SECTION 4

NUMERICAL METHODS
NUMERICAL SIMULATIONS OF PLUNGING AND SURGING WAVE BREAKING USING OPENFOAM

Bradley Chambers and Shanti Bhushan
Department of Mechanical Engineering
Center for Advanced Vehicular Systems
Mississippi State University
Starkville, Mississippi, USA

ABSTRACT
The simulation of solitary wave run up on a slope is evaluated using a volume of fluid method in OpenFOAM. The solitary wave is produced at the inlet by the Boussinesq wave equations. The simulated results are compared to experimental and nonlinear potential flow results for a 1 to 15 run up slope with an initial wave height ratio of 0.45. The profile of the wave near breaking is shown to agree well with previous results with the exception of a larger jet tip being calculated by OpenFOAM. The elevation throughout the run up of the wave is recorded and compared to the same data set. The OpenFOAM calculation shows a decreased peak amplitude when compared to experimental results and potential flow calculations. Reynold’s Number, Froude Number, and gravity values where changed and had no positive affect on the simulation results. The grid is refined multiple times and shows negligible affects with further resolution. Calculations for initial wave heights of 0.30 and 0.60 are also computed with OpenFOAM and compared with potential flow results. OpenFOAM’s profile is also compared to potential flow for several lower initial wave heights which result in surging and non-breaking run up on the same 1 to 15 slope used previously. OpenFOAM shows a good agreement with the nonlinear potential flow. The dissipation in large incident wave heights is investigated and a correction is applied to the OpenFOAM results. The corrected data shows a more accurate profile in the breaking region. The results shown indicate that more work needs to be done to improve two phase modeling within OpenFOAM for application to the case of solitary wave run up on a slope.

KEY WORDS: Solitary Wave, Multiphase, OpenFOAM

GLOSSARY
\[ \eta \] – free surface elevation
\[ \H \] – maximum free surface elevation
\[ H_0 \] – initial maximum solitary wave elevation
\[ h \] – still water depth
\[ h_0 \] – initial still water height
\[ U \] – x-direction (streamwise) velocity
\[ W \] – z-direction velocity
\[ c \] – wave celerity
\[ s \] – slope
\[ S_0 \] – breaking prediction slope parameter
\[ g \] – gravity
\[ t \] – time
\[ t^* \] – non-dimensional time \[ t^* \equiv t \frac{g}{h_0} \]
\[ \omega \] – wave frequency

INTRODUCTION
Breaking waves are classified as: plunging, spilling, surging, and collapsing, as shown in Figure 1. Plunging breaking is characterized by the wave forming a tube-like feature that spills onto the forward face of the wave. This spilling of the crest onto the forward face causes a violent crash and large amounts of turbulence during breaking. A wave break caused by the faster moving wave peak continuously spilling on to the front face is classified as a spilling breaking wave. The constant spilling causes a considerable amount of rolling on the front face of the wave. The time scale of breaking in a spilling breaking wave is much larger than that of a plunging breaking wave. The third type of breaking wave, classified as surging, occurs on very steep slopes. Surging breaking waves never spill or plunge; instead the wave just runs onto the shore. Collapsing occurs between surging and plunging. The wave gains amplitude as a plunging wave would on run up, but just collapses and runs up as the surging wave would. Spilling and plunging breaking waves will be the focus of this research, as they are the types of waves applicable to ground vehicle fording and swimming.

Figure 1. Illustration of wave break types from USDOT report FHWA-NHI-09-112 [1].
Wave breaking occurs when a wave reaches a critical steepness or slope. Breaking can be caused by wave run up onto a slope, or it can be from the generation of a wave, which has breaking characteristics. Chou and Ouyang [2] proposed that the wave breaking occurs when the average horizontal velocity at the wave peak (u) is same as the wave celerity (c). The wave celerity is the phase velocity with which a wave crests moves. For shallow water, the celerity of a solitary wave is defined as,

\[ c = \sqrt{gh \left[ 1 + \frac{1}{2} \left( \frac{H}{h} \right) - \frac{1}{4} \left( \frac{H}{h} \right)^2 + \frac{1}{15} \left( \frac{H}{h} \right)^3 - \frac{1}{128} \left( \frac{H}{h} \right)^4 \right]} \]  

(1)

where H is the wave amplitude above the still water level at any instant, h is the still water depth at any given instant, and g is acceleration due to gravity. A visual indicator of a wave reaching its critical point is the forward face of the wave becoming vertical near the crest.

Grilli et al. [3] performed nonlinear potential flow simulations for solitary breaking waves across a range of slopes from 1:8 to 1:100. The computations were performed beyond the break point almost to the point of reconnection to the forward face of the wave. Based on the results, a breaking criterion was proposed,

\[ \frac{H_0}{h_0} > 16.9s^2 \]  

(2)

where s is the slope of the bed, \( H_0 \) is the initial wave amplitude and \( h_0 \) is the initial undisturbed water depth. The above criteria define that only the waves with an initial amplitude more than \( H_0 \) will break at some point on the slope. The type of breaking is related to the slope parameter \( S_0 \):

\[ S_0 = \frac{s}{\sqrt{h_0/g}} \begin{cases} 0.025 < S_0 < 0.3 & \text{Spilling} \\ 0.3 \leq S_0 < 0.37 & \text{Plunging} \\ 0.37 \leq S_0 < 0.45 & \text{Surging} \\ S_0 \geq 0.45 & \text{No Break} \end{cases} \]  

(3)

Li [4] performed experiments of breaking and non-breaking solitary wave run up on slopes and recorded data for wave height, velocity, and profile. Some of the data collected were compared to the numerical results of Grilli et al. [3] for a slope of 1:15. The experimental results showed that the numerical model of Grilli et al. [3] produced a wave that travelled slightly faster and a slightly higher amplitude prediction. The plunging jet shape at the point of reconnection to the front face of the wave showed good agreement with the experiments.

Li’s experiment focused on capturing wave height at the onset of breaking and wave breaking profiles throughout wave breaking. Li compared the experimental results to the result from the potential flow calculations of Grilli et al. [3], and concluded that the potential flow calculations produced an elevated wave height as well as a thicker plunging jet wave profile.

The objective of this study is to further validate OpenFOAM’s application to multiphase flow. Specifically, OpenFOAM’s ability to accurately model viscous flow behavior in the event of wave breaking cases applicable to fording vehicle simulations. The wave behavior was also tested in relation to correctly modeling if the wave would be plunging breaking, surging breaking, or no break run up.

PROBLEM DESCRIPTION

Simulations were performed to evaluate the predictive capability of the numerical schemes and models of OpenFOAM [5] for plunging breaking wave simulations. Two-dimensional simulations were performed for a travelling solitary wave with initial peak elevation of \( H_0 = 0.45h_0 \) over a slope \( s = 1/15 \). The solitary wave boundary conditions were generated from the analytic solution of the first-order Boussinesq equations in shallow water, which gives:

\[ \eta(t) = H_0 \text{sech}^2(-\omega t + X_s) \]  

(4)

where \( \eta \) is the wave elevation, \( \omega \) is the wave frequency, and \( X_s \) is a user specified time lag to allow for the entire wave to enter the domain. The wave frequency is:

\[ \omega = \frac{3}{4} \frac{H_0}{h_0} \left( \frac{1 + \frac{H_0}{h_0}}{g} \right)^{1/2} \]  

(5)

where g is acceleration due to gravity. The wave velocities are specified as:

\[ U(t) = \sqrt{g} \frac{H_0}{h_0} \text{sech}^2(-\omega t + X_s)[1 - \frac{H_0}{4h_0} \text{sech}^2(-\omega t + X_s)] \]

\[ W(z,t) = -\sqrt{g} \frac{H_0}{h_0} \left[ 1 - \frac{H_0}{4h_0} \text{sech}^2(-\omega t + X_s) \right] \]  

(6)

For these flow conditions, \( H_0/h_0 > 16.0s^2 = 0.075 \) and \( 0.025 < S_0 = 0.15 < 0.3 \); thus plunging wave breaking was expected.

Grilli et al. [3] ran a numerous amount of cases for several different slope values. For a 1 to 15 slope, Grilli et al. [3] published run data with initial wave elevation values of 0.06, 0.08, 0.10, 0.30, 0.45, and 0.60. Figure 7 shows a comparison of the profile estimates by OpenFOAM for the 0.06 and 0.10 cases with fully non-linear potential flow calculations of Grilli et al. [3]. The initial wave heights noted should cause surging waves.

The computational domain is shown in Figure 2. Laminar and URANS simulations were performed using grid points in streamwise x and wall normal z directions, respectively. The predictions are compared with experiments of Li [4] and potential flow computations of Grilli et al. [3] using non-dimensional results for valid comparison.

The simulations are performed using 2nd-order accurate time and spatial discretization schemes, PIMPLE [5] for the pressure-velocity coupling, and VoF for air-water interface tracking. The numerical schemes have been previously well validated for single-phase turbulent flows [6] and are investigated herein for two-phase flows. URANS simulations were performed using k-\( \omega \) SST turbulence model.
RESULTS AND DISCUSSION

Five different grid resolutions were tested for a solitary wave run up on a 1 to 15 slope. Table 1 shows a comparison of grid sizes and features. An initial coarse grid was used to implement boundary conditions and validate proper wave generation. A second coarse grid was created with an initial flat region to allow for proper wave profile creation. A medium grid was then used to compare the solitary wave run up to experimental and fully non-linear potential flow. Two final fine grids were made to attempt to accurately model the wave elevation and shape used for validation. One grid was resolved in horizontal and vertical directions evenly; another grid was made with resolution focused in the horizontal direction to alleviate a jagged edge produced on the plunging face of the wave. The fine grids did not improve the accuracy of the solution as the solution was still jagged, and there were no improvements in elevation accuracy; thus, the medium grid was used for its equivalent accuracy and easier computations.

Several different fluid property values were also experimented with to see if the diminished elevation could be remedied. The initial Froude and Reynold’s values where increased and decreased by editing the gravity and density of the simulation. When the OpenFOAM values for gravity and density were edited the properties were not changed within the boundary conditions as it would change the wave shape and characteristics. The change of fluid properties made no change to the elevation of the simulation. The gravity value used for boundary condition calculations was changed to create a milder or steeper slope for the incoming wave. This change only affected the solution by delaying or speeding the break in the x direction, but it did not help with elevation.

Table 1. Comparison of grid and numerical schemes.

<table>
<thead>
<tr>
<th>Grid</th>
<th>Number of Points</th>
<th>Run Up</th>
<th>Numerical Method</th>
<th>Turbulence</th>
<th>Max H/H₀</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coarse</td>
<td>131300</td>
<td>No</td>
<td>Euler</td>
<td>No</td>
<td>0.501</td>
</tr>
<tr>
<td>Coarse Run Up</td>
<td>146450</td>
<td>Yes</td>
<td>Euler</td>
<td>No</td>
<td>0.499</td>
</tr>
<tr>
<td>Medium</td>
<td>396774</td>
<td>Yes</td>
<td>Euler</td>
<td>No</td>
<td>0.503</td>
</tr>
<tr>
<td>Medium</td>
<td>396774</td>
<td>Yes</td>
<td>Least Squares</td>
<td>No</td>
<td>0.506</td>
</tr>
<tr>
<td>Medium</td>
<td>396774</td>
<td>Yes</td>
<td>Crank Nicolson</td>
<td>No</td>
<td>0.5104</td>
</tr>
<tr>
<td>Medium</td>
<td>396774</td>
<td>Yes</td>
<td>Backward</td>
<td>No</td>
<td>0.5187</td>
</tr>
<tr>
<td>Medium</td>
<td>396774</td>
<td>Yes</td>
<td>Backward and Least Square</td>
<td>No</td>
<td>0.5191</td>
</tr>
<tr>
<td>Medium</td>
<td>396775</td>
<td>Yes</td>
<td>Backward and Least Square</td>
<td>Yes</td>
<td>0.5021</td>
</tr>
<tr>
<td>Fine Even Distribution Horizontal</td>
<td>747082</td>
<td>Yes</td>
<td>Least Square</td>
<td>No</td>
<td>0.5189</td>
</tr>
<tr>
<td>Fine Even Horizontal</td>
<td>747082</td>
<td>Yes</td>
<td>Least Square</td>
<td>No</td>
<td>0.5188</td>
</tr>
</tbody>
</table>

Figure 3. Comparison of simulated wave shape for a Euler differencing scheme and Gauss linear gradient scheme ( ) to experimental results of Li [4] ( ) and potential flow calculations of Grilli et al. [3] (▲). Non-dimensional simulation time is listed below each comparison (t*).
Several different time differencing and gradient schemes were tested for better solution accuracy. The first runs used an Euler time differencing scheme and a Gauss linear scheme for gradients. This combination produced the best jet profile, which is shown in Figure 4.

Figure 5 shows the x-direction velocity. It can be seen that the profiles of velocities do not vary much for laminar schemes. The only difference in velocity profile develops from the difference in the shape of the plunging jet tip. In Figure 5 it can be seen that the RANS turbulence scheme predicts somewhat lower velocity profiles. The difference in the plunging jet tip can also be noted in Figure 5.
Since the Euler scheme is of the first order involving the most dissipation, the backward and Crank Nicolson time differencing schemes were used as an attempt to more accurately model wave elevation through the use of higher order schemes. These schemes allowed for an increase in the elevation of the solution, but resulted in the jet tip of the plunging wave to become wider and jagged. A fourth order least squares scheme was run for the gradient scheme as well. A comparison of all the methods run is shown in Figure 4, which displays $H/H_0$ versus $x/h_0$ for an initial wave elevation of 0.45. These comparisons show that the most accurately obtained elevation estimates within OpenFOAM are occurring with a backward differencing scheme and a least square gradient scheme being run together.

Figure 6 plots the magnitude of vorticity for the differencing schemes discussed as well as the RANS turbulent results. The first three time steps shown have similar results for the magnitude of vorticity. The last time step shows Euler and Crank Nicolson producing similar results. A backward differencing scheme produces the most vorticity due to the nature of its breaking profile creating higher velocities and a jagged profile. While the RANS turbulent scheme run produced the same jagged profile the backward scheme produced the magnitude of vorticity was lower than backward for the final time step, but it was greater than Euler or Crank Nicolson. This is due to the lower velocities produced by the turbulent scheme.

Potential flow was run for a numerous amount of cases at several different slope values. For a 1 to 15 slope, potential flow results were published for initial wave elevation values of 0.06, 0.08, 0.10, 0.30, 0.45, and 0.60. Figure 7 shows a comparison of the profile estimates by OpenFOAM for the 0.06 and 0.10 cases with fully non-linear potential flow calculations. The initial wave heights noted should cause surging waves. OpenFOAM closely models the profile, but at the very end of each run the wave has a very slight plunging characteristic.

To quantify the amount of dissipation during run up of the wave a long, flat domain was created consisting of 100 units in the streamwise direction. The boundary condition for the bottom of the domain was set to be slip in order to remove any dissipative effects from viscous interactions. The results of this run are shown in Figure 8 by plotting $H/H_0$ and $U/U_0$ at the wave peak versus $x/h_0$. The results of this run point to numerical dissipation in the OpenFOAM solver. In Figure 9 the peak $H/H_0$ value is tracked throughout run up and plotted versus $x/h_0$ location for initial wave elevations of 0.30, 0.45, and 0.60 along with data from potential flow calculations. All three of these cases demonstrate the correct plunging behavior.
but they also show a diminished wave height for each case. The dissipation found from the flat domain test was used to apply a correction to the plunging wave cases which are also shown in Figure 9. Since solitary waves should not decay without dissipation, and since no condition for the wave to experience dissipation was included, we can conclude that any deviation of the values of $H/H_0$ and $U/U_0$ from 1 would be caused by numerical dissipation.

The difference in $H/H_0$ from the ideal value was added to the cases elevation values for the correction. For an incident wave height of 0.45 the corrected profile agrees well with the breaking region of the experimental results. For an incident wave height of 0.3 the breaking region agrees well with the non-linear potential flow calculations. Both incident wave heights of 0.3 and 0.45 have an initially elevated solution which is not ideal, but if breaking region physics are of interest this solution may be of interest. For an incident wave height of 0.6 the comparison to potential flow is still not ideal due largely to the greater amounts of dissipation shown in higher level incident wave heights.

CONCLUSIONS

OpenFOAM predicts a similar wave profile for a solitary wave run up on a slope. The elevation predictions of OpenFOAM are decreased when compared to both experimental and non-linear potential flow. When adjusted for the dissipation simulated in a flat flow, the solution for simulations involving an incident wave height of 0.45 produce an accurate breaking region, but an initially elevated inflow. The profiles also still fall short of the elevated potential flow calculations. Investigation into the source of dissipation in Volume of Fluid calculations in OpenFOAM should be investigated further in the future.

ACKNOWLEDGEMENTS

Effort sponsored by the Engineering Research & Development Center under Cooperative Agreement number W912HZ-15-2-0004. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Engineering Research & Development Center or the U.S. Government. Technical points of contact for this effort are: Justin Foster (Justin.W.Foster@erdc.dren.mil) and Matthew Farthing (Matthew.W.Farthing@erdc.dren.mil).

REFERENCES


TEMPORAL DIRECT NUMERICAL SIMULATIONS FOR FLAT-PLATE BOUNDARY LAYER

S. Muthu and S. Bhushan
Department of Mechanical Engineering
Center for Advanced Vehicular Systems
Mississippi State University
Starkville, Mississippi, USA

ABSTRACT

“Temporally developing” direct numerical simulations (DNS) for free-stream turbulence induced bypass transition over a flat-plate boundary layer under a zero-pressure gradient are performed to assess the temporally developing approach’s viability as a numerically inexpensive alternative to the commonly used spatially developing approach. A formulation for the domain speed is developed using the integral boundary layer solution, and is validated using spatially developing DNS results. The results show that domain speed is same both for the boundary layer growth and free-stream decay, and periodicity does not significantly affect the transition growth. The growth/decay of the integral boundary layer parameters, and mean and turbulent velocity profiles compare very well with the spatial DNS results, validating the ability of the temporal approach to accurately predict the interaction between the turbulent boundary layer and free-stream turbulence.

KEY WORDS: Bypass Transition, Turbulent Boundary Layer, Pseudo-spectral Methods, Temporal DNS, Turbulence Modeling

INTRODUCTION

Bypass transition from laminar to turbulent conditions is important in many engineering applications, and significantly impacts important flow quantities, such as drag or heat transfer. Bypass transition occurs due to the presence of strong disturbances (high free-stream turbulence, large wall roughness elements, flow separation, pressure gradient effects etc.) [1], and entails strongly nonlinear phenomena. A review of the literature shows that its modeling is not in general mature enough to predict transition flow behavior accurately over a wide range of complex geometries and flow conditions, and remains a critical challenge [2–5]. Direct numerical simulations (DNS) of canonical flows such as flat-plate boundary layers are very useful in understanding the physics of transition flow. DNS have been performed for spatially developing flat-plate simulations under zero-pressure gradient up to Re = 4×10^6 [6–12]. However, even when present these limited datasets are typically for moderate Reynolds numbers (Re_e ~ 10^5) and very high inlet turbulence intensities (Tu_{in} ≥ 5%) [7,8,12]. This is primarily because of high computational costs associated with the spatially developing approach, which requires a relatively large flow domain to resolve the entire region of the boundary layer that contains the relevant flow physics.

An alternative approach is the temporally developing approach, wherein the simulations are performed using a periodic boundary condition in the streamwise direction, such that the simulation domain moves along with the flow, as demonstrated in Fig. 1. This approach has been applied successfully for DNS/ large eddy simulations (LES) of mixing layer and jet flows [13–17]. Spalart [18] proposed numerical approximations to allow a consistent growth of the fast and slow scale turbulent and mean flows, respectively, to maintain streamwise periodicity for developing flows. The temporally developing simulations are less numerically expensive (by 1 to 2 orders of magnitude for high Re) than their spatially developing counterparts, as they require an order of magnitude smaller domain along streamwise direction. Further, they allow straightforward application of the pseudo-spectral solvers (especially FFT in the streamwise direction) without involving a debatable fringe region for purposes of recycling the boundary conditions. Recently, they have been applied for fully developed turbulent flat-plate boundary layer simulations, induced by moving plates [17] and with suction [16]. However, they have not been applied for general problems involving different flow regimes and growth regions, as encountered in bypass transition flows.

![Figure 1: Schematic diagram showing translation of the domain for the temporally developing simulations. The domain moves with velocity V_D.](image-url)
The present work aims to demonstrate that the temporally developing approach is a viable method (can predict as good as spatial DNS to an extent) for transitional boundary layer simulations. The study seeks to answer the four primary numerical issues associated with the temporally developing approach:

(a) Do the temporally developing simulations predict the growth/decay of the laminar, transition, and turbulent boundary layer accurately?
(b) How does periodicity affect the flow during laminar, transition, and turbulent regions?
(c) Does the entire simulation domain move with the same speed, i.e., are the boundary layer growth and free-stream turbulence decay consistent throughout the simulation?
(d) What is the domain translation velocity, which can help relate the temporal results to the spatial results?

To achieve these objectives, temporally developing DNS are performed using a pseudo-spectral solver for free-stream turbulence induced flat-plate boundary layer bypass transition under zero pressure gradient using different domain and time step sizes, and free-stream turbulence intensities $\bar{Tu} = 0.1\%$ to $3.5\%$. To address issue (a), the predictions of the integral boundary layer parameters and mean and turbulent velocity profiles are compared with spatial DNS results [8] (referred to as S-DNS henceforth) and experimental data [19] in the boundary layer coordinates. To address issue (b), the growth/decay of the temporal integral boundary layer is analyzed. To address issue (c), the decay of the free-stream turbulence in spatial coordinates is compared with the spatial DNS results. To address issue (d), a formulation of the domain translation velocity is derived from the integral boundary layer equations. The formulation is validated for the predictions of the growth/decay of the integral boundary layer parameters in the spatial coordinates (or distance along the plate $Re_{x}$) against the spatial DNS results.

### NUMERICAL METHOD AND SIMULATION SETUP

The Navier-Stokes equations are discretized using the pseudo-spectral method, wherein FFT is used along the periodic streamwise ($x$) and spanwise ($z$) directions and Chebyshev polynomials in the wall normal direction ($y$). The discretized equations are solved using a fractional step method, where the equations at every time step are solved in a three-step process. Readers are referred to Bhushan et al. [20] for details.

Temporally developing DNS are started with an initial condition, and allowed to develop. The initial flow condition is obtained by superimposing Blasius mean streamwise velocity ($U_{0}$) at $Re_{x} = 106$ or $Re_{x} = 2.55\times10^{4}$ with turbulent fluctuations following spatial DNS [5,6]. To obtain the initial turbulent fluctuations, first an artificial isotropic turbulence flow field was generated using Rogallo’s approach with von Karman energy spectra and $\bar{Tu} = 5\%$. The artificially generated turbulence was allowed to develop over the flat plate, and the free-stream turbulence decay was monitored. The solution corresponding to the desired $\bar{Tu}$ was used for the generation of the initial turbulent fluctuations. The streamwise and spanwise flow directions involved periodic boundary condition. A no-slip boundary condition was used for the wall. The free-stream boundary condition was:

$$\frac{\partial u}{\partial y} = 0; \frac{\partial w}{\partial y} = 0; \text{and } \frac{\partial v}{\partial y} = -\left(\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z}\right)$$

where, $u$ is the streamwise velocity, $v$ is the wall normal velocity and $w$ is the spanwise velocity. This produced almost identical mean velocity and skin friction to those obtained with the Blasius boundary condition used in the spatial DNS [8,9].

### Table 1: Details of the simulations performed in the study.

<table>
<thead>
<tr>
<th>Case</th>
<th>$\bar{Tu}$</th>
<th>Domain Length ($x10^{4}$)</th>
<th>Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>2.8%</td>
<td>$20\delta_{0}$</td>
<td>Validate growth/decay of integral boundary layer, and mean and turbulent velocity profiles using spatial DNS and experimental data</td>
</tr>
<tr>
<td>4a</td>
<td>0.01%</td>
<td>$20\delta_{0}$</td>
<td>Evaluate the effect of $\bar{Tu}$ on the $V_{0}$ formulation.</td>
</tr>
</tbody>
</table>

Herein, four sets of simulations are performed (Table 1). The first set uses five different domain sizes (a) $20\delta_{0}$, (b) $25\delta_{0}$, (c) $30\delta_{0}$, (d) $40\delta_{0}$, and (e) $50\delta_{0}$ domains, where $\delta_{0}$ is the initial boundary layer thickness. The simulations are performed using a time step size $\Delta t = 0.0005L/U_{0}$, where $L$ is the simulation domain length in the streamwise direction and $U_{0}$ is the mean free-stream velocity. The cases (a) through (e) correspond to $Re_{x} = 16.25\times10^{3}, 20.32\times10^{3}, 24.38\times10^{3}$, $32.5\times10^{3}$, and $40.64\times10^{3}$, respectively. Note that the grid and domain size is about an order of magnitude smaller than the spatial DNS [8,9]. The inlet turbulence intensity was $\bar{Tu} = 3.5\%$ to match the conditions from the spatial DNS data [8,9]. The second set of simulations are performed for $20\delta_{0}$ domain $\bar{Tu} = 3.5\%$ but with different step sizes of (a) $\Delta t = 0.00033L/U_{0}$, and (b) $\Delta t = 0.00075L/U_{0}$. The third set of simulations are performed for lower turbulence intensity $\bar{Tu} = 2.8\%$ for the domain sizes of (a) $20\delta_{0}$, (b) $30\delta_{0}$, and (c) $40\delta_{0}$. Case (4) was performed for very low $\bar{Tu} = 2.8\%$ for the domain sizes of $20\delta_{0}$. All the simulations are performed using a grid size of $192\times193\times192$.

The result for case 1(a) is primarily used for the validation of the growth/decay of the integral boundary layer (skin friction coefficient $C_{f}$, displacement thickness $\delta'$, momentum thickness $\theta'$, and shape parameter $H = \delta'/\theta'$, and mean and turbulent velocity profiles in laminar, transition and turbulent regions using S-DNS and experimental data in boundary layer coordinates, i.e., with respect to Reynolds number based on momentum thickness $Re_{\theta}$. The predictions for cases 1(a-e) are correlated with the spatial location traversed by the simulation.
domain, and the integral boundary layer and free-stream turbulence decay predictions are compared with the S-DNS to validate the $V_D$ formulation. Results from cases 1 and 2 are used to evaluate the robustness of the $V_D$ formulation on different domain sizes. Cases 3 and 4, provide further evaluation of the $V_D$ formulation on turbulence intensity, including those for the laminar flow.

**RESULTS AND DISCUSSION**

**Validation in Boundary Layer Coordinates (Re₀).** The growth/decay of $C_f$, Reynolds number based on displacement thickness $Re_0 ≈ H$ and $H$ is shown with respect to $Re_0$ for all domain and time step sizes in Fig. 2. The $C_f$ profile for $Tu_{in} = 3.5\%$, shows an initial decrease up to $Re_0 = 300$ (laminar region), then shows gradual increase up to $Re_0 = 650$ (transition region), and shows an unsteady behavior with gradual decay for larger $Re_0$ (turbulent region). The predictions compare quite well with the spatial DNS and experimental data. Note that the temporal results show unsteadiness in the turbulent region, whereas S-DNS is steady. This is because the former shows the instantaneous (plane averaged) values, whereas the latter show the ensemble averaged values. The results on different domain sizes show similar patterns in the transition and turbulent regions, and the small variations could be attributed to the changes in grid resolution and free-stream turbulence length scale. However, they show significant differences in the laminar region. The smaller domains somewhat over predict $C_f$, whereas the larger domains compare better. However, note that the simulations on smaller $20δ_0^3$ domains using lower TIs, i.e., both the laminar case $Tu_{in} = 0.01\%$ and $Tu_{in} = 2.8\%$, compare well with the laminar profile. Step time size does not show much effect on the boundary layer growth.

$$\frac{d\delta^*}{dt} = \frac{C_f}{2}$$

As shown in Fig. 3 (a), the predictions satisfy Eq. (1) very well qualitatively. The percentage error in the predictions for $\frac{d\delta^*}{dt} - \frac{C_f}{2}$ over $C_f/2$ are determined using a $L^1$ norm on all five domain sizes in the laminar ($106 \leq Re_0 \leq 350$), transition ($350 \leq Re_0 \leq 650$), and turbulent ($650 \leq Re_0 \leq 1050$) regimes. Based on the bar chart (Fig. 3 b) we can infer that the $20δ_0^3$ domain size has the largest error in all three regimes and is not large enough to contain all the near-wall streaky structures. The error decreases as the domain size increases. The larger domain sizes ($40δ_0^3$ and $50δ_0^3$) have a lower error than $20δ_0^3$ but their grid resolution is too coarse in the larger domain sizes to resolve all scales in the turbulence regime. Hence, the error starts increasing again. $30δ_0^3$ is the optimum domain size to maintain periodicity with an error $\leq 2\%$ on all three regimes. Thus, the periodic boundary condition does not affect the flow significantly, suggesting that the spatial growth within the domain is small.

As discussed earlier, the above temporal integral boundary layer analysis demonstrates that the streamwise domain size requirements in the laminar region depend on the free-stream turbulence intensity, i.e., higher intensities require larger domain sizes. This is because the near-wall streaky structures are expected to be more elongated for transition at lower $Re$ than those at higher $Re$ [21]. Thus, the simulations for $Tu_{in} = 3.5\%$ case requires larger domain size than the $Tu_{in} = 2.8\%$ case to capture the streaky structures as well as the large-scale motions.

Comparison of the mean and turbulent velocity profiles at selected laminar, transition and turbulent locations in Figure 4 show good agreement with S-DNS and experimental data.

**Validation in Plate Length Coordinates (Reₜ).** To compute $Reₜ$, from the solution times, the domain translation velocity is required. Overall, the exact form of domain velocity is not known, and is a topic of research. From linear theory (for free-surface wave packet), the waves move with a group speed $U_g = U_P/2$, where $U_P$ is the phase speed of the wave packet. Based on this, we expect the domain to move with a velocity $V_D = U_P/2$ due to mean flow. However, for turbulent flows domain translation is also expected due to turbulent velocity. Recently, Bobke et al. [16] performed temporally developing LES for turbulent flat plate boundary layer with suction. They
estimated the domain translation velocity to be \( V_D = 1.3 U_0 \) by comparing spatial and temporal predictions.

\[ V_D = U_0 \]

and demonstrated that in the fully developed turbulent region the \( C_f \) vs \( \text{Re}_\theta \) is independent of the initial condition. However, a close inspection of the results shows that in the turbulent regime, the scatter in the prediction of the skin friction coefficient is more when plotted against \( \text{Re}_\theta \) compared to \( \text{Re}_0 \) or \( \text{Re}_\theta^\ast \). He and Seddighi [22] performed temporally developing DNS of a transient channel flow. They determined the domain velocity by best-fitting the Blasius solution with the early response of the transient flow and found it to be \( V_D = 0.74 U_0 \).

The domain translation velocity can be derived from the approximate solution to the boundary-layer equations using momentum integral approach for zero pressure gradient (Appendix II). This gives,

\[ V_{D,MI} = \frac{U_0}{2} \frac{\partial}{\partial t} (\theta - U') \]  

(2)

Figure 3: (a) Variation of \( \frac{d\delta^*}{dt} \) (Primary axis) & \( \frac{C_f}{2} \) (Secondary axis) vs \( \text{Re}_\theta \) obtained for \( \text{Tu}_0 = 3.5\% \), for different domain sizes and \( \Delta t = 0.0005 L/U_0 \). (b) \( L^1 \) error analysis for all domain sizes for the temporal simulations using the integral boundary layer equation

\[ \text{Laplacian} \frac{\partial}{\partial x} (U - U_0 - \theta') + \frac{dV_D}{dt} = 0 \]

Figure 4: Comparison of mean velocity profile (a) and streamwise turbulent velocity fluctuations (b) in laminar, transition, and turbulent regimes \( (\text{Tu}_0 = 3.5\%, \text{ domain size } = 30\delta_0 \text{, and } \Delta t = 0.0005 L/U_0) \). Note that the plots are not at same \( \text{Re}_0 \), as transition location is different for our simulations compared with experimental data [19] and S-DNS [8].

Kozul et al. [17] performed temporally developing DNS for the fully developed turbulent boundary layer, wherein the moving plate governed the flow. They used domain speed \( V_D = U_0 \) and demonstrated that in the fully developed turbulent region the \( C_f \) vs \( \text{Re}_\theta \) is independent of the initial condition. However, a close inspection of the results shows that in the turbulent regime, the scatter in the prediction of the skin friction coefficient is more when plotted against \( \text{Re}_\theta \) compared to \( \text{Re}_0 \) or \( \text{Re}_\theta^\ast \). He and Seddighi [22] performed temporally developing DNS of a transient channel flow. They determined the domain velocity by best-fitting the Blasius solution with the early response of the transient flow and found it to be \( V_D = 0.74 U_0 \).

The domain translation velocity can be derived from the approximate solution to the boundary-layer equations using momentum integral approach for zero pressure gradient (Appendix II). This gives,

\[ V_{D,MI} = \frac{U_0}{2} \frac{\partial}{\partial t} (\theta - U') \]  

(2)

Figure 5: \( \text{Re}_\theta \) vs \( \text{Re}_\theta^\ast \) variations (LEFT) and \( \text{Tu}_0/\text{Tu}_0 \) decay along the plate length (RIGHT) for \( \text{Tu}_0 = 3.5\% \) obtained on different domain sizes using (a) \( V_D = \frac{1}{2} U_0 \), (b) \( V_{D,MI} \).

In Figure 5, \( \text{Re}_\theta \) growth in the boundary layer and free-stream turbulence \( (\text{Tu}/\text{Tu}_0) \) decay along the plate length is compared using constant domain velocity formulation \( (V_D = U_0/2) \) and the domain velocity derived from momentum integral approach \( V_{D,MI} \). The results obtained using \( V_D \) compare very well with the S-DNS and analytic profiles in the laminar region. This is expected, as the laminar region does not have turbulence. But these regimes start to deviate from each other and from the S-DNS and analytic profiles starting from the transition region. A similar behavior is observed for the free-stream turbulence decay.

The results obtained using \( V_{D,MI} \) show very good agreement for the \( \text{Re}_\theta \) vs \( \text{Re}_\theta^\ast \) with S-DNS and analytic profiles, and results on all the domains collapse together. A similar behavior is observed for the free-stream turbulence decay. Thus, the results demonstrate that \( V_{D,MI} \) is same for both the boundary layer and the free-stream. Figure 6 shows the variation of \( V_{D,MI} \) during the simulation. All the domains show similar velocities, wherein the \( V_{D,MI} = 0.45 U_0 \) in the laminar
region, close to the linear theory estimate of $0.5U_0$. The domain velocity starts to increase towards the end of the laminar regime, and peaks somewhere in the middle of transition, decreases thereafter and achieves a quasi-steady value $V_{D,MI} = 0.7U_0$ in the turbulent region. The domain velocity in the turbulent region is found to be consistent with the estimates of the He and Seddighi [22] study.

CONCLUSIONS AND FUTURE WORK

Temporally developing DNS for free-stream turbulence induced bypass transition over plate boundary layer under zero pressure gradient are performed, and the numerical issues associated with the approach are investigated. The predictions of the integral boundary layer parameters and mean and turbulent velocity profiles in the boundary layer coordinates compare very well with the S-DNS results and experimental data. This validates the ability of the temporal approach to accurately predict the interaction between the turbulent boundary layer and free-stream turbulence. The temporal integral boundary layer analysis shows that the periodic boundary condition effects and the grid size requirements decrease and increase, respectively, with the increase of the domain size. The optimal domain size is found to be $30\delta_0^3$ for which errors due to periodicity are $\leq 2\%$.

A formulation of the domain translation velocity is derived from the integral boundary layer equations, which shows good agreement with the S-DNS for the prediction of the growth/decay of the integral boundary layer parameters in the spatial coordinates (or distance along the plate $Re_\theta$). Further, the domain speed was found to be same both for the boundary layer growth and free-stream decay. Overall, results demonstrate that temporally developing simulations are a viable inexpensive alternative to the spatial approach for transition flow physics simulations.

ACKNOWLEDGEMENTS

This effort was partially supported by the Center for Advanced Vehicular Systems. All simulations were performed on Talon HPC system at High Performance computing Collaboratory, Mississippi State University.

REFERENCES


**APPENDIX I – INTEGRAL BOUNDARY LAYER EQUATION FOR TEMPORALLY DEVELOPING FLOWS**

For temporally developing boundary layer at zero-pressure gradient, assume that

\[ \frac{\partial}{\partial x} = 0 \]

Thus, boundary layer equation is:

\[ \frac{\partial \bar{u}}{\partial t} + \bar{v} \frac{\partial \bar{u}}{\partial y} = \frac{\partial}{\partial y} \left( \bar{v} \frac{\partial \bar{u}}{\partial y} \right) - \left[ \frac{\partial \bar{u} \bar{v}'}{\partial x} + \frac{\partial \bar{u} \bar{v}'}{\partial y} \right] \]

Integrating the above equation over the boundary layer:

\[ \frac{\partial}{\partial t} \int_{0}^{\delta} \bar{u} \, dy + \bar{v} \delta \bar{u}\bigg|_{\delta} - \int_{0}^{\delta} \bar{u} \frac{\partial \bar{v}}{\partial y} \, dy = - \frac{1}{\rho} \tau_w - \bar{u} \bar{v}' \bigg|_{0}^{\delta} \]

\[ \Rightarrow \frac{\partial}{\partial t} \int_{0}^{\delta} \bar{u} \, dy = - \frac{1}{\rho} \tau_w \]

\[ \delta^* = \int_{0}^{\delta} \left(1 - \frac{\bar{u}}{U_0}\right) \, dy \]

\[ \Rightarrow \frac{\partial}{\partial t} (\delta^* - \delta) = \frac{1}{\rho U_0} \tau_w \]

**APPENDIX II - V_0 BASED ON THE MOMENTUM INTEGRAL METHOD (2D TURBULENT BOUNDARY LAYER EQUATION)**

\[ \frac{\partial \bar{u}}{\partial t} + \bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} = - \frac{1}{\rho} \frac{\partial \bar{p}}{\partial x} + \frac{\partial}{\partial y} \left( \bar{v} \frac{\partial \bar{u}}{\partial y} \right) - \left[ \frac{\partial \bar{u} \bar{u}'}{\partial x} + \frac{\partial \bar{u} \bar{v}'}{\partial y} \right] \]

For spatially developing steady state boundary layer at zero-pressure gradient:

\[ \frac{1}{\rho} \frac{\partial \bar{p}}{\partial x} = 0 \]

\[ \Rightarrow \bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} = \frac{\partial}{\partial y} \left( \bar{v} \frac{\partial \bar{u}}{\partial y} \right) - \left[ \frac{\partial \bar{u} \bar{u}'}{\partial x} + \frac{\partial \bar{u} \bar{v}'}{\partial y} \right] \]

Integrating the above equation over the boundary layer:

\[ \int_{0}^{\delta} \bar{u} \frac{\partial \bar{u}}{\partial x} \, dy + \int_{0}^{\delta} \bar{v} \frac{\partial \bar{u}}{\partial y} \, dy = - \int_{0}^{\delta} \bar{u} \frac{\partial \bar{v}}{\partial y} \, dy - \int_{0}^{\delta} \frac{\partial \bar{u} \bar{v}'}{\partial x} \, dy \]

\[ = \frac{1}{\rho} \tau_w - \frac{\partial}{\partial x} \left( \bar{u} \bar{u}' \bigg|_{0}^{\delta} - \bar{u} \bar{v}' \bigg|_{0}^{\delta} \right) \]

where, \( \tau_w = - \frac{\partial}{\partial y} \left( \bar{u} \bar{v}' \bigg|_{0}^{\delta} \right) \). The other terms are simplified as below:

(1) \( \bar{u} \bar{v}' \bigg|_{0}^{\delta} = U_0 \bar{v}_0 \)

From the continuity equation:

\[ \bar{v}_0 = - \int_{0}^{\delta} \frac{\partial \bar{u}}{\partial x} \, dy \]

\[ \bar{u} \bar{v}' \bigg|_{0}^{\delta} = \frac{\partial}{\partial x} \int_{0}^{\delta} U_0 \bar{u} \, dy \]

(2) \( \bar{u} \bar{v}' \bigg|_{0}^{\delta} = 0 \)
(3) \(- \int_0^\delta \frac{\partial u}{\partial y} dy = \int_0^\delta \frac{\partial u}{\partial x} dy = \int_0^\delta \frac{1}{2} \frac{\partial u^2}{\partial y} dy\)

\[ \Rightarrow \frac{\partial}{\partial x} \int_0^\delta \bar{u}^2 dy - \frac{\partial}{\partial x} \int_0^\delta U_0 \bar{u} dy = -\frac{1}{\rho} \tau_w \frac{\partial}{\partial x} \int_0^\delta \bar{u} \bar{u}' dy \]

\[ \Rightarrow \frac{\partial}{\partial x} \int_0^\delta \bar{u} (\bar{u} - U_0) dy = -\frac{1}{\rho} \tau_w \frac{\partial}{\partial x} \int_0^\delta \bar{u} \bar{u}' dy \]

\[ \theta = \int_0^\delta \frac{\bar{u}}{U_0} \left( 1 - \frac{\bar{u}}{U_0} \right) dy \Rightarrow \frac{\partial \theta}{\partial x} = \frac{1}{\rho U_0^2} \tau_w + \frac{1}{U_0 \partial x} \int_0^\delta \bar{u} \bar{u}' dy \]

\[ \frac{1}{U_0^2} \int_0^\delta \bar{u} \bar{u}' dy = U' \quad \Rightarrow \frac{\partial}{\partial x}(\theta - U') = \frac{1}{2} C_f \]

Assuming domain moves with velocity \(V_D\): \(x = V_D t\)

\[ \frac{1}{V_D} \frac{\partial}{\partial t}(\theta - U') = \frac{1}{2} C_f \]

\[ \Rightarrow V_D = \frac{1}{2} C_f \frac{\partial}{\partial t}(\theta - U') \]
ASSESSING THE FLOW FIELD PHYSICS CAPTURING CAPABILITIES OF THE INTEGRO-DIFFERENTIAL SCHEME WHEN SOLVING THE NAVIER STOKES EQUATIONS NUMERICALLY

Julio Cesar Mendez, David N.D. Dodoo-Amoo
Mechanical Engineering Department
North Carolina A&T State University
Greensboro, NC, U.S.A

Mookesh Dhanasar, Frederick Ferguson
Mechanical Engineering Department
North Carolina A&T State University
Greensboro, NC, U.S.A

ABSTRACT
Currently, Computational Fluid Dynamics (CFD) plays an important role in the engineering design industry. Applications vary greatly, ranging from the aerospace industry for the design of missiles and hypersonic vehicles to the biomedical industry for the design of artificial lungs. CFD is now providing reasonable information that along with experimental data improves the understanding of complex fluid phenomena. However, current CFD numerical techniques are neither accurate, efficient nor inexpensive. In fact, most numerical techniques lead to large errors, time consuming grid creation methods and many hours of High Performance Computing (HPC) resources. In addition, current solution methods do not provide unique indications of the flow physics within the solution domain. An important aspect of this research is coupling the primitive numerical solution to specially created Flow Physics Extraction Functions (FPEF). These functions can provide direct evidence of complex flow field interactions.

The objective of this study is to demonstrate the capabilities of a new scheme called Integro-Differential Scheme (IDS) [1] capable of overcoming the current limitations of CFD. Further, this paper focuses on the solution of two complex flow interaction problems using the IDS under realistic Reynolds number conditions. The solutions obtained herein demonstrate the two most important characteristics of the IDS capabilities; its accurate physics capturing capabilities, and its efficiency when executed on HPC platforms.

KEY WORDS: CFD, Hypersonic Flows, Conservation Laws, Validation, Shock Interactions.

INTRODUCTION
The equations that govern fluid flows are described by the conservation of mass, momentum and energy principles. These equations, when coupled, are known as the Navier-Stokes Equations (NSE). The coupled NSE form a set of nonlinear partial differential equations which lend themselves to analytical solutions only under 'highly simplified' scenarios. In general, the mathematical behavior of the NSE is complex and unpredictable. It is not possible to categorize the NSE strictly as hyperbolic, elliptic and parabolic types, as the NSE is very sensitive to the properties of its boundary and initial conditions. The very nature of the NSE makes the implementation of most numerical methods very unpredictable when they are tasked in providing the solutions.

Many numerical procedures have been proposed, ranging from laminar to turbulent, and from subsonic to hypersonic. However, there is not a unique method capable of solving a wide range of problems. One commonly acceptable numerical procedure for solving the NSE is the method of directly resolving all the temporal and spatial scales. This approach is known as Direct Numerical Simulations (DNS). However, the computational cost associated with the DNS approach is prohibitively expensive. In fact, for three dimensional problems, the computational cost is determined to be a function of the Reynolds number, and can be expressed in the order of $N$, such that, $N = O(Re^{9/4})$ [2]. Consequently, solving fluid flow problems for realistic engineering applications will not be routinely possible in the near future using DNS. Other numerical approaches, such as: Reynolds Averaged Navier Stokes Equations (RANS) and Large Eddy Simulations (LES) are less computationally expensive but unreliable. Although LES has been demonstrated to deliver more accurate results, they are limited to flows at low and moderate Reynolds numbers. In addition, these numerical models typically introduce errors that are difficult to quantify [3, 4]. On the other hand, current experimental facilities are not always capable of simulating the solution required by aircraft designers and propulsion engineers.

In an effort to overcome the shortcomings mentioned above, Ferguson et al. [1] introduced a new methodology called the Integro-Differential Scheme (IDS) that preserves the advantages of the finite volume and finite difference methods. Although the numerical details of the IDS are beyond the scope of this technical paper, interested readers are referred to [5]. Nevertheless, the IDS is potentially accurate, efficient and inexpensive and is applicable of solving a wide range of fluid dynamic problems under realistic boundary conditions.
The objective of this research is to qualitatively and quantitatively verify the physics predicted by the IDS solution when it is applied to the NSE for high Reynolds numbers. Previous studies [1, 6, 7] have shown that the IDS scheme is accurate solving a wide variety of flows including the Lid Driven Cavity flow, which is a laminar subsonic flow. However, these studies [1, 6, 7] relied on traditional methods that are geared towards desktop computing. In addition, these approaches relied on qualitative speculations that were inferred from the behavior of the primitive variables. These approaches did not effectively demonstrate the physics capturing capabilities of the IDS. In general, the previously established techniques that are based on the manipulation of the primitive variables cannot be relied upon to accurately detect shock and expansion waves, and other flow field phenomena. Rather, direct methods that are designed to extract meaningful information from CFD solutions are preferred. To this end, a set of Flow Physics Extraction Functions (FPEF) are created and implemented with the primary objective of demonstrating the flow physics capturing capabilities of the IDS. In this paper, this CFD methodology is applied to problems that contain complex flow physics at high Reynolds and Mach numbers.

THE NAVIER-STOKES EQUATIONS

In this research project, the NSE are of paramount importance and they are listed as follows:

\[
\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_i} = 0 \tag{1}
\]

\[
\frac{\partial \rho u_i}{\partial t} + \frac{\partial (\rho u_i u_k + p \delta_{ik} - \tau_{ik})}{\partial x_k} = 0 \tag{2}
\]

\[
\frac{\partial (\rho e)}{\partial t} + \frac{\partial (\rho (e + RT) u_i - \tau_{ik} u_i + q_k)}{\partial x_k} = 0 \tag{3}
\]

respectively. In equations (1–3) the symbols; \( \rho \), \( u_k \), \( t \), represent the density, the velocity components of an elementary control fluid element, and time, respectively. In addition, the symbols \( e \), \( p \), \( \tau_{ik} \) and \( q_k \), in equations (1-3) represent the internal energy, the pressure, the stress tensor and the heat flux associated with an elementary control volume, respectively. In this research, internal energy, pressure, stress tensor and heat flux are defined by:

\[
e = C_v T + \frac{1}{2} u_i u_i \tag{4}
\]

\[
P = \rho RT \tag{5}
\]

\[
\tau_{ik} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \delta_{ik} \frac{\partial u_i}{\partial x_j} \tag{6}
\]

\[
q_k = -k \frac{\partial T}{\partial x_k} \tag{7}
\]

respectively. In equation (5), \( R \) is the gas constant. The symbols \( \mu \) and \( k \) represent the viscous and thermal properties of the fluid of interest. For air, the viscosity of the fluid is evaluated using Sutherland’s law and the thermal conductivity expression,

\[
k = f(T) \tag{8}
\]

is provided. In the case of 3D fluid flows, the NSE (1-8) can be treated as a closed system of five equations relative to 5 unknowns. These unknowns are the primitive variables: \( \{ \rho, u_k, T \} \), \( k = 1, 2, 3 \). To obtain a unique solution initial and boundary conditions must be provided.

FLOW PHYSICS EXTRACTION FUNCTIONS (FPEF)

The solution of equations (1 - 3) merely provides information about the primitive variables. For 2D problems, these variables are: \( \rho, u, v \) and \( T \). Although these variables are derived from the conservation laws, they do not provide direct evidence of the flow field physics; evidence such as, shock and expansion waves or even the vortical structures. However, an accurate prediction of the flow features is mandatory, and must be done in an objective and consistent manner.

In efforts to identify the associated flow physics predicted by the IDS procedure, specialized PFEF were created. Besides, the FPEF are tailored to extract the physics directly from the predicted primitive variables. Therefore, the techniques presented in this research are rigorous and unbiased. Similar analyses were performed by [8-12]. However, this research introduces two new functions that are based on kinematics and thermodynamics properties. These functions are described in equations (9) and (10). Equation (9) represents the gradient of density in the direction of the velocity [9], and (10) denotes the Mach number normal to shock waves [8].

\[
\frac{d\rho}{dn} = \nabla \rho \cdot \mathbf{T} \tag{9}
\]

\[
Ma_n = \frac{Ma \cdot \nabla p}{|\nabla p|} = 1 \tag{10}
\]

In this analysis, equations (9) and (10) are used to locate the shock waves within the flow field. Since pressure and density are the variables that change the most through shock waves, they are also effective in shock detection. In some cases, however, false indications may occur, so a small degree of filtering is required [12]. Although this study used equation (9) without the need for filtering, the filtering criteria proposed by [12] with a threshold of \( \epsilon = 0.007 \) is used in equation (10).

Additionally, two FPEF functions, based on the combined kinematic and thermodynamic behavior of the fluid, are introduced. These functions are described in equations (11 - 12), and they represent the magnitude of entropy gradient and the Q-criterion, respectively.

\[
|\nabla S| = \sqrt{u_i \left( \frac{\nabla \times \mathbf{F}}{T} \right) \frac{\partial h_0}{\partial x} \frac{\partial h_0}{\partial x} + u_i \left( \frac{\nabla \times \mathbf{F}}{T} \right) \frac{\partial h_0}{\partial y} \frac{\partial h_0}{\partial y}} \tag{11}
\]

\[
Q = \frac{1}{2} \left( \left\| \mathbf{q} \right\|^2 - \left\| \mathbf{s} \right\|^2 \right) \tag{12}
\]
In equation (11) the symbol $h_0$ is the stagnation enthalpy, and $||\Omega||$ and $|S|$ represent the Euclidean norm of the vorticity and rate of strain tensor, respectively. It is of interest to note that equation (11) is based on Crocos’ theorem [13], and as such, it is applicable only in inviscid flow regimes. Nevertheless, equation (11) is extremely effective in detecting shock and expansion waves, as they are inviscid phenomena. On the other hand, equation (12) which was introduced by [14] provides information about the vector field topology, and it represents a local balance between shear strain and vorticity. Using equation (12) regions with dominant rates of strains and vorticity are detected.

THE IDS PHYSICS CAPTURING CAPABILITY

To demonstrate the physics capturing capability of the IDS, two established high Reynolds number fluid dynamic problems were numerically solved. The problems of interest in this study are:

1. Hypersonic flow over a flat plate, and
2. The hypersonic flow cross jet interaction problem.

Although similar problems were solved in [7], in this study revised solutions are warranted. Unlike the solutions generated in [6], the studies conducted herein were done with improved IDS capability. At this stage, the IDS has both FPEF and cutting edge parallel libraries (MPI). These new features unmasked important flow characteristics that were not shown before; due to merely technical limitations. In addition, a no turbulence model is used in this research study.

HYPERSONIC FLOW OVER A FLAT PLATE

The hypersonic flat-plate problem is of interest to this analysis, since it encompasses the two most important flow regimes of interest to fluid dynamists; namely, kinetic flow and continuum flow regimes [15]. At hypersonic conditions, which are of interest to this paper, a strong leading edge shock emanates from the leading-edge of the plate. This shock is called a ‘bow shock’, due to its characteristic curvature. The region between the surface and the shock wave is called the ‘shock layer’ [16], refer to Figure 1. Further, the shock layer is divided into two sublayers, each dominated by either inviscid or viscous effects. The sub-layer closest to the plate surface is known as the ‘boundary layer’, and the outer sublayer is the so-called ‘entropy layer’. Typically, the boundary layer undergoes an important transition; usually from a laminar to a turbulent boundary layer. However, this phenomenon is not the focus of this study.

The flow along the plate can also be characterized by two regions; one very near the leading edge and another farther away. In the leading-edge region, the viscous-inviscid interactions are very strong, and they affect both sublayers: the inviscid entropy as well as the viscous boundary sublayers. Further, this strong interaction results in the merging of the ‘entropy’ and ‘boundary’ layers. In contrast, farther away from the leading edge, the viscous-inviscid interaction is weak, and the two sub-layers remain separated. The two zones that are mainly characterized by the ‘inviscid-viscous’ interactions are referred to as the ‘strong’ and ‘weak’ interaction regions, respectively. The flow phenomena in the ‘strong’ and ‘weak’ interaction regions at the leading edge of the hypersonic plate problem are of paramount importance to this analysis. Moreover, the flow physics in these regions are challenging to predict, and are only determined by CFD tools capable of resolving sharp gradients while negotiating systems of partial differential equations of varying types. In other words, not only are the grids expected to be extremely fine to fully resolve the sharp gradients manifesting in these regions, the CFD schemes are also expected to remain computationally stable, accurate and timely.

The case of a hypersonic flow over a flat plate of length 1 meter is considered. The freestream Mach number is set at 8.6, the Reynolds number set to $3.475771 \times 10^6$ (based on the length of the plate), the Prandtl number to 0.70 and the specific heat ratio, $\gamma$, to 1.4. The freestream density, temperature, viscosity, and pressure were assumed to be $2.2497 \times 10^{-2}$ kg/m$^3$, 360 K, $2.117 \times 10^{-5}$ kg/ms and 2324.39 Pa, respectively. The boundary conditions are set as follows; at the inflow and the far field boundaries, the primitive variables are assigned to their free stream values. At the exit plane, the flow primitive variables are extrapolated. The bottom boundary is divided in three sections, representing the stationary wall (the flat plate), the leading and trailing edge gaps. A symmetric boundary condition is imposed to the leading and trailing edge gap. A no-slip boundary condition and a fixed wall temperature are assigned to the stationary wall, whereas density is extrapolated from the inner nodes. The temperature at the bottom wall is assigned as 1.0.

![Figure 1. Illustration of the Flat Plate Problem [17]](image)

In efforts to obtain a grid independent solution, 5 sets of grids with resolutions ranging from 1,001 by 1,001 to 5,001 by 16,001 nodes in the streamwise and vertical direction, respectively, were studied. It is of interest to note that the gradients in the direction normal to the wall are stronger than those in the direction parallel to the wall, as such thinner grids were placed in the vertical direction [18]. In addition to the 5 sets of grids described above, an extra case, termed the modified grid, was also considered. This was done in an effort to more thoroughly evaluate the IDS capabilities in predicting the viscous dissipation effects inside the boundary layer. In the case of the modified grid, the height of the domain was reduced by half, resulting in an equivalent grid size of 6,001 by 32,001.
This reduction in height effectively reduced the cell height by a factor of 2, resulting in a finer set of grids with relative constant computational load.

The IDS flow field solutions resulting from the above described grid independence study are summarized in figures 2 and 3. Note, that the modified grid is represented by the grid size of 6,001 by 16,001*. The results presented in figures 2 and 3 seem to suggest that grid independence was obtained for a grid size of 6,001 by 16,001*. Figure 2 illustrates the streamwise velocity component in the y-direction at a location of x = 0.5 meters from the leading edge. A careful observation of the data demonstrates that the height of the boundary layer is approximately 0.0088 of non-dimensional units. In addition, no evidence of shock is shown in figure 2. Similarly, figure 3 depicts the temperature profile at 0.5 meters from the leading-edge. However, in this case, the effect of viscous dissipation within the boundary layer is clearly demonstrated [16]. As noted in figure 3, the temperature increases from the outer edge of the boundary layer towards the wall, reaching two peaks. The outer peak indicates the presence of a shock and while the inner peak indicates the effects of boundary layer dissipation. Similar trends were also found in [19, 20].

A closer observation of figure 3, reveals the existence of the two expected sub-layers; namely the 'entropy' layer and the 'viscous boundary' layer. As supported by figure 3, although the height of the shock wave was fully resolved with mesh sizes; 4,001 by 8,001; 5,001 by 16,001 and 10,001 by 16,001*, the dissipation effects were clearly not. The conclusion herein, is that the boundary layer needs an extremely finer set of grids to resolve its physics, when compared to the mesh size needed to resolve the entropy layer and the shock wave. In efforts to fully predict the dissipative effects within the hypersonic boundary layer, the grid independent studies associated with this problem continues.

Figure 4 shows the flow field distribution of the normal Mach number in the direction of the pressure gradient, as described through the use of the FPEF described earlier in this paper to [8]. In this case, the shock is extracted using equation (10) for regions where the value of the FPEF is close to 1.0 in accordance with [8]. It can be observed that the bow shock starts slightly ahead of the leading-edge tip and it certainly displays the 'characteristic' curvature. This characteristic was also reported in [15]. In addition, figure 4 accurately predicts the growth of the boundary layer.

More importantly, the similarities of the predicted features illustrated in figure 1 closely match the IDS computed results illustrated in figure 4. Of greater significance is the fact that the two interaction zones; namely the 'strong' and 'weak' inviscid-viscous interactions zones, merged and all, are vividly computed. In figure 5, the contour plot of the Q-criterion, obtained using the FPEF defined by equation (12), is presented.
The strain dominates over vorticity near the wall; consequently, the Q-criterion becomes negative close to the wall. The Q-criterion behavior observed in this analysis is typical within viscous sub layers where the shear stress is laminar [21]. A second layer, named the turbulent layer, where the swirling motions common in turbulent flows makes the Q-criterion positive. In this region, viscosity contributes to create entropy and consequently vorticity [22]. Thus, this zone is characterized by positive values of Q-criterion.

![Q-Criterion Contour](image)

**Figure 5. Q-Criterion Contour**

A close-up investigation of the flow physics at the hypersonic leading edge was conducted. Figure 7 illustrates IDS prediction in the form of the Q-Criterion at the leading edge. It is of interest to note that at the tip of the plate, the region with the greatest rate of strain within the flowfield is observed, albeit a small region. Immediately following this region is a similarly small region with the greatest rate of rotation. The two regions with the greatest rate of strain and rate of rotation are located at the leading edge, and it is from these two regions the shock wave and the boundary layer respectively emanates. Figure 7 shows the two regions appearing to merge for a very short distance but then quickly separating into the shock wave and boundary layer, and between them the formation of the entropy layer. At this point, the technical details associated with the growth of these sub-layers are not fully resolved, and as such, further analysis are warranted.

The hypersonic flat-plate solution predicted by the IDS confirms that the scheme is capable of accurately resolving the complex flow physics associated the leading edge invicid-viscous interactions as well as the steep gradients inside the boundary layer. The IDS scheme is currently being upgraded with parallel to handle three dimensional flows. Once these capabilities are validated, the hypersonic flat-plate problem will be revisited.

**HYPERSONIC FLOW CROSS JET INTERACTION**

Consider a hypersonic flow over a flat plate interacting with a sonic jet that is injected perpendicular to its path. Under appropriate conditions, the hypersonic flow bends the sonic jet stream to produce a complicated set of flow field interaction. This type of flow interaction leads to multiple shock and expansion waves, multiple flow separation regions, multiple re-attachment points, complex vortical structures, sonic zones and many re-attachments points. Figure 8 shows a representation of the flow field of interest to this problem. As figure 8 indicates, several of the flow features described above are depicted. Inviscid shock such as: barrel, reattachment, bow and separation are present. Also, there are two main recirculation zones, located behind and ahead of the injection point. No doubt, predicting the exact nature of this type of fluid flow is a challenging task for any CFD tool. However, in the case of testing the IDS scheme, this problem provides the best testbed.

![Q-Criterion (x=0.27)](image)

**Figure 6. Q-Criterion (x=0.27)**

In this analysis, the hypersonic cross flow problem is evaluated with a Mach number of 6.0, a Reynolds number of 1.3047x10^7 (based on the length of the plate), a Prandtl number of 0.789 and the specific ratio set to 1.4. In addition, the freestream density, temperature, viscosity, and pressure were assumed to be 0.090 kg/m^3, 57.23 K, 3.7655x10^-6 kg/ms, and 1478.26 Pa, respectively.
The 2-D computational domain used in this analysis was developed for a flat plate with a length of 0.6 and a height of 0.12 meters, respectively. The lower plane, i.e.: the bottom wall, corresponds to the solid surface of the flat plate. Vertical gradient for temperature and density were set to zero and no-slip conditions were imposed on the flat plate. The jet was assumed to have a step profile, i.e: no boundary layer in the injection point was considered. The injector was simulated by a small gap of width $1.644 \times 10^{-3}$ meters at the bottom wall at 0.350 meters from the leading edge. The Mach number at the injection zone was set to 1. The injection temperature and density were set to non-dimensional values of 4.5 and 21.0, respectively. Under these conditions, the non-dimensional pressure ratio at the injection point was computed, yielding a value of 94.49. Extrapolation boundary conditions were used on the top boundary and at the exit plane (outlet). Symmetric boundary conditions were used behind the flat plate. The boundary condition at the inlet was divided into two portions; one above and the other below the boundary layer height. The primitive variables were set to the freestream values for both portions, except for the horizontal component of the velocity vector. Although the flow upstream of the injector is expected to be fully developed, and therefore turbulent, a laminar profile was imposed.

The strongest recirculation appears behind the injection section. Ahead of the injection point, a weaker recirculation zone is found; these two recirculation zones were also reported in [24]. Although the size and location differed from the ones presented herein, it is important to recall that this analysis was conducted under realistic Reynolds number conditions. It is of interest to note that the recirculation zone located ahead of the injection point is predicted by two major vortices; an elongated zone that leads to formation of the bow shock and another located nearest to the injection point that is characterized by a triangular shape, refer to figure 10.

The IDS solution, illustrated in figure 11, revealed an important observation that has previously eluded many researchers. The flow field details associated with figure 11 are derived from the use of the FPEF supported by the Q-criterion, expressed by equation (12). As noted in figure 11, not only is there a barrel shock above the injected flow region, but there is also a shear layer riding on the surface of this shock. A close-up view of the figure shows that above the barrel shock and behind the bow shock there is a thin layer with intense vorticity. This curved layer is colored in red, and it extends from the merging of the bow shock to the barrel shock until a region with intense strain.

Similarly, figures 12 and 13 represent the Normal Mach number and the Normal density gradient plots, respectively. Both figures agree on the prediction of the compression waves and they showed the complete shock structures, namely: bow, separation, barrel, reattachment shocks, expansion waves and the weak leading edge shock. These shock structures were also reported in [24-26] except the secondary separation shock.

Figure 8. Sonic jet with supersonic cross flow [23]

Figure 9. "U" Velocity Contour

Figure 9 shows the horizontal components of the velocity vector as predicted by the IDS. It can be inferred from figure 9 that the flow injection acts as an obstruction to the main stream flow, causing a bow shock. The stream-tracers corroborate this hypothesis merging ahead of the injection point. The high momentum of the injected fluid causes the separation of the boundary layer that appears ahead of the injection point. It is important to mention that to the authors’ knowledge, there are no published studies that have produced these findings at the conditions reported herein.

Figure 10. U-Velocity Contour ahead the injection

Figure 11 represents the Q-criterion, and it corroborates that there are mainly two zones with intense vorticity, colored in light red. Also, figure 11 demonstrates that the features shown as sharp gradients on figure 9 and 10 correspond to zones with intense strain, thus shock waves. These shock waves are represented by areas with large negative values of $Q$-criterion, colored by light green.

The IDS solution, illustrated in figure 11, revealed an important observation that has previously eluded many researchers. The flow field details associated with figure 11 are derived from the use of the FPEF supported by the Q-criterion, expressed by equation (12). As noted in figure 11, not only is there a barrel shock above the injected flow region, but there is also a shear layer riding on the surface of this shock. A close-up view of the figure shows that above the barrel shock and behind the bow shock there is a thin layer with intense vorticity. This curved layer is colored in red, and it extends from the merging of the bow shock to the barrel shock until a region with intense strain.

Similarly, figures 12 and 13 represent the Normal Mach number and the Normal density gradient plots, respectively. Both figures agree on the prediction of the compression waves and they showed the complete shock structures, namely: bow, separation, barrel, reattachment shocks, expansion waves and the weak leading edge shock. These shock structures were also reported in [24-26] except the secondary separation shock.
It is of interest to note the disadvantage of relying only on the primitive variables. Although figures 9 and 10 agree with the bow shock predicted by figures 12 and 13, it is not possible to identify the reattachment, barrel and the secondary separation shock shown in figures 12 and 13. In contrast to figure 12 (Normal Mach number), figure 13 shows positive and negative values for the normal density gradient; both representing different physics. Positive values of the normal density gradient correspond to compression waves, whereas negative values correspond to expansion waves. Thus, figure 13 provides direct evidence of expansion regions that were not shown in figure 9, 10 and 12.

Figures 14 and 15 provide evidence of the dynamics of the flow in the injection section. As the jet emerges into the main stream, the supersonic flow expands, as shown in figure 14, with the vectors representing the Prandtl-Meyer expansion fans.

One important difference is the size of the recirculation zone that forms behind the injection point and the compression shocks that emanates at the reattachment point. In contrast to previous research studies such as [24-26], the injection plume is pushed along the main flow, decreasing penetration of the cross-flow. Also, the region behind the injection is attached to the bottom wall, whereas the previous studies show a separation bubble. Again, the difference in the momentum ratio, Mach and Reynolds numbers may be the cause of the discrepancies.

Recall, the IDS is a relatively new approach to solving the integral form of the Navier-Stokes equation numerically, and its past applications have demonstrated limited success. In efforts to further demonstrate the capability of the IDS, and to provide uncontroversial evidence that the IDS accurately predicts the behavior of the full NS equations, the analysis described herein.
was conducted. The two fluid dynamics problems, namely; the hypersonic flat plate problem and the sonic jet hypersonic cross flow interaction problem, were specially selected. These problems were chosen because of the complex nature of their respective flow fields, and because of the computational challenges they present to the CFD community.

As noted in the solutions generated by the IDS tool, and those presented in figures 1 - 14, it can be concluded that the IDS tool predicts the flow physics associated with complex fluid flow interactions extremely well. It is also of interest to note that the IDS provides solutions in the form of the primitive variables. However, to provide uncontroversial evidence of the flow field physics, special FPEF were derived using the IDS-provided solution. The final product are the plots, illustrated in figures 4 - 5, 7, 11 – 13, that serve as the uncontroversial evidence of shock and expansion waves, flow separation, reattachment, stagnation, vortical flow structures, and directions of fluid particle flow.

However, at this stage of the IDS development, it is yet to demonstrate how well it can predict the realistic 3-Dimensional fluid flow physics associated with turbulence. The problem of turbulence still remains the most vexing problems in the minds of fluid dynamists.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Professor Kenneth M Flurchick from the Department of Physics and Computational Science and Engineering program at North Carolina Agricultural and Technical State University (NCAT) for his guidance and advice on developing the parallel version of IDS. The computational resources used in this research were made available in part by a grant from the National Science Foundation Grant no. #1429464, entitled MRI: Acquisition of CRAY XC-30 HPC Cluster to Support NCAT Research Computing, Education and Outreach.

REFERENCES


VALIDATION OF AN IMPROVED CONTACT METHOD FOR MULTI-MATERIAL EULERIAN HYDROCODES IN THREE-DIMENSIONS

Kenneth C. Walls and David L. Littlefield
The University of Alabama at Birmingham
Birmingham, Alabama USA

ABSTRACT
Eulerian hydrocodes used in computational structural mechanics applications traditionally approach multi-material elements using mixed-element thermodynamic and constitutive models. These traditional approaches treat discontinuous pressure and stress fields that exist in elements with material interfaces by using a single approximated pressure and stress field. However, this approximation often has little basis in the physics taking place at the contact boundary and can easily lead to non-physical behavior due to artificial bonding that takes place at the material interface boundary. This work presents a significant departure from traditional approaches for handling Eulerian contact by solving the conservation equations separately for each material and then imposing inequality constraints associated with contact to the solutions for each material with the appropriate tractions included. This work validates the new multi-material Eulerian contact methodology with several computational examples. Comparisons are made between the method put forth in this work and traditional treatment of multi-material contact in Eulerian methods to demonstrate the advantages of this approach.

BACKGROUND
Due to their highly nonlinear nature, dynamic contact and impact problems involving multiple materials are usually solved using explicit finite element and finite difference programs. In order to obtain an adequate degree of accuracy, impact problems require small time steps and a fine mesh, enabling the problem to model deformations with a fair degree of certainty. Explicit finite element and finite difference formulations are therefore extremely useful in modeling crash worthiness, shock responses, and a wide range of other problems. There are several methods for modeling dynamic problems involving contact/impact, including the Lagrangian method and the Eulerian and ALE methods. Improvement of contact treatment in the latter is the focus of this work.

Due to their natural ability to form new free surfaces, Eulerian hydrocodes have become the preferred method for high velocity impact and penetration problems. Since the Eulerian mesh is fixed in space, problems that hinder Lagrangian formulations such as mesh distortion are eliminated. However, because material is allowed to move through the mesh, additional information is required to describe the contents of each element and update the calculations being performed. The addition of these material transport and advection calculations to Eulerian methods makes them much more computationally expensive than Lagrangian methods.

The principle issue with Eulerian hydrocodes comes from the treatment of multiple materials contained within a single computational element. In problems that involve the contact and separation of surfaces, contact forces between interacting materials must be accounted for. Typically, Eulerian hydrocodes treat subelement thermodynamics and stresses with mixture theory [2]. In this work, the term mixture theory is used to describe a method used to approximate the stress and pressure field discontinuities that occur at interface boundaries within an element with a single pressure and stress field. Individual materials have their own sets of properties, and each material is assumed to occupy some portion of the element volume. This requires the implementation of specific thermodynamic conditions to define a nonlinear system of equations where the volume fraction of each material within a cell acts as the unknown. Several mixture theories exist; however, none of these methods is based in the physics taking place at the interface boundary, and they are thus considered to be ad hoc methods, which will affect the accuracy of the solution.

Another issue associated with traditional Eulerian methods is that only a single velocity field is determined for all materials within the mixed cell. This implies that materials within this cell are in perfect bonding and, as such, no slip may occur between the materials. In other words, the slide lines that were used to naturally develop contact in Lagrangian hydrocodes do not exist in typical Eulerian codes [1]. This assumption results in a tremendous loss of accuracy in instances where contact is important and severely limits the ability of Eulerian and ALE methods to model the actual physics taking place in the problem. As a result, multi-material Eulerian formulations in the traditional sense have been limited to hypervelocity impacts and other problems that cannot be solved by any other means. An improvement to the handling of contact in Eulerian
hydrocodes that removes the need for mixture theory is the focus of the work being validated in this paper.

RESEARCH APPROACH

The theory developed in this work represents a significant departure from traditional treatment of contact-impact in an Eulerian framework. It essentially represents a generalization of a technique used in Lagrangian formulations and implements it in an Eulerian framework in order to avoid the mesh distortion drawbacks of a Lagrangian method while also removing issues related to mixture theory associated with an Eulerian approach. In this work, mixed cell thermodynamic and constitutive models traditionally used in an Eulerian framework are not used. Rather, the governing equations are solved for each material individually, and then specific contact constraints are imposed. After these constraints are enforced, the traction, which is traditionally handled implicitly in Eulerian methods, is determined explicitly. This results in a set of coupled equations that can be approximated using an uncoupled system. The following sections describe the finite element formulations and contact constraints used in this work. Readers can reference previous work by Littlefield [3,4,5], Walls and Littlefield [6,7], and Walls [8,9] for descriptions of the conservation equations, interface tracking formulation, and other background information about the work presented here. The formulations developed in this section have been implemented in the multi-material finite element code ALEAS in order to examine the advantages of this approach.

FINITE ELEMENT FORMULATION

In this work, the conservation equations are solved using a finite element formulation that makes use of two operator splits. The first operator split is traditionally performed in Eulerian formulations and is solved without accounting for the traction, t. The second operator split is unique to this work and enforces contact by determining the traction and applying the Signorini contact constraints. This second operator split is valid everywhere, but since the traction is zero except in elements where contact occurs, it is only performed in elements where the contact constraints are not identically satisfied. In this work the values for density, $\rho$, and specific internal energy, $e$, are considered to be piecewise linear constants, while the velocity function, $v$, is a piecewise linear function.

The finite element formulation for the conservation of momentum equation is presented here. Similar formulations can be developed for the conservation of mass and energy equations and a detailed derivation can be found in [9]. The first step is the Lagrangian step, in which the problem is advanced in time, and thus involves only those components which are time dependent. In this step the mesh moves with the material as it would in a traditional Lagrangian calculation. During this step the materials are permitted to move independently of one another, so the traction term is not included. The geometric interpretation of this step of the finite element formulation can be seen in Figure 1 (b). The conservation of momentum equation can now be written in a discretized form as:

$$M_{m,l}^{\text{lag}} \frac{\partial}{\partial t}(v_{m,l}^{\text{lag}}) = \sum_{k=1}^{n_e} \left( N_l f_l - \nabla N_l \cdot \sigma_{m,k} \right) \phi_{m,k} dy$$

$$l = 1, 2, ..., n_n$$

where $M_{m,l}^{\text{lag}}$ is the lumped mass of material $m$ at node $l$, $N_l$ is the nodal shape function, $f_l$ is the externally applied force, $\sigma_{m,k}$ is the Cauchy stress for element $k$, $\phi_{m,k}$ is the volume fraction of material $m$ in element $k$, $n_e$ is the total number of elements, and the superscript $\text{lag}$ indicates the values are for the Lagrangian step.

At this point, the mesh has been moved, and the formulation is identical to that of a Lagrangian scheme. However, now a second step is implemented to restore the mesh to its original position and to calculate the amount of material transported between adjacent elements is calculated. This step is known as the remap or Eulerian step, and it handles the transport of mass, energy, momentum, stress, and other quantities. During this step we must employ the interface tracking methods described in [5] to determine the shape and contents of the advection volumes for material transport. Upon finding the advection volumes we can then use them to update velocities, energies, masses, and other quantities. A geometric interpretation of the advection step is shown in Figure 1 (c). In this step the right hand side of the conservation equations are set equal to zero and the spatial quantities which were neglected in the Lagrangian step are now included. This gives:

$$m_{m,k}^{\text{remap}} \sum_{l=1}^{n_e} N_l \frac{\partial}{\partial t}(r_{m,l}^{\text{remap}}) = 0$$

$$k = 1, 2, ..., n_e$$

where $m_{m,k}^{\text{remap}}$ is the mass of material $m$ in element $k$, $r_{m,i}$ is the material specific advection velocity for material $m$ at node $i$, $N_l$ and $N_e$ are shape functions, $n_e$ is the unit normal vector, $n_s$ is the total number of nodes and the remap superscript indicates the quantity is from the remap step. It should be noted that Eq. 2 contains multiple velocity unknowns and is not diagonalized. In the work presented here the Half-Index Shift (HIS) algorithm developed by Benson was used to diagonalize the system [10].

At this point the finite element formulation is the same as that of a traditional Eulerian scheme, however since the method developed in this work makes use of material specific velocity fields the bodies have been permitted to move without
interacting with one another. As a result, it is possible for multiple materials to occupy the same space at once. Therefore, a second operator split must be applied to determine the traction and apply contact constraints in order to satisfy the conservation equations. The first step of the second operator split is known as the contact enforcement step. The conservation of momentum equation for this step is given by:

\[
M_{m,k}^{ce} \frac{\partial}{\partial t} (\mathbf{v}_{m,k}^{ce}) = \sum_{l=1}^{n_e} \int_{\Gamma_k} \mathbf{t} \cdot d\mathbf{s}
\]

where \( M_{m,k}^{ce} \) is the nodal lumped mass associated with the contact enforcement step and the superscript \( ce \) indicates that the quantity is associated with the contact enforcement step. The right-hand side of this equation can be solved in a number of different ways and in this work is handled by the Eulerian contact enforcement method described in the Contact Constraints section of this paper. The contact enforcement step can be visualized in Figure 1 (d).

The final step, shown in Figure 1 (e), is the contact enforcement remap step, which is necessary to handle the transport of mass, momentum, energy and other quantities since the mesh was moved during the contact enforcement step. The conservation of momentum equation for this step is given by:

\[
\sum_{l=1}^{n_e} \int_{\Gamma_k} \mathbf{t} \cdot d\mathbf{s} \nonumber \\
+ \sum_{l=1}^{n_e} \int_{\Gamma_k} \rho_m \mathbf{c}_{m,k}^{ce} \cdot \mathbf{n} \phi_{m,k} d\mathbf{z} = 0
\]

where \( \mathbf{c}_{m,k}^{ce} \) is the advection velocity associated with contact enforcement, and the superscript \( ce \) remap indicates that the quantity is associated with the contact enforcement remap step.

**CONTACT CONSTRAINTS**

When contact between two bodies occurs, a traction must be imposed along the interface boundary. In order for the tractions to be implemented correctly, several constraints must be enforced. In the Signorini form for frictionless contact, these constraints are:

1. **The No-Penetration Constraint**
   
   \[ (x_a - x_b) \cdot \mathbf{n}_b \geq 0 \]

2. **The traction must be compressive or zero**
   
   \[ t \cdot n_a \leq 0 \]

3. **And the product of the first two conditions must be zero**
   
   \[ (t \cdot n_a) (x_a - x_b) \cdot \mathbf{n}_b = 0 \]

In these constraints, \( x_a \) and \( x_b \) are the location along the interface boundary for each body, and \( \mathbf{n}_a \) and \( \mathbf{n}_b \) are the unit normal vectors on the boundaries of each domain. The first constraint specifies that two materials cannot occupy the same place at the same time. The second condition requires a compressive traction force to be present along the boundary region of interest in order for the traction integral to be applied. The final condition dictates that the product of the first two conditions is always zero, or in other words, the two bodies do not have the same unit normal vectors.

These inequality constraints must be satisfied everywhere along the material interface boundary. Since an Eulerian formulation implicitly prescribes the relative locality of materials, a check to satisfy Eq. 5 is straightforward. A volume based interpretation of Eq. 5 can be inferred at node \( l \) by the relation:

\[
\sum_{k=1}^{n_e} \max \left[ \sum_{m=1}^{n_m} \phi_{m,k} \right] - 1, 0 \right] = 0
\]

where \( n_e \) is the number of elements attached to node \( l \), \( n_m \) is the number of materials, and \( \phi_{m,k} \) is the volume fraction of material \( m \) in element \( k \). When this condition is not satisfied nodal forces are added that alter the velocity of the neighboring nodes. By selecting one material as the master and one as the slave material the surface normals are not required to be equal and opposite, as they typically would be in a traditional continuous Eulerian formulation.
The enforcement of contact thus leads to velocity updates for each node in every element connected to node $i$. These velocity update equations can be diagonalized, thus avoiding matrix inversion, and can be written as:

$$v_{a,i}^{n,ce} = v_{n,remap}^{a,i} - \frac{M_{c(e)}^{(a)}}{\Delta t(M_{c(e)}^{(a)} + M_{c(e)}^{(b)})} \sum_{k=1}^{n_k} \frac{1}{A_k}$$

and

$$v_{b,i}^{n,ce} = v_{n,remap}^{b,i} + \frac{M_{c(e)}^{(a)}}{\Delta t(M_{c(e)}^{(a)} + M_{c(e)}^{(b)})} \sum_{k=1}^{n_k} \frac{1}{A_k}$$

where $v$ is the normal component of velocity, $M_{c(e)}^{(a,b)}$ is the lumped mass, $A_k$ is the area of the interface in element $k$, $\Delta t$ is the time step, and $V_e^n$ is the excess volume in the elements connected to node $i$, and further, the subscripts $a$ and $b$ are indicative of master and slave values respectively.

The excess volume, $V_e^n$, is given by:

$$V_e^n = \sum_{k=1}^{n_k} \max \left( \left( \frac{\sum_{m=1}^{n_m} \phi_{m,k}}{} - 1 \right) \cdot V_{0,k}, 0 \right)$$

where $V_{0,k}$ is the undeformed volume of the element. When $V_e^n$ is greater than zero, in other words when the element is overfilled, we must find the normalized areas of the contact interface for each material as well as the normal vector for each contact surface. The methods for doing so are described in [9].

**DISCUSSION**

There are several advantages to using the form of the conservation equations described in this work. Since this formulation treats each material separately, and there are no quantities associated with the whole cell, such as temperature, pressure, or strain rate; there is no need for the use of traditional cell mixture theory involving mixed cell thermodynamic, strength, and fracture algorithms, all of which typically contribute to significant amounts of error in simulations where contact is important. Additionally, the form of the conservation equations used in these formulations guarantees element-level conservation of mass, momentum, and energy. This allows for the amount of material being transported between adjacent elements to be determined easily. One other feature that distinguishes the approach used in this work is that each material has its own momentum balance and, consequently, separate densities, velocities, energies, and stresses are updated for each material.

The main complication of this method is a result of the ability of each material to have its own velocity field. Because of this property it is possible for advection volumes to become complicated in shape or even disjoint. This difficulty is accounted for by using interface tracking, however not in the traditional sense. In a typical multi-material Eulerian contact, formulation interface tracking is used to determine the contents of an advection volume whose shape is known. However, in this formulation the contents of the advection volume are known, and thus the shape of the advection volume can be determined. One of the current limitations to the accuracy of the presented method is the piecewise-linear nature of the Modified Young’s interface tracking algorithm implemented here. Since the contents of the element are represented by intersecting planar polygons, the interface edges are not necessarily continuous between elements. This limitation also results in difficulties resolving corners or other complex topologies within the element since the interface must be planar. A higher order interface tracking algorithm that allows for continuous material interfaces, while also conserving volume would be advantageous, but currently no such method has been developed.

The advantages of using the formulation presented in this work over traditional Eulerian methods for classes of problems where large deformations and high strains are present, but contact is important are evident and can be seen in the results of a set of validation problems described below.

**RESULTS**

In order to demonstrate the effectiveness of the method presented in this work, a set of validation problems have been developed. These test cases have been developed to demonstrate the superiority of the method described in this work to a traditional Eulerian formulation as well as to determine the applicability of the newly developed method in cases that are handled well by traditional Eulerian methods with mixture theory. The simulations presented here were carried out in the multi-material hydrocode ALEAS. Simulations using the method developed for this work were carried out using the multi-material momentum (MMM) option, while comparison runs using mixture theory were run using the single-material momentum (SMM) option.

<table>
<thead>
<tr>
<th>Table 1 Mie-Gruneisen equation-of-state parameters used in validation problems</th>
<th>p0 (g/cm$^3$)</th>
<th>C0 (cm/s)</th>
<th>S (-)</th>
<th>T (-)</th>
<th>C (erg/cm$^3$K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steel (RHA)</td>
<td>7.85</td>
<td>4.5e5</td>
<td>1.49</td>
<td>1.89</td>
<td>4.1208e6</td>
</tr>
<tr>
<td>Uranium</td>
<td>16.82</td>
<td>2.487e5</td>
<td>1.56</td>
<td>2.32</td>
<td>1.05132e6</td>
</tr>
<tr>
<td>Copper</td>
<td>8.93</td>
<td>3.94e5</td>
<td>1.49</td>
<td>1.99</td>
<td>3.92951e6</td>
</tr>
<tr>
<td>Tungsten</td>
<td>17.76</td>
<td>4.03e5</td>
<td>1.237</td>
<td>1.67</td>
<td>3.53292e6</td>
</tr>
<tr>
<td>Steel (Hardened)</td>
<td>7.85</td>
<td>4.5e5</td>
<td>1.49</td>
<td>2.17</td>
<td>4.1208e6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2 Johnson-Cook strength parameters used in validation problems</th>
<th>v (-)</th>
<th>A (d/cm$^3$)</th>
<th>B (d/cm$^3$)</th>
<th>C (-)</th>
<th>M (-)</th>
<th>N (-)</th>
<th>(T_{ref} ) (K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steel (RHA)</td>
<td>0.283</td>
<td>1.0e10</td>
<td>0.78e10</td>
<td>0.004</td>
<td>1.0</td>
<td>0.106</td>
<td>1783.0</td>
</tr>
<tr>
<td>Uranium</td>
<td>0.3</td>
<td>1.79e10</td>
<td>1.12e10</td>
<td>0.007</td>
<td>1.0</td>
<td>0.25</td>
<td>1497.9</td>
</tr>
<tr>
<td>Copper</td>
<td>0.333</td>
<td>8.97e8</td>
<td>2.9187e9</td>
<td>0.025</td>
<td>1.09</td>
<td>0.31</td>
<td>1380.7</td>
</tr>
<tr>
<td>Tungsten</td>
<td>0.281</td>
<td>1.36e10</td>
<td>0.176e10</td>
<td>0.016</td>
<td>1.0</td>
<td>0.12</td>
<td>3695.0</td>
</tr>
<tr>
<td>Steel (Hardened)</td>
<td>0.299</td>
<td>0.81e10</td>
<td>0.5095e10</td>
<td>0.014</td>
<td>1.03</td>
<td>0.26</td>
<td>1818.0</td>
</tr>
</tbody>
</table>

Frictionless Block Sliding Problem
The problems associated with multi-material contact in a traditional Eulerian scheme can be illustrated using the simple example of the angled frictionless block sliding problem. In this problem two blocks are angled at 45° to the orientation of the mesh in order to increase the number of mixed elements along the interface. As can be seen in Figure 2 (c), there was a large amount of deformation along the interface boundary in the traditional Eulerian calculation. This also resulted in the presence of a large amount of trailing material as the blocks separated from one another. This is indicative of significant material replacement and bonding taking place along the interface boundary.

Figure 2: Initial condition (a) and results of the frictionless block sliding validation problem for ALEAS MMM (b) and ALEAS SMM (c) at 40µs

When the contact method developed in this work was implemented, however, there was no deformation present along the material interface of the blocks and no material replacement took place. As such, the pure advection problems have been validated using the multi-material Eulerian scheme described in the formulation section. Figure 2(b) illustrates the Eulerian formulation with contact modeling, while Figure 2(c) depicts traditional Eulerian methods in the angle block sliding problem.

Taylor Impact Test

The Taylor impact test is a widely used validation problem for computational codes and the associated material models. The test involves a flat-nosed cylindrical projectile striking a rigid target plate at nominal incidence. This test is useful because it subjects the material to a wide range of strain rates throughout the sample. Strain rates at the impact front are very high, while they are significantly lower in the undeformed section of the projectile. As such it is very good at revealing the strain hardening behavior of the material.

The specific test used to validate the contact model in ALEAS came from a report by Banerjee [11]. In this case, an annealed OFHC copper cylinder with a diameter of 7.62 mm and a length of 23.47 mm was impacted onto a rigid plate at 210 m/s. For the purposes of the simulation the copper cylinder was represented using a Mie-Gruneisen equation of state and a Johnson-Cook strength model. The material parameters are shown in Tables 1 and 2.

Figures 3 and 4 demonstrate the advantages of using the method developed for this work. The ALEAS MMM formulation, shown in Figure 3 (b) and outlined in blue in Figure 4, shows a much more significant spread of material at the rigid boundary interface and at late time the impactor rebounds off of the impact plate. This rebounding is a direct result of the use of multiple velocity fields and is not due to ad hoc fracture methods, which have currently not been implemented in ALEAS. Figure 3 (c) illustrates the limitation of the single velocity field formulation. Since only one velocity is possible for both materials the spread of the impactor at the interface is significantly reduced. This is a direct result of the bonding and material replacement properties of traditional Eulerian methods that utilize mixture theory. In reality, the material should be free to move laterally along the surface of the rigid plate; however, since only a single velocity field is present the velocities of both materials within the element are the same, and the shearing velocity gradients are non-zero, resulting in the restriction of lateral movement of material.

Figure 4 also includes a deformation profile from a run conducted using the Sandia National Laboratories Eulerian hydrocode CTH, which was used as a means of validating the ALEAS SMM results since it relies on mixture theory to handle mixed elements. In both the ALEAS SMM and CTH runs the bar remains bonded to the target at late time as the deformation reaches its maximum and the sign of the velocity changes. This should result in rebounding of the bar as was seen in ALEAS MMM, but since no ad hoc fracture method was implemented in these runs, the bar remains bonded to the plate, as is typical of mixture theory.

Figure 3: Initial condition (a) and results of the Taylor impact test for ALEAS MMM (b) and ALEAS SMM (c) at 100µs
Figure 4: Comparison of maximum deformation profile for ALEAS MMM, ALEAS SMM, CTH, and experiment Taylor impact test

Long Rod Penetration of Oblique Plates

Many ballistic penetration and defense applications can result in yawed and oblique impacts. For example, in an oblique impact of an unyawed penetrator into multi-layered spaced armor the first layer deflects and rotates the projectile resulting in a yawed interaction with subsequent layers. Fugelso and Taylor [12] examined the penetration of uranium alloy (U 0.75 wt% Ti) rods into steel targets at a variety of yaws, obliquities, thicknesses, and velocities. As a means of validating the advantages of the method implemented in ALEAS MMM this work used the three test conditions presented by Cagliostro, et al. [13] which utilized data from Fugelso and Taylor [12] to validate the MESA code.

The penetrators were cylindrical rods with hemispherical tips composed of uranium alloy (U-0.75 wt% Ti). They had a length of 7.67 cm and an aspect ratio (L/D) of 10. The mass of the projectile was 65 g. The plates had a thickness of 0.64 cm and were composed of rolled homogeneous armor (RHA). The initial velocities and orientations of the rods and plates were designed in such a way that the rod impact velocity relative to the plate was 1.29 km/s at an obliquity of 65° for yaws of -9.3°, 0°, and +10.3°. Rod and plate materials were modeled using the Mie-Gruneisen equation of state and Johnson-Cook strength model using the characteristics for Uranium and Steel (RHA) in Tables 1 and 2.

The penetration of oblique plates by projectiles of various yaw serves as a means of demonstrating the shortcomings of traditional Eulerian formulations with mixture theory. Due to the increased interaction between materials as the rod passes through the plate an artificial slowing of the rod compared to experimental data is often seen due to material bonding when mixture theory is employed. This result is not desirable, especially for applications such as multi-layered armor or other events where the ability to accurately model a secondary or further subsequent impact is important.

Results of the -9.3° yaw case are shown in Figure 5, and the relative exit velocity versus yaw angle is summarized in Figure 6 for all cases. Figure 6 includes experimental results from Fugelso [12] shown as purple dots, as well as results from ALEAS MMM, depicted by red squares, ALEAS SMM, depicted by blue diamonds, and CTH, depicted by green triangles. ALEAS MMM showed significant improvements over ALEAS SMM and CTH which use mixture theory. Both of the mixture theory runs show a significant artificial slowing of the penetrator due to material bonding. In the CTH runs this effect is mitigated by the use of an ad hoc fracture model, which has not yet been implemented in ALEAS. Cagliostro saw a similar effect when fracture was included in his validation of the MESA code [13]; however, the inclusion of fracture is not sufficient to obtain good agreement to experimental results. Figure 5 (c) illustrates the shortcomings of the mixture theory formulation. As the penetrator passes through the plate large chunks of plate material bond to the rod. This is non-physical and is not seen in the ALEAS MMM formulation shown in Figure 5(b). In the 0° and +10.3° yaw cases (not shown here) the impactor showed a pronounced hooking at the nose due to bonding of material artificially pulling the nose downward. Again, this behavior is non-physical and was not seen in either experimental radiographs from Fugelso [12] or the ALEAS MMM simulations.

Figure 5: Initial condition (a) and results of -9.3° yaw case for ALEAS MMM (b) and ALEAS SMM (c) at 100µs
Long Rod Penetration of Semi-infinite Targets

Another class of impact and penetration events considered for the validation of ALEAS was the penetration of tungsten alloy long rod projectiles into semi-infinite steel targets. This class of problems is frequently used as a validation of hydrocodes, including work done by Park, et al. [14] to verify the implementation of the two-dimensional Eulerian X-FEM contact method in the ALEGRA code, which serves as the source of data for the validations carried out here. The setup for the simulations consists of a tungsten-heavy-alloy cylindrical penetrator with diameter of 0.5 cm and length of 5 cm impacting a semi-infinite hardened steel target at initial velocities of 500, 1000, 1500, 2000, and 3000 m/s.

The role of contact in this class of problems is less important due to hydrodynamic erosion, so traditional Eulerian formulations that use mixture theory often perform very well. As such, it was desirable to ensure that the new method developed in this work was capable of obtaining reasonable results. As can be seen in Figure 7, the depth of penetration for the 3000 m/s impact case showed good agreement between both method; however, the ALEAS SMM case showed greater amounts of erosion of the impactor along the crater wall, while material continued to slide down the wall and collected at the base of the crater at late time in the ALEAS MMM case. Due to the hydrodynamic erosion that takes place in the rod at this velocity a trail of material should be present along the length of the crater, and this is seen experimentally. This serves to demonstrate a limitation of the method developed in this work. Since contact is not an important factor in long rod penetration of semi-infinite targets, the method developed for this work was not expected to provide perfect agreement; however, when comparing penetration efficiency, the primary measure for assessing these simulations, the agreement between the new contact method and mixture theory proved to be adequate.

ALEAS MMM showed very good agreement at high impact velocities, but provided somewhat less accurate results for penetrator velocities of 1500 m/s and lower. Figure 8 shows the penetration efficiency (P/L) versus projectile striking velocity. While there are relatively large differences between the simulation results and the fit to experimental data for some cases it should be noted that a mesh convergence study has not been conducted due to the current limitations of ALEAS. As such the most refined mesh possible given ALEAS’s current memory limits was used. While this limitation certainly affects the validity of the solution, as can be seen in Figure 8, the results obtained for the zoning size used in these ALEAS simulations compare favorably to those reported for the same zoning size at 1500 m/s in the two-dimensional axisymmetric X-FEM contact implementation in the ALEGRA code in [14]. While this might be coincidental, it provides reason to believe that an increased mesh resolution would improve the performance of the contact formulation. In the current implementation of the contact method a fairly refined mesh is required due to averaging that takes place for the determination of the nodal contact normal vector and excess volumes. Furthermore, the contact model implemented in ALEAS MMM relies on averaging of interface areas in order to simplify the algorithm. It is possible that this results in contact velocity changes that are slightly too high in the direction opposite to the impact direction and that this leads to decreased penetration depths. Future work will include improvements to the contact formulation, and a mesh resolution study will be conducted once the memory limitation in ALEAS is resolved.

![Figure 7: Initial condition (a) and results of the penetration of L/D=10 tungsten alloy rod into steel target with V₀=3000m/s at 50 μs for ALEAS MMM (b) and ALEAS SMM (c)](image)
mixed element algorithms are used, the results are much more accurate using traditional Eulerian methods due to material.

This class of problems is especially difficult to model a long rod penetrator into oblique plates at various yaw angles. The practical applications of the method presented here less appealing for modeling cases where accurately capturing the interface is less important.

While the contact method presented here is very promising, its development is still in the early stages and there is considerable room for improvement. Areas of future work include improvements to the contact formulation, methods for modeling contact with friction, and extension of the method to more than two materials.

CONCLUSIONS

This study has validated a new method for the treatment of multi-material contact-impact problems in Eulerian hydrocodes used for computational structural mechanics. Contact constraints have been added to a multi-material Eulerian formulation using numerical approximations to impose the contact inequalities using a finite element model. The conservation equations have also been solved separately for each material in the problem space. As a result, a separate velocity field is assigned to each contacting body, leading to a tremendous improvement over traditional Eulerian contact methods that make use of mixture theory for problems where large deformations are present, but contact is an important factor.

This work represents a major shift in the implementation of Eulerian hydrocodes. This method is most applicable to problems involving large stresses and deformations, but where the role of sliding contact is important. It is best suited to be used when stresses and deformations are too large to be handled by Lagrangian codes, but sliding contact, which cannot be handled properly by traditional Eulerian codes, is also critical to the accuracy of the solution.

The advantages of using the method presented in this work have been shown in a series of validation problems. The block sliding problem indicates that no material replacement takes place along the contact interface in the new method, while artificial bonding occurs as a result of mixture theory. The Taylor impact test shows that since both the rigid plate and the copper impactor have their own unique velocity fields and no mixed element algorithms are used, the results are much more physically accurate. The practical applications of the method presented in this work were further indicated by simulations of a long rod penetrator into oblique plates at various yaw angles. This class of problems is especially difficult to model accurately using traditional Eulerian methods due to material bonding that takes place as the penetrator passes through the moving plate. This results in a significant reduction in velocity of the penetrator that is not seen experimentally. The ALEAS MMM result much more closely approximates the experimental result, with no artificial slowing of the rod and no plate material being erroneously bonded to the penetrator. Finally, the normal penetration validation problems, which primarily rely on hydrodynamic deformation, have shown that this method performs admirably in cases where traditional Eulerian formulations perform well, but the added computational cost of the method and slightly reduced accuracy make the choice of the method presented here less appealing for modeling cases where accurately capturing the interface is less important.

REFERENCES


