

SHOCK INDUCED SURFACE INSTABILITIES AND NONLINEAR WAVE INTERACTIONS

Brian Boston John W. Grove* Richard L. Holmes

Abstract

We discuss front tracking simulations of shock reflections and shock accelerated interfaces. Some key features of the front tracking method are the elimination of numerical diffusion and the reduction of wall heating. In computations of the regular Mach reflection of a shock at an oblique ramp, we see enhanced resolution of the primary waves in the interaction. In addition, tracking allows very precise measurements to be made of the states and location of the Mach triple point. Our computations of the growth rate of a Richtmyer-Meshkov unstable interface are the first numerical results that are in quantitative agreement with experimental results of a shocked air-SF₆ interface. Previous attempts to model the growth rate of the instability have produced values that are almost twice that of the experimental measurements. Moreover, the failure of the impulsive model, and the linear theory from which it is derived, to model experiments correctly is understood in terms of time limits on the validity of the linear model.

1. Introduction

In this article we present results of simulations using front tracking combined with a second order Godunov finite difference method. Two classes of problems are discussed, the oblique reflection of shock waves at ramps, and the computation of the instability growth rate of a perturbed, shock-accelerated interface.

Supported in part by the U.S. Army Research Office through the Mathematical Sciences Institute of Cornell University under subcontract to SUNY Stony Brook, ARO contract number DAAL03-91-C-0027.

Supported in part by the National Science Foundation Grant no. DMS-9201581

Supported in part by the National Science Foundation Grant no. DMS-9057429

Supported in part by the U. S. Army Research Office, grant no. DAAL03-92-G-0185.

Keywords : Front tracking, Mach reflection, Richtmyer-Meshkov

Our code achieves excellent resolution of the simulated flows, even on the relatively coarse grids used here. In both cases our computed results are shown to be in excellent agreement with experiments. Indeed, we present computations of the Richtmyer-Meshkov instability that for the first time agree with experimentally measured growth rates of interface perturbations.

The ramp reflection simulations model the interaction of a planar shock wave with an oblique wall. We are interested in determining the structure of the reflection process for ramp angles that are very close to the mechanical equilibrium condition for bifurcation to regular reflection, as defined by the coincidence of regular and Mach reflection. The use of front tracking allows us to conduct numerical experiments that are extremely close to this point. Ordinary shock capturing methods are unable to resolve the Mach triple point configuration in this regime due to the extreme closeness of this point to the wall. We measure several quantities for the reflection process, including the trajectory of the triple point, and the Mach number of the flow behind the foot of the Mach stem. Comparison of these quantities to experiment shows a good agreement between our values and the experimentally measured quantities.

The Richtmyer-Meshkov instability concerns the growth of interface perturbations on a shock accelerated material interface. When a shock wave collides with the interface between two different materials, small perturbations of this interface grow into nonlinear structures having the form of "bubbles" and "spikes". The occurrence of this shock-induced instability was predicted by Richtmyer [18] and confirmed experimentally by Meshkov [15]. The Richtmyer-Meshkov instability is similar to the more familiar Rayleigh-Taylor instability and is important in both natural phenomena (supernovae) and technological applications (inertial confinement fusion).

Theory and computation have so far failed to provide an understanding of the Richtmyer-Meshkov instability that is in quantitative agreement with existing experiments [3, 4, 7, 17, 19]. Computations of the Richtmyer-Meshkov instability for singly shocked, sinusoidally perturbed interfaces have over-predicted growth rates by factors from 40% to 100% [7] as compared to experiments. The

main theoretical model used in this area, Richtmyer's impulsive model [18], also consistently predicts a growth rate that is too large.

Our computations of the Richtmyer-Meshkov instability are further validated by a comparison of small amplitude perturbation, early time simulations with solutions to a linearization of the equations of motion. An analysis of the time interval of validity for the linearized model explains the failure of the linearized and impulsive models to agree with experiment.

2. The Front Tracking Method

Front tracking is a computational method for the sharp resolution of a set of distinguished waves in a flow. It combines a standard, rectangular grid based finite difference method with a set of lower dimensional, dynamically moving grids that follow the tracked wave fronts. A general description of this method, including an outline of the structure of our computer program, is given in [12].

The numerical solution for flows in two space dimensions is represented on the union of a rectangular grid and a set of piecewise linear curves. The state at each point on the rectangular grid represents the cell average over the corresponding cell of the dual grid centered at that point. The solution at a point on a tracked front is multivalued, with values corresponding to the limits of the solution on either side of the wave. The numerical representation of the flow explicitly includes the jump discontinuities across the tracked waves and thus eliminates numerical diffusion.

Points of intersection between tracked waves, called nodes, correspond to two-dimensional interactions between wave fronts. An important example of such a node in these computations is the Mach triple point.

A global solution operator for the evaluation of the state of the flow at arbitrary locations is constructed from a front-limited triangulation of the computational grid and the tracked fronts. This triangulation is constrained so that no triangle crosses a tracked front. A side of an individual triangle in this construction is either a rectangular lattice cell boundary, or an edge on a tracked

front. A corner of such a triangle is thus either a grid cell corner, a point on a tracked front or an intersection of a tracked front with a lattice cell boundary. The states at these points serve as data for a linear interpolant of the solution into the interior of the triangle.

The representation of the solution in our front tracking code differs from the more standard triangular representations of a flow in that the tracked waves are dynamic and move with time so that the triangulation must be regenerated at each time step in the computation. The method also differs from the unstructured finite volume techniques in that the main solution is computed using a regular, rectangular grid. Subsequently the states at grid points that lie within the domain of influence of the tracked waves over the time step must be corrected to account for the presence of these waves. It is important to note that the front tracking code combines both front tracking with shock capturing, so that secondary waves (such as breaking shocks or induced slip lines) are resolved within the ability of the underlying mesh spacing and the finite difference scheme.

The propagation operator that updates the numerical solution over a single time step consists of three basic parts: propagation of the tracked wave fronts (point propagation), propagation of points of interaction between tracked waves (node propagation), and update of the states on the rectangular grid (interior solver). For the latter operation, the fronts at the beginning and end of the time step serve as internal boundaries for the regions adjoining those waves.

The first propagation phase consists of the propagation of the non-nodal points on the tracked waves. At each tracked point a local rotation of coordinates is performed that aligns the coordinate axes with the normal and tangential directions of the curve at that point. The tangent to a piecewise linear curve at a point is defined as the line through that point which is parallel to the secant through the neighboring points. Operator splitting is used to divide the front propagation into two one-dimensional units: a normal propagation step and a tangential propagation step. The normal propagation of the tracked waves is computed using a second order Riemann problem-type method as de-

scribed in [6]. This operator solves a piecewise linear Cauchy problem and is similar to the van Leer [20] flux computation as used in the second order Godunov method. The tangential operator uses a one-dimensional finite difference method, which in our code is precisely the same as the interior finite difference scheme. The tangential stencil at a given point on the partially propagated wave is formed by projecting the adjacent curve states onto the tangent at that point. The interested reader is referred to references [6, 12] for a more detailed description of these operations.

An interaction between tracked waves is locally approximated by a two dimensional Riemann problem, which is defined as the Cauchy problem for initial data that is scale invariant with respect to the node position at the start of the time step, i.e. constant along rays from the node position. The numerical solution is computed using shock polar analysis. References [10, 11] contain a description of the node propagation algorithm. The primary nodes of interest in the present calculations are the Mach triple point and the diffraction node formed during the Richtmyer-Meshkov calculation by the intersection of a tracked shock wave with the material interface.

The interior finite difference scheme that computes the solution on the rectangular grid is an operator split, second order MUSCL scheme [2, 8]. Our implementation uses a five point stencil with linear reconstruction.

The rectangular grid update consists of two passes: a regular and an irregular grid update. First, the finite difference equations are solved for the rectangular grid alone, ignoring the tracked fronts. The second pass then updates the states at the rectangular grid points near the tracked fronts. If a tracked front crosses the finite difference stencil of a rectangular grid point during the time step, the states computed at that location by the first interior sweep must be discarded and recomputed to account for the presence of the tracked wave. In our implementation we use a locally modified stencil at each such point to compute the updated solution. We start with copies of the states on a five point stencil centered at the given location. We then find the tracked fronts, if any, on either side of the stencil center that are closest to the middle location. We

replace any states in the stencil lying on the opposite side of the nearest front by copies of the state on the correct side of the front at the stencil crossing, as computed by linear interpolation between the tracked points on the curve. Thus the final solution never uses finite differencing across tracked waves.

The organization of the interior solver into two passes allows the bulk of the code for this part of the computation to be vectorized. Note that even though our computations used a nonvector machine (Sun Sparcstation2), this organization provides a substantial improvement in performance on vector machines and adds little additional overhead on nonvector computers.

3. Shock Wave Reflection at Ramps

Before going to the results of our simulations of Mach reflections, we include a description of the initialization and propagation algorithms. Our computation is initialized just after the shock crosses the ramp corner, when the reflected wave bubble is just a few mesh blocks in height. Tracking the complete reflected shock bubble requires an estimate of the initial geometry of this object to serve as a seed for the final computed configuration. We use the following technique to install the initial reflected waves. These procedures have also been described in [10], and all relevant equations may be found in [9].

We assume the Mach stem is initially normal to the ramp and short enough to be modeled as a single, straight line segment. The Mach trajectory angle χ is then determined algebraically by the condition that the turning angle through the incident and reflected waves, in a frame that moves with the Mach triple point, is the same as the turning angle through the Mach stem, i.e. the flow behind the configuration must be parallel to the slip line on both sides. The data for this system of equations consists of the angle of the Mach stem and the incident shock data (ahead and behind states). This system is solved using a root-finding iteration on χ , which determines the local states and wave angles about the triple point. After guessing a value for χ , we can determine a node velocity using the angle of the incident shock. Once this transformation to the

steady frame is known, it is a straightforward application of shock polar theory to determine the states behind the reflected and Mach shocks. If these two flows are not parallel, we modify χ and recompute.

Figure 1 shows a schematic of the basic geometry of the reflection. The Mach stem and contact are installed as straight line segments from the triple point to the wall, at the computed angles. The position of the bow of the reflected shock behind the ramp is found by solving a head-on reflection Riemann problem for the incident shock. The shape of the initial reflected wave is composed of two pieces. The first is a straight piece at the triple point tangent to the computed reflected wave, and the second an ellipse from the end of this segment to the bow point, with minor axis and center on the bow wall. Depending on the resolution, this ellipse can often degenerate to a single line segment. It is important to note that this construction is only performed once at the beginning of the computation. The subsequent propagation of the Mach triple point and the bow node uses only the local information at each of these points. In particular there is no restriction that the Mach stem remain straight, or that the reflected wave have any particular shape. These properties are determined dynamically by the computation. In fact, the shapes of the waves at later times appear to be independent of any reasonable initial configuration.

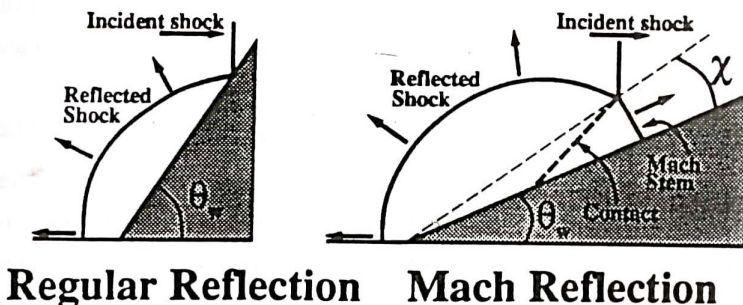


Fig. 1. A schematic of regular and Mach reflection. Here, χ is the Mach node trajectory angle, and θ_w is the ramp angle. The incident shock is moving from the left to the right.

The propagation of the Mach node for subsequent times requires a different procedure, as we can no longer assume the Mach stem is straight. As before, we use the incident shock information – ahead state and behind pressure. The piece of data that replaces the angle of the Mach stem is the pressure behind the Mach stem. This pressure is obtained by solving a non-local Riemann problem across the Mach stem, as described in the discussion of the front tracking algorithm.

Once again, the problem lies in finding the node velocity. After this is known, the computation of the new states at the node is straightforward, as in the initialization. The basic relation is

$$\tan^2(\theta) = \frac{(p_1 - p_0)}{\rho_0 q_0^2} \left(\frac{\rho_0 q_0^2}{m^2} - 1 \right). \quad (3.1)$$

This relation is applied across the incident, reflected and Mach shocks, where ρ_0 is the density ahead of the shock, q_0 the steady flow speed ahead of the shock, m is the mass flux, p_0 and p_1 the pressures ahead and behind respectively, and θ the turning angle across the shock. This gives three equations, one each for the incident, reflected and Mach shocks, in thirteen variables. The ahead state and all pressures are assumed known, which gives the mass fluxes and the densities. We then have enough information to compute the turning angles across the incident and Mach shocks, and the turning angle across the reflected wave is eliminated by forcing the behind flow to be parallel on both sides of the slip line. The only remaining unknowns are the steady flow speeds ahead and behind the incident. We eliminate the latter by applying the following across the incident shock:

$$q_0^2 - q_1^2 = (p_1 - p_0)(1/\rho_0 + 1/\rho_1). \quad (3.2)$$

This system of equations is solved numerically for q_0 , which gives the node speed since the ahead velocity is known. The angle of the node is found by propagating the incident shock normally at the node (via a non-local Riemann problem) and intersecting it with a circle of radius q_0 centered at the old node position. We then have the complete node velocity (magnitude and direction) and can use the shock polar equations to finish the computation of the states and wave angles at the new node. The other waves are then attached to the

new triple point, making sure to enforce the computed angles of each.

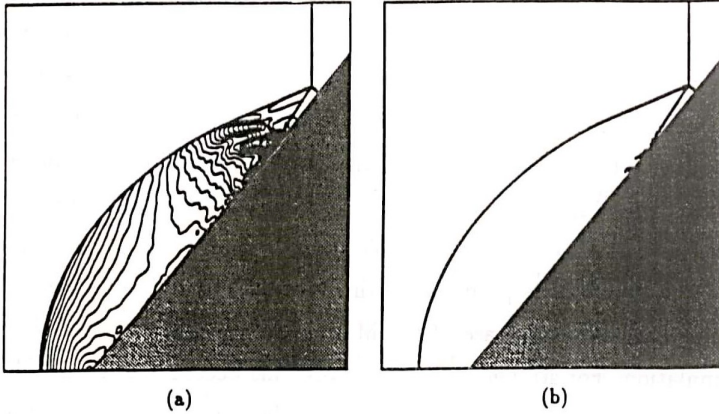


Fig. 2. (a) Density contours for the single Mach reflection run, using $\theta_w = 46^\circ$, $p_0 = 0.3$ bars, $T_0 = 300^\circ K$, and $M_0 = 2.0$. The Mach triple point trajectory angle $\chi = 2.557^\circ$. (b) The tracked wave fronts from the same computation.

The first wall reflection simulation we present consists of a shock wave in air (modeled as a perfect gas with $\gamma = 1.4$) colliding with an oblique ramp. The gas ahead of the shock wave is at an ambient pressure of 0.3 bars and a temperature of $300^\circ K$, with an incident shock Mach number of 2.0. The ramp angle is $\theta_w = 49^\circ$. A computational grid of 256×256 zones was used, and the run was conducted on a Sun Microsystems Sparcstation2 with 64 megabytes of RAM, taking approximately 20 CPU hours to complete. Our investigations have shown that grids as coarse as 100×100 would resolve the basic structure (front locations), and much of the interior structure of the reflected shock bubble.

Figure 2 shows the results of our computation of the simulation outlined above. Figure 2a shows density contours, while Figure 2b shows only the tracked wave fronts. The density difference between adjacent contours is 4% of the density of the unshocked air. We see very sharp resolution of the tracked waves, and in particular at the triple point. This resolution becomes even more important as we move closer to the bifurcation point to regular reflection, which we will

discuss more fully in our last set of Mach reflection simulations.

There is a considerable amount of activity in the flow at the point where the slip line induced from the Mach triple point reaches the wall. The slip boundary conditions at the wall require that the flow there be parallel to it, and hence there is a large gradient in the velocity as the flow adjusts from being parallel to the slip line to being parallel to the wall. Tracking the slip line reduces the amount of numerical spreading of this wave at the wall, which in turn enhances the resolution of the flow about the triple point by preventing its contamination by the transient waves produced at the boundary. In Figure 2, we see the compression band near the base of the Mach stem just beginning to sharpen in this simulation. For stronger incident waves, this becomes an actual shock, and forms what is known as the double Mach reflection, with two triple point configurations.

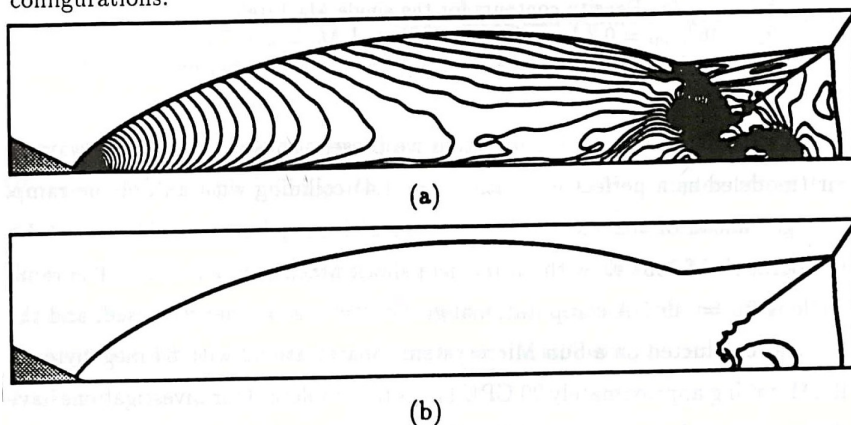


Fig. 3. (a) Density contours for the double Mach reflection. Here $\theta_w = 30^\circ$, the ahead state is at standard temperature and pressure, and $M_0 = 10.0$. (b) The tracked wave fronts from the same computation.

Our next simulation was chosen to demonstrate this double Mach configuration, and is intended to match a simulation done by Colella and Berger [5]. The gas is once again air, but the state ahead of the incident is at standard temperature and pressure. The incident shock is much stronger, with a Mach number of 10, and makes an angle of 60° with the ahead wall, which translates

to $\theta_w = 30^\circ$. Our computational grid was 300×60 zones, which is three times finer than the base grid in [5]. However, we use no automatic mesh refinement, and our grid is 25% coarser than their finest level of refinement. The results are presented in Figure 3.

Once again, the resolution of the tracked wave fronts is very good, and there are some new features here that were not present in the previous run. The compression band generated at the base of the slip line has sharpened into an actual shock, which is resolved by the interior finite difference solver. There is no slip line seen at this second triple point, but such waves are typically very weak. No such wave is visible in [5] either.

The kink in the reflected wave is very well defined despite the fact that the shock producing it is spread over three to four mesh blocks. The angle of the reflected wave on either side of the kink could be accurately measured from the picture alone. A feature of our code, however, is the ability to measure various physical quantities very precisely. For example, we know the exact states around the triple point or at the base of the Mach stem, or the exact angle of the reflected wave at any point on the curve, by simply reading them from the data structures in the computation. Examples of this are given by the density profiles along the reflected wave and lower wall, shown in Figure 4. From these, we can measure the strength of the secondary shock and the slip line.

We also see very nice roll-up of the contact at the wall as it is swept along behind the Mach stem. For a grid this coarse, such resolution is simply not possible without some sort of enhancement such as front tracking or the adaptive mesh refinement used in [5]. The roll-up would otherwise be lost due to the secondary shock and the hot spot where the contact begins to roll back under itself.

Let us mention again that we are not enforcing any global structure on the tracked fronts. The kink in the reflected wave, the roll-up in the slip line, and the forward bow in the Mach stem are all being produced solely by the numerics in our code.

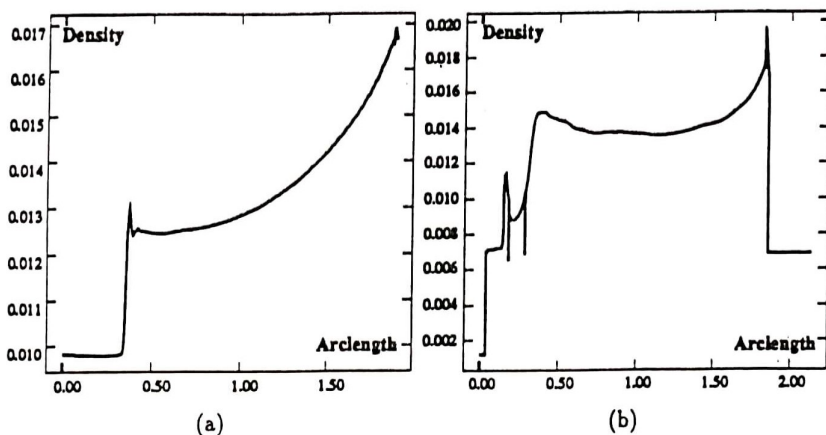


Fig. 4. Arclength versus density in the double Mach reflection run (a) along the reflected shock, triple point to bow, and (b) along the lower wall and ramp, from right to left.

The density contours of both runs show that our computation is doing a good job of reducing the effect of wall heating at the ramp boundary. Most of the contours are relatively smooth going into the boundary, which is consistent with the inviscid model used for these computations. We made no attempt to model the boundary layer effects that are present in a real experiment. Notice also that the stagnation point on the ramp is captured very nicely by the interior solver, as seen by the sparseness of the contours near this region.

In a separate series of runs, we further applied front tracking to the study of Mach reflections by comparing them with experiments performed at the University of Sydney by Henderson and Virgona [14]. Considerable effort was made to simplify these experiments in order to facilitate comparison with numerical results. For example, the Mach reflections were generated in argon ($\gamma = 1.667$) so as to eliminate vibrational non-equilibrium, dissociation, and chemical reactions. The strength of the incident shock i was sufficiently large to ensure that the flow downstream of the reflected shock r was supersonic, but not so strong as to ionize the argon. More precisely, the average strength of i used in the experiments, or rather the average inverse strength $\xi_i \equiv p_0/p_1$ was $\xi_i = 0.1534$,

corresponding to an incident Mach number of 2.327. The ahead state pressure and temperature were $p_0 = 14.1 \pm 3.0$ kPa and $T_0 = 293.15 \pm 4.0^\circ$ K. The Mach reflections were generated by diffracting incident shocks over a series of symmetrical wedges of different apex semi-angles θ_w (Figure 5). This design eliminated shock-boundary layer interaction at the apex of every wedge. The more conventional concave corner model (Figure 1) undergoes significant shock-boundary layer interaction as the reflected shock r sweeps back over the lower wall. The newer model eliminates this effect, and is mathematically equivalent to the model in Figure 1 in the inviscid case.

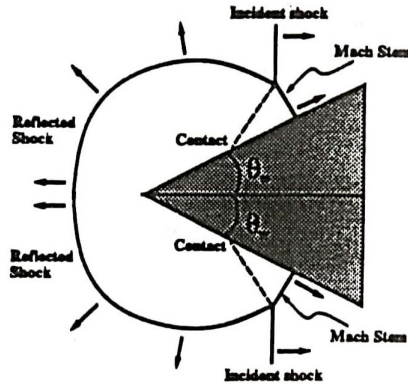


Fig. 5. A schematic representation of the experimental used to produce Mach reflections. This configuration produces a pair of symmetric reflections as the incoming shock passes the apex of the wedge.

A graph of the Mach triple point trajectory angle versus the ramp angle is shown in Figure 6a. These figures show the experimentally measured value of χ together with the values computed by our front tracking code, as well as those computed by Colella [14] using a highly resolved shock capturing scheme. This measurement was attractive because χ cannot be computed from the shock polar analysis used to compute the local configuration at the node. Its value is entirely a result of the interaction of the numerics in our code, and its experimental value is particularly robust. We have also found the Mach number behind the base of

the Mach stem to be a very useful measurement, for exactly the same reasons.

As can be seen from the figure, our results agree very well with both the experimental and the shock capturing results. We note that both sets of computational results are on the high side of the experimental values. This difference has been attributed to boundary layer effects.

An application of important interest is the transition conditions between regular and Mach reflection at a wall. It is well known that for certain flow regimes there is an overlap between the regions in phase space where regular and Mach reflections are possible. Both experimental and computational investigations have shown that the boundary layer at the wall plays an important role in the process that selects the type of wave produced by the wall reflection. In order to quantify the effect of the wall boundary layer it is important to understand the inviscid limit of the solution where this boundary layer is absent. It is in the computation of this inviscid limit that a major strength of the front tracking method is revealed. Since we are explicitly tracking the most singular parts of the calculation, we can make very precise statements about exactly where a given discontinuity is located; there is no numerical diffusion of the fronts. This allows us to perform computations near the transition to regular reflection, yielding structure that is simply not resolvable in either experiment or standard shock capturing codes. We can resolve the full Mach triple point configuration for angles χ as small as 0.1° (Figure 6b). In such simulations the Mach stem is less than a grid block long. By contrast, shock capturing codes generally lose the resolution of the Mach triple point when the length of Mach stem is less than two or three grid blocks. This loss of resolution is due to the presence of a numerical boundary layer at the wall, so that the entire structure is contained in a smear of contours. This regime is also difficult to approach experimentally due to the real viscous boundary layer at wall. Currently, front tracking appears to be the only method that can conduct numerical simulations of inviscid wall reflections to within a small fraction of a degree of the mechanical equilibrium condition.

Wall Angle vs Mach Node Trajectory

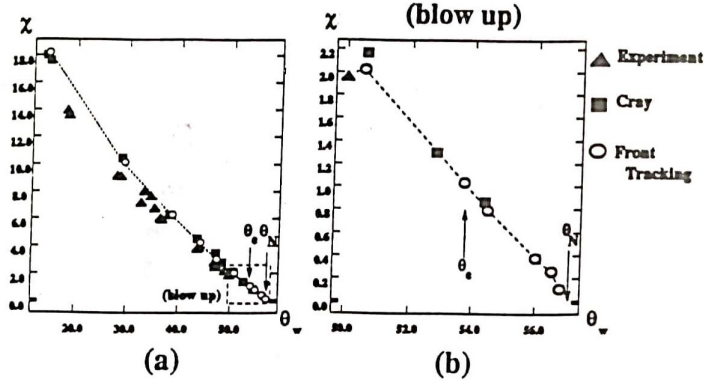


Fig. 6. A comparison of the Mach triple point trajectory for experiments, fine grid shock capturing computations, and front tracking. We note that both numerical methods are in substantial agreement with each other, and are close to the experimental measurements. Front tracking provides approximately the same value for χ as the shock capturing code, using only a fraction of the grid resolution.

The resolution for the front tracking runs was achieved on grids which are much coarser than those used in standard finite difference simulations of this problem. Most of our grids were 100×100 , up to about 150×150 close to the transition point. This also gives a commensurate savings in time – the runs took between one and four hours.

It should also be pointed out that for the region between θ_e and θ_N , our code can simulate either regular or Mach reflection – both are theoretically possible in this region. However, based on the shape of the curve outside this region (see Figure 6b), our results definitely seem to converge to the point θ_N , and there is no reason to expect a discontinuity in the curve at θ_e . This would require a corresponding pressure jump for a shock on a curved ramp, for example, which smoothly passes through the transition point. Such a wave has never been observed experimentally. We feel that this is a very strong statement

that in this parameter regime, the bifurcation from Mach Reflection to regular reflection takes place at the mechanical equilibrium condition and not at the detachment point.

4. Numerical Simulation of the Richtmyer-Meshkov Instability

We focus on the simplest case of the shock tube experiments of the Richtmyer-Meshkov instability where a sine shaped material interface is accelerated by a single shock wave, as in the experiments of Meshkov [15], Benjamin [3, 4], and others. The general configuration of the computation and experiments is shown in Fig. 7. A thin membrane was used in the experiments to separate the two gases at the material interface. Quantitative agreement was achieved between our computational results and the experimental measurements of Benjamin [4] for the rate of growth of a shocked air-SF₆ interface [13]. The collision results in a transmitted shock and a reflected wave that can be either a shock or a rarefaction depending on the values of the fluid parameters. The experiments considered in this paper are of the reflected shock type. Viscosity and heat conduction are negligible here, so the fluid motion is described by the Euler equations.

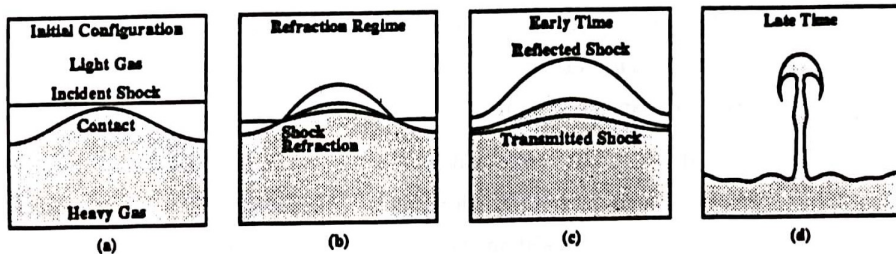


Fig. 7. A schematic representation of the geometry of the Richtmyer-Meshkov instability modeled in this paper. The interaction consists of the collision of a shock wave with a material interface. The refraction of the shock by the interface produces reflected and transmitted waves. The instability consists of the growth of perturbations of the material interface with time.

The impulsive model proposed by Richtmyer [18] is commonly used to estimate the growth rate of a shock accelerated interface. This model is derived by assuming that the shock acceleration can be treated as being impulsive, and that the flow is nearly incompressible once the shock wave has passed through the material interface. It is also assumed that the flow is observed in a frame where the average position of the material interface is at rest, and the position, $y(x, t)$, of the material interface at time t can be given by $y(x, t) = a(t) \sin kx$, where k is the wave number of the perturbation. Richtmyer's formula gives the growth rate of $a(t)$ as

$$\dot{a}(t) = k \Delta u \frac{\rho_1 - \rho_2}{\rho_1 + \rho_2} a(0+), \quad (4.1)$$

where Δu is the difference between the shocked and unshocked mean interface velocities, the ρ_i are the post-shocked densities on the two sides of the interface (the incident shock moves from material "2" to material "1"), and $a(0+)$ is the perturbation amplitude immediately after the collision of the shock with the material interface. This formula implicitly assumes the initial preshocked amplitude, $a(0-)$, is small compared to the wavelength.

Given that $a(0-)$ is small so that $ka(0-) \ll 1$, a more exact calculation of the amplitude growth rate can be made. The Euler equations are linearized around the solution of a one dimensional Riemann problem defined by the head-on collision of a planar shock with a zero amplitude (planar) material interface, using the initial amplitude of the sinusoidal perturbation as a small expansion parameter. The result of the linearization is a system of partial differential equations in one spatial dimension with associated boundary conditions. This system can be solved numerically for the growth rate of the perturbed interface. This approach, following Richtmyer [18], has recently been generalized to include reflected rarefactions as well as reflected shocks [21]. Simple order of magnitude estimates limit the validity of the linearized equations to the dimensionless time interval

$$t_{\min} \equiv ka(0-) \ll t_* \ll 1/[ka(0-)] \equiv t_{\max}. \quad (4.2)$$

Here the dimensionless time $t_* = kc_0 M_0 t$, where M_0 is the incident shock Mach

number, and c_0 is the sound speed of the fluid ahead of the incident shock. The limits t_{*min} and t_{*max} represent respectively the transit time of the incident shock through the perturbed interface and the time required for the perturbation to grow to unit amplitude. Necessarily, these time limits apply to the derivation of the impulsive model as well, since it is an approximation to the linear theory. Recent systematic comparisons of the impulsive model and the linear theory have revealed both regions of agreement and of disagreement in parameter space [21].

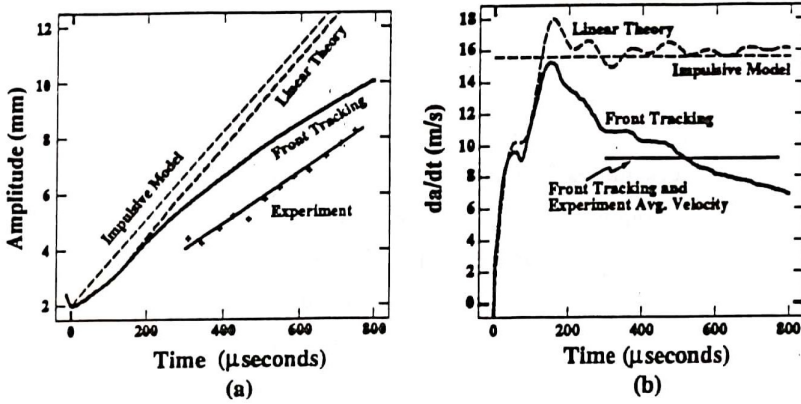


Fig. 8. Perturbation amplitude, $a(t)$, and amplitude growth rate, $\dot{a}(t)$, of a shocked air-SF₆ interface. This graph compares the results of experimental averages, front tracking simulation, linear theory and Richtmyer's impulsive model. Also shown are results of a least squares fit to the front tracking amplitude data over the period of experimental observation. The plus marks (+) show the results of one particular experiment, while the experimental growth rate represents an average over several experiments.

We compared our simulations of a singly shocked air-SF₆ interface to the experiments of Benjamin [4]. The material interface is accelerated by a shock wave with Mach number 1.2 moving from air into SF₆. The initial amplitude, $a(0-)$, was 0.00637 times the period of the sinusoidal perturbation. For these experiments, $t_{*max} \approx 2.5$, while the observational time interval is $15 \leq t_{*observational} \leq 50$. The observational times and the validity of the linear theory fail to overlap

by a factor of about 6. We conclude that the linear theory has no relationship to this experiment.

Fig. 8 shows plots of the amplitude and amplitude growth rate of the material interface as obtained from experiment, the front tracking simulation, the linearized theory, and Richtmyer's impulsive model. The time axis in these figures is shifted so that $t = 0$ corresponds to the time at which the shock wave has completed its refraction through the interface.

As can be seen from these figures the front tracking results are in substantial agreement with the experimental results in the sense that the growth rate derived by a least squares analysis of the front tracking amplitude data, $9.2m/s$, is identical to the same quantity derived from the experimental data. Note that for late (i.e., experimentally observed) times the linearized theory and the impulsive model growth rates are a factor of two larger than those found in experiment or in our simulation. This may be due to the fact that this particular configuration has a relatively large initial amplitude and quickly leaves the region of validity of the linearized theory and impulsive model. The displacement of the experimental curve with respect to the front tracking curve is possibly due to membrane effects, i.e. the material strength of the membrane or the influence of its fragmentation may affect the fluid flow.

The front tracking results indicate a decay in amplitude growth rates while Benjamin [4] finds a fairly constant growth rate during the measurement period. Other experiments and simulations, however, have shown a decaying growth rate [1, 16, 17]. Mesh refinement studies show that the decay rate is independent of the mesh. We also tested our simulation against changes in other numerical parameters and found that the value of $\dot{a}(t)$ was insensitive to these changes. These parameters include artificial viscosity and the frequency of redistribution of the points on the tracked interfaces. Since the fluid between the interface and the reflected and transmitted shocks is moving rather slowly after the interaction, a transfer of momentum from the fluid near the expanding interface into the slower moving regions is to be expected. This transfer of momentum would result in a reduced growth rate. In addition, some sort of

decay in amplitude growth rates also seems reasonable as the interface develops highly nonlinear “mushroom caps”. We conclude that this decay is a real effect and not due simply to numerical dissipation as has been suggested [4]. Models of momentum transfer from the interface to surrounding regions are being studied and we hope these will provide a quantitative explanation of the decaying growth rate.

In addition to the amplitude growth rate it is instructive to compare translational velocities of the interfaces. Benjamin has found the mean interface speed, defined as the average of the speeds of the peak and the trough, to be 85 m/s . Solution of the 1-d Riemann problem, i.e. an unperturbed interface, gives a velocity of 66.8 m/s . Our simulations show a fairly constant velocity of $70.5 \pm 0.5\text{ m/s}$ after $t = 300\mu\text{s}$. Benjamin says that the discrepancy between experimental and 1-d velocities could be an indication of diffusion across the membrane before shock acceleration. In some simple tests of the effects of diffusion we ran 1-d unperturbed interface simulations using a linear density profile between the air and SF_6 instead of a discontinuous interface. Various transition lengths were used and the interface velocity was measured. None of our tests showed an appreciable increase in translational velocity. An understanding of the difference in expected and measured interface velocities would be very useful, as it may provide clues to the still unresolved discrepancy in interface amplitudes.

A further validation of the nonlinear simulations can be accomplished by comparison to the small amplitude theory (Fig. 9). This serves both to determine the range of validity of the linear theory and to validate the solution of the full Euler equations at small amplitudes. As can be seen in Fig. 9, the front tracking calculation is converging to the linear result as we reduce the amplitude. We note that the interval of convergence of the nonlinear simulations to the linear theory appears to be finite. This is in contrast to formula 4.2 which suggests that the domain of validity of the linearized equations should increase with decreasing initial amplitude. This point deserves further study.

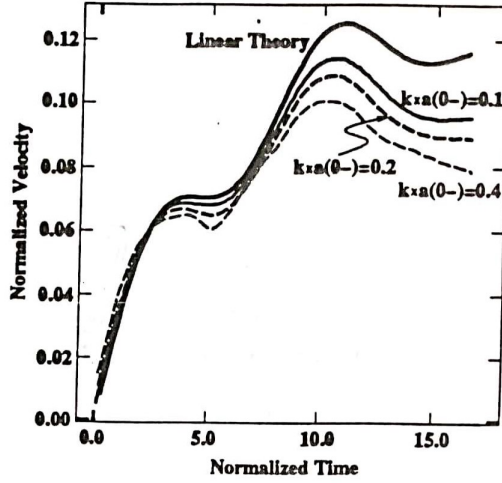


Fig. 9. The convergence of the nonlinear simulations to the linearized solution for small amplitudes. A comparison of three separate calculations of the normalized perturbation growth rate, $\dot{a}(t)/[kc_0M_0a(0-)]$, of a shocked air- SF_6 interface with three different initial amplitudes where k is the wave number, c_0 is the sound speed ahead of the incident shock, and M_0 is the incident shock Mach number. The horizontal axis is in dimensionless time units kc_0M_0t .

Of interest is the question of why our results agree with experiment while results found through other numerical methods do not. Prior disagreement between the growth rates measured in experiments and those predicted by numerical simulation has led to the suggestion that mass diffusion and membrane effects may have an important role in the behavior of the interface instabilities. Our work does not exclude this possibility, but the agreement of our computations with experiment suggests that a proper numerical resolution of the material interface is essential to obtain agreement with experiment, and also that if other effects are important, they may be offsetting one another. It is also clear that there is still much to learn about the highly nonlinear aspects of the Richtmyer-Meshkov instability. These effects include the possible coupling between nonlinear modes, and their study will require experiments on singly shocked interfaces as well as computations with random interfaces which

have been run to late times. Similarly, understanding the effects of reshocking remains an important theoretical challenge. For the single mode case, a systematic study of mass diffusion, membrane effects, and a detailed comparison to earlier calculations of others would be desirable.

Acknowledgments

We thank James Glimm for his encouragement and guidance. We also thank Dave Sharp, Yumin Yang and Qiang Zhang for many helpful discussions during the course of this research.

References

- [1] A. N. Aleshin, E. G. Gamalli, S. G. Zaitsev, E. V. Lazareva, I. G. Lebo, and V. B. Rozanov. Nonlinear and transistional stages in the onset of the Richtmyer-Meshkov instability. *Sov. Tech. Phys. Lett.*, 14:466–468, 1988.
- [2] J. Bell, P. Colella, and J. Trangenstein. Higher order Godunov methods for general systems of hyperbolic conservation laws. *J. Comput. Phys.*, 82:362–397, 1989.
- [3] R. Benjamin. Experimental observations of shock stability and shock induced turbulence. In W.P. Dannevik, A.C. Buckingham, and C.E. Leith, editors, *Advances in Compressible Turbulent Mixing*, pages 341–348. National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd. Springfield VA 22161, 1992.
- [4] R. Benjamin, D. Besnard, and J. Haas. Shock and reshock of an unstable interface. LANL report LA-UR 92-1185, Los Alamos National Laboratory, 1993.
- [5] M. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *J. Comp. Phys.*, 82:64–84, 1989.

- [6] I-L. Chern, J. Glimm, O. McBryan, B. Plohr, and S. Yaniv. Front tracking for gas dynamics. *J. Comput. Phys.*, 62:83–110, 1986.
- [7] L. D. Cloutman and M. F. Wehner. Numerical simulation of Richtmyer-Meshkov instabilities. *Phys. Fluids A*, 4:1821–1830, 1992.
- [8] P. Colella. A direct Eulerian MUSCL scheme for gas dynamics. *SIAM Journal on Computing*, 6:104, 1985.
- [9] R. Courant and K. Friedrichs. *Supersonic Flow and Shock Waves*. Springer-Verlag, New York, 1976.
- [10] J. Glimm, C. Klingenberg, O. McBryan, B. Plohr, D. Sharp, and S. Yaniv. Front tracking and two dimensional Riemann problems. *Adv. Appl. Math.*, 6:259–290, 1985.
- [11] J. Grove. The interaction of shock waves with fluid interfaces. *Adv. Appl. Math.*, 10:201–227, 1989.
- [12] J. Grove. Applications of front tracking to the simulation of shock refractions and unstable mixing. *J. Appl. Num. Math.*, 14, 1994.
- [13] J. Grove, R. Holmes, D. H. Sharp, Y. Yang, and Q. Zhang. Quantitative theory of Richtmyer-Meshkov instability. *Phys. Rev. Lett.*, 71(21):3473–3476, November 1993.
- [14] L. F. Henderson, P. Colella, and R. J. Virgona. Strong shock reflection in pseudo-stationary flow, 1993.
- [15] E. E. Meshkov. Instability of a shock wave accelerated interface between two gases. *NASA Tech. Trans.*, F-13:074, 1970.
- [16] E. E. Meshkov. Instability of shock-accelerated interface between two media. In W.P. Dannevik, A.C. Buckingham, and C.E. Leith, editors, *Advances in Compressible Turbulent Mixing*, pages 473–503. National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd. Springfield VA 22161, 1992.

- [17] K. A. Meyer and P. J. Blewett. Numerical investigation of the stability of a shock-accelerated interface between two fluids. *Phys. Fluids*, 15:753–759, 1972.
- [18] R. D. Richtmyer. Taylor instability in shock acceleration of compressible fluids. *Comm. Pure Appl. Math*, 13:297–319, 1960.
- [19] V. Rupert. Shock-interface interaction: current research on the Richtmyer-Meshkov problem. In K. Takayama, editor, *Shock Waves, proceedings of the 18th international symposium on shocks waves*. Springer-Verlag, New York, 1992.
- [20] 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.
- [21] Y. Yang, Q. Zhang, and D. H. Sharp. Small amplitude theory of Richtmyer-Meshkov instability. *Physics of Fluids A*, 6(5), 1856–1873, 1994.

Department of Applied Mathematics and Statistics,

State University of New York at Stony Brook,

Stony Brook, NY 11794-3600

Email: boston@ams.sunysb.edu, grove@ams.sunysb.edu,

holmes@ams.sunysb.edu