# Visualization, Extraction and Quantification of Discontinuities in Compressible Flows.

Ravi Samtaney[1]

NAS-99-002
March 1999

**Abstract**

Scientific visualizations of two-dimensional compressible flow of a gas with discontinuities are presented. The numerical analogue to experimental techniques such as schlieren imaging, shadowgraphs, and interferograms are discussed. Edge detection techniques are utilized to identify the discontinuities. In particular, the zero crossing of the Laplacian of a field (usually density) is recommended for extracting the discontinuities. An algorithm to extract and quantify the discontinuities is presented. To illustrate the methods developed in this paper, the example chosen is that of an unsteady interaction of a shock wave with a contact discontinuity.

[1]MRJ Technology Solutions Inc., NASA Ames Research Center, Moffett Field, CA 94035-1000. email: samtaney@nas.nasa.gov

# 1 Introduction

The inviscid flow of a compressible fluid is governed by a system of hyperbolic conservation laws (which are also called the compressible Euler equations) [1]. It is only in exceptional and rather rare circumstances that these nonlinear partial differential equations allow a closed form analytical solution. In most situations, and for almost all problems of practical importance, these equations have to be solved numerically.

It is well-known, that for nonlinear systems of hyperbolic conservation laws with $C^\infty$ Cauchy data, the solution may develop discontinuities in a finite time. Examples include the formation of a shock on a wing in transonic flight or the formation and propagation of a shock wave from compressive piston motion. The most common discontinuities which develop in gas dynamics are: (a) shock waves and (b) contact-discontinuities. In the theory of hyperbolic conservation laws, shocks are called genuinely nonlinear waves while, contact discontinuities are called linearly degenerate waves. Numerically, the discontinuities are handled by "shock-capturing" techniques which typically smear the discontinuities over several grid cells [2]. Furthermore, with grid refinement, the physical extent of the smeared shock reduces in extent, while the number of grid cells over which it is smeared still remains the same for a given numerical method. Consequently, although the derivatives of various field quantities (such as the density or the pressure) are ill-defined, the captured discontinuities in the numerical solution exhibit large gradients over a very small spatial extent, and a numerical evaluation of the derivatives is permitted. The reader is reminded that there are other types of discontinuities in compressible flows such as detonations which will not be discussed in this paper.

There are scant instances of visualizations of flow fields with discontinuities in the scientific visualization literature. Noteworthy efforts in shock wave visualization include the work of Pagendarm and Seitz [3], Ma *et al.* [4] and Lovely and Haimes [5]. Most of the discussion in literature pertains to shock wave detection in *steady* three-dimensional flow fields. Some of these shock-detection algorithms rely on the gradients of the density field and isosurfaces of unit Mach number. This works because the Mach number (denoted by $M$) changes from greater than one (supersonic flow) to less than one (subsonic flow) across a shock. However, this criterion ($M = 1$) is not useful for unsteady flows. We do note that Lovely and Haimes [5] have provided correction terms in their algorithm for unsteady flows. Sev-

eral visualization algorithms which assume at least a continuous field (if not continuity of several derivatives) run into unexpected problems. While the visualization community has not paid enough attention to flows with discontinuities, the experimental literature is full of beautiful instances of shock-wave visualizations. The earliest scientific work on shock-wave visualization is due to Toepler who developed the schlieren method; followed by Dvorak, one of Mach's assistants who modified the schlieren method to give the shadowgraph method [6].

In this paper, we review filtering functions which may be used on the numerical solution to generate visual images which correspond to experimental techniques. The main focus of this paper is to go beyond generating pictures of flow fields with discontinuities and to present quantifications of shock waves and contact-discontinuities in the flow. Many details of the quantification algorithm are included.

We will consider the interaction of shock waves with contact discontinuities in two dimensions as a canonical problem which is unsteady and exhibits several interesting features of the discontinuities including bifurcations. Shown in Fig. 1 is a schematic depiction of the physical problem. A shock wave of Mach number $M$ translating from left to right encounters an interface initially inclined at an angle $\alpha$ separating two gases. The gas on the left (right) has a density $\rho_1$ ($\rho_2$). At the interface, the shock wave refracts and bifurcates into a transmitted shock and a reflected wave which may be a shock or an expansion wave. Further reflections of these waves at the top and bottom boundaries and secondary interactions lead to a complex flow field rich in discontinuities. A 3-tuple $(M, \rho_2/\rho_1, \alpha)$ defines the principal parameters in this interaction. For results shown in this paper, we will use uniform meshes with square cells. This is not a restriction as the proposed techniques can be extended to body-fitted curvilinear meshes.

The sample case for which results are shown in this paper will now be described. The principal parameters defining the interaction are $(2.0, 3.0, \pi/4)$ and the ratio of specific heats of the gases is $\gamma = 1.667$. The domain of simulation is: $[-0.5, 1.5] \times [0, 1.0]$, and is discretized by a uniform mesh with 800 points in the $x$-direction and 400 points in the $y$-direction. Unless specified in the figure caption, the images and results in the paper are shown at time $t = 0.72$. Note that time is normalized such that it takes unit time for a sound wave in the unshocked incident gas to traverse the width of the shock-tube. Finally, a second order Godunov method was employed for the simulation (details are in reference. [7]).

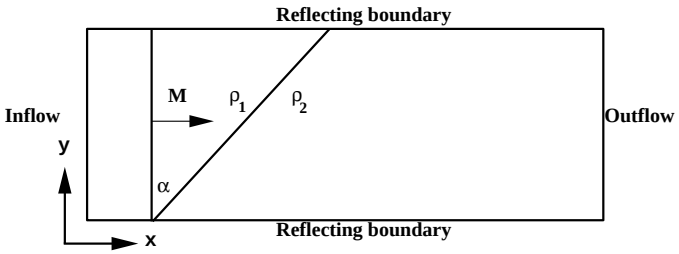



Figure 1: Schematic of the initial conditions for an unsteady two-dimensional shock contact-discontinuity interaction. The physical geometry is a two-dimensional rectangular shock-tube.

# 2 Visualization techniques

In this section, we will first discuss the numerical analogue of experimental flow visualization techniques.

## 2.1 Numerical analogue of experimental flow visualization techniques

In an experiment, as a light ray travels through a compressible gas with density variations (density variations are related to the variations in index of refraction via the Gladstone-Dale formula [8]), it undergoes three effects. The first is a displacement from its path which it would have taken in a uniform medium. The second is the angular displacement with respect to an undisturbed path, and the third is a phase shift from the undisturbed light ray. These three effects corresponds to the three main experimental visualization techniques for flows with discontinuities.

1. *Schlieren imaging.* This experimental technique depends upon the change in the refractive index as a function of the density of the gas. In fact, schlieren imaging relies on the angular deflections of light rays. The intensity of the schlieren images corresponds to the gradient of the density [8]. In the edge detection literature, the gradient is used in various methods to identify edges. These methods include the Roberts cross, Sobel, Compass, and Prewitt edge detectors [9]. Each of these

methods uses a different "convolution mask" (convolution mask is the jargon used in image processing). The Roberts cross edge detector is written below in our notation:

$$\begin{aligned}
\nabla_x \rho_{i,j} &\equiv \frac{\rho_{i+1,j+1} - \rho_{i,j}}{\sqrt{2}h}, \\
\nabla_y \rho_{i,j} &\equiv \frac{\rho_{i,j+1} - \rho_{i+1,j}}{\sqrt{2}h}, \\
\nabla \rho_{i,j} &\equiv [(\nabla_x \rho_{i,j})^2 + (\nabla_y \rho_{i,j})^2]^{\frac{1}{2}}.
\end{aligned} \tag{1}$$

where $\rho$ is the density field, and $h$ is the mesh spacing. The Sobel edge detector is written below in our notation:

$$\begin{aligned}
\nabla_x \rho_{i,j} &\equiv \frac{1}{8h}\left[2(\rho_{i+1,j} - \rho_{i-1,j}) + \right. \\
&\quad \left. \rho_{i+1,j+1} - \rho_{i-1,j+1} + \rho_{i+1,j-1} - \rho_{i-1,j-1}\right], \\
\nabla_y \rho_{i,j} &\equiv \frac{1}{8h}\left[2(\rho_{i,j+1} - \rho_{i,j-1}) + \right. \\
&\quad \left. \rho_{i+1,j+1} - \rho_{i+1,j-1} + \rho_{i-1,j+1} - \rho_{i-1,j-1}\right], \\
\nabla \rho_{i,j} &\equiv [(\nabla_x \rho_{i,j})^2 + (\nabla_y \rho_{i,j})^2]^{\frac{1}{2}}.
\end{aligned} \tag{2}$$

The density image is shown in Figs. 2 at time $t = 0.72$. The schlieren images, corresponding to the above two methods, are shown in Figure 3. The Sobel edge detector, because of its larger stencil, is smoother and less sensitive to noise. We do note that for this example, visually there is little difference between the Sobel and the Roberts cross edge detectors.

2. *Shadowgraphs.* This technique relies on the displacement of a light ray due to the change in refractive index because of spatial density variations in the gas [8]. It can be shown that the displacement experienced by a light ray depends on the second derivative of the density. The numerical equivalent of this is obtained by taking the the second derivative of the density field. A central difference approximation to calculate this at a point $(i, j)$ on a uniform mesh is the following:

$$\nabla^2 \rho_{i,j} \equiv \frac{\rho_{i+1,j} + \rho_{i-1,j} + \rho_{i,j+1} + \rho_{i,j-1} - 4\rho_{i,j}}{h^2}, \tag{3}$$

This formula is applied to the density field in the shock-contact simulation. The shadowgraph image is shown in Figs. 4, at time $t = 0.72$.

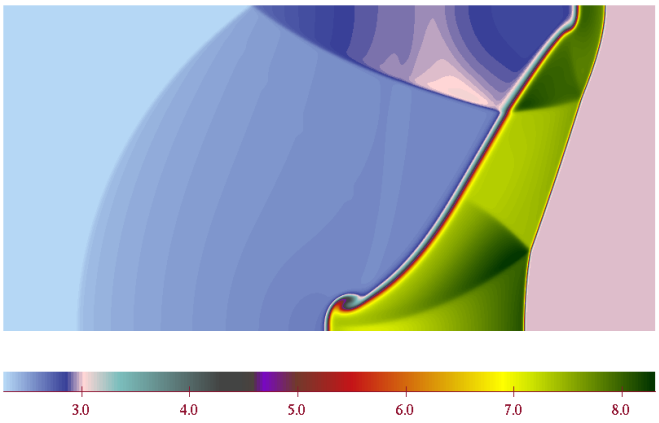



Figure 2: Density field of a two-dimensional shock contact-discontinuity interaction at time $t = 0.72$.

In edge detection literature, this technique of finding edges is sometimes referred to as the "Marr Edge Detector"[10]. Note that, in actual practice, shocks and contact discontinuities actually cause a significant amount of light diffraction as opposed to light refraction. Nonetheless, the technique that is proposed highlights the discontinuities in the numerical flow field.

3. *Interferometry.* The fringe patterns in an interferogram arise due to the phase shift of light as it moves through a density field [8]. Numerically we approximate the interferogram as follows:

$$I_{i,j} = 0(resp.1) \qquad (4)$$
$$if \quad mod\left(integer(N_f \frac{\rho_{i,j} - \rho_{min}}{\rho_{max} - \rho_{min}}), 2\right) = 0(resp.1),$$

where $N_f$ is the number of fringes in the range $[\rho_{min}, \rho_{max}]$ determined by the user. $I$ is the intensity pattern on the resulting image. Shifts in the fringe patterns occur at the discontinuities (see Figure 5).

Note that the above three methods are extensively used to visualize experiments wherein the density in one direction is integrated to elicit information

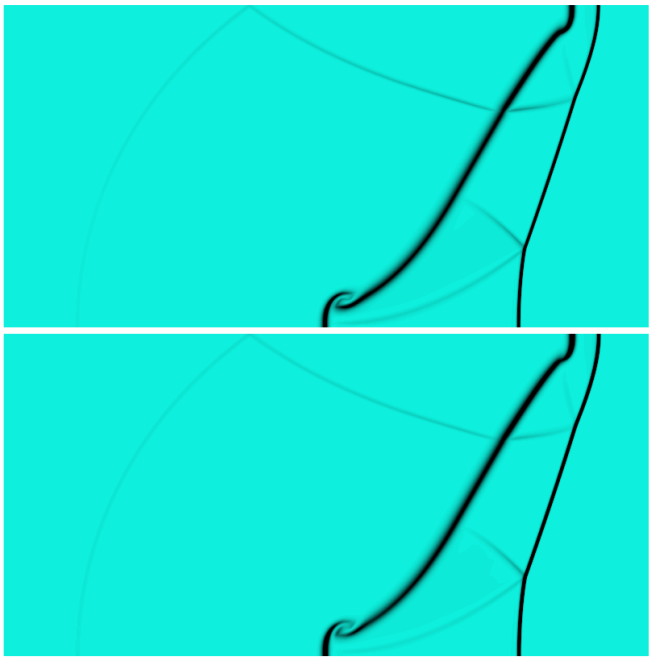


Figure 3: Numerical schlieren images of a two-dimensional shock contact-discontinuity interaction. The top image is generated using the Roberts cross edge detector, while the bottom image is generated using the Sobel edge detector. The techniques discussed here may be applied to flow variables other than the density field.



Figure 4: Numerical shadowgraph of a two-dimensional shock contact-discontinuity interaction at $t = 0.72$.

about the flow field in two dimensions. Therefore the above techniques are useful in visualizing two-dimensional experiments. Experimental techniques to obtain schlieren images and interferograms in color also exist. Furthermore, there are several variants to schlieren and interferometry which we will not discuss here. The reader is refered to the book by Merzkirch [8]. For our purposes, the numerical shadowgraph ($\nabla^2\rho$) in particular proves to be useful in isolating discontinuities in unsteady two-dimensional numerical experiments.

## 2.2 Smoothing and noise suppression

Because derivative operations are generally an order less in accuracy than the computed solution, the shadowgraph and the schlieren images are susceptible to error noise. This problem is further exacerbated since most shock-capturing methods reduce to first-order accuracy near discontinuities to maintain monotonicity. To mitigate the effects of noise, the following smoothing techniques have been examined. The first one employs a window around a point as follows

$$\tilde{q}_{i,j} = \sum_{k=-n/2}^{n/2} \sum_{l=-n/2}^{n/2} w_{k,l} q_{i+k,j+l} \tag{5}$$

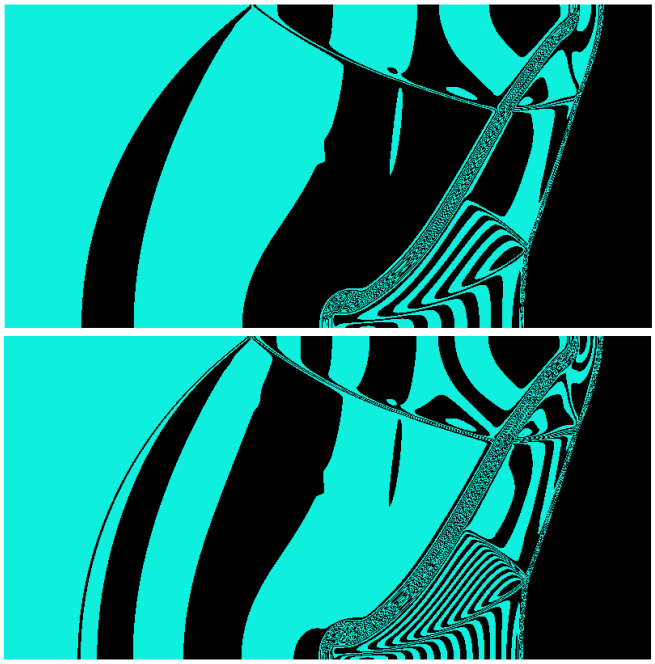


Figure 5: Numerical interferograms of a two-dimensional shock contact-discontinuity interaction. The top (bottom) image is generated with 64 (128) fringes.

such that the weights $\sum w_{k,l} = 1$. In the above equation $\tilde{q}$ is the resulting smoothed field. Another smoothing function which has been prominently employed in the image processing literature is to convolve the field with an isotropic Gaussian as

$$\tilde{q}_{i,j} = \sum_k \sum_l G_{k,l} q_{i+k,j+l}$$

$$G_{k,l} = \frac{1}{2\pi\sigma^2} \exp(-\frac{x_{k,l}^2 + y_{k,l}^2}{2\sigma^2}) \tag{6}$$

where $\sigma$ is the standard deviation in the Gaussian distribution. It is common practice in edge detection [10] to combine the Laplacian and the Gaussian operations into one convolution mask called the Laplacian of Gaussian or LoG.

## 2.3 Selection of variables

The selection of the field to highlight discontinuities in the flow is a non-trivial issue. From our experience, gathered by applying the edge detection algorithms to various fields such as the density, pressure, entropy, etc., we recommend the following variables:

- Magnitude of gradient and Laplacian of the density to visualize shocks and discontinuities.
- Gradient magnitude and Laplacian of the pressure, as well as the divergence of the velocity field to visualize shocks (see Fig. 6). Contact discontinuities do not show up in these variables because, in theory, the pressure and normal velocity are both continuous across contact discontinuities. The divergence of the velocity field ($\nabla.U$) proves to be useful in picking out the shock fronts (see Fig. 7). Note that, because shocks in perfect gases are always compressive, $\nabla.U$ is always negative at the shocks.
- It has been shown that the entropy jump across a shock wave is a third-order quantity, i.e., $\Delta s = O(M-1)^3$[11] where $M$ is the Mach number of the shock. Consequently, entropy gradients are useful to identify strong shocks in the flow field. Note that entropy is also discontinuous across contacts. The variable $\nabla^2 s$ is shown in Fig. 8. The transmitted shock and the primary contact are very clear, while the reflected shocks, which are significantly weaker, are not detected.



Figure 6: Laplacian of the pressure field in a two-dimensional shock contact-discontinuity interaction.



Figure 7: Divergence of the velocity field in a two-dimensional shock contact-discontinuity interaction.



Figure 8: Laplacian of the entropy field in a two-dimensional shock contact-discontinuity interaction.

# 3 Quantification of shocks and contacts

By quantification of discontinuities we mean the representation of discontinuities in a two-dimensional flow field by curves along which certain properties, such as the shock strength, are determined. The discontinuity is extracted as the contour corresponding to the zero crossing of the Laplacian of the field (typically chosen as the density field). This is further modified by requiring the gradient of the field at the zero crossing to be larger than a user specified threshold.

## 3.1 One-dimensional example

In this section, we will examine the following question: How accurate is the zero crossing of the Laplacian (of density in this example) in quantifying the shock and contact discontinuity locations? This issue is addressed by simulating a one dimensional shock contact-discontinuity interaction ($\alpha = 0$ in Fig 1). In this interaction, the incident shock bifurcates into a reflected and a transmitted shock. The one-dimensional interaction also permits an analytical solution [7]. We examine the difference in location of the zero crossing of $\nabla^2\rho$ in the numerical solution and the analytical solution for various resolutions (See Fig 9). We observe that, for all resolutions, the difference in the computed shock location and the analytical location differs

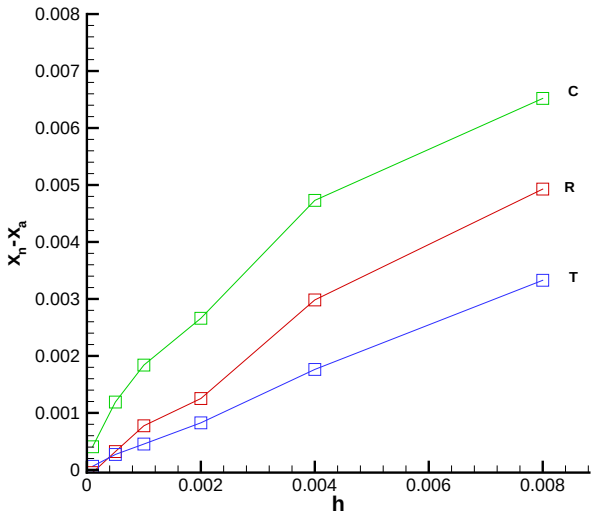



Figure 9: Difference in the numerical location $x_n$ and analytical location $x_a$ of the reflected shock (R), contact discontinuity (C), and transmitted shock (T) at various resolutions in a one dimensional shock contact-discontinuity interaction. The interaction parameters are $(M, \rho_2/\rho_1, \alpha) = (2.0, 3.0, 0.0)$. The mesh spacing is $h$.

by less than one grid cell. As the resolution is increased, this difference approaches zero. The contact discontinuity, which is typically smeared over a larger extent, is located accurately to within $2h$, twice the mesh spacing. Furthermore, with further mesh refinement, it appears that the zero crossing of $\nabla^2\rho$ does not converge to the analytical location for a contact discontinuity.

## 3.2 Algorithm

The details of the algorithm to quantify shocks and contacts in two-dimensional compressible flows follow.

1. *Simplicial decomposition of the mesh*
   The mesh is composed of quadrilaterals and numbered such that quadrilateral $(i,j)$ has four vertices at $\bar{x}(i,j)$, $\bar{x}(i+1,j)$, $\bar{x}(i+1,j+1)$ and $\bar{x}(i,j+1)$ where $i = 0,1,2,\cdots,M$, $j = 0,1,2,\cdots,N$. Each mesh quadrilateral is decomposed into two triangles. Note that such a decomposition is not unique, but we will not concern ourselves with this issue at this time. Each triangle is given a unique id number which is given by $id = 2(M-1)j + 2i + k$, and where $k = 0,1$. A table of triangles with attributes, called the Triangle Table, is generated. The initial setup by this step in the algorithm is schematically depicted in Fig. 10. Then, a global table of edges in the mesh (called the Edge Table) is generated. Each edge is given a unique identification number given by $eid = 3Mj + 3i + k$, $k = 0,1,2$. Included in the attributes for each entry in the table of triangles are the unique identification tag for the triangle and three data structures $(E0, E1, E2)$ which contain two pointers. One of these pointers points to the global Edge Table. Since each edge is shared by two triangles or is at the boundary of the domain, the second pointer points to an entry in the Triangle Table which is the neighboring triangle sharing this edge. If the edge is on the boundary, the second pointer is set to NULL. Each entry in the global Edge Table essentially contains pointers to a Vertex Table (not shown in the schematic figure) wherein the coordinates of the vertices are actually stored.

2. *Zero crossing of the Laplacian*
   The next step in the process is to compute the Laplacian of the field variable of interest. The edges intersected by the zero contour of the Laplacian are identified. Furthermore, we exclude those edges intersected by the zero contour of the Laplacian where the gradient of the field is below a threshold value. Mathematically, the intersection point is calculated as

$$\begin{aligned} \bar{x} \;\; = \;\; & \bar{x}_1 + (\bar{x}_2 - \bar{x}_1)\frac{\nabla^2\rho(\bar{x}_1)}{\nabla^2\rho(\bar{x}_2) - \nabla^2\rho(\bar{x}_1)} \\ & if \;\; \nabla^2\rho(\bar{x}_2)\cdot\nabla^2\rho(\bar{x}_1) < 0 \\ & \quad and \;\; (\nabla\rho(\bar{x}_2) + \nabla\rho(\bar{x}_1)) > 2(\nabla\rho)_{threshold}, \end{aligned} \tag{7}$$

   where $\bar{x}_1, \bar{x}_2$ are the vertices of the edge. Thus, in the above equation, $\bar{x}$ is the location of the zero crossing of the Laplacian on the edge whose

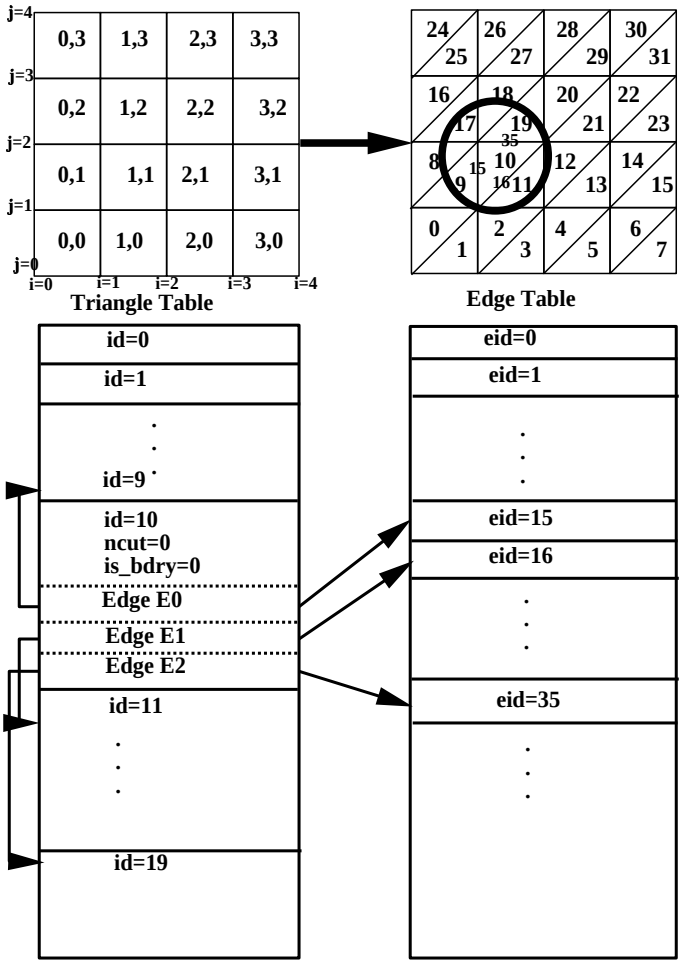



Figure 10: Simplicial decomposition of the mesh and generation of Triangle Table and Edge Table.

end points are at $\bar{x}_1$ and $\bar{x}_2$; and we choose only those edges for which the average gradient of the field of interest is larger than a user-specified threshold value $(\nabla\rho)_{threshold}$. This is done to eliminate locations where we have a zero Laplacian but which do not lie within a high gradient region. At the end of this step, we have identified all intersection points of the zero contour with all the edges.

3. *Extraction of the discontinuity curve*
   In this step, the intersection points identified in the previous step are connected to form the curves which define the discontinuities in the flow. Each discontinuity is a curve which is stored as a linked list of points. The process of identifying curves is recursive, and the pseudo-code is given in Appendix A.

4. *Spline interpolation*
   In the above step, we have isolated curves which are a list of points. The distribution of these points is, clearly, not uniform. In this step we fit natural cubic splines [12] to these. The curves are re-meshed so that points along the curve are uniformly distributed.

5. *Shock and contact discontinuity identification*
   For each curve (this curve is now the fitted spline curve), we identify whether the discontinuity is a shock wave, a contact discontinuity or neither, i.e., a spurious discontinuity is identified. The process of identifying shocks is as follows. Recall that the curve is discretized with equally spaced points. At each point along the curve, a line normal to the curve is generated with equally spaced points. For equally spaced points on either side of the curve the flow variables such as the density,pressure and velocity $(\rho, p, \bar{u})$ are calculated using bicubic interpolation. Then we evaluate the normal jump conditions, given below, using equally spaced points on either side of the curve. The question that arises is how far must we go in the normal direction so that we are not in the smeared zone of the discontinuity. For shocks, we travel along the normal direction and find the location where a certain cost function, to be defined later, is minimized. Clearly, the points closest to the shocks do not satisfy the jump conditions because of smearing.

   Let $W$ and $u_n$ be the shock and fluid velocity, respectively, in a direction normal to the shock front. The shock speed is calculated using the

following jump condition

$$W = \frac{\rho_2 u_{n2} - \rho_1 u_{n1}}{\rho_2 - \rho_1}. \tag{8}$$

We define a cost function, $\mathcal{S}$ using the three jump conditions for a shock moving with speed $W$, as

$$\begin{aligned} S_1 &= 1 - \frac{\mu^2 p_r + 1}{(\mu^2 + p_r)\rho_r}, & (9)\\ S_2 &= 1 - \frac{p_2 + \rho_2 (W - u_{n2})^2}{p_1 + \rho_1 (W - u_{n1})^2} & (10)\\ S_3 &= 1 - \frac{h_2 + (W - u_{n2})^2/2}{h_1 + (W - u_{n1})^2/2} & (11)\\ \mathcal{S} &= \omega_{s,i} S_i, \quad i = 1, 2, 3, & (12) \end{aligned}$$

where $\rho_r \equiv \rho_2/\rho_1$, $p_r \equiv p_2/p_1$, $\mu^2 \equiv (\gamma + 1)/(\gamma - 1)$, $\gamma$ is the ratio of specific heats of the gas, and $h$ is the enthalpy. Ideally, the jump conditions across the shock must be satisfied and therefore $S_i = 0, i = 1, 2, 3$ across the shock. Numerically, due to shock smearing and, since we have only approximately determined the shock front, the jump conditions are not exactly satisfied. The final form of the cost function $\mathcal{S}$ is a weighted average of $S_i$ with weights $\omega_{s,i}$. For curves which are not shocks, the jump conditions are obviously not satisfied. We have some simple physically-based constraints which eliminate points as not belonging to shocks. For example, both the density ratio and pressure ratio across shocks in perfect gases have to be greater than unity. Furthermore, the local normal Mach number, computed by using the relative velocities, changes from larger than unity to smaller than unity across the shocks. Therefore, to eliminate computational costs, points not satisfying these constraints are eliminated and not processed any further.

For contact discontinuities, we use the fact that the pressure and normal velocities are continuous across the contacts. One difficulty with contacts is that they tend to diffuse out more than shocks. The cost function for a contact discontinuity is defined as follows:

$$\begin{aligned} C_1 &= 1 - \frac{p_2}{p_1}, C_2 = 1 - \frac{u_{n2}}{u_{n1}} & (13)\\ \mathcal{C} &= \omega_{c,i} C_i, \quad i = 1, 2. & (14) \end{aligned}$$

We find the locations on either side of the discontinuity which minimize the cost functions, ($\mathcal{S}$ for shocks and and $\mathcal{C}$ for contact discontinuities). Then the properties at these locations are evaluated, and one can then assign the shock speed, shock strength, etc. along a shock front, and strength of the vortex sheet along a contact discontinuity.

## 3.3 Two-dimensional example

We now apply the above algorithm of extracting discontinuities to the two-dimensional interaction of a shock with a contact discontinuity. The threshold used in Eqn. 7 is $\nabla\rho_{threshold} = 0.008\nabla\rho_{max}$ ($\rho_{max}$ is the maximum gradient magnitude of density). Furthermore, the field $\nabla^2\rho$ was subjected to smoothing in Eqn. 5 with $n = 2$ and weights $w_{\pm1,\pm1} = 1/16$, $w_{0,\pm1} = 1/8$, $w_{\pm1,0} = 1/8$ and $w_{0,0} = 1/4$, applied recursively four times. The extracted curves are shown in Fig. 11. Curves labeled '1', '3', '5', '6', and '7' are shock waves, while the remaining curves are contact discontinuities. A brief explanation of the numbering system follows. The algorithm starts by scanning the xy domain from left to right and bottom to top. Whenever a discontinuity is encountered, a label is generated for it. Thus, in our example, the reflected shock labeled '1' is first encountered, followed by the primary discontinuity labeled '2'. The next discontinuity that the algorithm encounters is the transmitted shock '3' and so on. Note that in regions where the discontinuities intersect each other, the extracted curves do not intersect. In fact, curves in these regions show an unphysical turning with a high curvature. We simply caution the reader that this region of unphysical turning must be ignored in the quantification process. As an example of this phenomenon consider Fig. 12. There, we have magnified the region where shocks labeled '3' and '5' meet the contact discontinuity labeled '4'.

We now apply the quantification part of the above algorithm. In practice, we found that the cost function which best quantifies the shock is the one with weights $\omega_{s,1} = 1, \omega_{s,2} = \omega_{s,3} = 0$. This cost function has a very well-defined minimum as we move along a direction normal to the shock curve. The magnitude of the normal shock velocity for shocks labeled '1' and '3' is plotted in Fig. 13 as a function of the length of the shock curve. For reference, we also plot the speeds of the reflected and transmitted shocks in a one-dimensional interaction. Note that shock '3' is really comprised of three different shock fronts. These are the Mach stem extending from the lower boundary to 'T1', followed by the shock front from 'T1' to 'T2', and

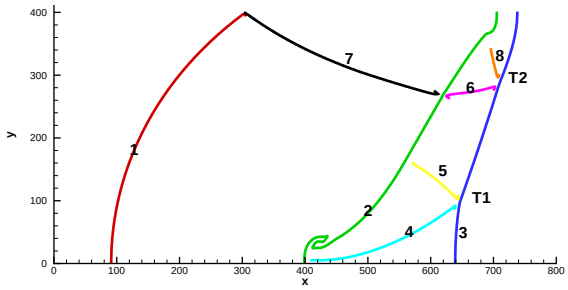



Figure 11: Extracted shocks and contacts in the two-dimensional shock contact-discontinuity interaction. Curves labeled '1', '3', '5', '6', and '7' are shock waves, while the remaining curves are contact discontinuities. Labels 'T1' and 'T2' are locations of triple points where three shocks and one contact discontinuity meet. The $x$ and $y$ axes in the figure are normalized by the mesh spacing.

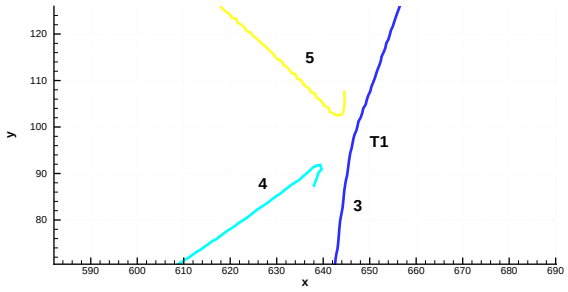



Figure 12: Magnified neighborhood of triple point 'T1'. The extracted shocks are '3' and '5' and contact is '4'. The $x$ and $y$ axes in the figure are normalized by the mesh spacing.

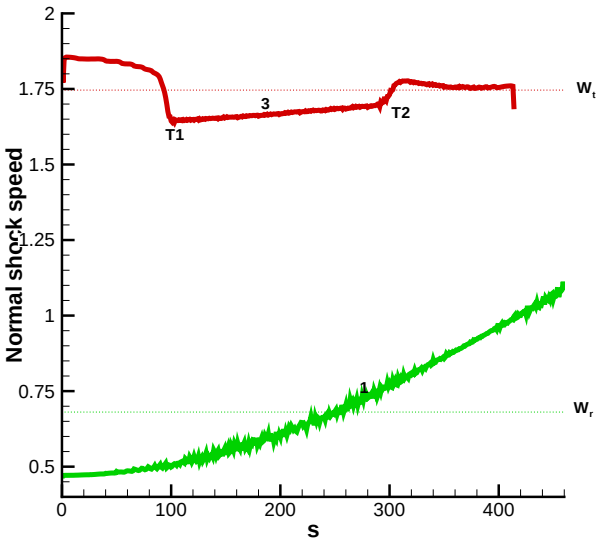



Figure 13: Normal shock speed as a function of arc length $s$ of the shock fronts identified by labels '1' and '3'. Labels 'T1' and 'T2' on shock number '3' are approximate locations of the triple points on the shock front. The horizontal lines are the speeds of the reflected (labeled $W_r$) and the transmitted shock (labeled $W_t$) in a 1D shock-contact interaction, and they are shown for reference.

finally from 'T2' to the upper boundary. The zero crossing of the Laplacian of density (in fact even other variables) fails to distinguish between these three shocks and identifies these as a single shock. However, in the quantification of the shock front, we see large changes in the normal shock speed. We also observe that the normal shock speed of shock '1' is not smooth. This may be due to several sources of error: the numerical method used to compute the flow, the error in the identification method of the zero crossing (which essentially employs linear interpolation), or the cost function used to quantify the shock front, or a combination of these. A thorough analysis of these errors is beyond the scope of the present study and is left for future work.

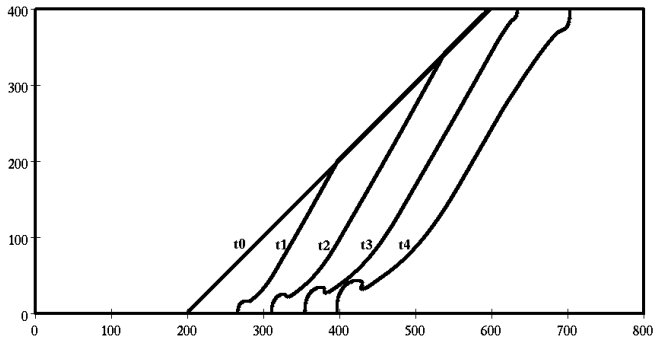



Figure 14: The zero level set at various times in the two-dimensional shock contact-discontinuity interaction. The times shown are $t0 = 0.0$, $t1 = 0.20$, $t2 = 0.38$, $t3 = 0.54$, $t4 = 0.72$. The $x$ and $y$ axes in the figure are normalized by the mesh spacing.

## 3.4 Tracking contacts via the level set approach

In our example, we have an interface between two gases which we labeled as our "primary" contact discontinuity. A useful method to extract and track such a contact discontinuity in compressible flows is to use the level-set approach [13]. In this approach, one solves the following partial differential equation in two dimensions in addition to the equations governing fluid motion:

$$\frac{\partial \rho\zeta}{\partial t} + \frac{\partial \rho\zeta u}{\partial x} + \frac{\partial \rho\zeta v}{\partial y} = 0, \tag{15}$$

where $\zeta$ is the level-set variable. We initialize $\zeta = \pm 1$ on either side of the interface so that at time $t$ the level set $\zeta(t) = 0$ defines the interface (see Fig. 14). A comparison of the extraction of the primary contact, using the algorithm presented in the previous section, and the level set method showed little difference between the two except in the roll-up in the vicinity of the lower boundary. This region is shown in Fig. 15.

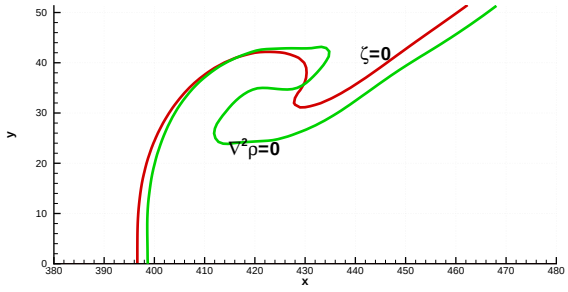



Figure 15: A comparison of the zero level set and zero crossing of the Laplacian for the primary contact discontinuity in the two-dimensional shock contact-discontinuity interaction. The $x$ and $y$ axes in the figure are normalized by the mesh spacing.

# 4 Conclusion

In this paper, we presented the numerical analogue of experimental flow visualization techniques for the compressible flow of a gas with shocks and contact discontinuities. An algorithm based on the zero crossing of the Laplacian of a field quantity (usually density) was developed to extract the discontinuities. The discontinuities were characterized by curves which were extracted using a recursive technique. Furthermore, we also quantified properties along the extracted curve such as the local strength of the shock or the normal shock speed. This determination is based on the minimum of a cost function in a direction normal to the shock front. The extracted discontinuities with associated properties may be thought of as a means of data reduction. By way of illustration, we applied the methods developed in this paper to the unsteady interaction of a shock wave with a contact discontinuity which yields a flow field rich in bifurcations and discontinuities. An extension of the extraction and quantification algorithm to three dimensions is left for the future. Several other topics were briefly mentioned in this paper. These include the selection of appropriate variables. It is recommended that density be used to extract both contact discontinuities and shock; pressure and the divergence of velocity to extract only shocks; and entropy to extract contacts and strong

shocks. Different edge detection techniques were covered in the paper. We recommend the use of the zero crossing of the Laplacian of a field variable (a la Marr edge detector) to extract discontinuities. One-dimensional test indicate that the location of the contact may not be exactly correct. For special cases, the level-set method proves useful in tracking contact discontinuities.

Finally, we remark that there are several sources of noise in the entire process: from the simulation method, to the extraction technique, to the form of the cost function used to quantify the properties along the discontinuity. A judicious application of some smoothing techniques mitigates some noise. Quantification of the error sources is also left for future work. In addition, tracking of the discontinuities from one time step to the other is also left for the future.

# Acknowledgments

This work was supported in part by NAS, NASA Contract NAS2-14303. The author is grateful to Prof. Norman J. Zabusky, Rutgers University, for many valuable discussions.

# Appendix A: Pseudo-code to extract the discontinuity curves

```
Triangle triangle[NTriangles];
Curve curve[];
int n;
int ncurve;
int nedge;
int i;

for(n=0;n<NTriangles;n++){
// If triangle is cut only once
// this is the beginning of a curve.
 if(triangle[n].ncut==1){
// Determine which edge is cut.
  for(i=0;i<3;i++){
   if(triangle[n].edge[i].iscut)
```

```
    nedge=i; break;}
 }

// Get the coordinates of the
// intersection point with the cut edge.
 triangle[n].edge[nedge].
        GetIntersectionPoint(&x, &y);
 triangle[n].edge[nedge].iscut=0;
 if(triangle[n].edge[nedge].
        neighbor_triangle!=NULL){
  nt=triangle[n].edge[nedge].
     neighbor_triangle.id;
  triangle[n].ncut=0;
// Add intersection point to the curve.
  curve[ncurve].AddPoint(x,y);
// Traverse the curve using the
// following recursive routine.
  TraverseCurve(ncurve,nt);
 }
 ncurve++;

// The curve can also start at the boundary.
 if(ncut==2 && triangle[n].IsOnBoundary){

  for(i=0;i<3;i++){
   if(triangle[n].edge[i].iscut &&
      triangle[n].edge[i].
      neighbor_triangle==NULL){
    nt=triangle[n].edge[i].
       neighbor_triangle.id;
    triangle[n].ncut=0;
    triangle[n].edge[i].iscut=0;
    triangle[n].edge[i].
        GetIntersectionPoint(&x, &y);
// Add intersection point to the curve.
    curve[ncurve].AddPoint(x,y);
// Traverse the curve using the
// following recursive routine.
```

```
    TraverseCurve(ncurve,nt);
    ncurve++;
    break;
   }
  }
 }
}

// Recursive routine to traverse the curve.
TraverseCurve(int ncurve, int n)
{
// Reached end of curve
 if(triangle[n].ncut==1){
  triangle[n].ncut=0;
  return;
 }

// Still on the curve.
 if(triangle[n].ncut==2 ) {

  for(i=0;i<3;i++){
   if(triangle[n].edge[i].iscut &&
      triangle[n].edge[i].
      neighbor_triangle==NULL){
    nt=triangle[n].edge[i].
       neighbor_triangle.id;
    triangle[n].ncut=0;
    triangle[n].edge[i].
       GetIntersectionPoint(&x, &y);
    triangle[n].edge[i].iscut=0;
// Add intersection point to the curve.
    curve[ncurve].AddPoint(x,y);
// Traverse the curve using the
// following recursive routine.
    TraverseCurve(ncurve,nt);
    break;
   }
  }
```

```
 }
// Should never reach here.
  return;
}
```

# References


[1] R. Courant and K. O. Friedrichs. *Supersonic Flow and Shock Waves.* Springer-Verlag, 1948.

[2] R. J. LeVeque. *Numerical methods for conservation laws.* Birkhauser Verlag, 1992.

[3] H.-G. Pagendarm and B. Seitz. An algorithm for detection and visualization of discontinuities in scientific data fields applied to flow data with shock waves. In P. Palamidese, editor, *Visualization in Scientific Computing.* Ellis Horwood Workshop Series, 1993.

[4] K.-L. Ma, J. Van Rosendale, and W. Vermeer. 3D Shock wave visualization on unstructured grids. In *Proceeding of the 1996 Symposium on volume visualization, San Francisco, California, October 28–29.*, pages 87–94. ACM SIGGRAPH, 1996.

[5] D. Lovely and R. Haimes. Shock detection from the results of computational fluid dynamics. *Preprint.*, 1998.

[6] P. Krehl and E. Engemann. August toepler – the first who visualized shock waves. *Shock Waves*, 5:1–18, 1995.

[7] R. Samtaney and N. J. Zabusky. Circulation deposition on shock-accelerated planar and curved density-stratified interfaces: models and scaling laws. *J. Fluid Mech.*, 269:45–78, 1994.

[8] W. Merzkirch. *Flow Visualization.* Academic Press, 1974.

[9] R. J. Schalkoff. *Digital Image Processing and Computer Vision.* John Wiley and Sons, Inc., 1988.

[10] J. R. Parker. *Algorithms for Image Processing and Computer Vision.* John Wiley and Sons, Inc., 1997.

[11] P. A. Thomson. *Compressible Fluid Dynamics.* McGraw Hill, New York, 1972.

[12] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C.* Cambridge University Press, 1988.

[13] J. A. Sethian. *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Material Science.* Cambridge University Press, 1996.