High-Order Shock Fitting with Finite Element Methods

Luke M. D'Aquila* and Brian T. Helenbrook†
Clarkson University, Potsdam, NY, 13699

Alireza Mazaheri‡
NASA Langley Research Center, Hampton, VA, 23681

A moving-grid, shock-tracking, finite element method has been implemented that can achieve high-order accuracy for flow simulations with shocks. In this approach, element edges in the computational mesh are fitted to the shock front and moved with the shock throughout the simulation. The Euler equations are solved on the moving mesh in an arbitrary Lagrangian-Eulerian framework. Three different methods for specifying the shock motion have been tested, and the order of accuracy of the resulting solutions was verified. It was found that one of the methods is applicable only for problems with a constant upstream flow state, whereas the other two methods converge to the expected solution for general shock problems. When using a finite element method with polynomial degree p, the two general shock motion methods converged with the expected order of accuracy of $p + 1$. Finally, accurate solutions are obtained for two-dimensional supersonic nozzle and blunt body bow shock problems. Based on the results, it is concluded that the shock tracking method can preserve high-order solution accuracy in the vicinity of a shock wave.

I. Introduction

High-order numerical methods are preferred for large scale computational fluids dynamics simulations because they provide accurate solutions at a lower cost than second order methods. Finite element methods are particularly useful for applications with complex geometries because they can attain high-order accuracy on unstructured meshes [1–5]. These methods are well-suited for providing accurate numerical solutions for flow problems with continuous solutions; however, discontinuous flow features degrade solution accuracy. As a result, shock waves pose a significant challenge to the accurate numerical simulation of supersonic and hypersonic flow fields.

To incorporate shocks into the solution, many researchers have implemented shock capturing techniques. In shock capturing, the shock discontinuity is spread across a number of mesh elements and a method must be found for maintaining the monotonicity of the solution in the shock region. It has been widely reported that, for strong shocks, many classical shock capturing schemes can provide unreliable results under some circumstances [6]. Some of the reported failings of shock capturing schemes include the carbuncle phenomenon [7] and odd-even decoupling along the shock wave [8]. Furthermore, in Ref. [9], a number of shock capturing flux functions were investigated for 1D and 2D hypersonic shock problems, and none were found to be universally stable across all dimensions and test cases. Additionally, as shown by Godunov [10], all linear, monotone schemes will necessarily be reduced to first order accuracy near the discontinuity. Among existing shock capturing schemes, high-order compact shock capturing techniques such as weighted essentially nonoscillatory (WENO) polynomial limiters are the preferred methods for high-order schemes as they preserve the order of accuracy in the smooth region, and are shown to be stable for strong and complicated shock structures [4].

Another approach for incorporating shocks into numerical simulations is the shock tracking technique. This approach aims to mitigate the problems associated with the shock capturing method by maintaining the shock wave as a true flow discontinuity. Mesh edges are aligned with the shock boundary throughout the simulation, and the Rankine-Hugoniot shock jump conditions are used to specify the shock behavior. The idea of shock tracking has been around for many years [11]. More recently, shock tracking techniques have been accurately applied to the solution of numerous shock problems [2, 12, 13], demonstrating the promise of utilizing these methods in practical applications.

There are several possible implementations of shock tracking including fixed grid (also called floating shock fitting) approaches [14–19] and moving grid approaches [16, 20]. Fixed grid approaches allow the shock to move over a

*Graduate student, Department of Mechanical and Aeronautical Engineering, 8 Clarkson Ave, Postdam NY, 13699
†Professor, Department of Mechanical and Aeronautical Engineering, 8 Clarkson Ave, Postdam NY, 13699
‡Research Scientist, Aerothermodynamics Branch, 17 Victory St, Hampton VA, 23681

background mesh, and require local remeshing and/or solution interpolation in the shock region to maintain the shock as a flow discontinuity. Most fixed grid approaches have been limited to second-order accuracy, however recent work [16] has demonstrated higher-order accuracy by using one-sided finite difference schemes. In Ref. [16], moving grid shock fitting methods were found to be more robust than fixed grid approaches for problems with relatively simple shock geometries.

In the moving grid approach, the entire computational mesh moves with the fitted shock so that a single mesh boundary remains aligned with the shock interface throughout the entire simulation. This approach has been used previously for shocks to obtain high order accurate results [16, 21] by using either spectral or high-order finite difference approaches. However, the high-order solutions were typically limited to fairly simple shock and physical geometries. More complex geometries have been evaluated using low order methods, for example [22] performed a simulation of an aircraft with interacting shocks. More recently, complex shock interactions including triple junctions, reflected shocks, and contact discontinuities have been simulated using a cell-centered moving-mesh approach on moving shock-fitted grids [20].

Regardless of the implementation approach, shock tracking requires an accurate evaluation of the shock motion to determine the location of the shock interface during the solution process. Multiple approaches have been used to define the shock motion, including methods which calculate the shock velocity [14, 20] and methods which calculate the shock acceleration [23, 24]. In Ref. [16], three different methods for determining the shock motion were tested on a one-dimensional shock-entropy wave interaction problem. The first method computed the shock velocity using a Riemann invariant propagating upstream from behind the shock with the Rankine-Hugoniot (RH) shock jump conditions [11, 20]. In the second method, a time derivative of the RH conditions was used with a momentum equation immediately downstream from the shock to evaluate the shock acceleration [24]. In the third method, a time derivative of the RH conditions was used with a characteristic relation immediately downstream from the shock to determine the shock acceleration [23]. Although the results converged to the expected solution state for all methods, the third method was the only one to preserve the expected 5$^{\text{th}}$ order spatial accuracy. The authors did not provide an explanation as to why this was the case. These results indicate that the accuracy of the shock tracking method is heavily dependent on the approach used to evaluate the shock motion. In this paper, we further investigate these methods in an effort to better understand why some may perform better than others.

In the present work, we combine the finite element method with an arbitrary Lagrangian-Eulerian mesh movement simulation which tracks the position of a single shock. The method is implemented in one and two spatial dimensions, and the three shock motion evaluation approaches presented in Ref. [16] are implemented for comparison. The spatial and temporal order of accuracy of the flow solution is evaluated for each method in 1D. Then the best method is applied to a 2D supersonic nozzle problem and a blunt body bow shock problem.

II. Governing Equations

In this work we evaluate the compressible Euler equations in one and two spatial dimensions:

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}_j}{\partial x_j} = 0, \tag{1}$$

where

$$\mathbf{w} = \begin{bmatrix} \rho \\ \rho u_i \\ \rho E \end{bmatrix}, \quad \mathbf{f}_j = \begin{bmatrix} \rho u_j \\ \rho u_i u_j + P\delta_{ij} \\ \rho E u_j + P u_j \end{bmatrix}, \quad E = \frac{P}{\rho(\gamma - 1)} + \frac{1}{2} u_i u_i.$$

Here, ρ is the fluid density, u_i are the flow velocity components, E is the total energy, P is the pressure and γ is the ratio of specific heats.

These equations are solved on a moving domain based on an arbitrary Lagrangian-Eulerian (ALE) mesh movement that tracks the position of the shock. To evaluate (1) over a moving domain, it is convenient to write the governing equations in unsteady curvilinear coordinates (ξ_k, τ) with the following notation

$$x_i = x_i(\xi_k, \tau),$$
$$t = \tau,$$

where x_i are the cartesian spacial coordinates, ξ_k are the curvilinear spacial coordinates, and t and τ are the time variables in fixed and moving coordinate systems, respectively. Introducing the Jacobian matrix of the coordinate transformation, $J_{ij} = \frac{\partial x_i}{\partial \xi_j}$, the Jacobian determinant, $|J|$, and the domain velocity, $x_{i,\tau}$, the governing equations can be written in curvilinear coordinates as

$$x_{i,\tau} = \frac{\partial x_i}{\partial \tau}, \tag{2}$$

$$\frac{\partial(|J|\mathbf{w})}{\partial \tau} + \frac{\partial \widetilde{\mathbf{f}}_j}{\partial \xi_j} = 0, \tag{3}$$

where the effective flux $\widetilde{\mathbf{f}}$ is defined as

$$\widetilde{\mathbf{f}}_i = |J| J_{ij}^{-1} \left(\mathbf{f}_j - \mathbf{w} x_{j,\tau} \right). \tag{4}$$

If an explicit time integration scheme is used, the governing equation must take the form $\partial \mathbf{w} / \partial \tau = \mathbf{f}$; therefore, Eq. (3) is not suited for explicit schemes because $|J|$ is inside the time derivative. To achieve the desired form, we apply the chain rule to Eq. (3), and arrive at the final form of the governing equation used for explicit schemes:

$$\frac{\partial \mathbf{w}}{\partial \tau} + \frac{\mathbf{w}}{|J|} \frac{\partial |J|}{\partial \tau} + \frac{1}{|J|} \frac{\partial \widetilde{\mathbf{f}}_j}{\partial \xi_j} = 0. \tag{5}$$

III. Shock Tracking Method

Two separate finite element codes are used to examine the proposed shock tracking methods. The first code is developed in one spatial dimension to determine whether the proposed methods can attain high-order accuracy for geometrically simple shock problems. The second code solves two dimensional problems and is used to demonstrate the utility of the method for solving more realistic problems. The two codes differ in temporal and spatial discretization but employ a similar moving mesh shock tracking methodology.

A. Shock Evaluation

In the shock tracking approach, the computational domain is divided into two regions separated by the shock discontinuity. The domain interface, which is aligned with the shock front, moves throughout the simulation based on the shock motion to follow the shock position. As a result, both the upstream and downstream domain regions contain entirely smooth flow fields which can be evaluated with a standard finite element method. During each time iteration of the solution process, the shock velocity is evaluated and time integrated consistently with the flow variables to obtain the new shock position.

Some shock motion evaluation methods calculate the shock velocity directly, while others calculate the shock acceleration. If the shock acceleration is calculated, it must be time integrated to determine the shock velocity. This adds an additional unknown variable to the solution process, increasing the computational cost and complexity of the method. Therefore, if two methods provide solutions with an equivalent error level, the method which calculates the shock velocity is preferred to the method which calculates the shock acceleration.

The flow velocity relative to the shock on the upstream side of the shock front is supersonic while the downstream side is subsonic; therefore, for the upstream domain the shock boundary is a supersonic outlet and the behavior of the upstream flow is independent of the shock physics. The RH shock jump conditions are used when evaluating the shock motion; specifically, the RH conditions establish the relationship between the upstream fluxes and downstream fluxes given the shock velocity. In general, the shock velocity is unknown and downstream information must be introduced to determine it. There are multiple approaches to evaluating the shock motion; here, we investigate three different methods that are presented in Ref. [16] to assess their practicality and accuracy.

1. Method 1 [20]

In this method, the RH shock jump conditions are used with a Riemann invariant, which propagates upstream from behind the shock, to determine the shock velocity. First, the upstream Mach number relative to the shock front, M_u, and the Riemann invariant, R, are introduced in the direction normal to the shock front as

$$M_u = \frac{(\mathbf{u}_u \cdot \mathbf{n} - u_s)}{c_u}, \tag{6}$$

$$R = \frac{2c_d}{\gamma - 1} - \mathbf{u}_d \cdot \mathbf{n}, \tag{7}$$

where subscripts u and d denote upstream and downstream respectively, u_s is the normal component of the shock velocity, c is the speed of sound, and $\mathbf{n}$ is the shock normal vector. The shock normal vector is defined as an outward normal from the upstream side of the shock pointing into the downstream flow region. Combining Eq. (7) with the RH shock jump conditions provides, after some algebra, an equation for the shock Mach number in terms of known quantities:

$$\frac{1}{\gamma-1}\sqrt{(2\gamma M_u^2-(\gamma-1))((\gamma-1)M_u^2+2)/M_u^2}+\frac{M_u^2-1}{M_u}=\frac{\gamma+1}{2c_u}(R+\mathbf{u}_u\cdot\mathbf{n}). \tag{8}$$

Equation (8) is solved numerically using the Newton-Raphson method with an initial guess $M_u^{t+\Delta t}=M_u^t$. With the known value of M_u and the upstream flow properties, the shock velocity, u_s, can then be evaluated using Eq. (6).

2. Method 2 [24]

In this method, the shock acceleration is calculated using an equation for the momentum normal to the shock immediately downstream of the shock with the RH shock jump conditions. Here, we introduce the notation for flow velocity normal to the shock front, $\mathbf{u}\cdot\mathbf{n}=u^n$. First, an expression for the normal momentum downstream of the shock, $\rho_d u_d^n$, is determined based on the conservation of mass across the shock,

$$\rho_d u_d^n=\rho_d u_s+\rho_u(u_u^n-u_s), \tag{9}$$

and the jump condition for density across the shock,

$$\rho_d=\rho_u\left[\frac{(\gamma+1)M_u^2}{(\gamma-1)M_u^2+2}\right],\quad M_u^2=\frac{\rho_u(u_u^n)^2}{\gamma p_u}. \tag{10}$$

Combining Eqs. (9) and (10) provides an equation for the momentum downstream of the shock:

$$\rho_d u_d^n=\frac{\rho_u(u_s-u_u^n)(\gamma(\rho_u(u_s-u_u^n)u_u^n-2p_u)+\rho_u(2u_s^2-3u_su_u^n+(u_u^n)^2))}{\gamma(2p_u+\rho_u(u_s-u_u^n)^2)-\rho_u(u_s-u_u^n)^2}. \tag{11}$$

For a constant upstream flow state, the chain rule can be applied to the rate of change of the downstream momentum to develop an equation for the shock acceleration:

$$\frac{\partial u_s}{\partial\tau}=\left(\frac{\partial(\rho_d u_d^n)}{\partial u_s}\right)^{-1}\frac{\partial(\rho_d u_d^n)}{\partial\tau}. \tag{12}$$

The first part of the right-hand side product, $\partial(\rho_d u_d^n)/\partial u_s$, is computed directly from Eq. (11). The second part, $d(\rho_d u_d^n)/d\tau$, is evaluated directly from the momentum equation in the Euler system of equations. The shock acceleration and velocity normal to the shock front are both added to the system of unknown variables for time advancement.

This method is well-suited for solving unsteady shock problems with a constant upstream flow state. However, if the properties upstream of the shock do not remain constant, Eq. (12) must account for the time-dependency of the flow properties as well as the time-dependency of the shock velocity. To generalize this method for all unsteady shock problems, Eq. (12) would take the form

$$\frac{\partial u_s}{\partial\tau}=\left(\frac{\partial\left(\rho_d u_d^n\right)}{\partial u_s}\right)^{-1}\left[\frac{\partial\left(\rho_d u_d^n\right)}{\partial\tau}-\frac{\partial\left(\rho_d u_d^n\right)}{\partial\rho_u}\frac{\partial\rho_u}{\partial\tau}-\frac{\partial\left(\rho_d u_d^n\right)}{\partial u_u^n}\frac{\partial u_u^n}{\partial\tau}-\frac{\partial\left(\rho_d u_d^n\right)}{\partial p_u}\frac{\partial p_u}{\partial\tau}\right]. \tag{13}$$

As originally developed in Ref. [24] and implemented for comparison with other methods in Ref. [16], this method uses Eq. (12) to calculate the shock acceleration rather than Eq. (13). For this investigation, the method is implemented following Refs. [16, 24] identically. After inaccurate results were obtained for problems with an unsteady upstream flow state, the method was further investigated in an effort to explain the erroneous results. Thus the general form in Eq. (13) was developed, but not considered in this work.

3. Method 3 [23]

In this method, the shock acceleration is calculated based on a characteristic relation behind the shock and a time derivative of the RH shock jump conditions. We start with the conservation law in the reference coordinate frame as given by Eq. (5). The flux Jacobian matrix of Eq. (5) evaluated immediately downstream of the shock, $\widetilde{\mathbf{A}}_d = \partial\widetilde{\mathbf{f}}_d/\partial\mathbf{w}_d$, has the eigenvalues $\lambda_d = (u^n - u_s,\ u^n - u_s + c,\ u^n - u_s - c)_d$. For a positive flow velocity ($u^n > 0$), the eigenvalue with a negative sign is $\lambda_d = (u^n - u_s - c)_d$. This eigenvalue corresponds to the characteristic field approaching the shock from the downstream side. The corresponding left eigenvector $\mathbf{l}_d$ is

$$\mathbf{l}_d = \begin{bmatrix} \left(\frac{2cu^n+\gamma(u^n)^2-(u^n)^2}{2(\gamma-1)}\right) \\ -\left(\frac{c-u^n+\gamma u^n)}{\gamma-1}\right) \\ 1 \end{bmatrix}_d . \tag{14}$$

This eigenvalue-eigenvector pair satisfy the relationship

$$\mathbf{l}_d^T\widetilde{\mathbf{A}_d} = (u^n - u_s - c)_d\mathbf{l}_d^T. \tag{15}$$

The characteristic relationships are used with a time derivative of the RH jump conditions to evaluate the shock acceleration. The shock jump conditions state that the effective flux is conserved,

$$\widetilde{\mathbf{f}_d} \cdot \widetilde{\mathbf{n}} = \widetilde{\mathbf{f}_u} \cdot \widetilde{\mathbf{n}}, \tag{16}$$

which can be expanded using Eq. (4):

$$|J|J^{-1}(\mathbf{f}_d^n - \mathbf{f}_u^n + u_s(\mathbf{w}_u - \mathbf{w}_d)) = 0. \tag{17}$$

Taking the time derivative of Eq. (17) provides, after substituting $\widetilde{\mathbf{A}}$ and performing some algebra,

$$[\mathbf{w}_u - \mathbf{w}_d]\frac{\partial u_s}{\partial\tau} = \left[\widetilde{\mathbf{A}}_u\frac{\partial\mathbf{w}_u}{\partial\tau} - \widetilde{\mathbf{A}}_d\frac{\partial\mathbf{w}_d}{\partial\tau}\right] . \tag{18}$$

Multiplying Eq. (18) by $\mathbf{l}_d$ and substituting relationship (15) provides the final form of the shock acceleration equation:

$$\left[\mathbf{l}_d^T(\mathbf{w}_u - \mathbf{w}_d)\right]\frac{\partial u_s}{\partial\tau} = \left[\mathbf{l}_d^T\widetilde{\mathbf{A}}_u\frac{\partial\mathbf{w}_u}{\partial\tau} - (u^n - c - u_s)_d\mathbf{l}_d^T\frac{\partial\mathbf{w}_d}{\partial\tau}\right] . \tag{19}$$

Equation (19) is used to determine the normal component of the shock acceleration. To evaluate the time derivative terms $\partial\mathbf{w}/\partial\tau$, the governing equation is written in the form

$$\frac{\partial\mathbf{w}}{\partial\tau} = -\frac{\mathbf{w}}{|J|}\frac{\partial|J|}{\partial\tau} - \frac{1}{|J|}\frac{\partial\widetilde{\mathbf{f}}}{\partial\xi} . \tag{20}$$

The time derivatives are computed based on (20), where the right hand side is evaluated following the same discretization used for the interior points of the flowfield. The shock acceleration and velocity in the direction normal to the shock front are added to the system of unknown variables for time advancement.

IV. One-Dimensional Accuracy Verification

A. Solution Method

The 1D shock tracking code utilizes a discontinuous Galerkin ALE spatial discretization. The solution is evaluated over a finite spatial domain Ω decomposed into N finite elements. In the discontinuous Galerkin method, the solution vector $\mathbf{w}$ in each element is approximated as a polynomial of degree p. In each element, the discrete solution vector $\mathbf{w}$ is approximated as

$$\mathbf{w}(\tau,\xi) = \sum_{j=1}^{DOF}\overline{\mathbf{w}}_j(\tau)\phi_j(\xi), \tag{21}$$

where $\overline{\mathbf{w}}$ are unknown coefficients, ϕ_j are the solution basis functions, and DOF is the number of solution degrees of freedom per element. For 1D elements, the number of degrees of freedom is $p+1$. After applying the standard explicit discontinuous Galerkin formulation, the discretized system can be written for a single element in a simple form

$$m_{ij}\frac{\partial \overline{\mathbf{w}}_j}{\partial t} + \mathbf{r}_j(\mathbf{w}) = 0 \,, \tag{22}$$

where m_{ij} is a symmetric matrix of size DOF x DOF

$$m_{ij} = \left[\; \int_{-1}^{1} \phi_i \phi_j |J| d\xi \; \right],$$

and $\mathbf{r}_j$ is the residual vector of size DOF by the number of solution variables in each element:

$$\mathbf{r}_j = \left[\; \int_{-1}^{1} \phi_j \mathbf{w} \frac{\partial |J|}{\partial \tau} + \phi_j \hat{\mathbf{f}}(\mathbf{w}) \Big|_{-1}^{1} - \int_{-1}^{1} \frac{\partial \phi_j}{\partial \xi} \widetilde{\mathbf{f}}(\mathbf{w}) d\xi \; \right],$$

where the local Lax-Friedrichs flux is used to evaluate the numerical flux $\hat{\mathbf{f}}$.

Time advancement is performed with an explicit, 4-stage, 4$^{\text{th}}$ order accurate Runge-Kutta scheme. The conserved flow variables, the shock velocity, and the shock acceleration, when necessary, are time advanced together as a system of equations. After each time step in the solution process, the mesh point, which is aligned with the shock front, arrives at a new position determined by the time integration of the shock velocity, while the outer domain boundaries remain fixed in space. The interior mesh points deform linearly to ensure mesh quality is maintained in the vicinity of the shock.

B. Method of Manufactured Solution

Many of the classical 1D shock problems used for code verification contain additional discontinuities in solution or slope, such as contact discontinuities and expansion fans, which degrade the rate of convergence with mesh size. For this reason, the method of manufactured solution (MMS) is implemented to verify the order of accuracy of the shock tracking code for solving shock problems with smooth upstream and downstream flowfields. In this test case, the shock velocity varies sinusoidally in time and the downstream flow properties change according to the RH shock jump conditions. The results provide insight into the accuracy of the moving grid shock tracking method as well as the relative merits of each shock motion evaluation method.

The upstream flow properties, shock position, and shock velocity are specified in SI units as

$$\begin{bmatrix} \rho_u \\ M_u \\ p_u \end{bmatrix} = \begin{bmatrix} 1.225 \; kg/m^3 \\ 3 \\ 101,325 \; N/m^2 \end{bmatrix}, \tag{23}$$

$$x_s = -0.1\, L \cos\left(\frac{u_u}{L} t\right), \tag{24}$$

$$u_s = 0.1\, u_u \sin\left(\frac{u_u}{L} t\right), \tag{25}$$

where x_s is the shock position, and u_s is the shock velocity. The domain Ω is defined as $\Omega \in [-L, L]$ where $L = \frac{\pi}{2}$. To specify the downstream flow state, the RH conditions are used to evaluate the properties immediately downstream of the shock ρ_{RH}, u_{RH} and p_{RH}. The solution in the entire downstream region is then formulated as

$$\begin{bmatrix} \rho_d \\ u_d \\ p_d \end{bmatrix} = \begin{bmatrix} \rho \\ u \\ p \end{bmatrix}_{RH} \quad (1 + 0.1 \sin((x - x_s) 2\pi/L)) \,. \tag{26}$$

This solution is plugged into the Euler equations to determine the MMS source term. This source term is included in the Euler equations, driving the numerical solution to the chosen solution form. Error values are calculated by comparing the numerical solution to the chosen solution form.

Figure 1 shows the density solution after one full cycle of the shock motion predicted with polynomial degree $p = 3$. The circles indicate the numerical solution evaluated at the endpoints of each element, whereas the solid line shows the exact solution evaluated at the same locations. The shock is properly tracked and good agreement between the exact and numerical solutions is observed throughout the entire domain. Here, the solution is shown using shock motion evaluation Method 1. Methods 2 and 3 produce nearly identical results and therefore are not shown.

A grid refinement study is used to verify the order of accuracy of the solution. Three grid sizes are used, all of which are initially uniform in the upstream and downstream regions. The refined grids are created by dividing each element in the coarser grid in half. As the grid sizing is reduced by a factor of two, the time step used for the solution evaluation is also halved; therefore, the temporal and spatial error are reduced simultaneously. As a result, this test verifies both the spatial and temporal order of accuracy of the solution method. The convergence results are shown in Fig. 2, and the L^2 error of density after one full cycle of shock motion is tabulated in the Table 1. The L^2 error is calculated by integrating the density on the final grid at the end of the simulation over the entire domain, including both the upstream and downstream regions. The solution achieves the expected 4$^{\text{th}}$-order accuracy.

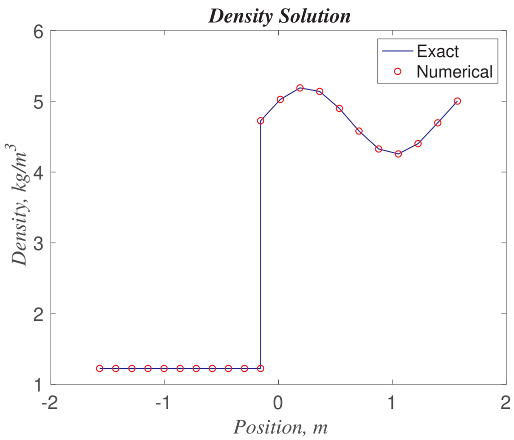

Fig. 1 Density for MMS test case after one cycle of shock motion with $N = 20$ grid elements.

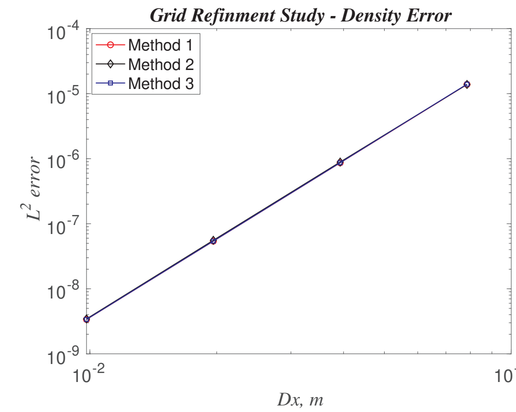

Fig. 2 Grid refinement results for MMS test case with solution polynomial order $p = 3$.

Table 1 L^2 Error of Density and Convergence Rates for MMS

Mesh Spacing [m]	Method 1	Method 2	Method 3
0.1571	1.3872e-5	1.3855e-5	1.3997e-5
0.0785	8.6543e-7	8.8870e-7	8.6675e-7
0.0393	5.4033e-8	5.5652e-8	5.4048e-8
0.0196	3.3767e-9	3.4440e-9	3.3774e-9
Convergence Rate	4.0014	3.9919	4.0054

A second, more rigorous MMS test case is used to ensure that the solution accuracy is not degraded when the upstream flow properties are unsteady. This test case is formulated identically to the previous case, except the upstream properties vary sinusoidally in time:

$$\begin{bmatrix} \rho_u \\ M_u \\ p_u \end{bmatrix} = \begin{bmatrix} 1.225 \ kg/m^3 \\ 3 \\ 101,325 \ N/m^2 \end{bmatrix} \left(1 + 0.1 \sin\left(\frac{u_u}{L} t\right)\right). \tag{27}$$

The solution state is again plugged into the Euler equations to determine the MMS source term. Unlike the previous test case, this provides a nonzero source term for the upstream region as well as the downstream region.

A solution plot is not included for this case, as it remains identical to Fig. 1 for Methods 1 and 3. Method 2 is not used, as it is not suitable where the upstream flow is time-dependent. A convergence study is performed following the same approach used for the previous test case. Convergence results are shown in Fig. 3, and the L^2 error of density is tabulated in Table 2. Again, the solution achieves the expected 4^{th}-order accuracy. While both Methods 1 and 3 provide nearly identical results, Method 1 is the preferred method due to its simplicity and lower computational cost.

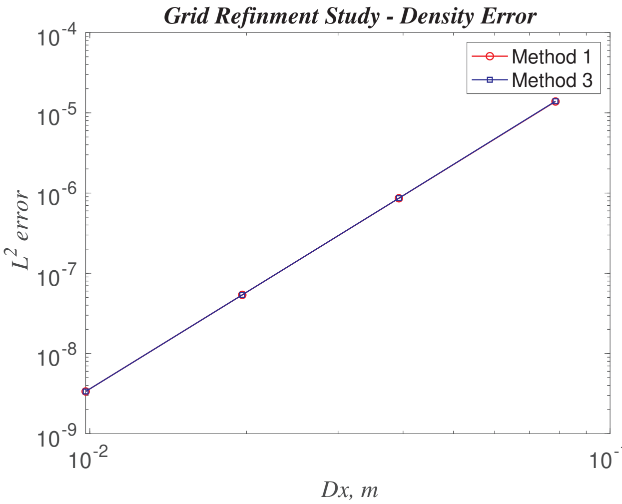

Fig. 3 Grid refinement results for unsteady upstream MMS test case with solution polynomial order $p = 3$.

Table 2 L^2 Error of Density and Convergence Rates for unsteady upstream MMS

Mesh Spacing [m]	Method 1	Method 2	Method 3
0.1571	1.3854e-5	N/A	1.3969e-5
0.0785	8.6167e-7	N/A	8.6316e-7
0.0393	5.3834e-8	N/A	5.3856e-8
0.0196	3.3641e-9	N/A	3.3641e-9
Convergence Rate	4.0024	N/A	4.0062

C. Shock-Entropy Wave Interaction

The same three shock motion evaluation methods were implemented to solve a one-dimensional shock interaction with an entropy wave in Ref. [16]. The test case is a modified version of the Shu-Osher problem [25] to maintain continuous solutions downstream of the shock. Here, the downstream solution is continuous up to third derivatives. The initial condition takes the form

$$(\rho, u, p) = \begin{cases} \left(\frac{(\gamma+1)M^2}{(\gamma-1)M^2+2}, \sqrt{\gamma}\left[\frac{2(M^2-1)}{(\gamma+1)M} - M\right], 1 + 2\gamma\frac{M^2-1}{\gamma+1}\right), & x \leq x_s \\ \left(1 + \epsilon \sin^4\left[2.5\pi(x + M\gamma^{(1/2)}t)\right], -M\sqrt{\gamma}, 1\right), & x > x_s \end{cases}, \tag{28}$$

where M is Mach number, $\gamma = 1.4$ is the specific heat ratio, x_s is the shock location and $\epsilon = 0.2$. The shock is initially located at $x_s = 2$ and the computational domain spans only the downstream flow region $x \in [0, x_s]$; thus the shock is a moving domain boundary. The properties given in Eq. (28) for the upstream region ($x > x_s$) specify the upstream flow properties for the shock computation. Here, we consider Mach 3. The unsteady simulation is run until $t = 0.32$ using shock evaluation Methods 1 and 3. Method 2 is not suitable to solve this problem, as the density upstream of the shock is time-dependent. This test is identical to that used in Ref. [16].

Following the approach used in Ref. [16], a reference solution is obtained using Method 3 and a fine mesh ($N = 12,800$) as a basis to evaluate the order of accuracy – no exact solution exists for this problem. The density profile at $t = 0.32$ is shown in Fig. 4 using Method 1 and a polynomial degree $p = 3$. Circles indicate the numerical solution evaluated at the endpoints of each element, while the solid line shows the reference solution evaluated at the same points. There is good agreement between the numerical and reference solutions.

Next, a grid refinement study is performed. The L^2 errors of density are shown in Fig. 5 for three grid sizes ($N = 800, 1,600, 3,200$). As in the MMS test cases, the initial grids are uniform and each finer grid is a subset of the coarser grid. The density errors are also tabulated in Table 3. Error is quantified as the L^2 error of density integrated over the final domain, which spans only the region downstream of the shock. Both the Methods 1 and 3 achieve the expected 4$^{\text{th}}$-order accuracy.

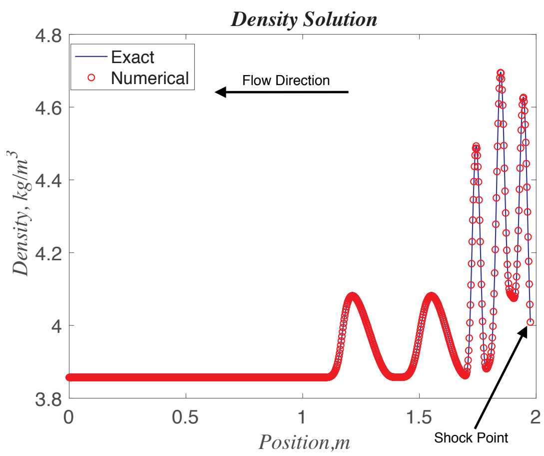

Fig. 4 Density for shock-entropy wave interaction at $t = 0.32$ with $N = 800$ grid elements.

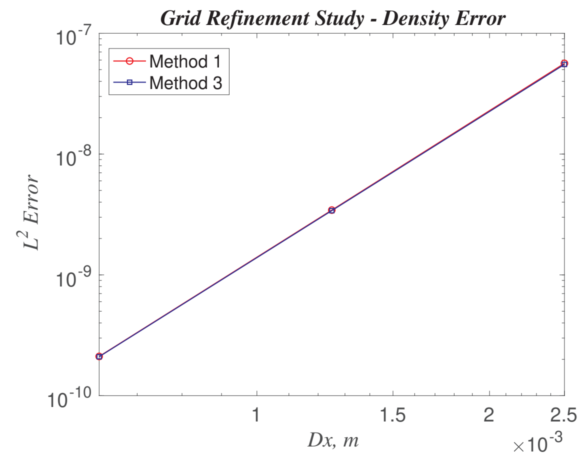

Fig. 5 Grid refinement results for shock-entropy wave interaction with solution polynomial order $p = 3$.

Table 3 L^2 Error of Density and Convergence Rates for Shock-Entropy Wave Interaction

Mesh Spacing [m]	Method 1	Method 2	Method 3
0.0025	5.6309e-8	N/A	5.5224e-8
0.00125	3.4372e-9	N/A	3.4033e-9
0.000625	2.1128e-10	N/A	2.1022e-10
Convergence Rate	4.0290	N/A	4.0186

Based on the results of the 1D test cases, we have verified that the shock tracking method with shock motion evaluation Methods 1 and 3 achieve the expected order of accuracy of the spatial discretization for unsteady problems containing a single shock. Of the shock motion evaluation methods implemented, Methods 1 and 3 are both accurate for all shock problems, while Method 2 is accurate only for problems with a constant upstream flow state. Method 1 is the preferred method as it is accurate, simple to implement and has the lowest computational cost.

Our conclusions differ from those published in Ref. [16]. Using the same shock-entropy wave interaction problem, the authors of Ref. [16] found that Method 3 was the only one to maintain the expected order of accuracy of the spatial discretization, while Method 2 reduced the solution order of accuracy slightly and Method 1 was only first order accurate. It is unclear why our results differ so drastically from those presented in Ref. [16]. It is possible that in Ref. [16] inconsistency was introduced in the evaluation of the reference solution for the shock-entropy wave interaction problem. In Ref. [16], the reference solution was calculated on a very fine grid using shock motion Method 3. If the calculation of the reference solution was inaccurate, the only method to converge with the expected order of accuracy would be Method 3. Additionally, it is unclear how the authors in Ref.[16] obtain accurate solutions using Method 2, which we have shown to be inadequate for solving problems with an unsteady upstream flow state. We speculate that in

Ref. [16] a more general form of Method 2 as given in Eq. (13) is used. Here, we only considered a simpler version of Method 2 as given in Eq. (12).

V. Two-Dimensional Implementation

A. Solution Method

The 2D code is based on a continuous Streamwise Upwind Petrov Galerkin (SUPG) ALE discretization [3, 26]. In the SUPG formulation, the discrete form of the residual R for the compressible Euler equations takes the form

$$R = \sum_{e=1}^{N_{el}} \left\{ \int_{\Omega_e} \left[\phi_m \frac{\partial w_m}{\partial \tau} - \frac{\partial \phi_m}{\partial \xi_j} \widetilde{f}_{mj} \right] d\Omega_e + \int_{\Omega_e} \frac{\partial \phi_m}{\partial \xi_j} A_{jml} T_{ln} \left[\frac{\partial w_n}{\partial \tau} + \frac{\partial \widetilde{f}_{nk}}{\partial \xi_k} \right] d\Omega_e \right\} + \int_{\Gamma} \phi_m \widetilde{f}_{mj} \widetilde{n}_j d\Gamma, \tag{29}$$

where N_{el} is the number of mesh elements, ϕ is the vector of test functions, Ω_e is the element area, Γ is the domain boundary, $\widetilde{\mathbf{n}}$ is the outward normal in curvilinear coordinates, A_j is the derivative of the inviscid flux with respect to the solution variable for each spatial dimension indexed with j, and T is a stabilization matrix. The form of the stabilization matrix T follows that of Ref. [27]. For this solution method, the approximation space is continuous everywhere in the domain except at the shock interface.

The computational mesh, which is made up of unstructured triangular elements, is divided into two regions separated by the shock interface. The shock interface is aligned with a series of mesh edges forming an internal domain boundary which is able to move and deform according to the shock motion. To maintain mesh quality in the region immediately surrounding the shock interface, the interior mesh vertex points deform according to a spring method, similar to that given in Ref. [28]. Additionally, a mesh adaptation method, following the basic formulation described in Refs. [29, 30], is used to maintain the quality of the mesh when significant interface deformation occurs. The mesh adaptation makes localized changes to the mesh structure in order to avoid global mesh regeneration and to reduce the need to interpolate solutions between meshes.

The solution is time advanced using a third order, four stage, A-stable diagonally implicit Runge-Kutta (DIRK) scheme [31]. The implicit stages are solved using a full Newton-Raphson iteration. All solution variables, including the flow solution and shock interface mesh positions, are solved for simultaneously during each stage of the Runge Kutta scheme. The Jacobian matrix is inverted using the MUMPS linear solver package [32] through the PETSc library [33].

B. Two-Dimensional Shock Interface Motion

Only shock evaluation Method 1 is used to determine the normal component of the shock velocity. Method 1 is chosen because it is more generally applicable than Method 2 and simpler than Method 3. With the shock velocity evaluated based on Eqs. (8) and (6), the normal velocity of the mesh edges which are aligned with the shock interface, $\mathbf{x}_\tau \cdot \mathbf{n}$, is set equal to the shock normal velocity, u_s. Therefore, the normal component of the interface motion is determined by

$$u_s - \mathbf{x}_\tau \cdot \mathbf{n} = 0. \tag{30}$$

In the 1D case, the shock corresponds to a single point in the grid, and the normal velocity of the shock entirely determines the shock interface motion. In the 2D case, however, the mesh edges along the shock must be distributed in the tangential direction as the shock moves and deforms. Following Ref. [34], the tangential position of each mesh element along the shock interface is determined by

$$\frac{\partial}{\partial \xi} \left(k_s \sqrt{x_\xi^2 + y_\xi^2} \right) = S_0(\xi), \tag{31}$$

where ξ is the curvilinear coordinate along the shock interface, $k_s = 1/l^2$ is a diffusivity constant, l is the initial element edge length before mesh movement, and S_0 is a source term. The diffusivity constant k_s is introduced to make the equation more suitable for meshes with an initially non-uniform distribution of edge lengths; for a given change in length of the interface, the percentage change in length of any element face is proportional to the original length of that face. The source term S_0 is evaluated on the initial mesh such that the initial mesh satisfies Eq. (31) exactly.

During initialization, the shock topology and position are assumed, and a mesh boundary is placed at this initial location. The shock evaluation requires the initial shock location to be chosen such that the upstream flow is supersonic. An unsteady simulation allows the shock boundary to move and deform until it arrives at the correct location and topology which satisfy the steady state equations.

C. Supersonic Nozzle

A 2D symmetric inviscid nozzle with the following profile

$$H(x) = A - Be^{-Cx}\cos(Dx - \pi) \quad \forall x \in [0, 10], \tag{32}$$

is considered. The shape parameters A, B, C, and D, given in Table 4, which corresponds to a nozzle with inlet to throat and outlet to inlet area ratios of 2.5 and 1.5/2.5, respectively. The throat height in this case is 1 and is located at $x = 5$. This nozzle shape is inspired by the NASA test case presented in Ref. [35], but modified to be continuous for all derivatives (C^∞). The shock is initialized as a vertical line at location $x = 7$. Figure 6 shows the nozzle geometry.

Table 4 Nozzle shape parameters.

A	B	C	D
1.3845	1.1155	0.2009	0.5594

To initialize the flow field, the quasi-1D isentropic flow relationships are used. For isentropic nozzle flow, the area ratio is given by

$$\frac{A}{A^*} = \frac{1}{M}\left(\frac{2 + (\gamma - 1)M^2}{\gamma + 1}\right)^{\frac{\gamma+1}{2(\gamma-1)}}, \tag{33}$$

where A^* is the area at the throat of the nozzle, and A and M are the area and Mach number at a chosen x-position in the nozzle. Based on the area profile given in Eq. (32), the Mach number of the flow in the region upstream from the shock is evaluated from Eq. (33) using Newton's method. In the convergent region the Mach number is chosen such that $M < 1$ and in the divergent region the Mach number is chosen such that $M > 1$. The remaining isentropic flow relations are used to initialize the pressure, temperature, and x-velocity upstream from the shock based on the Mach number and the stagnation properties. The initial y-velocity is evaluated by enforcing conservation of mass in the domain:

$$\frac{\partial \rho v}{\partial y} = -\frac{\partial \rho u}{\partial x}, \qquad \rho v(y = 0) = 0. \tag{34}$$

With the upstream flow properties known, the RH shock jump conditions are used to evaluate the properties on the downstream side of the shock. Similar to the upstream region, the region downstream of the shock is initialized based on the isentropic flow relationships and the specified nozzle shape. All quantities are made nondimensional by the stagnation pressure and temperature, and γ is equal to 1.4.

At the inlet of the nozzle, a characteristic boundary condition is applied, and at the outlet a subsonic outflow is applied. A slip wall condition is applied on the nozzle walls. An unsteady simulation is run until the shock reaches a steady state position. The steady state Mach number distribution is shown in Fig. 6. The shock interface curves slightly and becomes normal to the nozzle wall as the solution reaches steady state, but the shock remains around the expected position $x = 7$.

The solution is evaluated on three mesh configurations and a Richardson extrapolation is used to verify the solution order of accuracy. For the Richardson extrapolation, it is assumed that the solution error is proportional to h^α, where h is the mesh element edge length and α is the order of accuracy. Three steady state solutions (u_1, u_2, u_3), where u_i is some quantity of interest, are obtained for three different mesh configurations (h_1, h_2, h_3). For this problem the net force exerted in the x-direction on the nozzle walls, F_w, is chosen as the quantity of interest. The convergence rate α of the numerical solution is evaluated from the three solutions using the formula

$$\alpha = \frac{\log[(u_1 - u_2)/(u_2 - u_3)]}{\log(h_1/h_2)}. \tag{35}$$

For the mesh refinement approach used here, $h_1/h_2 = h_2/h_3 = 2$.

The solution is run with approximation polynomial orders $p = 1$, 2, and 4. The coarsest mesh is created first, and each mesh element edge is divided in half to create the finer meshes. For $p = 1$ and $p = 2$, the coarsest mesh had 13 edges across the inlet and 409 total elements. For the $p = 4$ case, the initial mesh was coarser, with 4 edges across the inlet and 44 elements to avoid round off error due to increased solution accuracy. The results of the mesh convergence study are shown in Table 5, where N_{el} is the number of mesh elements and DOF is the number of solution degrees of freedom. For each p used, convergence rates greater than $p + \frac{1}{2}$ are obtained.

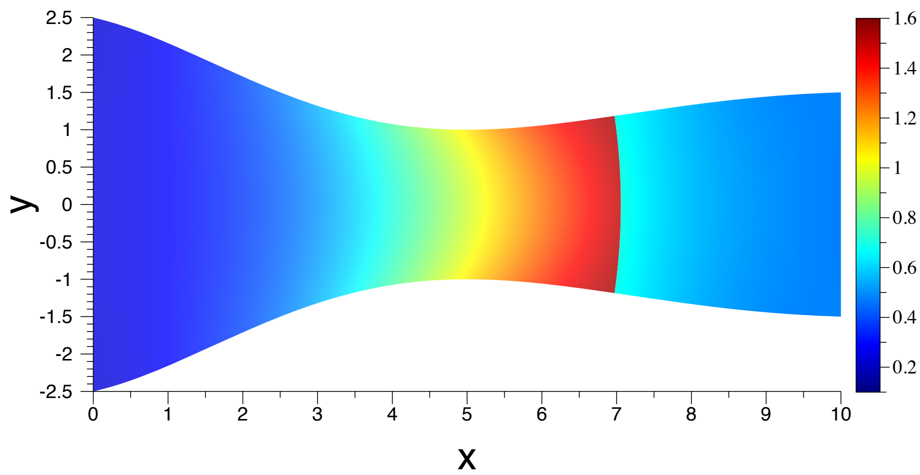

Fig. 6 Steady state Mach number distribution in nozzle.

Table 5 Nozzle convergence calculation.

	$p = 1$			$p = 2$			$p = 4$		
	N_{el}	DOF	F_w	N_{el}	DOF	F_w	N_{el}	DOF	F_w
Mesh 1	409	249	2.05979601	409	905	2.06342701	44	410	2.06336118
Mesh 2	1,636	905	2.06241820	1,636	3,444	2.06338256	176	1,522	2.06337492
Mesh 3	6,544	3,444	2.06310065	6,544	13,430	2.06337512	704	5,858	2.06337544
α	1.9420			2.5799			4.7254		

D. Blunt Body Problem

Here, we consider two-dimensional Mach 3 flow ($\gamma = 1.4$) over a cylinder with a radius $r = 0.5$. A slip wall boundary condition is applied at the cylinder wall.

To start, the bow shock position is initialized with the function $x = 0.25y^2 + 1$ (see Fig. 7). This initial guess does not need to be accurate, and is chosen based on the domain dimensions and the expected shape of the bow shock. The flow in the region upstream of the shock is initialized based on the free stream flow condition. The downstream region is initialized based on incompressible potential flow over a cylinder at a velocity equivalent to the flow velocity immediately downstream of the shock at the center line. Accuracy is not required by the initial condition; however, the initialization must be performed such that spurious shocks are not generated. If the initial flow velocity in the downstream region at the intersection of the shock and outlet boundary is subsonic, a spurious shock may be formed which is detrimental to solution convergence, as no shock-capturing capability is currently present in our code. Therefore, the initial velocity is increased to a supersonic level in the downstream region near the shock-outlet boundary intersection to ensure that the solution converges.

An unsteady simulation is run with a 4$^{\text{th}}$-order polynomial approximation ($p = 4$) until a steady state solution is achieved. Figure 8 shows the initial shock location as well as the steady state shock location; throughout the unsteady simulation the shock standoff distance from the cylinder reduces slightly, and the shock shape remains nearly unchanged. Steady state contour plots are shown in Fig. 9; the left frame shows Mach number, the middle frame shows density, and the right frame shows temperature. Additionally, line plots along the cylinder surface are shown in Fig. 10, while variations in temperature and density along the stagnation line are shown in Fig. 11. All flow variables are nondimensionalized based on the free stream temperature, speed of sound, and pressure. The results are as expected, with a tracked, detached bow shock in front of the cylinder and a smooth variation of properties along the surface.

To verify the order of accuracy of the solution, the net pressure force on the wall of the cylinder is evaluated on a

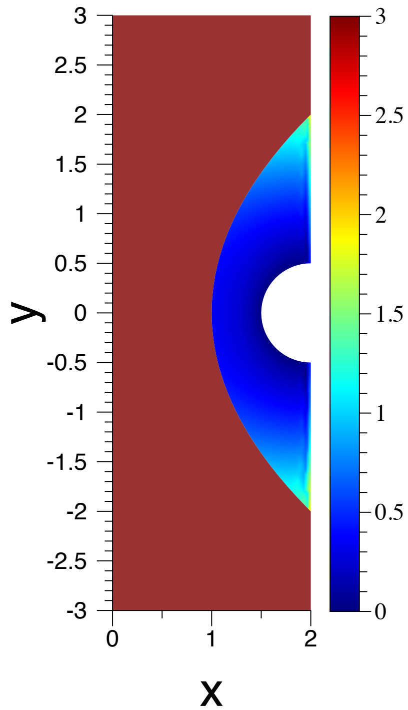

Fig. 7 Initial shock boundary shape and Mach number distribution.

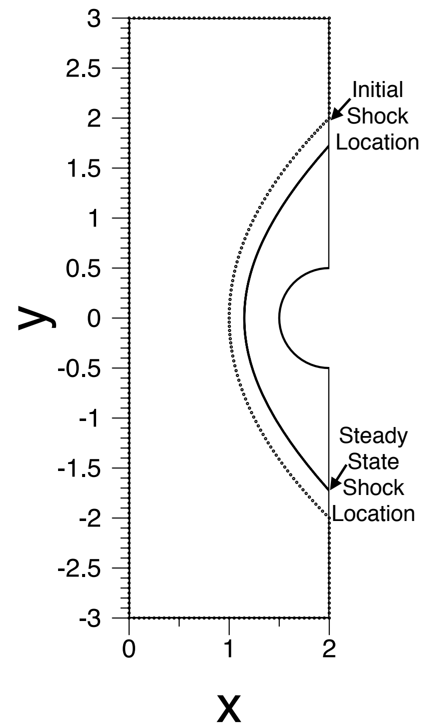

Fig. 8 Initial shock location and steady state shock location.

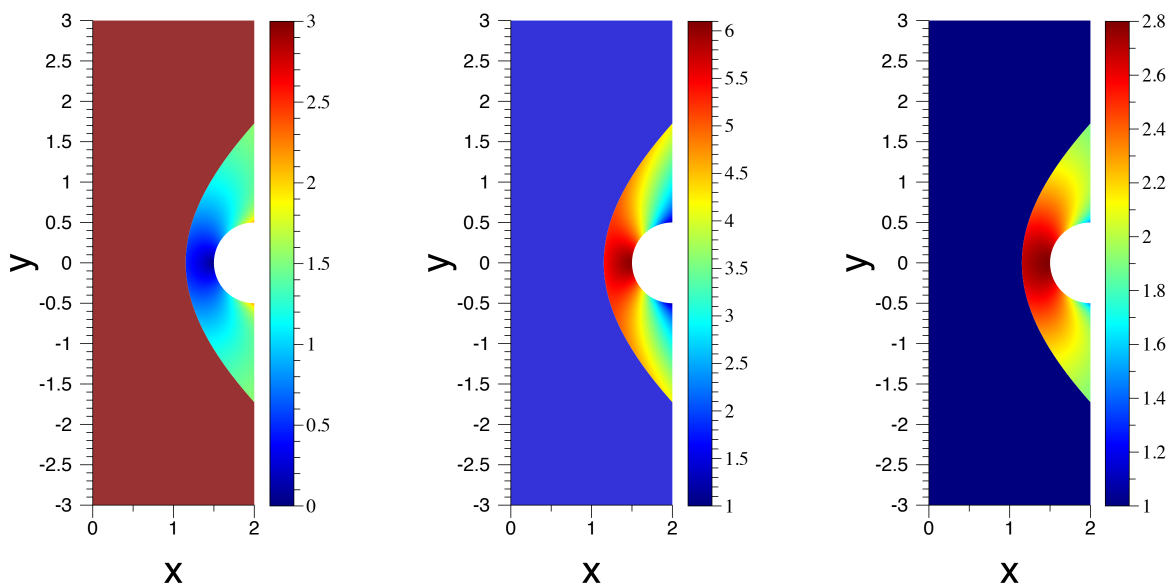

Fig. 9 Left frame: Steady state Mach number distribution. Middle frame: Steady state nondimensional density distribution. Right frame: Steady state nondimensional temperature distribution.

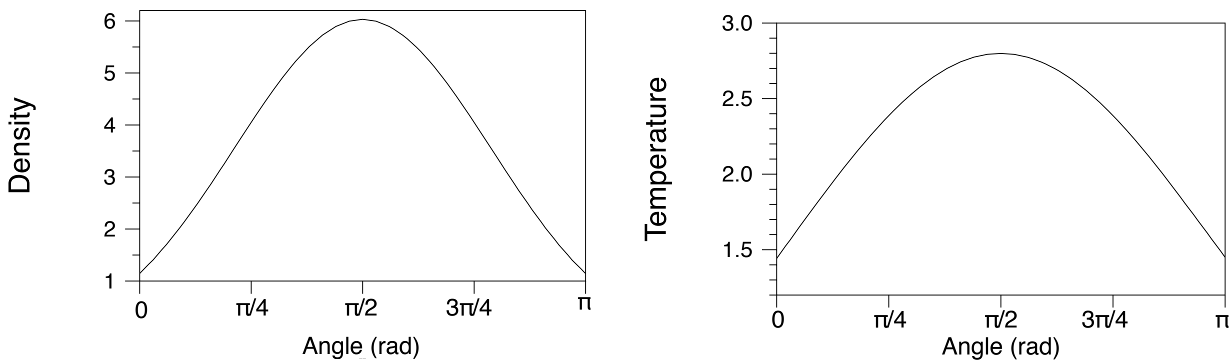

Fig. 10 Nondimensional surface density (left) and temperature (right) for the Mach 3 blunt body problem.

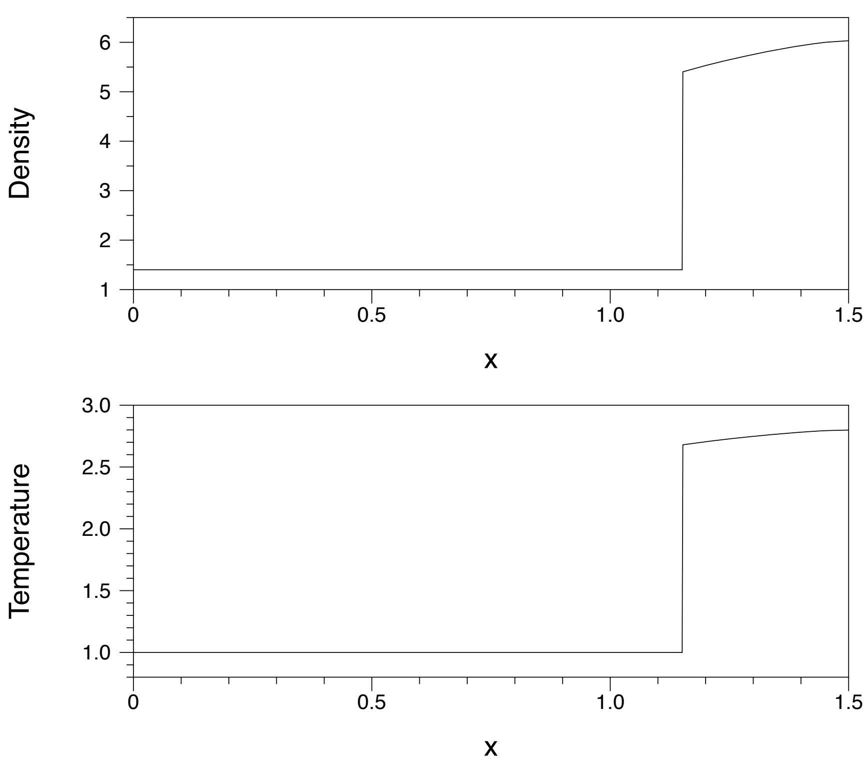

Fig. 11 Nondimensional stagnation line density (top) and temperature (bottom) for the Mach 3 blunt body problem.

Table 6 Blunt body convergence calculation.

	$p = 1$			$p = 2$			$p = 4$		
	N_{el}	DOF	F_w	N_{el}	DOF	F_w	N_{el}	DOF	F_w
Mesh 1	6,136	3,274	8.614782	657	1,451	8.617356	163	1,440	8.618011
Mesh 2	24,544	12,682	8.616728	2,628	5,528	8.617659	652	5,486	8.617774
Mesh 3	98,176	49,906	8.617429	10,512	21,566	8.617768	2,608	21,402	8.617765
α	1.4719			2.7585			4.8821		

series of three meshes and a Richardson extrapolation is performed. The Richardson extrapolation process follows that used for the nozzle test case. The results, which are tabulated in Table 6, closely follow those found for the nozzle case, with convergence rates ranging from $p + 1/2$ to $p + 1$.

VI. Conclusions

An arbitrary Lagrangian-Eulerian moving grid shock tracking finite element method has been developed in one and two spatial dimensions for solving supersonic flow problems containing a single shock wave. Three different methods for evaluating the shock motion were implemented. Methods 1 and 3 are found to provide accurate results for arbitrary shock problems, while Method 2 only provides accurate results when the upstream flow state is constant. Based on the 1D results, Method 1 is considered a preferred method due to its simplicity and accuracy. This result contradicts a previous assessment of these three methods.

The shock tracking method is also implemented for a two dimensional continuous Galerkin finite element code with mesh adaptation and refinement. Accurate results are demonstrated for a supersonic nozzle problem and a blunt body bow shock problem. The results of multiple 1D and 2D test cases indicate that the proposed method provides high-order accurate results for supersonic flow problems with a single shock wave.

References

[1] Bassi, F., and Rebay, S., "High-order accurate discontinuous finite element solution of the 2D Euler equations," *Journal of computational physics*, Vol. 138, No. 2, 1997, pp. 251–285.

[2] Corrigan, A., Kercher, A., and Kessler, D., "A Moving Discontinuous Galerkin Finite Element Method for Flows with Interfaces," Tech. rep., US Naval Research Laboratory, 2017.

[3] Helenbrook, B., and Hrdina, J., "High-Order Adaptive Arbitrary-Lagrangian–Eulerian (ALE) Simulations of Solidification," *Computers & Fluids*, Vol. 167, 2018, pp. 40 – 50. doi:https://doi.org/10.1016/j.compfluid.2018.02.028, URL `http://www.sciencedirect.com/science/article/pii/S0045793018300914`.

[4] Mazaheri, A., Shu, C.-W., and Perrier, V., "Bounded and Compact Weighted Essentially Nonoscillatory Limiters for Discontinuous Galerkin Schemes: Triangular Elements," *Journal of Computational Physics*, Vol. 395, 2019, pp. 461–468.

[5] Mazaheri, A., and Nishikawa, H., "Efficient High-Order Discontinuous Galerkin Schemes with First-Order Hyperbolic Advection-Diffusion System Approach," *Journal of Computational Physics*, Vol. 321, 2016, pp. 729–754.

[6] Quirk, J. J., "Contributed to the Great Riemann Solver Debate," *International Journal for Numerical Methods in Fluids*, Vol. 18, No. 6, 1994, pp. 555 – 574.

[7] Chauvat, Y., Moschetta, J.-M., and Gressier, J., "Shock Wave Numerical Structure and the Carbuncle Phenomenon," *International Journal for Numerical Methods in Fluids*, Vol. 47, No. 8-9, 2005, pp. 903 – 909.

[8] Gressier, J., and Moschetta, J.-M., "Robustness Versus Accuracy in Shock-Wave Computations," *International Journal for Numerical Methods in Fluids*, Vol. 33, No. 3, 2000, pp. 313 – 332.

[9] Kitamura, K., Roe, P., and Ismail, F., "Evaluation of Euler Fluxes for Hypersonic Flow Computations," *AIAA Journal*, Vol. 47, No. 1, 2009, pp. 44–53.

[10] Godunov, S. K., "A Difference Scheme for Numerical Solution of Discontinuos Solution of Hydrodynamic Equations," *Math. Sbornik (translated US Joint Publ. Res. Service, JPRS 7226, 1969.)*, Vol. 47, 1959, pp. 271–306.

[11] Moretti, G., and Abbett, M., "A Time-Dependent Computational Method for Blunt Body Flows," *AIAA Journal*, Vol. 4, No. 12, 1966, pp. 2136–2141.

[12] Zahr, M. J., and Persson, P.-O., "An Optimization-Based Approach for High-Order Accurate Discretization of Conservation Laws with Discontinuous Solutions," *Journal of Computational Physics*, Vol. 365, 2018, pp. 105–134.

[13] Ciallella, M., Ricchiuto, M., Paciorri, R., and Bonfiglioli, A., "Shifted Shock-Fitting: A New Paradigm to Handle Shock Waves For Euler Equations," *Italian Association of Aeronautics and Astronautics XXV International Congress*, 2019.

[14] Paciorri, R., and Bonfiglioli, A., "A Shock-Fitting Technique for 2D Unstructured Grids," *Computers & Fluids*, Vol. 38, No. 3, 2009, pp. 715 – 726.

[15] Ivanov, M. S., Bonfiglioli, A., Paciorri, R., and Sabetta, F., "Computation of Weak Steady Shock Reflections by Means of an Unstructured Shock-Fitting Solver," *Shock Waves*, Vol. 20, No. 4, 2010, pp. 271–284.

[16] Rawat, P. S., and Zhong, X., "On High-Order Shock-Fitting and Front-Tracking Schemes for Numerical Simulation of Shock–Disturbance Interactions," *Journal of Computational Physics*, Vol. 229, No. 19, 2010, pp. 6744–6780.

[17] Paciorri, R., and Bonfiglioli, A., "Shock Interaction Computations on Unstructured, Two-Dimensional Grids Using a Shock-Fitting Technique," *Journal of Computational Physics*, Vol. 230, No. 8, 2011, pp. 3155–3177.

[18] Bonfiglioli, A., Grottadaurea, M., Paciorri, R., and Sabetta, F., "An Unstructured, Three-Dimensional, Shock-Fitting Solver for Hypersonic Flows," *Computers & Fluids*, Vol. 73, 2013, pp. 162–174.

[19] Bonfiglioli, A., Paciorri, R., and Campoli, L., "Unsteady Shock-Fitting for Unstructured Grids," *International Journal for Numerical Methods in Fluids*, Vol. 81, No. 4, 2016, pp. 245–261.

[20] Zou, D., Xu, C., Dong, H., and Liu, J., "A Shock-Fitting Technique for Cell-Centered Finite Volume Methods on Unstructured Dynamic Meshes," *Journal of Computational Physics*, Vol. 345, 2017, pp. 866–882.

[21] Hussaini, M. Y., Kopriva, D. A., Salas, M. D., and Zang, T. A., "Spectral Methods for the Euler Equations: Part Ii - Chebyshev Methods and Shock Fitting." *AIAA Journal*, Vol. 23, No. 2, 1985, pp. 234 – 240.

[22] Marconi, F., and Salas, M., "Computation of Three Dimentional Flows about Aircraft Configurations," *Computers & Fluids*, Vol. 1, No. 2, 1973, pp. 185–195.

[23] Zhong, X., "High-Order Finite-Difference Schemes for Numerical Simulation of Hypersonic Boundary-Layer Transition," *Journal of Computational Physics*, 1998.

[24] Henrick, A., Aslam, T., and Powers, J., "Simulations of Pulsating One-Dimensional Detonations with True Fifth Order Accuracy," *Journal of Computational Physics*, 2006.

[25] Shu, C. W., and Osher, S., "Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes II," *J. Comp. Phys.*, Vol. 83, 1989, pp. 32–78.

[26] Brazell, M. J., and Helenbrook, B. T., "Continuous Finite Elements on Tetrahedra with Local Mass Matrix Inversion to Solve the Preconditioned Compressible Navier–Stokes Equations," *Computers & Fluids*, Vol. 88, No. 0, 2013, pp. 643 – 652. doi:http://dx.doi.org/10.1016/j.compfluid.2013.10.025, URL `http://www.sciencedirect.com/science/article/pii/S0045793013004088`.

[27] Helenbrook, B. T., and Cowles, G. W., "Preconditioning for Dual-Time-Stepping Simulations of the Shallow Water Equations including Coriolis Effects," *J. Comp. Phys.*, Vol. 227, No. 9, 2008, pp. 4425–4440.

[28] Venkatakrishnan, V., and Mavriplis, D. J., "Implicit Method for the Computation of Unsteady Flows on Unstructured Grids," *Journal of Computational Physics*, Vol. 127, No. 2, 1996, pp. 380–397.

[29] Baker, T. J., "Mesh Movement and Metamorphosis," *Eng. with Computers*, Vol. 18, No. 3, 2002, pp. 188–198.

[30] Baker, T. J., "Mesh Deformation and Modification for Time Dependent Problems," *Int. J. Num. Meth. Fluids*, Vol. 43, 2003, pp. 747–768.

[31] Russell Williams, I. C., Kevin Burrage, and Kerr, M., "A four-stage index 2 Diagonally Implicit Runge-Kutta method," *Applied Numerical Mathematics*, Vol. 40, No. 3, 2002, pp. 415–432.

[32] Agullo, E., and et. al., "MUltifrontal Massively Parallel Solver (MUMPS 5.3.0) User's Guide," Tech. rep., CERFACS, ENS Lyon, INPT(ENSEEIHT) - IRIT, Inria, Mumps Technologies and University of Bordeaux, 2020.

[33] Balay, S., Abhyankar, S., Adams, M., Brown, J., and Brune, P., "PETSc Users Manual," Tech. rep., Argonne National Laboratory, 2019.

[34] Helenbrook, B. T., "A Two-Fluid Spectral Element Method," *Comp. Meth. Appl. Mech. Eng.*, Vol. 191, No. 3–5, 2001, pp. 273–294.

[35] Slater, J. W., "Converging-Diverging (CDV) Nozzle," Tech. rep., NASA, 2008.