Influences of Meshing and High-Performance Computing towards Advancing the Numerical Analysis of High-Velocity Impacts

Arash Ramezani and Hendrik Rothe
University of the Federal Armed Forces
Hamburg, Germany
Email: ramezani@hsu-hh.de, rothe@hsu-hh.de

***Abstract*—By now, computers and software have spread into all fields of industry. The use of finite-difference and finite-element computer codes to solve problems involving fast, transient loading is commonplace. A large number of commercial codes exist and are applied to problems ranging from fairly low to extremely high damage levels. Therefore, extensive efforts are currently made in order to improve the safety by applying certain numerical solutions. For many engineering problems involving shock and impact, there is no single ideal numerical method that can reproduce the various aspects of a problem. An approach which combines different techniques in a single numerical analysis can provide the "best" solution in terms of accuracy and efficiency. But, what happens if code predictions do not correspond with reality? This paper discusses various factors related to the computational mesh that can lead to disagreement between computations and experience. Furthermore, the influence of high-performance computing is a main subject of this work. The goal is to find an appropriate technique for simulating composite materials and thereby improve modern armor to meet current challenges. Given the complexity of penetration processes, it is not surprising that the bulk of work in this area is experimental in nature. Terminal ballistic test techniques, aside from routine proof tests, vary mainly in the degree of instrumentation provided and hence the amount of data retrieved. Here, both the ballistic trials as well as the analytical methods will be discussed.**

Keywords-solver methologies; simulation models; meshing; high-performance computing; high-velocity impact; armor systems.

I. INTRODUCTION

In the security sector, failing industrial components are ongoing problems that cause great concern as they can endanger people and equipment. Therefore, extensive efforts are currently made in order to improve the safety of industrial components by applying certain computer-based solutions. To deal with problems involving the release of a large amount of energy over a very short period of time, e.g., explosions and impacts, there are three approaches, which are discussed in detail in [1].

As the problems are highly non-linear and require information regarding material behavior at ultra-high loading rates, which are generally not available, most of the work is experimental and may cause tremendous expenses. Analytical approaches are possible if the geometries involved are relatively simple and if the loading can be described through boundary conditions, initial conditions, or a combination of the two. Numerical solutions are far more general in scope and remove any difficulties associated with geometry [2].

For structures under shock and impact loading, numerical simulations have proven to be extremely useful. They provide a rapid and less expensive way to evaluate new design ideas. Numerical simulations can supply quantitative and accurate details of stress, strain, and deformation fields that would be very costly or difficult to reproduce experimentally. In these numerical simulations, the partial differential equations governing the basic physics principles of conservation of mass, momentum, and energy are employed. The equations to be solved are time-dependent and nonlinear in nature. These equations, together with constitutive models describing material behavior and a set of initial and boundary conditions, define the complete system for shock and impact simulations.

The governing partial differential equations need to be solved in both time and space domains (see Figure 1). The solution for the time domain can be achieved by an explicit method. In the explicit method, the solution at a given point in time is expressed as a function of the system variables and parameters, with no requirements for stiffness and mass matrices. Thus, the computing time at each time step is low but may require numerous time steps for a complete solution.

The solution for the space domain can be obtained utilizing different spatial discretization techniques, such as Lagrange [3], Euler [4], Arbitrary Lagrange Euler (ALE) [5], or "mesh free" methods [6]. Each of these techniques has its unique capabilities, but also limitations. Usually, there is not a single technique that can cope with all the regimes of a problem [7].

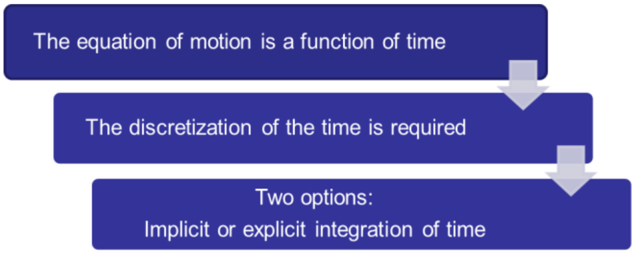
The equation of motion is a function of time
The discretization of the time is required
Two options:
Implicit or explicit integration of time

Figure 1. Discretization of time and space is required.

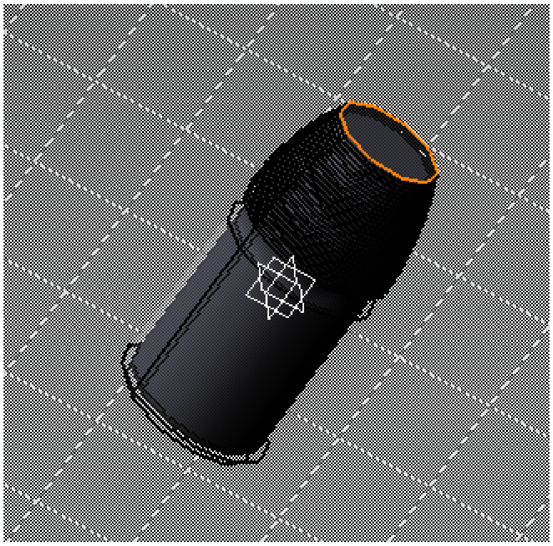

Figure 2. Native CAD geometry of an exemplary projectile.

This work will focus on high-speed dynamics, esp. impact simulations. By using a computer-aided design (CAD) neutral environment that supports direct, bidirectional, and associative interfaces with CAD systems, the geometry can be optimized successively. Native CAD geometry can be used directly without a translation to IGES or other intermediate geometry formats [8]. An example is given in Figure 2.

The work will also provide a brief overview of ballistic tests to offer some basic knowledge of the subject, serving as a basis for the comparison and verification of the simulation results.

The objective of this work is to compare current simulation methodologies to find the most suitable model for high-speed dynamics and impact studies. Lagrange, Euler, ALE, and "mesh free" methods, as well as combinations of these methods, are described and applied to a modern amor structure impacted by a projectile. It aims to clarify the following issues: What is the most suitable simulation model? How does the mesh density affect the results? What are the benefits of high-performance computing?

The results shall be used to improve the safety of ballistic structures, esp. for armored vehicles. Instead of running expensive trials, numerical simulations should be applied to identify vulnerabilities of structures. Contrary to the experimental results, numerical methods allow an easy and comprehensive study of all mechanical parameters.

Modeling will also help to understand how the armor schemes behave during impact and how the failure processes can be controlled to our advantage.

After a brief introduction and description of the different methods of space discretization in Section III, there is a short section on ballistic trials where the experimental set-up is depicted, followed by Section V describing the analysis with numerical simulations. In Section VI, the possible deployment of high-performance computing is discussed. The paper ends with a concluding paragraph.

II. State-of-the-art

Simulating penetration and perforation events requires a numerical technique that allows one body (penetrator) to pass through another (target). Traditionally, these simulations have been performed using either an Eulerian approach, i.e., a non-deformable (fixed) mesh with material advecting among the cells, or using a Lagrangian approach, i.e., a deformable mesh with large deformations. The main point of criticism of the Eulerian approach has been that the shape of the penetrating body, usually an idealized rigid projectile, becomes "fuzzy" as the penetration simulation proceeds, due to the mixing of advected materials in the fixed Eulerian cells. Lagrangian methods require some form of augmentation to minimize or eliminate large mesh distortions. The so-called "pilot hole" technique and the material erosion are the two most often used augmentations for Lagrangian penetration simulations. In the pilot hole technique, elements are removed a priori from the target mesh along the penetrator trajectory, which works well for normal impacts where the trajectory is known a priori. The latter technique removes distorted elements from the simulation based upon a user supplied criterion. They are also removed along the penetrator trajectory, but with no general guidance for selecting certain criteria, i.e., they are ad hoc.

The focus of the present work is to assess a relatively new class of numerical methods, so-called mesh free methods, which offer analysts an alternate analytical technique for simulating this class of ballistic problems without a priori trajectory knowledge or the need to resort to ad hoc criteria. The assessment is made by comparing projectile residual speeds provided by the various techniques, when used to simulate a ballistic impact experiment. The techniques compared are the mesh free method known as Smooth Particle Hydrodynamics (SPH), a multi-material ALE technique, and Lagrangian with material erosion. Given that comparing these inherently different methods is hardly possible, large efforts have been made to minimize the numerous ancillary aspects of the different simulations and focus on the unique capabilities of the techniques.

III. Methods of Space Discretization

The spatial discretization is performed by representing the fields and structures of the problem using computational points in space, usually connected with each other through computational grids. Generally, the following applies: the finer the grid, the more accurate the solution. For problems of dynamic fluid-structure interaction and impact, there typically is no single best numerical method which is applicable to all parts of a problem. Techniques to couple types of numerical solvers in a single simulation can allow the use of the most appropriate solver for each domain of the problem [9].

The most commonly used spatial discretization methods are Lagrange, Euler, ALE (a mixture of Lagrange and Euler), and mesh-free methods, such as Smooth Particles Hydrodynamics (SPH) [10].

A. Lagrange

The Lagrange method of space discretization uses a mesh that moves and distorts with the material it models as a result of forces from neighboring elements (meshes are imbedded in material). There is no grid required for the external space, as the conservation of mass is automatically satisfied and material boundaries are clearly defined. This is the most efficient solution methodology with an accurate pressure history definition.

The Lagrange method is most appropriate for representing solids, such as structures and projectiles. If however, there is too much deformation of any element, it results in a very slowly advancing solution and is usually terminated because the smallest dimension of an element results in a time step that is below the threshold level.

B. Euler

The Euler (multi-material) solver utilizes a fixed mesh, allowing materials to flow (advect) from one element to the next (meshes are fixed in space). Therefore, an external space needs to be modeled. Due to the fixed grid, the Euler method avoids problems of mesh distortion and tangling that are prevalent in Lagrange simulations with large flows. The Euler solver is very well-suited for problems involving extreme material movement, such as fluids and gases. To describe solid behavior, additional calculations are required to transport the solid stress tensor and the history of the material through the grid. Euler is generally more computationally intensive than Lagrange and requires a higher resolution (smaller elements) to accurately capture sharp pressure peaks that often occur with shock waves.

C. ALE

The ALE method of space discretization is a hybrid of the Lagrange and Euler methods. It allows redefining the grid continuously in arbitrary and predefined ways as the calculation proceeds, which effectively provides a continuous rezoning facility. Various predefined grid motions can be specified, such as free (Lagrange), fixed (Euler), equipotential, equal spacing, and others. The ALE method can model solids as well as liquids. The advantage of ALE is the ability to reduce and sometimes eliminate difficulties caused by severe mesh distortions encountered by the Lagrange method, thus allowing a calculation to continue efficiently. However, compared to Lagrange, an additional computational step of rezoning is employed to move the grid and remap the solution onto a new grid [7].

D. SPH

The mesh-free Lagrangian method of space discretization (or SPH method) is a particle-based solver and was initially used in astrophysics. The particles are imbedded in material and they are not only interacting mass points but also interpolation points used to calculate the value of physical variables based on the data from neighboring SPH particles, scaled by a weighting function. Because there is no grid defined, distortion and tangling problems are avoided as well. Compared to the Euler method, material boundaries and interfaces in the SPH are rather well defined and material separation is naturally handled. Therefore, the SPH solver is ideally suited for certain types of problems with extensive material damage and separation, such as cracking. This type of response often occurs with brittle materials and hypervelocity impacts. However, mesh-free methods, such as SPH, can be less efficient than mesh-based Lagrangian methods with comparable resolution.

Figure 3 gives a short overview of the solver technologies mentioned above. The crucial factor is the grid that causes different outcomes.

The behavior (deflection) of the simple elements is well-known and may be calculated and analyzed using simple equations called shape functions. By applying coupling conditions between the elements at their nodes, the overall stiffness of the structure may be built up and the deflection/distortion of any node – and subsequently of the whole structure – can be calculated approximately [12].

Due to the fact that all engineering simulations are based on geometry to represent the design, the target and all its components are simulated as CAD models [13]. Therefore, several runs are necessary: from modeling to calculation to the evaluation and subsequent improvement of the model (see Figure 4).

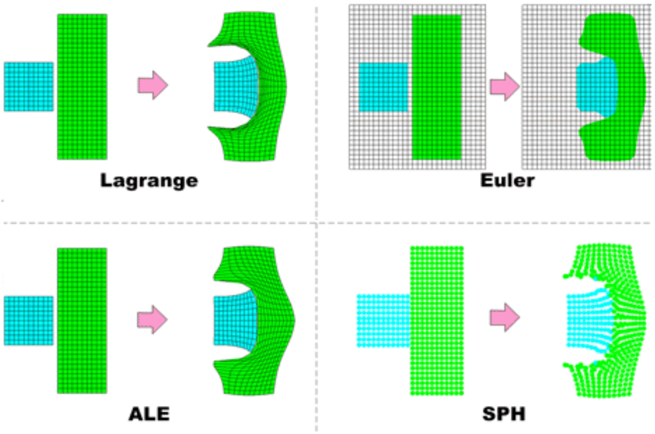

Figure 3. Examples of Lagrange, Euler, ALE, and SPH simulations on an impact problem [11].

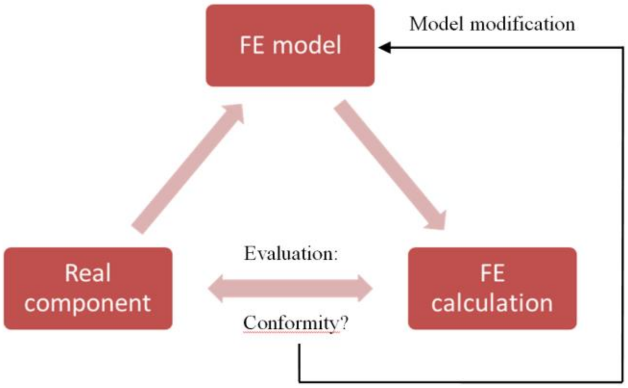

Figure 4. Iterative procedure of a typical FE analysis [12].

The most important steps during an FE analysis are the evaluation and interpretation of the outcomes followed by suitable modifications of the model. For that reason, ballistic trials are necessary to validate the simulation results. They can be used as the basis of an iterative optimization process.

IV. Effects of Meshing

Engineers and scientists use finite element analysis (FEA) software to build predictive computational models of real-world scenarios. The use of FEA software begins with a CAD model that represents the physical parts being simulated as well as knowledge of the material properties and the applied loads and constraints. This information enables the prediction of real-world behavior, often with very high levels of accuracy.

The numerical model becomes complete once the mesh is created. Different phenomena and analyses require varied mesh settings. For example, in wave propagation problems, such as modeling elastic waves in structural mechanics or electromagnetic waves in radio frequency analysis, the size of the largest element has to be substantially smaller than the wavelength in order to resolve the problem. In fluid flow, boundary layer meshes may be required in order to resolve boundary layers, while the cell Reynolds number may determine the element size in the bulk of the fluid.

In many cases, different parts of a CAD geometry have to be meshed separately. The model variables have to be matched by the FEA software at the interfaces between the different parts. The matching can be done through continuity constraints (i.e., boundary conditions that relate the finite element discretizations of the different parts to each other). Due to the possible non-local character of these conditions, they are often called multi-point constraints.

The accuracy that can be obtained from any FEA model is directly related to the finite element mesh that is used. The finite element mesh is used to subdivide the CAD model into smaller domains called elements, over which a set of equations are solved. These equations approximately represent the governing equation of interest via a set of polynomial functions defined over each element. As these elements are made smaller and smaller, as the mesh is refined, the computed solution will approach the true solution.

This process of mesh refinement is a key step in validating any finite element model and gaining confidence in the software, the model, and the results.

A good finite element analyst starts with both an understanding of the physics of the system that is to be analyzed and a complete description of the geometry of the system. This geometry is represented via a CAD model. A typical CAD model will accurately describe the shape and structure, but often also contain cosmetic features or manufacturing details that can prove to be extraneous for the purposes of finite element modeling. The analyst should put some engineering judgment into examining the CAD model and deciding if these features and details can be removed or simplified prior to meshing. Starting with a simple model and adding complexity is almost always easier than starting with a complex model and simplifying it.

The analyst should also know all of the physics that are relevant to the problem, the materials properties, the loads, the constraints, and any elements that can affect the results of interest. These inputs may have uncertainties in them. For instance, the material properties and loads may not always be precisely known. It is important to keep this in mind during the modeling process, as there is no benefit in trying to resolve a model to greater accuracy than the input data admits.

Once all of this information is assembled into an FEA model, the analyst can begin with a preliminary mesh. Early in the analysis process, it makes sense to start with a mesh that is as coarse as possible – a mesh with very large elements. A coarse mesh will require less computational resources to solve and, while it may give a very inaccurate solution, it can still be used as a rough verification and as a check on the applied loads and constraints.

After computing the solution on the coarse mesh, the process of mesh refinement begins. In its simplest form, mesh refinement is the process of resolving the model with successively finer and finer meshes, comparing the results between these different meshes. This comparison can be done by analyzing the fields at one or more points in the model or by evaluating the integral of a field over some domains or boundaries.

By comparing these scalar quantities, it is possible to judge the convergence of the solution with respect to mesh refinement. After comparing a minimum of three successive solutions, an asymptotic behavior of the solution starts to emerge, and the changes in the solution between meshes become smaller. Eventually, these changes will be small enough that the analyst can consider the model to be converged. This is always a judgment call on the part of the analyst, who knows the uncertainties in the model inputs and the acceptable uncertainty in the results.

When it comes to mesh refinement, there is a suite of techniques that are commonly used. An experienced user of FEA software should be familiar with each of these techniques and the trade-offs between them.

Reducing the element size is the easiest mesh refinement strategy, with element sizes reduced throughout the modeling domains. This approach is attractive due to its simplicity, but the drawback is that there is no preferential mesh refinement in regions where a locally finer mesh may be needed (see Figure 5).

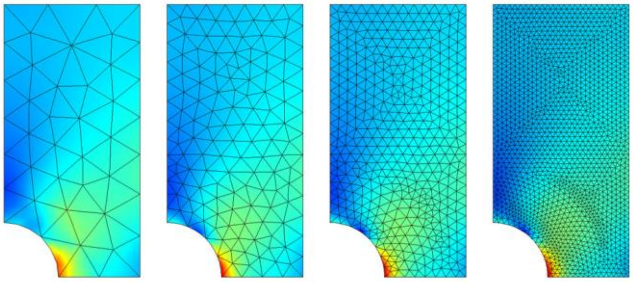

Figure 5. The stresses in a plate with a hole, solved with different element sizes.

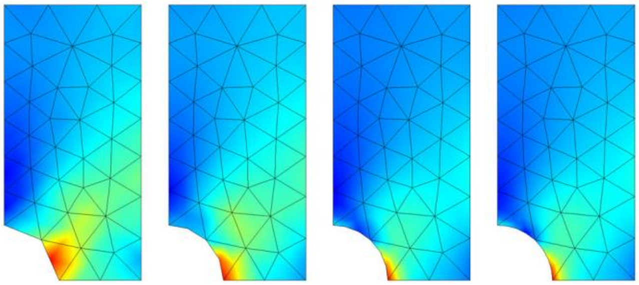

Figure 6. The same finite element mesh, but solved with different element orders.

Increasing the element order is advantageous in the sense that no remeshing is needed; the same mesh can be used, but with different element orders. Remeshing can be time consuming for complex 3D geometries or the mesh may come from an external source and cannot be altered. The disadvantage to this technique is that the computational requirements increase faster than with other mesh refinement techniques (see Figure 6).

V. Ballistic Trials

Ballistics is an essential component for the evaluation of our results. Here, terminal ballistics is the most important sub-field. It describes the interaction of a projectile with its target. Terminal ballistics is relevant for both small and large caliber projectiles. The task is to analyze and evaluate the impact and its various modes of action. This will provide information on the effect of the projectile and the extinction risk.

Given that a projectile strikes a target, compressive waves propagate into both the projectile and the target. Relief waves propagate inward from the lateral free surfaces of the penetrator, cross at the centerline, and generate a high tensile stress. If the impacts were normal, we would have a two-dimensional stress state. If the impacts were oblique, bending stresses would be generated in the penetrator. When the compressive wave was to reach the free surface of the target, it would rebound as a tensile wave. The target could fracture at this point. The projectile could change direction in case of perforation (usually towards the normal of the target surface). A typical impact response is illustrated in Figure 7.

Because of the differences in target behavior due to the proximity of the distal surface, we must categorize targets into four broad groups. In a semi-infinite target, there is no influence of distal boundary on penetration. A thick target is one in which the boundary influences penetration after the projectile has already travelled some distance into the target. An intermediate thickness target is a target where the boundaries exert influence throughout the impact. Finally, a thin target is one in which stress or deformation gradients are negligible throughout the thickness.

There are several methods which may cause a target to fail when subjected to an impact. The major variables are the target and penetrator material properties, the impact velocity, the projectile shape (especially the ogive), the geometry of the target supporting structure, and the dimensions of the projectile and target.

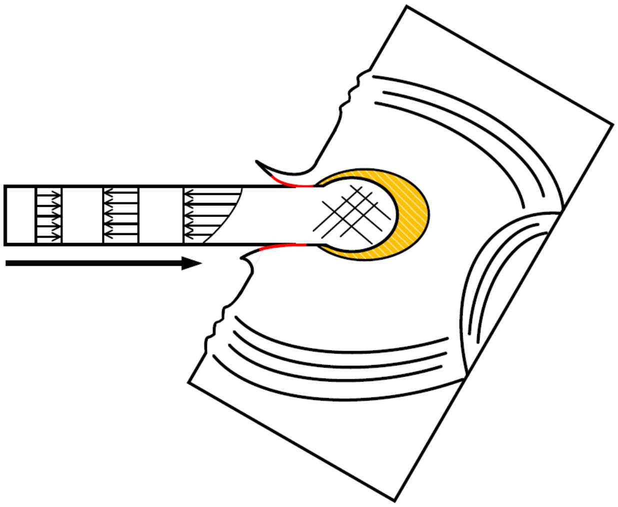

Figure 7. Wave propagation after impact.

The results of the ballistic tests were provided prior to the simulation work to aid calibration. A series of metal plate impact experiments, using several projectile types, have been performed. For the present comparative study, the only target considered is 0.5 inch (12.7 mm) thick 6061-T6 aluminum plate. The plate has a free span area of 8 by 8 inches (203 by 203 mm) and was fixed in place.

The plate was nominally center impacted by a blunt projectile, also made from 6061-T6 aluminum, with an impact speed of 3181 feet/second (970 meters/second). The orientation of the projectile impact was intended to be normal to the target. The projectile is basically a right circular cylinder of length 0.974 inches (24.7 mm) and diameter 0.66 inch (16.7 mm), with a short length of reduced diameter (shoulder) at the rear of the projectile.

The projectile's observed exit speed was 1830 feet/second. The deformed target and projectile are shown in Figures 8 and 9, respectively. As can be seen the target is essentially "drilled out" by the projectile, i.e., a clean hole remains in the target plate. Also, the lack of "petals" on the exit surface of the target indicates the hole was formed by concentrated shear around the perimeter of the hole.

The deformed projectiles, shown in Figure 9, indicate the increasing amount of projectile deformation as it perforates increasingly thicker targets: 0.125 to 0.5 inch. The deformed projectile on the right is the case of present interest. It is worth noting that the simulation of deformable projectiles perforating deformable targets is a challenging class of ballistic simulations. The vast majority of perforation simulations involve nearly rigid projectiles impacting deformable targets. Although deformable projectile calculations form a special, and limited, class in ballistics, establishing confidence in the simulation of this challenging class of problems will lend further confidence to the comparatively easier simulation of near rigid projectile perforating deformable targets [14].

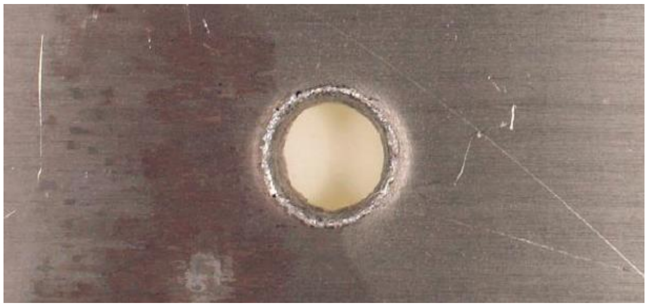

Figure 8. Front view of a perforated aluminum 0.5 inch thick target.

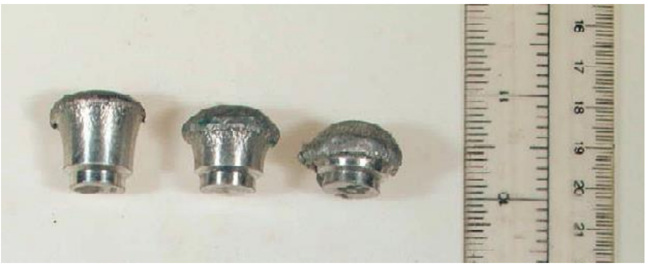

Figure 9. Deformed 6061-T6 aluminum projectiles after perforation 0.125, 0.25, and 0.5 inch thick (left-to- right) aluminum targets.

VI. Numerical Simulation

The ballistic tests are followed by computational modeling of the experimental set-up.

Three mesh refinement models were constructed using the two-dimension axisymmetric solver in ANSYS. While the three-dimensional solver could also be used, using the two-dimension axisymmetric solver allows more efficient solutions, especially with a large number of elements. The particulars of the three meshes are summarized in Table I.

Figure 10 shows two of the three axisymmetric mesh configurations. The mesh discretizations are similar in that each mesh uses one number as the basis for determining the number and size of all the elements in the mesh. The target plate elements immediately below the projectile have the same mesh refinement as the projectile. The configuration is based on [14].

A suite of impact simulations was performed using the above-described 6061-T6 aluminum projectile and 6061-T6 aluminum target. The projectile was given an initial velocity of 3181 feet/second (970 meters/second) and the projectile's speed was recorded at a point near the rear of the projectile. The resulting residual speed was thought to best correspond to the experimental measurement technique for residual speed.

TABLE I. Summary of mesh configurations

	Smallest Element (mm)	Number of Elements
Coarse	0.4445	3,174
Medium	0.22225	12,913
Fine	0.14816	28,922

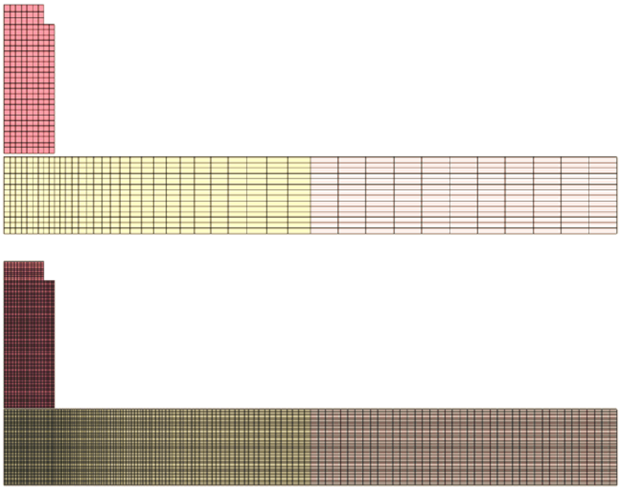

Figure 10. Two of the three axisymmetric mesh discretizations.

The overall projectile and target plate dimensions were previously given in the description of the ballistic experiment. The axisymmetric model is fully constrained around the outer diameter of the target plate, i.e., fully fixed (clamped). Different solver methodologies have been applied. The comparison is presented in the following section.

A. *Solver Evaluation*

Using the Johnson-Cook failure criterion eliminates the need to select an erosion criterion and a value for the criterion at which to erode elements. These are two significant difficulties most often overlooked when using an erosion-based simulation technique. Many users select an ad hoc erosion criterion and assign ad hoc values for erosion. In so doing, they seem to ignore the fact that the results are then also ad hoc, which is not desirable when making predictive calculations.

As mentioned above, the Johnson-Cook failure model is not regularized via element characteristic lengths. Thus, we expect the results to be mesh-dependent. It is the purpose of this section to assess this mesh dependency using four successively refined meshes. Subsequently, theses Lagrange erosion results will be compared with the corresponding ALE and SPH results.

1) Lagrange method: Figure 11 shows the initial and deformed (t = 0.053 ms) mesh configurations for the medium discretized mesh. Also shown is an illustration of the eroded element distribution at the end of the simulation. The eroded elements are indicated relative to their initial position using a different color to differentiate them from the non-eroded elements of the same part. Table II summarizes the residual speed of the projectile for the three mesh configurations considered. With the exception of the medium mesh speed, which indicates a somewhat larger projectile speed reduction, the projectile speeds are decreasing nearly uniformly with increasing mesh refinement.

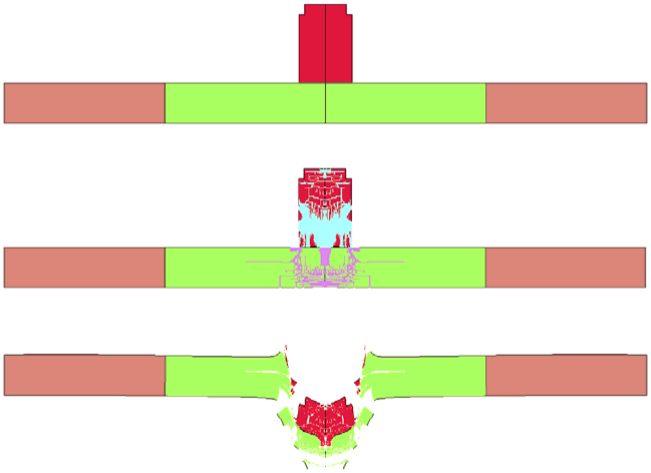

Figure 11. Initial, eroded, and deformed Lagrange elements with a medium mesh configuration.

Figure 12 shows a plot of the residual speed versus the mesh refinement parameter. This plot indicates that the results do not follow the developing trend. Based on this plot, no claim can be made that the results are in the asymptotic regime, much less converged. This is disappointing since the mesh densities for these two cases are likely to be much greater than it would have been attempted in typical three dimensional simulations.

2) ALE method: As mentioned above, failure criteria such as the Johnson-Cook failure criterion, cannot be used with Eulerian formulations as cell (element) deletion is not allowed. If a user attempts to use a failure model, the deletion of failed cells will eventually cause the calculation to terminate inaccurately. Thus, all the ALE simulations in this section omit the Johnson-Cook failure model.

In the absence of a failure criterion, it will be demonstrated that the residual speed of the projectile is quite low. It is the purpose of this section to assess the mesh dependency of the ALE solution using successively refined meshes. Subsequently, these results will be compared with the corresponding Lagrange and SPH results.

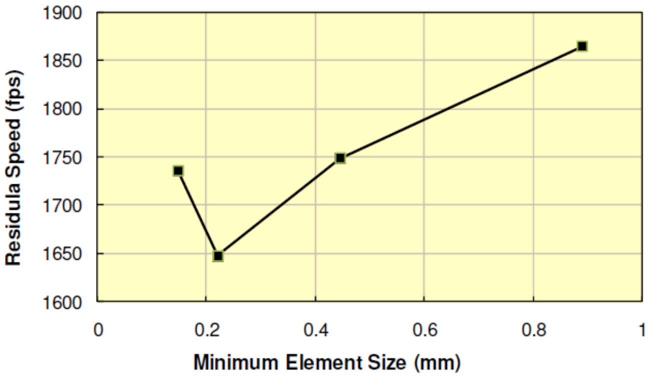

Figure 12. Plot of residual speed versus mesh refinement parameter.

TABLE II. COMPARISON OF LAGRANGE WITH EROSION AND ALE PROJECTILE RESIDUAL SPEEDS

	Residual Speed (fps)	
Mesh	**Lagrange**	**ALE**
Coarse	1748	1693
Medium	1647	1788
Fine	1737	1834
Experiment	1830	

Note: although the same mesh densities are used in both the Lagrange and ALE simulations in this demonstration, ALE mesh densities generally need to be greater than corresponding Lagrange with erosion mesh densities. The advection of materials from cell-to-cell, and especially the assumption of uniform strain-rate increments for all materials occupying a cell, introduces numerical errors to the ALE solution that can only be minimized by increasing the mesh densities. For the present demonstration, it is posited that the Lagrange mesh densities are greater than they would typically be for such a perforation simulation, making the ALE mesh densities probably appear typical in terms of expectations.

Table II compares the previous Lagrange with erosion results with the corresponding ALE projectile residual speeds. The vast majority of perforation simulations involve nearly rigid projectiles impacting deformable targets. Figure 13 shows the ALE simulation at t = 0.1 ms with a medium number of elements. It is interesting to note that the ALE deformed projectile is quite similar in shape to the deformed projectile after the test.

3) SPH method: Failure criteria like the Johnson-Cook failure criterion, are not typically used with the Smooth Particle Hydrodynamic (SPH) formulations, as particle methods are designed to avoid mesh distortions, which is the primary motivation for using failure/erosion criteria. It is the purpose of this section to assess the mesh dependency of the SPH solution using three successively refined particle meshes. These results will be compared with the corresponding Lagrange with erosion and ALE results.

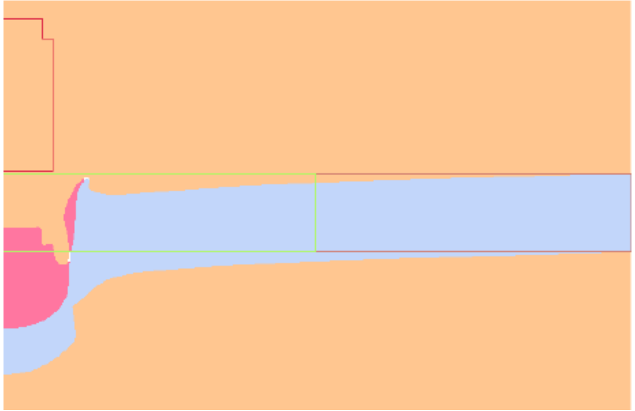

Figure 13. ALE simulation with a medium discretized mesh (t = 0.1 ms).

Much like the mesh refinements used in the Lagrange and ALE calculations, refinements of the SPH particle spacing requires changing the spacing in both the impacted region of the target plate and the projectile, which is also modeled using SPH particles. Figure 14 shows the coarsest SPH model with the projectile and center of the target plate. It was modeled using SPH particles, while the outer portion of the plate was modeled with Lagrange solid elements. Table III summarizes the three SPH meshes.

Figure 15 shows the initial and final (t = 0.1 ms) deformed projectile and target plate configuration for the finest SPH mesh. In addition to the target plate 'plug' being removed from the plate by the projectile (darker brown particles on the right side of target plate), there is considerable front surface ejecta of both the projectile (light brown particles) and the target plate (darker brown particles). For this mesh refinement, the deformed projectile remains relatively intact, with the exception of the front surface ejecta and portions of the projectile that remain attached to the target plate.

B. *Simulation Results*

In the previous sections, the results from a laboratory experiment were used as a basis to assess the accuracy of the numerical simulations with respect to mesh refinement.

Examining the Lagrange results first without considering the experimental observation, it would seem like the Lagrange method provides the "best" results. All three sets of the Lagrange-with-erosion results have an observed order of convergence which is less than two and thus considered a favorable indication, since few numerical methods have orders of accuracy greater than two.

A general trend seems to be that, as the mesh is refined, the resulting deformed projectile more closely resembles the observed deformed projectile. The exception to this trend is the "point" that protrudes from the front of the projectile. Due to target elements, this "point" appears to be eroded erroneously along the axis of symmetry.

Also, it can be deduced that, for ALE simulations, meshes need to be more dense than it is required for the corresponding Lagrange mesh density. The current status is as follows: the ALE meshes were refined enough, and the Lagrange meshes were more refined than necessary. It is more likely that the advection of material, e.g., from target plate into the surrounding vacuum, over-predicts the motion of the target plate, thus effectively reducing its stiffness and allowing for a "soft catch" of the projectile and an associated reduced projectile residual speed.

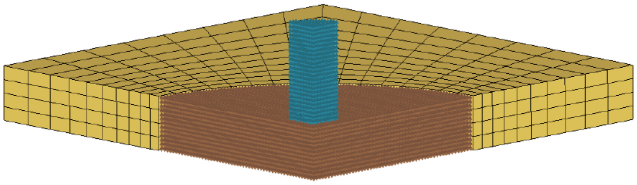

Figure 14. Coarsest SPH model (0.96 mm particle spacing).

TABLE III. SUMMARY OF SPH RESIDUAL SPEEDS FOR THREE PARTICLE MESH REFINEMENTS

		Number of Particles		
Mesh	**Particle Spacing (mm)**	**Projectile**	**Target**	**Residual Speed (fps)**
Coarse	0.96	1,536	28,665	1094
Medium	0.64	4,860	98,080	1312
Fine	0.43	17,064	333,840	1424
Experiment				1830

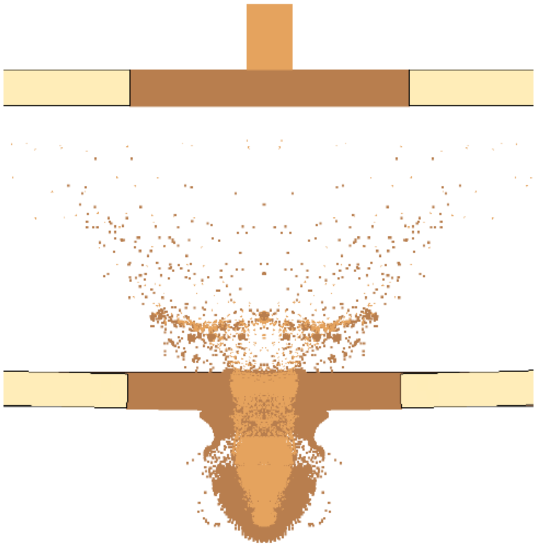

Figure 15. Initial and final (t = 0.1 ms) configurations for finest SPH mesh (0.43 mm spacing).

Here, it needs to be recalled that the Johnson-Cook failure model cannot be included in the Eulerian simulations as the notion of removal of a cell in the Eulerian context is not permitted. However, these results do indicate that they converge in or at least near the asymptotic range.

Just like the ALE results, the SPH residual speeds increase with increasing mesh density, thus being opposite to the general trend for the Lagrange results. An increasing speed with mesh refinement leads to predictions for a converged result that is greater than the calculated values. Finally, the SPH deformed projectile, previously shown in Figure 15, bears little or no resemblance to the deformed projectile recovered after the perforation test (see Figure 9). Thus, the SPH residual speed results should perhaps be considered reasonable, at least compared to the ALE results. However, the lack of uniformity of the deformed projectile shape between the SPH simulations and the actual experiment might be an indication that the "right" answer might be obtained for the "wrong" reason. Future perforation experiments should include additional diagnostics, e.g., strain measurements on the target plates, so that assessments of agreement can be more extensive than solely considering residual speed.

The SPH deformed projectile looked the least like the observed deformed projectile of any of the three simulation techniques reported. Rather than forming a rounded impact end on the projectile, the SPH deformed projectile seems to form more of a “jet” with a narrow diameter at the fore and a tapered diameter toward the rear. Also, only the refined mesh appears to maintain the integrity of the projectile, i.e., the other two mesh configurations indicate the projectile separating into two parts. Finally, it appears as if some of the projectile material remains on the inner diameter of the hole formed in the target plate. However, it is uncertain if this was observed in the test.

VII. High-Performance Computing

The objective is to develop and improve the modern armor used in the security sector. To develop better, smarter constructions requires an analysis of a wider range of parameters. However, there is a simple rule of thumb: the more design iterations that can be simulated, the more optimized is the final product. As a result, a high-performance computing (HPC) solution has to dramatically reduce overall engineering simulation time [15].

High-performance computing, otherwise known as HPC, refers to the use of aggregated computing power for handling compute- and data-intensive tasks – including simulation, modeling, and rendering – that standard workstations are unable to address. Typically, the problems under consideration cannot be solved on a commodity computer within a reasonable amount of time (too many operations are required) or the execution is impossible, due to limited available resources (too much data is required). HPC is the approach to overcome these limitations by using specialized or high-end hardware or by accumulating computational power from several units. The corresponding distribution of data and operations across several units requires the concept of parallelization [16]. When it comes to hardware setups, there are two types that are commonly used:

- Shared memory machines.
- Distributed memory clusters.

In shared memory machines, random-access memory (RAM) can be accessed by all of the processing units [17]. Meanwhile, in distributed memory clusters, the memory is inaccessible between different processing units, or nodes [18]. When using a distributed memory setup, there must be a network interconnect to send messages between the processing units (or to use other communication mechanisms), since they do not have access to the same memory space. Modern HPC systems are often a hybrid implementation of both concepts, as some units share a common memory space and some do not.

HPC is primarily used for two reasons. First, thanks to the increased number of central processing units (CPUs) and nodes, more computational power is available. Greater computational power enables specific models to be computed faster, since more operations can be performed per time unit. This is known as the speedup [19].

The speedup is defined as the ratio between the execution time on the parallel system and the execution time on the serial system. The upper limit of the speedup depends on how well the model can be parallelized. Consider, for example, a fixed-size computation where 50% of the code is able to be parallelized. In this case, there is a theoretical maximum speedup of 2. If the code can be parallelized to 95%, it is possible to reach a theoretical maximum speedup of 20. For a fully parallelized code, there is no theoretical maximum limit when adding more computational units to a system. Amdahl’s law explains such a phenomenon (see Figure 16) [20].

Second, in the case of a cluster, the amount of memory available normally increases in a linear fashion with the inclusion of additional nodes. As such, larger and larger models can be computed as the number of units grows. This is referred to as the scaled speedup. Applying such an approach makes it possible to, in some sense, “cheat” the limitations posed by Amdahl’s law, which considers a fixed-size problem. Doubling the amount of computational power and memory allows for a task that is twice as large as the base task to be computed within the same stretch of time. Gustafson-Barsis' law explains this phenomenon (see Figure 17) [21].

HPC adds tremendous value to engineering simulation by enabling the creation of large, high-fidelity models that yield accurate and detailed insights into the performance of a proposed design. HPC also adds value by enabling greater simulation throughput. Using HPC resources, many design variations can be analyzed.

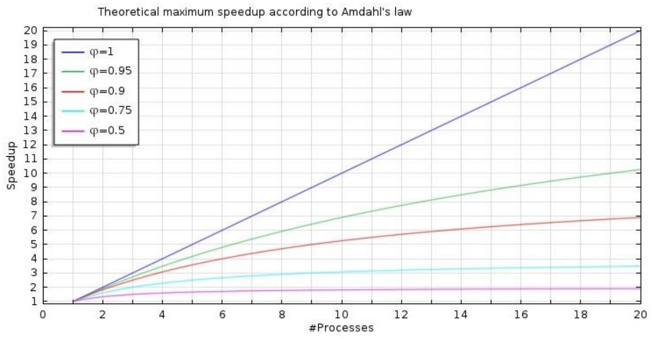

Figure 16. The theoretical maximum speedup, as noted by Amdahl's law.

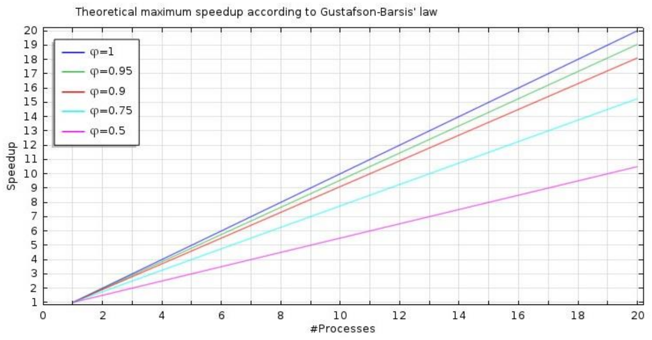

Figure 17. The theoretical maximum speedup, as noted by Gustafson-Barsis' law.

In 1965, Gordon Moore made a prediction that would set the pace for our modern digital revolution. From careful observation of an emerging trend, Moore extrapolated that computing would dramatically increase in power, and decrease in relative cost, at an exponential pace [22]. Moore's Law predicts that the number of transistors that can be economically placed on an integrated circuit will double about every two years. The insight, known as Moore's Law, became the golden rule for the electronics industry, and a springboard for innovation.

Moore's observation transformed computing from a rare and expensive venture into a pervasive and affordable necessity. All of the modern computing technology we know and enjoy sprang from the foundation laid by Moore's Law. From the Internet itself, to social media and modern data analytics, all these innovations stem directly from Moore and his findings.

Performance and cost are two key drivers of technological development. As more transistors fit into smaller spaces, processing power increased and energy efficiency improved, all at a lower cost for the end user. This development not only enhanced existing industries and increased productivity, but it has spawned whole new industries empowered by cheap and powerful computing.

This research will evaluate the performance of the following server generations: HP ProLiant SL390s G7, HP ProLiant DL580 G7, and HP ProLiant DL380p G8.

Taking the influence of the software into account, different versions of ANSYS will be applied here. Regarding the Lagrange solver in a complex 3D multi-material simulation model (modern composite armor structure instead of 6061-T6 aluminum target), the following benchmark is obtained for the different simulations (see Table IV below).

The results indicate the importance of high-performance computing in combination with competitive simulation software to solve current problems of the computer-aided engineering sector.

VIII. CONCLUSION

This work focuses on the comparison of current simulation methodologies to find the most suitable model for high-speed dynamics and impact studies. The influence of meshing on the simulation results is pointed out based on an example. The benefits of high-performance computing are discussed in detail.

The reader is reminded that the ballistic simulation attempted in this work is among the most difficult as both the projectile and target experience significant deformation. The deformation of the projectile as it interacts with the target affects the deformation of the target, and vice versa.

TABLE IV. BENCHMARK TO ILLUSTRATE THE INFLUENCE OF DIFFERENT SERVER AND SOFTWARE GENERATIONS

	ANSYS 14.5	ANSYS 15.0
SL390s G7	27m31s	24m47s
DL580 G7	21m44s	19m51s
DL380p G8	19m16s	14m32s

The introduction of a failure criterion, such as the Johnson-Cook failure criterion, is clearly necessary for Lagrange models, and appears to also be necessary for SPH models. A better overall approach than on-off failure models, like the Johnson-Cook failure model, would be the use of continuum damage models. These models allow for the gradual reduction in strength of highly deformed materials and can be used in all three solution techniques.

Many modern computer-aided modeling, analysis, and manufacturing systems provide both interactive and automatic finite element mesh generation of surface and solid entities that describe the parts or products being virtually engineered as new designs. Unfortunately, for complex products, the interactive approach is too time consuming to factor into the design process and the quality of automatically created meshes often does not meet engineers' criteria for element shape and density. Though commercial finite element analysis packages have some ability to control and direct the automatic mesh generation process, determining a correlation between these user controlled mesh parameters and acceptable quality of the generated mesh is difficult if not impossible. Since the validity of analysis results is heavily dependent upon mesh quality, obtaining better meshes in the shortest amount of time is essential for the integration of FEA into the automated design process.

The importance of mesh refinement has been emphasized in this work. This relatively simple to perform assessment of how the key results change with mesh density is all too often overlooked in computational solid mechanics. Further, establishing that the results are in the asymptotic regime provides some confidence that the mesh density is adequate.

When predictions are required, analysts want as many checks and assurances as possible that their results are credible. Mesh refinement studies provide the analyst some confidence the results are at a minimum not being affected by ad hoc choices of discretization.

A technique that is frequently employed in industry is that of modifying existing nodes and elements. Mesh smoothing routines have likewise long been an effective method of improving mesh quality in a pre-existing mesh. Many techniques are available for performing mesh smoothing. Some of the more advanced ones use gradient-based optimization techniques to quickly determine the optimal distribution of existing nodes. Others iterate using brute-force methods, such as Laplacian smoothing, to improve mesh distribution and corresponding element quality. Beyond this geometrical optimization of element shape, some schemes have been developed to modify and optimize the topology of the mesh by editing the node-adjacency structure of the mesh. Routines and optimizers include methods and operators such as edge swapping, vertex removing, edge collapsing, etc. to edit and improve the mesh topology. Special operators are required for maintaining a valid mesh in the case of quadrilateral and hexahedral meshes. Still other mesh improvement methods involve generating a new mesh based on information learned from previous attempts. Several algorithms use a posteriori techniques to improve the mesh once regions of inaccuracy are located.

Although the above techniques have undoubtedly improved the quality of meshes available to the mesh researcher, the accessibility to such techniques within commercial FEA is still limited. It is a well-accepted fact that it takes software companies years to adopt and dispense new methods and techniques. Often the effectiveness of their implementation is called into question: smoothing algorithms, for example, are restricted by the node/element configuration of the starting mesh and may not be able to improve a mesh to meet the desired criteria. This paper proposes a strategy for generating an optimal mesh within the framework of existing FEA software. Rather than optimizing initial node placement or operations to be performed on existing elements, the mesh control parameters available in a commercial FEA package can be optimized to yield a high-quality mesh [23].

Meshing is considered to be one of the most difficult tasks of preprocessing in traditional FEA. In modern FEA packages, an initial mesh may be automatically altered, during the solution process in order to minimize or reduce the error in the numerical solution. This is referred to as adaptive meshing.

If creating the mesh is considered a difficult task, then selecting and setting the solvers and obtaining a solution to the equations (which constitute the numerical model) in a reasonable computational time is an even more difficult task. The difficulty is associated with a variety of challenges.

This work demonstrates how a small number of well-defined experiments can be used to develop, calibrate, and validate solver technologies used for simulating the impact of projectiles on armor systems.

New concepts and models can be developed and easily tested with the help of modern hydrocodes. The initial design approach of the units and systems has to be as safe and optimal as possible. Therefore, most design concepts are analyzed on the computer.

The gained experience is of prime importance for the development of modern armor. By applying the numerical model, a large number of potential armor schemes can be evaluated and the understanding of the interaction between different materials under ballistic impact can be improved.

The most important steps during an FE analysis are the evaluation and interpretation of the outcomes followed by suitable modifications of the model. For that reason, ballistic trials are necessary to validate the simulation results.

They are designed to obtain information about

- the velocity and trajectory of the projectile prior to the impact,
- changes in configuration of the projectile and target due to the impact,
- masses, velocities, and trajectories of fragments generated by the impact process.

The combined use of computations, experiments and high-strain-rate material characterization has, in many cases, supplemented the data achievable by experiments alone at considerable savings in both cost and engineering manhours.

REFERENCES

[1] A. Ramezani and H. Rothe, “Simulation Methodologies for the Numerical Analysis of High-Speed Dynamics,” The Seventh International Conference on Advances in System Simulation (SIMUL 2015) IARIA, Nov. 2015, pp. 59-66, ISBN 978-1-61208-442-8

[2] J. Zukas, “Introduction to Hydrocodes,” Elsevier Science, February 2004.

[3] A. M. S. Hamouda and M. S. J. Hashmi, “Modelling the impact and penetration events of modern engineering materials: Characteristics of computer codes and material models,” Journal of Materials Processing Technology, vol. 56, Jan. 1996, pp. 847–862.

[4] D. J. Benson, “Computational methods in Lagrangian and Eulerian hydrocodes,” Computer Methods in Applied Mechanics and Engineering, vol. 99, Sep. 1992, pp. 235–394, doi: 10.1016/0045-7825(92)90042-I.

[5] M. Oevermann, S. Gerber, and F. Behrendt, “Euler-Lagrange/DEM simulation of wood gasification in a bubbling fluidized bed reactor,” Particuology, vol. 7, Aug. 2009, pp. 307-316, doi: 10.1016/j.partic.2009.04.004.

[6] D. L. Hicks and L. M. Liebrock, “SPH hydrocodes can be stabilized with shape-shifting,” Computers & Mathematics with Applications, vol. 38, Sep. 1999, pp. 1-16, doi: 10.1016/S0898-1221(99)00210-2.

[7] X. Quan, N. K. Birnbaum, M. S. Cowler, and B. I. Gerber, “Numerical Simulations of Structural Deformation under Shock and Impact Loads using a Coupled Multi-Solver Approach,” 5th Asia-Pacific Conference on Shock and Impact Loads on Structures, Hunan, China, Nov. 2003, pp. 152-161.

[8] N. V. Bermeo, M. G. Mendoza, and A. G. Castro, “Semantic Representation of CAD Models Based on the IGES Standard,” Computer Science, vol. 8265, Dec. 2001, pp. 157-168, doi: 10.1007/ 978-3-642-45114-0_13.

[9] G. S. Collins, “An Introduction to Hydrocode Modeling,” Applied Modelling and Computation Group, Imperial College London, August 2002, unpublished.

[10] R. F. Stellingwerf and C. A. Wingate, “Impact Modeling with Smooth Particle Hydrodynamics,” International Journal of Impact Engineering, vol. 14, Sep. 1993, pp. 707–718.

[11] ANSYS Inc. Available Solution Methods. [Online]. Available from: http://www.ansys.com/Products/Simulation+Technology/Structural+Analysis/Explicit+Dynamics/Features/Available+Solution+Methods [retrieved: August, 2015]

[12] P. Fröhlich, “FEM Application Basics,” Vieweg Verlag, September 2005.

[13] H. B. Woyand, “FEM with CATIA V5,” J. Schlembach Fachverlag, April 2007.

[14] L. E. Schwer, “Aluminum Plate Perforation: A Comparative Case Study using Lagrange with Erosion, Multi-Material ALE, and Smooth Particle Hydrodynamics,” 7th European LS-DYNA Conference, Salzburg, Austria, May. 2009.

[15] ANSYS Inc. “The Value of High-Performance Computing for Simulation,” [Online]. Available from: http://investors.ansys.com/~/media/Files/A/Ansys-IR/annual-reports/whitepapers/the-value-of-high-performance-computing-for-simulation.pdf [retrieved: November, 2016]

[16] G. S. Almasi, and G. Allan. “Highly parallel computing,” Benjamin-Cummings Publishing Co., Inc., 1988.

[17] H. El-Rewini, and A. Mostafa, “Advanced computer architecture and parallel processing,” Vol. 42, John Wiley & Sons, 2005.

[18] J. E. Savage, “Models of computation,” Exploring the Power of Computing, 1998.

[19] J. L. Hennessy, and D. A. Patterson, “Computer architecture: a quantitative approach,” Elsevier, 2011.

[20] G. M. Amdahl, “Validity of the single processor approach to achieving large scale computing capabilities,” Proceedings of the April 18-20, 1967, spring joint computer conference. ACM, 1967.

[21] J. L. Gustafson, “Reevaluating Amdahl's law,” Communications of the ACM 31.5: 532-533, 1988.

[22] G. E. Moore, “Cramming more components onto integrated circuits, Reprinted from Electronics, volume 38, number 8, April 19, 1965, pp. 114 ff.” IEEE Solid-State Circuits Newsletter 3.20:33-35, 2006.

[23] J. P. Dittmer, C. G. Jensen, M. Gottschalk, and T. Almy, “Mesh Optimization Using a Genetic Algorithm to Control Mesh Creation Parameters,” Computer-Aided Design & Applications, vol. 3, May 2006, pp. 731–740.