

Computational Validation of Nuclear Explosion Energy Coupling Models for Asteroid Fragmentation

Ben Zimmerman* and Bong Wie†

Iowa State University, Ames, IA 50011

The objective of this research is the formulation, implementation, and validation of a numerical method to accurately model and simulate explosion blast and shock waves for the disruption of asteroids. While the cases presented in this paper are not explosions caused by actual nuclear devices, they will aid in validation of the numerical model for future nuclear-explosion simulations. Current work employs the two-dimensional high-order correction procedure via reconstruction (CPR) method with explicit time-stepping to solve the Euler equations. Accurate capturing of shocks is obtained through a shock detector coupled with a slope limiter. The CPR method is compared with simulations from AUTODYN, a commercial hydrodynamics code by ANSYS, where excellent matchings are obtained. Additionally, all numerical simulations are completed using NVIDIA CUDA (Compute Unified Device Architecture), where superior computational speeds are demonstrated. Finally, surface and subsurface explosions are simulated, and the energy transferred to a target body is monitored. Coupling factors are computed and the desired results are obtained. This new computational tool is to be incorporated into a mission design software tool for the analysis and design of an asteroid disruption mission.

I. INTRODUCTION

The geological record demonstrates that asteroids and comets have impacted Earth in the past, and such events will happen again in the future. These events can occur at any time, and humanity must therefore discover incoming objects in advance and be prepared to act to mitigate the threat they pose. Due to the impact of the Chelyabinsk asteroid ($\sim 17\text{m}$) in Russia on February 15, 2013 and a near miss by asteroid 2012 DA₁₄ ($\sim 45\text{ m}$ diameter) on the same day, there is now a growing national and international interest in developing a global plan to protect the Earth from impact by hazardous asteroids or comets. As a result, on June 18, 2013, NASA announced an “Asteroid Grand Challenge” to find all asteroid threats to human populations and know what to do about them.

Mitigation of the hazard posed by asteroids and comets whose orbits bring them near Earth’s orbit, collectively referred to as near-Earth objects (NEOs), will be a continuously growing concern, as most proposed methods require substantial warning time in order to guarantee the effectiveness

*Research Assistant, Dept. of Aerospace Engineering, 1200 Howe Hall, bzimmer@iastate.edu.

†Vance Coffman Endowed Chair Professor, Asteroid Deflection Research Center, Dept. of Aerospace Engineering, bongwie@iastate.edu.

of a deflection or disruption mission. For short warning time cases, a report by the United States National Research Council (NRC) suggests the use of a nuclear explosive device [1]. The implementation of a nuclear NEO mitigation option requires thorough investigation, as incorrect use could yield undesirable fragmentation of the target NEO, e.g., due to low energy coupling to the NEO [2]. However, disruption due to high energy coupling can significantly reduce the amount of mass remaining on an Earth-impacting trajectory. Hence, high accuracy in predictions of energy coupling to NEOs is required for the success of an asteroid disruption mission.

An innovative concept being developed in a NASA Innovative Advanced Concepts (NIAC) Phase 2 study utilizes the concept of blending a hypervelocity kinetic impactor with a subsurface nuclear explosion for optimal disruption of an asteroid with short warning time [3, 4, 5, 6].

Previous research has produced promising but inconsistent results. A Hypervelocity Asteroid Intercept Vehicle (HAIV) concept [3, 4, 5, 6] was developed to use a kinetic impactor followed by a nuclear explosive device to effect a subsurface explosion. Subsurface explosions are desired because for a given explosive energy and detonation location they can be 20 times more efficient at coupling energy to an asteroid than a surface explosion. This efficiency factor is cited in a study published by the NRC [7]. However, while the results obtained from a developed in-house SPH (Smooth Particle Hydrodynamics) code [6] visually demonstrate the effectiveness of a subsurface explosion, they do not indicate the aforementioned energy coupling efficiency. Further calculations of the efficiency from the developed code yield poor results. In a more recent study, a computational tool, AUTODYN, by ANSYS, demonstrated computed efficiency factors of 11 [8] where a factor of 20 should be observed.

Current research involves verifying the efficiency factor described above, which first requires accurate simulation of blast and shock waves. To achieve this, the high-order Correction Procedure via Reconstruction (CPR) method, developed by Huynh [9] and extended by Wang [10], is implemented to solve the Euler equations. The CPR method combines the efficiency and simplicity of the finite difference method with compactness and high accuracy. This is due to the construction of the solution, which is approximated by high-order polynomials within each element and assumed discontinuous at cell interfaces. Due to this compactness, the method is well-suited to Graphics Processing Unit (GPU) computing, where significant speed-ups have already been demonstrated [11, 12]. A computationally efficient method is sought because coupling blast waves into target bodies and the eventual propagation of shock waves through the body is expected to be computationally intense. Hence, both efficiency and accuracy are desired, which can be found in high-order methods.

This paper is organized in the following way: First, the CPR method is reviewed along with an explanation of the shock sensor and limiting technique. Then, application to GPU computing is discussed and speed results are shown to demonstrate the computational superiority of GPUs. The implemented CPR method in one and two dimensions is then validated by comparing numerical results with both analytic solutions and results from AUTODYN. Finally, a case is presented for which a subsurface explosion is 10 times more effective than a surface explosion, and simulations are completed in order to achieve this factor.

II. NUMERICAL METHOD

A. CPR Formulation

Consider the hyperbolic conservation law for two dimensions described by

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{q})}{\partial x} + \frac{\partial \mathbf{g}(\mathbf{q})}{\partial y} = 0 \quad (1)$$

where \mathbf{q} is a vector of the conserved variables

$$\mathbf{q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix} \quad (2)$$

and $\mathbf{f}(\mathbf{q})$ and $\mathbf{g}(\mathbf{q})$ are flux vectors

$$\mathbf{f}(\mathbf{q}) = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ \rho uv \\ u(e + p) \end{bmatrix}, \quad \mathbf{g}(\mathbf{q}) = \begin{bmatrix} \rho v \\ \rho uv \\ p + \rho v^2 \\ v(e + p) \end{bmatrix} \quad (3)$$

In Equations (2) and (3), ρ is the density, (u, v) is the velocity in (x, y) directions, e is the total energy per unit volume, and pressure $p = (\gamma - 1)(e - \frac{1}{2}\rho(u^2 + v^2))$ where γ is the ratio of specific heats. Current work employs the ideal gas equation of state to close the system. The domain is discretized into N non-overlapping elements of volume V_i for element i . In one dimension, the i^{th} element is defined as $V_i = [x_{i-1/2}, x_{i+1/2}]$ which can be transformed into a standard element of $[-1, 1]$. In two dimensions, the same can be achieved. A quadrilateral element, $V_i = [x_1, x_2] \times [y_1, y_2]$ can be transformed into a $[-1, 1] \times [-1, 1]$ element. The method is described in one dimension now, as the extension to two dimensions for quadrilateral elements is straightforward, since all operations are completed in a one-dimensional manner. The governing equation for one dimension reduces from Equation (1) to the following:

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{q})}{\partial x} = 0 \quad (4)$$

In addition, the state vector and flux vector become

$$\mathbf{q} = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \quad \mathbf{f}(\mathbf{q}) = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ u(e + p) \end{bmatrix} \quad (5)$$

Within an element i , $k + 1$ solution points, located at $x_{i,j}$, $j = 1, \dots, k + 1$, are defined which contain the values of the state variable $\mathbf{q}_{i,j}$. The solution can then be approximated by a degree k polynomial, i.e. $\mathbf{q}_i(t) \in P^k$. Lagrange interpolation polynomials are used to approximate the solution

$$\mathbf{q}(x, t) \approx \mathbf{q}_i(x, t) = \sum_{j=1}^{k+1} \mathbf{q}_{i,j}(t) L_j(x) \quad (6)$$

Table 1. Correction coefficients for $P^1 - P^5$

j	P^1 $g'_{L,j}$	P^2 $g'_{L,j}$	P^3 $g'_{L,j}$	P^4 $g'_{L,j}$	P^5 $g'_{L,j}$
1	2.0	4.5	8.0	12.0	18.5
2	-1.0	-0.75	-0.5938	-0.2612	0.2513
3	-	1.5	0.9688	0.9375	0.8518
4	-	-	-2.0	-1.1451	-1.1244
5	-	-	-	0.5	1.3103
6	-	-	-	-	-3.0

Likewise, the flux polynomial can also be constructed as

$$\mathbf{f}(\mathbf{q}_i) \approx \bar{\mathbf{f}}_i(x, t) = \sum_{j=1}^{k+1} \mathbf{f}(\mathbf{q}_{i,j}(t)) L_j(x) \quad (7)$$

The dependence on time is assumed and dropped from the remainder of equations for simplicity. The reconstructed fluxes are considered to be continuous only within each element and discontinuous across the element interfaces (a Rusanov [13] or Roe [14] solver is employed at interfaces to evaluate the discontinuous solution). Thus, a new flux function, $\hat{\mathbf{f}}_i(x)$, must be defined, so that $\hat{\mathbf{f}}_i(x)$ is a polynomial of one order higher than the solution, i.e. $(k+1)$, it is close to $\bar{\mathbf{f}}_i(x)$, and becomes the values of the Riemann fluxes, $\tilde{\mathbf{f}}(\mathbf{q}^-, \mathbf{q}^+)$, at element interfaces, as follows:

$$\hat{\mathbf{f}}_i(x_{i-1/2}) = \tilde{\mathbf{f}}(\mathbf{q}_{i-1}(x_{i+1/2}), \mathbf{q}_i(x_{i-1/2})) = \tilde{\mathbf{f}}_{i-1/2} \quad (8)$$

$$\hat{\mathbf{f}}_i(x_{i+1/2}) = \tilde{\mathbf{f}}(\mathbf{q}_i(x_{i+1/2}), \mathbf{q}_{i+1}(x_{i-1/2})) = \tilde{\mathbf{f}}_{i+1/2} \quad (9)$$

In Equations (8) and (9), $x_{i-1/2}$ and $x_{i+1/2}$ indicate the left and right boundaries of an element i respectfully. The new flux function, once evaluated, can be used to update the solution at every element i and each solution point j with the following differential equation:

$$\frac{\partial \mathbf{q}_{i,j}}{\partial t} + \frac{\partial \hat{\mathbf{f}}_i(x_{i,j})}{\partial x} = 0 \quad (10)$$

The core idea of the CPR method is how to define this $\hat{\mathbf{f}}_i(x)$ at all points within an element, since it is only defined at element interfaces. This new flux function is rewritten as

$$\hat{\mathbf{f}}_i(x) = \bar{\mathbf{f}}_i(x) + \delta_i(x) \quad (11)$$

where $\delta_i(x)$ is a correction polynomial defined within each element i which is close to 0. Furthermore, in order to satisfy the two boundary conditions, the function is written as

$$\delta_i(x) = [\tilde{\mathbf{f}}_{i-1/2} - \bar{\mathbf{f}}(x_{i-1/2})] g_L(x) + [\tilde{\mathbf{f}}_{i+1/2} - \bar{\mathbf{f}}(x_{i+1/2})] g_R(x) \quad (12)$$

Here, $g_L(x)$ and $g_R(x)$ are $k+1$ polynomial correction functions, which satisfy the following relations on element boundaries

$$g_L(x_{i-1/2}) = 1, \quad g_L(x_{i+1/2}) = 0, \quad g_R(x_{i-1/2}) = 0, \quad g_R(x_{i+1/2}) = 1 \quad (13)$$

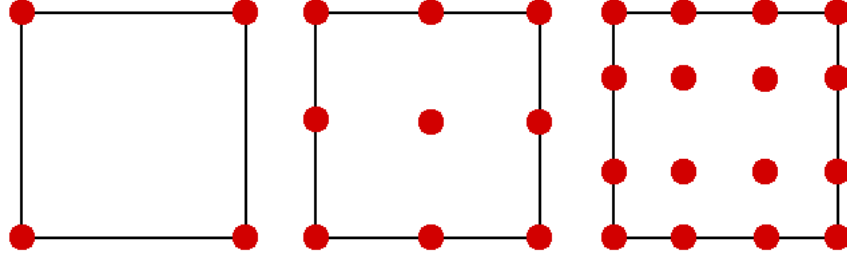


Figure 1. Legendre-Lobatto points in quadrilateral element for P^1 , P^2 , and P^3 reconstruction.

The newly defined flux can be substituted into Equation (10) to obtain the following

$$\frac{\partial \mathbf{q}_{i,j}}{\partial t} + \frac{\partial \bar{\mathbf{f}}_i(x_{i,j})}{\partial x} + \left[\tilde{\mathbf{f}}_{i-1/2} - \bar{\mathbf{f}}(x_{i-1/2}) \right] g'_L(x) + \left[\tilde{\mathbf{f}}_{i+1/2} - \bar{\mathbf{f}}(x_{i+1/2}) \right] g'_R(x) = 0 \quad (14)$$

The terms $g'_L(x)$ and $g'_R(x)$ are the derivatives of the correction functions, which can be calculated and imported into the numerical solver. Many correction functions are described by Huynh [9]. This research considers the g_2 scheme, where $g_L(x)$ has vanishing derivatives at the right boundary and all interior Legendre-Lobatto points. Similarly, $g_R(x)$ has vanishing derivatives at the left boundary and all other interior solution points. This results in a simple scheme, since the correction functions are only applied at interfaces. Additionally, due to symmetry, only correction functions for one interface need to be computed [9]. Table 1 shows the values of the correction functions for the left interface for P^1 to P^5 reconstructions. In order to approximate the flux derivative in Equation (14) the chain rule is applied as

$$\frac{\partial \bar{\mathbf{f}}_i(\mathbf{q}_{i,j})}{\partial x} = \frac{\partial \bar{\mathbf{f}}_i(\mathbf{q}_{i,j})}{\partial \mathbf{q}} \frac{\partial \mathbf{q}_{i,j}}{\partial x} \quad (15)$$

The term $\frac{\partial \bar{\mathbf{f}}_i(\mathbf{q}_{i,j})}{\partial \mathbf{q}}$ can be calculated analytically [15], while $\frac{\partial \mathbf{q}_{i,j}}{\partial x}$ can be approximated by the Lagrange interpolation

$$\frac{\partial \mathbf{q}_{i,j}}{\partial x} = \sum_{j=1}^{k+1} \mathbf{q}_{i,j}(t) \frac{\partial L_j(x)}{\partial x} \quad (16)$$

Extension to two dimensions is straightforward. All operations are completed in a one-dimensional manner, and the correction functions only change the flux at the interfaces of the element. Figure 1 illustrates P^1 , P^2 , and P^3 reconstructions in the standard quadrilateral element. Only on interface points does the correction function activate, making the scheme efficient in multiple dimensions as well. Upon the final update of the flux derivative at the boundaries of elements, explicit time-stepping is completed via a 3^{rd} order Runge-Kutta [16] scheme.

B. Element Transformation

The computational domain is composed of quadrilateral elements in two-dimensions. Each element is transformed from its standard coordinate system (x, y) to a standard quadrilateral element $(\xi, \eta) \in [-1, 1] \times [-1, 1]$. The transformation for each element takes the form

$$\begin{pmatrix} x \\ y \end{pmatrix} = \sum_{i=1}^N M_i(\xi, \eta) \begin{pmatrix} x_i \\ y_i \end{pmatrix} \quad (17)$$

where N is the number of points which define the element, (x_i, y_i) are the Cartesian coordinates of those points, and $M_i(\xi, \eta)$ are the shape functions which are determined via the node locations [17]. The Jacobian matrix \mathbf{J} is then given by

$$\mathbf{J} = \frac{\partial(x, y)}{\partial(\xi, \eta)} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \quad (18)$$

and when the transformation is non-singular, the inverse transformation must exist

$$\mathbf{J}^{-1} = \frac{\partial(\xi, \eta)}{\partial(x, y)} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \quad (19)$$

Additionally, the metrics can also be computed as

$$\xi_x = \frac{y_\eta}{|\mathbf{J}|}, \quad \xi_y = \frac{-x_\eta}{|\mathbf{J}|}, \quad \eta_x = \frac{-y_\xi}{|\mathbf{J}|}, \quad \eta_y = \frac{x_\xi}{|\mathbf{J}|} \quad (20)$$

The governing equation is then transformed from the physical to computational domain as

$$\frac{\partial \check{\mathbf{q}}}{\partial t} + \frac{\partial \check{\mathbf{f}}(\mathbf{q})}{\partial x} + \frac{\partial \check{\mathbf{g}}(\mathbf{q})}{\partial y} = 0 \quad (21)$$

where the transformed solution and flux vectors become

$$\check{\mathbf{q}} = |\mathbf{J}| \mathbf{q} \quad (22)$$

$$\begin{pmatrix} \check{\mathbf{f}} \\ \check{\mathbf{g}} \end{pmatrix} = |\mathbf{J}| \begin{bmatrix} \mathbf{f}\xi_x + \mathbf{g}\xi_y \\ \mathbf{f}\eta_x + \mathbf{g}\eta_y \end{bmatrix} \quad (23)$$

which allows the polynomials to be constructed within standard quadrilateral elements.

III. SHOCK CAPTURING

It is well known that high-order spatial reconstruction yields oscillations near sharp gradients. This causes both non-physical results and numerical instability. To resolve such discontinuities, slope limiting and shock sensing are employed. The basic idea is to apply slope limiting near or at solution discontinuities, thus reducing the system to 1st order, but maintaining high-order accuracy throughout the remainder of the domain. To accomplish this, a discontinuity sensor, as proposed by Persson and Peraire [18], is implemented. We define two solutions, one of order k and one of order $k - 1$ as follows:

$$\mathbf{q} = \sum_{i=1}^{n(k)} \mathbf{q}_i \phi_i, \quad \hat{\mathbf{q}} = \sum_{i=1}^{n(k-1)} \mathbf{q}_i \phi_i \quad (24)$$

The term $n(k)$ is the number of expansion terms and ϕ_i are basis functions. The solution of order k is \mathbf{q} , while the truncated solution is $\hat{\mathbf{q}}$. An indicator for smoothness can be defined as

$$s_i = \frac{(\mathbf{q} - \hat{\mathbf{q}}, \mathbf{q} - \hat{\mathbf{q}})_i}{(\mathbf{q}, \mathbf{q})_i} \quad (25)$$

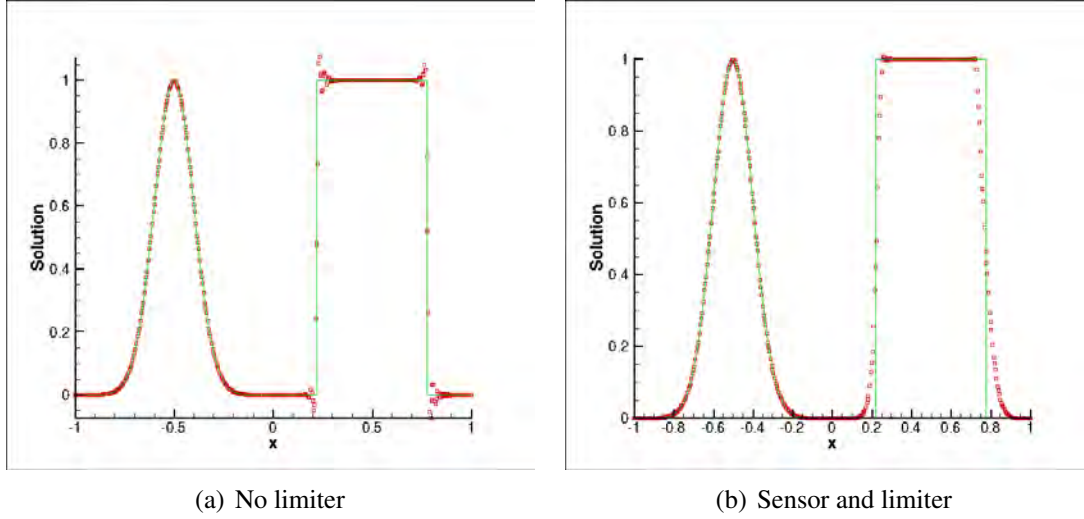


Figure 2. Exact solution and P^4 reconstruction at time $t = 2.0$ s.

where $(*, *)_i$ is the standard inner product in element i . Within each element i , the sensor compares the approximated solution with a truncated one. If the approximated solution is smooth, then the truncated solution will be smooth, and the value of s_i will be small. However, if the approximated solution has sharp gradients, then the inner product of the approximated and smooth solutions will become large. Once s_i becomes large enough (Persson and Peraire suggest $1/(k-1)^4$) the cell becomes marked and slope limiting activates within the marked cells, which is currently the simple *minmod* limiter. As an example, consider the one dimensional advection equation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \quad (26)$$

We initialize the domain, $[-1, 1]$, with a smooth and discontinuous solution across 100 elements. Cyclic conditions are imposed at the boundaries to simulate an infinite domain. At time $t = 2.0$ seconds (s) the solution will advect to the right one complete cycle. It is observed in Figure 2 how the high-order approximations excite the solution, causing oscillations near the sharp gradients. We also see the impact of the sensor and limiter. Note that the smooth portion maintains its peak. Limiting is only imposed on the rightmost portion of the wave, leaving the smooth portion a high-order approximation.

IV. GPU COMPUTING

Traditionally, computational facilities consist of Central Processing Unit (CPU) servers in order to complete large and complex numerical simulations. However, if the physics require large amounts of degrees of freedom to simulate correctly, the desired computational resources grow. This can be troublesome, particularly for academic research teams who do not have access to such computational power. To satisfy computational needs, GPUs are used for simulations in this research. NVIDIA released scientific computing support for its graphics cards in 2006 [19]. Since then, interests have shifted towards GPU compute unified device architecture (CUDA) computing. While CPUs contain only a few cores per processor, a single GPU can contain hundreds.

Due to the compactness of the CPR method, it is extremely efficient for message passing in-

Table 2. GPU and CPU speed comparison.

Order	$DoFs$	Speed (8 CPUs)	Speed (1 GPU)	Speed (4 GPUs)	Speed-up (1 card)	Speed-up (4 cards)
P^1	80,000	0.02511275	0.0021889	0.0009065	11.5	27.7
P^2	360,000	0.0534573	0.004310891	0.001469154	12.4	36.4
P^3	640,000	0.09591973	0.006305916	0.0019765	15.2	48.5
P^4	1,000,000	0.1372839	0.01099534	0.00319163	12.5	43.0

terface (MPI) and GPU CUDA computing. Most of the computational work per time step is completed locally, within each element. This includes function derivatives, evaluating the correction polynomial, and updating the solution with time. Only the Riemann solver at element interfaces require element to element communication, significantly reducing the amount of information required to couple elements together. This property is specific to high-order methods, making them ideal for parallel computing. Due to the vast parallelism, which can be exploited in high-order methods, GPUs seem an excellent fit. As a demonstration, consider a two-dimensional 40,000 element domain, where a CPU and GPU simulation is completed. The CPU used is an Intel XEON E5520 at 2.27 GHz while a Tesla K20 card is used for the GPU. Table 2 shows the speed comparisons of using 8 processors, one GPU, and four GPUs. The degrees of freedom (DoFs) are defined as the total number of elements times the number of solution points per element. For example, a 20 element domain and P^3 reconstruction (16 solution points reside within one element) yield $DoFs = 16 \times 20 = 320$. The simulation was run for several thousand iterations, and the average time per iteration was calculated and is shown in the table. It is easily noticed how GPUs affect the speed of a computation, using either one or four cards for one simulation.

V. VALIDATION RESULTS

A. 1D Sod's Shock Tube

To examine the shock sensor and limiting technique for a system of equations, consider Sod's shock tube, which requires the solution of the one-dimensional Euler equations. The test separates two gases with differing density and pressure, creating a discontinuity at the initial time $t = 0$ s. The domain, $x \in [0, 1]$, is discretized with 125 elements with $(\rho, u, p) = (1.0, 0, 1.0)$ for $x > 0.5$, and $(\rho, u, p) = (0.125, 0, 0.1)$ for $x \leq 0.5$ as the initial conditions. At time $t = 0.2$ s the density and pressure are compared against the analytic solution. Figure 3 demonstrates good agreement between the analytic and numerical solutions. Further mesh refinement will lead to much sharper solutions near the discontinuities.

B. 2D Riemann Problem

To test the implementation of the limiter and sensor in multiple dimensions, a two dimensional Riemann problem is simulated. The domain, $[0, 0.3] \times [0, 0.3]$, is discretized with 40,000 elements and P^2 reconstruction of the solution within each element. For $x + y < 0.5$, the density and pressure were set at $\rho = 1.0$ and $p = 1.0$. Throughout the remainder of the domain, $\rho = 0.125$ and $p = 0.4$. The velocity is set to zero everywhere. All four boundaries are wall conditions, causing complex shock and fluid interactions in the simulation. This case does not have an exact solution, but symmetry should be observed across the line $y = x$. Figure 4 shows contours of density at two different times. In both situations, symmetry across $y = x$ is observed, indicating that the sensor

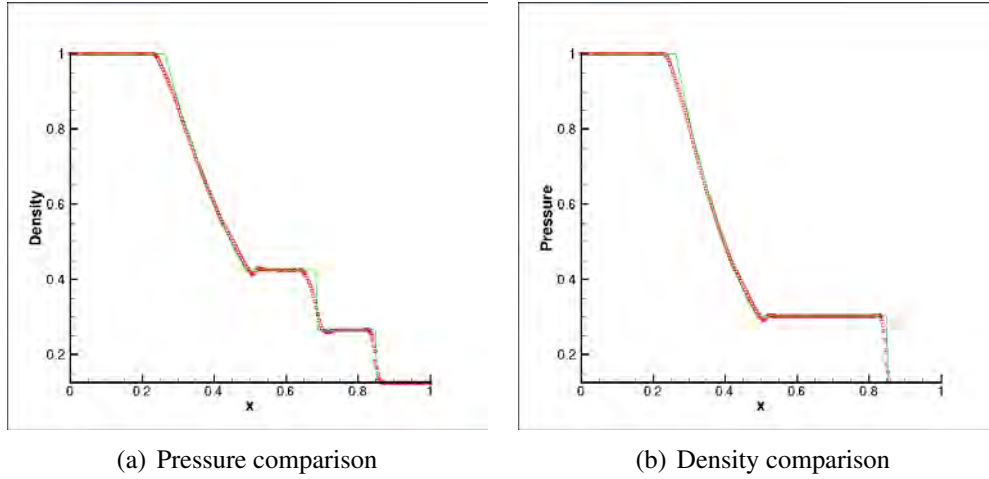


Figure 3. Sod's shock tube P^2 reconstruction.

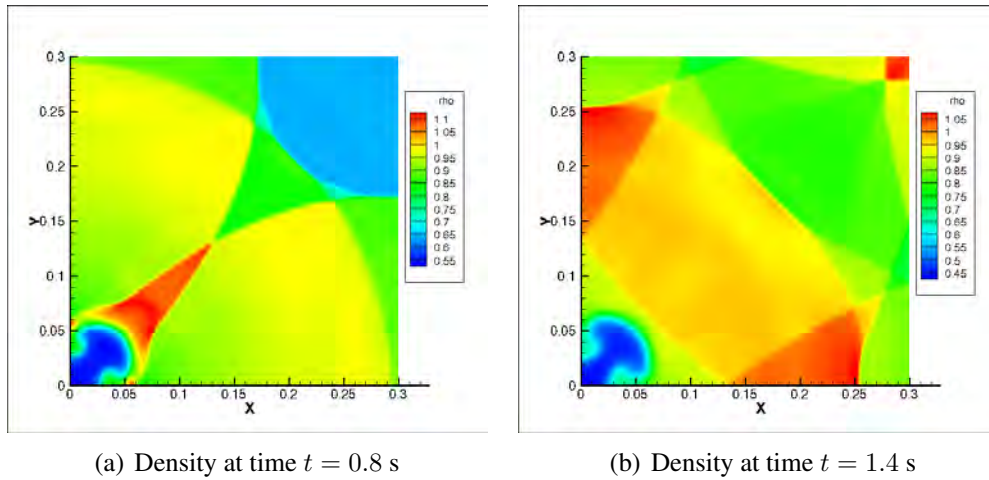


Figure 4. 2D Riemann problem P^2 reconstruction.

and limiter are implemented correctly.

C. AUTODYN Comparison

The CPR method is compared to results from AUTODYN, a program by ANSYS. AUTODYN uses smoothed-particle hydrodynamics and finite element solvers for numerical simulations. The intent of the test is to produce a more accurate solution with the CPR method using the same computational mesh (in this case we use 400 quadrilateral elements in each direction). The domain, $[-1, 1] \times [-1, 1]$, contains an initial density and pressure, $\rho = 1.0$ and $p = 1.0$, for $r = \sqrt{x^2 + y^2} < 0.4$ and $\rho = 0.125$ and $p = 0.1$ elsewhere. The velocity was set to zero throughout the domain. Figure 5 illustrates the density and pressure values at time $t = 0.25$ s. Data was taken along the centerline ($y = 0$) and compared to results from AUTODYN and Toro [21]. The results indicate that, for the same computational mesh, the CPR method does capture the solution better than AUTODYN. This gives confidence in our method and implementation as we move forward.

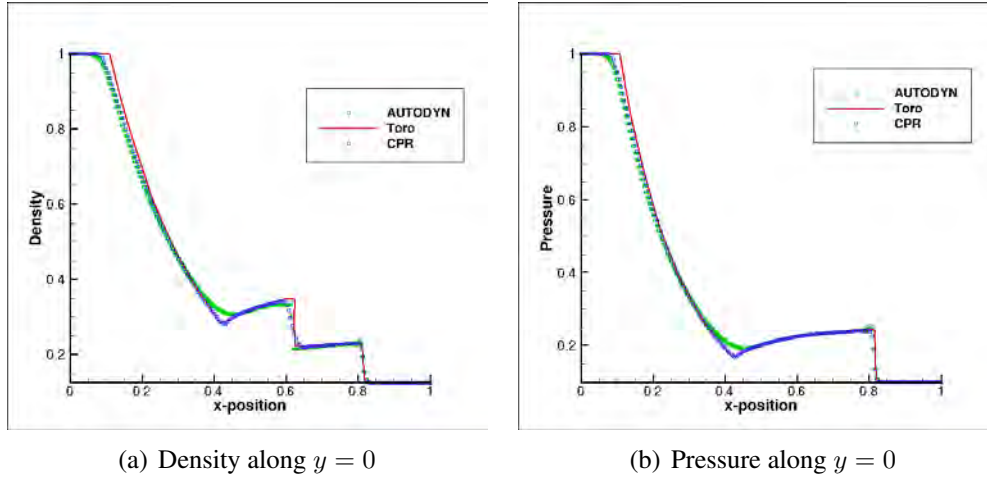


Figure 5. 2D blast wave problem at time $t = 0.25$ s.

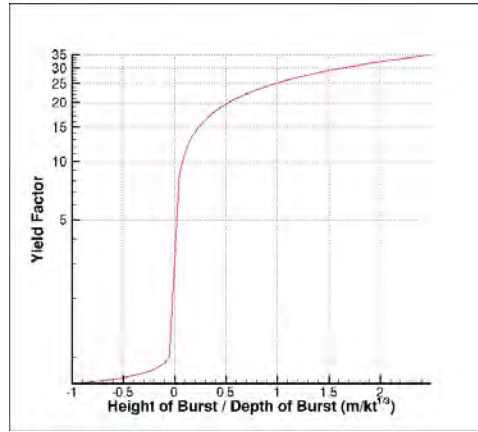


Figure 6. Energy coupling from nuclear device.

VI. ENERGY COUPLING

A. Problem Description

It is the intent of this research to validate the energy coupling factors presented by a study from the National Research Council (NRC) [7]. The results from the NRC study, presented in Figure 6, suggest that at a $HoB/DoB \approx 0.1$ (Height of Burst divided by Depth of Burst) the coupling factor of a subsurface explosion is 10 times more efficient than that from a surface explosion, and at a $HoB/DoB \approx -0.05$, the coupling factor becomes 1. HoB/DoB is defined as the detonation location relative to the surface location divided by the energy of the explosive to the 1/3 power, which can be related to a specific coupling factor, or the energy transferred to the target body from a subsurface explosion divided by the energy transferred from a surface explosion. This coupling factor is computed from the kinetic energy component which travels through a material [7]. In the current study, the measured kinetic energy into a system is defined as follows:

$$E_k = \frac{1}{\Gamma} \int_{\Gamma} \frac{\mathbf{v} \cdot \mathbf{v}}{2} d\Omega \quad (27)$$

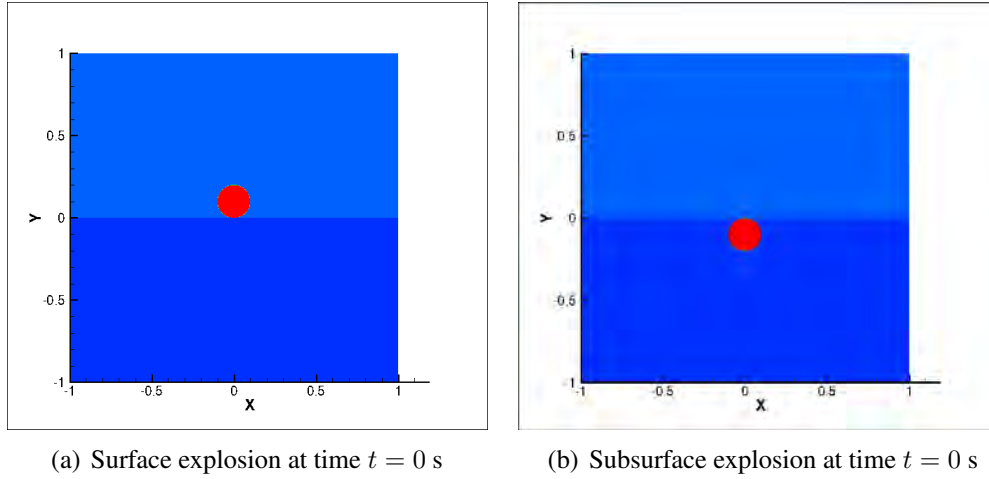


Figure 7. Initial conditions for surface and subsurface explosions.

where Γ is the size of the target body and \mathbf{v} is the velocity vector. The kinetic energy into the system, E_k , is recorded at each time step for both surface and subsurface explosions. At the same computational time, the kinetic energy values from the explosions are compared and a factor is computed. Currently, only the governing equations for fluids are implemented into the solver. It is expected that at lower depths, a solid material solver is necessary, but closer to the surface, the ground material can be assumed to behave more like a fluid. Finally, it should be noted that a nuclear explosion differs from a traditional explosion in two ways [22]. First, the temperatures associated with nuclear explosions are much higher. This issue is currently neglected, and only an ideal gas equation of state for air is used. Second, the mass of the explosive device can be neglected. This issue is addressed, and the density of the device is set as the density of the surrounding or outside air. As an example, consider the following simulations for an explosive device placed just under and above the surface as demonstrated in Figure 7. The lighter and darker blue represent the outside air and the target body respectfully, while the red circle represents the explosive device. Below $y = 0$, high density air (the target body) is initialized at a density 20 times higher than that of the outside air. Additionally, at a radius of $r < 0.1$, a high energy source is defined at 100 times greater than that of the surrounding material. Again, the density of the explosive device is set as the density of the outside air. A total of four computational grids were tested: Ω_1 (10,000 elements), Ω_2 (20,000 elements), Ω_3 (160,000 elements), and Ω_4 (250,000 elements). The numerical solution is captured at time $t = 0.1$ and $t = 0.2$ s.

B. Numerical Results

Figures 8, 9, 10, and 11 clearly illustrate that refinement leads to better numerical solutions. To verify the 10 times efficiency from Figure 6, it is necessary to run two separate simulations. One simulation must be completed at a $HoB/DoB = -0.05$ (coupling factor here is 1), and an additional simulation at $HoB/DoB = 0.1$ (coupling factor of roughly 10). The kinetic energy transferred to the target is monitored for all time in both simulations. For the case described, with an initial energy of 100 times greater than the surrounding material, the height of burst for the surface explosion becomes $HoB = -0.05 \times (100)^{1/3} = -0.232$ (negative values indicate an explosion above the target surface) and for subsurface explosion $HoB = 0.1 \times (100)^{1/3} = 0.464$. On each

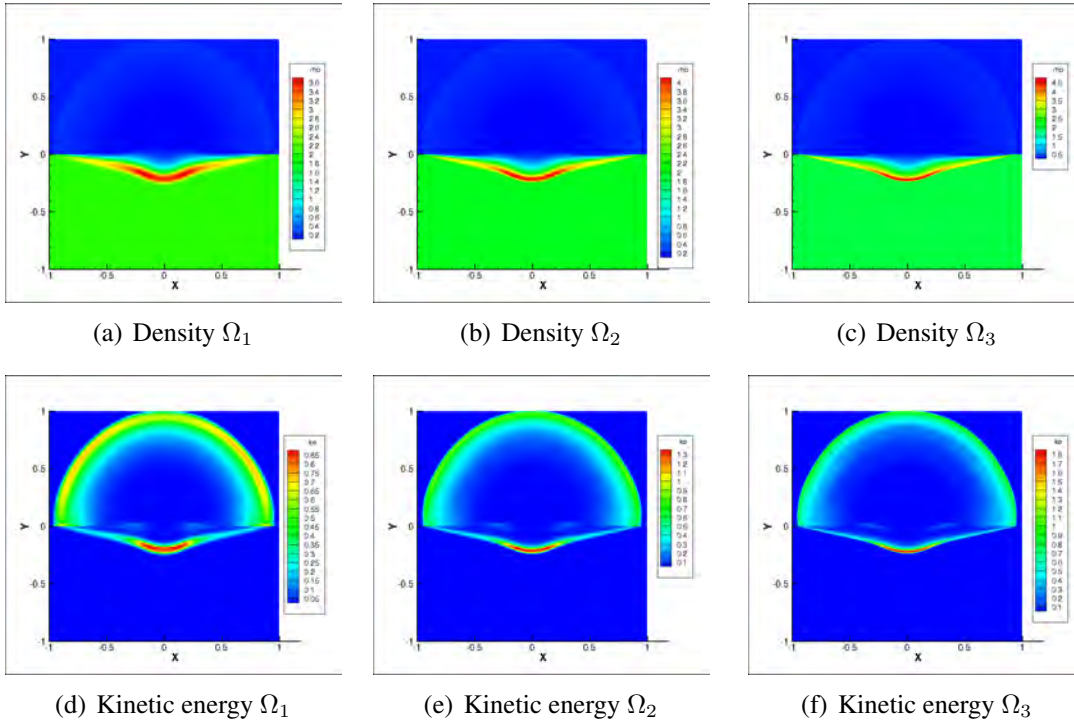


Figure 8. Surface explosion at $t = 0.1$ s.

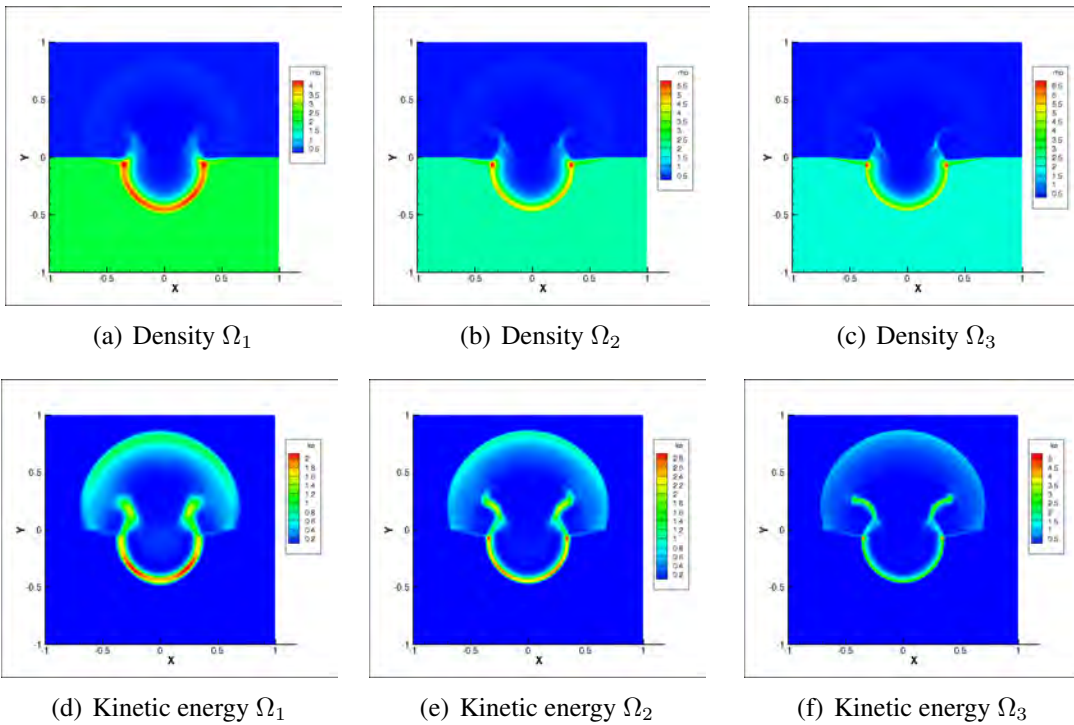


Figure 9. Subsurface explosion at $t = 0.1$ s.

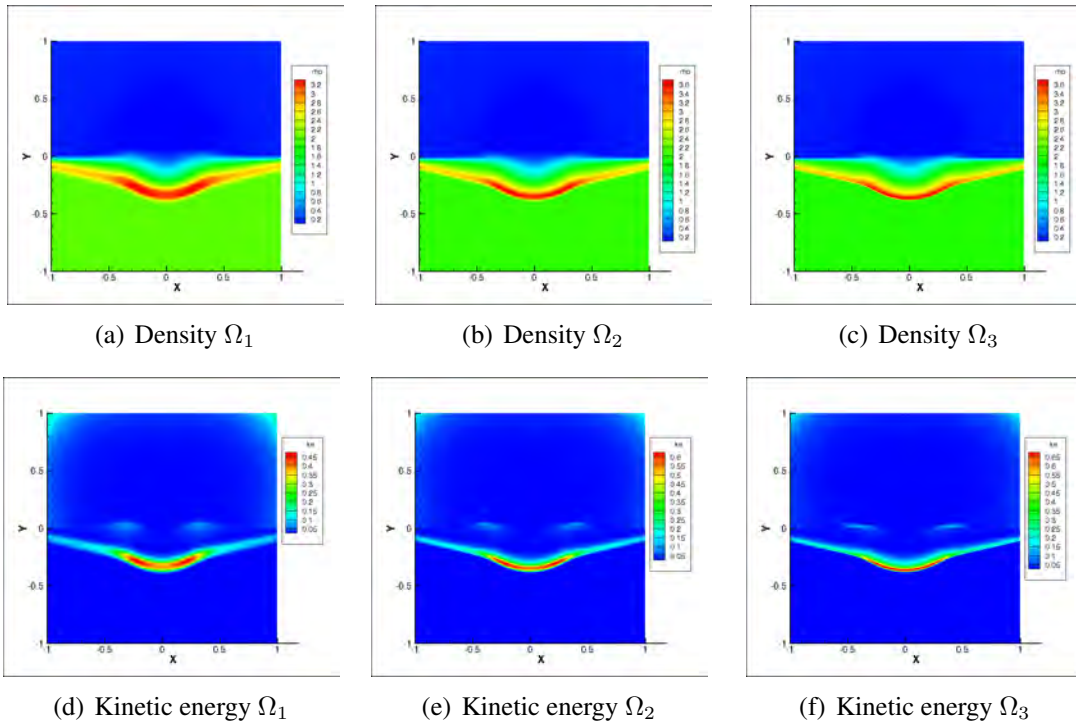


Figure 10. Surface explosion at $t = 0.2$ s.

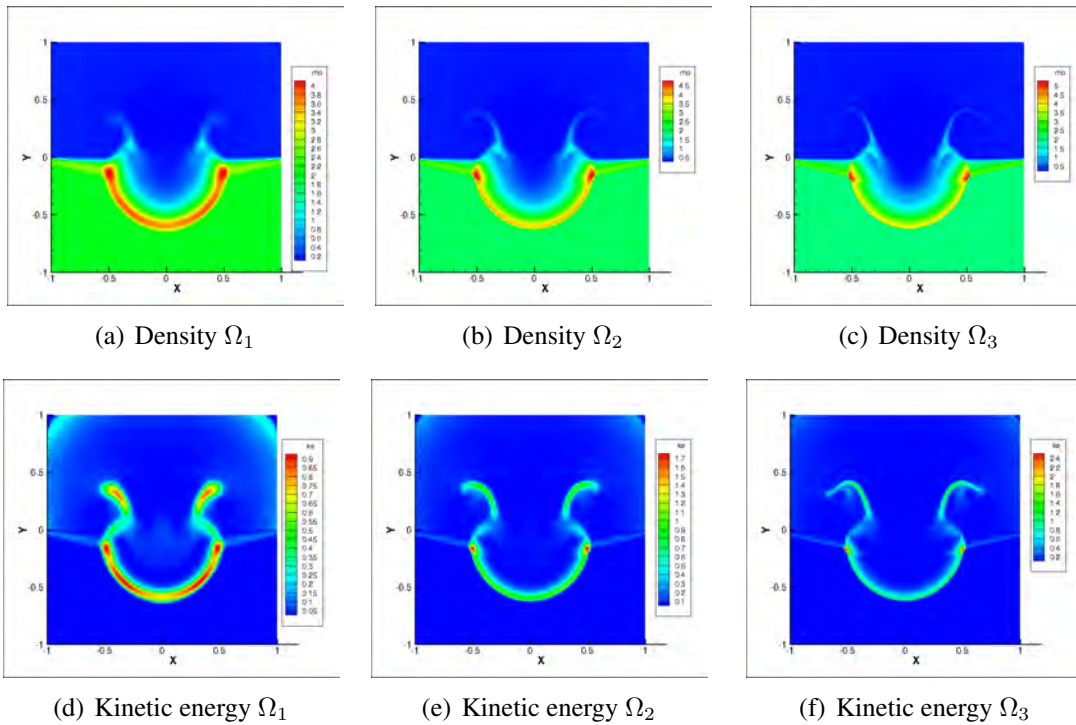


Figure 11. Subsurface explosion at $t = 0.2$ s.

Table 3. Energy coupling factors.

	P^1	P^2	P^3	P^4	P^5
Ω_1	12.36	11.42	11.16	11.13	11.11
Ω_2	10.98	11.38	10.27	10.23	10.21
Ω_3	10.50	10.08	10.03	10.02	10.01

grid, the solution polynomial was increased from P^1 to P^5 to achieve a converged solution. From Table 3, it is clear that low solution polynomials, even on fine grids, over-predict the coupling factor. However, by increasing the solution polynomial order, convergence is shown clearly on Ω_3 , and the 10 times efficiency factor is observed. It should be noted that the coupling factor is infinity near simulation start-up, since the surface explosion has not yet met the target body. Hence, the factors should be monitored after shock wave from the surface explosion comes in contact with the target body. After this point, a clear maximum can be observed. While this simulation does not account for the large explosive energies associated with actual nuclear devices, it does give the authors confidence in moving forward.

C. 100 Kiloton Test

In the prior simulation, 100 units of energy were initialized as the explosive device. Now, consider and initial energy equivalent to 100 kiloton of Trinitrotoluene (TNT), or 4.184×10^{14} joules. The computational domain is extended to 10 meters in x and y directions, and the size of the explosive device is kept at a radius of 0.1 meters. The outside air is initialized at $\rho = 1.225 \frac{kg}{m^3}$ and the pressure as one atmosphere, or $p = 101325$ pascals. The target body is set at the same pressure, but at a density of twenty times greater than that of the outside air. Again, the ideal gas equation of state is used for the entire domain. The locations are initialized such that a factor of 10 should be observed. Figure 12 (a) and (b) illustrate simulations on the computational grids Ω_2 and Ω_3 respectfully for P^1 , P^2 , and P^3 reconstructions. An infinite coupling factor is observed at very small times, but drops significantly through the simulation. No clear maximum coupling factor is observed, unlike the previous case, however both grids demonstrate factors between 8 and 10 for a significant amount of time. Hence, both the benefits of subsurface explosions are demonstrated and the expected coupling factor of roughly 10 is observed.

VII. CONCLUSIONS AND FUTURE WORK

The CPR method provides a computationally efficient and accurate method for simulating blast and shock waves. In addition, the implemented limiter and shock sensor have produced excellent results. The use of GPUs has proved to be a valuable computational resource, as simulations can be carried out at speeds orders of magnitude faster than the CPU counterpart. Finally, the desired efficiency factors have been simulated, verifying the benefits for subsurface explosions. Many assumptions and simplifications were made in order to achieve this. A physically accurate temperature model is still required, as the ideal gas equation of state does not hold for the temperatures associated with a nuclear explosion. In addition, the target body was specified as simply a high dense fluid. This of course does not capture all the effects an actual material would exhibit. The next step is to implement a better equation of state for higher temperatures and replace the high dense fluid with an actual material, yielding a more physically accurate problem, and perhaps the

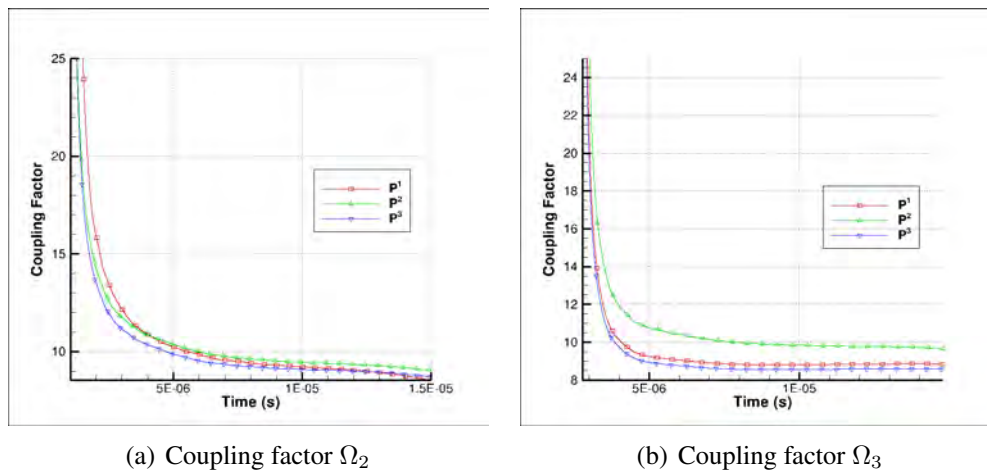


Figure 12. 100 kiloton device coupling factors.

20 times coupling factor efficiency.

Acknowledgments

This research has been supported by a NIAC (NASA Innovative Advanced Concepts) Phase 2 study entitled “An Innovative Solution to NASA’s Asteroid Impact Threat Mitigation Grand Challenge and Its Flight Validation Mission Design.”

References

- ¹ National Research Council. *Defending Planet Earth: Near-Earth Object Surveys and Hazard Mitigation Strategies*. 2010.
- ² J. Sanchez, M. Vasile, & G. Radice. On the consequences of a fragmentation due to a NEO mitigation strategy. IAC-08-C1.3.10, Glasgow, United Kingdom, September 29 - October 3 2008. 59th International Astronautical Congress.
- ³ B. Wie. Hypervelocity nuclear interceptors for asteroid disruption. *Acta Astronautica*, 90:146-155, 2013.
- ⁴ A. Pitz, B. Kaplinger, G. Vardaxis, T. Winkler, & B. Wie. Conceptual design of a hypervelocity asteroid intercept vehicle (HAIV) and its flight validation mission. *Acta Astronautica*, 94:42-56, 2014.
- ⁵ B. Barbee, B. Wie, M. Steiner, & K. Getzandanner. Conceptual design of HAIV flight demonstration mission. *AIAA*, (2013-4544), 2013.
- ⁶ B. Kaplinger, P. D. Premaratne, C. Setzer, & B. Wie. GPU-Accelerated 3D modeling and simulation of a blended kinetic impact and nuclear subsurface explosion. *AIAA*, (2013-4548), 2013.
- ⁷ National Academy of Sciences. *Effects of Nuclear Earth-Penetrator and Other Weapons*. 2005.

- ⁸ P. Premaratne, B. J. Zimmerman, C. Setzer, J. Harry, & B. Wie. Nuclear explosion energy coupling models for optimal fragmentation of asteroids. *American Astronomical Society*, 2014.
- ⁹ H. T. Huynh. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. *AIAA*, (2007-4079), 2007.
- ¹⁰ Z. J. Wang. *Adaptive High-Order Methods in Computational Fluid Dynamics*. 2011.
- ¹¹ M. Hoffmann, C-D. Munz, & Z. J. Wang. Efficient implementation of the CPR formulation for the Navier-Stokes equations on GPUs. *ICCFD*, 7-2603, 2012.
- ¹² B. J. Zimmerman & Z. J. Wang. The efficient implementation of correction procedure via reconstruction with GPU computing. *AIAA*, (2013-2692), 2013.
- ¹³ V. V. Rusanov. Calculation of interaction of non-steady shock waves with obstacles. *Journal of Computational Mathematical Physics*, USSR(1), 1961.
- ¹⁴ P. L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43:357-372, 1981.
- ¹⁵ J. C. Tannehill, A. A. Anderson, & R. H. Pletcher. *Computational Fluid Mechanics and Heat Transfer*. 2:1997.
- ¹⁶ C. W. Shu. Total-variation-diminishing time discretizations. *SIAM Journal on Scientific and Statistical Computing*, 9, 1988.
- ¹⁷ O. C. Zienkiewicz & R. C. Taylor. *The Finite Element Method The Basics*. 2000.
- ¹⁸ P. O. Persson & J. Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. *AIAA*, (2006-112), 2006.
- ¹⁹ NVIDIA. *CUDA C Programming Guide*. 2014.
- ²⁰ P. Collela & P. R. Woodward. The numerical of two-dimensional flow with strong shocks. *Journal of Computational Physics*, 54:115-173, 1984.
- ²¹ E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 3:2009.
- ²² C. E. Needham. *Shock Wave and High Pressure Phenomena Blast Waves*. 2010.