An unsplit, second-order accurate Godunov method for tracking deflagrations and detonations

J.E Pilliod[1], **E.G. Puckett**[2]

1. Lawrence Berkeley National Laboratory, 1 Cyclotron Road, MS 50D-3427, Berkeley, CA, 94720, USA
2. Department of Mathematics, University of California, Davis, CA, 95616, USA

Abstract: We have developed a second-order accurate Godunov method for tracking deflagrations and detonations in two-dimensional, compressible fluid flow. This method is based on coupling a second-order accurate unsplit Godunov method to model the compressible Euler equations with a second-order accurate volume-of-fluid interface tracking method to track the reaction front. The motion of the front is determined by solving a one-dimensional Riemann problem for reacting gas flow in the direction normal to the front. We present results of computations designed to test the accuracy and validity of our method.

Key words: Deflagration, Detonation, Godunov method, reacting gas flow, Riemann problem.

1. Introduction

We present an unsplit, second-order accurate Godunov method for modeling combustion in two-dimensional, compressible flow. Our method is based on a "thin flame" approximation to the front in which the reaction is treated as occurring across an infinitely thin discontinuity in the flow field. Combustion is viewed as occurring instantaneously across the front and the front is viewed as separating "burned" and "unburned" gas. In our model the speed of the combustion front and the precise nature of the flow field immediately behind the front (i.e., on the burned side of the front) is determined by solving a one-dimensional Riemann problem for reacting gas flow (e.g., Teng et al. (1982)) in the direction normal to the front. This Riemann problem admits both deflagration and detonation solutions, thereby providing a mechanism for modeling deflagration to detonation transition (DDT).

Our motivation for adopting a thin flame model is that the temporal and spatial scales needed to accurately model the internal structure of the reaction zone are orders of magnitude smaller than those required to accurately model the domain geometry and the fluid dynamics of the flow field. Thus, numerical models that attempt to accurately resolve both the reaction zone and the entire flow field will be far more expensive computationally than thin flame models of the type proposed here. In particular, the computational cost may be prohibitively expensive for three-dimensional problems with complex geometry.

One of key questions that needs to be answered regarding the use of a thin flame model is to what extent does the model fail to reproduce important features of the flow field? We have attempted to address this question by studying the extent to which our model reproduces the "characteristic cellular" structure that is often found behind a detonation front. We have done so by modeling a problem similar to a computation conducted by Bourlioux and Majda to study the growth of modes in unstable detonation waves. Since Bourlioux and Majda's numerical method contains grid points within the reaction zone, which contains portions of the perturbed state, our initial conditions are not identical to theirs. Nevertheless, our numerical results contain a growing mode which is similar to, but not identical to, the growing mode computed by Bourlioux and Majda. We do not know how this instability is related to the characteristic cellular structure found behind unstable detonations waves - or even that it is related at all. This issue requires further investigation.

We have also developed a number of algorithmic innovations in this work. In particular, we present a method for choosing the correct physical solution of the one dimensional Riemann problem for reacting gas flow when there are several possible valid solutions (usually one detonation solution and many deflagration solutions). We have also presented several new ideas for propagating a front with a volume-of-fluid method given only the velocity at the front.

2. An outline of the method

In our method we decouple the solution of the underlying fluid flow equations (the compressible Euler equations) from the propagation of the front and solve each of these problems separately. Data from one problem is used to define the initial and boundary conditions for the other. The two solutions are coupled in a stable and conservative manner using a flux redistribution method due to Colella. An outline of our method follows.

Step I We begin by constructing the geometry of the front from the volume fraction field us-

ing a second-order accurate, piecewise linear interface reconstruction method of the type described in Pilliod and Puckett (1997). (See also Puckett (1991) and Pilliod (1992).) This decomposes the computational domain into two subdomains containing burned and unburned fluid respectively. We consider the reconstructed front to be an interior boundary separating two flow problems that each require the solution of the compressible Euler equations with the given initial data and boundary geometry. However, we need boundary conditions for each of these problems, which we obtain by extending each state beyond the front, using a procedure originally developed in Chern and Colella (1987) and Bell et al. (1991). We now describe this procedure in some detail.

Step II Given the interior boundary geometry, we wish to "extend" both the burned and unburned fluid states two cells beyond the front to provide boundary conditions at the front. Let $\Lambda_{i,j}^{n,u}$ (*resp.* $\Lambda_{i,j}^{n,b}$) denote the volume fraction of unburned (*resp.* burned) fluid in the (i,j)th cell and define $I_{i,j}$ to be 1 if the (i,j)th cell contains the front or is adjacent to a cell that contains the front and 0 otherwise. For each cell for which $\Lambda_{i,j}^{n,l} = 0$ $(l = u, b)$ and $I_{i,j} = 1$, we set

$$\overline{U}_{i,j}^{l} = \left(\frac{\sum_{nbh(i,j)} \Lambda^{n,l} U^{n,l}}{\sum_{nbh(i,j)} \Lambda^{n,l}} \right)$$

where $nbh(i,j)$ is the 3×3 grid with the (i,j)th cell in the center. This provides boundary data one cell away from the front. We then define

$$\widetilde{U}_{i,j}^{l} = \left(\frac{\sum_{nbh(i,j)} \overline{U}^{l}}{\sum_{nbh(i,j)} I} \right)$$

for each cell that borders a cell with $I_{i,j} = 1$. This provides boundary data two cells away from the front. We define the extended state as

$$U_{i,j}^{ext,n,l} = \begin{cases} U_{i,j}^{n,l} & \text{if } \Lambda_{i,j}^{n,l} > 0 \\ \overline{U}_{i,j}^{l} & \text{if } \Lambda_{i,j}^{n,l} = 0 \text{ and } I_{i,j} = 1 \\ \widetilde{U}^{l} i, j & \text{otherwise.} \end{cases}$$

Step III. Using these extended states as boundary conditions, we use an unsplit, second-order accurate extension of Godunov's method of the type described in Colella (1990) to solve the compressible Euler equations in each of burnt and unburnt fluid regions separately, thereby obtaining the fluid flow solution at the new time on either side of the front.

Step IV. Next we propagate the front in time as follows. The velocity of the front is found by solving a one-dimensional Riemann problem for reacting gas flow normal to the front. This provides a velocity field at the surface of the front. However, in order to propagate the front we need velocities at cell edges of cells close to the front (two cell widths away). To obtain these velocities we extend the velocity field from the surface of the front to the surrounding grid cells in the manner described in §4 below. We then propagate the front using the second-order unsplit algorithm developed by Pilliod and Puckett (1997).

Step V. Finally, following Chern and Colella (1987) and Bell et al. (1991) the information obtained from the Godunov solve and the front propagation is coupled together in a conservative and stable manner using the flux redistribution method described in Pilliod (1996).

3. The Riemann solver

We solve a Riemann problem for reacting gas flow at the flame front to find the velocity of the front and the flux of material through the front. We solve this one-dimensional problem normal to the flame front. We assume the burned and unburned states are constant throughout the cell, and use these states as the initial conditions.

An important difference between solving the Riemann problem for non-reacting gas flows and the Riemann problem for reacting gas flows is that the Riemann problem for reacting gas flows may not have a unique solution. The first source of non-uniqueness comes from the fact that there may be multiple deflagration solutions. Once a deflagration solution has been chosen, the Riemann problem may still have more than one solution, since there may also be a detonation solution. In this section we discuss the procedure we use to choose between the various deflagration and detonation solutions.

Three procedures that have been used to choose the solution to this Riemann problem are the following:

1. Choose the solution that has the smallest pressure change across the front (Teng et al. (1982)).
2. If it exists, choose the deflagration solution (Hilditch and Colella (1995)).
3. Choose the solution that has the smallest total variation in the pressure for the entire solution (Pilliod (1996)).

Procedure 3 is stated more precisely as follows. Let P_L denote the pressure for the burned state, $P^\star$ the pressure of the post-deflagration state, P_0 the pressure of the pre-deflagration state, $P^{\star\star}$ the pressure of the post-detonation state, P_{CJ} the pressure of the CJ-detonation, and P_R the pressure of the unburned state. If

$$|P_L - P^\star| + |P^\star - P_0| + |P_0 - P_R| <$$
$$|P_L - P^{\star\star}| + |P^{\star\star} - P_{CJ}| + |P_{CJ} - P_R|,$$

then we choose the deflagration solution, otherwise we choose the detonation solution. This differs from procedure 1 in that in procedure 3 we consider the pressure changes across *all* of the waves, not just the combustion wave.

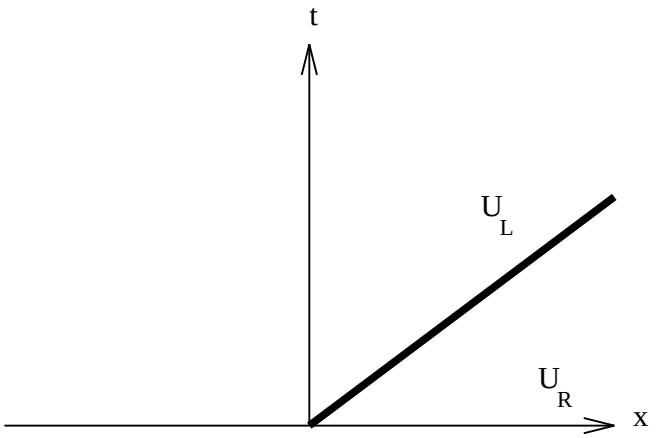

Figure 1. Detonation solution for the Riemann problem with initial data U_L, U_R.

One of our design criteria for the fluid flow solver is that it must be able to solve the Riemann problem correctly for arbitrary initial data. We now present an example for which methods that use procedures 1 and 2 fail to do this. First, consider the Riemann problem with the initial data $P_L = 3$, $\tau_L = 0.5789473684$, $u_L = 0.9176629355$, $q_L = 0$, $P_R = 1$, $\tau_R = 1$, $u_R = 0$, $q_R = 1$. This Riemann problem has both a detonation and a deflagration solution. The detonation solution is the simple detonation wave shown in Fig. 1. The deflagration solution is shown in Fig. 2, where $P_0 = 2.916912842$ and $P_1 = 2.900016834$.

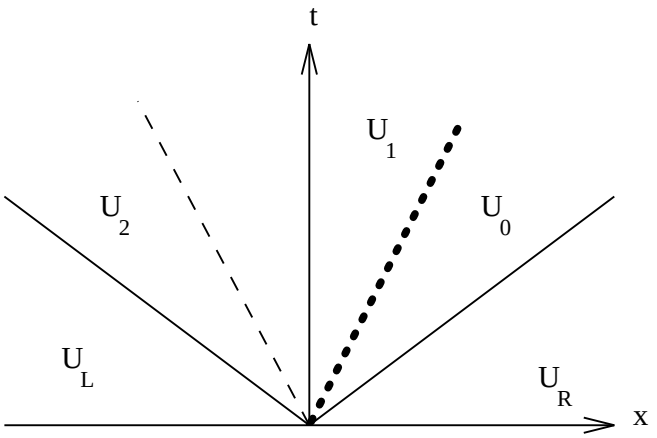

Figure 2. Deflagration solution for the Riemann problem with initial data U_L, U_R.

Now consider the Riemann problem with initial data U_R as defined above on the right and $P_4 = 1$, $\tau_4 = 1.222222222$, $u_4 = 2.051924681$, $q_4 = 0$ on the left. The solution to this Riemann problem is shown in Fig. 3. There is no deflagration solution. Suppose that we solve this Riemann problem, with the above initial data, on the grid shown in Fig. 3 using a conservative finite difference method with a Riemann solver that employs one of the three procedures described above. Since there is only one solution to the Riemann problem with initial data U_4, U_R, during the first time step the fluid flow solver will produce the same solution regardless of which procedure the Riemann solver uses to choose between a deflagration and a detonation solution. However, at some later time step, there will be a grid cell in which the Riemann solver will have initial data U_L, U_R from the Riemann problem shown in Figs. 1 and 2. Procedure 2 will pick the deflagration solution shown in Fig. 2, since it exists. Procedure 1 will also pick the deflagration solution, since the pressure change across the deflagration wave is much less than that across the detonation wave. However, in order to obtain the correct solution shown in Fig. 3, the Riemann solver must pick the detonation solution shown in Fig. 1. Thus, when used in conjunction with a fluid flow solver, neither of procedures 1 or 2 will produce the correct solution to the Riemann problem with the initial data U_4, U_R for all time. On the other hand, procedure 3 will pick the correct solution given initial data U_L, U_R, and hence will produce the correct solution to the Riemann problem with the initial data U_4, U_R for all time.

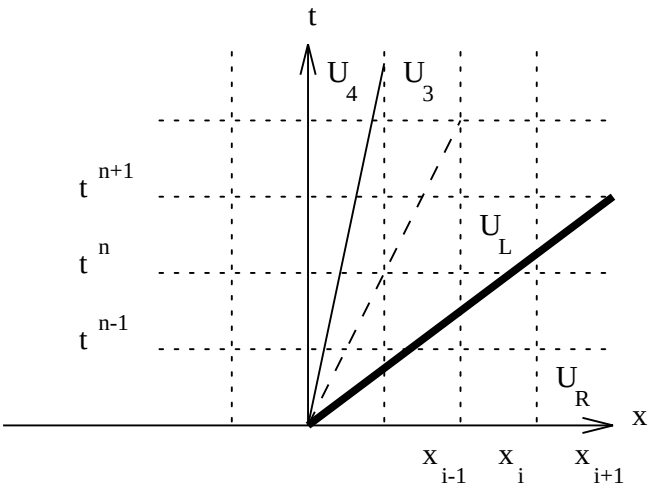

Figure 3. Exact solution for the Riemann problem with initial data U_4, U_R, with space-time grid lines superimposed over the solution.

4. Advancing the front in time

Our procedure for advancing the front in time is based on a conservative finite difference update of the volume fraction field associated with each fluid. Since the volume flux at a cell edge is $F = uV$, where u is the velocity normal to that cell edge and V is the volume of fluid crossing the edge in time Δt, we must first define a veloc-

ity field that is associated with the motion of the front on cell edges that are adjacent to the front. In other words, given the velocity of the front $\mathbf{s}^f = (s^{f,x}, s^{f,y})$ known only on the front itself, we now describe a procedure for defining a velocity field on cell edges adjacent to the front. These velocities are defined so that when the volume fraction field is updated with a conservative finite difference procedure - using these velocities to define the fluxes - the front location reconstructed from the new volume fraction field will be equivalent to advancing the old front normal to itself at the appropriate velocity. (A more detailed description of this procedure may be found in Pilliod (1996).)

We begin by using the Riemann solver described in §3 above to determine the front velocity $\mathbf{s}^f_{i,j}$ in each cell that contains a portion of the front (i.e., in each cell that contains a volume fraction $\Lambda^l_{i,j}$ that lies strictly between 0 and 1, $0 < \Lambda^l_{i,j} < 1$). This yields a velocity field in cells containing the front, which we treat as cell-centered quantities.

Now let $s^f_{i,j}$ denote the magnitude of the velocity of the flame front in the (i,j)th cell. If a cell with a volume fraction $\Lambda^l_{i,j} = 1$ is adjacent to a cell with a volume fraction $\Lambda^l_{i',j'} = 0$ then the flame front is assumed to lie on the edge joining these two cells. In this case the velocity of the front is found by calling the Riemann solver using the states in these two cells for initial data. If the cell with the zero volume fraction already has a non-zero velocity (i.e., if it is adjacent to other cells that have a volume fraction of 1), then these velocities are averaged.

Once we have the velocity field in those cells that contain the front, we extend this field to other cells that lie within two cell widths of front. To find the velocity in the (i,j)th cell we average the velocities of the adjacent cells that contain the flame front

$$s_{i,j} = \frac{1}{N} \sum_{nbh(i,j)} s^f_{i,j}$$

$$s^x_{i,j} = \sum_{nbh(i,j)} s^{f,x}_{i,j} \quad s^y_{i,j} = \sum_{nbh(i,j)} s^{f,y}_{i,j}$$

where $nbh(i,j)$ are those cells that contain a portion of the front that are within one cell of the (i,j)th cell and N is the number of cells in $nbh(i,j)$. (If $N = 0$, then $nbh(i,j)$ is redefined to be those cells that contain a portion of the front that are within two cells of the (i,j)th cell.) The components of the velocity in the (i,j)th cell $s^x_{i,j}$ and $s^y_{i,j}$ are then adjusted so that the magnitude of the velocity in the (i,j)th cell is $s_{i,j}$.

Finally we obtain the velocities at each cell edge by averaging across the edge

$$u_{i+\frac{1}{2},j} = \frac{1}{2}(s^x_{i,j} + s^x_{i+1,j}), \quad v_{i,j+\frac{1}{2}} = \frac{1}{2}(s^y_{i,j} + s^y_{i,j+1}).$$

Note that here we use $s^x_{i,j} = s^{f,x}_{i,j}$ and $s^y_{i,j} = s^{f,y}_{i,j}$ for those cells that contain the front. This algorithm has the property that it will propagate a uniformly expanding circle as a circle for arbitrarily long times.

5. Computational results

We now present results from two computations designed to test the accuracy and validity of our numerical method. We first demonstrate that our method yields a quantitatively correct (average) wave speed and qualitatively correct wave shape when we use it to reproduce a experiment containing only deflagration waves. Then we present results from using our method to compute an unstable detonation wave and compare these results to results obtained by Bourlioux and Majda (1992) on a similar problem with a numerical method that was designed to resolve the reaction zone.

5.1. A deflagration computation

In this computation, a long narrow tube (36 cm × 6 cm, $\Delta x = 0.003$ m) is filled with H_2 and O_2, in stoichiometric ratios, at standard temperature $T = 298$ K and pressure $P = 1$ atm. This mixture is ignited at one end by placing volume-fractions of 0.5 in the two cells in the center of the left hand wall, producing a deflagration wave which travels down the tube. Photographs of this experiment may be found in Fig. 1.12(b) of Oppenheim (1972).

In our computation we used the flame speed law

$$s = v + K(RT)^Q \tag{1}$$

where s is the flame speed, v is the velocity of the unburned fluid, T is the temperature of the unburned fluid, R is the universal gas constant, and K and Q are constants that were chosen as described below.

We arrived at a value of $Q = 0.228842$ by considering the case when $v = 0$ in (1) and using the fact that $s = 2919.19$ cm/s at $T = 2380$ K (Kanury (1975)) and $s = 181$ cm/s at $T = 298$ K (Dwyer et al. (1996)). We chose $K = 0.013$ m/s so that (1) yields flame speeds that best correspond to our estimates of the speed of the flames shown in Fig. 1.12(b) of Oppenheim (1972). Using the ideal gas law and data from Weast (1984) we derived the chemical energy released during combustion to be $q_0 = 23831.531$ m^2/kg s^2.

We ran the computation for $t = 275$ μs, plotting the flame front every $t = 25$ μs. The results are presented in Fig. 4. The flame in Fig. 1.12(b) of Oppenheim (1972) has essentially the same shape. The photographs of this experiment are ruled at intervals of 20 *cm*, making it very difficult to estimate instantaneous flame speeds. However, the average speed of the computed flame front shown in Fig. 4 is comparable to the average speed of the flame front in the experiment, since both are at approximately $x = 12.5$ cm at time $t = 275$ μs.

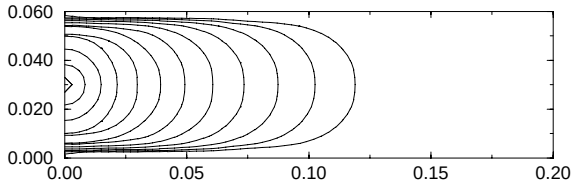

Figure 4. Computation of a deflagration experiment from Oppenheim (1972) showing the flame front position at every 25 microseconds up to time $t = 275$ μs.

5.2. An unstable detonation computation

In this section we present the results of a computation to investigate the ability of our method to reproduce the characteristic cellular structure that is typically found behind unstable detonation waves. Our computation is based on a similar computation conducted by Bourlioux and Majda (1992) with a numerical method designed to resolved the reaction zone.

For this problem we used the flame speed law $s = u_u + K\exp(-Q\rho_u/P_u)$ where (u_u, v_u), ρ_u, and P_u are the velocity, density, and pressure of the unburned fluid, respectively, and $Q = 10$ and $K = 0.1\exp(Q)$ are constants. The computational domain had 320×60 cells with $\Delta x = \Delta y = 0.25$, periodic boundary conditions on the top and bottom walls and zero derivative boundary conditions on the left and right walls. Our initial conditions were: $P_b = 13.41737525$, $\rho_b = 3.3453379$, $u_b = 2.9505135781$, $v_b = 0$ $q_b = 0$, $P_u = 1$, $\rho_u = 1$, $u_u = v_u = 0$, $q_u = 10$. We imposed a small $(0.04\,\Delta x)$ sinusoidal perturbation on the flame front at time $t = 0$. We took great care to eliminate the growth of instabilities not associated with this initial perturbation; e.g., instabilities due to "numerical noise" such as round off error.

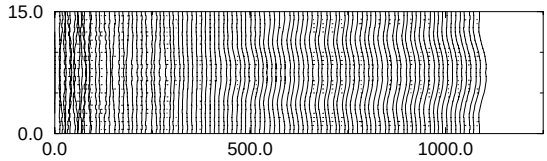

Figure 5. Reaction front positions for an unstable detonation wave up to time $t = 250$, with an amplification factor of 100.

We chose these initial conditions to (as much as possible) duplicate a computation conducted by Bourlioux and Majda (1992) to study the growth of unstable modes in detonation waves. Since Bourlioux and Majda's numerical method contains grid points within the reaction zone, which contains portions of the perturbed state, our initial conditions are not identical with theirs. Following Bourlioux and Majda, we have exaggerated the amplitude of the perturbations shown in Fig. 5 by a factor of 100 so that they can be more easily seen.

It is apparent from Fig. 5 that our numerical results contain a growing mode which is similar to - but not identical to - the growing mode computed by Bourlioux and Majda. This suggests that the continuous version of our thin flame model may support instabilities of the type that are often associated with the characteristic cellular structure found behind unstable detonation waves. The qualitative and quantitative features of our growing mode differ from the one found by Bourlioux and Majda in that our front remains smooth and does not develop the sharp kinks found in their work and the characteristic time for our mode to develop is longer than in Bourlioux and Majda's computation. Whether or not the instability shown in Fig. 5 is physically significant requires further study.

6. Conclusions

We have developed a formally second-order accurate numerical method for modeling deflagrations and detonations in reacting gas flows. Our method is based on a "thin flame model" of the reacting front in which we use a second-order accurate volume-of-fluid interface tracking algorithm to propagate the front and a second-order accurate unsplit Godunov method to approximate solutions of the compressible Euler equations on either side of the front. Boundary conditions are determined at the front for each of these two individual gas solves by computing "extended states" across the front and the speed of the front is determined by solving a 1D Riemann problem for reacting gas flow in the direction normal to the front.

The advantage to using a thin flame model is that one does not need to resolve the length and time scales associated with the reaction zone. Since these scales are typically orders of magnitude smaller than the length and time scales associated with features pertaining to the geometry of the problem domain and the dynamics of the fluid flow, our method has the advantage of greatly reducing the cost of a given computation as compared to a method that does resolve the length and time scales necessary to accurately model the reaction zone. This may be of particular benefit for large scale computations in three dimensions with complex geometry.

An important question that must be addressed concerning our method, and indeed all thin flame models, is to what extent does one obtain physically meaningful results with these models. We have attempted to (partially) address this question by presenting the results of two computations, comparing the results of the first computation with an experiment and the results of the second computation with results from a computation made with a method that resolves the reaction zone.

In the first computation the average flame speed

and qualitative shape of the flame compare well with photographs of an experiment conducted by Oppenheim (1972). In the second computation we used our method to compute the evolution of a perturbed detonation wave and show that our numerical results contain a growing mode which is similar to - but not identical to - a growing mode computed by Bourlioux and Majda (1992). We postulate that the instability in our computation is an inherent feature of (the continuous version of) our model, rather than a numerical artifact. We do not know how this instability is related to the characteristic cellular structure found behind unstable detonations waves - or even that it is related at all. This issue requires further investigation. Further work to understand the true advantages and limitations of thin flame models of deflagration and detonation is required.

References

Bell JB, Colella P, Welcome ML (1991) Conservative Front Tracking for Inviscid Compressible Flow. Proceedings of the 22nd Annual Fluid Dynamics, Plasma Dynamics, and Lasers Conference, 24-26

Bourlioux A, Majda AJ (1992) Theoretical and Numerical Structure for Unstable Two-Dimensional Detonations. Combustion and Flame 90:211-229

Chern IL, Colella P (1987) A Conservative Front Tracking Method for Hyperbolic Conservation Laws. UCRL-97200

Colella P (1990) Multidimensional Upwind Methods for Hyperbolic Conservation Laws. J. Comput. Phys. 87:171-200

Dwyer HA, UC Davis, Davis, CA (private communication)

Hilditch J, Colella P (1995) A Front Tracking Method for Compressible Flames in One Dimension. SIAM J. Sci. Comput. 16:755-772

Kanury AM (1975) Introduction to Combustion Phenomena. Gordon and Breach Science Publishers, New York

Oppenheim AK (1972) Introduction to Gasdynamics of Explosions. Springer, New York

Pilliod JE (1992) An Analysis of Piecewise Linear Interface Reconstruction Algorithms for Volume-Of-Fluid Methods. Master's thesis, U. C. Davis

Pilliod JE (1996) A Second-Order Unsplit Method for Modeling Flames in Two-Dimensional Compressible Flow. PhD dissertation, U. C. Davis

Pilliod JE, Puckett EG (1997) Second-Order Volume-of-Fluid Algorithms for Tracking Material Interfaces. (submitted to J. Comput. Phys.)

Puckett EG (1991) A Volume-of-Fluid Interface Tracking Algorithm with Applications to Computing Shock Wave Refraction. Proceedings of the 4th International Symposium on Computational Fluid Dynamics, Davis, CA

Teng ZH, Chorin AJ, Liu TP (1982) Riemann Problems for Reacting Gas with Applications to Transition. SIAM J. Sci. Stat. Comput. 42:964-981

Weast RC (1984) Handbook of Chemistry and Physics. CRC Press

▷ PREVIOUS PAPER

▷ NEXT PAPER

▷ CURRENT LOCATION IN TABLE OF CONTENTS

▷ CURRENT LOCATION IN REFERENCE NUMBER INDEX

▷ TOP OF TABLE OF CONTENTS

▷ TOP OF KEYWORDS INDEX

▷ TOP OF AUTHOR INDEX

▷ TOP OF REFERENCE NUMBER INDEX