The Characteristic Tracking Method: A Method for Integrating Nonlinear Hyperbolic Equations that Accurately Resolves both C^0 and C^1 Discontinuities

John Weatherwax, Dimitri Vaynblat, Oscar Bruno, and Rodolfo R. Rosales

September 18, 2004

Abstract

In this paper we develop a new method for the numerical solution of hyperbolic systems of conservation laws in 1-D, which we call the *Characteristic Tracking Method* (CTM). The CTM has several important advantages over other numerical methods currently in use. Specifically, the CTM gives *one grid point* resolution of discontinuity waves (shocks, contacts, etc), rarefaction fan edges, and boundaries between simple waves of different types. This last property, the ability to capture C^1 discontinuities to the accuracy of a single grid point is, to the authors knowledge, beyond the reach of current numerical methods. In addition, the CTM both (i) does not generate spurious waves as discontinuities travel through the numerical grid, and (ii) preserves linearly degenerate waves (like contact discontinuities) sharp, without any spreading.

1 Introduction

In this paper we report on a new method for the solution of hyperbolic systems of conservation laws in 1-D, which we call the *Characteristic Tracking Method* (CTM). The CTM was developed during the investigation of a problem involving martensitic phase transitions in solids [3, 4, 51]. There the need arose for accurate calculations of the interactions between rarefaction fans and discontinuous waves — such as shocks, contact discontinuities, and phase transformations. The Characteristic Tracking Method has several important advantages over other methods currently in existence for the numerical solution of nonlinear hyperbolic systems of conservation laws. The CTM is very accurate at following interfaces between regions, since it resolves discontinuities, fan edges, and boundaries between simple waves of different types[1] "exactly". By this we mean that the CTM is able to specify, at each time step (to within the resolution of the method) exactly where any discontinuous wave is located. There is no smearing of waves such as shocks, contacts, and rarefaction fan edges. We found that the CTM performs extremely well in practice. A bullet–summary of the algorithm follows. A detailed description can be found in section 4 of this paper.

- **a** • The CTM is an *event-driven* algorithm, where the "time steps" span the interval between wave interactions — in between, each wave is followed "exactly" with a moving numerical grid (one node per wave).
- **b** • At each event, a (single) Riemann problem [27] (at the location of the interacting waves) is solved, and the interacting waves are replaced by the new waves produced by the solution of the Riemann problem.
- **c** • Only at initialization must a Riemann problem be solved at each node of the computational grid. At this time the initial data are discretized in the same way as for standard Godunov-type schemes [27], and the

[1] Boundaries of this last type occur in the investigations reported in the papers [3, 4, 51] cited earlier.

initial set of waves is produced by solving a Riemann problem at each node.

d • For **(a-c)** above to be possible, all the waves produced by the solution of each Riemann problem must be "discrete". Thus, the CTM uses a *Discretized Riemann Solver*, where "continuous" waves (i.e. expansion fans) are replaced by a set of many "discrete" waves (each a small discontinuity).

e • In regards to **(d)**, we note that "standard" Godunov-type schemes use a full Riemann solver, whose solution they must discretize at each time step. This they do by (effectively) averaging the Riemann solver answer over the numerical cells. On the other hand, the CTM uses a Riemann solver whose output is discrete to begin with, avoids the need to average its answer, and thus eliminates wave smearing.

f • Because the numerical grid follows the waves (there is no real grid, just a tracking of where the waves are), the typical noise generated by wave-grid interactions [6, 30, 54] is avoided. In the same fashion, because the wave-wave interactions are solved explicitly, no numerical noise [12, 48] arises from them either. Finally, no spreading of linearly degenerate waves (such as contacts [10, 13, 16]) occurs.

The description above applies to what we may call a "pure" CTM strategy. A potential drawback to worry about for such a strategy is the possibility of **wave interaction proliferation**, where the wave interactions become too frequent and the time steps tinny. Even though the method can tolerate much smaller time steps than a standard Godunov-type scheme,[2] there is a point at which efficiency would be badly affected. At any rate, it is clear that

g • There are situations where wave interaction proliferation is not a problem, and where the CTM is very efficient, and accurate. An example

[2] Since at every time step a single Riemann problem must be solved, as opposed to one at every numerical node.

of this occurs in the application that motivated the development of the algorithm [3, 4, 51].

h • In cases where wave interaction proliferation is a problem, one can switch to a mixed strategy, where the CTM approach is only used for the waves of "interest" (say: as determined by some threshold value for their strength), and elsewhere a "standard" scheme is used. Since the standard scheme would only have to deal with the regions where the solutions are smooth, it could be rather simple.

These are issues that we are currently investigating, and will be reported elsewhere. In this paper we will restrict the description to that of a pure CTM scheme. Furthermore, *we will describe the scheme in its simplest setting only.* Namely:

i • For a 2×2 system, where the existence of a complete set of Riemann invariants introduces a considerable simplification into the structure of the Discrete Riemann Solver.

j • For the case of a first-order scheme, where the numerical solution is piece-wise constant (constant between numerical nodes, with possible jumps at the nodes).

Modifications to the method needed when phase transitions occur will be presented elsewhere [52]. We are currently working on developing Discrete Riemann Solvers for cases where Riemann invariants are not available [49, 50] — e.g. the Euler equations of gas dynamics. As for higher order versions of the method, we believe that they can be done using ideas analogous to the ones that were succesful in extending the first order Godunov scheme to higher orders [5, 6, 45, 46, 47]. Unfortunately, we do not foresee any extension to more than 1-D.

The structure of this paper is as follows: in section 2 we present a brief history of the research that has been done on the numerical computation of solutions to nonlinear hyperbolic systems of conservation laws. In section 3

we present a brief overview of the mathematical equations chosen here to demonstrate the features of the characteristic tracking method. In section 4 we describe fully the CTM algorithm. In section 5 we present some results of the characteristic tracking method. Finally, section 6 has the conclusions and suggestions for additional work.

2 Prior work

Numerical methods for systems of nonlinear hyperbolic conservation laws have a long history. The methods of Courant et al. [8] and Lax and Friedrichs [22], developed in the early 1950s, stand out as one of the earliest attempts at the numerical solution of problems of this type. Since then a large number of schemes have been developed.

The early approaches relied on the application of *finite difference* techniques. Of particular importance is the pioneering work of Lax and Wendroff [23, 24, 25] on space–centered schemes. This work lead to several centered three–point schemes for one–dimensional hyperbolic equations, a popular one being that of MacCormack [29].

Later on finite difference methods more closely related to the physical propagation properties of the equations were derived. These are non–space–centered schemes, known as upwind schemes (since they bias the finite differences to attempt to capture the propagation of information via the characteristics). The first explicit upwind scheme appeared in the work of Courant et al. [8]. Since then several extensions have appeared [36]. The flux vector splitting methods of Steger and Warming [44] and van Leer [26] can be considered upwind methods due to their directional discretization of the flux derivatives.

A very different, and original, approach to the numerical solution of nonlinear hyperbolic problems was taken by Godunov [12]. He approached the solution to the global problem by solving many *local* Riemann problems. The

use of an exact nonlinear solution to locally approximate the flow has lead to a host of schemes that introduce different Riemann solvers [11, 33, 37, 38]. The original scheme by Godunov was only first order, but it has been successfully extended to higher orders [5, 6, 14, 15, 17, 19, 20, 46, 47].

The principal difficulty encountered in the numerical solution of nonlinear hyperbolic systems stems from the fact that the mathematical equations themselves can develop discontinuities in finite time, even if starting from smooth initial conditions [27]. These discontinuities (e.g. shocks and contact discontinuities) are notoriously hard to resolve numerically. Most of the schemes described above suffer their greatest difficulties near the locations of these moving interfaces. The numerical problems involved range from miscalculation of their speeds, to non-physical oscillations present before or after their location, to excessive smearing of the wave profiles. In addition, small parasitic oscillations are generated as discontinuous waves (e.g. shocks) move through the numerical grid — or when wave interactions occur [6, 12, 30, 48, 54].

Many numerical schemes have been invented to correct the problems mentioned above. One classic idea, aimed specifically at correcting the oscillations generated by a shock, is to introduce a non–physical viscosity into the governing equations themselves [32]. Another approach aims at preventing the generation of numerical oscillations before they occur. This approach is based on the concept of non–linear *limiters*, introduced by Boris and Book [2] and van Leer [45]. This work eventually lead to the concept of *total variation diminishing (TVD)* schemes, first introduced by Harten [14]. Many of these classical ideas and concepts are discussed in LeVeque's book [27].

One difficulty with the approach proposed by van Leer [45, 46, 47] for the generation of high order Godunov schemes, is that the complexity of the algorithm grows rapidly with the order — to our knowledge, nothing higher than second order [6] has ever been implemented for realistic physical systems. The problem stems from the add hoc "switches" that must

be used to prevent the generation of oscillations (near discontinuities) by the cell-interpolating polynomials, which rapidly become extremely complicated. A way around this difficulty is provided by the Essentially Non-Oscillatory (ENO) schemes, that were proposed and developed by Chakravarthy, Enquist, Harten, and Osher [16, 17, 19, 20, 41, 42]. For each computational cell, ENO methods consider several neighboring cells, and choose to represent the dependent variables by the interpolating polynomial that has the least oscillation over all possible choices (up to a fixed order). This allows the generation of high order schemes with relative ease, as the complexity of the algorithms grows much less rapidly than with the van Leer approach to higher order Godunov schemes.

A variant of ENO are the WENO (for Weighted ENO) schemes [21, 28, 40] — where [40] is a survey article for both ENO and WENO. These methods combine the results obtained when using all possible stencils, rather than choosing only one. A weighted combination of the results from all stencils is used, using weights that select smoother approximations.

Finally, we should mention the vital and important level set techniques [34, 35, 39]. These methods are ideally suited for the numerical computations of interfaces (e.g. shock fronts), and include the ability to break propagating fronts into many pieces (if required) as the flow evolves. These, however, are eminently multi-dimensional computational techniques, and are not relevant to the subject of our discussion here.

A numerical approach that motivated much of the work presented here is that of Harten and Hyman [18]. Harten and Hyman developed a scheme where the strong shocks are tracked by some of the points in the numerical grid, with the remaining grid points stationary. In this fashion they are able to get shock resolutions of one grid point. However, their approach breaks down when wave interactions occur.

None of the methods described above captures rarefaction wave edges. In addition, all have trouble with contacts — which are "smeared" over several

numerical grid points, with the amount of smearing growing with time. This difficulty arises because contacts are linearly degenerate, so that there is no nonlinear "sharpening" to counteract the (always present) numerical viscosity. The first attempt to curb this diffusion of the contact discontinuities was made by Harten [13], the idea being to numerically destroy the linear degeneracy of the waves — by introducing a (numerical) steepening mechanism. An algorithm introducing this idea (within the context of ENO schemes for the Euler equations of gas dynamics) was developed by E and Young in [10].

Artificial steepening of contacts requires extreme care with implementation, so as not to generate non-physical discontinuities — say, by applying too much artificial convergence of the characteristics in places where it is not needed at all. Another drawback is that (in the inviscid Euler equations) shocks should be the only structures capable of mechanical energy dissipation and entropy production; but this approach makes the contacts into shock-like structures, potentially capable of dissipation and entropy production.

A different approach to the problem of contact smearing, also introduced by Harten [16], is that of subcell resolution. The main idea here is to *detect* the presence of a contact discontinuity (from the local information about the flow provided by the numerical scheme), and then use this information so it is propagated without smearing. The "subcell" in the name of the approach stems from the fact that knowledge of the location of the detected contact implies the need to infer information about events at a resolution higher than that provided by the numerical cell size. Our approach eliminates the need to infer subcell resolution information, by making sure that the waves are always at a numerical node — or rather: that a node is always where the wave is.

Basic ideal requirements for an optimal scheme.

Based on the discussion above, we list here some simple idealized requirements that an "optimal" numerical scheme for nonlinear hyperbolic systems of conservation laws should satisfy.

1 • Detection and resolution of discontinuities as they are created, automatically, without user intervention or a priori knowledge of the solution to the equations.

2 • Resolution of discontinuities should be as exact as possible — meaning that as few computational grid points as feasible should be used for each discontinuity (ideally: one).

3 • No non-physical oscillations should be generated by the propagation (or interaction) of discontinuities. This is particularly problematic if the noise generated "detaches" from the wave, and acquires independent status as a numerical noise wave.

4 • Discontinuities must propagate with the correct physical speeds — i.e. they must satisfy the Rankine-Hugoniot conditions.

5 • Conservation of all the appropriate quantities (mass, momentum, etc) should occur.

Obviously, **3-5** need not apply strictly, and errors can be tolerated, as long as they are small (and remain so). The characteristic tracking method developed in this paper satisfies all of the above requirements for the numerical solution of one-dimensional nonlinear hyperbolic conservation laws, as well as some additional requirements that are not listed. For instance, in the example that motivated the development of the algorithm [3, 4, 51], the presence of a non-convex equation of state can produce an expanding simple wave region with a *common space-time boundary* with a compressive simple wave region[3]. The CTM algorithm can track this boundary accurately – which turns out to be very important for a proper treatment of the physics of the problem.

In the next section we present the mathematical equations that will be used to describe the CTM. At mentioned in the introduction (see items **i** and **j** there), in this paper we will present the method for the simplest setting —

[3]For an complete description of such a situation see [3, 4, 31, 43, 51].

which is very close to the situation that motivated the development of the method (save for the fact that we will not include phase transitions here).

3 Mathematical Equations

As mentioned in the introduction[4], in this paper we describe the CTM algorithm as it applies to the simplest setting of a 2×2 system of equations. For this purpose we use the *p-System* [43], which (in Eulerian coordinates) has the form:

$$\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial x}(\rho\, u) = 0\,, \qquad \frac{\partial}{\partial t}(\rho\, u) + \frac{\partial}{\partial x}(\rho\, u^2 + p) = 0\,. \tag{1}$$

Here ρ is the *density*, u is the *flow velocity*, p is the *pressure*, and the equations represent the conservation of mass and momentum. The system is closed by an *equation of state*, relating the pressure to the density:

$$p = p(\rho) \quad \text{or} \quad p = p(\mathrm{v})\,, \tag{2}$$

where $\mathrm{v} = 1/\rho$ is the *specific volume*. On the other hand, in Lagrangian mass–coordinates, the p-System is given by:

$$\frac{\partial}{\partial t}\mathrm{v} - \frac{\partial}{\partial \xi}\mathrm{u} = 0\,, \qquad \frac{\partial}{\partial \mathrm{t}}\mathrm{u} + \frac{\partial}{\partial \xi}\mathrm{p(v)} = 0 \tag{3}$$

where the Lagrangian mass-coordinate ξ is related to the Eulerian space coordinate by:

$$\frac{\partial}{\partial t}x = u \quad \text{and} \quad \frac{\partial}{\partial \xi}x = \mathrm{v} \tag{4}$$

The Euler equations of gas dynamics reduce to the p-System [53] under the assumption of constant entropy. The equations also appear in 1-D nonlinear elasticity [9], where $T = -p$ is the stress. They have also been used to model

[4] See item **i** there.

flyer-plate experiments [9] — the main subject [3, 4, 51] that motivated the derivation of the CTM.

For materials under normal conditions (i.e. *no phase transitions*) the equation of state is convex, with

$$c^2 = \frac{dp}{d\rho} > 0\,, \quad \text{and} \quad \frac{d^2p}{d\rho^2} > 0\,, \tag{5}$$

where $c = c(\rho) > 0$ is the *sound speed*. We will assume this in here, and will use the equations in their Eulerian form. The convexity condition is, typically, violated when a phase transition occurs[5] — see subsection 3.2. An extension of the algorithm to include phase transitions will appear elsewhere [52].

3.1 Some useful formulas

For completeness, we list some well known (but useful) facts and formulas for the p-System [7].

Rankine-Hugoniot jump conditions:

$$-S\,[\,\rho\,] + [\,\rho\,u\,] = 0\,, \qquad \text{and} \qquad -S\,[\,\rho\,u\,] + \left[\,\rho\,u^2 + p\,\right] = 0\,, \tag{6}$$

where S is the speed of the discontinuity, and the brackets denote the jumps in the enclosed variables across the discontinuity. Alternatively, in terms of the relative flow velocity $U = u - S$,

$$[\,\rho\,U\,] = 0\,, \qquad \text{and} \qquad \left[\,\rho\,U^2 + p\,\right] = 0\,. \tag{7}$$

A useful formula that follows from (6), upon elimination of S from the equations, is:

$$[\,p\,]\,[\,\mathrm{v}\,] = -\,[\,u\,]^2 = -\,[\,U\,]^2\ . \tag{8}$$

[5] As happens in the problem that motivated the development of the CTM [3, 4, 51].

At shocks $U \neq 0$, and at contacts $U = 0$ — in the p-System contacts occur at material interfaces, where the equation of state (2) changes: from $p = p^L(\mathrm{v})$ on one side, to $p = p^R(\mathrm{v})$ on the other.

Entropy condition at a shock:
Shocks must be compressive. Thus, for a *right shock:*

$$0 > U_L > U_R\,, \quad p_L > p_R\,, \quad \rho_L > \rho_R\,, \quad \text{and} \quad \mathrm{v_L} < \mathrm{v_R}\,, \tag{9}$$

where the subscripts R and L denote the states to the right and left of the shock, respectively. For a *left shock:* $0 < U_R < U_L$, and the other inequalities are reversed.

Riemann invariants (for continuous solutions):

$$\left.\begin{array}{lrcll} \text{left} \quad \text{invariant:} & s & = & -\frac{1}{2}\,(u-l)\,, & \text{constant along} \quad dx/dt = u - c = \lambda_L\,, \\ \text{right invariant:} & r & = & +\frac{1}{2}\,(u+l)\,, & \text{constant along} \quad dx/dt = u + c = \lambda_R\,, \end{array}\right\} \tag{10}$$

where $l = l(\rho)$ is defined by:

$$l(\rho) = \int_{\rho_0}^{\rho} \frac{c}{\rho}\,\mathrm{d}\rho = \int_{p_0}^{p} \frac{\mathrm{d}p}{\rho c}\,, \tag{11}$$

with p_0 (resp. ρ_0) some arbitrary reference pressure (resp. density). Sometimes it is convenient to use s and r as the dependent variables, instead of u and ρ.

Simple wave:
In a simple wave, one of the Riemann invariants is constant. For example, in a *right simple wave* $s \equiv s_*$ is constant. Then:

$$u = l - 2s_*\,, \quad r = l - s_*\,, \quad \lambda_R = l + c - 2s_*\,, \quad \text{and} \quad p\,, \tag{12}$$

are all *strictly monotone increasing functions of* ρ. A similar property holds for a *left simple wave.*

3.2 Equation of State for Martensitic Phase Transformations

Here we describe the specific pressure-volume relationship used in our investigation of martensitic phase transformations [3, 4, 51] – where convexity fails to apply. This equation of state is given by (see figure 1 for a graph):

$$p = \begin{cases} p^A(\mathrm{v}) & \text{for } p < p^A_{\mathrm{crit}}\,, \\ p^M(\mathrm{v}) & \text{for } p > p^M_{\mathrm{crit}}\,, \end{cases} \tag{13}$$

where $p^A = p^A(\mathrm{v})$ and $p^M = p^M(\mathrm{v})$ are both convex functions satisfying standard thermodynamic constraints [7] — see below, equation (14). The threshold pressures (p^A_{crit} and p^M_{crit}) are known constants. Furthermore $p^A_{\mathrm{crit}} > p^M_{\mathrm{crit}}$, with the corresponding densities satisfying $\mathrm{v}^{\mathrm{A}}_{\mathrm{crit}} < \mathrm{v}^{\mathrm{M}}_{\mathrm{crit}}$.

This equation of state models phase transitions between two states (Austenite and Martensite, described by $p^A(\mathrm{v})$ and $p^M(\mathrm{v})$, respectively). The transitions occur at different pressures:

- As the material is compressed, from Austenite to Martensite, at $p = p^A_{\mathrm{crit}}$ — with a sudden increase in density.
- As the pressure is lowered, from Martensite to Austenite, at $p = p^M_{\mathrm{crit}}$ – with a sudden decrease in density.

Away from the phase transition, each material (phase) satisfies a pressure–volume relationship given by its own thermodynamically stable (and convex) equation of state. The functions $p^A(\mathrm{v})$ and $p^M(\mathrm{v})$ are given by a Mie-

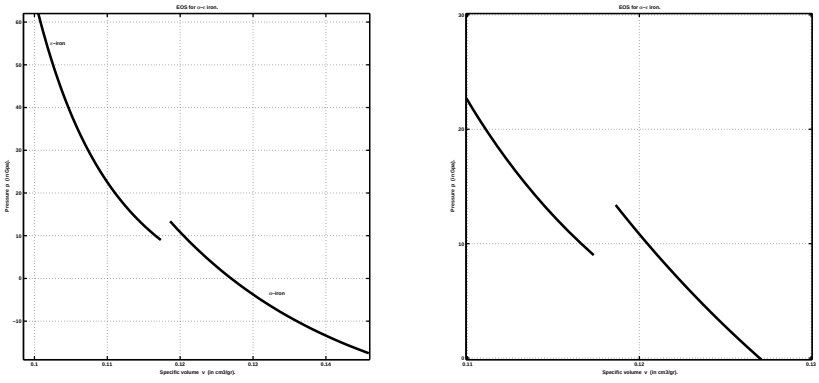

Figure 1: Plots of the equation of state in (13 – 14), with the constants given in table 1. The plot on the right is an enlargement of the region near the phase transformation thresholds.

Grüneisen [9] equation of state:

$$p(\mathrm{v}) = \frac{\rho_0 \, \mathrm{a}_0^2 \, \eta(\mathrm{v})}{(1 - \mathrm{s} \, \eta(\mathrm{v}))^2}\,, \qquad \text{with} \qquad \eta(\mathrm{v}) = 1 - \frac{\rho_0}{\rho} = 1 - \frac{\mathrm{v}}{\mathrm{v}_0}\,, \tag{14}$$

where a_0, s, and ρ_0 are known constants. Table 1 gives the values of the various constants for the α–ϵ phase transition in iron. Figure 1 gives a graphical example of equation (13).

Phase	ρ_0 (kg/m^3)	c_0 (m/s)	s unitless	p_{crit} (Pa)
α-iron	7874	4630	1.33	13.38×10^9
ϵ-iron	7874	3200	2.30	9.00×10^9

Table 1: Material constants used in the equation of state for α and ϵ iron.

Using this equation of state, we will demonstrate the CTM's ability to track phase transformation fronts, and the boundaries between simple waves of different compressibilities [3, 4, 51] — see also section 5. Even though we will

not consider here situations where phase transformations occur[6], we point out that incorporating them into the algorithm is rather simple conceptually [52] (albeit a bit cumbersome). The restriction of no phase transformations amounts to assuming either that $p^A(\mathrm{v}) \equiv \mathrm{p}^{\mathrm{M}}(\mathrm{v})$ in equation (13), or that $p > p^M_{\mathrm{crit}}$ everywhere and for every time, or that $p < p^A_{\mathrm{crit}}$ everywhere and for every time. Thus, only shocks and rarefaction waves will appear in the solution. In fact, we will also incorporate contacts, by allowing regions with different material properties. Effectively, the equation of state in (2) will be function of the material coordinate: $p = p(\mathrm{v}, \xi)$, but with a rather simple dependence on ξ (piece-wise constant).

4 Description of the Algorithm

In this section we fill in the details for the bullet description of the algorithm given in section 1 (items **a**–**f**). The main issue to consider is the *Discrete Riemann Solver* used by the CTM, which is at the core of the algorithm. Since standard Riemann solvers for the p-System are well known, we will concentrate mostly on the differences with them. The (first order) algorithm works as follows[7]:

1 • *Computational grid.*

The CTM uses an irregular, moving grid. The computational domain is divided into several (not necessarily equal) consecutive intervals (the *cells*), whose end points are the *nodes*. Specifically, let $x_0 = x_0(t) < \ldots x_n = x_n(t) < x_{n+1} = x_{n+1}(t) < \ldots x_N = x_N(t)$ be the nodes, where the computational domain is the interval[8] $x_0 < x < x_N$, and the n-th cell is the interval $x_{n-1} < x < x_n$. For a first order method, such as

[6]The wave dynamics in this case has the interesting feature that phase transformation fronts can change "type" as the result of the interactions with other waves.

[7]The CTM is a close cousin of the Godunov-type methods [12, 27], but there are important differences.

[8]Boundary conditions may be needed at x_0 and/or x_N.

the one that we will describe here, the numerical solution is constant on each cell. We point out that *the number of nodes N changes during the course of the computation.*

2 • *Discrete Riemann Solver.*

The Discrete Riemann Solver (DRS) is an approximate Riemann solver, whose output is discretized. To be precise, consider an arbitrary Riemann problem [27] to be solved. Then the output of the DRS is an (approximate) solution consisting of a set of waves (discontinuities) traveling at speeds $w_1 < \ldots < w_i < \ldots < w_M$, separated by constant states (ρ_i, u_i) , $0 \leq i \leq M$. In a DRS, the expansion fans (that can arise in the solution to a Riemann problem) are replaced by a set of small discontinuities — separated by constant states. The maximum size Δ_{DRS} of these discontinuities is a numerical parameter — see item **11**. The strategy used by the DRS to deal with expansions fans is the only way in which it differs from a standard Riemann solver. Thus, this is the only point on which we will expand (see subsection 4.1).

3 • *Algorithm initialization.*

The initial conditions for the algorithm must be provided as a piecewise constant function, over a grid such as the one described in item **1** above. To start the CTM, a Riemann problem is solved at each node of the initial grid — using the DRS. Each DRS solution gives rise to a set of waves, and each wave (moving at a known velocity, from a known starting point) gets a node assigned to it. This creates the initial set of waves with which the CTM operates. Further details relevant to the algorithm initialization are given in subsection 4.2, item **A**.

4 • *Wave interactions = events.*

The CTM propagates each wave (node) undisturbed till it interacts (collides) with another. In the CTM there are no time-steps, only events. Each event affects *only* the two colliding nodes. In principle, triple (and higher) order collisions are possible, but such things have

(essentially) zero probability. Thus we will not consider them in the description here — though the CTM has no particular problem with such events, as should be clear from point **5** below.

5 • *Riemann problem at an event.*

The CTM procedure at an event is very simple. Right there the colliding nodes become one, and the local situation is that of a Riemann problem that needs solution. Thus, the DRS is invoked, the colliding nodes are replaced by the ones produced by the DRS, and the algorithm goes forward till the next event happen. Figure 2 illustrates this.

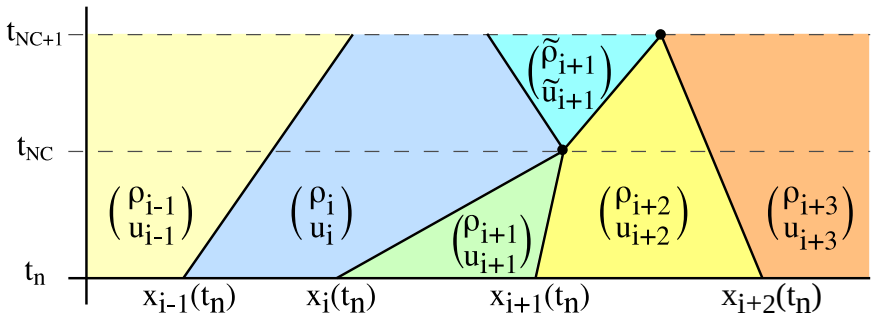

Figure 2: Cartoon of the CTM space-time structure, assuming that each wave-interaction Riemann problem produces exactly two waves. The picture starts at some arbitrary time t_n, with a given set of propagating waves. The first interaction time $t_{\rm NC}$ after t_n (before which the algorithm takes no action) is defined as the earliest time for a wave collision. Here this involves the nodes propagating from $x_i(t_n)$ and $x_{i+1}(t_n)$. At this point a Riemann problem is solved, the interacting nodes are replaced by the waves produced by the DRS solution, a new state is inserted into the flow (i.e. $(\tilde{\rho}_{i+1}\,,\ \tilde{u}_{i+1})$), and the algorithm proceeds. The next collision occurs at time $t_{\rm NC+1}$, where the process is repeated. The solution is constant in each colored polygon.

The items listed above, modulo the details of how expansion fans are discretized (see subsection 4.1), gives a fairly complete description of the method. There are, however, **several important further points to consider**, as follows:

6 • *Riemann problems at events are "special".*
The Riemann problem that must be solved at an event is not an "arbitrary" Riemann problem, but results from the interaction of two waves — both of which are known. Furthermore, the large majority of the wave interactions involve only "weak" waves, or a "weak" and a "strong" wave. This situation can be exploited, to make the solution of these problems more efficient — see subsection 4.2.

7 • *Lack of grid-noise and interaction-noise generation.*
A direct benefit of the grid strategy used, and the lack of any cell averaging, is that no parasitic waves are generated as strong shocks move through a grid, a common problem [6, 30, 54]. In addition, no spurious entropy is generated by shock interactions [12, 48].

8 • *No smearing of contacts and rarefaction fan edges.*
The slow diffusion of contact discontinuities [10, 13, 16] and rarefaction fan edges, by numerical viscosity, is not a problem.

9 • *Node proliferation problem.*
A potential problem for the CTM grid strategy is that of node proliferation, as new waves are born and are inserted into the flow. This in itself is not a problem (since the algorithm does nothing with the nodes, unless they interact), but it can be one if it gives rise to wave interaction proliferation.

10 • *Wave interaction (event) proliferation problem.*
We have already mentioned this problem in the introduction — see the discussion following item **f** there. Let us point out here that one factor reducing the number of interactions is that waves of the same family [9] tend to have collisions well separated (since, in the smooth parts of the flow, their space-time paths are nearly parallel).

[9] Note that the waves, when weak, are just the way this method represents characteristics.

11 • *Error control.*

There are two ways in which errors are controlled in the CTM. One occurs at the algorithm initialization (see item **3**), via how well the initial conditions are approximated on the initial grid. The second is provided by the parameter Δ_{DRS} introduced in item **2**: the smaller Δ_{DRS}, the better the approximation by the DRS of the rarefaction fans. We note that the parameter Δ_{DRS} also appears in the developments of subsection 4.2 — in particular, see item **A** there.

The following subsections complete the CTM description by: (i) giving the details of how the DRS treats rarefaction fans — item (**2**) above; and (ii) showing how knowledge of the waves colliding at an event can be used to simplify the solution of the resulting Riemann problem — item (**6**) above.

4.1 Rarefaction Fans

In this subsection we describe how rarefaction fans are discretized by the Discrete Riemann Solver for the p-System. A very simple (though perhaps not optimal) strategy for how to deal with fans in the DRS is presented. The approach is based on the fact that Riemann invariants are available for the p-System, and so it is particular to this case. At the end of the subsection we consider the issue of how to generalize the approach to general systems.

First of all: small enough rarefaction fans are simply treated as propagating discontinuities (what, precisely, small means here will become clear in the discussion that follows). We call these discontinuities *characteristic waves* or *numerical characteristics*, as in fact they are a discrete numerical approximation to the characteristics that carry the rarefaction wave. Larger rarefactions are decomposed into several smaller discontinuities, each moving along its own path. The specific details of how the fans are split into smaller pieces are not too important, as long as the smaller parts are roughly the same "size". The simplest way to split rarefaction fans occurs when Riemann

invariants are available for the conservation laws. In this case, the fan is split in such a way that the jumps in the Riemann invariant corresponding to the fan are all of equal size.

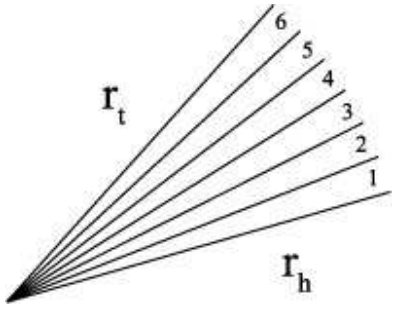

Figure 3: Discretization of right-facing rarefaction fan with 6 constant state wedges and 7 characteristic waves.

As an example, consider the p-System and a right (i.e. carried along the characteristics with speed $\lambda_R = u + c$) rarefaction fan, as in figure 3. In this case the left invariant is constant (say: $s \equiv s_0$) throughout the fan, while the right invariant changes from a value $r = r_h$ (on the right, or head of the wave) to a value $r = r_t$ (on the left, or tail of the wave). Let now N be the smallest integer such that $\Delta r = |r_t - r_h|/(N+1) \leq \Delta_{\text{DRS}}$, where Δ_{DRS} is the numerical parameter introduced in item **2** at the beginning of this section. Then discretize the fan into N cells/wedges of constant states and $N + 1$ "characteristic" waves, as follows below (figure 3 illustrates the process for a case where $N = 6$).

- The value of s is constant throughout, $s \equiv s_0$.
- The value of r is equi-distributed along the cells, jumping by an amount Δr across each of the numerical characteristic.
- Each of the numerical characteristics has a speed which is the average of the characteristic speeds $\lambda_R = u + c$ on each side of the wave.

The process is entirely similar for a left-facing rarefaction fan. In this way a continuous centered rarefaction fan is replaced by $N+1$ characteristic waves,

separated by N wedges where the solution is constant. Note that each of the numerical characteristics is a simple wave discontinuity.

In the general case, without Riemann invariants, the process is the same (split rarefaction fans into sequences of constant state wedges, separated by small discontinuities), but the implementation is a bit more complicated [49, 50]. Each of the small discontinuities is still a simple wave, and the jumps across them are all equal in some appropriate measure (there is no Riemann invariant that can be used for this purpose, so other means are needed).

4.2 Event Riemann problems

In this subsection we consider the event Riemann problems, as functions of the interacting waves and their types. The main point of this subsection is to describe how to use the information about the interacting waves to produce a faster Riemann solver[10]. The idea is as follows: First, each wave (node) in the CTM is assigned a *type*, which the algorithm keeps track of. The type of a wave is set when created, and kept till the wave is destroyed at an event. The types are:

A • *Numerical characteristics.*

We will also call them characteristic waves, or just characteristics. They are small size, and they approximate the true characteristics of the system — carrying information about the continuous parts of the flow. These waves are simple wave discontinuities, with the jump in the corresponding Riemann invariant smaller than Δ_{DRS}. They are produced when rarefaction fans are discretized (see subsection 4.1), at algorithm initialization (from the smooth parts of the initial conditions), and at wave interaction events (see the rest of this subsection).

[10] This an "optional" process. A plain Riemann solver can be used at each interaction, ignoring the available wave information — at a considerable efficiency cost.

B • *True discontinuities.*

These are arbitrary size, and represent the true discontinuities that the solution has — they can either be shocks or contacts (sub-types). They arise from the initial conditions, from interactions involving other true discontinuities, or from the wave breaking produced by the nonlinearity in the system (see the rest of this subsection, in particular item $\mathbf{C}_2$ in subsubsection 4.2.2). We will frequently abbreviate their name to just "discontinuities".

The type (and sub-type) information is used at a collision (event) to simplify the solution by the DRS (Discrete Riemann Solver). The possible wave-wave interactions that can occur are as follows:

1 • The two colliding waves are both true discontinuities.

2 • The two colliding waves are both numerical characteristic waves. This is, by far, the most common type of event. It is also the one for which the DRS simplifications are the largest.

3 • The colliding waves types differ: a true discontinuity with a numerical characteristic.

In the next few sub-subsections we consider each of the situations enumerated above (for the p-System). In each case we show what the DRS solution is, by describing:

4 • What types (and sub-types) of waves are produced, and what their velocities are.

5 • The values of the dependent variables in the wedges between pairs of produced waves.

The presentation here is only for the p-System, without phase transformations, and so there are only two kinds of discontinuities: shocks and contacts — with the contacts arising because we allow the waves to propagate through different materials (see the end of section 3). Phase transformations will be

considered in [52]. A final point to be made is that, while the idea of using information about the waves colliding at an event to speed up the DRS is general, the specific details of how this is done are system dependent. This is because the speed up is produced by the fact that we can predict which waves will be produced by the interaction, based on which waves collide there — which is a system dependent property.

4.2.1 Event is the collision of two discontinuity waves

There is no particular simplification possible in this case, and a full DRS solution must be done. Specifically, the solution of the Riemann problem at the wave interaction location is computed using a standard Riemann solver, and then any rarefaction fans present in the solution are discretized as explained in subsection 4.1. Fortunately these are the less frequent types of events, so the fact that no simplification is possible has little effect on the computational speed.

4.2.2 Event is the collision of two characteristic waves

There are two sub-cases to consider,

- $\mathbf{C}_1$ • The two characteristic waves that collide are from *different* families. That is: a head-on wave collision — see figure 4 (left).
- $\mathbf{C}_2$ • The two characteristic waves that collide are from the *same* family. That is, we have convergence of characteristics, with one wave in the pair overtaking the other — see figure 4 (right). This type of event is the manifestation of the nonlinear steepening, leading to wave breaking and shock formation in the solution.

Case $\mathbf{C}_1$ (See the left panel in figure 4 for the notation).
In this case the solution by the DRS consists of two characteristic waves: (i) W_L^O, belonging to the left family — characteristic speed $\lambda_L = u - c$, and (ii)

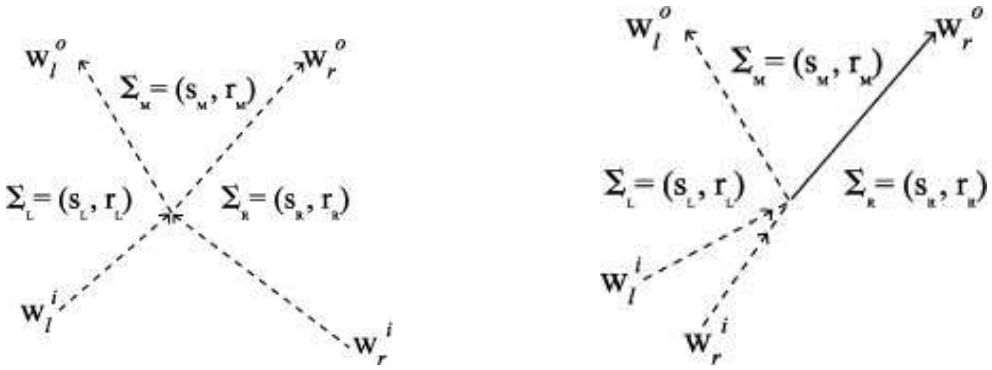

Figure 4: Collisions of characteristic waves event. On the left a head-on collision (case $\mathbf{C}_1$) of characteristics from different families. On the right a convergence collision (case $\mathbf{C}_2$) of characteristics from the same family. Characteristic waves are drawn with dashed lines and discontinuities with solid lines. In each case two input characteristics waves (W_L^i and W_R^i) participate in the interaction, and a pair of output waves (W_L^O and W_R^O) are produced. The states to the left and right of the intersection point are denoted by $\Sigma_L = (s_L,\, r_L)$, and $\Sigma_R = (s_R,\, r_R)$, respectively. The middle state (produced after the collision) is denoted $\Sigma_M = (s_M,\, r_M)$. The method used to assign numerical values to Σ_M and the wave velocities is given in subsubsection 4.2.2. The head-on interaction on the left produces two outgoing characteristic waves (one for each family). The characteristic-convergence on the right produces a discontinuity (shock wave) in the same family as the colliding characteristics, and a characteristic wave in the other family.

W_R^O, belonging to the right family — characteristic speed $\lambda_R = u + c$. Each wave has a velocity which is the average of the corresponding characteristic speeds for the (constant) states on each side. The state in the wedge between the outgoing waves follows from the fact that the right Riemann invariant r should be constant across the left family wave W_L^O, and the left Riemann invariant s should be constant across the right family wave W_R^O. Thus the middle state is given by

$$(s_M, r_M) = (s_R, r_L)\,. \tag{15}$$

We must check that the outgoing waves have strengths below the threshold $\Delta_{\rm DRS}$. This is easy, as the same argument leading to (15) shows that the state in the wedge between the incoming waves must be given by $s = s_L$ and $r = r_R$. Because the incoming waves are numerical characteristics: $|r_L - r_R| \leq \Delta_{\rm DRS}$ and $|s_L - s_R| \leq \Delta_{\rm DRS}$, which proves the point.

We note that this solution is only an *approximate solution*. However, the error is consistent with the order of the method: it is second order in the "size" of the characteristic waves, which is the actual parameter controlling the numerical error, and has a user selected upper bound $\Delta_{\rm DRS}$.

Case $\mathbf{C}_2$ (See the right panel in figure 4 for the notation).
The first thing we must show is that *this situation can only arise in a compressive region of the flow*, showing that the CTM truly represents nonlinear steepening and wave breaking. For definiteness, let us *assume that the converging characteristics belong to the right family* — with characteristic speed $\lambda_R = u + c$ (the other possibility is treated in an entirely similar fashion). Then the left Riemann invariant will be constant throughout the incoming flow: $s_L = s_* = s_R$, where we use the $*$ subscript to indicate values in the wedge between the incoming waves[11]. The convergence of the waves implies that $(\lambda_R)_L + (\lambda_R)_* > (\lambda_R)_* + (\lambda_R)_R$, as follows from the rule for the numerical characteristics propagation speed. That is to say, we must have:

$$(\lambda_R)_L > (\lambda_R)_R \quad \Longrightarrow \quad \rho_L > \rho_R\,, \quad p_L > p_R\,, \quad u_L > u_R\,, \quad \text{and} \quad r_L > r_R\,, \tag{16}$$

where we have used the result in equation (12). This proves the desired result.

In this case the solution by the DRS consists of two waves:

(i) W_L^O, a left numerical characteristic — characteristic speed $\lambda_L = u - c$, and

(ii) W_R^O, a right shock wave.

[11] That is, we have a *simple wave* situation — see equation (12).

Even though W_R^O is a rather weak shock wave, it is rather important that it be designated as so — otherwise the process by which shock waves are born from a smooth flow would not be properly captured by the algorithm.

In order to determine the parameters for the outgoing waves, we note that across the outgoing left characteristic wave, the right Riemann r invariant is constant. Thus $r_M = r_L$. Then, using the Rankine-Hugoniot equation (8) for the outgoing shock, we obtain:

$$(u_R - u_M)^2 = -(\mathrm{v_R} - \mathrm{v_M})\,(\mathrm{p_R} - \mathrm{p_M})\,. \tag{17}$$

But $u_M + l_M = 2\,r_M = 2\,r_L$, so we can write:

$$(l(\mathrm{v_M}) - 2\,\mathrm{r_L} + \mathrm{u_R})^2 + (\mathrm{v_R} - \mathrm{v_M})\,(\mathrm{p_R} - \mathrm{p}(\mathrm{v_M}) = 0\,, \tag{18}$$

where p and l are as in (2) and (11). This is a (scalar) nonlinear equation for $\mathrm{v_M}$, which must be solved numerically. This is easy, since $\mathrm{v_M}$ must be rather close to $\mathrm{v_L}$ — the wave W_L^O is very weak. Once this is done, the rest of the unknown quantities (e.g. the wave speeds) follow easily.

As in case $\mathbf{C}_1$, the DRS *solution presented here is approximate.* The error is, again, consistent with the order of the method. In fact, better than this, since the error originates from the approximation of the W_L^O wave by a numerical characteristic, and W_L^O is very weak, of size $O((\Delta_{\mathrm{DRS}})^2)$.

Remark 1 *Consider the particular case where the value of the right Riemann invariant in the wedge between the incoming waves satisfies either:*

$$r_* \geq r_L \qquad or \qquad r_R \geq r_*\,. \tag{19}$$

Then the wave W_L^O in the (approximate) solution can be eliminated, with the state behind W_R^O given by $(s\,, r) = (s_L\,, r_L)$, and with W_R^O a numerical characteristic (not a shock) — note that $s_L = s_R$, as pointed out earlier. This can be done because the conditions (19) *— together with $|r_L - r_*| \leq \Delta_{\mathrm{DRS}}$,*

$|r_R - r_*| \leq \Delta_{\mathrm{DRS}}$, *and the fact that* $r_L > r_R$ — *guarantee that* $|r_L - r_R| \leq \Delta_{\mathrm{DRS}}$.

On the other hand, $r_L > r_* > r_R$ *can cause the strength of* W_R^O *to exceed* Δ_{DRS}, *and there is no choice but to make* W_R^O *a shock wave.*

We note that the extra simplification provided by the observations in this remark are not too significant, as the number of events where they apply is (typically) rather small.

Beyond this point, rrr has not gone (IN THIS SECTION ONLY).

4.2.3 Collision of a Characteristic Wave and a Discontinuity

In this subsection we consider the third and most complicated CTM wave interaction. This is the situation where one of the colliding waves is a discontinuity wave and the other is a characteristic wave. The method used to propagate the flow forward from the point of intersection depends heavily on the true characteristic structure of the actual wave represented by the discontinuity wave[12] and the direction that each wave is traveling when it collides. Here we consider several representative examples to demonstrate the procedure used in each case.

As a first example, consider a right-facing shock that collides with either a left-facing or a right-facing characteristic wave. The algorithmic procedure performed in either case is the same but for concreteness see figure 5 (left) for a diagram of the situation where the characteristic wave is left facing. As

[12]Types of discontinuous waves considered in this paper include shocks, contact discontinuities, and phase transformation fronts.

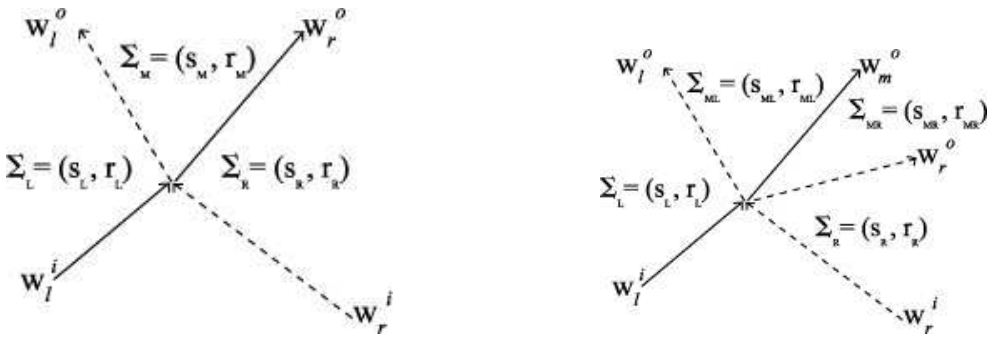

Figure 5: Example CTM waves produced when a characteristic wave and a discontinuity wave collide. The discontinuity wave is a shock (left) and the discontinuity wave is a contact discontinuity (right). Characteristic waves are drawn as dashed lines, discontinuity waves are drawn as solid lines. The head-on collision of a shock and a characteristic produces a shock and a characteristic. The head-on collision of a characteristic and a contact discontinuity produces *three* waves, a left-facing characteristic, a right-facing contact discontinuity, and a right-facing characteristic wave.

motivation, we note that physically the characteristic structure of a right-facing shock is such that the C_+ characteristics converge on the shock from both sides, while the C_- characteristics pass through. At the CTM wave collision, we produce a right-going discontinuity wave (shock) and a left-facing characteristic wave as shown in figure 5 (left). As before we must specify the values assigned to the middle state and speeds to the two CTM waves. In this case the middle state and two CTM wave velocities are constructed exactly as in the case when two characteristic waves of the same family collide, see subsection 4.2.2 for explicit formulas for their calculation.

In this example the true characteristic structure of shocks was used to motivate the choice of CTM waves produced from this intersection. In the next example, the characteristic structure of the entering discontinuity wave allows both left and right-facing characteristics to emanate from the intersection point.

As a second example consider the case that occurs when the discontinuity

wave is a contact discontinuity. For a specific example with a left-facing characteristic wave, see figure 5 (right). The characteristic structure of a contact discontinuity is that the true characteristics of the flow from both families pass *through* the contact discontinuity. When a numerical characteristic wave and a contact discontinuity collide the CTM specifies the production of a left-facing characteristic wave, a right-facing characteristic wave and a central contact discontinuity. This wave structure is shown in figure 5 (right). It remains to specify the values assigned to each of the states. To this end we proceed as follows: across the outgoing left-facing characteristic wave W_l^O, the right-going Riemann invariant is constant, and across the outgoing right-facing characteristic wave W_r^O the left-going Riemann invariant is constant. These two conditions give constraints for the "middle" invariants of

$$r_{ML} = r_L\,, \qquad \text{and} \qquad s_{MR} = s_R\,. \tag{20}$$

Across the contact discontinuity, physically, two equations hold

$$p(l_{ML}) = p(l_{MR})\,, \qquad \text{and} \qquad u_{ML} = u_{MR}\,. \tag{21}$$

When these equations are written in terms of the Riemann invariants (r,s) on either side of the contact discontinuity, we obtain the following two equations

$$p(s_{ML} + r_{ML}) = p(s_{MR} + r_{MR})\,, \tag{22}$$

and

$$u_{ML} \equiv r_{ML} - s_{ML} = r_{MR} - s_{MR} \equiv u_{MR}\,. \tag{23}$$

Equation (23) can be solved for one of the unknown invariants, either s_{ML} or r_{MR}. Here we choose to solve for r_{MR} and substitute this relation into equation (22) obtaining

$$p(s_{ML} + r_{ML}) = p(2\,s_{MR} - s_{ML} + r_{ML})\,. \tag{24}$$

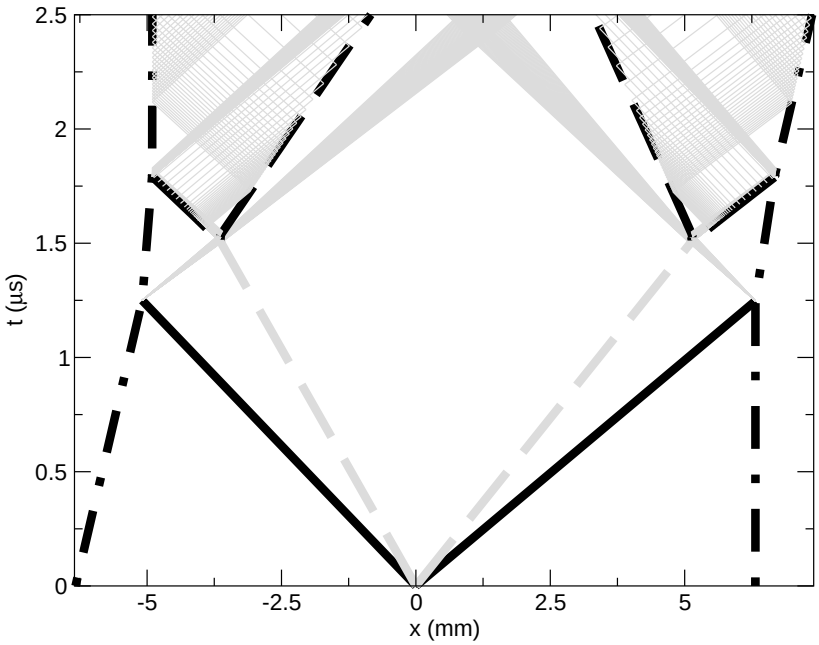

Figure 6: (x, t)-diagram for Barker-Hollenbach experiment number 1.

This is one equation for the single unknown invariant, s_{ML} that can be numerically solved once an equation of state is specified. With s_{ML} determined, the states (in terms of the invariants) between all the waves are now specified. The specification of the wave velocities completes the required algorithmic description. Physically, contact discontinuities travel at the fluid velocity through which they pass, which is equivalent to velocities in either Σ_{ML} or Σ_{MR} since they are restricted to be the same by equation (23). The velocities of each characteristic wave are computed in the same way as they are in the case of a head-on collision of two characteristics,

Other types of discontinuous waves, besides those presented above, are treated in a similar manner. For example, in [51] the characteristic tracking method is used to solve the hyperbolic equations produced in an investigation of high pressure phase transitions in iron. There a different type of wave called a phase-transition wave is found. This phase transformation wave is

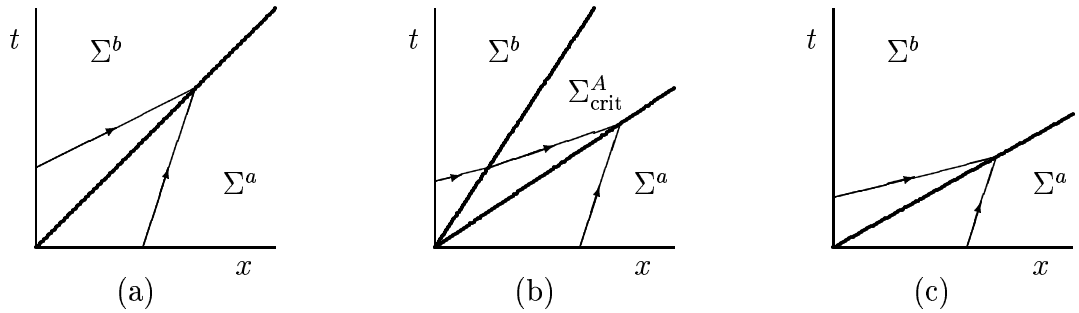

Figure 7: Possible right facing waves, and their characteristic structure for the compressive branch of the Riemann problem arising under an equation of state given by equation (13). The initial states are represented as $\Sigma^a = (\rho_a, u_a)$ and $\Sigma^b = (\rho_b, u_b)$. From left to right, the state Σ^b belongs to: **(a)** The first compression sub-branch; **(b)** The second compression sub-branch; and **(c)** The third compression sub-branch. The trajectories of discontinuity fronts are represented by thick lines and the C_+ characteristics by thin lines. This figure was borrowed with permission from [**?**].

a type of "under-compressed" shock wave. The characteristic structure of these waves is such that the characteristics of the same family as the wave converge onto it from behind but not from ahead (see figure 7 (b)). In this paper we will not develop the mathematical equations used in handling this complication but will present all results for waves of this type including how to change the waves type as the flow evolves in another paper [52].

In the next section we present algorithm results using the CTM. The results show the assertions made in the introduction of this paper. Namely, the resolution of shocks, contact discontinuities, and phase transformation fronts to *one* node of resolution. Also shown will be the single point resolution of rarefaction fan edges and boundaries between simple waves.

5 Algorithm Results

As stated earlier, the CTM was developed and (first) implemented for a hyperbolic system that models martensitic phase transformations under shock

loading — see [3, 4, 51] and subsection 3.2. In these type of situations, a phase transformation is triggered in a material by the passage of a very strong[13] shock wave. The wave is produced by impacting the sample material with a rapidly moving plate — see "flyer plate experiments" in [9]. Typically: a cylindrical sample of (say) iron is impacted on one end by a very fast moving cylinder (impactor) of (say) copper. Then the position of the other end of the sample cylinder is recorded as a function of time (this is the data provided by the experimental measurements). The challenge is to predict this measured position in terms of a theoretical description of what happens inside the sample.

In the "simplest" model (where a one dimensional approximation is made, and only longitudinal motions along the cylinder axes are considered), the mathematical description of the problem to be solved is as follows:

1. A mathematical model for the behavior of the materials (both sample and impactor) is given. In our case: the p-System in section 3, with the equations of state in subsection 3.2, as well as an equation of state for the impactor material.

2. Initial data for the model are provided. Time $t = 0$ corresponds to the moment of impact, when: (i) The pressure is constant (ambient) throughout both the sample and the impactor. (ii) The velocity is identically zero on the sample (to the right of the impact point, with the coordinates we use). (iii) The velocity is identically constant on the impactor (to the left of the impact point), and equal to the velocity the impactor cylinder had right before impact.

3. Boundary data are provided at the right end of the sample and at the left end of the impactor. These are modeled as contact discontinuities, with vacuum on the other side.

[13] Strong in the sense that the pressure behind the shock wave is enormous relative to the (ambient) pressure ahead. As it turns out, as measured by the Mach numbers, the wave is not so strong, with Mach numbers relative to the flows ahead and behind satisfying: $M_0 < 1.7$ and $M_1 > 0.6$.

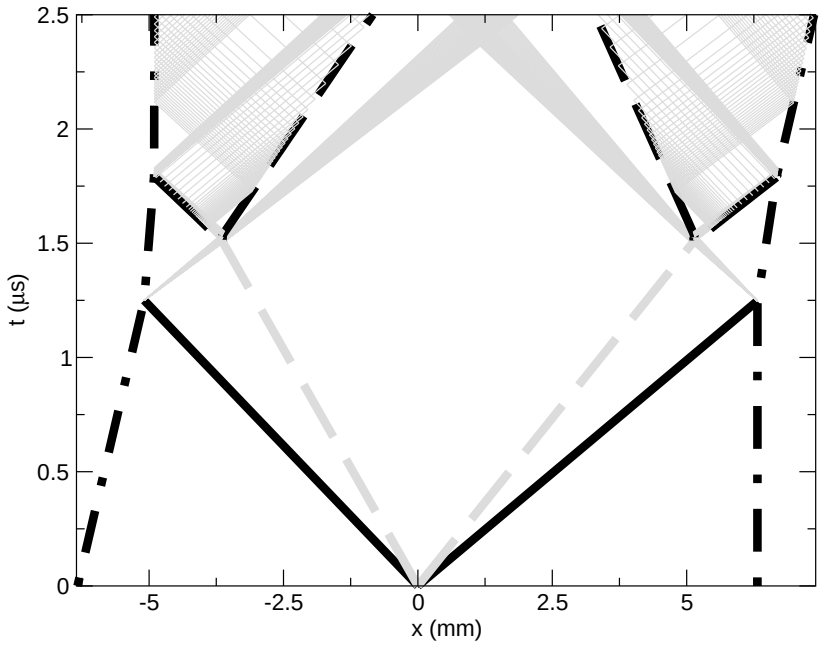

Figure 8: Space-time diagram for the Barker-Hollenbach experiment number 1, where phase transitions involving α-iron and ϵ-iron are triggered by shock loading (see description in the text). The horizontal (space) axis is in mm and the vertical (time) axis is in μs. The keys used to represent the various types of waves that arise in this computation can be found in equation (25).

4. Finally, the problem is solved and a prediction for the position of the right end of the sample is made (which can be matched against the experiments). This involves resolving all the waves that result from the impact, as well as their interactions.

For the purpose of illustrating how the CTM works in practice, in this section we present some of the numerical results obtained in our research on martensitic phase transformations under shock loading. To be precise, we will show the results of computations modeling the situation in the "first impact experiment" of Barker and Hollenbach [1]. This particular computation was selected because in it all the various types of wave interactions that the

characteristic tracking method is designed to deal with occur. We note that, in addition to shocks, contact discontinuities, and numerical characteristics, phase transformation fronts arise. These involve either: (1) Transitions from austenite (α-iron) to martensite (ϵ-iron), which we call *forward transformation fronts*; and (2) Transitions from martensite to austenite, which we call *backward transformation fronts*. We will not describe the details of these w
st

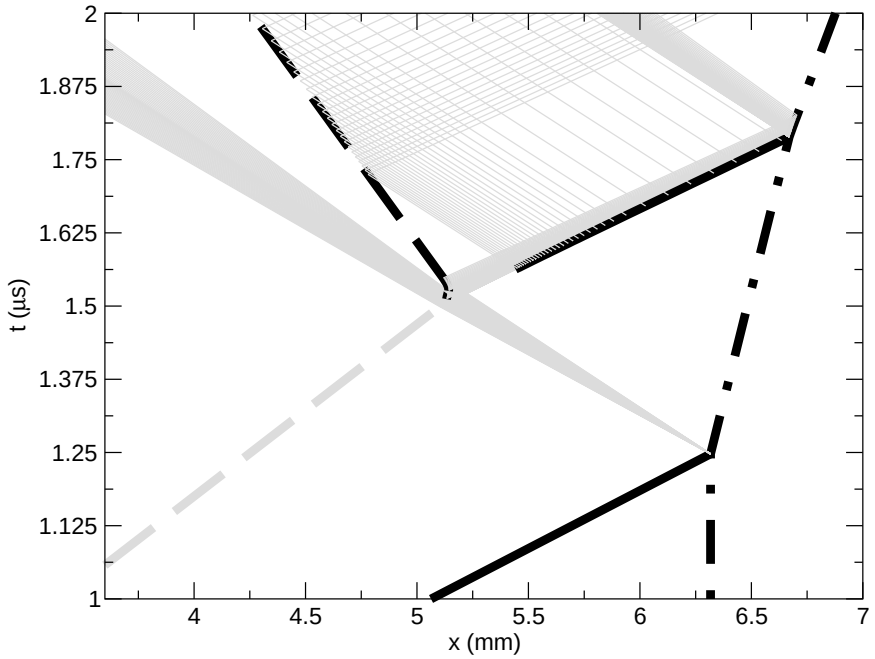

Figure 9: Magnification of a region in the space-time diagram for the Barker-Hollenbach experiment in figure 8. This region shows the interaction of a left rarefaction fan with a forward transformation front. The horizontal (space) scale is in mm, and the vertical (time) scale is in μs. The notation conventions in equation (25) are used.

Figure 8 presents a space-time diagram showing all the wave interactions that occur in this experiment. Both numerical characteristic waves and discontinuous (shocks, contacts and phase transformation fronts) waves are

shown. The following notation conventions for the line markings showing the various waves are used in this figure, and the ones that follow it:

1. Shocks are printed with *thick solid black* lines.
2. Forward transformation fronts are printed with *thick dotted gray* lines.
3. Contact discontinuities are printed with *thick dot-dashed black* lines.
4. Backwards transformation fronts are printed with *thick long dashed* lines.
5. Numerical characteristic waves are printed with *thin gray* lines.
6. Constant state regions are in pure *white*.

(25)

In figure 8, the locations and widths of several centered rarefaction fans and compressive simple waves can be clearly seen. Since a most interesting effect takes place when a reflected[14] fan interacts with a right forward transformation wave, we show this region magnified in figures 9 and 10, and spend some time discussing the waves there.

In figure 9 we see that the interaction of the rarefaction fan and the forward transformation front produces several waves. A left and right simple waves are produced, along with a backwards phase transformation. The left simple wave is expansive, and the right simple wave is compressive. Both show up in the calculation as groups of left (or right) numerical characteristic waves. Note that regions where the solution is constant are shown in pure white. A net, important, *result of all these interactions is to "turn around" the phase transformation wave, from a right forward transformation front into a left backward transformation front. Thus, the phase transformation never reaches the right edge of the sample.*

In figure 11 we see C^1 discontinity correspoinding to changes in simple wave types.

It is interesting to observe that the right simple wave (compressive) breaks

[14]From the right end of the sample, as a result of a shock interacting with the contact discontinuity there.

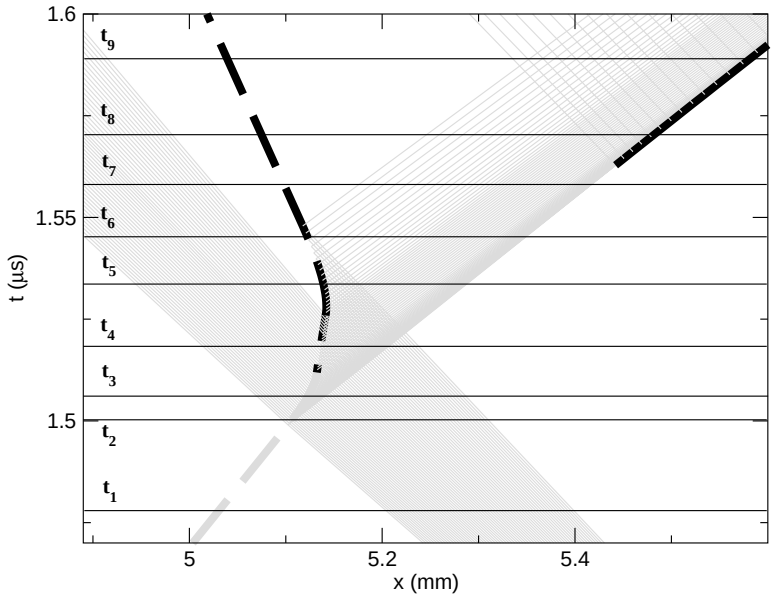

Figure 10: Further magnification of the interaction region shown in figure 8. The horizontal lines indicate the times at which the pressure profiles shown in figures 12 through 16 occur. The interaction of a left rarefaction fan with a forward transformation front is shown. The horizontal (space) scale is in mm, and the vertical (time) scale is in μs. The notation conventions in equation (25) are used.

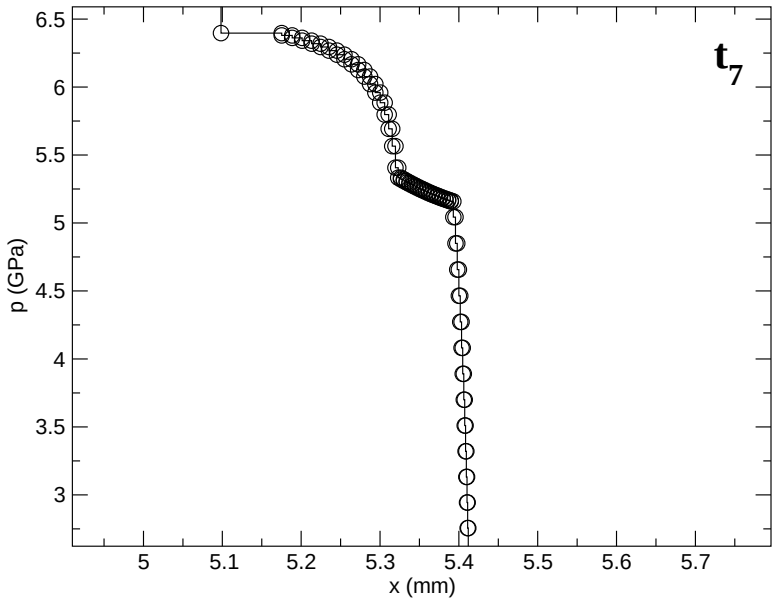

Figure 11: Blow up of C^1 discontinuities inside the simple waves. The top arched portion from $5.25 \leq t \leq 6.5$ is compressive, the middle portion from $5.0 \leq t \leq 5.25$ is expansive, the bottom portion from $3.0 \leq t \leq 5.0$ is compressive again. It is the codes behavior with corners I want to emphasize more.

on its way to the free surface, and a shock is produced — this is clearly seen in figure 9. This shock first starts at the crossing of two right numerical characteristics,[15] and then grows in strength as more right numerical characteristics converge towards it. At each interaction of the shock with a right numerical characteristic, a (rather weak) reflected left numerical characteristic is produced — these can also be clearly seen in figure 9. The rates of change of the state variables are directly related to how close the numerical characteristics are; thus the "grey areas" in the figure are regions where rapid

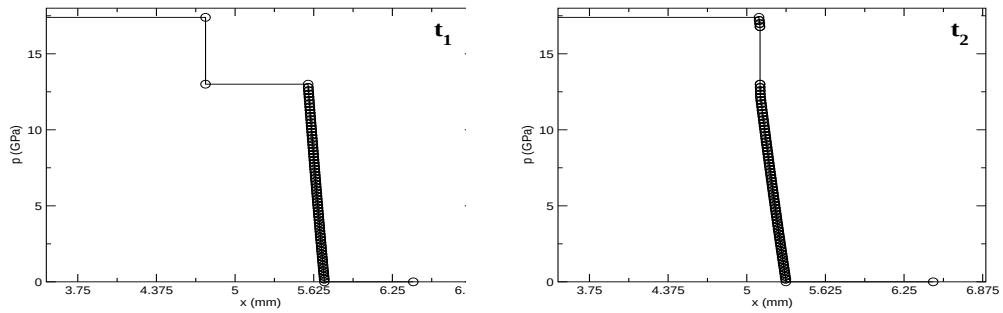

Figure 12: Pressure plots for the interaction of a rarefaction fan with a phase transformation front. The times correspond to the horizontal lines in figure 10. Times $t = t_1$ on the left and $t = t_2$ on the right. The horizontal space-scale is in mm, and the vertical pressure-scale is in Giga-Pascals.

Finally, in order to show the details of the solutions (and how they are captured by the code), we show the pressure profiles (p as a function of x) near the interaction region for ten different values of time. The times selected are indicated in the space-time diagram of figure 10 by horizontal (constant time) lines. The pressure profiles are shown in figures 12 through 16.

(t_1) In the plot for time $t = t_1$ we see the left rarefaction fan approaching the right forward phase transformation front.

(t_2) In the plot for time $t = t_2$ the two waves have just collided, and a

[15] See subsection 4.2.2, case $\mathbf{C}_2$.

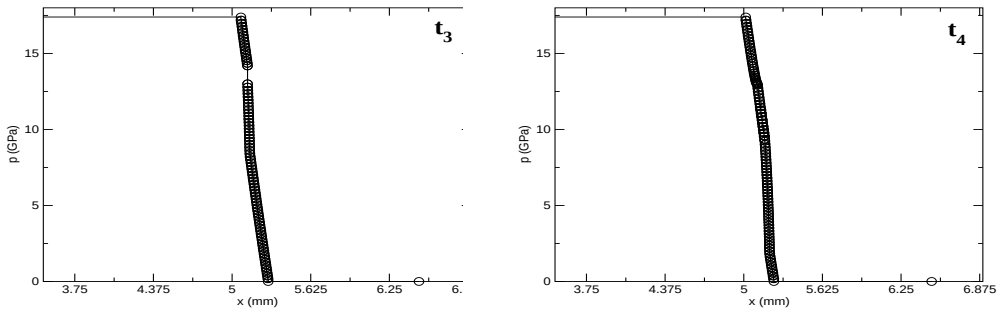

Figure 13: See figure 12 caption. Times $t = t_3$ on the left and $t = t_4$ on the right.

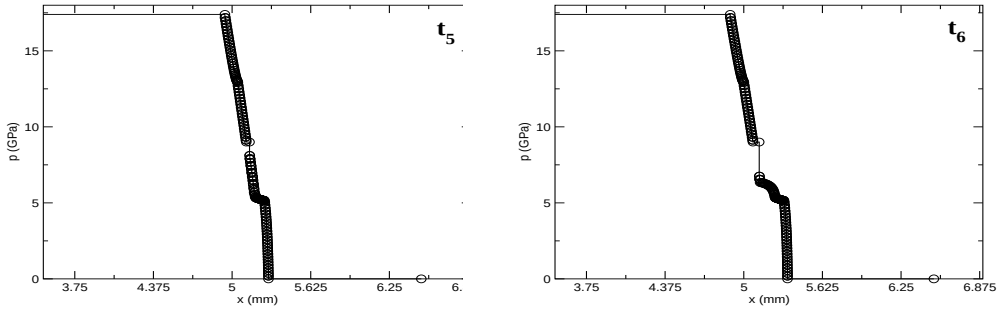

Figure 14: See figure 12 caption. Times $t = t_5$ on the left and $t = t_6$ on the right.

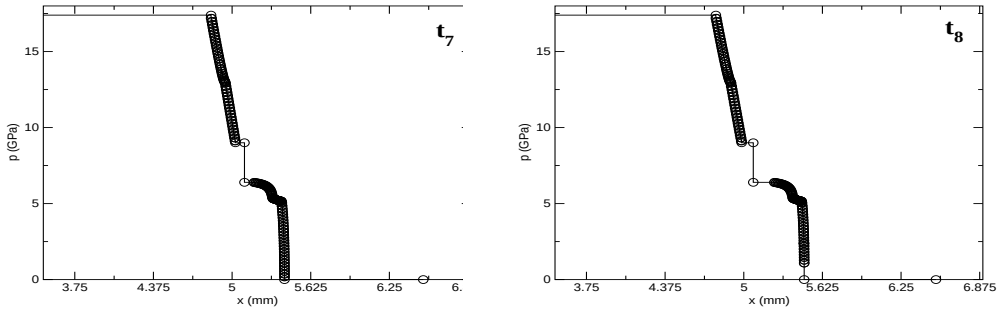

Figure 15: See figure 12 caption. Times $t = t_7$ on the left and $t = t_8$ on the right.

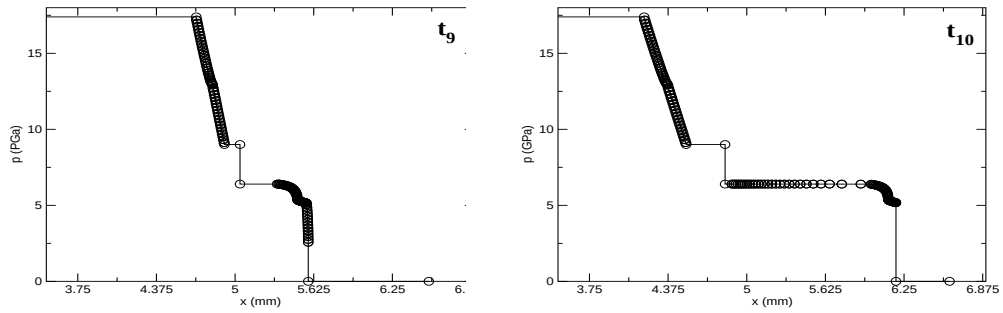

Figure 16: See figure 12 caption. Times $t = t_9$ on the left and $t = t_{10}$ on the right.

bit of the rarefaction fan has been transmitted through the forward transformation front.

(t_3) In the plot for time $t = t_3$ much more of the rarefaction fan can be seen behind the forward phase transformation font. Further, the forward phase transformation has a much diminished magnitude, and we can see a reflected right compressive simple wave beginning to show — this as a slight change in the slope of the p versus x curve ahead of the transformation front.

(t_4) In the plot for time $t = t_4$ the right forward transformation front has weakened to the point that the jump in the pressure has vanished. Thus the front has become a contact discontinuity between the two phases. This is the time at which the transformation front changes type, and reverses direction.

(t_5) In the plot for time $t = t_5$ the contact discontinuity seen in time $t = t_4$ has evolved into a weak left backwards transformation front, while the reflected right simple wave from the interaction in beginning to steepen.

(t_6) In the plot for time $t = t_6$ the process described for time $t = t_5$ continues.

(t_7) In the plot for time $t = t_7$ the right simple compressive wave continues its steepening.

(t_8) At time $t = t_8$ the right simple compressive wave *breaks*, and a weak right shock appears.

(t_9) In the plot for time $t = t_9$ one can clearly see: (i) The two components of the transmitted simple wave; (ii) A fully developed left backwards phase transformation front; and (iii) The three components of the reflected simple wave, namely: two compressive fans converging into a right shock wave.

(t_{10}) The plot for time $t = t_{10}$ corresponds to a time slightly before the newly developed right shock strikes the free surface (right end of the sample).

6 Conclusions

In this paper we developed the Characteristic Tracking Method (CTM) a numerical method for integrating nonlinear hyperbolic conservation laws in one dimension. The CTM was developed and demonstrated for a specific instance of the p-System encountered during the investigation of high pressure phase transitions in metals [3, 4, 51]. There the CTM was shown to produce very accurate approximations to the solutions — which include interactions of shocks, rarefaction fans, compressive simple waves and phase transformation fronts. Work is currently underway to generalize and apply this algorithm to other systems of interest (such as the Euler equations of gas dynamics).

References

[1] L. M. Barker and R. E. Hollenbach. Shock wave study of the $\alpha \rightleftharpoons \epsilon$ phase transition in iron. *J. Appl. Phys.*, 45:4872–4887, 1974.

[2] J. P. Boris and D. L. Book. Flux corrected transport I. SHASTA, a fluid transport algorithm that works. *J. Comp. Phys.*, 11:38–69, 1973.

[3] O. P. Bruno and D. D. Vaynblat. Shock-induced martensitic phase transitions: critical stresses, Riemann problems and applications. *Proceedings of the Royal Soc. of London. Series A. Mathematical and Engineering Sciences*, 457(2016):2871–2920, 2001.

[4] O. P. Bruno and D. D. Vaynblat. Two-wave structures in shock-induced martensitic phase transitions. *Mathematical and Computer Modeling*, 34(12-13):1261–1271, 2001.

[5] P. Colella. A direct Eulerian MUSCL scheme for gas dynamics. *SIAM J. Sci. Stat. Comput.*, 6:104–177, 1985.

[6] P. Colella and P. R. Woodward. The piecewise parabolic method (PPM) for gas-dynamical simulations. *J. Comput. Phys.*, 54:174–201, 1984.

[7] R. Courant and K. O. Friedrichs. *Supersonic Flow and Shock Waves.* Interscience Publishers, Inc., New York, 1948.

[8] R. Courant, E. Isaacson, and M. Reeves. On the solution of nonlinear hyperbolic differential equations by finite differences. *Comm. Pure Appl. Math.*, 5:243–255, 1952.

[9] D. S. Drumheller. *Introduction to Wave Propagation in Nonlinear Fluids and Solids.* Cambridge University Press, Cambridge, U.K., 1998.

[10] W. E. and H. Yang. Numerical study of oscillatory solutions of the gas-dynamic equations. *Stud. Appl. Math.*, 85:29–52, 1991.

[11] B. Engquist and S. Osher. Stable and entropy satisfying approximations for transonic flow calculations. *Mathematics of Computation*, 34:45–75, 1980.

[12] S. K. Godunov. A difference scheme for the numerical computation of a discontinuous solution of the hydrodynamic equations. *Math. Sbornik*, 47:271–306, 1959.

[13] A. Harten. The artificial compression method for the computation of shocks and contact discontinuities. III. self-adjusting hybrid schemes. *Math. Comput.*, 32:363–389, 1978.

[14] A. Harten. High resolution schemes for hyperbolic conservation laws. *J. Comp. Phys.*, 49:357–393, 1983.

[15] A. Harten. On a class of high resolution total-variation-stable finite-difference schemes. *SIAM J. Numer. Anal.*, 21:1–23, 1984.

[16] A. Harten. ENO schemes with subcell resolution. *J. Comput. Phys.*, 83:148–184, 1989.

[17] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *J. Comput. Phys.*, 71:231–303, 1987.

[18] A. Harten and J. M. Hyman. Self-adjusting grid methods for one-dimensional hyperbolic conservation laws. *J. Comp.Phys.*, 50:235–269, 1983.

[19] A. Harten and S. Osher. Uniformly high-order accurate nonoscillatory schemes. I. *SIAM J. Numer. Anal.*, 24:279–309, 1987.

[20] A. Harten, S. Osher, B. Engquist, and S. Chakravarthy. Some results on uniformly high order accurate essentially non-oscillatory schemes. *Appl. Numer. Math.*, 2:347–377, 1986.

[21] G. Jiang and C. W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202–228, 1996.

[22] P. D. Lax. Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Comm. Pure Appl. Math.*, 7:159–193, 1954.

[23] P. D. Lax. Hyperbolic systems of conservation laws II. *Comm. Pure Appl. Math.*, 10:537–566, 1957.

[24] P. D. Lax and B. Wendroff. Systems of conservation laws. *Comm. Pure Appl. Math.*, 13:217–237, 1960.

[25] P. D. Lax and B. Wendroff. Difference schemes for hyperbolic equations with high order of accuracy. *Comm. Pure Appl. Math.*, 17:381–398, 1964.

[26] B. Van Leer. Flux vector splitting for the Euler equations. In *Proc. 8th International Conference on Numerical Methods in Fluid Dynamics*. SIAM–AMS, Springer Verlag, 1982.

[27] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser Verlag, Basel, 1992.

[28] X. D. Liu, S. Osher, and T. Chan. Weighted essentially non–oscillatory schemes. *J. Comput. Phys.*, 115:200–212, 1994.

[29] R. W. MacCormack. The effect of viscosity in hypervelocity impact cratering. *AIAA Journal*, 69:354–365, 1969.

[30] K. R. Meadows, D. A. Caughey, and J. Casper. Computing unsteady shock waves for aeroacoustic applications. *15th Aeroacoustics Conference*, AIAA-93-4329:1–11, 1993.

[31] R. Menikoff and B. J. Plohr. The Riemann problem for fluid flow of real materials. *Rev. Mod. Phys.*, 61:75–130, 1988.

[32] J. Von Neumann and R. D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *J. Appl. Phys.*, 21:232–237, 1950.

[33] S. Osher. Shock modeling in aeronautics. In K. W. Morton and M. J. Baines, editors, *Numerical Methods for Fluid Dynamics*, pages 179–218. Academic Press, London, 1982.

[34] S. Osher and R. P. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer-Verlag, New York, N. Y., 2002.

[35] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comp. Phys.*, 79:12–49, 1988.

[36] S. Osher and F. Solomon. Upwind difference schemes for hyperbolic systems of conservation laws. *Math. Comput.*, 38:339–374, 1982.

[37] P. L. Roe. Approximate Riemann solvers, parameter vectors and difference schemes. *J. Comp. Phys.*, 43:357–372, 1981.

[38] P. L. Roe. The use of the Riemann problem in finite difference schemes. In *Lecture Notes in Physics*. Springer Verlag, Berlin, 1981.

[39] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press, Cambridge, U.K., 1999.

[40] C. W. Shu. Essentially non–oscillator and weighted essentially non–oscillatory schemes for hyperbolic conservation laws. ICASE Report No. 97–65 NASA/CR-97-206253 97–65, NASA Langley Reasarch Center, NASA Langley Reasarch Center, 1987.

[41] C. W. Shu and S. Osher. Efficient implementation of essentially non–oscillatory shock capturing schemes. *J. Comput. Phys.*, 77:439–471, 1988.

[42] C. W. Shu and S. Osher. Efficient implementation of essentially non–oscillatory shock capturing schemes II. *J. Comput. Phys.*, 83:32–78, 1989.

[43] J. Smoller. *Shock Waves and Reaction–Diffusion Equations*. Springer, New York, N.Y., 1983.

[44] J. L. Steger and R. F. Warming. Flux vector splitting of the inviscid gas–dynamic equations with application to finite difference methods. *J. Comp. Phys.*, 14:361–370, 1982.

[45] B. van Leer. Towards the ultimate conservative difference scheme II. Monotinicity and conservation combined in a second order scheme. *J. Comput. Phys.*, 14:361–370, 1974.

[46] B. van Leer. Towards the ultimate conservative difference scheme IV. A new approach to numerical convection. *J. Comput. Phys.*, 23:276–299, 1977.

[47] B. van Leer. Towards the ultimate conservative difference scheme V. A second order sequel to Godunov's method. *J. Comput. Phys.*, 32:101–136, 1979.

[48] D. Vaynblat. *The Strongly Attracting Character of Large Amplitude Nonlinear Resonant Acoustic Waves Without Shocks. A Numerical Study*. PhD dissertation, MIT, Department of Mathematics, Aug 1996.

[49] J. L. Weatherwax. The characteristic tracking method as applied to general non–linear hyperbolic conservation laws. In preparation.

[50] J. L. Weatherwax. The characteristic tracking method as applied to the Euler equations of gas dynamics. In preparation.

[51] J. L. Weatherwax, D. D. Vaynblat, O. P. Bruno, and R. R. Rosales. An investigation of proposed higher order effects in a model for martensitic phase transformations. In preparation.

[52] J. L. Weatherwax, D. D. Vaynblat, O. P. Bruno, and R. R. Rosales. Modifications to the characteristic tracking method to account for phase transitions. In preparation.

[53] G. B. Whitham. *Linear and Nonlinear Waves*. Wiley–Interscience, New York, N.Y., 1974.

[54] P. R. Woodward and P. Colella. The numerical simulation of two dimensional fluid flow with strong shocks. *J. Comput. Phys.*, 54:115–173, 1984.