

Title: A Virtual Test Facility for Simulating Detonation- and Shock-induced Deformation and Fracture of Thin Flexible Shells

Authors: Ralf Deiterding (Oak Ridge National Laboratory)
Fehmi Cirak (University of Cambridge)
Sean P. Mauch (California Institute of Technology)
Daniel I. Meiron (California Institute of Technology)

Corresponding author: Dr. Ralf Deiterding
Oak Ridge National Laboratory
P.O. Box 2008 MS 6367
Oak Ridge, TN 37831-6367, USA
Telephone ++1-865-241-0782
Fax ++1-865-574-0680
E-mail: deiterdingr@ornl.gov

Running head: A Virtual Test Facility

Key words: Fluid-structure interaction, detonation model, thin-shells, large deformations, fracture, dynamic mesh adaptation, parallelization, water hammer

Abstract: The coupling of a dynamically adaptive Eulerian Cartesian detonation solver with hierarchical time step refinement to a Lagrangian thin-shell finite element solver with fracture and fragmentation capabilities is presented. The approach uses a level set function to implicitly represent arbitrarily evolving solid structures on the Cartesian mesh. The auxiliary algorithm used to efficiently transform the shell solver mesh on-the-fly into a distance function is sketched briefly. We detail the derivation of the employed engineering combustion model that eliminates the numerical stiffness otherwise inherent to detonation waves and describe our approach to modeling fracture. The thin-shell solver utilizes a subdivision finite element discretization and achieves element separation with interface edges and a cohesive law. For method validation and benchmarking, the simulation of the deformation of a circular thin copper plate under impulsive pressure loading is presented. As a realistic computational application, we consider a three-dimensional setup in which the passage of an ethylene-oxygen detonation wave induces large plastic deformations and rupture of a thin-walled tubular specimen made of aluminum. Special attention is paid to the verification of the hydrodynamic loading conditions. The computational fluid-structure interaction results are found to be in agreement with experimental observations.

1 Introduction

The Center for Simulation of Dynamic Response of Materials at the California Institute of Technology has developed a virtual test facility (VTF) for studying the three-dimensional dynamic response of solid materials subjected to strong shock and detonation waves propagating in fluids [1, 2]. The VTF targets highly coupled problems, such as the high rate deformation of metals due to the explosion of high-energetic materials or the rupture and fragmentation of brittle materials under shock wave impact. This application regime requires the coupled utilization of computational fluid dynamics (CFD) solvers for compressible hydrodynamics and computational solid dynamics (CSD) solvers for large plastic material deformations. CFD and CSD solvers both need to be time-accurate and have to consider all arising supersonic wave phenomena (shear and dilatation waves in the plastic solid, shock waves in the compressible fluid) correctly. Applicable numerical schemes are usually shock-capturing and time-explicit. Hence, we employ a straightforward temporal splitting technique for coupling in which CFD and CSD solver exchange data only at the interface between disjoint computational domains after consecutive time steps. For compressible fluids, stable solutions are obtained reliably with such a “weakly coupled” method, when the evolving interface geometry and velocities are imposed as boundary conditions on the CFD solver and the hydrodynamic pressure is used as force boundary condition acting on the solid exterior [3, 4, 5].

While a Lagrangian representation is most suitable to account numerically for large solid deformations, contact and fracture, shock-capturing methods for compressible flows are most easily formulated in an Eulerian frame of reference [1]. For fluid-structure interaction (FSI) problems with moderate structural motion it is possible to cast the fluid equations into a local arbitrary Lagrangian-Eulerian (ALE) frame of reference [6]. Such methods permit very accurate implementations of the fluid-solid coupling conditions [7], but face complex re-meshing and re-mapping problems, when the solid structure undergoes major topology evolutions. The need to re-mesh is also an inherent bottleneck in massively parallel simulations [5].

To ensure general applicability to FSI problems involving large deformations, fracture, fragmentation and contact, we have implemented a different approach in our FSI framework. In the VTF, the fluid solution is represented on a fixed Cartesian mesh and geometrically complex solid structures are considered with an embedded boundary approach, similar in spirit to the Immersed Boundary Method of Peskin [8]. Specific to the VTF is that scalar level set functions storing the distance information to the embedded surface are utilized to represent the geometry on the Cartesian mesh implicitly and a ghost-fluid-type approach is used to impose fluid boundary conditions [9, 10, 4, 11]. Inaccuracies in the boundary approximation inherent to Cartesian methods that omit the construction of boundary intersected cut-cells (see [12] for a comparison) are mitigated by dynamic block-structured adaptation of the fluid mesh [13].

In the present paper, we employ the VTF framework to simulate the deformation of thin-walled metallic structures under transient pressure and detonation wave loading. Our main application is the rupture of aluminum tubes due to the passage of gaseous detonations in ethylene-oxygen mixtures at equivalence ratio 1 ($\text{C}_2\text{H}_4 + 3\text{O}_2$) as investigated by Chao [14]. The initial gas pressure varies from $p_0 = 80$ kPa to $p_0 = 180$ kPa and the initial temperature is 295 K. In Sec. 2, we sketch the employed adaptive Cartesian finite volume CFD solver with level-set-based embedded boundary capability and detail in particular the choice of an appropriate detonation model as a prerequisite for engineering simulations of the full-sized problem. Section 3 describes the CSD solver that has been developed to enable FSI simulations of thin-walled (possibly fracturing) solid structures [15]. The solver is based on a Kirchhoff-Love-type thin-shell formulation in Lagrangian coordinates and achieves a consistent finite element discretization of the underlying energy functional even in the case of fracture by employing subdivision elements. In Sec. 4, we outline the highly efficient auxiliary algorithm based on geometric characteristic reconstruction and scan conversion that we have developed to transform an evolving triangulated surface mesh efficiently into a distance function. The fluid-structure coupling methodology and its implementation are sketched in Sec. 5. A non-reactive three-dimensional validation computation of a thin copper plate deformed by a strong pressure wave is discussed and benchmarked. In Sec. 6, we present a series of computations towards our main FSI application. After validating the loading conditions from the detonation model introduced in Sec. 2, we compare transient FSI results for detonation-driven large plastic deformations with experimentally obtained Schlieren images and finally discuss a preliminary result for the rupturing case.

2 Eulerian detonation solver

A detonation is a shock-induced combustion wave that internally consists of a discontinuous hydrodynamic shock followed by a smooth region of decaying combustion. The compression due to the shock rises the temperature above the ignition limit and the resulting energy release drives the shock itself forward.

2.1 Governing equations

An accepted model for detonation propagation in premixed gases with realistic chemistry are the inviscid Euler equations for multiple thermally perfect species with reactive source terms [16, 17]. These equations form a system of inhomogeneous hyperbolic conservation laws that reads

$$\begin{aligned}
 \partial_t \rho_i &+ \nabla \cdot (\rho_i \mathbf{u}) &= W_i \dot{\omega}_i, \\
 \partial_t (\rho \mathbf{u}) &+ \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p &= 0, \\
 \partial_t (\rho E) &+ \nabla \cdot ((\rho E + p) \mathbf{u}) &= 0,
 \end{aligned} \tag{1}$$

with $i = 1, \dots, K$. Herein, ρ_i denotes the partial density of the i th species and $\rho = \sum_{i=1}^K \rho_i$ is the total density. The ratios $Y_i = \rho_i / \rho$ are called mass fractions. We denote the velocity vector by \mathbf{u} and E is the

specific total energy. The hydrostatic pressure p is given as the *sum* of the partial pressures $p_i = \mathcal{R}T\rho_i/W_i$ with \mathcal{R} denoting the universal gas constant and W_i the molecular weight, respectively. The evaluation of the last equation requires the previous calculation of the temperature T by iteration (cf. [18]). The chemical production rate for each species is derived from a reaction mechanism of J chemical reactions as

$$\dot{\omega}_i = \sum_{j=1}^J (\nu_{ji}^r - \nu_{ji}^f) \left[k_j^f \prod_{l=1}^K \left(\frac{\rho_l}{W_l} \right)^{\nu_{jl}^f} - k_j^r \prod_{l=1}^K \left(\frac{\rho_l}{W_l} \right)^{\nu_{jl}^r} \right]. \quad (2)$$

Herein, $\nu_{ji}^{f/r}$ are stoichiometric coefficients and the reaction rates are usually calculated by the Arrhenius law $k_j^{f/r}(T) = A_j^{f/r} T^{\beta_j^{f/r}} \exp(-E_j^{f/r}/\mathcal{R}T)$. By assuming a stationary one-dimensional situation consisting of a shock and a succeeding reaction, the governing equations can be transformed into a system of ordinary differential equations depending only on the distance behind the shock front. Detailed accounts of the elaborate derivation can be found for instance in [16, 19, 18].

The one-dimensional detonation structure for our configuration for the case $p_0 = 100$ kPa and the experimentally measured averaged detonation velocity $D_{\text{Ex}} = 2291.7$ m/s is depicted in Fig. 1. Figure 1 has been produced with the GRI-3.0 reaction mechanism [20] that considers ~ 220 chemical reactions for ethylene-oxygen combustion. The left plot of Fig. 1 shows the mass fraction distributions of some of the 34 active species. It can be seen that the chemical depletion is mostly completed ~ 0.05 mm behind the detonation front, which corresponds to a time frame of only ~ 20 ns. Due to the large computational expense of detailed chemistry and its apparent stiffness, a model simplification is indispensable for engineering calculations.

2.2 Detonation modeling for engineering

A reasonable first step in detonation modeling for engineering is to reduce the detailed chemical reaction network to a simple exothermic chemical reaction $R \longrightarrow P$ with an energy release $q > 0$. Further, it is usually sufficient to assume that the species R and P are calorically perfect gases with the same constant adiabatic exponent γ . This simplification allows the expression of the hydrodynamic pressure directly in terms of the conserved quantities in the explicit equation of state $p = (\gamma - 1)(\rho E - \frac{1}{2}\rho \mathbf{u}^T \mathbf{u} - \rho Y_R q)$, cf. [16, 18]. A one-step reaction can be modeled with the forward reaction rate $k^f(T) = A \exp(-E_R/\mathcal{R}T)$ yielding the mass production rates

$$W_R \dot{\omega}_R = -A \rho_R \exp(-E_R/\mathcal{R}T), \quad W_P \dot{\omega}_P = -W_R \dot{\omega}_R, \quad (3)$$

but this approach has the disadvantage that the reaction zone still needs to be well-resolved in numerical simulations. For the specific setup considered here, we utilize the constant volume burn model suggested by Mader [21] instead. This model neglects the internal detonation structure completely, but ensures the right propagation speed and the correct state in chemical equilibrium at all grid resolutions. The burn model is

intended to be applied together with the fractional step method that numerically decouples chemical reaction and hydrodynamic transport. First, the *homogeneous* system (1) is advanced at a full time step, then the partial densities ρ_R , ρ_P , pressure p , and total energy E are modified locally in each cell; the total density ρ and the velocity vector \mathbf{u} remain unaltered. The algorithm for the detonation model reads:

$$\begin{aligned}
V &:= \rho^{-1}, \quad V_0 := \rho_0^{-1}, \quad V_{\text{CJ}} := \rho_{\text{CJ}} \\
Y'_R &:= 1 - (V - V_0)/(V_{\text{CJ}} - V_0) \\
&\text{if } 0 \leq Y'_R \leq 1 \text{ and } Y'_R > 10^{-8} \\
&\quad \text{if } Y'_R < Y'_R \text{ and } Y'_R < 0.9 \text{ then } Y'_R := 0 \\
&\quad \text{if } Y'_R < 0.99 \text{ then } p' := (1 - Y'_R)p_{\text{CJ}} \text{ else } p' := p \\
\rho_R &:= Y'_R \rho, \quad \rho_P := (1 - Y'_R) \rho \\
E &:= p' / (\rho(\gamma - 1)) + Y'_R q + \frac{1}{2} \mathbf{u}^T \mathbf{u}
\end{aligned}$$

In the latter, the index 0 indicates the unreacted state (assumed to be constant), while CJ refers to the values in chemical equilibrium that can be calculated in advance following Chapman-Jouguet theory [16, 18] for a given detonation velocity.

2.3 Finite volume scheme with thin-walled embedded structures

As shock-capturing finite volume upwind scheme, we utilize a straightforward extension of the flux-vector splitting method by Van Leer (cf. [18]). Second-order accuracy in smooth solution regions is achieved with the MUSCL-Hancock variable extrapolation technique [22]. Geometrically complex moving boundaries are incorporated into the upwind scheme by using some of the finite volume cells as ghost cells for enforcing immersed moving wall boundary conditions [9]. The boundary geometry is mapped onto the Cartesian mesh by employing a scalar level set function ϕ that stores the unsigned distance to the boundary surface and allows the efficient evaluation of the boundary outer normal in every mesh point as $\mathbf{n} = -\nabla\phi/|\nabla\phi|$. A cell is considered to be a valid fluid cell, if the distance at the cell *midpoint* satisfies the condition $\phi > h/2$ and as an exterior ghost cell otherwise. The mesh received from the shell solver corresponds to a two-dimensional manifold surface mesh (cf. Sec. 3) and the utilization of condition $\phi > h/2$ is a straightforward, unambiguous solution to achieve the mandatory thickening of this mesh by the shell thickness h . The contour line $\phi = h/2$ effectively represents the embedded boundary for the fluid solver (depicted as dotted line around shell elements in Fig. 2). The hydrodynamic load on each shell element is evaluated as the difference between the approximated pressure values at $\phi = h/2$ in the positive and negative direction of each element's normal, i.e. $p^F := p^+ - p^-$.

For the governing equations (1), the boundary condition at a rigid wall moving with velocity \mathbf{w} is $\mathbf{u} \cdot \mathbf{n} = \mathbf{w} \cdot \mathbf{n}$. Enforcing this condition with ghost cells, in which the discrete values are located at the cell centers, requires the mirroring of the primitive values ρ_i , \mathbf{u} , p across the embedded boundary. The normal velocity in the ghost cells is set to $(2\mathbf{w} \cdot \mathbf{n} - \mathbf{u} \cdot \mathbf{n})\mathbf{n}$, while the mirrored tangential velocity remains unmodified.

Mirrored values are constructed by calculating spatially interpolated values at the point $\tilde{\mathbf{x}} = \mathbf{x} + 2\phi\mathbf{n}$ from neighboring interior cells. We employ a dimension-wise linear interpolation for this operation, but it has to be emphasized that directly near the boundary the number of interpolants needs to be decreased to ensure the monotonicity of the numerical solution. This property is essential in simulating hyperbolic problems with discontinuities, like detonation waves. Figure 2 also highlights the necessary reduction of the interpolation stencil for some exemplary cases. The interpolation locations are indicated by the origins of the arrows normal to the contour line that defines the embedded boundary. After the application of the numerical scheme, cells that have been used to impose internal boundary conditions are set to the entire state vector of the nearest cell in the fluid interior. This operation ensures proper values in case such a cell becomes a regular interior cell in the next step due to boundary movement. The consideration of \mathbf{w} in the ghost cells guarantees that the embedded boundary propagates at most one cell in every time step.

Note that the described technique does not require a modification of the numerical stencil itself and is therefore generically applicable, but causes a diffusion of the boundary location throughout the method and results in an overall non-conservative scheme. We reduce such errors and the unavoidable staircase approximation of the boundary with this approach effectively by using dynamic mesh adaptation to refine the Cartesian mesh appropriately along the boundary.

2.4 Structured adaptive mesh refinement

In order to supply a fine local temporal and spatial resolution efficiently, we use the block-structured adaptive mesh refinement (SAMR) method of Berger and Colella [23]. Characteristic for the SAMR method is that a specific finite volume method is technically not implemented in a cell-based fashion, but rather in a routine operating on equidistant subgrids. The subgrids become computationally decoupled during one update cycle through the use of ghost or halo cells. Starting from the base mesh on level 0, the time step size and all spatial mesh widths on level $l > 0$ are r_l -times finer than on level $l - 1$ and a time-explicit finite volume scheme will (in principle) remain stable on all levels of the recursively nested refinement hierarchy. Ghost cell values at coarse-fine interfaces are constructed by interpolating coarse level data, which mandates a recursive order of update (cf. Sec. 5.1). SAMR in the VTF is provided generically by the AMROC (Adaptive Mesh Refinement in Object-oriented C++) framework [24] that can be used on all parallel systems that provide the MPI library.

3 Lagrangian thin-shell solver

The Kirchhoff-Love thin-shell model applied here has been discretized with smooth subdivision finite elements, as previously introduced in [25, 26]. Notably, the underlying kinematic assumptions allow for finite

strains, displacements and rotations. The subdivision shell elements have also been extended to the range of applications that involve fracture and fragmentation [27]. Thereby, fracture initiation and propagation is considered as a progressive failure phenomenon in which the separation of the crack flanks is modeled with a cohesive law. In the present implementation, cohesive interface elements are inserted at all inter-element edges and constrain the opening of the crack flanks to the deformation of the shell mid-surface and its normal.

3.1 Governing equations in weak form

To kinematically describe a possibly fractured thin-shell as sketched in Fig. 3, we first consider the shell in its undeformed configuration \bar{V} . The position vector $\bar{\varphi}$ of a material point on the undeformed shell body is assumed to be

$$\bar{\varphi} = \bar{\mathbf{x}} + \theta^3 \bar{\mathbf{n}} \quad (4)$$

with the uniform thickness \bar{h} and $-\bar{h}/2 \leq \theta^3 \leq \bar{h}/2$. The position vector of the shell mid-surface is denoted by $\bar{\mathbf{x}}$ and its out-of-surface unit normal by $\bar{\mathbf{n}}$. In other words, the shell mid-surface represents a two-dimensional manifold in \mathbb{R}^3 . The deformation mapping φ maps the shell body into the deformed configuration V

$$\varphi = \mathbf{x} + \theta^3 \lambda \mathbf{n} \quad (5)$$

where \mathbf{x} and \mathbf{n} are the deformed mid-surface and its normal. The thickness stretch parameter λ is the ratio of the deformed shell thickness h to the reference thickness \bar{h} . In the presence of a crack, the deformation is discontinuous across the crack and has a jump, i.e.

$$[[\varphi]] = \varphi^+ - \varphi^- = [[\mathbf{x}]] + \theta^3 [[\mathbf{n}]], \quad (6)$$

where the superscripts $+$ and $-$ refer to the opposing crack flanks. Further, the first term describes the discontinuity of the deformation of the shell mid-surface, and the second term the discontinuity in the shell out-of-surface normal. The discontinuities in the deformations can also be interpreted as the opening displacement of the crack. Further, note that the Kirchhoff-Love constraint is satisfied, i.e. $\mathbf{x} \cdot \mathbf{n} = 0$, on both sides of the crack.

A standard semi-inverse approach is followed for obtaining the shell equilibrium equations in weak form (see e.g. [28]). To this end, the assumed reduced kinematic equations for the shell body (Equations (4) and (6)) are introduced into the conventional virtual work expression for the three-dimensional body. As previously mentioned, we consider fracture as a gradual separation phenomenon, resisted by cohesive tractions. Consequently, the internal virtual work expression contains the virtual work of the cohesive interface ($\delta\Pi_{C,\text{int}}$) in addition to the virtual work of the bulk material ($\delta\Pi_{S,\text{int}}$)

$$\delta\Pi_{S,\text{int}} + \delta\Pi_{C,\text{int}} - \delta\Pi_{\text{ext}} = 0 \quad (7)$$

with the external virtual work $\delta\Pi_{\text{ext}}$ and

$$\delta\Pi_{S,\text{int}} = \int_{\bar{\Omega}} \int_{-\bar{h}/2}^{\bar{h}/2} \mathbf{P} : \delta\mathbf{F} \mu d\theta^3 d\bar{\Omega}, \quad \delta\Pi_{C,\text{int}} = \int_{\bar{\Gamma}_C} \int_{-\bar{h}/2}^{\bar{h}/2} \mathbf{T} \cdot \llbracket \boldsymbol{\varphi} \rrbracket \mu d\theta^3 d\bar{\Gamma}_C,$$

where \mathbf{P} is the first Piola-Kirchhoff stress tensor, \mathbf{T} the related traction vector at the cohesive surface, and \mathbf{F} the deformation gradient. The virtual work expression for the bulk material is integrated over the undeformed shell mid-surface $\bar{\Omega}$ and for the cohesive interface over the crack path $\bar{\Gamma}_C$. The scalar factor μ accounts for the curvature of the shell in the volume computation [26].

3.2 Subdivision thin-shell elements

Next, we briefly outline the discretization of the governing equation (7) firstly for the non-fractured case. A detailed presentation of the used subdivision finite element discretization technique can be found in [25] and [26]. In this approach, the reference ($\bar{\mathbf{x}}$) and deformed (\mathbf{x}) shell surfaces are approximated using smooth subdivision surfaces belonging to the Sobolev space H^2 with square-integrable curvatures. The subdivision interpolation within one element is accomplished with shape functions, which have support on the element as well as on the one-ring of neighboring elements (Fig. 4(a)). The overlapping local subdivision interpolants, each defined over one patch, together lead to a global interpolant with square-integrable curvatures. Importantly, smoothness is achieved without introducing nodal rotations as degrees of freedom. The absence of nodal rotations is particularly appealing in the presence of finite rotations.

In the presence of fracture, the smoothness and/or continuity of the interpolation has to be relaxed and the subdivision interpolant needs to be modified (see [27] for details). The topological changes necessary to the non-local subdivision functions and the underlying control mesh in order to describe the dynamic propagation of a single crack are complicated. Therefore, we chose to pre-fracture the element patches, such that each patch possesses its own nodes and acts independently for the purpose of interpolation (see Fig. 4(b)). Prior to crack nucleation, the coupling of the elements is enforced by applying stiff elastic cohesive interfaces at all edges. Once fracture nucleates along an element edge, the element patches on both sides of the cracked edge interact through cohesive tractions. The cohesive tractions are self-balanced internal forces derived from a cohesive fracture model.

3.3 Constitutive models for the shell

An irreversible cohesive constitutive model as proposed by Ortiz et al. [29] is used for modeling the cracks. Thereby, the opening displacement $\llbracket \boldsymbol{\varphi} \rrbracket$ plays the role of a deformation measure while the traction \mathbf{T} is the conjugate stress measure. Further, a scalar effective opening displacement is defined

$$\delta = \sqrt{\beta^2 |\boldsymbol{\delta}_t|^2 + |\boldsymbol{\delta}_n|^2}, \quad (8)$$

where δ_t and δ_n are the tangential and normal displacement components of $[[\varphi]]$ with respect to the crack surface. The parameter β assigns different weights to the tangential and normal opening displacements. The cohesive tractions \mathbf{T} are given by

$$\mathbf{T} = \frac{t}{\delta}(\beta^2 \delta_t + \delta_n). \quad (9)$$

The scalar effective traction t is computed from a cohesive law as shown in Fig. 5. In addition to the parameter β , the model parameters are the maximal tensile stress σ_c and the critical opening displacement δ_c . The following relationship between the cohesive law and the critical fracture energy G_c exists:

$$G_c = \int_0^\infty t d\delta \quad (10)$$

which can be used for determining δ_c . For further details see e.g. [29].

The inelastic behavior of the bulk material, i.e. the relation between \mathbf{P} and \mathbf{F} , is described with a conventional J_2 viscoplasticity model with isotropic power-law hardening as described in [30]. The power-law hardening for the flow stress g has the form

$$g(\epsilon^p) = \sigma_y \left(1 + \frac{\epsilon^p}{\epsilon_0^p}\right)^{1/n}, \quad (11)$$

where σ_y is the initial yield stress, ϵ^p and ϵ_0^p are the total and the reference plastic strains, respectively, and $1/n$ is the hardening exponent. The rate-dependent behavior is described in terms of the effective von Mises stress σ_{eff} with a power viscosity law and constant rate sensitivity

$$\sigma_{\text{eff}} = g(\epsilon^p) \left(1 + \frac{\dot{\epsilon}^p}{\dot{\epsilon}_0^p}\right)^{1/m}, \quad (12)$$

where $\dot{\epsilon}_0^p$ is the reference plastic strain rate and $1/m$ the strain rate sensitivity exponent.

The thin-shell typical plane stress condition is enforced with a local Newton-Raphson iteration at each quadrature point [31]. Thereby, the thickness stretch parameter λ (Eq. 5) is the unknown variable in the iteration.

4 Efficient level set evaluation

In Sec. 2, we have sketched the concept of employing a distance function to represent a complex embedded boundary on a Cartesian mesh. While distance functions are easily prescribed for single elementary geometric objects, their evaluation can be cumbersome for complex shapes. In coupled Eulerian-Lagrangian simulations, this complex shape is defined by the deforming shell surface mesh.

One can efficiently compute the distance on a grid by solving the eikonal equation with the method of characteristics and utilizing polyhedron scan conversion [32]. For a given grid point, the relevant closest point on the triangular mesh lies on one of the primitives (faces, edges and vertices) that comprise the surface.

The characteristics emanating from each of these primitives form polyhedral shapes. Such a *characteristic polyhedron* contains all of the points which are possibly closest to its corresponding face, edge or vertex. The closest points to a triangle face must lie within a triangular prism defined by the face and its normal; the closest points to an edge lie in a cylindrical wedge defined by the line segment and the normals to the two incident faces (see Fig. 6 for face (a) and edge (b) polyhedra for a particular example). Analogously, polygonal pyramids emanating from the vertices are also possible (not shown). We then determine the grid points that lie inside a characteristic polyhedron with polyhedron scan conversion. The polyhedron is first sliced along each sheet of the grid lattice to produce polygons, cf. Fig. 7. Simple geometric formulas are finally used to calculate the distance once a polyhedron has been assigned uniquely to each grid point.

By utilizing the outlined techniques, and evaluating the distance exactly only within a small distance around the surface, a highly efficient algorithm can be formulated that has linear computational complexity both in the number of Cartesian mesh points and the surface triangles [32, 2].

5 Fluid-structure coupling

The explicit fluid and solid solvers are weakly coupled by successively applying appropriate boundary conditions in a time-operator splitting technique. In the case of inviscid flows, the compatibility conditions are simply the continuity of the velocity component normal to the embedded boundary u_n in solid (S) and fluid (F), i.e. $u_n^S = u_n^F$, and the continuity of the normal component of the solid's Cauchy traction vector, $p^S = (\boldsymbol{\sigma}\mathbf{n})\mathbf{n}$ with $\boldsymbol{\sigma} = 1/\det(\mathbf{F})\mathbf{F}\mathbf{P}$, and the hydrodynamic pressure p^F , i.e. $p^S = p^F$. We use the following update algorithm to implement these coupling conditions numerically:

```

update  $\phi(t)$ 
 $\mathbf{w}_F^{+/-} := \mathbf{u}^S(t)$ 
update_fluid( $\Delta t$ )
 $p^S := p^F(t + \Delta t)$ 
update_solid( $\Delta t$ )
 $t := t + \Delta t$ 

```

After evaluating the distance function ϕ for the currently available shell surface mesh, the embedded wall boundary velocities for the fluid solver are set to the solid velocities in the nearest shell element midplane. The same velocity \mathbf{w} is enforced in the fluid on upper (+) and lower (-) side of each element. After setting embedded rigid wall boundary conditions and the fluid update, a new hydrodynamic pressure load $p^F := p^+ - p^-$ on each shell element (compare Fig. 2) is derived by evaluating p^\pm with the linear interpolation / extrapolation operation already sketched in Sec. 2.3. With these new boundary conditions, the cycle is completed by advancing the solid by Δt , which in practice is typically done by taking multiple, smaller time steps in the solid solver to effectively accommodate the more restrictive stability condition in the solid.

5.1 Application of SAMR in the fluid solver

While the implementation of a loosely coupled FSI method is straightforward with conventional solvers with consecutive time update, the utilization of the recursive SAMR method with hierarchical time step refinement in the fluid is non-apparent. In the VTF, we treat the fluid-solid interface as a discontinuity that is a-priori refined at least up to a coupling level l_c . The resolution at level l_c has to be sufficiently fine to ensure an accurate wave transmission between fluid and structure, but will often not be the highest level of refinement. To incorporate the fluid-structure data exchange into the recursive SAMR algorithm it has to be ensured that the updated mesh positions and nodal velocities are received *before* a regridding of the coupling level l_c is initiated and that the hydrodynamic pressure loadings on the interface are evaluated *after* the highest available refinement level has reached the same discrete time as the updated level l_c . We visualize the data exchange between solid and SAMR fluid solver in Fig. 8 for an exemplary SAMR hierarchy with two additional levels with $r_{1,2} = 2$. Figure 8 pictures the recursion in the SAMR method by numbering the fluid update steps (F) according to the order determined by the SAMR method. The order of the solid update steps (S) on the other hand is strictly linear. The red arrows correspond to the sending of the interface pressures p^F from fluid to solid at the end of each time step on level l_c . The blue arrows visualize the sending of the interface mesh and its nodal velocities \mathbf{u}^S after each solid update. The modification of refinement meshes is indicated in Fig. 8 by the gray arrows; the initiating base level, that remains fixed throughout the regridding operation, is indicated by the gray circles.

5.2 Software implementation

The fluid-structure coupling software “Virtual Test Facility” is a collection of C++ classes for implementing high-speed FSI problems on distributed memory machines. At present, only the loose coupling of time-explicit solvers is supported. The design follows a classical framework approach in which the instantiation of the main objects is usually done in a short generic main program and customization is achieved through subclass derivation. Figure 9 depicts the relationships in Unified Modeling Language (UML) notation of the main classes for a typical FSI application. The solver components CoupledSAMRSolver and CoupledShellSolver are interface layers to the stand-alone parallel SAMR-CFD and CSD solvers, described in Sec. 2 and Sec. 3, respectively. The SAMR-CFD solver with level-set-based boundary representation LevelSetSAMRSolver uses the Closest Point Transform algorithm, sketched in Sec. 4, for distance function evaluation.

In our current implementation, CFD and CSD solver are parallelized separately using independent rigorous domain decomposition methods. In order to facilitate an efficient communication of the distributed fluid-shell boundary information we have implemented a non-blocking high-level communication module InterSolverCommunication that determines the processor-to-processor communication patterns by intersecting

Cartesian bounding boxes enclosing the local domains. Details on this communication library and a more detailed algorithmic description are given in [2].

5.3 Validation

In order to validate the presented fluid-structure coupling technique, but also to demonstrate the versatility of approach and implementation, we simulate the plastic deformation of a thin circular copper plate (0.25 mm thickness) under pressure wave impact. The setup under consideration has been developed by Deshpande et al. and uses a water shock tube of 1.3 m length and 32 mm inner radius [33]. The fluid is modeled with the homogeneous Euler equations for a polytropic single species and the “stiffened” gas equation of state

$$p = (\gamma - 1) \left(\rho E - \frac{1}{2} \rho \mathbf{u}^T \mathbf{u} \right) - \gamma p_\infty \quad (13)$$

with parameters $\gamma = 7.415$ and $p_\infty = 296.2$ MPa. Cavitation is roughly considered through an internal energy correction ensuring $p \geq 0$ after every fluid time step. The pressure loading is created through the motion of a piston with mass per unit area $\bar{m} = 74.1$ kg/m² at the upper shock tube end. A separate level set function is used to model the piston boundary in contact with the fluid. The piston boundary is initially positioned at $x = 1.3$ m and moves with velocity $v_0 = -22.94$ m/s. Level set position and velocity $v(t)$ are updated according to the law of motion $\bar{m}\dot{v} = -(\bar{p} - p_A)$, where $\bar{p}(t)$ denotes the area averaged hydrodynamic pressure and p_A is the atmospheric pressure. The shape of the simulated incoming pressure wave, cf. Fig. 10, corresponds well to analytic predictions in [33].

Since no rupturing of the plate specimen occurs, we simulate only the shock tube interior and use $p^+ \equiv p_A$ in the evaluation of p^F (cf. Sec. 2.3). To ensure the correct boundary conditions throughout the whole simulation, the three-dimensional fluid domain covers with $[-0.05 \text{ m}, 1.35 \text{ m}] \times [-0.04 \text{ m}, 0.04 \text{ m}] \times [-0.04 \text{ m}, 0.04 \text{ m}]$ the entire shock tube. For efficiency, flow field and piston boundary conditions are taken from a one-dimensional computation $t = 0.82$ ms after piston impact when the head of the pressure wave is close to $x = 0.05$ m. The fluid domain is discretized with an SAMR base mesh of $350 \times 20 \times 20$ cells and uses two additional levels uniformly refined by the factors $r_{1,2} = 2$. Full refinement of the fluid mesh is ensured in the plate vicinity and for the incoming pressure wave. The FSI coupling level is set to $l_c = 2$. The mesh for the shell solver mesh uses 8896 triangles and is constrained in the normal direction for $r \geq 32$ mm and fully fixed for $r \geq 41$ mm. The material parameters for the J_2 plasticity model are given in Table 1. It is assumed that the annealed copper is strain-rate insensitive and strain-softening effects need not be considered.

Figure 10 shows pressure traces from the FSI simulation in the tube middle axis. The impact of the pressure wave on the plate at $t \approx 0.03$ ms and cavitation immediately after can be clearly inferred. The fluid-structure interaction is separated into two phases. During the first 0.2 ms, the plate deformation occurs

with constant velocity as the cavitating water does not transmit major forces onto the plate. Afterwards, the deformation is nonlinear and driven by a pressure of ~ 4 MPa that declines from $t \approx 0.76$ ms on. The displacement remains basically constant from this point until the end of the simulation at $t_e = 1.0$ ms. A visual comparison of the finally deformed solid mesh in the FSI simulation with a photograph of a target plate is given in Fig. 11. The agreement is apparently very good. The obtained maximum deflection of 14.4 mm is also in good agreement with Qui et al.’s [34] analytic estimate of 16.1 mm (Equation 21a in [34]) that is necessarily larger since it has been derived under the assumptions of ideally plastic material.

The simulation was run on 8 nodes of a Intel-3.4 GHz-Xeon dual processor system (12 fluid and 4 solid processes) and required ~ 130 h CPU (8.1 h wall clock). 2000 coupled time steps were calculated with fixed step size to t_e with 5 solid solver sub-steps at the coupling level (cf. Sec. 5.1). To quantify the parallel performance of our current MPI-based implementation, a scalability study has been performed for the first 200 coupled time steps. Further, the test was re-run with a uniform fluid mesh of $1400 \times 80 \times 80$ cells. While 8.96 M finite volume cells are necessary in the uniform case, the adaptive simulation uses only ~ 1.2 M cells on average. The right graphic of Fig. 12 shows a typical snapshot of the computational setup (showing only part of the fluid domain) that displays the regions of different refinement in gray tones. The left graphic of Fig. 12 compares the average wall clock times for a single coupled time step. A decrease in parallel efficiency due to a relative increase in costs for parallel data synchronization can be clearly inferred, but since the test series was run on a system with low-bandwidth interconnect, the timings provide most likely a *lower* bound for the parallel performance of typical VTF applications. The benefit from using dynamic mesh adaptation in the fluid is apparent.

6 Computational results

In this section, we present our efforts in carrying out full-scale three-dimensional fluid-structure interaction simulations for the configuration by Chao [14]. The setup consists of a detonation tube of 1.52 m length to which thin-walled aluminum (Al6061-T6) test specimen are attached. The specimen have a length from 45.7 cm to 89.6 cm, an inner radius of 1.975 cm, and a wall thickness of 0.89 mm. While the lower end of the device is closed, a thin diaphragm seals the upper end. The entire apparatus is filled with a perfectly stirred ethylene-oxygen mixture at the conditions sketched already in Sec. 1. Since the lower end of the detonation tube is closed, a rarefaction wave occurs immediately behind the detonation, which mandatorily needs to be considered in accurate numerical simulations. In all computations, we utilize a constant adiabatic mixture coefficient of $\gamma = 1.24$, which is a good approximation to the value in chemical equilibrium and a reasonable compromise between the constant value behind the rarefaction wave of ~ 1.12 and the value 1.4 in the air

surrounding tube and specimen.

6.1 Verification of detonation model and loading conditions

In order to ensure the correct function of the engineering detonation model described in Sec. 2.2, we carry out one-dimensional detonation simulations for an initial pressure of $p_0 = 100$ kPa. The heat release is set to $q = 4.70408$ MJ/kg leading to a detonation velocity of 2291.7 m/s. We use a one-dimensional setup with a domain length of 2.15 m encompassing the detonation tube and the longest specimen, reflective wall boundary conditions at the lower end and zero gradient outflow conditions at the upper domain boundary. A base mesh of 1148 cells plus one additional level of dynamic refinement with factor $r_1 = 4$ is employed. The refinement criteria are scaled gradients of total density, pressure and mass fraction Y_R . Figure 13 compares pressure distributions from this computation and an equivalent simulation that utilizes the one-step Arrhenius reaction (3) instead. The Arrhenius parameters are set to $E_R = 25,000$ J/mol and $A = 2 \cdot 10^7$ s⁻¹ to match the length scales found in the detailed chemistry detonation structure analysis of Fig.1 approximately. The verification calculation uses an SAMR base mesh of 4000 cells and three additional levels with identical refinement factors $r_{1,2,3} = 4$ to achieve a proper resolution of the reaction zone and required several hours CPU compared to only seconds in the previous case. Figure 13 exhibits clearly that the internal detonation structure, with a short-term duration of less than 100 nanoseconds, has negligible influence on the rarefaction wave following immediately behind. Only the enlargement in Fig. 13 reveals the internal detonation structure. Since our succeeding simulations involve fluid-structure interaction times of several hundred microseconds, it is physically justified to employ the CV burn model in the following. This conclusion is also supported by experimentally measured pressure traces. Figure 14 shows a comparison of measured pressure evolutions at the locations $x = 0.38$ m (transducer 1), $x = 0.78$ m, and $x = 1.18$ m (from left to right in Fig. 14) and those derived from the CV burn model simulation (time origins in both traces adjusted to $t = 0$ when the detonation front reaches transducer 1). The agreement is very good considering the natural fluctuations in experimental measurements.

6.2 FSI simulation of detonation-driven venting

As a verification test towards rupture that involves large plastic material deformations, an experiment has been conducted in which an “H” shape pattern is cut close to the middle into a specimen of 0.896 m. Each cut has a length of 25 mm. The combustible mixture is the same as in Sec. 6.1. When the detonation wave passes the pre-flawed region, the two flaps open up and the high pressure in the Taylor wave causes a venting of the combustion products into the air. To allow for an undisturbed leakage we use a relative large computational domain of $[-0.920 \text{ m}, 0.896 \text{ m}] \times [-0.0375 \text{ m}, 0.5625 \text{ m}] \times [-0.390 \text{ m}, 0.390 \text{ m}]$ and an SAMR

base mesh of $242 \times 80 \times 104$ cells with 3 additional level and refinement factors 2, 2, and 4. The flow field is initialized with the data from the previous 1d simulation taken at the moment when the detonation enters the specimen and shifted by $x = -1.52$ m. Additional to the refinement criteria in Sec. 6.1, that capture the detonation front, the walls of the specimen are always fully refined enabling an offset parameter of $h = 0.81$ mm. An exemplary snapshot of the evolving mesh is depicted in Fig. 15. The figure shows Schlieren of the fluid density on the three refinement levels displayed in different gray tones and highlights the enormous gain from dynamic mesh adaptation. An equivalent unigrid CFD calculation would require $> 7.9 \cdot 10^9$ cells, but the SAMR computation uses only $\sim 4.0 \cdot 10^7$ cells on average. The CSD sub-problem employs a triangular input mesh of 17,056 elements. A J_2 plasticity model for aluminum with power-law hardening is applied as bulk material model [30]. The parameters given in Table 2 have been adapted from the Johnson-Cook material parameters reported by Lesuer et al. [35].

The computation ran on 72 Opteron-2.2 GHz processors connected with Infiniband network for about 60 h wall clock time (~ 4300 h CPU) to a final time of $t_e = 460 \mu\text{s}$. In Fig. 16, a series of Schlieren photographs are compared to corresponding simulated images at a nearby time. The computational graphics display Schlieren of the fluid density in the plane perpendicular to the z -axis together with a side view of the deforming solid mesh. The time origin is set to the moment when the detonation passes the middle of the longitudinal slot. The agreement in flow evolution and solid deformation is quite good confirming the correct function of the fluid-structure coupling methodology and the appropriateness of the chosen computational setup.

6.3 FSI simulation of detonation-driven fracture

Finally, we present one exemplary fluid-structure interaction computation that considers the rupture of the test specimen. The initial pressure is $p_0 = 180$ kPa and the specimen has a length of 0.457 m. To ensure a reproducible fracture pattern, the specimen has a central longitudinal notch of 63.2 mm parallel to the middle axis, which is modeled as an initial crack in the computations. The material model for the cohesive interface elements is a linearly decreasing envelope with a plane stress fracture toughness $K_{IC} = 30 \text{ MPa}\sqrt{m}$ [27]. In accordance with Li et al.'s [36] numerical computations of thin-sheet ductile fracture, the crack initiation stress is chosen to be $\sigma_c = 2\sigma_y$, where σ_y is the yield stress of the bulk material.

Figure 17 visualizes the results for a shell mesh of 8665 elements and a uniform Cartesian fluid mesh of $725 \times 40 \times 40$ cells that required ~ 900 h CPU and ran for about 16.5 h wall clock time on 27 nodes of a Pentium-4-2.4 GHz dual processor system (21 fluid and 33 solid processes). 1300 coupled time steps with fixed step size to a final time of $t_e = 260 \mu\text{s}$ have been calculated (20 solid solver sub-steps in each fluid time step). The left graphic of Fig. 17 shows the beginning of the crack opening $\sim 150 \mu\text{s}$ after the detonation has passed the initial crack. The right snapshot shows the rupture at the final time $260 \mu\text{s}$. The venting of high

pressurized reacted gas from the opening slit and the cracking of the material are clearly visible. It is worth pointing out that during this simulation, the dynamic level set evaluation with the algorithm sketched in Sec. 4 (C++) and the update with the core Cartesian finite volume scheme (Fortran 77) have about the same accumulated computational costs on each fluid processor. This result confirms that our approach utilizing distance functions for implicit geometry representation is suitable for computing even complex FSI problems with large deformations and evolutions in the mesh topology with high computational efficiency.

7 Conclusions

A loosely coupled parallel level-set-based fluid-structure coupling methodology for the time-accurate simulation of thin flexible shells dynamically responding to detonation and shock waves has been described. The approach has been demonstrated to handle arbitrarily evolving thin solid structures surrounded by fluid without problems. Reasonable agreement with experimental results has been achieved for large plastic deformations induced by pressure waves in water and by ethylene-oxygen detonation waves. In particular, we have detailed the detonation modeling mandatory for realistic engineering computations and the verification of the initially one-dimensional hydrodynamic loading conditions. As enabling components for high computational efficiency, we have highlighted dynamic mesh adaptation in the fluid sub-solver and an effective distance function evaluation algorithm. The integrated implementation of the solvers and algorithms described in this paper within the software framework “Virtual Test Facility” is freely available for research purposes (cf. <http://www.cacr.caltech.edu/asc>).

Acknowledgements

This work has been supported by the ASC program of the Department of Energy under subcontract no. B341492 of DoE contract W-7405-ENG-48 and was carried out largely when the first two authors were at the ASC Center for the Dynamic Response of Materials at the California Institute of Technology. R. D. also received support by the Mathematical, Information, and Computational Sciences Division; Office of Advanced Scientific Computing Research; U.S. Department of Energy at the Oak Ridge National Laboratory, which is managed by UT-Battelle, LLC under Contract No. De-AC05-00OR22725.

The authors thank J. E. Shepherd and his group (J. C. Krok, Z. Liang, J. Karnesky and F. Pintgen) from the Graduate Aeronautical Laboratory of the California Institute of Technology for kindly providing the unpublished experimental validation results shown in the Figs. 14 and 16. Further, we thank V. S. Deshpande and N. A. Fleck from the Department of Engineering of the University of Cambridge for providing the validation result shown in Fig. 11.

References

- [1] Aivazis, M., Goddard, W., Meiron, D. I., Ortiz, M., Pool, J. and Shepherd, J. A virtual test facility for simulating the dynamic response of materials. *Comput. Science & Engineering* 2(2):42–53, 2000.
- [2] Deiterding, R., Radovitzky, R., Mauch, S. P., Noels, L., Cummings, J. C. and Meiron, D. I. A virtual test facility for the efficient simulation of solid materials under high energy shock-wave loading. *Engineering with Computers* 22(3-4):325–347, 2006.
- [3] Specht, U. *Numerische Simulation mechanischer Wellen an Fluid-Festkörper-Mediengrenzen*. Fortsch.-Ber. VDI Reihe 7 Nr. 398, VDU Verlag, Düsseldorf, 2000.
- [4] Arienti, M., Hung, P., Morano, E. and Shepherd, J. A level set approach to Eulerian-Lagrangian coupling. *J. Comput. Phys.* 185:213–251, 2003.
- [5] Löhner, R., Baum, J., Charman, C. and Pelessone, D. Fluid-structure interaction simulations using parallel computers. *Lecture Notes in Computer Science*, vol. 2565, Springer, Berlin, 3–23, 2003.
- [6] Hu, H., Patankar, N. and Zhu, M. Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian technique. *J. Comput. Phys.* 169:427–462, 2001.
- [7] Farhat, C., Lesoinne, M. and LeTallec, P. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Comput. Methods Appl. Mech. Engrg.* 157:95–114, 1998.
- [8] Peskin, C. S. and McQueen, D. M. A 3-dimensional computational method for blood-flow in the heart. 1. Immersed elastic fibers in a viscous incompressible fluid. *J. Comput. Phys.*, 81(2):372–405, 1989.
- [9] Fedkiw, R. P., Aslam, T., Merriman, B. and Osher, S. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152:457–492, 1999.
- [10] Fedkiw, R. P. Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method. *J. Comput. Phys.* 175:200–224, 2002.
- [11] Tseng, Y.-H. and Ferziger, J. H. A ghost-cell immersed boundary method for in complex geometry. *J. Comput. Phys.* 192:593–623, 2003.
- [12] Hewett, D. W. The embedded curved boundary method for orthogonal simulation meshes. *J. Comput. Phys.* 138:585–616, 1997.
- [13] Roma, A. M., Peskin, C. S. and Berger, M. J. An adaptive version of the immersed boundary method. *J. Comput. Phys.* 153:509–534, 1999.
- [14] Chao, T.-W. *Gaseous detonation-driven fracture of tubes*. PhD thesis, California Institute of Technology, 2004.
- [15] Cirak, F. and Radovitzky, R. A Lagrangian-Eulerian Shell-Fluid Coupling Algorithm Based on Level Sets. *Computers & Structures*, 83:491–498, 2005.
- [16] Fickett, W. and Davis, W. C. *Detonation*. University of California Press, Berkeley and Los Angeles, California, 1979.
- [17] Williams, F. A. *Combustion theory*. Addison-Wesley, Reading, 1985.
- [18] Deiterding, R. *Parallel adaptive simulation of multi-dimensional detonation structures*. PhD thesis, Brandenburgische Technische Universität Cottbus, 2003.
- [19] Shepherd, J. E. Detonation waves and propulsion. In *Combustion in high-speed flows: Proc. of a Workshop on Combustion, Oct 12 - 14, 1992, Hampton*, Buckmaster, J., Ed. Kluwer Acad. Publ., 373–420, 1994.
- [20] *Gas Research Institute Mechanism 3.0*. http://www.me.berkeley.edu/gri_mech.
- [21] Mader, C. L. *Numerical modeling of detonations*. University of California Press, Berkeley and Los Angeles, California, 1979.
- [22] Toro, E. F. *Riemann solvers and numerical methods for fluid dynamics*. Springer, Berlin, Heidelberg, 2nd edition, 1999.
- [23] Berger, M. and Colella, P. Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.*, 82:64–84, 1988.
- [24] Deiterding, R. *AMROC - Blockstructured Adaptive Mesh Refinement in Object-oriented C++*. <http://amroc.sourceforge.net>.
- [25] Cirak, F., Ortiz, M. and Schröder, P. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *Int. J. Numer. Meth. Engineering*, 47:2039–2072, 2000.

- [26] Cirak, F. and Ortiz, M. Fully C^1 -conforming subdivision elements for finite deformation thin-shell analysis *Int. J. Numer. Meth. Engineering*, 51:813–833, 2001.
- [27] Cirak, F., Ortiz, M. and Pandolfi, A. A Cohesive Approach to Thin-Shell Fracture and Fragmentation. *Computer Methods in Appl. Mechanics and Engineering*, 194:2604–2618, 2005.
- [28] Chapelle, D. and Bathe, K. J. *The Finite Element Analysis of Shells - Fundamentals*. Springer, 2003.
- [29] Ortiz, M. and Pandolfi, A. Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis. *Int. J. Numer. Meth. Engineering*, 44:1267–1282, 1999.
- [30] Cuitino, A. and Ortiz, M. A material-independent method for extending stress update algorithms from small-strain plasticity to finite plasticity with multiplicative kinematics. *Engineering Computations*, 9:437–451, 1992.
- [31] deBorst, R. The zero-normal stress condition in plane-stress shell elasto-plasticity. *Comm. Meth. Appl. Numer. Meth.*, 7:29–33, 1991.
- [32] Mauch, S. P. *Efficient Algorithms for Solving Static Hamilton-Jacobi Equations*. PhD thesis, California Institute of Technology, 2003.
- [33] Deshpande, V. S., Heaver, A., and Fleck, N. A. An underwater shock simulator. *Proc. R. Soc. A* 462:1021–1041, 2006.
- [34] Qui, X., Deshpande, V. S., and Fleck, N. A. Dynamic response of a clamped circular sandwich plate subject to shock loading. *J. Appl. Mech.*, 71:637–645, 2004.
- [35] Lesuer, D. R., Kay, G.J. and LeBlanc, M. M. *Modeling large-strain, high rate deformation in metals*. UCRL-JC-134118, Lawrence Livermore National Laboratory, 2001.
- [36] Li, W. and Siegmund, T. An analysis of crack growth in thin-sheet metal via a cohesive zone model. *Engineering Fracture Mechanics* 69:2073–2093, 2002.

Mass density	$\rho = 8920 \text{ kg/m}^3$
Young's modulus	$E = 130 \text{ GPa}$
Poisson's ratio	$\nu = 0.31$
Yield stress	$\sigma_y = 38.5 \text{ MPa}$
Reference plastic strain	$\epsilon_0^p = 0.0091$
Hardening exponent	$1/n = 0.627$

Table 1: Material properties for annealed copper.

Mass density	$\rho = 2719 \text{ kg/m}^3$
Young's modulus	$E = 69.0 \text{ GPa}$
Poisson's ratio	$\nu = 0.33$
Yield stress	$\sigma_y = 275 \text{ MPa}$
Reference plastic strain	$\epsilon_0^p = 0.001$
Hardening exponent	$1/n = 0.07$
Reference plastic strain rate	0.6 1/s
Strain rate sensitivity exponent	$1/m = 0.01$

Table 2: Material properties for Al 6061-T6.

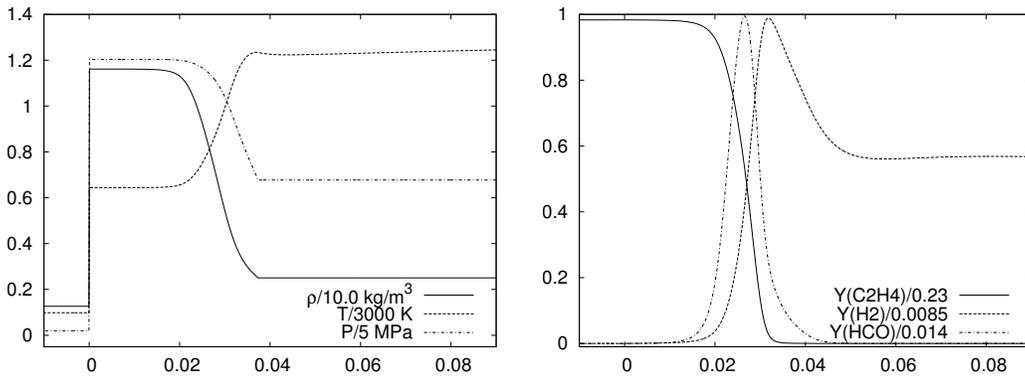


Figure 1: 1d detonation structure for stoichiometric $C_2H_4 + 3O_2$ at initial pressure $p_0 = 100 \text{ kPa}$ and detonation velocity $D = 2291.7 \text{ m/s}$. The abscissae display the distance behind the detonation front in mm.

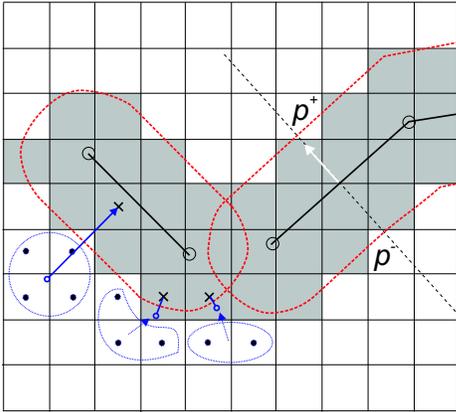


Figure 2: Ghost cells (shaded gray) around shell elements and construction of mirrored values.

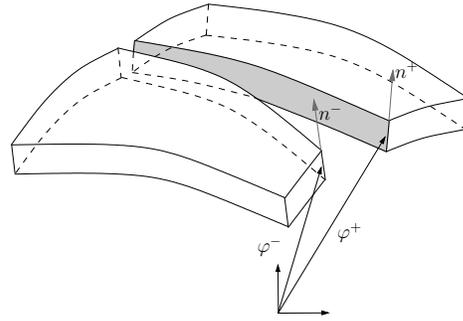
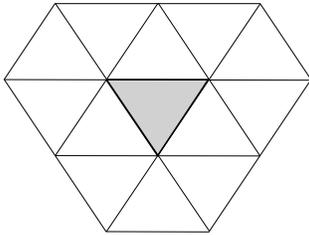
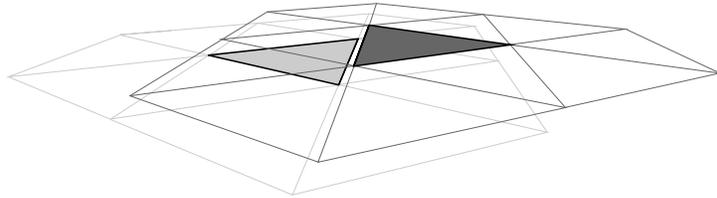


Figure 3: Fractured shell body: opposite crack flanks and corresponding normals.



(a)



(b)

Figure 4: (a) A triangular subdivision element (shaded triangle at the center) and its 12 control nodes. (b) A cohesive edge and its two adjacent elements.

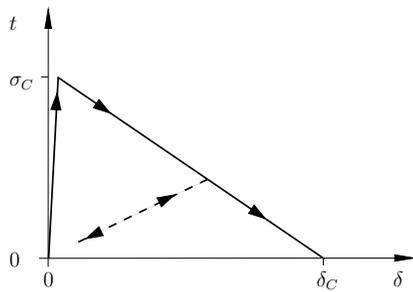


Figure 5: Irreversible linear cohesive law. Note that the initial stiff elastic response enforces the displacement continuity prior to crack initiation at σ_c . The dashed line represents the loading-unloading rule.

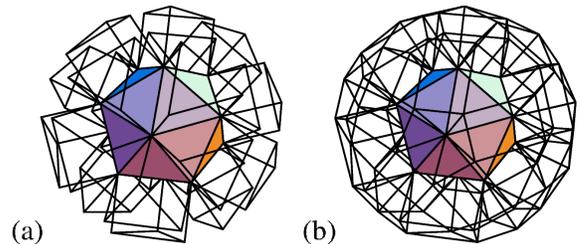


Figure 6: The characteristic polyhedra for faces and edges of an icosahedron.

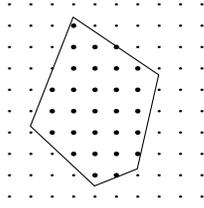


Figure 7: Scan conversion of a polygon in 2d and slicing of a polyhedron to form polygons.

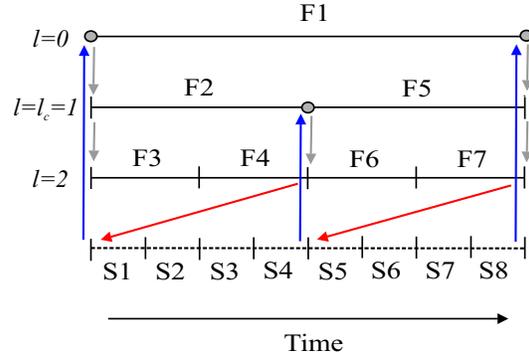


Figure 8: Data exchange between the recursive CFD solver and the linear thin-shell CSD solver throughout one SAMR root level time step.

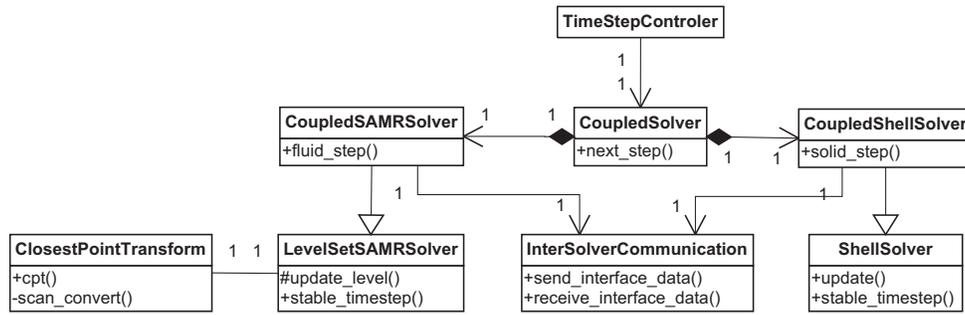


Figure 9: UML diagram for implementing a typical FSI problem with VTF classes.

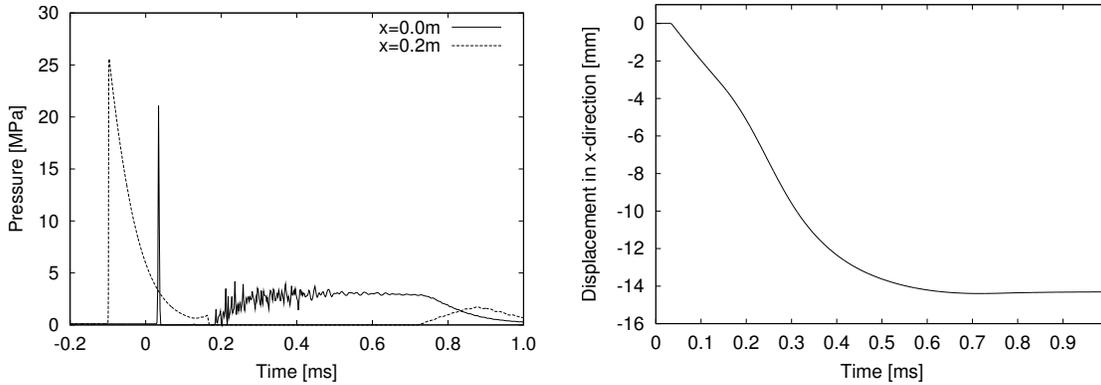


Figure 10: Left: fluid pressure traces at $x = 0$ m and $x = 0.2$ m. Right: maximal out-of-plane displacement of the plate versus time.

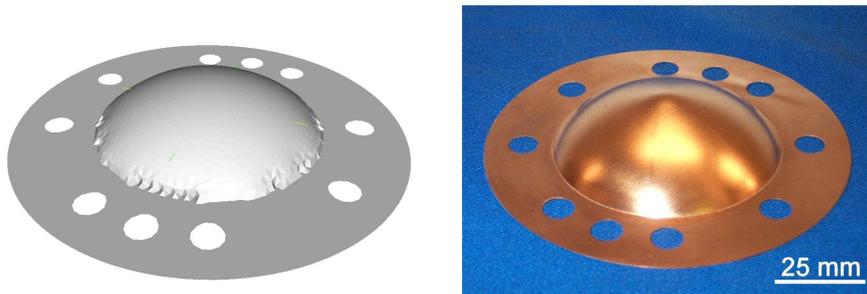


Figure 11: Deformed plate at end of simulation at $t_e = 1.0$ ms (left) and after the experiment (right).

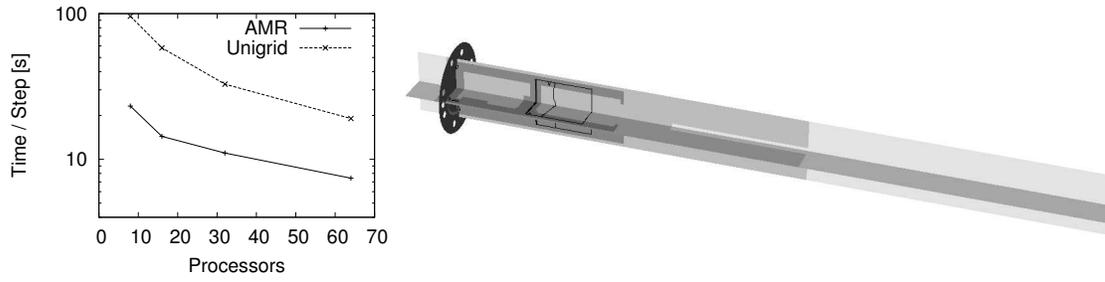


Figure 12: Scalability test for the simulation of Sec. 5.3 with fluid mesh adaptation and without (right). The left graphic visualizes the levels of block-structured mesh refinement in the fluid domain (shaded in gray tones) at $t = 0.2$ ms.

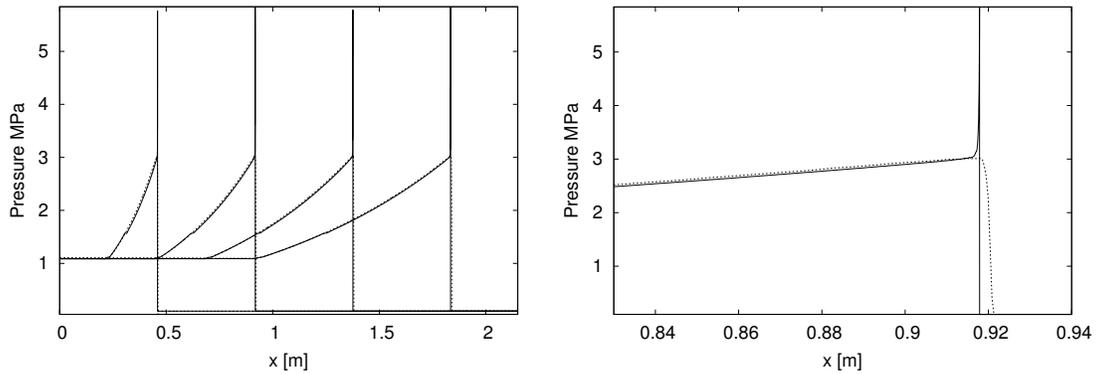


Figure 13: Left: comparison of pressure distribution for the one-step chemistry (solid) with the CV burn model (dotted) 0.2 ms, 0.4 ms, 0.6 ms, and 0.8 ms after ignition (time steps from left to right). Right: enlargement for 0.4 ms.

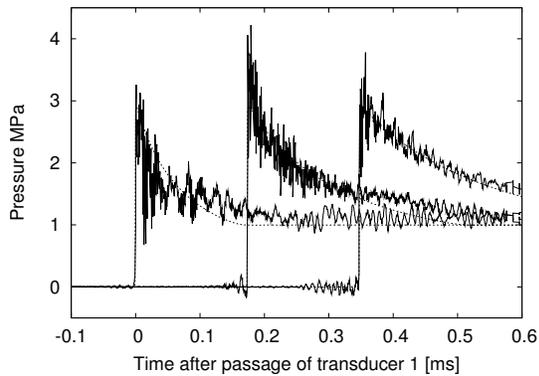


Figure 14: Comparison of pressure traces in experiment (solid) and in the 1d simulation with the CV burn model (dotted).

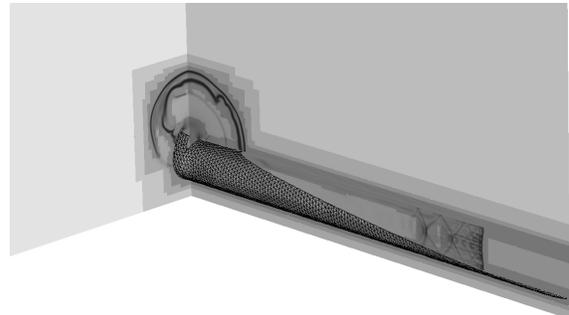


Figure 15: Fluid mesh adaptation and deforming solid mesh at $92 \mu\text{s}$.

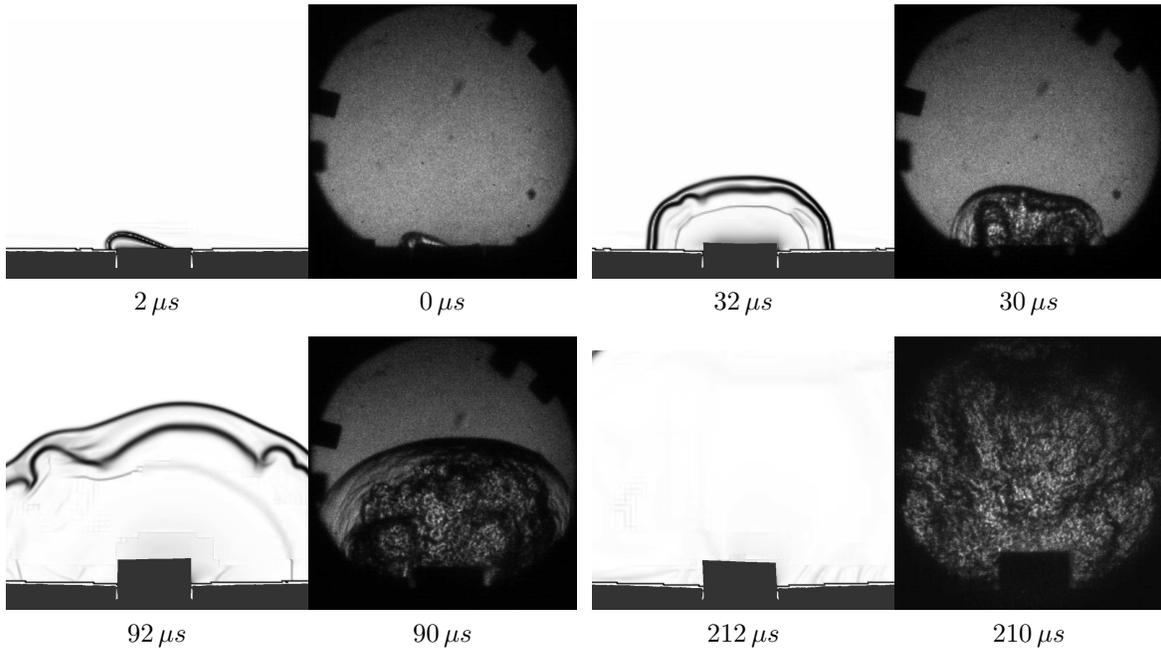


Figure 16: Simulated Schlieren pictures of fluid density and side view of the deforming solid mesh compared to Schlieren photographs taken in a corresponding experiment.

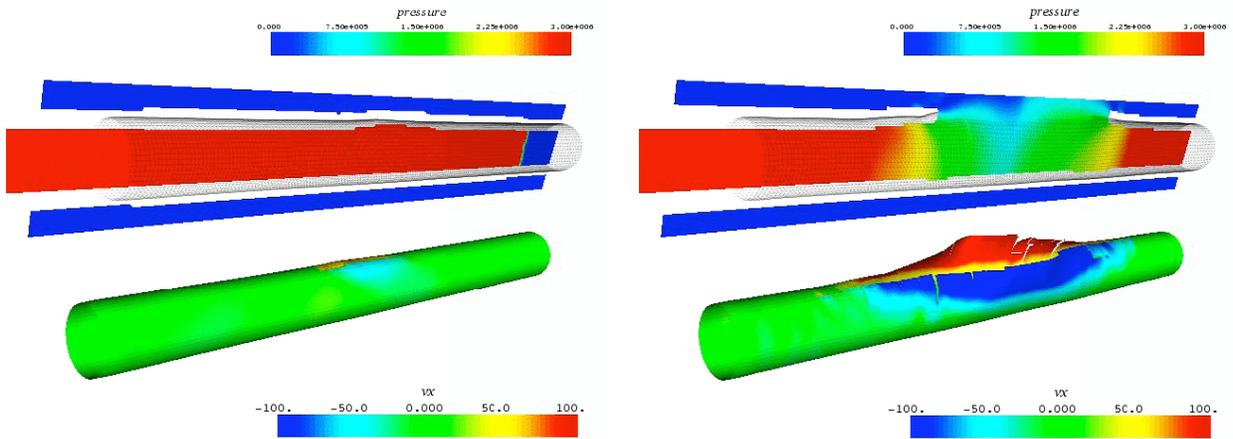


Figure 17: Detonation-driven rupture of the thin aluminum tube. Snapshots of the solid mesh with velocity iso-contours (lower row) and cuts through the fluid domain show the opening due to fracture and the resulting hydrodynamic venting (upper row).