 + y_{s,x}^2)^{1/2}}\hat{\jmath}$$

and the curvature is

$$\kappa = \vec{\nabla} \cdot \hat{n} = \frac{-y_{s,xx}}{(1 + y_{s,x}^2)^{3/2}}.$$

Substitution of these into (4) and assuming the slope of the front is small yield

$$y_{s,t} - (1 + \alpha y_{s,xx}) = 0$$

and changing variables $y_s(x, t) = z_s(x, t) + t$ gives

$$z_{s,t} - \alpha z_{s,xx} = 0,$$

which is the linear (parabolic) heat equation. Without going through the details, it is obvious that for $\alpha > 0$, the above linearized PDE is stable.

2.3 Numerical implementation

2.3.1 Interior differencing

Here a brief description of the numerical method we use for solving the level-set equation (4) on a fixed Eulerian finite difference grid is given. For the interior algorithm, the algorithm follows Osher and Sethian [8]. The time advance of the level-set equation

$$\frac{\partial \psi}{\partial t} + D_{CJ}|\vec{\nabla}\psi| - \alpha(\kappa)|\vec{\nabla}\psi| = 0 \tag{6}$$

is split into two steps. First, ψ is advanced using the sub-operator, L_P, defined by the first and third terms in equation (6). This step is then followed by the advance for the sub-operator, L_H, defined by the first and second terms in equation (6). The motivation for this operator splitting is related to the fact that L_H is a hyperbolic operator and L_P is a parabolic operator. Different numerical methods are thus appropriate for these different type operators. The differencing for each of the three terms in (6) is now considered separately.

For the time derivative, first-order, forward Euler differencing

$$\frac{\partial \psi}{\partial t} = \frac{\psi_{i,j}^{n+1} - \psi_{i,j}^{n}}{\Delta t} \tag{7}$$

is used, where i and j represent the x and y node and n represents the time level in the usual way. Higher-order Runge–Kutta type schemes can be used and have been derived in [8] and [12]. It will be shown in Section 2.7 that the forward Euler method is sufficient to yield second-order convergence.

The first-order spatial derivatives in the second term in (6) are calculated using a combination of upwinding and essentially nonoscillatory (ENO) interpolation. In

the following text, first-order interpolation is equivalent to first-order differencing and second-order interpolation is equivalent to second-order differencing. Consider first a 2-D problem using upwinding and first-order interpolation. An approximation to $|\vec{\nabla}\psi|$ is needed, and thus ψ_x and ψ_y are needed. First, construct four linear interpolants between node $\psi^n_{i,j}$ and the four surrounding nodes, $\psi^n_{i+1,j}$, $\psi^n_{i-1,j}$, $\psi^n_{i,j+1}$, and $\psi^n_{i,j-1}$. Define the usual forward and backward difference operators

$$D^+_x\psi^n_{i,j} = \frac{\psi^n_{i+1,j} - \psi^n_{i,j}}{\Delta x}, \qquad D^-_x\psi^n_{i,j} = \frac{\psi^n_{i,j} - \psi^n_{i-1,j}}{\Delta x},$$

$$D^+_y\psi^n_{i,j} = \frac{\psi^n_{i,j+1} - \psi^n_{i,j}}{\Delta y}, \qquad D^-_y\psi^n_{i,j} = \frac{\psi^n_{i,j} - \psi^n_{i,j-1}}{\Delta y}.$$

Next combine these differences to define the following first-order upwind difference:

$$\begin{aligned} |\vec{\nabla}\psi|^n_{i,j} = [\ & f^+_x(D^+_x\psi^n_{i,j}) + f^-_x(D^-_x\psi^n_{i,j}) \\ & + f^+_y(D^+_y\psi^n_{i,j}) + f^-_y(D^-_y\psi^n_{i,j})]^{\frac{1}{2}}, \end{aligned} \tag{8}$$

where

$$f^+_x(a) = \begin{cases} a^2, & \text{if } D^+_x\psi^n_{i,j} < 0 \\ 0, & \text{otherwise} \end{cases} \qquad f^-_x(a) = \begin{cases} a^2, & \text{if } D^-_x\psi^n_{i,j} > 0 \\ 0, & \text{otherwise} \end{cases}$$

$$f^+_y(a) = \begin{cases} a^2, & \text{if } D^+_y\psi^n_{i,j} < 0 \\ 0, & \text{otherwise} \end{cases} \qquad f^-_y(a) = \begin{cases} a^2, & \text{if } D^-_y\psi^n_{i,j} > 0 \\ 0, & \text{otherwise} \end{cases}$$

To achieve second-order spatial accuracy, four quadratic interpolants, each using three nodes, are used. For each of the four directions (interpolants between $\psi^n_{i,j}$ and $\psi^n_{i-1,j}$, $\psi^n_{i,j}$ and $\psi^n_{i+1,j}$, $\psi^n_{i,j}$ and $\psi^n_{i,j-1}$, and $\psi^n_{i,j}$ and $\psi^n_{i,j+1}$), there are two choices for the interpolant. For example, consider the linear interpolant between $\psi^n_{i,j}$ and $\psi^n_{i+1,j}$. To construct a quadratic interpolant, another node, either $\psi^n_{i-1,j}$ or $\psi^n_{i+2,j}$, is used. The choice is made by picking the node which gives the smallest second derivative in magnitude. If the second derivatives are of opposite sign, then the second-order correction is taken to be zero. This same procedure is used in the other three directions resulting in the following second-order scheme:

$$
\begin{aligned}
|\vec{\nabla}\psi| = [&f_x^+(D_x^+\psi_{i,j}^n - \frac{\Delta x}{2}\min\mathrm{mod}(D_x^-D_x^+\psi_{i,j}^n, D_x^+D_x^+\psi_{i,j}^n)) \\
&+f_x^-(D_x^-\psi_{i,j}^n + \frac{\Delta x}{2}\min\mathrm{mod}(D_x^-D_x^-\psi_{i,j}^n, D_x^+D_x^-\psi_{i,j}^n)) \\
&+f_y^+(D_y^+\psi_{i,j}^n - \frac{\Delta y}{2}\min\mathrm{mod}(D_y^-D_y^+\psi_{i,j}^n, D_y^+D_x^+\psi_{i,j}^n)) \\
&+f_y^-(D_y^-\psi_{i,j}^n + \frac{\Delta y}{2}\min\mathrm{mod}(D_y^-D_y^-\psi_{i,j}^n, D_y^+D_y^-\psi_{i,j}^n))]^{\frac{1}{2}}, \qquad (9)
\end{aligned}
$$

where the min mod function is defined by

$$
\min\mathrm{mod}(a,b) = \begin{cases} a, & \text{if } |a| \le |b| \text{ and } ab > 0 \\ b, & \text{if } |b| < |a| \text{ and } ab > 0 \\ 0, & \text{otherwise.} \end{cases}
$$

The third term in (6) is essentially a diffusion term, and second-order central differences are used to calculate κ, and thus $\alpha(\kappa)$. Central differences are also used to calculate $|\vec{\nabla}\psi|$ in this term.

The calculations in Sections 2.8 and 5.6 use the above second-order scheme.

2.3.2 Initial conditions

The level-set function, ψ, must be defined initially at $t = 0$ where $\psi(x, y, t = 0) = 0$ represents the initial shock locus. One can choose $\psi(x, y, t = 0)$ to be the signed distance from the initial shock locus, with $\psi(x, y, t = 0)$ positive in the unburnt material and $\psi(x, y, t = 0)$ negative in the burnt material. Thus the normal, $\hat{n}$, points into the unburnt material. For example, two initially expanding cylindrical shocks with radii equal to r located at (x_1, y_1) and (x_2, y_2) would be given by

$$
\psi(x, y, t = 0) = \min[\sqrt{(x - x_1)^2 + (y - y_1)^2} - r, \sqrt{(x - x_2)^2 + (y - y_2)^2} - r],
$$

while two collapsing cylindrical shocks at the same location and radii would be given by

$$
\psi(x, y, t = 0) = \max[r - \sqrt{(x - x_1)^2 + (y - y_1)^2}, r - \sqrt{(x - x_2)^2 + (y - y_2)^2}].
$$

2.4 Boundary conditions

Three types of boundary conditions have been implemented into our level-set formulation. These are symmetric (perfectly reflecting), nonreflecting (inflow/outflow), and angle (physical) boundary conditions. The formulation uses two levels of ghost nodes to enforce the particular boundary conditions. The symmetric boundary condition is trivially satisfied by reflecting the values of ψ from the interior to the exterior. For example if $x = 0$ is a symmetry plane and $\psi^n_{0,j}$ is at $x = 0$, then $\psi^n_{-1,j} = \psi^n_{1,j}$ and $\psi^n_{-2,j} = \psi^n_{2,j}$.

The nonreflecting boundary conditions are applied by using quadratic extrapolation. This condition is equivalent to keeping the second derivative along the normal to the boundary as a constant. The upwinded first-order spatial derivatives do not need to have ghost nodes, since they look in the proper direction. However, ghost nodes are used in the calculation of the second-order derivatives and the curvature at the boundary. For example, if nonreflecting boundary conditions are applied at $x = 0$, then $\psi^n_{-1,j} = 3\psi^n_{0,j} - 3\psi^n_{1,j} + \psi^n_{2,j}$, etc.

2.4.1 Angle boundary conditions

In [13], a set of model DSD boundary conditions were formulated that involve the angle that the local shock normal, $\hat{n}_s$, makes with the outward-pointing normal vector of the boundary, $\hat{n}_b$, which is refered to as ω. Equivalently ω is the angle between the tangent to the edge and the tangent to the shock. See Figure 4. A physical justification for the DSD angle boundary condition will be given next, followed by a summary of the model boundary conditions.

The condition to be applied depends on the flow type as witnessed by an observer riding with the point of intersection of the local shock and the edge. The boundary conditions are formulated by an analysis of the local singularities admitted by the

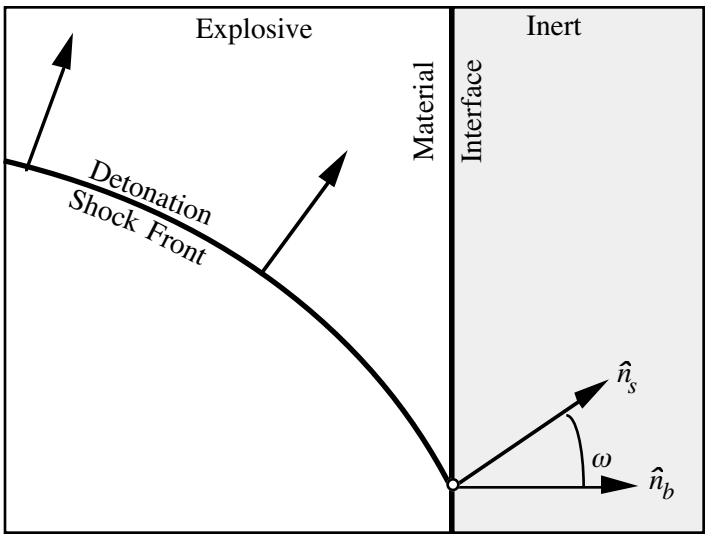

Figure 4: Definition of the angle, ω, and the normals, $\hat{n}_s$ and $\hat{n}_b$.

Euler equations [14] and the results are summarized in this section. The flow type is characterized by the local sonic parameter, $\mathcal{S}$, evaluated at the shock in the detonation reaction zone and as measured by an observer moving with the point of intersection of the detonation shock and the material interface

$$\mathcal{S} \equiv C^2 - (U_n)^2 - D_n^2 \cot^2(\omega) \quad , \tag{10}$$

where C is the sound speed in the explosive, U_n is the explosive particle velocity in the shock-normal direction and D_n is the detonation normal speed. When $\mathcal{S} < 0$, the flow is locally supersonic at the edge and no boundary condition is applied. The application of *no* boundary condition is, in practice, the application of a continuation boundary condition, where information flows from the interior to the exterior of the domain. More will be said about the numerical implementation of the continuation boundary condition in Section 3.3. When $\mathcal{S} > 0$, the flow is locally subsonic and the presence of the edge influences the reaction zone. The form of the boundary condition for the $\mathcal{S} > 0$ case is determined by the properties of the inert material

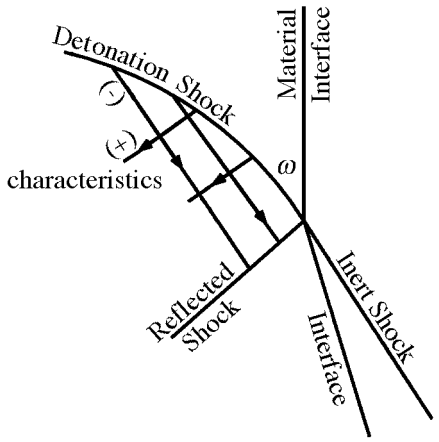

Figure 5: DSD boundary conditions. A snapshot of the x, y-plane showing the *supersonic* type of explosive–inert boundary interaction. The magnitude of ω controls the type of interaction that occurs. This figure corresponds to a supersonic flow in the explosive, measured relative to an observer riding with the shock–edge intersection point.

that is adjacent to the explosive.

The problem geometry and the various cases—supersonic, sonic and subsonic—that are modeled correspond to a steady flow in the reference frame of the shock–edge intersection point. Figures 5-7 show instantaneous time snapshots of the interaction between the explosive wave and confining inert material. The explosive induces a shock into the inert material (labeled *inert shock*), which typically generates a reflected wave into the explosive (labeled either the *reflected shock* or the *limiting characteristic* depending on whether the reflected wave is a shock or a rarefaction, respectively).

Figure 5 corresponds to a supersonic flow, $\mathcal{S} < 0$. As previously mentioned, no boundary condition is applied irrespective of the degree of confinement that the inert

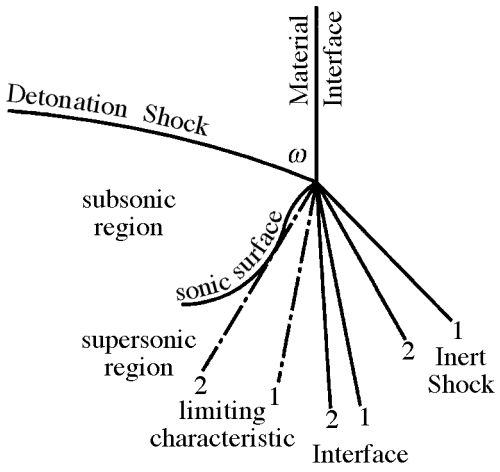

Figure 6: DSD boundary conditions. A snapshot of the x, y-plane showing the *sonic* type of explosive–inert boundary interaction. The magnitude of ω controls the type of interaction that occurs. This figure corresponds to a sonic flow in the explosive, measured relative to an observer riding with the shock–edge intersection point.

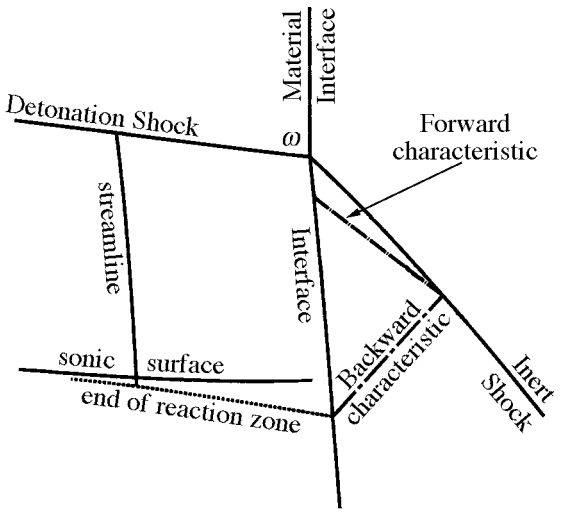

Figure 7: DSD boundary conditions. A snapshot of the x, y-plane showing the *subsonic* type of explosive–inert boundary interaction. The magnitude of ω controls the type of interaction that occurs. This figure corresponds to a subsonic flow in the explosive, measured relative to an observer riding with the shock–edge intersection point.

provides to the explosive. The shock reflected into the explosive does not influence the detonation shock. As the angle ω is increased to the value ω_s where $\mathcal{S} = 0$, the flow in the explosive turns sonic and therefore can sense the degree of confinement that the adjacent inert material provides. Note that ω_s is a constant in our model, given by the explosive equation of state.

Figure 6 shows two cases, labeled as 1 and 2, that correspond to different degrees of confinement provided by the inert material. For these cases, the pressure decreases towards the right of the explosive sonic locus. Case 1 corresponds to weak confinement, for which the pressure induced into the inert material is considerably below the detonation pressure at the edge. The influence of the confinement propagates into the explosive no farther to the left than the limiting characteristic labeled 1. The subsonic part of the reaction zone remains totally unaffected by the confinement, and the flow remains *sonic* at the shock–edge intersection point. The detonation propagates as if it were totally unconfined.

As the degree of confinement is increased further, the drop in pressure in going from the explosive to the inert material becomes less, until at some critical degree of confinement the influence of the inert material extends up to the limiting characteristic labeled 2. At this critical degree of confinement, the detonation continues to propagate as if it were unconfined. Any further *increase* in the confinement destroys the sonic isolation of the reaction zone from the influence of the confinement and leads to the case shown in Figure 7.

If for the angle ω_s, corresponding to $\mathcal{S} = 0$, the pressure induced into the confining inert material is greater than the pressure in the explosive, then the flow that develops is that shown in Figure 7. The reflected wave can now enter into the subsonic part of the reaction zone. This results in an increase in pressure in the reaction zone and the concomitant increase of the normal shock velocity, D_n. The angle ω increases until

the pressure in the inert and reaction zone balance. Since the flow in the explosive is subsonic, a reflected shock is not generated in the explosive. The value of ω at the point of pressure equilibrium is ω_c. The value of ω_c is a constant that depends only on the specific explosive–inert pair. It is easily calculated from a shock polar analysis, assuming no reflected wave in the explosive.

In summary, the boundary interaction has the following properties: (i) When the flow in the explosive is supersonic (i.e., $\omega < \omega_s$), continuation (outflow) boundary condition is applied. This corresponds to extrapolating the front to the exterior, without changing the angle at the boundary. (ii) When the flow turns sonic $\omega = \omega_s$, two cases can arise: (a) The pressure induced in the inert is below that immediately behind the detonation shock and the confinement has no influence on the detonation. The sonic boundary condition is applied, $\omega = \omega_s$. (b) The pressure induced in the inert is above that immediately behind the detonation shock. The angle ω increases (i.e., $\omega > \omega_s$) until the pressure in the inert and explosive are equilibrated. This angle $\omega = \omega_c$ is the equilibrium value for the angle and is regarded as a material constant that is a function of the explosive–inert pair. Thus the boundary condition recipe can be summarized as follows: (1) A continuation boundary condition is applied for supersonic flows and (2) when the flow becomes either sonic or subsonic, ω is bounded from above by a critical angle ω_c (unique for each explosive–inert pair) that is determined using the above discussion.

Figure 8 shows a time history of the evolution of the angle $\omega(t)$ along the edge of confinement that corresponds to a typical application. Figure 8(a) shows a detonation interacting with an edge at three different times, t_1, t_2, t_3. At time t_1, the shock–edge intersection is highly oblique and the supersonic (continuation) boundary condition applies. At time t_2, it is assumed that the intersection angle first becomes sonic, $\omega = \omega_s$. If the confinement is heavy enough, a rapid acoustic transient can take

place and a rapid adjustment to the equilibrium value, ω_c, can occur. After that adjustment, shown at t_3 (say), the angle remains at $\omega = \omega_c$ which corresponds to that for the explosive–confinement pair. The right-hand portion of Figure 8(a) shows the time history of the shock interaction at the edge. The value of $\omega(t)$ is determined by the solution for $\omega < \omega_s$. Once ω_s is attained, a rapid jump to ω_c occurs and from then on $\omega = \omega_c$ applies. This is shown in the right hand portion of the figure. If the confinement were sufficiently weak, no jump to ω_c would be needed, and the angle would simply remain at ω_s. This case is shown by the broken line.

Figure 8(b) shows a different scenario. It is assumed that the detonation is initially flat and $\omega = \pi/2$. For heavy confinement, a rapid acoustic transition to $\omega = \omega_c$ is assumed to occur and then maintained from then on. If the confinement is sufficiently light, then the transition is from $\omega = \pi/2$ to $\omega = \omega_s$. Again this is shown in the broken line.

2.4.2 Implementation of angle boundary conditions

Of the three boundary conditions, angle boundary conditions need the most attention. A class of physical boundary conditions within DSD theory concerns detonation waves interacting with inert boundaries were described in Section 2.1. For each inert–explosive pair, two angles are needed to define the boundary conditions at an interface. These are the sonic angle, ω_s, and the steady state angle, ω_c.

In general, the location of the inert–explosive interface, where angle boundary conditions need to be applied, can be quite complex. Unfortunately, it is not always simple to find a computational grid (body-fitted grid) whose boundaries coincide with the physical inert–explosive interface. Next an internal boundary method is developed to treat these boundary conditions numerically for arbitrarily complex interfaces on a uniform ($\Delta = \Delta x = \Delta y$) 2-D Cartesian grid. In spirit, this method is similar to the Cartesian boundary method of Leveque [15] and others [16], [17], although the

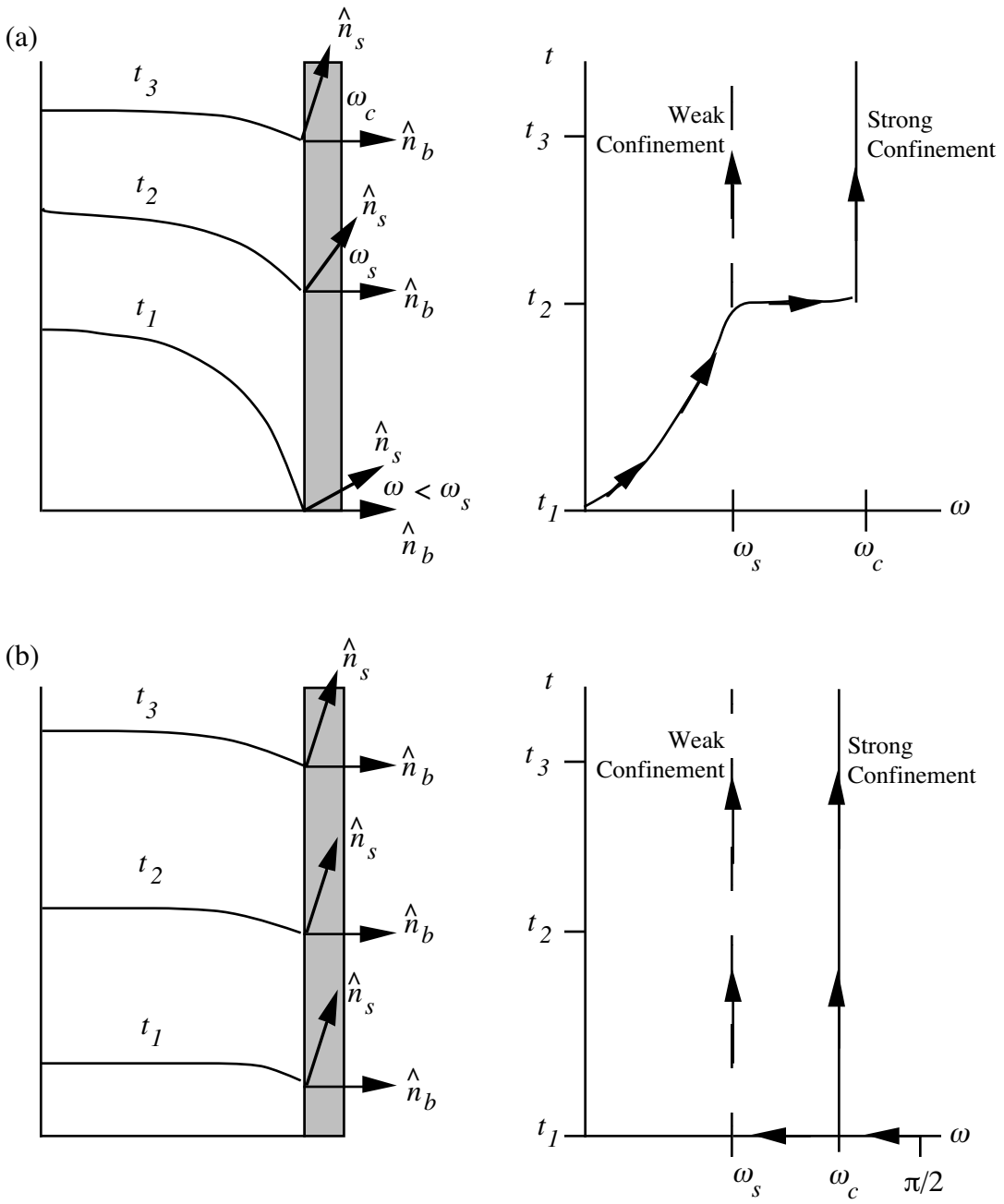

Figure 8: Time histories of shock–edge interactions for typical (a) oblique interactions and (b) normal interactions.

mathematical boundary conditions being applied are quite different. It will be shown that angle boundary conditions involve spatial derivatives of the level-set function, ψ (which are similar to Neumann boundary conditions). The mathematical boundary conditions, and the corresponding numerical implementation, are given next.

First, define a new (nonevolving) level-set function, $\phi(x, y)$, such that $\phi(x, y) = 0$ at the inert–explosive interface. The function $\phi(x, y)$ is defined at computational grid points as $\phi_{i,j}$ (where again i and j correspond to the x-location and y-location, respectively). Define ϕ to be the signed distance function from the inert–explosive interface, with ϕ negative in the explosive and ϕ positive in the inert. To enforce the angle boundary conditions on the interior of the computational domain, an array of (i, j) nodes near $\phi = 0$ will be used. This array of nodes is referred to as the internal boundary (IB) nodes. These IB nodes are found in the following manner. Sweep through the grid, and if at an (i, j) node $\phi_{i,j} > 0$ and if at any of the eight surrounding nodes one of the following conditions is true, $\phi_{i+1,j} \leq 0$, $\phi_{i-1,j} \leq 0$, $\phi_{i,j+1} \leq 0$, $\phi_{i,j-1} \leq 0$, $\phi_{i+1,j+1} \leq 0$, $\phi_{i-1,j-1} \leq 0$, $\phi_{i-1,j+1} \leq 0$ or $\phi_{i+1,j-1} \leq 0$, then the (i, j) node is an IB node. This search is analogous to computationally finding the $\phi = 0$ contour. This search for internal boundary points is done only once at the beginning of the computation. The angle boundary conditions will be enforced by specifying $\psi_{i,j}$ at these IB nodes. Furthermore, the interior differencing of Section 3.1 needs to be applied only at nodes where $\phi_{i,j} \leq 0$, since the others correspond to inert regions.

The inert–explosive interface normal, $\hat{n}_b$, at an IB node is given by

$$\hat{n}_b = n_{bx}\hat{\imath} + n_{by}\hat{\jmath} = \frac{\vec{\nabla}\phi}{|\vec{\nabla}\phi|} \quad , \tag{11}$$

which is approximated by second-order central differences at IB nodes. For each IB node, a locally rotated orthogonal stencil is defined which is aligned with the inert–explosive interface normal, $\hat{n}_b$, and inert–explosive interface tangential unit vector,

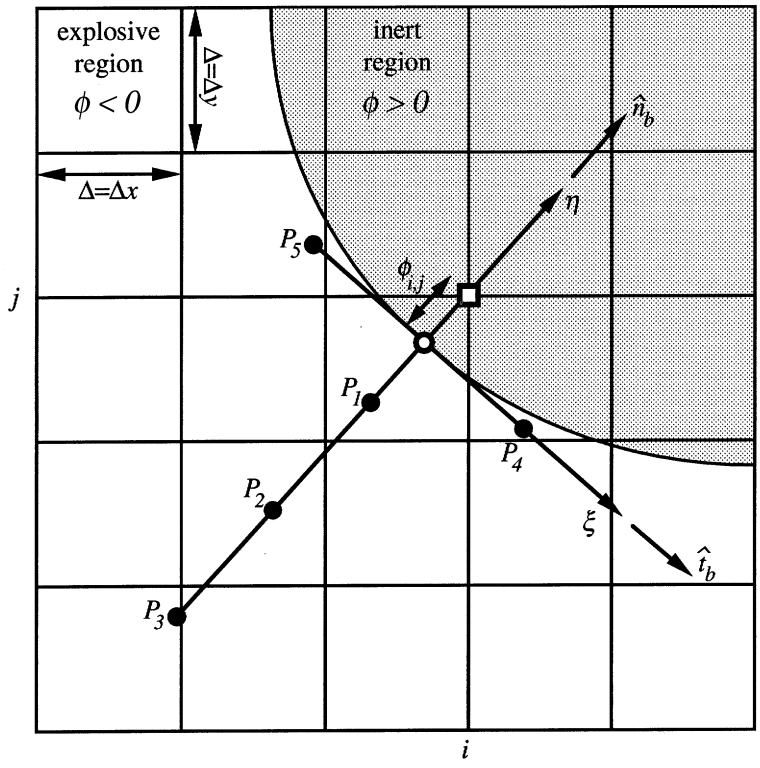

Figure 9: Schematic of internal boundary condition stencil. • interpolated stencil points, □ internal boundary node (i,j), ∘ point where boundary condition is to be applied.

$\hat{t}_b = n_{by}\hat{\imath} - n_{bx}\hat{\jmath}$. The coordinates associated with the $\hat{n}_b$ and $\hat{t}_b$ directions are η and ξ, respectively. See Figure 9. Since the angle boundary condition will involve spatial derivatives of the level-set function, ψ, values of ψ at the discrete points, labeled P_k, associated with each IB node need to be known. These points are given by:

$$P_1 = (\phi_{i,j} - \Delta)\hat{n}_b \quad P_2 = (\phi_{i,j} - 2\Delta)\hat{n}_b$$

$$P_3 = (\phi_{i,j} - 3\Delta)\hat{n}_b \quad P_4 = \Delta\hat{t}_b \quad P_5 = -\Delta\hat{t}_b$$

Values of ψ at these rotated stencil points, P_1, P_2, P_3, P_4, P_5, are given by second-order accurate bilinear interpolation. At every time step, the following algorithm is

applied:

Step 1: Quadratically extrapolate ψ from the interior to the IB nodes along the $\hat{n}_b$ direction. (Assume a supersonic interaction.)

Step 2: Check if interaction at each IB node is subsonic or supersonic.

Step 3: Apply angle boundary condition to all IB nodes which have a subsonic interaction.

Quadratic extrapolation is accomplished by solving

$$\psi_{i,j} = 3\psi_{P_1} - 3\psi_{P_2} + \psi_{P_3} \tag{12}$$

at all IB nodes. In general, $\psi_{P_1}, \psi_{P_2}, \psi_{P_3}$ can be dependent on IB nodes. For example, in Figure 9, ψ_{P_3} will be a linear combination of the three interior values $\psi_{i-1,j-1}$, $\psi_{i-1,j}$, $\psi_{i,j-1}$ and the IB node value $\psi_{i,j}$. Therefore, (12) will result in a system of linear equations, where the number of equations and unknowns is equal to the number of IB nodes. This system is solved by the following iterative method: View (12) as $\psi_{i,j} = F_1(\psi_{i,j})$. Start with an initial guess for each $\psi_{i,j}^{\text{guess}}$, say the value of $\psi_{i,j}$ at the old timestep. Evaluate $F_1(\psi_{i,j}^{\text{guess}})$, and set

$$\psi_{i,j}^{\text{new}} = (1-w)\psi_{i,j}^{\text{guess}} + wF_1(\psi_{i,j}^{\text{guess}}) \quad , \tag{13}$$

where $w < 1$ for the iterative method to converge. Repeat (13) until $\max(|\psi_{i,j}^{\text{new}} - \psi_{i,j}^{\text{guess}}|) < \epsilon\Delta$. The values $w = 0.9$ and $\epsilon = 10^{-3}$ work well and typically converge in ten iterations or less. Note that the number of equations being solved iteratively is of the order $(N_x N_y)^{1/2}$ where N_x and N_y are the number of x and y grid points, so the algorithm is relatively inexpensive compared with the interior scheme.

To check if an interaction at an IB node is subsonic or supersonic, an approximation for the angle, ω, between the shock normal, $\hat{n}_s$, and the inert–explosive interface

normal, $\hat{n}_b$, is needed. The vector, $\hat{n}_b$, is given from (11) and the normal, $\hat{n}_s$, is given by

$$\hat{n}_s = \frac{\psi_\eta}{\sqrt{\psi_\eta^2 + \psi_\xi^2}}\hat{n}_b + \frac{\psi_\xi}{\sqrt{\psi_\eta^2 + \psi_\xi^2}}\hat{t}_b$$

and therefore ω is given by

$$\cos\omega = \hat{n}_s \cdot \hat{n}_b = \frac{\psi_\eta}{\sqrt{\psi_\eta^2 + \psi_\xi^2}} \quad . \tag{14}$$

Approximations to the derivative terms in (14) are needed at the point where the boundary condition is to be applied; see Figure 9. Taylor series expansions reveal the following approximation:

$$\psi_\eta = \frac{\psi_{P_2} - 4\psi_{P_1} + 3\psi_{i,j}}{2\Delta} - \frac{\psi_{P_2} - 2\psi_{P_1} + \psi_{i,j}}{\Delta^2}\phi_{i,j} \quad , \tag{15}$$

where $\phi_{i,j}$ appears in (15) since it is the signed distance from the node (i, j) to the location where the boundary condition is to be applied. See Figure 9. A central difference approximation to ψ_ξ is

$$\psi_\xi = \frac{\psi_{P_4} - \psi_{P_5}}{2\Delta} \quad . \tag{16}$$

Equation (14), together with (15) and (16), gives an approximation to $\cos\omega$ at the boundary point corresponding to the IB node. If $\cos\omega > \cos\omega_s$ then the interaction is supersonic; else the interaction is subsonic.

All IB nodes that have a subsonic interaction must have the angle at the boundary point set to $\omega = \omega_c$. Therefore, the following needs to solved:

$$\cos\omega_c = \hat{n}_b \cdot \hat{n}_s = \frac{\psi_\eta}{\sqrt{\psi_\eta^2 + \psi_\xi^2}} \quad .$$

Solving for the derivative, ψ_η, yields

$$\psi_\eta = \cos\omega_c(\psi_\xi^2 \csc^2\omega_c)^{1/2} \quad . \tag{17}$$

Substitution of (15) and (16) into (17), and solving for $\psi_{i,j}$ yields

$$\psi_{i,j} = \frac{\cos\omega_c((\psi_{P_4} - \psi_{P_5})^2 \csc^2\omega_c)^{1/2} - \Delta^2(\psi_{P_2} - 4\psi_{P_1}) + \Delta\phi_{i,j}(2\psi_{P_2} - 4\psi_{P_1})}{3\Delta^2 - 2\Delta\phi_{i,j}}. \quad (18)$$

Now, the values $\psi_{P_1}, \psi_{P_2}, \psi_{P_4}, \psi_{P_5}$ appear on the right hand side of (18) in a nonlinear way. But this system of nonlinear equations can be solved by viewing (18) as $\psi_{i,j} = F_2(\psi_{i,j})$, and by applying the same iterative technique as before (but with F_1 replaced by F_2).

2.5 Extensions to three dimensions

Extensions of the level-set method described in the previous sections to three dimensions is relatively straightforward. Since each term in the hyperbolic part is treated separately (i.e. approximations to ψ_x, ψ_y and ψ_z); only an additional term in the approximation to $|\vec{\nabla}\psi|$ will be needed. The parabolic terms in the level-set formulation in three dimensions can again be calculated using second-order central differences, just as in two dimensions. Using the signed distance function as initial conditions works in three dimensions as well. Reflecting boundary conditions are simply applied in three dimensions. Nonreflecting boundary conditions are also easily applied by using quadratic extrapolation in the interface normal, $\hat{n}_b$, direction. An example of a 3-D problem is given in Section 2.8. The same methodology of Section 2.6.2 can be applied in three dimensions to enforce arbitrarily complex boundaries.

2.6 Creating a burn table

For a $D_n - \kappa$ relation such that D_n is always greater than zero, any initial wave will cross a node only once. This fact follows from $\psi_t = -D_n(\kappa)|\vec{\nabla}\psi| \leq 0$, and is hence monotonically decreasing in time. Thus, instead of saving several $\psi(x, y, t)$ arrays in time, and taking contours at $\psi(x, y, t) = 0$, it is more efficient to create a burn table. A burn table is just a record of wave arrival times as a function of space, $t_b(x, y)$.

Creating a burn table is accomplished numerically by checking each node to see if its value of ψ changes sign after each timestep. If it has, then the $\psi(x, y, t) = 0$ contour has passed the node, and linear interpolation in time is used to record the burn time.

2.7 Numerical stability and accuracy

In this section, the stability restrictions placed on the numerical algorithms described in Section 2.4 are given. Accuracy of the algorithms is also examined by making comparisons with an exact solution. Since the numerical algorithms are explicit in time, certain restrictions on the time step are required to ensure numerical stability. As one might expect, the hyperbolic operator will have a Courant–Friedrichs–Lewy (CFL) type restriction, $c_1 \Delta t/\Delta x \leq 1$, while the "mostly parabolic" operator will have a restriction such as $c_2 \Delta t/\Delta x^2 \leq 1$.

Since (6) is nonlinear, classical methods for determining the stability of difference equations can not be used. The following approach is taken. First, the time step restriction for the first-order hyperbolic part of the operator will be obtained; this consists of (7) and (8), by requiring that the scheme be monotone. Then a time step restriction for the second-order parabolic part of the operator, consisting of (7) with central-differenced curvature terms, will be found by a frozen coefficient analysis. Then, the time step required for a general $D_n(\kappa)$ will be given.

Here, the time step restriction for the $D_n = D_{CJ}$ case, with first-order accurate differences (8) and no curvature dependence is given. The resulting PDE is hyperbolic, and has the property of being monotone. Monotonicity implies the following (see [18] for details): If two sets of initial data are given (say in two dimensions), $\psi_1(x, y, t = 0)$ and $\psi_2(x, y, t = 0)$, such that $\psi_2(x, y, t = 0) \geq \psi_1(x, y, t = 0)$ for all x and y, then for all time and space, $\psi_2(x, y, t) \geq \psi_1(x, y, t)$. A scheme which has this property is called a monotone method. Denote the solution of our difference equation as $\psi_{i,j}^{n+1} = \mathcal{H}(\psi_{i,j}^n)$,

where the function $\mathcal{H}$ is given from (7) and (8). Obviously, $\mathcal{H}$ will depend on Δx and Δt. To ensure that a numerical scheme is monotone, it is required that

$$\frac{\partial}{\partial \psi_{i,j}^n} \mathcal{H}(\psi_{k,l}^n) \geq 0$$

for all i, j, k, l.

Carrying out all the possible forms of $\mathcal{H}$ (which depend on the upwinding) gives the following CFL restriction on a uniform grid

$$\frac{2D_{CJ}\Delta t}{\Delta x} \leq 1 \tag{19}$$

in two dimensions, and

$$\frac{\sqrt{6}D_{CJ}\Delta t}{\Delta x} \leq 1 \tag{20}$$

in three dimensions. In general, monotone schemes are limited to first-order accuracy. But the first-order stability results can be used for the second-order scheme, since the second-order scheme reduces to first-order in nonsmooth regions (although the second-order scheme will not strictly be monotone).

Now, consider the problem of determining the time step restriction due only to the curvature-dependent terms, i.e. (7) and central differenced curvature terms. Also assume a linear dependence on the curvature. Then the level-set equation becomes:

$$\psi_t = \alpha \frac{\psi_{xx}\psi_y^2 - 2\psi_{xy}\psi_x\psi_y + \psi_{yy}\psi_x^2}{\psi_x^2 + \psi_y^2} \quad ,$$

where α is a positive constant. Notice the above can be rewritten as

$$\psi_t = \alpha(a^2\psi_{xx} - 2ab\psi_{xy} + b^2\psi_{yy}) \quad , \tag{21}$$

where $a^2 + b^2 = 1$. For the purposes of this discussion, assume that a and b are constants and carry out the standard von Neumann stability analysis on the resulting

linear operator. The time step restriction for the "linearized" curvature dependent term is

$$\frac{2\Delta t|\alpha|}{(\Delta x)^2} \leq 1$$

in two dimensions, and

$$\frac{4\Delta t|\alpha|}{(\Delta x)^2} \leq 1$$

in three dimensions. Thus, for the linear $D_n(\kappa) = D_{CJ} - \alpha\kappa$, the time step restriction is

$$\frac{2D_{CJ}\Delta t}{\Delta x} + \frac{2\Delta t|\alpha|}{(\Delta x)^2} \leq 1 \tag{22}$$

in two dimensions, and

$$\frac{\sqrt{6}D_{CJ}\Delta t}{\Delta x} + \frac{4\Delta t|\alpha|}{(\Delta x)^2} \leq 1 \tag{23}$$

in three dimensions. The above time step restriction can be used for a nonlinear $D_n(\kappa)$, by replacing the constant, $|\alpha|$, in (22) or (23), with $\max(|\partial\alpha/\partial\kappa|)$.

Next, it is demonstrated that (7) and (9) with central-differenced curvature dependent terms give second-order convergence. Although the truncation error of this scheme is $O(\Delta t) + O((\Delta x)^2)$, second-order convergence is expected, since for stability $\Delta t \propto (\Delta x)^2$ as $\Delta x \to 0$. Second-order convergence is demonstrated by comparison with an exact solution.

The example problem will be an expanding quarter circle, whose center is at the origin with an initial radius, $r = 0.2$. The numerical domain will be $0 \leq x \leq 1$ and $0 \leq y \leq 1$, with symmetry conditions at $x = 0$ and $y = 0$, and nonreflective conditions at $x = 1$ and $y = 1$. The relation $D_n = 1 - 0.1\kappa$ is chosen to test the second-order ENO–upwinding scheme. For error analysis purposes the error is measured on the discrete L_1 norm,

$$E_1 = \sum_{i,j} |t_b^{\text{exact}} - t_b^{\text{numerical}}| \Delta x \Delta y \quad .$$

The exact solution is obtained by noticing that the problem is really one-dimensional, with the velocity of the front being only a function of the radius ($\kappa = 1/r$), and by integrating the resulting ODE for the radius as a function of time. This gives $t_b^{\text{exact}}(x,y) = \sqrt{x^2+y^2} - 0.2 + 0.1\log[(0.1-\sqrt{x^2+y^2})/(0.1-0.2)]$ (with $\sqrt{x^2+y^2} \geq 0.2$) Table 1 shows the error, E_1, for several $\Delta x = \Delta y$s. The timestep was taken to be 0.8 of the maximum allowed by (22). Also shown is the calculated numerical order of accuracy, R_c. Notice that second-order convergence is achieved.

TABLE 1: Numerical accuracy for expanding circle with $D_n = 1 - 0.1\kappa$.

$\Delta x = \Delta y$	E_1	R_c
1/40	1.14×10^{-3}	
1/80	3.26×10^{-4}	1.81
1/160	8.70×10^{-5}	1.91
1/320	2.16×10^{-5}	2.01

2.8 3-D Seven-point detonation in PBX9502

A demonstration of the ability of a level-set formulation to handle 3-D multiple front interaction easily is given by the following example. Take the $D_n(\kappa)$ relation from Figure 3, in a cube with length 64 mm. Initially, there are seven spherically expanding detonations, six in a hexagonal pattern, and one in the center. See Figure 10. The spherical detonations merge, then intersect the edges of the cube, and eventually burn out of the domain. There is no special differencing needed when the fronts collide, since the level-set function itself remains continuous.

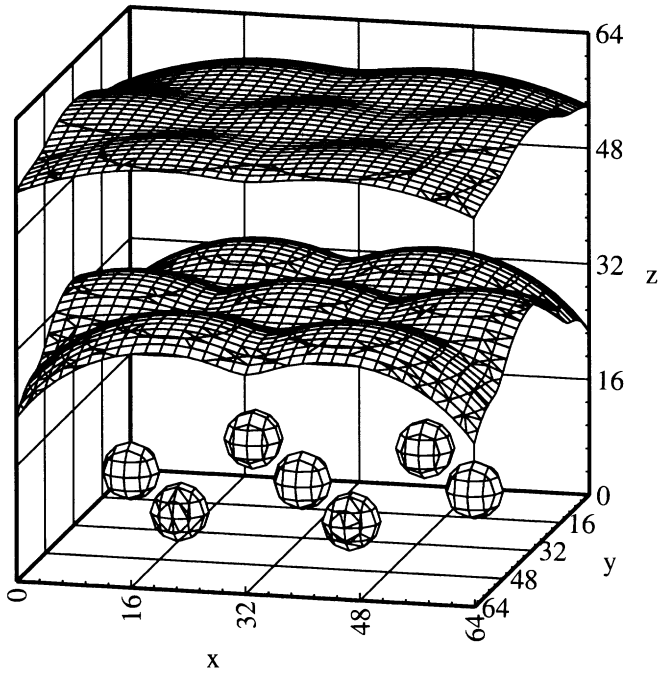

Figure 10: Seven-point detonation, shock front locations at 0, 3, 6 μs.

3 Dynamics of a $\dot{D}_n - D_n - \kappa$ relation

Here a formulation for the level-set method for evolving a front which obeys a $\dot{D}_n - D_n - \kappa$ relation is presented. This general formulation is then simplified for the case where the (1-D) front may be expressed as a function of $\psi(x, y, t) = y - y_s(x, t)$ for the stability analysis. It is shown that under appropriate conditions, this relation is of hyperbolic type. Also noted is that Whitham's equations for geometrical shock dynamics can be written as a $\dot{D}_n - D_n - \kappa$ relation, and can be solved using a level-set method described below. The DSD solutions using a $\dot{D}_n - D_n - \kappa$ relation will be given in Chapter 5, along with direct numerical simulations.

3.1 Level-set formulation

The equation for the level-set function, ψ, is basically unchanged from (4), except that the velocity, D_n, is now a variable. The level-set equation becomes

$$\frac{\partial \psi}{\partial t} + D_n |\vec{\nabla}\psi| = 0, \tag{24}$$

and the total derivative of D_n is given as

$$\frac{D(D_n)}{Dt} = \frac{\partial D_n}{\partial t} + D_n \hat{n} \cdot \vec{\nabla} D_n = \dot{D}_n(D_n, \kappa). \tag{25}$$

Using the definition of the normal yields

$$\frac{\partial D_n}{\partial t} + D_n \frac{\vec{\nabla}\psi}{|\vec{\nabla}\psi|} \cdot \vec{\nabla} D_n = \dot{D}_n(D_n, \kappa) \tag{26}$$

where the total curvature is given in (5). These equations form a coupled set of nonlinear PDEs for the evolution of the level-set function and its normal velocity. Next, the mathematical type of such a relation will be discussed, along with its linear stability.

3.2 Mathematical type and stability

A $\dot{D}_n - D_n - \kappa$ relation is a second-order PDE, and under certain conditions it is hyperbolic. Suppose a model, linear relation between the intrinsic quantities is given as follows:

$$\dot{D}_n = -\alpha\kappa + \beta(D_n - D_{CJ}). \tag{27}$$

Again, assuming the front may be written as

$$\psi(x, y, t) = y - y_s(x, t) = 0,$$

one can show that the PDE which describes the motion of the front will be

$$y_{s,t} - D_n(1 + y_{s,x}^2)^{1/2} = 0. \tag{28}$$

Now D_n is not a function of the curvature, κ, but rather the total derivative of D_n is given from (27):

$$D_{n,t} - \frac{y_{s,x}}{(1 + y_{s,x}^2)^{1/2}} D_n D_{n,x} = -\alpha\kappa + \beta(D_n - D_{CJ}). \tag{29}$$

Equations (28) and (29) are a coupled set of nonlinear PDEs for the evolution of the shock front and the shock front's normal velocity. Again, the stability and equation type are easily seen when the limit $y_{s,x} \to 0$ is taken. One can do the equation type analysis and stability for the general case, but there are no significant differences. Under the small shock approximation, (28) and (29) become

$$y_{s,t} - D_n = 0 \tag{30}$$

and

$$D_{n,t} = \alpha y_{s,xx} + \beta(D_n - D_{CJ}). \tag{31}$$

Substituting (30) into (31) gives a single second-order PDE

$$y_{s,tt} - \alpha y_{s,xx} = \beta(y_{s,t} - D_{CJ}), \tag{32}$$

which is the forced second-order linear wave equation (for $\alpha > 0$) with characteristic speeds $\pm\sqrt{\alpha}$. Standard linear analysis shows that (32) is stable for $\beta \leq 0$, and unstable otherwise. For a general $\dot{D}_n - D_n - \kappa$ relation, the relation may be linearized near a point in $\dot{D}_n - D_n - \kappa$ space and the local (frozen) stability results can be obtained.

3.3 Whitham's geometrical shock dynamics

It is interesting to note that other front theories outside of detonation dynamics obtains $\dot{D}_n - D_n - \kappa$ relations. For example Whitham's geometrical shock dynamics equations [19] may be interpreted as a $\dot{D}_n - D_n - \kappa$ relation (here D_n is the Mach number of the shock wave, and $\dot{D}_n$ is the shock acceleration in the normal direction). The relation is given by

$$\dot{D}_n = -\frac{(D_n^2 - 1)}{\lambda(D_n)}\kappa$$

where

$$\lambda(D_n) = \left(1 + \frac{2}{\gamma + 1}\frac{1 - \mu^2}{\mu}\right)\left(1 + 2\mu + \frac{1}{D_n^2}\right)$$

and

$$\mu^2 = \frac{(\gamma - 1)D_n^2 + 2}{2\gamma D_n^2 - (\gamma - 1)}.$$

For these equations, $\partial\dot{D}_n/\partial D_n$ changes sign at $\kappa = 0$, with $\kappa > 0$ being linearly stable and $\kappa < 0$ being linearly unstable (i.e. diverging shock waves are stable, while converging shocks are unstable). A contour plot of $\dot{D}_n$ is shown in Figure 11, for $\gamma = 1.4$.

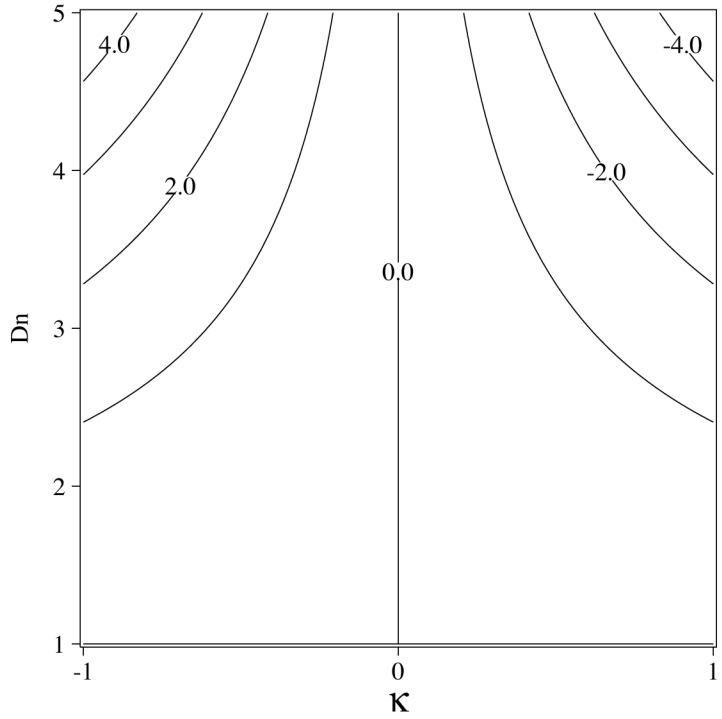

Figure 11: Contours of $\dot{D}_n$ for Whitham's $\dot{D}_n - D_n - \kappa$ relation with $\gamma = 1.4$.

3.4 Numerical implementation

Here, a numerical method for solving the level-set equations (24) and (26) is presented. As seen for the $D_n - \kappa$ relation, a level set method allows for very complex topological changes at the front, and coupled with an internal boundary method can solve most practical engineering problems.

But before the numerics are presented, note that any level set method may break down if one level curve passes over another, i.e. the level-set function has a shock form, with each level curve possessing a different value of D_n. The simplest example of such a problem is given as follows. Suppose that initially the level curve of interest is planar, for example at $t = 0$, $\psi(x, y, t = 0) = x$. This initial condition corresponds to a flat wave along the y-axis, with a normal vector pointing in the $\hat{\imath}$ direction. If $D_n(x, y, t = 0) = 1 - x$, and $\dot{D_n} = 0$, then it is expected that the all level curves will remain planar, and the level-set equations become

$$\psi_{,t} = -D_n(\psi_{,x}^2)^{1/2}$$

and

$$D_{n,t} + D_n D_{n,x} = 0.$$

Note that the equation for D_n is the inviscid Burgers equation, and that it is now decoupled from the level-set equation. Obviously, given the above initial conditions, the variables, ψ and D_n, will eventually form shocks in the normal direction, and the strong solution will become multivalued. It is not just a question of finding the proper weak solution to the above equations either, because the evolution of the level curve $\psi(x, y, t) = 0$ should not be affected by the evolution of the other level sets. A method to circumvent this problem, involving placing constraints on ψ and D_n away from $\psi = 0$, is given next.

From the above example, it is clear that if $D_n(x, y, t = 0) =$ constant, then there

would have been no shocks in the solution. Notice that shocks may form in the tangential direction of the surface, and the evolution may be well posed. Such an example would be Whitham's shock-shocks, where the velocity of the front jumps along with the angle of the front. Thus, the concern is with keeping discontinuities from forming in the normal direction.

Since the motion of the level curve $\psi(x, y, t) = 0$ is the only level curve of interest, one may alter the velocity, D_n, and level-set function, ψ, away from $\psi = 0$, and still not affect the motion of the level curve of interest ($\psi = 0$). In particular, if the level curves away from the zero level curve are kept from forming shocks, and the velocity, D_n, is kept from forming shocks in the normal direction of the fronts, then the evolution should remain well posed. A numerical method for keeping the solution from forming unwanted shocks is given next.

Constraints on the gradients of D_n and ψ are sufficient to keep the numerical problem well posed. Note that the gradients of ψ and D_n can be changed while keeping the ψ and D_n constant at $\psi = 0$. Next, two hyperbolic PDEs are described that will satisfy the above criteria when evolved to steady state. These are the redistance PDE of Sussman, Smereka and Osher [20],

$$\psi_{,\tau} = S(\psi)(1 - |\vec{\nabla}\psi|), \tag{33}$$

and a new PDE which keeps D_n from forming shocks in the normal direction,

$$D_{n,\tau} = -S(\psi)\frac{\vec{\nabla}\psi}{|\vec{\nabla}\psi|} \cdot \vec{\nabla} D_n, \tag{34}$$

where $S(\psi)$ is the sign function, and τ plays the role of time. It can be seen upon examination of the above PDEs, that they are hyperbolic, and that information travels outwards from the level curve $\psi = 0$. In characteristic form, the above equations become

$$\psi_{,\tau} + S(\psi)\hat{n} \cdot \vec{\nabla}\psi = S(\psi) \tag{35}$$

and

$$D_{n,\tau} + S(\psi)\hat{n} \cdot \vec{\nabla} D_n = 0 \tag{36}$$

where again

$$\hat{n} = \frac{\vec{\nabla}\psi}{|\vec{\nabla}\psi|}.$$

Thus, information propagates at speed 1 in the normal direction pointing away from the level curve, $\psi = 0$. Therefore, the state at $\psi = 0$ is unchanged by evolving the above PDEs. Notice that when the above PDEs reach steady state, the level-set function will be the distance function, and that the gradient of D_n along the normal direction will be zero. Thus, if any unwanted discontinuities start to form, the above PDEs will eliminate them. A simple one-dimensional example is given next.

Suppose the initial conditions for the reinitializing PDEs, (33) and (34), are $\psi(x, t = 0) = 2x$, and $D_n(x, t = 0) = 1 - x$. Since the problem is one-dimensional, solutions via the method of characteristics are easily obtained:

$$\psi = \begin{cases} x & |x| \geq \tau \\ 2x - \tau & |x| < \tau \end{cases}$$

and

$$D_n = \begin{cases} 1 & |x| \geq \tau \\ 1 - x + \tau & |x| < \tau. \end{cases}$$

Figure 12 shows this solution graphically at $\tau = 0$ and $\tau = 1$.

If it could be guaranteed that (33) and (34) were in equilibrium (i.e. steady state), then the level-set equations (24) and (26) would reduce to the following:

$$\psi_{,t} = -D_n \tag{37}$$

and

$$D_{n,t} = \dot{D}_n(D_n, \kappa). \tag{38}$$

Since the level curve zero is the only level curve of interest, the above reduced level-set equations would be valid near $\psi = 0$ after (33) and (34) are evolved for a short period

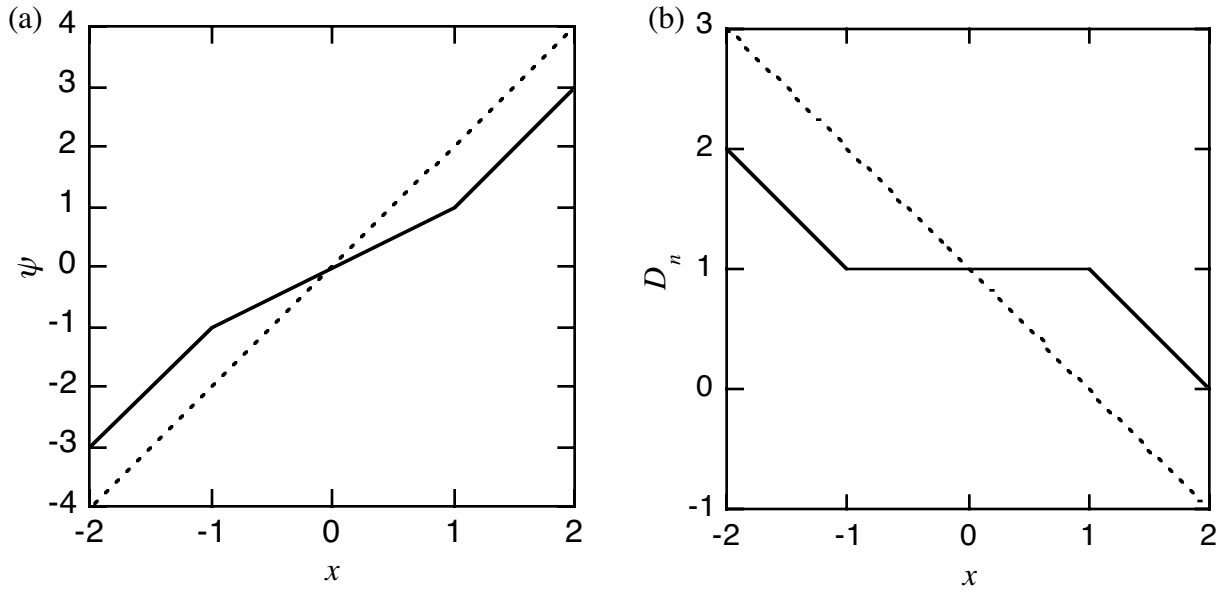

Figure 12: Solution to the reinitializing PDEs. (a) - - - $\psi(x, \tau = 0)$, —— $\psi(x, \tau = 1)$. (b) - - - $D_n(x, \tau = 0)$, —— $D_n(x, \tau = 1)$.

of time. Thus the following algorithm is proposed for solving $\dot{D}_n - D_n - \kappa$ relations using a level-set approach:

Step 1. Solve (33) and (34) to steady state near $\psi = 0$
at the beginning of every Runge–Kutta cycle.

Step 2. Update the variables, ψ and D_n, according to (37) and (38)
by a second-order Runge–Kutta method.

Step 3. Repeat *Step* 1 and *Step* 2 until final time is reached.

The discretization of the above PDEs is as follows.

First, the sign function is replaced by the "smooth" sign function, as in [20]:

$$S(\psi) = \frac{\psi}{\sqrt{\Delta x + \psi^2}}.$$

Notice that the above will be zero at the level curve zero. The $|\vec{\nabla}\psi|$ term in (33) is calculated using the second-order ENO–upwinding procedure, as in Section 2.4.

The time integration for this equation is simple forward Euler; since it will run to steady state, the time discretization errors are of no consequence.

Equation (34) also uses a forward Euler time integration procedure for the same reason as above. The normal, $\hat{n} = \vec{\nabla}\psi/|\vec{\nabla}\psi|$, is approximated by central-differences, and the spatial derivatives for D_n are given by second-order ENO–upwinding in each of the spatial directions.

The reduced level-set equations, (37) and (38), are integrated using the second-order Runge–Kutta algorithm as follows:

$$\psi^{(1)} = \psi^{(n)} - \Delta t D_n^{(n)}$$

$$D_n^{(1)} = D_n^{(n)} + \Delta t \dot{D}_n(D_n^{(n)}, \kappa^{(n)})$$

followed by

$$\psi^{(n+1)} = \frac{1}{2}\psi^{(n)} + \frac{1}{2}\psi^{(1)} - \frac{1}{2}\Delta t D_n^{(1)}$$

$$D_n^{(n+1)} = \frac{1}{2} D_n^{(n)} + \frac{1}{2} D_n^{(1)} + \frac{1}{2} \Delta t \dot{D}_n (D_n^{(1)}, \kappa^{(1)}),$$

where the superscript (n) denotes the old time level, superscript (1) denotes an intermediate state, and $(n+1)$ denotes the new time level.

Note, that in the beginning of every Runge–Kutta step, the equations (33) and (34) are evolved to steady state near $\psi = 0$. This procedure is accomplished by solving (33) and (34) for five "time steps", where $\Delta\tau = 0.283\Delta x$. This choice for $\Delta\tau$ can be shown to be monotone stable for the first-order upwind scheme. The analysis is the same as in Section 2.7. Since (33) and (34) propagate information at a speed of one, then after 5 "timesteps", the reinitializing PDEs will be in equilibrium for $|\psi| < 5\Delta\tau = 1.415\Delta x$, i.e. for $|\psi| < 1.415\Delta x$, ψ will be the distance function, and the directional derivative of D_n in the normal direction will be zero.

3.4.1 Internal boundary method

A simple, reflective internal boundary method has been used to treat arbitrarily complex two dimensional problems. It is a simplification of the previous, Section 2.6, implementation. If only the perfect reflection condition need be treated, then a very simple algorithm will give the appropriate state, ψ and D_n, at internal boundary nodes. See Section 2.6 for definitions, etc. First define a new nonevolving level-set function, ϕ, to be the signed distance function, equal to zero at the boundary, with $\phi > 0$ in the boundary region, and $\phi < 0$ in the flow region. Second, find all internal boundary nodes. These will be defined as points where $\phi > 0$, and at any of the eight surrounding nodes, $\phi \leq 0$. This procedure is exactly the same as in Section 2.6. Then find the location of the reflected point, P_r, which is located 2ϕ away from the internal boundary node, in the direction of $-\hat{n}_b$, where $\hat{n}_b = \vec{\nabla}\phi / |\vec{\nabla}\phi|$. See Figure 13. The variables, ψ and D_n, at the internal boundary node will be the state at the reflected point, P_r (for perfect reflection.) Again, the variables at P_r are given from

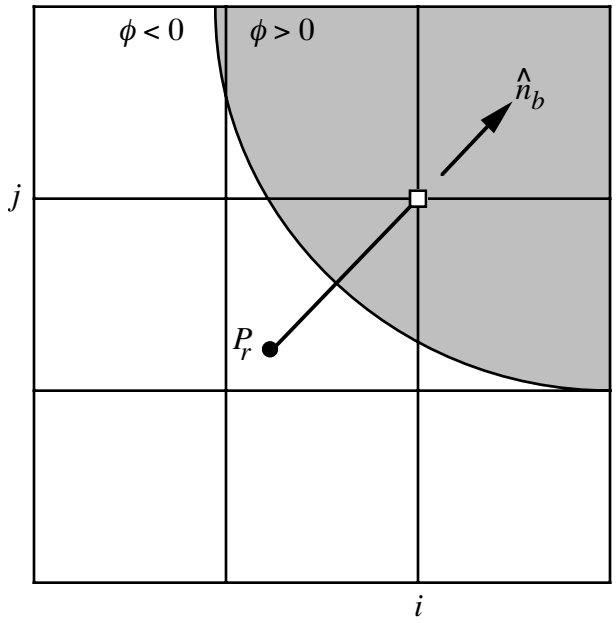

Figure 13: Schematic of internal boundary method grid for level-set solution of $\dot{D}_n - D_n - \kappa$ relations. • interpolated stencil point, □ internal boundary node (i, j).

second-order bilinear interpolation. Then, the iterative technique in Section 2.6 is used to solve for the internal boundary node state.

3.4.2 Shock–shock calculations

Here, some shock–shock problems using Whitham's $\dot{D}_n - D_n - \kappa$ relation are computed to see if the simple algorithm described above can reproduce the correct shock–shock angles described in [19]. Five tests are run with an incident Mach number of 100, converging on a perfectly reflecting wall, at an angle, θ, relative to the incident shock normal direction. The exact solution, in the strong shock limit for Whitham's equations, is a simple discontinuity in the velocity of the front, with a kink in the shock locus to satisfy the reflective boundary conditions. See Figure 14 for the details

of the geometry. The weak solution given by Whitham's Mach number area rule is shown as the solid line in Figure 14. The level-set solutions for the five different incident angles are also shown in the figure. The agreement is adequate, considering that the angles were calculated by picking off locations on the numerical grid. The error bars in Figure 14 represent the error in accuracy of the measurement from the numerical grid (about $\pm$ 1 degree). This seems encouraging, that the above simple discretization may give the proper jump conditions. But, it seems that when the entire problem is rotated on the grid, the shock–shock angles can be off by as much as 5 degrees. Similar anisotropic grid effects have been observed, and fixed, for level set methods with one-dimensional-based discretizations [21]. It is likely that these other "grid-free" discretizations would fix this problem.

3.4.3 Shock diffraction over a circular cylinder

Here, an example of the level-set solution to Whitham's geometrical shock dynamics is presented to demonstrate the level-set capability of dealing with topological changes and complex boundary problems. It was originally studied by Bryson and Gross [9] to test Whitham's theory. They used a characteristic based method which dealt with the shock–shock interactions separately.

The problem is diffraction of a planar shock by a circular cylinder. The shock has an incident Mach number of 2.81, and is traveling to the right in Figure 15. Once the incident shock hits the cylinder, the shock splits into two pieces, and a pair of shock–shocks are formed. Then each piece diffracts around the cylinder and join together again to form a second pair of shock–shocks. Qualitatively, the results are similar to Bryson and Gross, although there are small differences in the shock shape. Again, this may be due to anisotropic grid effects. Also, Bryson and Gross did not start to integrate the solution when the shock first hits the cylinder, but just after the first shock–shocks form. This may also cause some slight discrepancies.

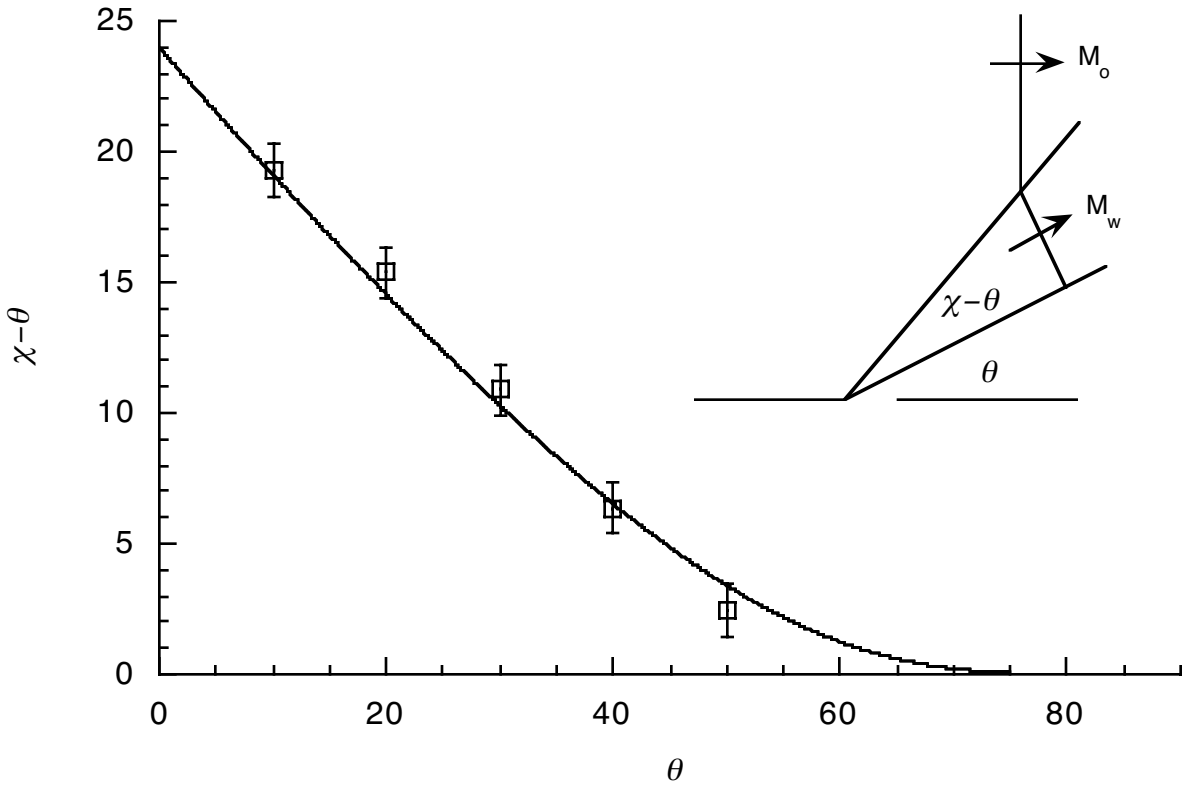

Figure 14: Comparison between the true weak solution angles of Whitham's GSD, with five measured level-set weak solution angles.

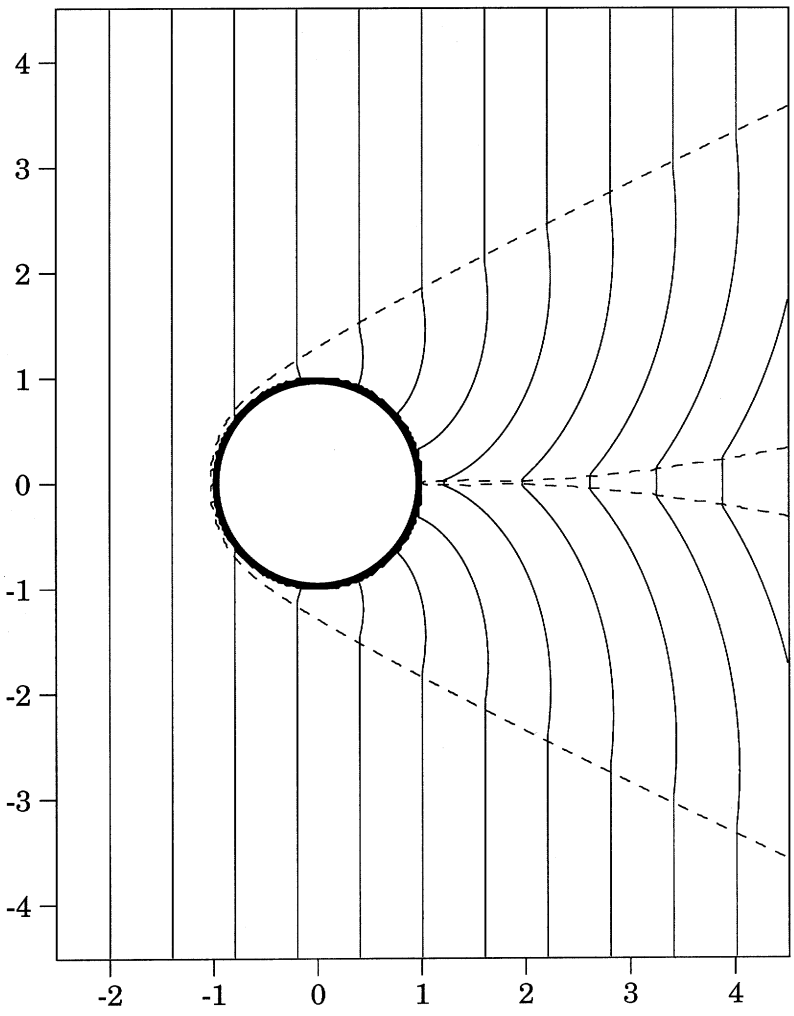

Figure 15: Level-set GSD solution to a planar, $M_o = 2.81$, shock diffracting over a circular cylinder. Shock travels from left to right. Solid lines represent fronts at various times; dashed lines are locations of shock–shocks.

4 Dynamics of a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation

Here, the dynamics of a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation are discussed. Recently, Stewart and Yao [4] derived a new detonation shock evolution equation that was based on the combined limits of near-Chapman–Jouguet (CJ) velocity, small shock curvature, slow temporal variation measured on the particle passage time through the reaction zone and large dimensionless activation energy. It is a relation between the second normal time derivative of normal detonation velocity, $\ddot{D}_n$, the first time derivative of the normal detonation velocity, $\dot{D}_n$, the normal detonation velocity, D_n, the time derivative of the total curvature, $\dot{\kappa}$, and the total curvature, κ.

A level-set formulation is presented, followed by a discussion of the mathematical type and linear stability. A numerical method for solving $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relations in a channel geometry is given. And a numerical example will demonstrate the cellular behaviour of the intrinsic relation.

4.1 Level-set formulation

Here a formulation for the level-set method for evolving a front which obeys a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation is presented. The equation for the level-set function, ψ, is exactly as in (24):

$$\frac{\partial \psi}{\partial t} + D_n |\vec{\nabla}\psi| = 0, \tag{39}$$

and again, total derivative of D_n is given as

$$\frac{D(D_n)}{Dt} = \frac{\partial D_n}{\partial t} + D_n \hat{n} \cdot \vec{\nabla} D_n = \dot{D}_n. \tag{40}$$

Using the definition of the normal yields

$$\frac{\partial D_n}{\partial t} + D_n \frac{\vec{\nabla}\psi}{|\vec{\nabla}\psi|} \cdot \vec{\nabla} D_n = \dot{D}_n. \tag{41}$$

The total derivative of acceleration, is given as

$$\frac{D(\dot{D}_n)}{Dt} = \frac{\partial \dot{D}_n}{\partial t} + D_n \hat{n} \cdot \vec{\nabla} \dot{D}_n = \ddot{D}_n(\dot{D}_n, D_n, \dot{\kappa}, \kappa). \tag{42}$$

Using the definition of the normal yields

$$\frac{\partial \dot{D}_n}{\partial t} + D_n \frac{\vec{\nabla}\psi}{|\vec{\nabla}\psi|} \cdot \vec{\nabla} \dot{D}_n = \ddot{D}_n(\dot{D}_n, D_n, \dot{\kappa}, \kappa), \tag{43}$$

where $\dot{\kappa}$ is the total derivative of the curvature:

$$\dot{\kappa} = \frac{\partial \kappa}{\partial t} + D_n \frac{\vec{\nabla}\psi}{|\vec{\nabla}\psi|} \cdot \vec{\nabla} \kappa. \tag{44}$$

Again, the above are for general problems. Note that the PDEs (39), (41), and (43) can be rewritten as a single third-order PDE. This will be useful for discussing the mathematical type and stability. These PDEs can be simplified for fronts which can be represented by $\psi(x, y, t) = y - y_s(x, t) = 0$. A numerical algorithm for such a simplification is presented in 4.3.

4.2 Mathematical type and stability

An evolution equation governed by a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation is a wave hierarchy in the sense discussed by Whitham [19]. To illustrate this, consider the perturbation of the detonation shock from CJ, steady, plane, ($D_n = 1$). Represent the shock surface as $\psi = y - t - \phi(x, t)$, where ϕ is small. The definitions of the normal velocity, curvature and normal surface derivatives imply, $D_n - 1 \sim \phi_{,t}$, $\dot{D}_n \sim \phi_{,tt}$, $\ddot{D}_n \sim \phi_{,ttt}$, $\kappa \sim -\phi_{,xx}$ and $\dot{\kappa} \sim -\phi_{,xxt}$. Substitutions show that a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation is a third-order PDE in time and second order in space. Our interest is in the choice of physical parameters such that the intrinsic relation is hyperbolic and admits transverse waves on the shock.

The $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation derived in [4] is:

$$(D_n - 1) + C_1 \theta \ddot{D}_n e^{-2\mu\theta(D_n - 1)} + [C_2\theta + C_3 + C_4\theta(D_n - 1)]\dot{D}_n e^{-\mu\theta(D_n - 1)}$$

$$+C_5(\theta\dot{D}_n)^2 e^{-2\mu\theta(D_n-1)} + C_6\frac{\mathcal{L}_{CJ}}{\theta} + (C_7\theta + C_8)e^{-2\mu\theta(D_n-1)}\dot{\kappa} = 0, \tag{45}$$

where $\mathcal{L}_{CJ}$ is

$$\mathcal{L}_{CJ} = \ln|1 + d\,\theta\,\kappa e^{-\mu\theta(D_n-1)}|, \tag{46}$$

The coefficients, μ, d and C_1 through C_8 are functions of only of γ and δ, where $\delta = 1/M_{CJ}^2$ and M_{CJ} is the Mach number of the plane CJ detonation. The expressions are derived and expressed in a sequential list in [4]. Figure 16 displays the relevant $C_1 - C_8$, μ and d versus δ for $\gamma = 1.6$, representative of a stoichiometric mixture of H_2, O_2, diluted with 70% Ar. The term θ is a scaled activation energy.

The highest-order wave operator comes from a pairing of $\ddot{D}_n$ and $\dot{\kappa}$, which is a time derivative of a second-order wave operator of the form $\partial/\partial t(\phi_{,tt} - a^2\phi_{,xx})$. The square of the hyperbolic wave speed is simply the ratio of the coefficient of $\dot{\kappa}$ and $\ddot{D}_n$,

$$a^2 = \frac{C_7\theta + C_8}{C_1\theta}. \tag{47}$$

Similarly, a subhyperbolic wave operator appears of the form $\phi_{,tt} - a_{sub}^2\phi_{,xx}$, from the pairing of $\dot{D}_n$ and κ and the square of the subhyperbolic wave speed is simply the ratio of the coefficient of κ and $\dot{D}_n$.

$$a_{sub}^2 = \frac{d\,C_6}{C_2\theta + C_3}. \tag{48}$$

Finally, $\phi_{,t}$ is the operator associated with $D_n - 1$, and has zero wave speed.

The conditions for a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation to be hyperbolic and subhyperbolic are respectively

$$C_7\theta + C_8 > 0 \text{ and } \; C_2\theta + C_3 > 0. \tag{49}$$

A stability criterion based on the wave speeds from [19] is

$$0 < a_{sub}^2 < a^2, \;\text{ stable, and }\; 0 < a^2 < a_{sub}^2, \;\text{ unstable.} \tag{50}$$

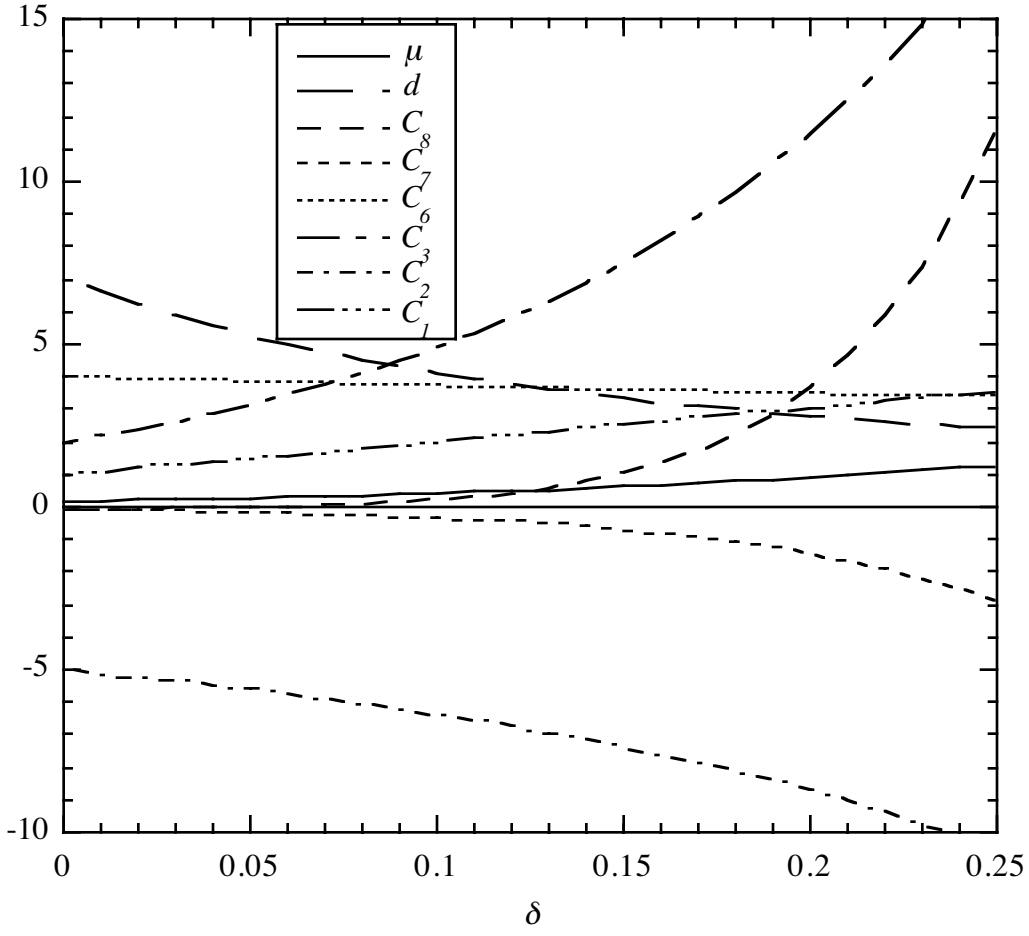

Figure 16: Constants $C_1, C_2, C_3, C_6, C_7, C_8, d$ and μ versus δ for $\gamma = 1.6$.

The $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation linearized about the plane, CJ detonation is

$$\phi_{,t} + (C_1\theta)\phi_{,ttt} + (C_2\theta + C_3)\phi_{,tt} - (d\,C_6)\phi_{,xx} - (C_7\theta + C_8)\phi_{,xxt} = 0. \tag{51}$$

The dispersion relation with $\phi = e^{\omega t + ikx}$ is

$$\omega + (C_1\theta)\omega^3 + (C_2\theta + C_3)\omega^2 + (d\,C_6)k^2 + (C_7\theta + C_8)k^2\omega = 0. \tag{52}$$

The neutrally stable frequency $\omega = i\omega_{Im}$, and wave number are

$$(\omega_{Im}^2)_{ns} = \frac{1}{C_1\theta}\left(\frac{a_{sub}^2}{a_{sub}^2 - a^2}\right), \quad k_{ns}^2 = \frac{1}{C_1\theta}\left(\frac{1}{a_{sub}^2 - a^2}\right), \tag{53}$$

with the corresponding wave length

$$\lambda_{ns} = \frac{2\pi}{k_{ns}} = 2\pi\sqrt{C_1\theta(a_{sub}^2 - a^2)}. \tag{54}$$

For the special one-dimensional case of $k = 0$, the neutral stability boundary is given by the condition $(C_2\theta + C_3) = 0$ which also corresponds to $a_{sub}^2 = \infty$.

The limiting form of the complex growth rate ω, for $k \to \infty$, as calculated from the dispersion relation, is to $O(1)$

$$\omega = \frac{dC_6}{2C_1\theta}\left(\frac{1}{a^2} - \frac{1}{a_{sub}^2}\right) \pm i\,ka^2 + \ldots \tag{55}$$

Consistent with this result, it can be shown for the unstable condition $a^2 < a_{sub}^2$, that there is a continuum of unstable wave numbers that in general runs from k_{ns} to $k = \infty$. Likewise for $a_{sub}^2 < a^2$, all k are stable.

The valid parameter range for the application of (45) is partially indicated by its dispersion relation. For hyperbolic evolution, one must have parameters, γ and δ such that $C_7\theta + C_8 > 0$. At the hyperbolic boundary where $C_7\theta + C_8 = a^2 = 0^+$, formula (55) shows that real growth rates for small disturbances are infinitely large. Indeed, crossing this parameter boundary corresponds to a change in type of (45), from hyperbolic to elliptic. The conclusion is, of course, that on the unphysical

(elliptic) side of the boundary the asymptotic description breaks down, and additional physics in the form of dynamic and curvature corrections, must be included.

The stability boundary $a^2 = a^2_{sub}$, has stable transverse disturbances on the side where a^2 is larger than a^2_{sub}, and unstable transverse disturbances where a^2_{sub} is larger than a^2. Standard considerations of unstable wave hierarchies show that on the unstable side, one expects the formation of discontinuous solutions, [19]; in this case, shock–shocks. Indeed this is what is found when numerical solutions of (45) are carried out (as shown in [4] and in the next section). The parametric region where self-sustained, well-posed cellular dynamics occurs is given by the three inequalities, $C_2\theta + C_3 > 0$, $C_7\theta + C_8 > 0$, and $a^2_{sub} > a^2$.

4.3 Numerical method for channel-type geometry

The numerical solutions are carried out for the shock location $y = y_s(x,t)$, where y is the direction along the channel axis and x is the distance across the channel. Equations (39),(41) and (43) are solved as a system of three first-order PDEs with, y_s, D_n and $\dot{D}_n$ as the dependent variables. The "channel" equations are given as the level-set equations, with $\psi(x,y,t) = y - y_s(x,t)$. These are:

$$y_{s,t} = D_n \left(1 + y^2_{s,x}\right)^{1/2} \tag{56}$$

$$D_{n,t} = \frac{y_{s,x}}{\left(1 + y^2_{s,x}\right)^{1/2}} D_n D_{n,x} + \dot{D}_n \tag{57}$$

$$\dot{D}_{n,t} = \frac{y_{s,x}}{\left(1 + y^2_{s,x}\right)^{1/2}} D_n \dot{D}_{n,x} + \ddot{D}_n(\dot{D}_n, D_n, \dot{\kappa}, \kappa) \tag{58}$$

where the front normal is given as

$$\hat{n} = \frac{y_{s,x}}{\left(1 + y^2_{s,x}\right)^{1/2}} \hat{\imath} + \frac{1}{\left(1 + y^2_{s,x}\right)^{1/2}} \hat{\jmath}$$

and the curvature, and the normal time rate of change of the curvature are

$$\kappa = \vec{\nabla} \cdot \hat{n} = -\frac{y_{s,xx}}{\left(1 + y_{s,x}^2\right)^{3/2}}$$

$$\dot{\kappa} = \kappa_{,t} - \frac{y_{s,x}}{\left(1 + y_{s,x}^2\right)^{1/2}} D_n \kappa_{,x} \quad .$$

The time integration uses a third-order Runge–Kutta algorithm, described in detail in Section 5.2. All that is left then, is to evaluate the spatial derivatives and forcing on the right hand side. The advective spatial derivatives in (56), (57), (58), and in $\dot{\kappa}$ that appear are treated using second-order ENO–upwinding, as in Chapter 2 and 3. The curvature is calculated using second-order central-differences, and the forcing terms are just an evaluation.

4.4 Evolution of cellular dynamics

To visualize the solutions predicted by the $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation, one imagines a smooth shock segment, with a reaction zone starting at the shock at $n = 0$, and ending at the sonic locus at n_{CJ}. The locus n_{CJ} identifies the location of the fire, or thin zone of main reaction and the width of the induction zone (IZ) between the shock and fire. The leading order formula for n_{CJ}, from [4] is

$$n_{CJ} = -\frac{(\gamma - 1 + 2\delta)}{\gamma + 1} \left(\frac{c_s^4}{\alpha}\right) e^{-\mu\theta(D_n - 1)}, \tag{59}$$

where $c_s^4/\alpha = \{2(\gamma - 1 + 2\delta)^2[2\gamma + \delta(\gamma - 1)]^2\}/\{(1 - \delta)[\gamma(3 - \gamma) - \delta(3\gamma - 1)](\gamma + 1)^2\}$. Thus two fronts, one for the shock locus, and one for the sonic locus are determined by solving (45) and subsequently evaluating (59). The theory predicts the smooth and parametric variation of the states between the shock and sonic locus.

Here, a description of the character of the numerical solution to (45) for values of the parameters in the cell region of the $E - Q$ plane is given. The numerical solutions were generated for a simplified version of (45), where the terms $(D_n - 1)\dot{D}_n$

and $(\theta \dot{D}_n)^2$ where dropped, while the nonlinearity of $\mathcal{L}_{CJ}$ was retained, so that the following is solved

$$(D_n - 1) + C_1 \theta \ddot{D}_n e^{-2\mu\theta(D_n - 1)} + [C_2\theta + C_3]\dot{D}_n e^{-\mu\theta(D_n-1)} +$$

$$C_6 \frac{\mathcal{L}_{CJ}}{\theta} + (C_7\theta + C_8) e^{-2\mu\theta(D_n-1)} \dot{\kappa} = 0. \tag{60}$$

Figures 17 and 18 show an example of the solution to (60) in terms of a gray-scale contour plot of the value of D_n that is attained at each fixed point in the channel when the shock crosses that point, for parameters $Q = 2.5, E = 5$ and $\gamma = 1.6$. This contour plot effectively produces a simulated smoke foil. Two segments of the channel are shown. The shock travels from left to right, and starts from the top. At time $t = 0$ the shock is assumed to be flat, with an initial, sinusoidal velocity disturbance $D_n = 1 + 0.3\cos(0.96\pi x) + 0.02\sin(\pi x/25)$ and $\dot{D}_n = 0$. Initially the detonation shock appears quite flat, but the strength of the small transverse waves grows. As the detonation propagates the cells merge and grow in size and strength and appear to saturate.

Figure 18 shows an enlargement of the box in Figure 17, with instantaneous contours of the shock locus (solid line) and sonic locus (broken line). In regions where shock velocity is above CJ (overdriven) the fire is close to the shock, while when the shock velocity is below CJ (underdriven), the fire is relatively farther away from the shock, in accordance with observations by the experimentalists carried out with smoke foils. The variation in the normal detonation velocity as computed in the numerical solution, is between approximately 0.7 and 1.4, which is in both qualitative and quantitative agreement for that reported in the experimental accounts [22], [23].

Through the course of many simulations of (45), it is found that the neutral stability wavelength, λ_{ns}, is a good predictor of the final stable cell widths observed in the simulations, if the channel walls are wide enough. This yields a criterion for

determining cell spacing as $\lambda_{cell} = \lambda_{ns} = 2\pi/k_{ns} = 2\pi\sqrt{C_1\theta(a_{sub}^2 - a^2)}$. Figure 19 shows contours of constant λ_{ns} according to (54) in the $E - Q$ parameter plane, for $\gamma = 1.6$. The cell aspect ratio W/L can be predicted by the reciprocal of the transverse wave speed a, i.e. $W/L = 1/a$.

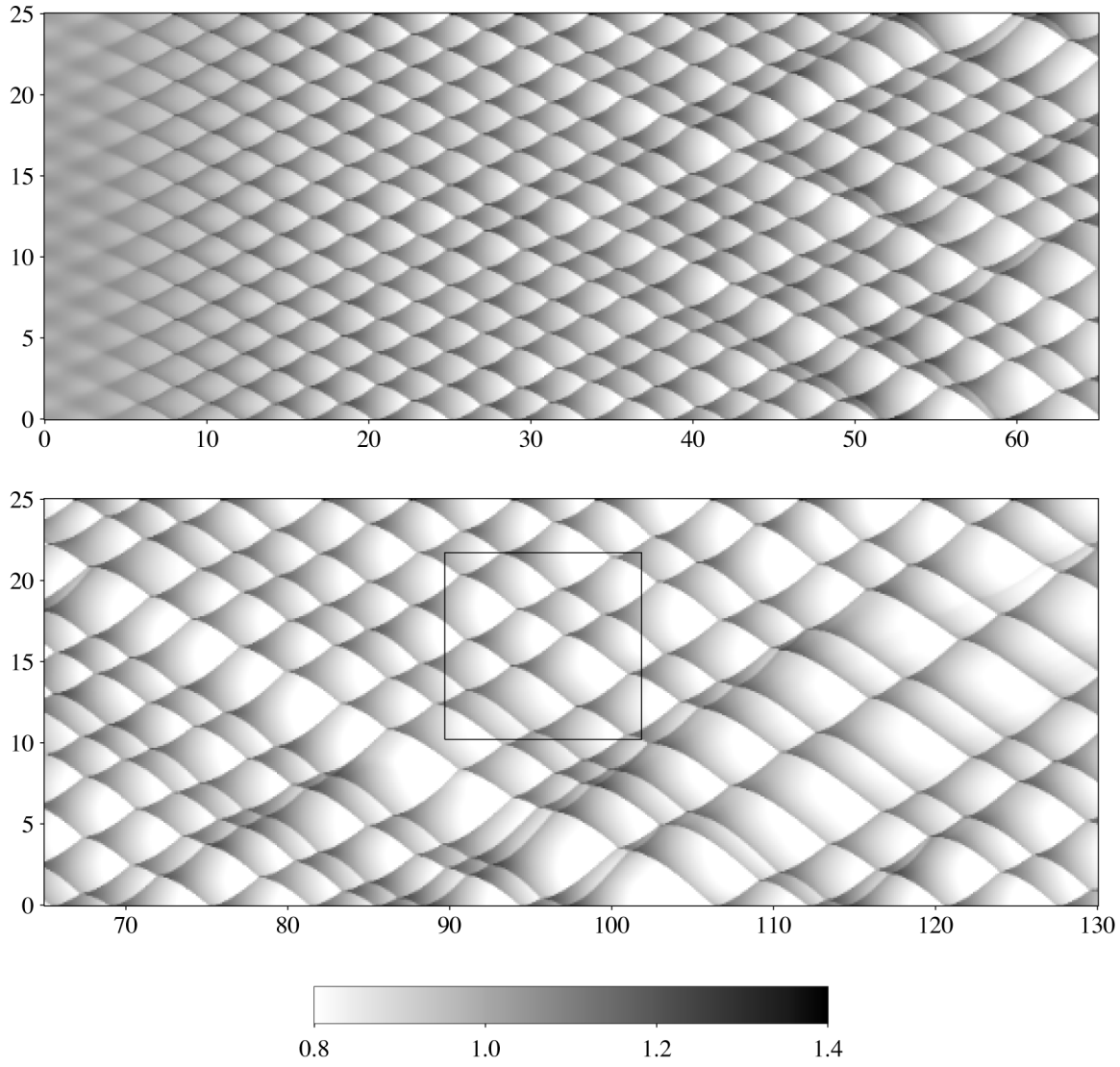

Figure 17: Gray-scale plot of D_n versus fixed locations in space. The boxed region is shown in more detail in Figure 18.

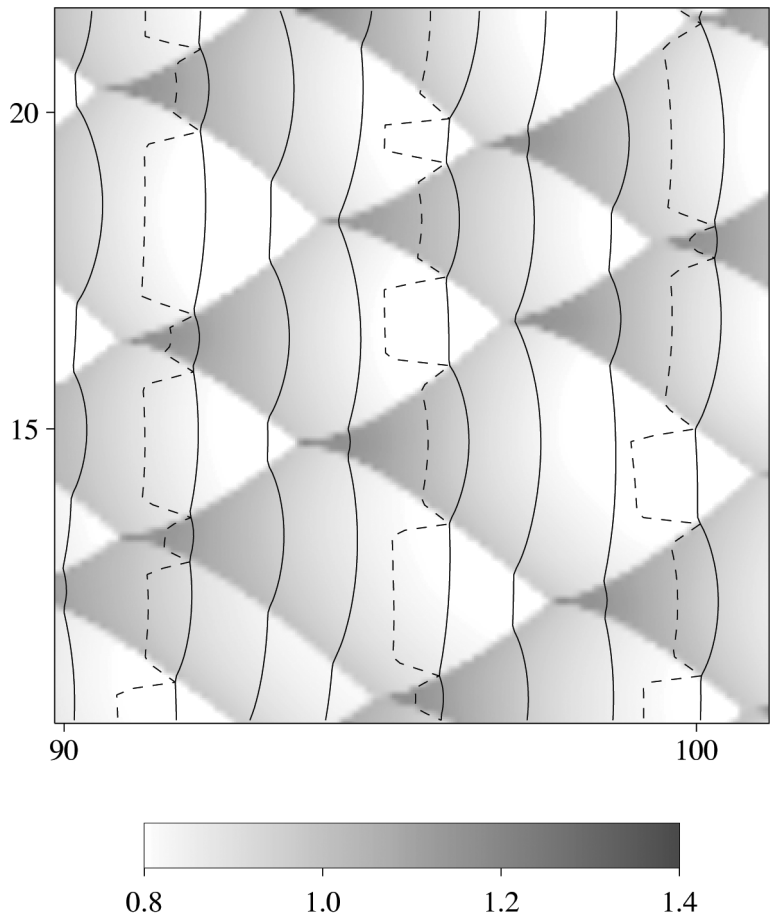

Figure 18: Gray-scale plot of D_n with instantaneous shock fronts given as solid lines. The location of the fire for three times are given as dashed lines.

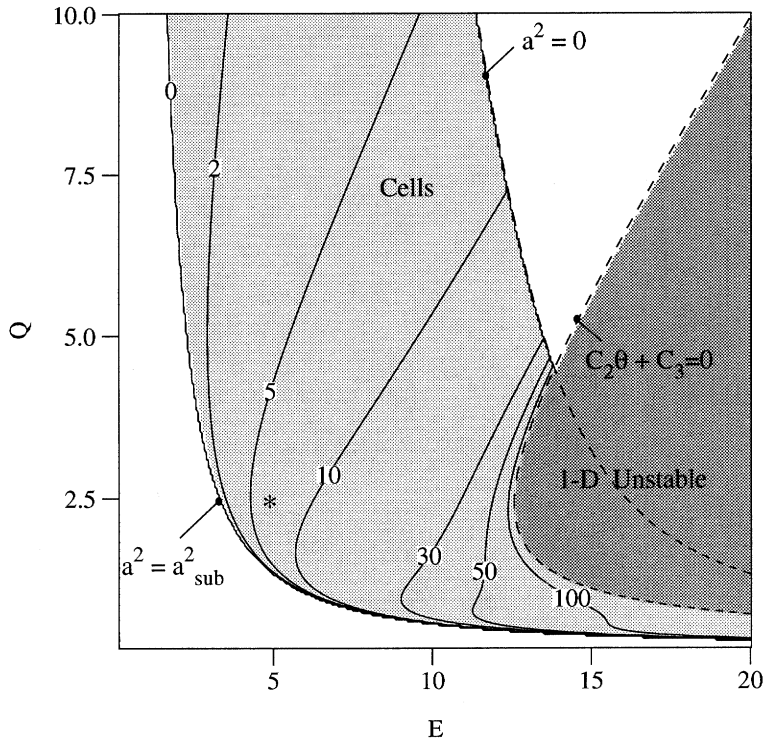

Figure 19: Regions of cellular dynamics and 1-D unstable dynamics are shown as light and dark gray shades respectively for $\gamma = 1.6$. Solid numbered lines are contours of constant λ_{ns}. The (*) represents the location in parameter space that corresponds with Figures 17 and 18.

5 Comparison of DNS and level-set solution of DSD

Here, comparisons between DSD theory with direct numerical simulation (DNS) of detonations are made. The direct numerical simulations were carried out with a code described in Section 5.2. The code is based on a high-order Godunov-type shock-capturing scheme. Of particular interest is the location and subsequent dynamics of the detonation front. In Section 5.1, the mathematical formulation of the detonation model used in the DNS is presented.

Since DSD theory is an approximate theory, one would like to know how well the theory predicts shock front evolution. One way of accomplishing this goal is to compare a DSD solution to an exact solution of a multi-dimensional detonation problem. Unfortunately, this is not possible. The best one can do is to have a resolved numerical simulation of the reactive, compressible Euler equations. Then comparisons of the dynamics of the shock front from DSD theory to the resolved DNS can be made. Next, an algorithm for solving the compressible reactive Euler equations will be presented, along with comparisons with DSD theory.

5.1 Reactive Euler equations

The reactive Euler equations are given by the conservation of mass, momentum, and energy and a reaction rate law as follows:

$$\begin{aligned}
\frac{D\rho}{Dt} + \rho\vec{\nabla}\cdot\vec{u} = 0 \quad , \\
\rho\frac{D\vec{u}}{Dt} + \vec{\nabla}p = 0 \quad , \\
\frac{De}{Dt} + p\frac{D(1/\rho)}{Dt} = 0 \quad , \\
\frac{D\lambda}{Dt} = r(p,\rho,\lambda) \quad ,
\end{aligned} \tag{61}$$

with the ideal equation of state

$$e = \frac{p}{\rho(\gamma - 1)} - Q\lambda \quad ,$$

where Q is the heat of detonation, λ is the reaction progress variable ($\lambda = 0$ for unreacted material, and $\lambda = 1$ for completely reacted material), and r is the reaction rate.

Written in conservative form, in 2-D Cartesian coordinates, these become:

$$(\rho)_t + (\rho u)_x + (\rho v)_y = 0 \quad ,$$

$$(\rho u)_t + (\rho u^2 + p)_x + (\rho uv)_y = 0 \quad ,$$

$$(\rho v)_t + (\rho uv)_x + (\rho v^2 + p)_y = 0 \quad ,$$

$$(E)_t + (uE + up)_x + (vE + vP)_y = 0 \quad ,$$

$$(\rho\lambda)_t + (\rho u\lambda)_x + (\rho v\lambda)_y = \rho r(p, \rho, \lambda) \tag{62}$$

where

$$E = \rho e + \frac{\rho}{2}(u^2 + v^2)$$

is the total energy. Next, a numerical method will be presented to solve the above conservation equations.

5.2 Numerical methods for simulation of the reactive Euler equations

The algorithm for numerically solving the reactive Euler equations will be based on Shu and Osher's semi-discrete (method of lines) ENO scheme [24]. The purpose of picking this algorithm is two-fold. First, by formulating the problem in a semi-discrete manner, spatial and temporal discretization are accomplished independent of one another. This makes the code easy to write for multi-dimensional forced problems.

The second reason is that by using high-order spatial and temporal discretization, very accurate solutions are obtainable (formally at least in continuous regions of the flow).

The method of lines, in effect, splits the temporal differencing from the spatial differenecing. Thus each may be described separately as follows. View (62) in the following vector form:

$$\boldsymbol{u}_t = \boldsymbol{\mathcal{L}}(\boldsymbol{u}) \tag{63}$$

where the vector $\boldsymbol{u}$ represents the vector of conservative variables in (62), and $\boldsymbol{\mathcal{L}}(\boldsymbol{u})$ is the spatial derivative operators and source term. A numerical approximation to $\boldsymbol{\mathcal{L}}(\boldsymbol{u})$ will be denoted as $\boldsymbol{L}(\boldsymbol{u})+O(\Delta x^m)$, where m is the spatial order of the accuracy. Once the initial conditions are given, then the time discretization along with the spatial discretization define the algorithm completely.

First, a third-order Runge–Kutta time discretization will be given, one that advances $\boldsymbol{u}^{(n)}$ to $\boldsymbol{u}^{(n+1)}$, where n represents the time level of the solution. The third-order time integration can be written as the following:

$$\begin{aligned}
\boldsymbol{u}^{(1)} &= \boldsymbol{u}^{(n)} + \Delta t \boldsymbol{L}(\boldsymbol{u}^{(n)}) \\
\boldsymbol{u}^{(2)} &= \frac{3}{4}\boldsymbol{u}^{(n)} + \frac{1}{4}\boldsymbol{u}^{(1)} + \frac{1}{4}\Delta t \boldsymbol{L}(\boldsymbol{u}^{(1)}) \\
\boldsymbol{u}^{(n+1)} &= \frac{1}{3}\boldsymbol{u}^{(n)} + \frac{2}{3}\boldsymbol{u}^{(2)} + \frac{2}{3}\Delta t \boldsymbol{L}(\boldsymbol{u}^{(2)})
\end{aligned} \tag{64}$$

Other orders of time integration from first-order to fifth-order have been given in [12]. The above third-order scheme was the one used in the following computations. Now only the operator $\boldsymbol{L}(\boldsymbol{u})$ need be given to advance the solution according to (64).

The operator $\boldsymbol{L}(\boldsymbol{u})$ consists of the $x-$flux, $y-$flux and the source term in the following way:

$$\boldsymbol{L}(\boldsymbol{u}) = -\boldsymbol{f}_x - \boldsymbol{g}_y + \boldsymbol{s}(\boldsymbol{u})$$

For the scheme to be conservative, the discrete quantities $(\boldsymbol{f}_{i,j})_x$ and $(\boldsymbol{g}_{i,j})_y$ defined at node locations (i, j) must be of the form

$$(\boldsymbol{f}_{i,j})_x = (\boldsymbol{f}_{i+1/2,j} - \boldsymbol{f}_{i-1/2,j})/\Delta x$$

and

$$(\boldsymbol{g}_{i,j})_x = (\boldsymbol{g}_{i,j+1/2} - \boldsymbol{g}_{i,j-1/2})/\Delta y.$$

The source term, $\boldsymbol{s}(\boldsymbol{u})$, in this formulation is simply an evaluation at the node point (i, j). Next, an algorithm for computing the intermediate flux function, $\boldsymbol{f}_{i+1/2,j}$, is presented. The others are computed in an analogous fashion.

The method of calculating $\boldsymbol{f}_{i+1/2,j}$ consists of the following steps:

Step 1: Define an averaged state between nodes (i, j) and $(i + 1, j)$,
by Roe's method.

Step 2: Calculate eigenvalues, $\boldsymbol{\lambda}$, right eigenvectors, $\boldsymbol{r}^{(p)}$ and
left eigenvectors, $\boldsymbol{l}^{(p)}$ of the averaged Jacobian matrix,
here p represents each characteristic field $(p = 1, 2, ..., 5)$.

Step 3: Decompose the flux vector, $\boldsymbol{f}$, into characteristic
field fluxes, $\boldsymbol{f}_c$, through the use of the left eigenvectors.

Step 4: Interpolate each characteristic flux, $f_c^{(p)}$, via ENO, and upwinding
in the appropriate direction given by the sign of the
eigenvalue for that characteristic flux field.

Step 5: Map the characteristic fluxes back to the primitive flux
function via the right eigenvectors.

The details of this algorithm are as follows.

The algorithm first needs to define an averaged state, to define an averaged Jacobian matrix. Roe's method reduces to the following averages in the x-direction:

$$\bar{\rho}_{i+1/2,j} = \sqrt{\rho_{i,j}\rho_{i+1,j}}$$

$$\bar{u}_{i+1/2,j} = \frac{u_{i,j}\sqrt{\rho_{i,j}} + u_{i+1,j}\sqrt{\rho_{i+1,j}}}{\sqrt{\rho_{i,j}\rho_{i+1,j}}}$$
$$\bar{v}_{i+1/2,j} = \frac{v_{i,j}\sqrt{\rho_{i,j}} + v_{i+1,j}\sqrt{\rho_{i+1,j}}}{\sqrt{\rho_{i,j}\rho_{i+1,j}}}$$
$$\lambda_{i+1/2,j} = \frac{\lambda_{i,j}\sqrt{\rho_{i,j}} + \lambda_{i+1,j}\sqrt{\rho_{i+1,j}}}{\sqrt{\rho_{i,j}\rho_{i+1,j}}}$$
$$\bar{H}_{i+1/2,j} = \frac{H_{i,j}\sqrt{\rho_{i,j}} + H_{i+1,j}\sqrt{\rho_{i+1,j}}}{\sqrt{\rho_{i,j}\rho_{i+1,j}}}$$

where

$$H = \frac{E+p}{\rho}$$

is the enthalpy. Any other component of (62) can be computed as a function of the above averages.

Next, the Jacobian matrix for (62) is evaluated at the averaged state:

$$\boldsymbol{J} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \frac{\beta}{2}\bar{q}^2 - \bar{u}^2 & (3-\gamma)\bar{u} & -\bar{v}\beta & \beta & Q\beta \\ -\bar{u}\bar{v} & \bar{v} & \bar{u} & 0 & 0 \\ -\beta\bar{u}(\bar{H} - \frac{\bar{q}^2}{2}) & \bar{H} - \beta\bar{u}^2 & -\bar{u}\bar{v}\beta & \gamma\bar{u} & \beta Q\bar{u} \\ -\bar{u}\bar{\lambda} & \bar{\lambda} & 0 & 0 & \bar{u} \end{bmatrix},$$

where $\beta = \gamma - 1$, and $\bar{q}^2 = \bar{u}^2 + \bar{v}^2$.

From this matrix, the eigenvalues, right eigenvectors and left eigenvectors can be computed:

$$\boldsymbol{\lambda} = \begin{pmatrix} \bar{u} - \bar{c} \\ \bar{u} \\ \bar{u} \\ \bar{u} \\ \bar{u} + \bar{c} \end{pmatrix}$$

are the eigenvalues and the right eigenvectors are the columns of

$$\boldsymbol{R} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \bar{u} - \bar{a} & \bar{u} & \bar{u} & \bar{u} & \bar{u} + \bar{a} \\ \bar{v} & \bar{v} - \bar{a} & \bar{v} & \bar{v} + \bar{a} & \bar{v} \\ \bar{H} - \bar{u}\bar{a} & \frac{\bar{q}^2}{2} - \bar{v}\bar{a} - Q & \frac{\bar{q}^2}{2} & \frac{\bar{q}^2}{2} + \bar{v}\bar{a} - Q & \bar{H} + \bar{u}\bar{a} \\ \bar{\lambda} & 1 & 0 & 1 & \bar{\lambda} \end{bmatrix}$$

i.e.

$$\boldsymbol{R} = [\, \boldsymbol{r}^{(1)} \quad \boldsymbol{r}^{(2)} \quad \boldsymbol{r}^{(3)} \quad \boldsymbol{r}^{(4)} \quad \boldsymbol{r}^{(5)} \,]$$

and the left eigenvectors are the rows of

$$\boldsymbol{L} = \frac{1}{2\bar{a}^2} \begin{bmatrix} \bar{u}\bar{a} + \beta\bar{q}^2 & -(\bar{a} + \beta\bar{u}) & -\beta\bar{v} & \beta & \beta Q \\ \bar{v}\bar{a} - \beta\frac{\bar{q}^2}{2}\bar{\lambda} & \beta\bar{u}\bar{\lambda} & -a + \beta\bar{v}\bar{\lambda} & -\beta\bar{\lambda} & \bar{a}^2 - \beta Q\bar{\lambda} \\ 2a^2 - \beta(1-\lambda)q^2 & 2\beta(1-\lambda)u & 2\beta(1-\lambda)v & -2\beta(1-\lambda) & -2\beta(1-\bar{\lambda})Q - 2\bar{a}^2 \\ -\bar{v}\bar{a} - \beta\frac{\bar{q}^2}{2}\bar{\lambda} & \beta\bar{u}\bar{\lambda} & \bar{a} + \beta\bar{v}\bar{\lambda} & -\beta\bar{\lambda} & a^2 - \beta\bar{\lambda}Q \\ -\bar{u}\bar{a} + \beta\frac{\bar{q}^2}{2}\bar{\lambda} & \bar{a} - \beta\bar{u} & -\beta\bar{v} & \beta & \beta Q \end{bmatrix}$$

i.e.

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{l}^{(1)} \\ \boldsymbol{l}^{(2)} \\ \boldsymbol{l}^{(3)} \\ \boldsymbol{l}^{(4)} \\ \boldsymbol{l}^{(5)} \end{bmatrix}.$$

A characteristic flux will be given as $f_c^{(p)} = \boldsymbol{l}^{(p)} \cdot \boldsymbol{f}$. Now, for each characteristic flux field there is an associated eigenvalue. If the eigenvalue is negative, then the wave is coming from the right, and a first-order scheme would approximate the characteristic flux as the left eigenvector dotted with the flux vector, evaluated at the $(i+1, j)$ node. If the eigenvalue for a characteristic flux is positive then the characteristic flux is evaluated at the (i, j) node. To achieve high-order accuracy, these characteristic flux functions need to be interpolated to the $(i+1/2, j)$ location. To make this more concrete, take an example where the eigenvalue for the p^{th} field is positive, then the first-order characteristic flux would be

$$f_c^{(p)}(i+1/2, j) = \boldsymbol{l}^{(p)} \cdot \boldsymbol{f}(i, j).$$

To achieve m^{th} order accuracy (for a positive eigenvalue), calculate the characteristic fluxes at each of the surrounding $m-1$ points (i.e. for third-order, calculate $f_c^{(p)}(h, j)$, where $h = i-2, i-1, i, i+1, i+2$. Then (for third-order accuracy) center-biased ENO interpolation is used on these five characteristic fluxes to obtain a high-order

approximation to the characteristic flux function at $(i+1/2, j)$. In essence, the third-order center-biased ENO algorithm chooses a three point stencil, which contains the node (i, j) and two others. Then a quadratic interpolant is used to evaluate the interpolated function at $(i + 1/2, j)$. It was pointed out in [25] and [26] that it is advantageous for accuracy and stability to make the scheme center-biased. This is accomplished by the following algorithm:

Step 1: Let $k = 0$.

Step 2: if $\mid f_c^{(p)}(i, j) - f_c^{(p)}(i-1, j) \mid < 2 \mid f_c^{(p)}(i+1, j) - f_c^{(p)}(i, j) \mid$ then $k = k - 1$.

Step 3: if $2 \mid f_c^{(p)}(k+i+1, j) - 2f_c^{(p)}(k+i, j) + f_c^{(p)}(k+i-1, j) \mid <$
$\mid f_c^{(p)}(k+i+2, j) - 2f_c^{(p)}(k+i+1, j) + f_c^{(p)}(k+i, j) \mid$ then $k = k - 1$.

Here, k represents the left most point of the stencil relative the node (i, j). Effectively, *Step* 2 finds the second point of the stencil, either $(i+1, j)$ or $(i-1, j)$, whichever has a smaller (in magnitude) first derivative (but it biases to the left due to the 2 on the right hand side of the inequality). And then, *Step* 3 adds the third point of the stencil, either the point to the left of the previous stencil or the point to the right, depending on which has a smaller (in magnitude) second derivative (now it biases to the right due to the factor of 2 on the left hand side of the inequality). The standard ENO scheme [12] is the above algorithm without the factors of 2 in front of the magnitude of the derivatives. In any event, once a stencil is chosen, a conservative quadratic fit is made through the resulting three nodes. This is equivalent to keeping that the integral of this fitted quadratic equation divided by Δx in each of the "cells" must give the "cell" average value back. The word "cell" is in quotes because the code really only uses point values, but it was shown in [12] that the interpolation procedure is the same. Again, all that is needed is the characteristic flux at the $(i + 1/2, j)$ location. This value is given by

if $k = -2$: then

$$f_c^{(p)}(i+1/2,j) = \tfrac{1}{3}f_c^{(p)}(i-2,j) - \tfrac{7}{6}f_c^{(p)}(i-1,j) + \tfrac{11}{6}f_c^{(p)}(i,j)$$

if $k = -1$: then

$$f_c^{(p)}(i+1/2,j) = -\tfrac{1}{6}f_c^{(p)}(i-1,j) + \tfrac{5}{6}f_c^{(p)}(i,j) + \tfrac{1}{3}f_c^{(p)}(i+1,j)$$

if $k = 0$: then

$$f_c^{(p)}(i+1/2,j) = \tfrac{1}{3}f_c^{(p)}(i,j) - \tfrac{5}{6}f_c^{(p)}(i+1,j) - \tfrac{1}{6}f_c^{(p)}(i+2,j).$$

Note, for characteristic fluxes which have a negative eigenvalue, one can use the above interpolation procedure by shifting the first point in the stencil to $(i+1,j)$ (i.e. the upwind point), and changing the $(i-1,j)$s to $(i+1,j)$s, etc.

Once all the characteristic fluxes have been computed at $(i+1/2,j)$, then the primitive flux functions may be recovered by multiplying by the rows of the $\boldsymbol{R}$ matrix, i.e. $f^{(1)}(i+1/2,j)$ would be the first row of $\boldsymbol{R}$ dotted with the characteristic flux vector, $\boldsymbol{f}_c$, etc. Likewise, the other intermediate flux functions may be calculated in a similar fashion.

This algorithm works well for most regions of the flow, but it may break down in regions where the eigenvalue changes sign between two nodes. Then that characteristic flux is calculated using a local-Lax–Friedrichs approach. This approach, instead of upwinding (which doesn't make much sense when the eigenvalues are of different sign), breaks the characteristic flux function into two parts, a "positive" flux function, f^+, and a "negative" flux function, f^-. Here, it is implied that characteristic fluxes are still being calculated. The definitions of these are:

$$f^+ = \frac{1}{2}(f + \alpha v)$$

and

$$f^- = \frac{1}{2}(f - \alpha v)$$

where f is the characteristic flux function, and v is the characteristic variable, given from the dot product of the left eigenvector with the primitive variable. The intermediate flux function is given as $f = f^+ + f^-$. The term, α, plays the role of a viscosity, and is equal to the maximum eigenvalue (in absolute sense) of the two nearby nodes, i.e. (i, j) and $(i + 1, j)$ when calculating the flux function at $(i + 1/2, j)$. Now, high-order approximations to f^+ and f^- are needed at $(i + 1/2, j)$. Say for third-order, the f^+s are defined at five nodes $(i+h, j)$, and the f^-s at the five nodes $(i+h+1, j)$, where $h = -2, ..., 2$. Each of these are interpolated separately by the ENO algorithm described above, and then the intermediate flux function, f is then computed. This algorithm is very similar to one used in [27] to fix Roe's (or any other upwind scheme) at points where characteristic speeds change sign on the grid.

Note, other interpolation schemes, such as min mod and Superbee TVD, higher-order ENO and weighted ENO schemes can easily be used with the above algorithm, by simply adding or subtracting the number of nodes where the characteristic fluxes are defined. The three direct simulations described below were all calculated using the above described third-order ENO scheme. This completes the description of the interior algorithm. Next, an internal boundary method for Euler equations is described.

5.3 An internal boundary method for the Euler equations

Here, a perfectly reflecting internal boundary method for the Euler equations is presented. This methodology is almost identical to the method for the internal boundary method as applied to the level-set $\dot{D}_n - D_n - \kappa$ algorithm. Again, a nonevolving level set function, ϕ, is defined as the signed distance function, with $\phi = 0$ at the boundary, $\phi < 0$ in the flow region, and $\phi > 0$ in the boundary region. The internal boundary points are defined as any point that has $\phi > 0$, and any of the m^{th} points to the left,

right, up or down have $\phi < 0$. Here, m is the order of the scheme (or the number of boundary points the scheme needs), for example choosing $m = 4$ would be appropriate for a fourth-order scheme. For each internal boundary point there exists an associated reflected point, P_r, in the flow region. The internal boundary point will have the same state as the reflected point, except that the normal component of the velocity (or normal component of the momentum) will be of equal magnitude, but of opposite sign. This again reduces to a set of (possibly) coupled linear algebraic equations, and is solved using the iterative technique in Section 2.6.

5.4 Numerical solutions to 2-D unsteady detonations

Here, comparisons between direct numerical simulation of detonation, and level-set solutions to three intrinsic PDEs are made. The three intrinsic relations are: the Huygens' construction, a $D_n - \kappa$ relation and a $\dot{D}_n - D_n - \kappa$ relation. For our comparison, we take

$$r = 2.5147\,\mu\text{s}^{-1}(1-\lambda)^{\frac{1}{2}} \quad ,$$

as the rate law and use $Q = 4$ mm$^2/\mu$s^2, $\gamma = 3$ and upstream conditions $p_o = 10^{-4}$ GPa, $\rho_o = 2$ gm/cc and $\vec{u} = 0$. These parameters were chosen to mock up a condensed phase explosive with the ideal equation of state. These parameters give $D_{CJ} = 8$ mm/μs, and a steady-state 1-D half-reaction-zone length of 1mm (complete reaction-zone length is 4mm.) Each of the following cases were computed with 10 points in the half reaction zone (or 40 points in the complete reaction zone.) Each was also given the same initial conditions. A (numerically) steady CJ detonation traveling to the right with the shock located at $x = 6$ mm. The numerical steady traveling wave was computed by placing the exact ZND solution on the grid and allowing it to come to steady state numerically. All shock capturing schemes have some transient initial start-up errors associated with the smearing of the initial shock

profile. Using the numerical initial condition was done for the purposes of measuring intrinsic quantities, described later in Section 5.5.

5.4.1 Expanding channel

The first example will have the planar detonation diffract around a 45° corner. It is expected that the detonation front will decelerate as a rarefaction wave is sent through the reaction zone. This phenomenon is shown in Figure 20. Shown are two instantaneous gray-scale plots of the density, at 1.4 μs and 2.8 μs. Also seen in the plots are a contact discontinuity associated with a change in the temperature of the shocked material near the corner. This indicates a lower shock pressure, and thus a detonation front traveling below the CJ speed. Also, notice that it takes a finite time for the front to feel the effects of the rarefaction wave. This is clearly shown in the density plots.

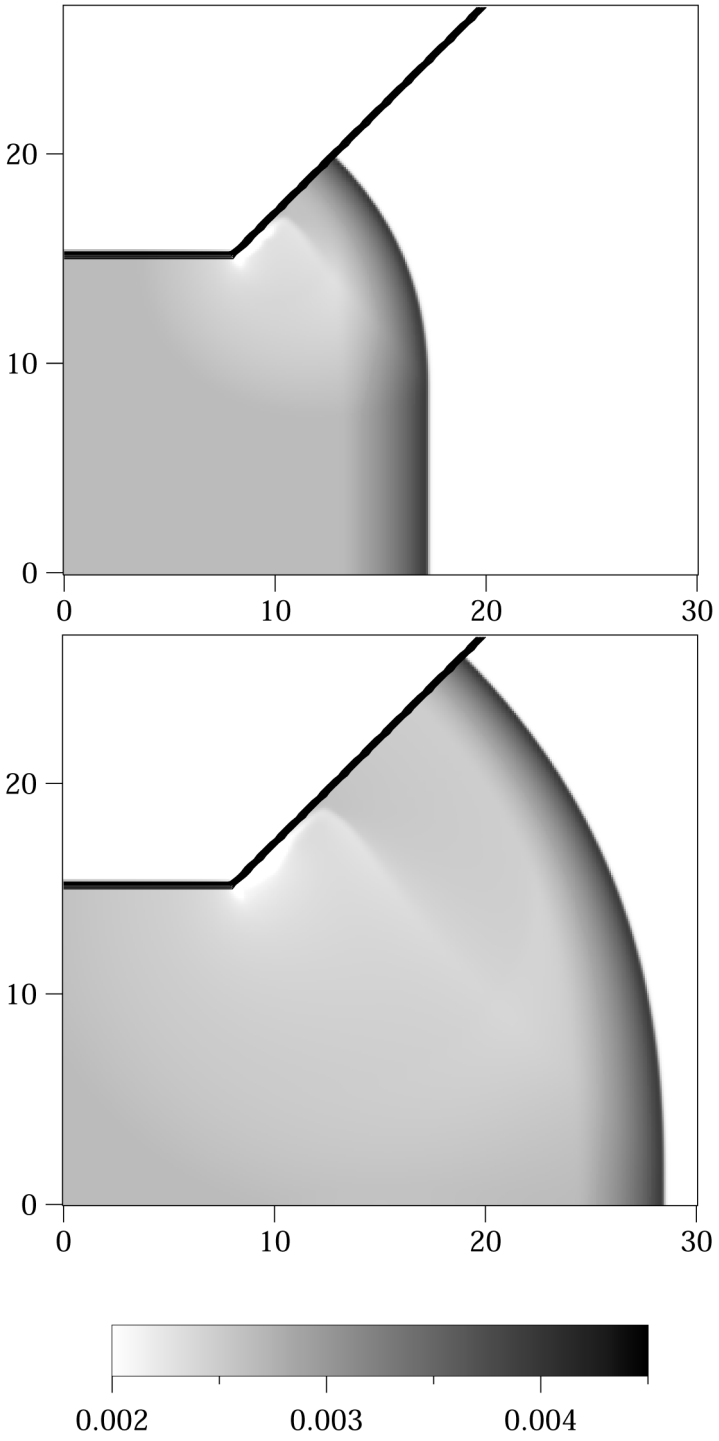

Figure 20: Density [gm/mm^3] gray-scale plot at 1.4 μs and 2.8 μs, as computed by the third-order ENO scheme.

5.4.2 Converging channel

This example focuses on the converging dynamics of detonations. A planer-CJ detonation will encounter a 20° ramp. The detonation shock initially forms a Mach-like reflection which slowly changes to a compressive wave. Now, the front speed is increased above the CJ value in the Mach-stem area. Figure 21 shows the density as a gray-scale image at times 1.75 μs and 3.5 μs. The classical Mach-reflection can be seen at early times, while at later times, the self-similar nature of the Mach-reflection is lost due to the reaction-zone effects.

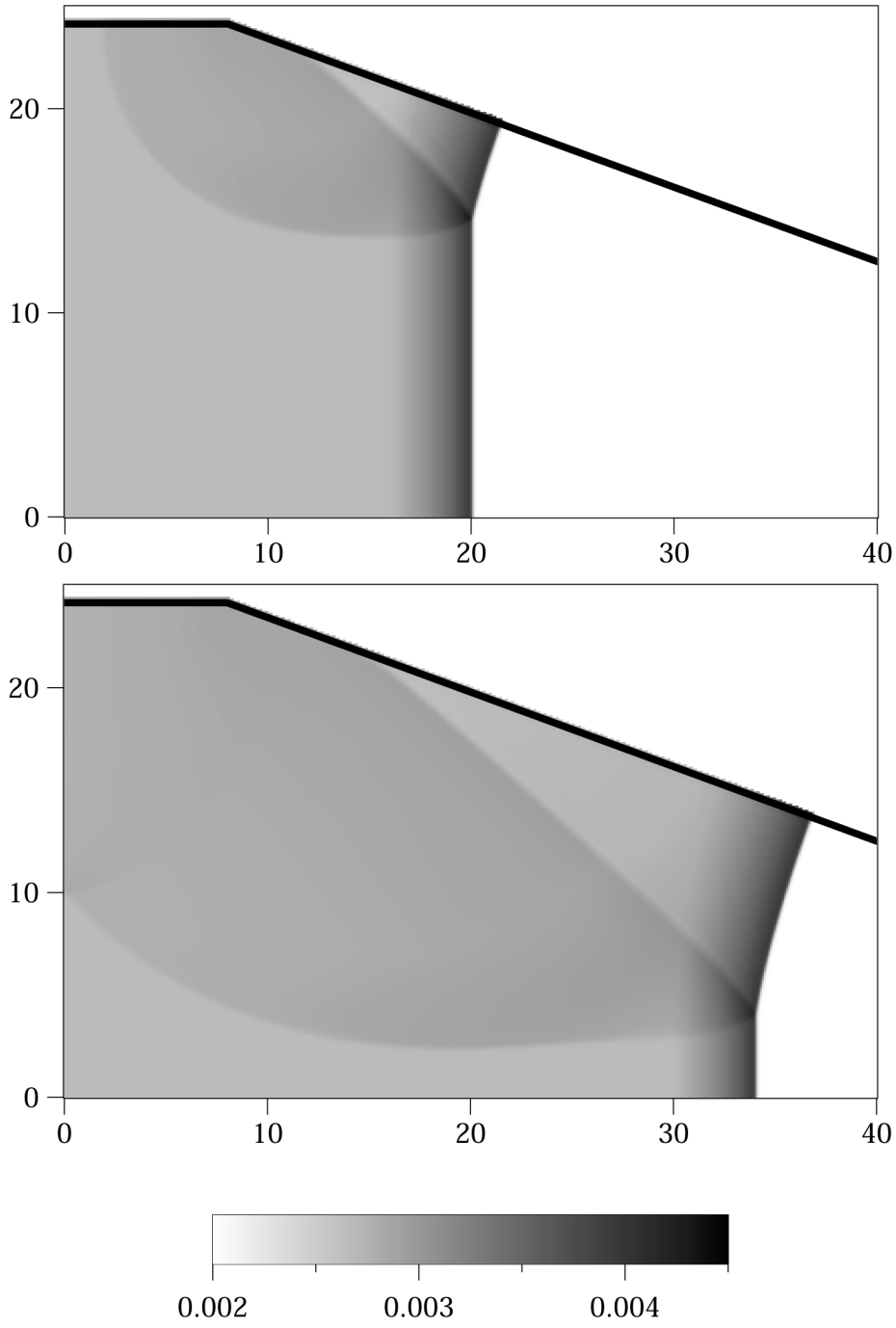

Figure 21: Density [gm/mm^3] gray-scale plot at 1.75 μs and 3.5 μs, as computed by the third-order ENO scheme.

5.4.3 Circular arc

The final example combines both converging and diverging aspects of detonation propagation. An initially planar detonation in a channel encounters a circular bend. The bend has an inner radius of 10 mm and an outer radius of 20 mm. See Figure 22. A rarefaction wave is initially generated at the inner bend, while a compressive wave is generated from the outer bend. These each influence the shape of the propagating detonation front. At roughly 1 μs after the detonation front encounters the bend, the compressive wave and rarefaction wave collide; eventually the front becomes kinked and forms a Mach-like reflection.

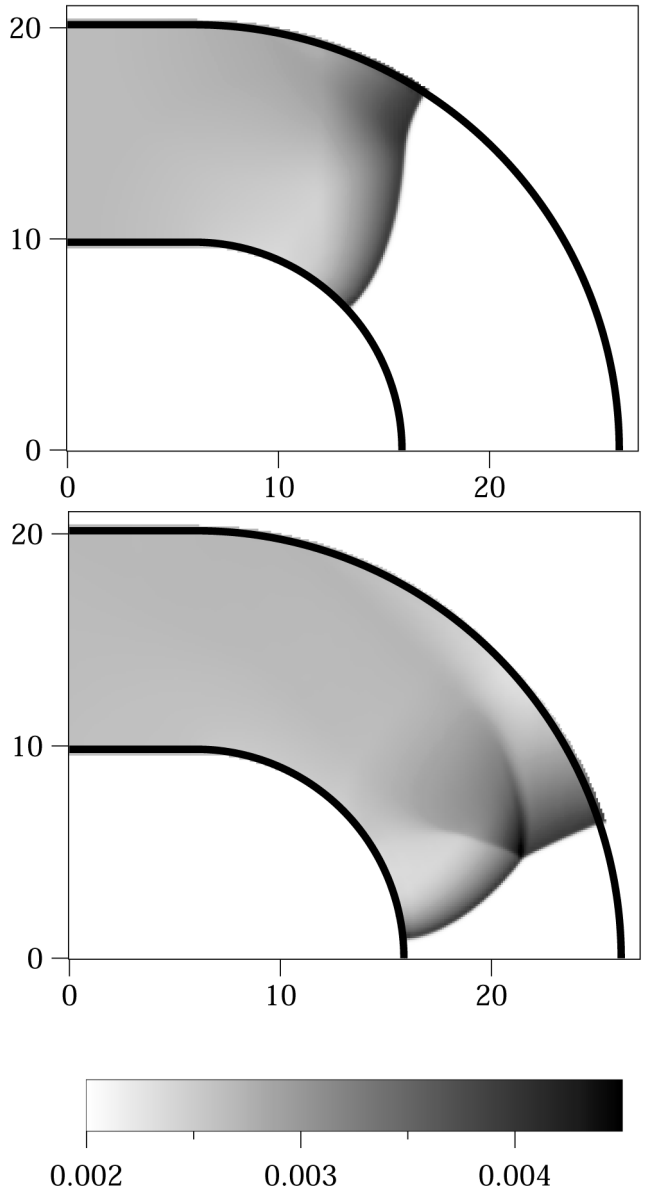

Figure 22: Density [gm/mm^3] gray-scale plot at 1.25 μs and 2.5 μs, as computed by the third-order ENO scheme.

5.5 Measuring intrinsic properties of the detonation shock front

One qualitative way of comparing DSD with a DNS is to simply look at the motion of the shock fronts generated by both solutions. Another method would be to suppose there exists an intrinsic relation that governs a detonation shock front, and try to measure this relation directly from a DNS.

As stated previously, one can obtain the dynamics of the detonation front by solving the compressible, reactive Euler equations with a DNS. Unfortunately, intrinsic shock-front information like the detonation shock speed, curvature of the shock front, etc. are not directly available from a DNS. But, notice that the fluid under goes a very strong shock (the Mach number of the shock is about 650), for this detonation model. Thus, the density jump at the shock is roughly a constant. So the detonation front may be approximated as some intermediate density (2.3gm/cc was used in these computations), between the undisturbed density (2 gm/c) and the shocked density (4 gm/cc). And for problems with quiescent upstream conditions, it is known that the detonation shock front will pass a fixed Eulerian point at most only once. So then it is possible to create a DNS burn table, by sweeping over the computational grid and searching for grid points where the quantity ($\rho - 2.3$ gm/cc) changes sign from one time level to the next. The first such occurrence will be when the shock passes over that fixed Eulerian point. Then, quadratic interpolation in time can be used to get an accurate estimate of the burn time, $t_b^{\mathrm{DNS}}(x, y)$. Once we have this DNS burn table, important quantities such as shock speed, curvature, etc. may be found. For example, the shock speed is given by $D_n = 1/|\vec{\nabla} t_b|$. The front locations are given simply as contours of $t_b^{\mathrm{DNS}}(x, y)$, and curvature of the shock front is given by (5) with t_b replacing ψ. The contours of the DNS burn times and instantaneous detonation velocities for the three previous examples are shown in the next section.

5.6 Comparison of DSD and DNS

All the problems considered here represent difficult tests for DSD, since the deviation of D_n from D_{CJ} is large. Next, results from the DNS–DSD comparison are given. For this model, DSD theory gives a $D_n(\kappa)$ relation shown in Figure 23, see [5]. DSD theory derives a $\dot{D}_n - D_n - \kappa$ relation which can not be written as $\dot{D}_n(D_n, \kappa)$, since $\dot{D}_n$ is not defined for certain regions of (D_n, κ) space, and is multivalued for others [4]. So, a $\dot{D}_n - D_n - \kappa$ relation was chosen to approximately give the steady $D_n(\kappa)$ relation when $\dot{D}_n = 0$, and the ratio of the coefficients between κ and $\dot{D}_n$ was chosen to give reasonable transverse propagation speeds, similar to the full Euler equations. The $\dot{D}_n - D_n - \kappa$ relation used here is given by:

$$\dot{D}_n(D_n, \kappa) = -0.261 D_n^2 \kappa + 2(\ln(8) - \ln(D_n))$$

Note that this $\dot{D}_n - D_n - \kappa$ relation was not derived, but rather emperically determined. A contour plot of the above relation is shown in Figure 24. Notice that the contour $\dot{D}_n = 0$ gives essentially the steady $D_n - \kappa$ relation of Figure 23. Each of the three relations will be solved using the level-set methods described in Chapters 2 and 3. Next, the detonation front dynamics generated from the DNS are compared with those from the three intrinsic relationships.

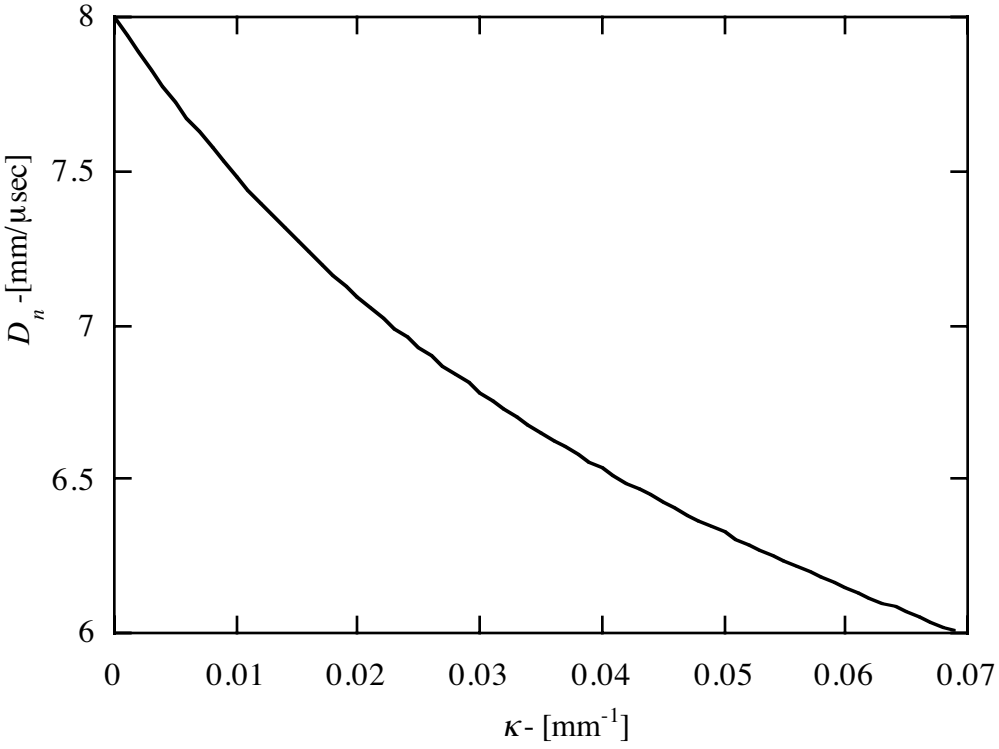

Figure 23: $D_n(\kappa)$ law for ideal equation of state model.

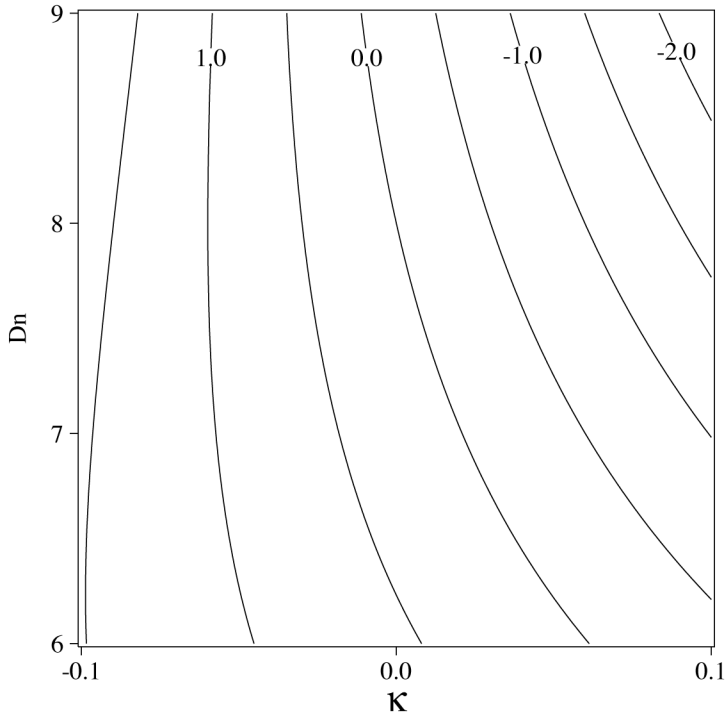

Figure 24: $\dot{D}_n - D_n - \kappa$ relation for ideal equation of state model.

5.6.1 Expanding channel

The measuring technique described in Section 5.5 is used to calculate the front locations and Eulerian records of the detonation velocity, D_n, from the DNS for the expanding channel problem. These records are displayed in Figure 25. The detonation velocity is clearly seen to decrease by roughly 40% from the D_{CJ} value of 8 mm/μs. Also, notice that the signaling speed is clearly evident in the simulation, and matches the correct speed given from acoustic theory [19].

The Huygens' solution is given in Figure 26. The dashed lines represent the fronts from the Huygens' solution, while the DNS fronts from Figure 25 are given as solid lines for comparison. Notice that there is a large discrepancy in the shapes and velocities of the fronts.

The $D_n - \kappa$ solution is given in Figure 27. The detonation front slows as the front goes around the corner. Also, since the underlying PDE is parabolic, the entire front instantaneously senses disturbances at the front, as seen by the gray-scale plot of the normal velocity. Although this is not physically correct, the dynamics of a $D_n - \kappa$ do predict velocity deficits, which were seen in the DNS.

The $\dot{D}_n - D_n - \kappa$ solution is given in Figure 28. Notice that the disturbances propagate at a finite speed from the corner, as predicted in Section 3.2. Notice also that for this problem the shapes and resulting detonation velocities compare well with the DNS.

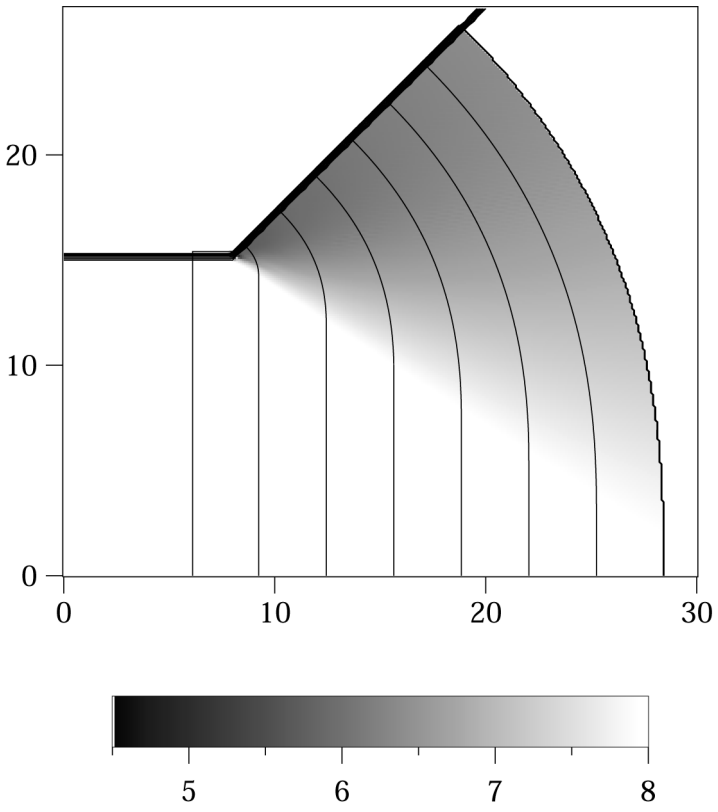

Figure 25: Fronts at intervals of 0.4 μs are shown as solid lines, and the detonation normal velocities [mm/μs] calculated from the DNS are given as the gray scale.

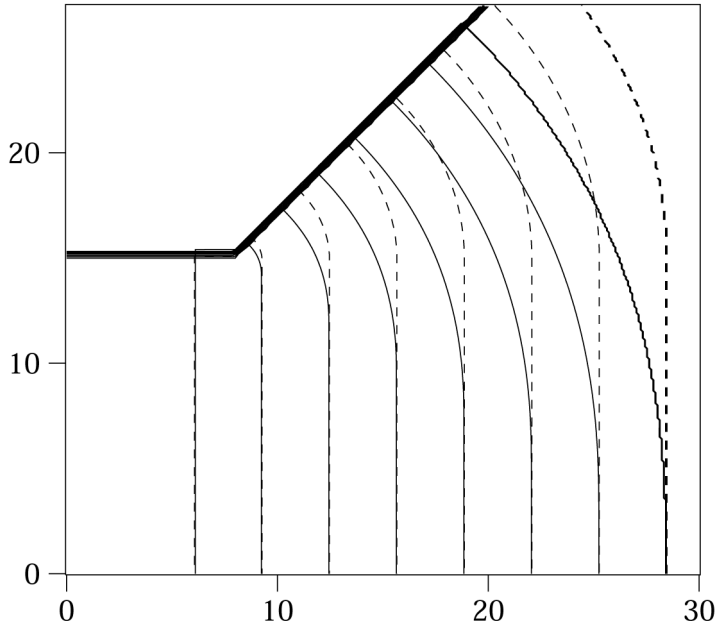

Figure 26: Fronts at intervals of 0.4 μs are shown as solid lines from the DNS, and as dotted lines from the Huygens' solution.

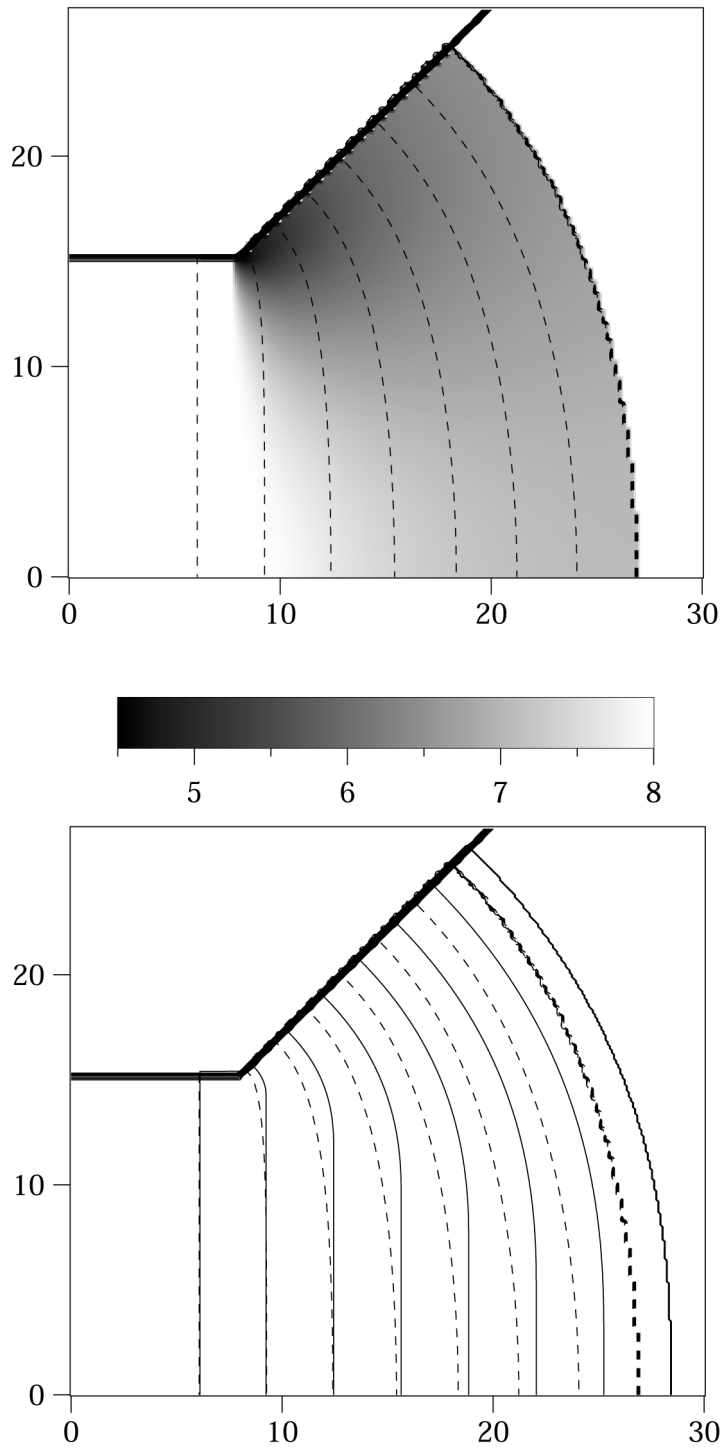

Figure 27: The top figure shows the fronts at intervals of 0.4 μs, and detonation velocities [mm/μs] as calculated from the level-set $D_n - \kappa$ solution. Fronts are shown as solid lines from the DNS, and as dotted lines from the $D_n - \kappa$ solution in the bottom figure.

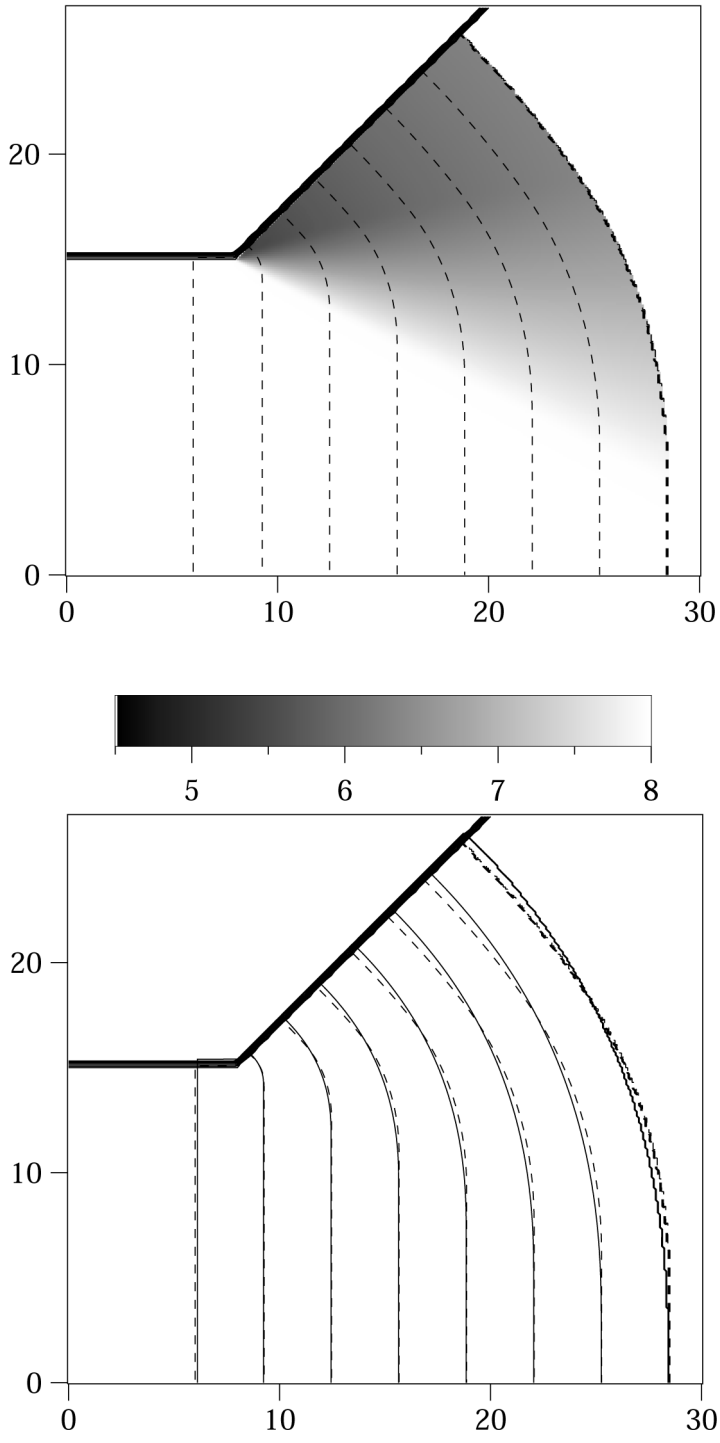

Figure 28: The top figure shows the fronts at intervals of 0.4 μs, and detonation velocities [mm/μs] as calculated from the level-set $\dot{D}_n - D_n - \kappa$ solution. Fronts are shown as solid lines from the DNS, and as dotted lines from the $\dot{D}_n - D_n - \kappa$ solution in the bottom figure.

5.6.2 Converging channel

The measuring technique described in Section 5.5 is again used to calculate the front locations and Eulerian records of the detonation velocity, D_n, from the DNS for the converging channel problem. These records are displayed in Figure 29. The detonation velocity is clearly seen to increase to about 9.5 mm/μs from the CJ value of 8 mm/μs. Also, notice that the disturbance from the wedge travels at a finite speed into the steady one-dimensional detonation region.

The Huygens' solution is given in Figure 30. The dashed lines represent the fronts from the Huygens' solution, while the DNS fronts from Figure 29 are given as solid lines for comparison. Notice that the Huygens' solution is just a flat wave solution, and no shape changes are predicted.

The $D_n - \kappa$ solution is given in Figure 31. The detonation front increases in speed as the front changes angle at the upper boundary to satisfy the reflection boundary condition. Since the underlying PDE is parabolic, the entire front instantaneously senses disturbances at the front, as seen by the gray-scale plot of the normal velocity. Again, this is not physically correct, but the $D_n - \kappa$ solution does predict a velocity increase. Notice that the front at 3.5 μs is almost perfectly cylindrical in shape, because the front senses both the bottom and top confinement everywhere.

The $\dot{D}_n - D_n - \kappa$ solution is given in Figure 32. Notice that the disturbances propagate at a finite speed from the ramp. Also notice that there is initially a kink in the wave front, associated with a shock–shock-like reflection from the ramp. Although it is difficult to detect from the plot, this solution, unlike Whitham's GSD, is not self-similar. The detonation velocity is actually decreasing along the ramp wall as a function of time. This is due to the $2(\ln 8 - \ln D_n)$ term in the $\dot{D}_n - D_n - \kappa$ relation. Notice also that for this problem the shapes and resulting detonation velocities compare well with the DNS.

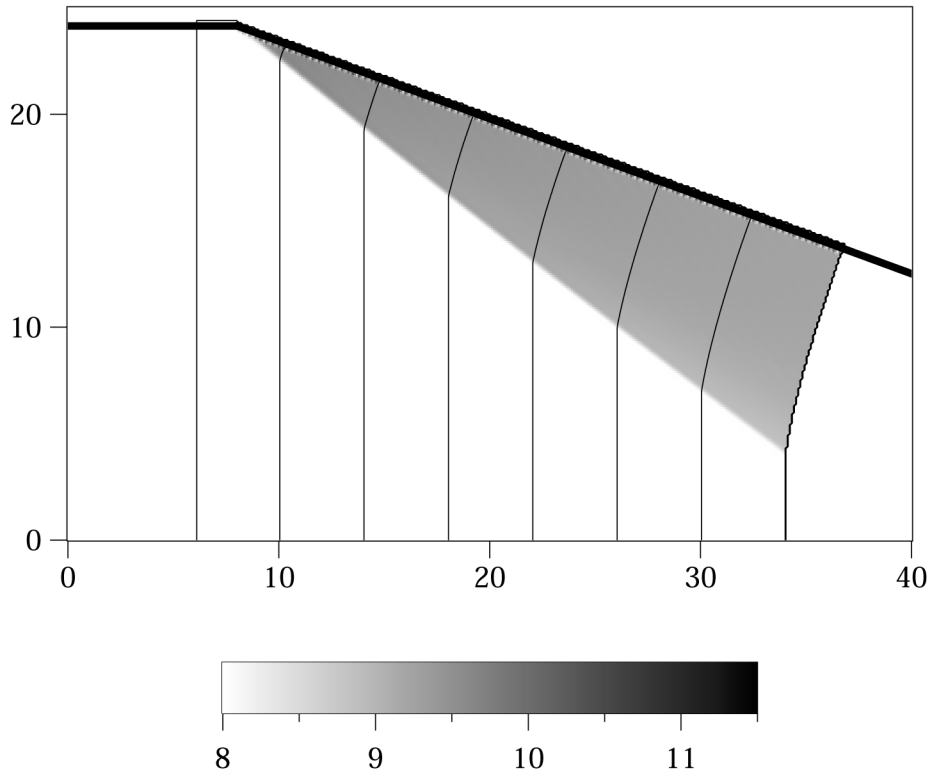

Figure 29: Fronts at intervals of 0.5 μs are shown as solid lines, and the detonation normal velocities [mm/μs] calculated from the DNS are given as the gray scale.

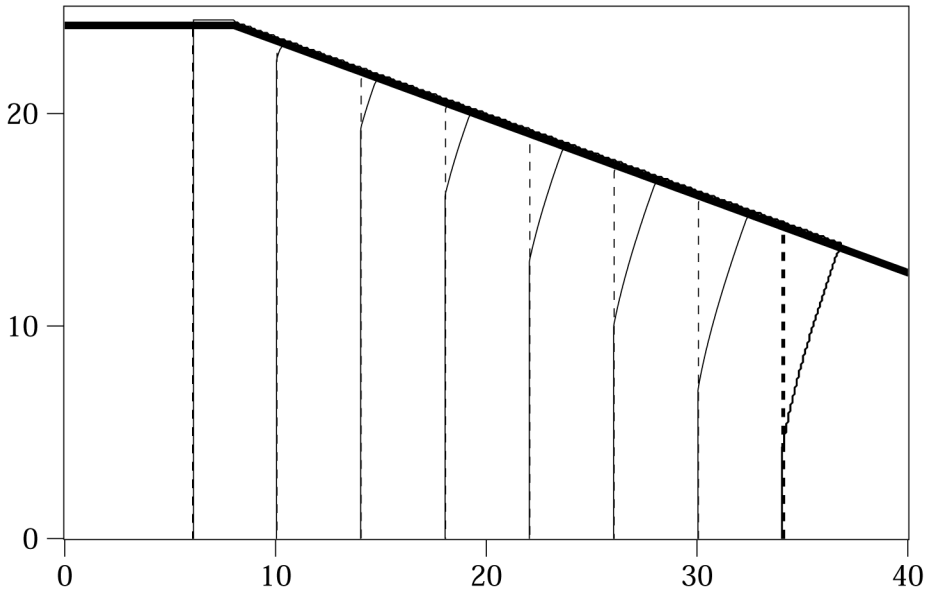

Figure 30: Fronts at intervals of 0.5 μs are shown as solid lines from the DNS, and as dotted lines from the Huygens' solution.

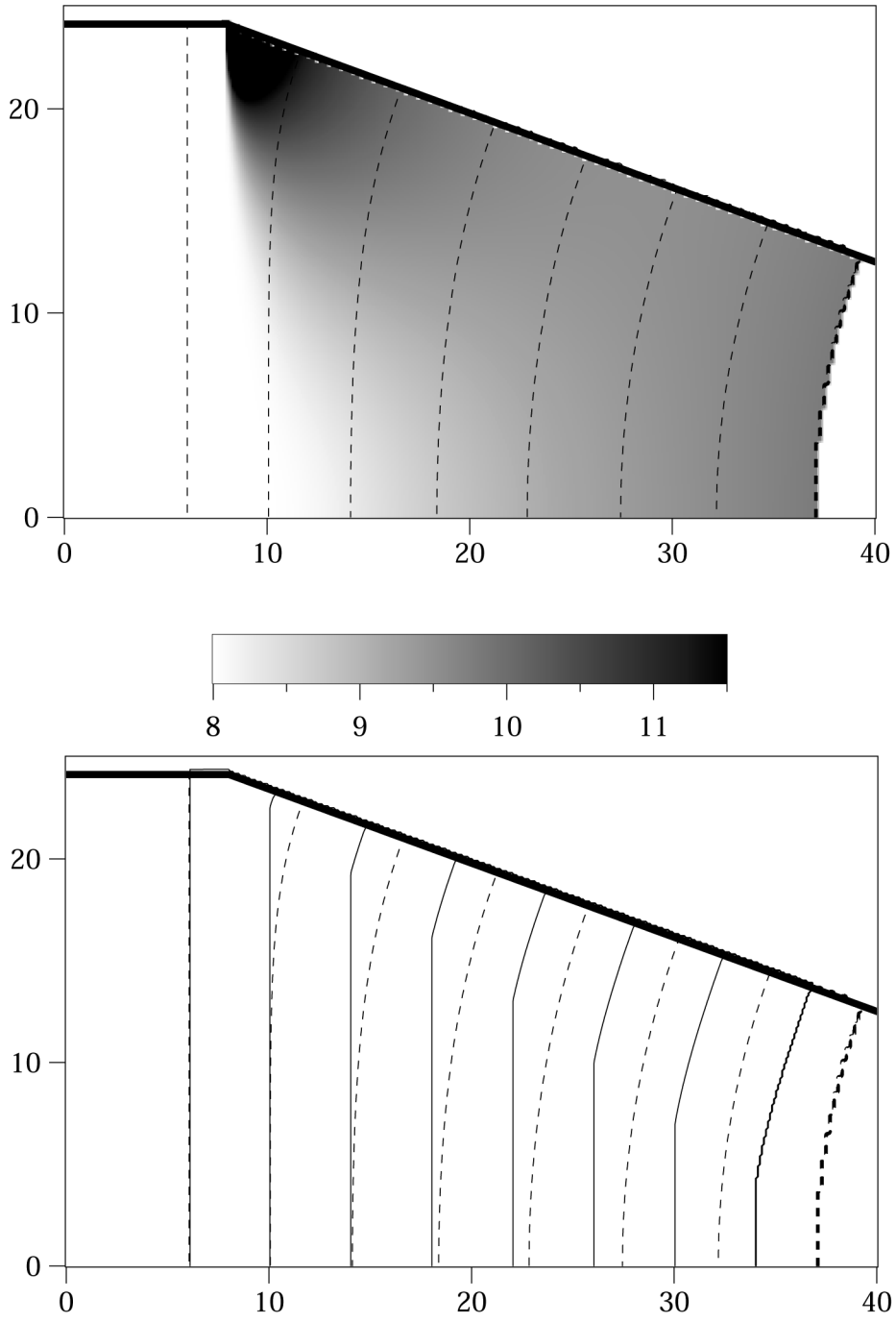

Figure 31: The top figure shows the fronts at intervals of 0.5 μs, and detonation velocities [mm/μs] as calculated from the level-set $D_n - \kappa$ solution. Fronts are shown as solid lines from the DNS, and as dotted lines from the $D_n - \kappa$ solution in the bottom figure.

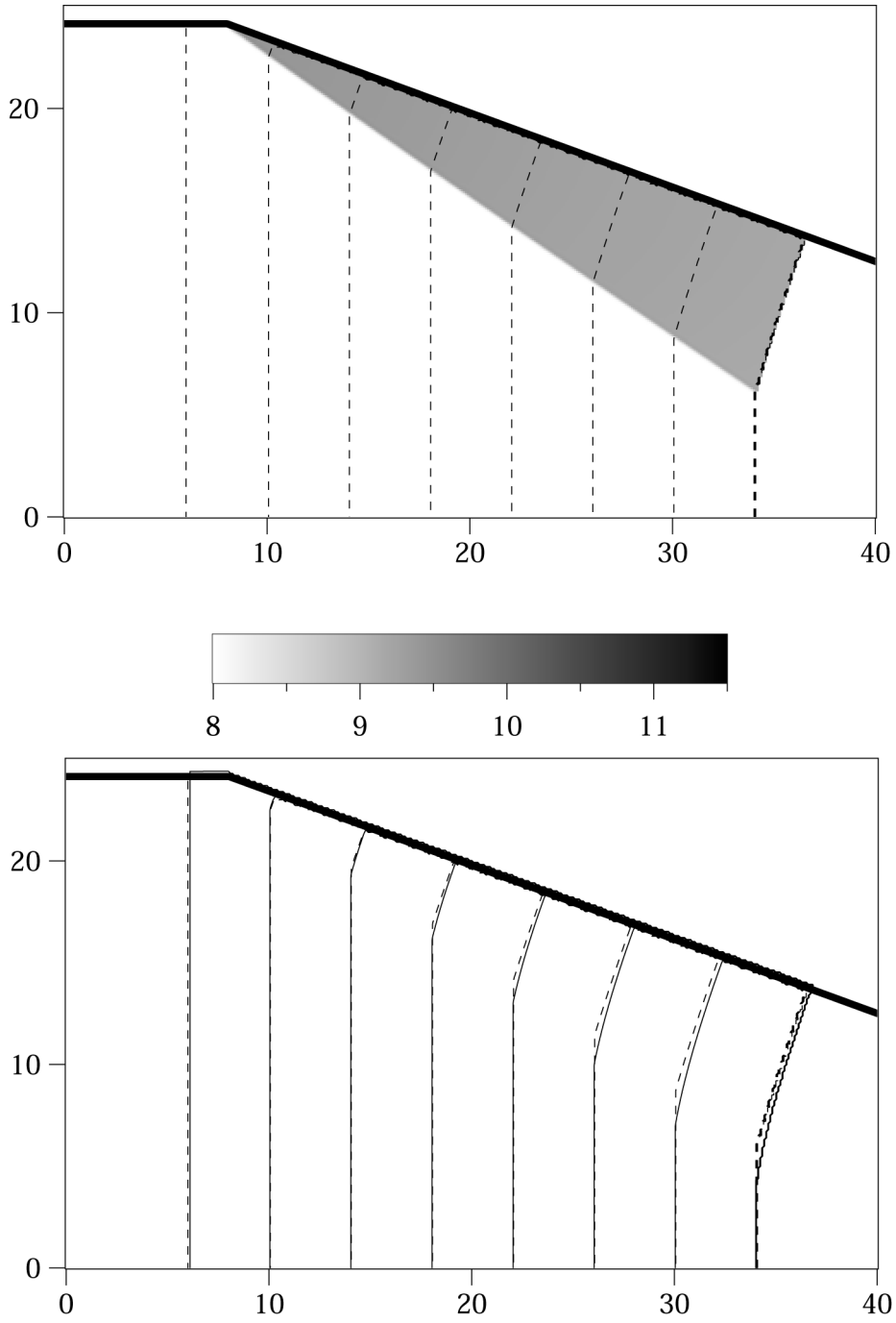

Figure 32: The top figure shows the fronts at intervals of 0.5 μs, and detonation velocities [mm/μs] as calculated from the level-set $\dot{D}_n - D_n - \kappa$ solution. Fronts are shown as solid lines from the DNS, and as dotted lines from the $\dot{D}_n - D_n - \kappa$ solution in the bottom figure.

5.6.3 Circular arc

Again the measuring technique described in Section 5.5 is used to calculate the front locations and Eulerian records of the detonation velocity, D_n, from the DNS for the circular arc problem. These records are displayed in Figure 33. The detonation velocity is clearly seen to increase along the outer bend, where the detonation senses a compressive wave, and is far below D_{CJ} along the inner bend, where there is a rarefaction wave, and the detonation diverges. Also, notice that the disturbance from the edges can be seen to travel at a finite speed into the steady one-dimensional detonation region.

The Huygens' solution is given in Figure 34. The dashed lines represent the fronts from the Huygens' solution, while the DNS fronts from Figure 33 are given as solid lines for comparison. Notice that the Huygens' solution predicts a flat wave along the top of the circular arc, and diffracts around the inner radius of the arc without any decrease in speed. Notice that the general shapes and locations are quite different than the DNS.

The $D_n - \kappa$ solution is given in Figure 35. The detonation front increases in speed along the upper boundary to satisfy the reflection boundary condition, and decreases along the inner radius. Also, the fronts become steady in a frame rotating with the arc very quickly, again this can be attributed to parabolic nature of the $D_n - \kappa$ relation. Although this relation does not predict the shapes very well, the fronts seem to be on average in roughly the right locations.

The $\dot{D}_n - D_n - \kappa$ solution is given in Figure 36. Notice that the disturbances propagate at a finite speed from the inner and outer bends. Also notice that there is a kink that eventually forms, when the compressive wave from the outer radius breaks and forms a shock–shock interaction. Notice also that for this problem the shapes and resulting detonation velocities compare well with the DNS.

Notice also that these three problems are very difficult tests, since the velocities vary far from D_{CJ}, and the curvatures, and time dependence are relatively large.

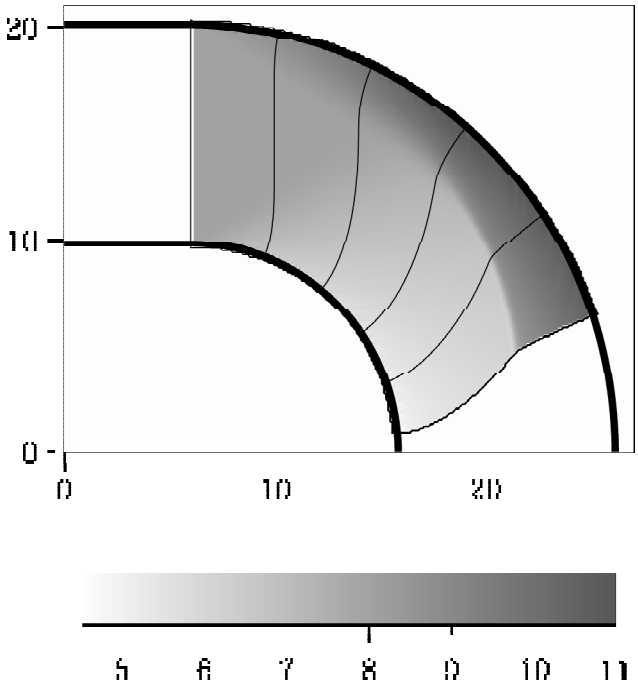

Figure 33: Fronts at intervals of 0.5 μs are shown as solid lines, and the detonation normal velocities [mm/μs] calculated from the DNS are given as the gray scale.

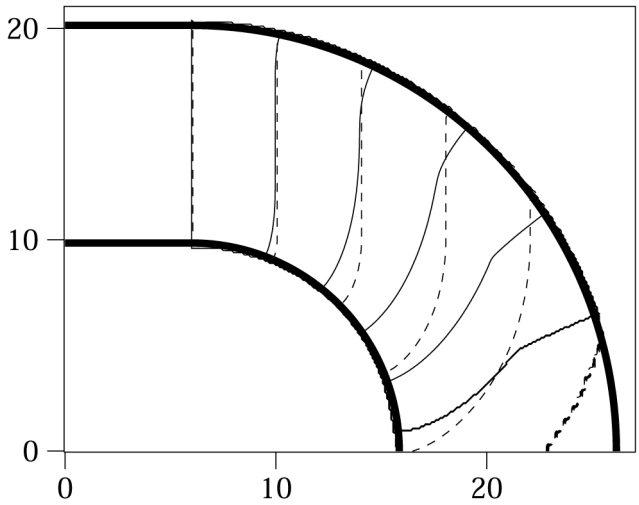

Figure 34: Fronts at intervals of 0.5 μs are shown as solid lines from the DNS, and as dotted lines from the Huygens' solution.

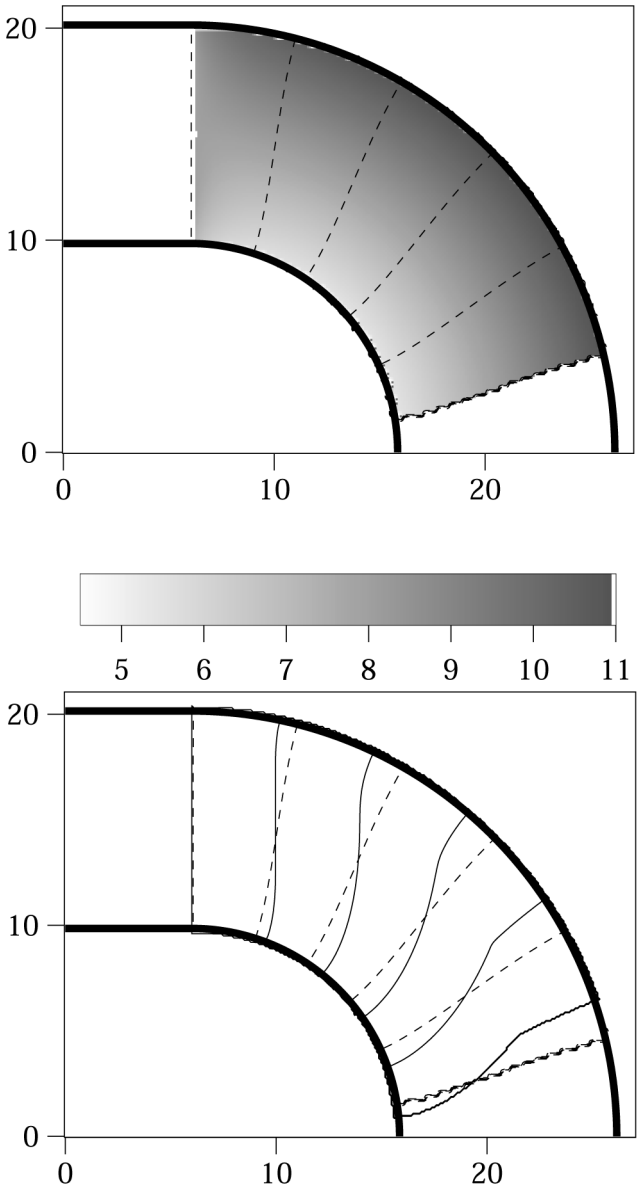

Figure 35: The top figure shows the fronts at intervals of 0.5 μs, and detonation velocities [mm/μs] as calculated from the level-set $D_n - \kappa$ solution. Fronts are shown as solid lines from the DNS, and as dotted lines from the $D_n - \kappa$ solution in the bottom figure.

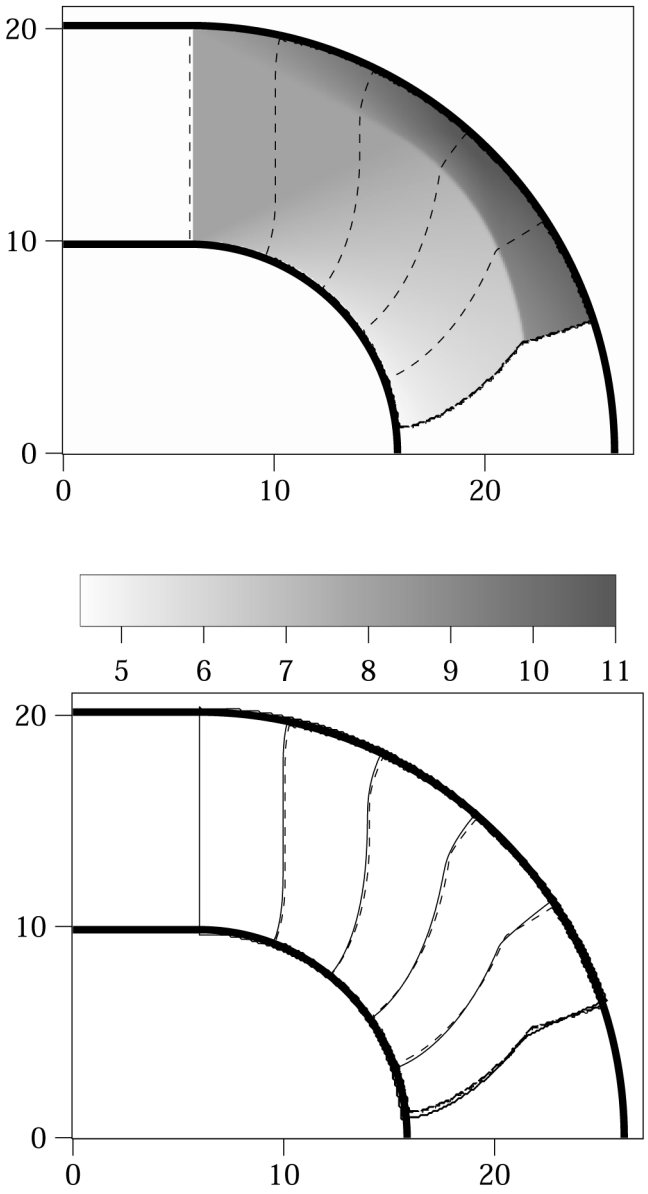

Figure 36: The top figure shows the fronts at intervals of 0.5 μs, and detonation velocities [mm/μs] as calculated from the level-set $\dot{D}_n - D_n - \kappa$ solution. Fronts are shown as solid lines from the DNS, and as dotted lines from the $\dot{D}_n - D_n - \kappa$ solution in the bottom figure.

6 Conclusions

Here, a few conclusions are given, along with possible avenues for future research in detonation shock dynamics and level-set methods.

First, it is clear that recently derived intrinsic relations have the ability to reproduce detonation phenomena seen experimentally and with direct numerical simulation. These new intrinsic relations will have a direct impact on how design engineers deal with detonation propagation. It is clear from Chapter 5 that stable detonation dynamics can be described quite well by a $\dot{D}_n - D_n - \kappa$ relation, and cellular dynamics can be described well by a $\ddot{D}_n - \dot{D}_n - D_n - \dot{\kappa} - \kappa$ relation. Work has been done on bringing the technology of detonation-front propagation to engineering and design codes.

The level-set approach to solving complex topological and geometric problems seems to be natural. This approach fits in well with the engineering applications, which generally are quite complex.

Also, the use of high-accuracy numerical methods for studying basic detonation problems can not be overlooked. It can serve as a tool for evaluating theoretical and experimental work.

Finally, a few areas of proposed future research are given. Notice that there exist several fields in mechanics which deal with front evolution. These include the previously discussed DSD theory and Whitham's GSD theory. Others include the motion of flame fronts, solidification/melting fronts, J. P. Best's theory of shock dynamics [28], [29], and motion of hydrodynamic jumps, to name a few. Level-set methods can be used as a tool for predicting the motion of these fronts, when resolving them is too computationally expensive. Also, the internal boundary method and level-set methods can be coupled to solve complex multi-dimensional, multi-phase fluid flow.

It would seem worthwhile to re-examine some original experiments originally intended for measuring $D_n - \kappa$ responses [11], and measure the acceleration terms, which clearly become important for certain problems, as seen in Chapter 5. Also, it may be possible to derive analytically or measure experimentally intrinsic relations that will give explicit criteria for detonation failure. Again, direct numerical simulation would be very useful to confirm such theories.

References

[1] Chapman, D. L., "On the rate of explosion in gases," *Philosophical Magazine*, 47, 90-104 (1899).

[2] Jouguet, E., "On the propagation of chemical reactions in gases," *Journal de Mathematiques Pures et Appliquees*, 1, 347-425 and 2, 5-85 (1906).

[3] Wood, W. W., and Kirkwood, J. G., "Diameter effect in condensed explosives. The relation between velocity and radius of curvature in the detonation wave," *Journal of Chemical Physics*, 22, 1920-1924 (1954).

[4] Yao, Jin, and Stewart, D. S., "On the dynamics of multi-dimensional detonation," *Journal of Fluid Mechanics*, 309, 225-275 (1996).

[5] Stewart, D. S., and Bdzil, J. B., "The shock dynamics of stable multidimensional detonation," *Combustion and Flame*, 72, 311-323 (1988).

[6] Fickett, W., and Davis, W. C., *Detonation*, University of California Press (Berkeley) (1979).

[7] Bdzil, J. B., and Stewart, D. S., "Modeling two-dimensional detonation with detonation shock dynamics," *Physics of Fluids A*, 1, 1261-1267 (1989).

[8] Osher, S., and Sethian, J. A., "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations," *Journal of Computational Physics*, 79, 12-49 (1988).

[9] Bryson, A. E., and Gross, R. W. F., "Diffraction of strong shocks by cones, cylinders and spheres," *Journal of Fluid Mechanics*, 10, 1-16 (1961).

[10] Stewart, D. S., and Bdzil, J. B., "A lecture on detonation shock dynamics," in *Mathematical Modeling in Combustion Science, Lecture Notes in Physics*, 299, 17-30, Springer-Verlag (New York) (1988).

[11] Bdzil, J. B., Davis, W. C., and Critchfield, R. R. "Detonation shock dynamics (DSD) calibration for PBX9502," submitted to *Physics of Fluids* (1995).

[12] Shu, C.W., and Osher, S., "Efficient implementation of essentially non-oscillatory shock-capturing schemes," *Journal of Computational Physics*, 77, 439-471 (1988).

[13] Aslam, T. D., Bdzil, J. B., and Stewart, D. S., "Level set methods applied to modeling detonation shock dynamics," submitted to *Journal of Computational Physics* (1995).

[14] Bdzil, J. B.,"Steady-state two-dimensional detonation," *Journal of Fluid Mechanics*, 108, 195-226 (1981).

[15] Leveque, R. J., "High resolution finite volume methods on arbitrary grids via wave propagation," *Journal of Computational Physics*, 78, 36-63 (1988).

[16] Berger, M. J., and Leveque, R. J., "An adaptive Cartesian mesh algorithm for the Euler equations in arbitrary geometries," AIAA Paper 89-1930-CP (1989).

[17] Quirk, J. J., "An alternative to unstructured grids for computing gas dynamic flows around arbitrarily complex two-dimensional bodies," *Computers in Fluids*, 23, 125-142 (1994).

[18] Leveque, R. J., *Numerical Methods for Conservation Laws*, Birkhauser Verlag (Basel) (1992).

[19] Whitham, G. B., *Linear and Nonlinear Waves*, Wiley (New York) (1974).

[20] Sussman, M., Smereka, P., and Osher, S., "A level set approach for computing solutions to incompressible two-phase flow," *Journal of Computational Physics*, 114, 146-159 (1994).

[21] Sethian, J. A., and Strain, J., "Crystal growth and dendritic solidification," *Journal of Computational Physics*, 98, 231-253 (1992).

[22] Strehlow, R., *Combustion Fundamentals*, McGraw-Hill (New York) (1984).

[23] Lee, J. H. S., "Dynamic parameters of gaseous detonations," *Annual Review of Fluid Mechanics*, 16, 311 (1984).

[24] Shu, C. W., and Osher, S., "Efficient implementation of essentially non-oscillatory shock-capturing schemes II," *Journal of Computational Physics*, 83, 32-78 (1989).

[25] Shu, C. W., "Numerical experiments on the accuracy of ENO and modified ENO schemes," *Journal of Scientific Computing*, 5, 127-149 (1990).

[26] Rogerson, A., and Meiburg, E., "A numerical study of the convergence properties of ENO schemes," *Journal of Scientific Computing*, 5, 151-167 (1990).

[27] Quirk, J. J., "A contribution to the great Riemann solver debate," *International Journal for Numerical Methods in Fluids*, 18, 555-574 (1994).

[28] Best, J. P., "A generalisation of the theory of geometrical shock dynamics," *Shock Waves*, 1, 251-273 (1991).

[29] Best, J. P., "Accounting for the transverse flow in the theory of geometrical shock dynamics," *Proceedings of the Royal Society of London A*, 442, 585-598 (1993).

Vita

Tariq Dennis Aslam was born in Chicago, Ill., on September 6, 1969. He attended Illinois Benedictine College for one year, and then transferred to the University of Notre Dame, where he received his bachelor's degree in Mechanical Engineering in May 1991. He then attended the University of Illinois at Urbana-Champaign, and received his master's degree in Theoretical and Applied Mechanics in May 1993. While completing his thesis, he worked at Los Alamos National Laboratory as a graduate research assistant. He has given presentations on detonation shock dynamics and level-set methods at Los Alamos National Laboratory, at the Midwestern Universities Fluid Mechanics Retreat, and at the Sixth International Conference on Numerical Combustion.