European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2006
P. Wesseling, E. Oñate and J. Périaux (Eds)

A VIRTUAL TEST FACILITY FOR SIMULATING DETONATION-INDUCED DEFORMATION AND FRACTURE OF THIN FLEXIBLE SHELLS

Ralf Deiterding*, Fehmi Cirak†, Sean P. Mauch* and Daniel I. Meiron*

*California Institute of Technology, Mail Code 158-79,
Pasadena, CA 91125, USA
e-mail: ralf,sean,dim@cacr.caltech.edu
web page: http://www.cacr.caltech.edu/asc
†University of Cambridge, Department of Engineering,
Cambridge, CB2 1PZ, UK
e-mail: fc286@eng.cam.ac.uk

Key words: Fluid-structure interaction, detonations, thin-shells, large deformations, fracture, dynamic mesh adaptation, parallelization

Abstract. *The fluid-structure interaction simulation of detonation- and shock-wave-loaded fracturing thin-walled structures requires numerical methods that can cope with large deformations as well as topology changes. We present a robust level-set-based approach that integrates a Lagrangian thin shell finite element solver with fracture and fragmentation capabilities with an Eulerian Cartesian detonation solver with optional dynamic mesh adaptation. As computational application, the induction of large plastic deformations and the rupture of thin aluminum tubes due to the passage of ethylene-oxygen detonation waves is presented.*

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, 24, 14]. The VTF targets highly coupled problems, such as the high rate deformation of metals due 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 [27, 3, 20].

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]. The idea behind the VTF is therefore to develop a fluid-structure interaction (FSI) framework, including fluid mesh adaptation, that supports above coupling methodology and allows the embedding of propagating triangulated surface meshes derived from an arbitrary CSD solver into easily exchangeable Cartesian Eulerian CFD solvers. This concept facilitates solver reuse and modularization (see also [21] for further discussion of modular concepts for FSI simulation). Specific to the VTF is that scalar level set functions storing the distance information to the embedded surface are used to represent the complex geometry on the Cartesian fluid mesh and a ghost-fluid-type approach is employed to impose fluid boundary conditions [16, 15, 3].

In the present paper, we apply the VTF to the challenging problem of simulating the deformations and fracture of thin-walled aluminum tubes due to the passage of gaseous detonations in ethylene-oxygen mixtures [6]. The configuration under investigation is motivated by accidents in cooling systems of power-plants and will serve as a validation case for the VTF. In Sec. 2, we sketch the adaptive Cartesian finite volume fluid solver with level-set-based embedded boundary capability and also detail the employed detonation model. Section 3 describes the CSD solver that has been developed to enable FSI simulations of thin-walled (possibly fracturing) solid structures [10]. The solver is founded 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 evolving triangulated surface meshes efficiently into signed or unsigned distance functions. The fluid-structure coupling algorithm and its implementation on distributed memory computing platforms are described in Sec. 5. In Sec. 6, we present a series of computations, including the comparison with experimental results, that validate our methodology for detonation-driven large plastic deformations and discuss preliminary results for the rupturing case.

2 EULERIAN DETONATION SOLVER

The governing equations of detonation wave propagation in gases are the inviscid Euler equations [17]. Throughout this paper, we consider only the simplified case of a single exothermic chemical reaction $A \longrightarrow B$ with a progress variable Y corresponding to the mass fraction ratio between the partial density of the reactant A and the total density ρ, i.e. $Y = \rho_\mathrm{A}/\rho$. The governing equations of the hydrodynamic model are

$$\begin{array}{rcl}
\partial_t \rho + & \nabla \cdot (\rho \boldsymbol{u}) & = 0 \;, \\
\partial_t (\rho \boldsymbol{u}) + & \nabla \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u}) + \nabla p & = 0 \;, \\
\partial_t (\rho E) + & \nabla \cdot ((\rho E + p) \boldsymbol{u}) & = 0 \;, \\
\partial_t (Y\rho) + & \nabla \cdot (Y \rho \boldsymbol{u}) & = \psi \;.
\end{array} \tag{1}$$

Herein, $\boldsymbol{u}$ is the velocity vector and E the specific total energy. The hydrostatic pressure p is given by $p = (\gamma - 1)(\rho E - \frac{1}{2}\rho \boldsymbol{u}^T \boldsymbol{u} - \rho Y q)$ with γ denoting the ratio of specific heats and q the heat release due to the chemical reaction per unit mass.

2.1 Constant-volume burn detonation model

A one-step reaction would typically be modeled with an Arrhenius law such as [17]

$$\psi = -kY\rho \exp\left(\frac{-E_\mathrm{A}\rho}{p}\right) , \tag{2}$$

but in the specific case considered here, we utilize the constant volume burn model suggested by Mader [22]. This model neglects the detailed chemical depletion, and therefore the internal detonation structure, but ensures the right propagation speed and the correct state in chemical equilibrium at all grid resolutions. The 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 reactant density ρ_A, pressure p, and total energy E are modified locally in each cell; the total density ρ and the velocity vector $\boldsymbol{u}$ remain unaltered. The algorithm for the detonation model reads:

```
$V := \rho^{-1}$, $V_0 := \rho_0^{-1}$, $V_\mathrm{CJ} := \rho_\mathrm{CJ}$
$Y' := 1 - (V - V_0)/(V_\mathrm{CJ} - V_0)$
if $0 \le Y' \le 1$ and $Y > 10^{-8}$
    if $Y < Y'$ and $Y' < 0.9$ then $Y' := 0$
    if $Y' < 0.99$ then $p' := (1 - Y')p_\mathrm{CJ}$ else $p' := p$
    $\rho_\mathrm{A} := Y'\rho$, $E := p'/(\rho(\gamma - 1)) + Y' q_0 + \frac{1}{2}\boldsymbol{u}^T\boldsymbol{u}$
```

In the latter, the index 0 indicates the unreacted state (assumed to be constant), while CJ refers to the equilibrium values that can be calculated in advance following Chapman-Jouguet theory [17, 12] for a given detonation velocity.

2.2 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. [12]). Second-order accuracy in smooth solution regions is achieved with the MUSCL-Hancock variable extrapolation technique [28]. 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 [16]. 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 $\boldsymbol{n} = -\nabla\phi/|\nabla\phi|$. Note that for topologically closed boundary surfaces the signed distance may be used instead of the unsigned distance [10]. 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. 1). 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 $\boldsymbol{w}$ is $\boldsymbol{u}\cdot\boldsymbol{n} = \boldsymbol{w}\cdot\boldsymbol{n}$. Enforcing the latter with ghost cells, in which the discrete values are located at the cell centers, requires the mirroring of the primitive values ρ, $\boldsymbol{u}$, p, ρ_A across the embedded boundary. The normal velocity in the ghost cells is set to $(2\boldsymbol{w}\cdot\boldsymbol{n} - \boldsymbol{u}\cdot\boldsymbol{n})\boldsymbol{n}$, while the mirrored tangential velocity remains unmodified. Mirrored values are constructed by calculating spatially interpolated values in the point $\tilde{\boldsymbol{x}} = \boldsymbol{x} + 2\phi\boldsymbol{n}$ 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 1 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 $\boldsymbol{w}$ in the ghost cells guarantees that the embedded boundary propagates at most one cell in every time step.

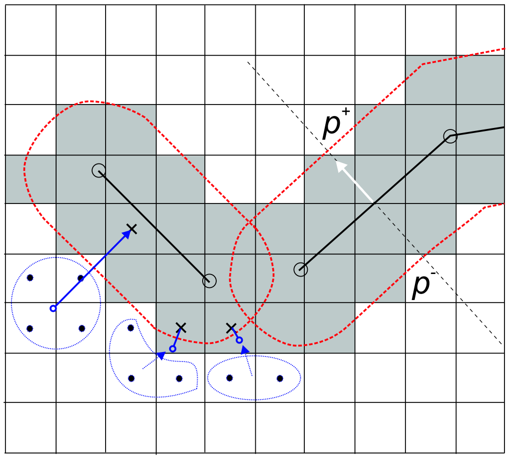


Figure 1: Ghost cells (shaded gray) around shell elements and construction of mirrored values.

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 alleviate such errors and the unavoidable staircase approximation of the boundary with this approach effectively by using the dynamic mesh adaptation technique described in the next sub-section to also refine the Cartesian mesh appropriately along the boundary. Some authors have also presented cut-cell techniques that utilize the correct boundary flux [26, 5], but the proposed numerical circumventions of the severe time step restriction in time-explicit schemes, that can result from small cells created by the boundary intersection, are inherently complicate and most approaches have not been extended successfully to three space dimensions yet.

2.3 Structured adaptive mesh refinement

In order to supply a fine local temporal and spatial resolution efficiently, the finite volume scheme described above has been incorporated into the block-structured adaptive mesh refinement (SAMR) method after Berger and Colella [4]. Characteristic for the idea of structured mesh adaptation is that the finite volume method is technically not implemented in a cell-based fashion but rather in a routine which operates on equidistant subgrids. The subgrids become computationally decoupled during one update through the use of ghost or halo cells. Cells being flagged for refinement (shaded in Fig. 2) are clustered recursively into non-overlapping rectangular subgrids and a hierarchy of successively embedded levels is thereby constructed (cf. Fig. 2). All mesh widths on level l are r_l-times finer than on level $l-1$, i.e. $\Delta t_l := \Delta t_{l-1}/r_l$ and $\Delta x_{k,l} := \Delta x_{k,l-1}/r_l$ with $r_l \geq 2$ for $l > 0$ and with $r_0 = 1$, and a time-explicit finite volume scheme will (in principle) remain stable on all levels of the hierarchy.

Different levels are integrated recursively in time allowing the derivation of temporally and spatially interpolated boundary conditions of Dirichlet-type from the coarser level at coarse-fine interfaces. Values of cells covered by finer subgrids are overwritten by averaged fine grid values subsequently. This operation leads to a modification of the numerical stencil on the coarse mesh and requires a special flux correction in cells abutting a fine grid. In order to ensure discrete conservation (at least for purely Cartesian problems without embedded boundaries), but particular to enforce a vonNeumann-type boundary condition matching at coarse-fine interfaces subsequently, the coarse flux approximation adjacent to modified coarse level cells is replaced with the sum of all overlying fine level fluxes. See [4] or [12] for details.

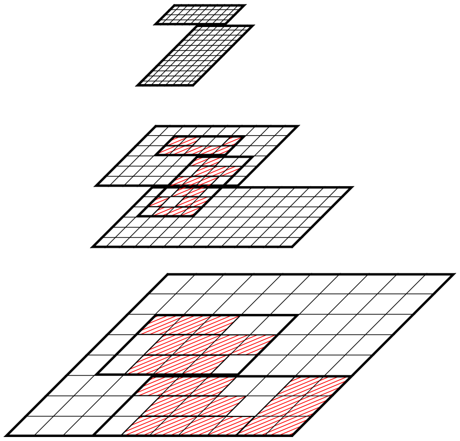


Figure 2: SAMR hierarchy.

SAMR in the VTF is provided generically by the

AMROC (Adaptive Mesh Refinement in Object-oriented C++) framework [2]. AMROC has been parallelized effectively for distributed memory machines [13] and can be used on all systems that provide the MPI library. The parallelization strategy is a rigorous domain decomposition approach that partitions the SAMR hierarchy from the root level on. The key idea is that all higher level domains are required to follow this “floor plan”. In AMROC, a generalization of Hilbert’s space-filling curve [25] is currently used to derive load-balanced root level distributions at run time.

3 LAGRANGIAN THIN-SHELL SOLVER

The Kirchhoff-Love type thin-shell model applied in this study takes the membrane as well as bending response of the surface into account, and has been discretized with subdivision finite elements [7, 8]. The underlying kinematic assumptions allow for arbitrarily large displacements as well as rotations of the shell. Fracture initiation and propagation is considered as a progressive failure phenomenon in which the separation of the crack flanks is resisted by cohesive tractions. The relationship between the crack-opening displacements and the tractions is given by a cohesive model. Cohesive interface elements are inserted at inter-element edges and constrain the opening of the crack flanks to the deformation of the shell middle surface and its normal. This approach allows for fracture in an in-plane or tearing mode, a shearing mode, or a “bending of hinge” mode.

To kinematically describe a fractured thin-shell as sketched in Fig. 3, we consider a shell of uniform thickness $\overline{h}$ occupying an undeformed configuration $\overline{V}$. The position vector $\overline{\boldsymbol{\varphi}}$ of a material point on the undeformed shell body is assumed to be

$$\overline{\boldsymbol{\varphi}} = \overline{\boldsymbol{x}} + \theta^3 \overline{\boldsymbol{n}} \tag{3}$$

with $-h/2 \leq \theta^3 \leq h/2$. The position vector of the shell middle surface is denoted by $\overline{\boldsymbol{x}}$ and its out-of-surface unit normal by $\overline{\boldsymbol{n}}$. In other words, the shell middle surface represents a two-dimensional manifold in $\mathbb{R}^3$. The deformation mapping $\boldsymbol{\varphi}$ maps the shell body into the deformed configuration V and is discontinuous across the crack, i.e.

$$[\![\boldsymbol{\varphi}]\!] = \boldsymbol{\varphi}^+ - \boldsymbol{\varphi}^- = [\![\boldsymbol{x}]\!] + \theta^3 [\![\boldsymbol{n}]\!] \, , \tag{4}$$

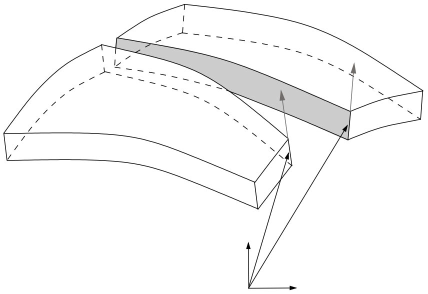


Figure 3: Fractured shell body: Opposite crack flanks and corresponding normals.

where the superscripts + and − refer to the opposing crack flanks. Further, the first term describes the discontinuity of the deformation of the middle shell 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.

A standard semi-inverse approach is followed for obtaining the shell equilibrium equations in weak from. To this end, the assumed reduced kinematic equations for the shell body (Equations (3) and (4)) 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 ($\delta\Pi_{C,\mathrm{int}}$) in addition to the virtual work of the bulk material ($\delta\Pi_{S,\mathrm{int}}$)

$$\delta\Pi_{S,\mathrm{int}} + \delta\Pi_{C,\mathrm{int}} - \delta\Pi_{\mathrm{ext}} = 0 \tag{5}$$

with

$$\delta\Pi_{S,\mathrm{int}} = \int_{\overline{\Omega}} \int_{-\overline{h}/2}^{\overline{h}/2} \boldsymbol{P} : \delta\boldsymbol{F}\,\mu\,d\theta^3 d\overline{\Omega}\ , \quad \delta\Pi_{C,\mathrm{int}} = \int_{\overline{\Gamma}_C} \int_{-\overline{h}/2}^{\overline{h}/2} \boldsymbol{T} \cdot [\![\boldsymbol{\varphi}]\!]\,\mu\,d\theta^3 d\overline{\Gamma}_C\ ,$$

where $\boldsymbol{P}$ is the first Piola-Kirchhoff stress tensor, $\boldsymbol{T}$ the related traction vector at the cohesive surface, and $\boldsymbol{F}$ the deformation gradient. The virtual work expression for the bulk material is integrated over the undeformed shell middle surface $\overline{\Omega}$ and for the cohesive interface over the crack path $\overline{\Gamma}_C$. The scalar factor μ accounts for the curvature of the shell in the volume computation [8].

Next, we briefly outline the discretization of the governing equation (5). A detailed presentation of the used subdivision finite element discretization technique can be found in [7] and [8]. In this approach, the reference ($\overline{\boldsymbol{x}}$) and deformed ($\boldsymbol{x}$) 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. The overlapping local subdivision interpolants, each defined over one patch, together lead to a global interpolant with square-integrable curvatures.

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 [9] for details). The topological changes necessary to the nonlocal subdivision functions and the underlying control mesh in order to describe the 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. Each element patch consists of a triangular element and the nodes in its neighborhood (see Fig.4). Prior to crack nucleation, the coupling of the elements is enforced by appying stiff elastic cohesive interfaces at all non-cracked 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 [9]. In this model, the opening displacement $[\![\boldsymbol{\varphi}]\!]$ plays the role of a deformation measure while the traction $\boldsymbol{T}$ is the conjugate stress measure.

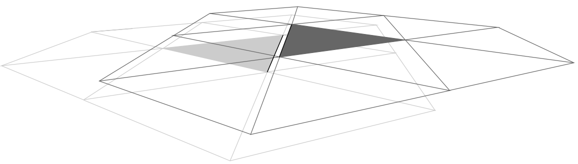


Figure 4: One cohesive edge and the two elements with their one neighborhoods

Finally, the inelastic behavior of the bulk material, i.e. the relation between $\boldsymbol{P}$ and $\boldsymbol{F}$, is described with a conventional J_2 plasticity model with isotropic power-law hardening [11]. The power-law hardening for the flow stress g has the form

$$g(\epsilon^p) = \sigma_y \left(1 + \frac{\epsilon^p}{\epsilon_0^p}\right)^{1/n} , \tag{6}$$

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

$$\sigma_{\text{eff}} = g(\epsilon^p) \left(1 + \frac{\dot{\epsilon}^p}{\dot{\epsilon}_0^p}\right)^{1/m} , \tag{7}$$

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

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 [23]. 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. 5 for face (a) and edge (b)

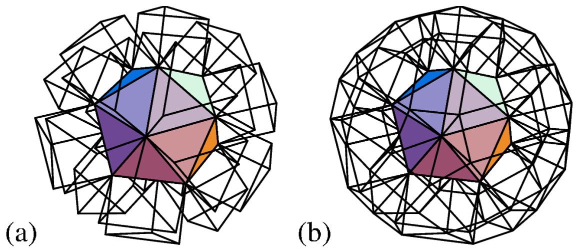


Figure 5: The characteristic polyhedra for faces and edges of an icosahedron.

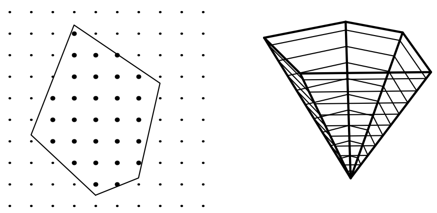


Figure 6: Scan conversion of a polygon in 2-D and slicing of a polyhedron to form polygons.

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. 6. 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 [23, 14].

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, $p^S = (\boldsymbol{\sigma n})\boldsymbol{n}$ with $\boldsymbol{\sigma} = 1/det(\boldsymbol{F})\boldsymbol{FP}$, 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 $\phi(t)$
$\boldsymbol{w}_F^{+/-} := \boldsymbol{u}^S(t)$
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 $\boldsymbol{w}$ is enforced in the fluid on upper ($^+$) and lower ($^-$) side of each element. After setting embedded rigid wall boundary conditions as sketched in Sec. 2 and the fluid update, a new hydrodynamic pressure load $p^F := p^+ - p^-$ on each shell element is derived (compare Fig. 1). 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.

While the implementation of a loosely coupled FSI method is straightforward with conventional solvers with consecutive time update, the utilization of the SAMR method 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. 7 for an exemplary SAMR hierarchy with two additional levels with $r_{1,2} = 2$. Figure 7 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 $\boldsymbol{u}^S$ after each solid update. The modification of refinement meshes is indicated in Fig. 7 by the gray arrows; the initiating base level, that remains fixed throughout the regridding operation, is indicated by the gray circles.

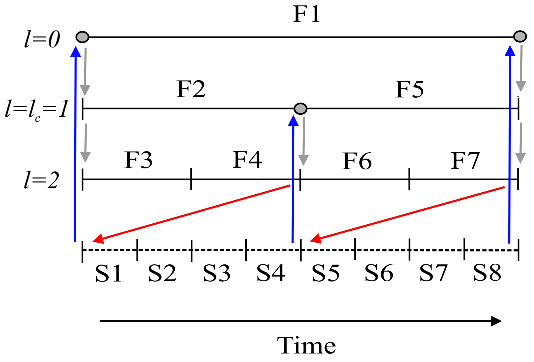


Figure 7: Data exchange between the recursive CFD solver and the linear thin-shell CSD solver throughout one SAMR root level time step.

In our current implementation, CFD and CSD solver are parallelized separately for distributed memory machines 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 library that determines the necessary point-to-point communication patterns by intersecting Cartesian bounding boxes enclosing the local domains. Details on this communication library and also a detailed algorithmic description of the coupled SAMR method can be found in [14].

6 COMPUTATIONAL RESULTS

The configuration we are interested in is an experimental setup developed by Chao [6]. It consists of a detonation tube of 1.52 m length to which thin-walled aluminum (Al6061-T6) test tubes are attached. The test 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 combustible mixture of ethylene and oxygen at equivalence ratio 1 ($C_2H_4 + 3\,O_2$) at room temperature 295 K. The initial pressure varies from $p_0 = 80\,$kPa to $p_0 = 180\,$kPa. The mixture is thermally ignited at the closed end and the combustion transitions quickly to a detonation wave. When it enters the test specimen, the detonation is close to the Chapman-Jouguet (CJ) limit of quasi-stationary self-sustained propagation. Its velocity is between 2300 m/s and 2400 m/s and the pressure values in the fully reacted Chapman-Jouguet state range between $p_{\rm CJ} \approx 2.60\,$MPa and

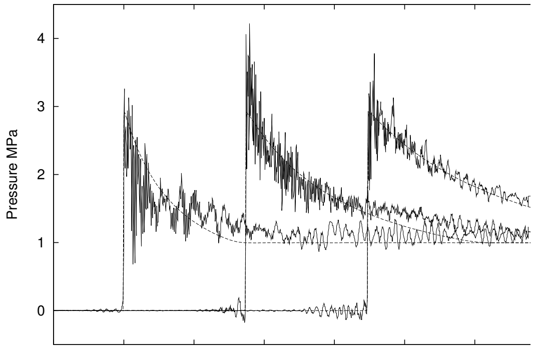


Figure 8: Comparison of the pressure traces at $x = 0.38\,\mathrm{m}$ (transducer 1), $x = 0.78\,\mathrm{m}$, $x = 1.18\,\mathrm{m}$ (from left to right) in an experiment and in a 1d simulation with the CV burn model.

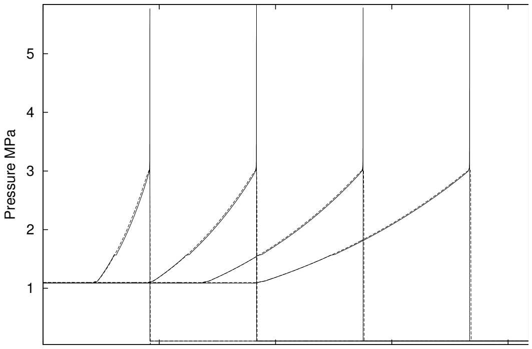


Figure 9: Left: comparison of pressure distribution for the one-step chemistry (solid) with the CV burn model (dashed) $200\,\mu$s $400\,\mu$s, $600\,\mu$s, and $800\,\mu$s after ignition (time steps from left to right).

$p_{\mathrm{CJ}} \approx 6.10\,\mathrm{MPa}$, depending on the initial pressure p_0. As the lower end of the 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 the CJ state 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 Detonation solver validation

In order to ensure the correct function of the detonation model described in Sec. 2.1, we carry out one-dimensional detonation simulations for an initial pressure of $p_0 = 100\,\mathrm{kPa}$. Separate calculations using Chapman-Jouguet theory (cf. [17]) that employ the full GRI 3.0 reaction mechanism predict a detonation velocity of $D_{\mathrm{CJ}} \approx 2376\,\mathrm{m/s}$, but we set the only parameter of the constant-volume burn model, the heat release parameter q, to $q = 4.70408075\,\mathrm{MJ/kg}$ leading to a detonation velocity of $D_{\mathrm{Ex}} = 2291.74\,\mathrm{m/s}$. This average speed of propagation has been measured in experiments for the particular configuration used by Chao [6] in which we are interested in here. 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 simple scaled gradients of total density, pressure and mass fraction Y. In the computation, the detonation is initiated by setting the values in all cells with midpoint $< 5\,\mathrm{mm}$ to the CJ values. Note that is a mandatory computational idealization as the CJ limit is reached in the experiment only close to the test specimen.

Figure 8 gives a comparison of the temporal pressure traces at the locations $x = 0.38\,\mathrm{m}$ (transducer 1), $x = 0.78\,\mathrm{m}$, and $x = 1.18\,\mathrm{m}$ in the one-dimensional simulation with experimentally measured pressure traces (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 and the idealized computational initial conditions.

To verify the equivalence of the chosen detonation model to fully resolved detonation computations for the time scales relevant to us, a simulation has been undertaken utilizing the one-step Arrhenius reaction (2) instead of the CV burn model. Based on stationary calculations of the internal detonation structure according to the theory after Zel'dovich, von Neumann, and Döring (see [17] for a detailed account) using the detailed chemical kinetics of GRI 3.0, we set the activation energy to $E_\mathrm{A} = 25,000\,\mathrm{J/mol}$ and the frequency parameter to $k = 2 \cdot 10^7\,\mathrm{s}^{-1}$. As the utilization of Eq. (2) requires an extremely high local resolution in the reaction zone, this comparative computation uses an SAMR base mesh of 4000 cells and three additional levels with identical refinement factors $r_{1,2,3} = 4$. Despite the use of mesh adaptation the computation requires several hours CPU compared to only seconds in the previous case.

Figure 9 compares the pressure distributions of the one-dimensional computations with one-step Arrhenius reaction (2) and CV burn model as the detonation propagates down the tube. Due to a reaction zone in the range of 10^{-2} to 10^{-1}mm the internal detonation structure with its leading vonNeumann pressure spike appears as an isolated peak value in Fig. 9. Apparently, it has negligible influence on the Taylor rarefaction wave following immediately behind the detonation front. As our succeeding simulations involve detonation-structure interaction times of several hundred microseconds it is physically justified to employ the CV burn model in the following.

6.2 Fluid-structure interaction validation

In three space-dimensions, we ensure the correct consideration of the Taylor rarefaction wave by initializing the flow field with the data from corresponding one-dimensional simulations taken at the moment when the detonation enters the specimen. While only the test specimen is simulated in the CSD solver, the CFD solver considers an additional tubular domain 0.92 m upstream. The extension is modeled by prescribing the level set function for the embedded boundary method directly and its purpose is to ensure the correct inflow of the Taylor wave into the specimen.

As a validation test for large plastic material deformations an experiment has been conducted in which an "H" shape pattern is cut into a specimen of 89.6 cm. Each cut has a length of 25 mm. One cut is in the longitudinal direction with its midpoint 44.4 cm, away from the inlet. The two other cuts extend perpendicular into the circumferential direction. 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 $[-92.0\,\mathrm{cm}, 89.6\,\mathrm{cm}] \times [-3.75\,\mathrm{cm}, 56.25\,\mathrm{cm}] \times [-39.0\,\mathrm{cm}, 39.0\,\mathrm{cm}]$, where the beginning of the specimen is set to

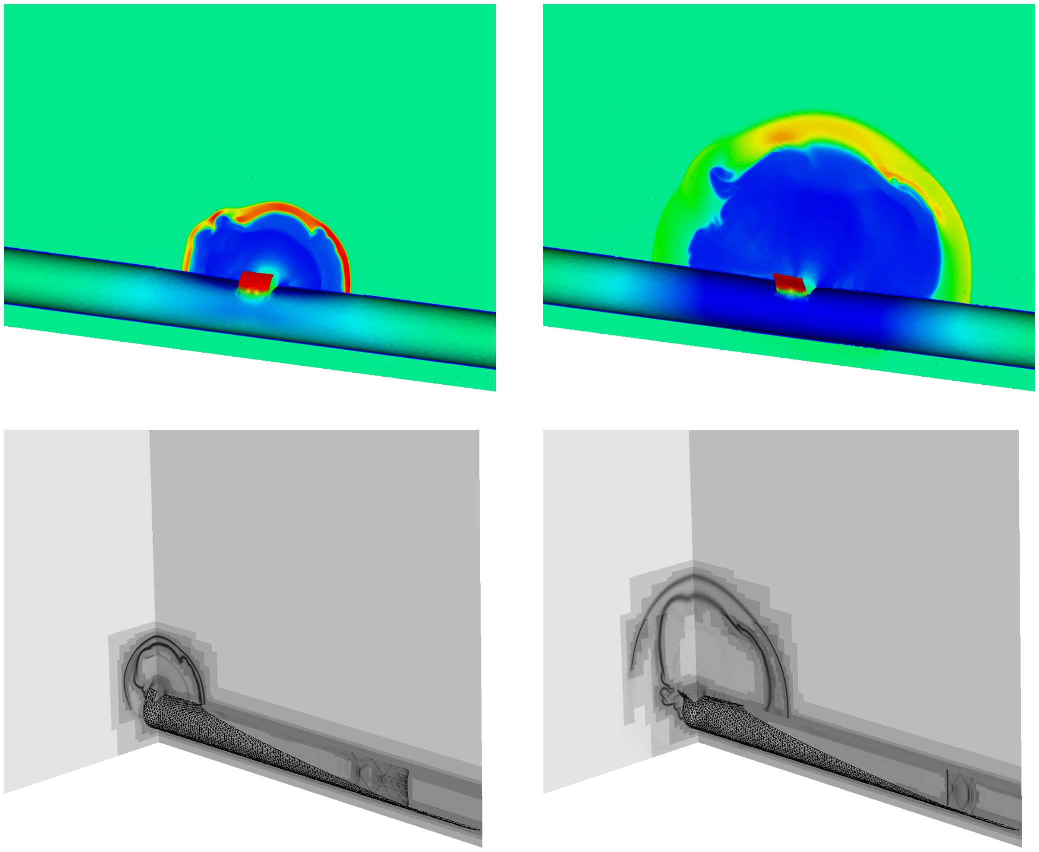
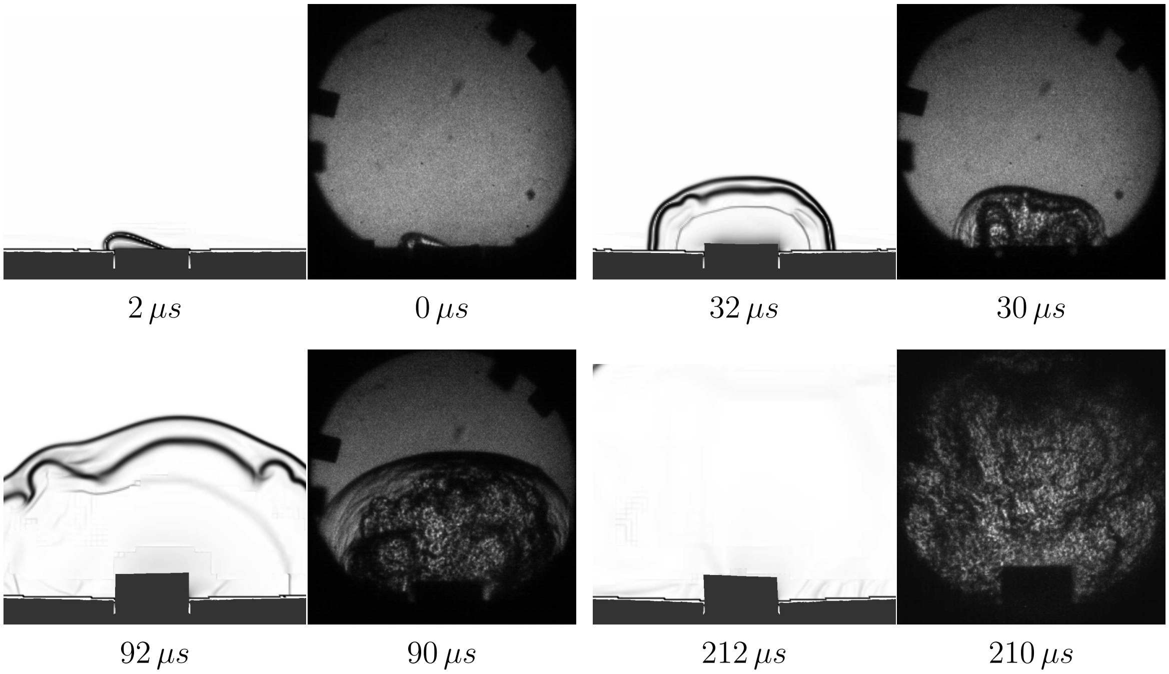


Figure 10: Upper row: color plot of fluid density in plane perpendicular to z-axis and shell displacements in the z-direction $t = 92\,\mu s$ (left) and $t = 212\,\mu s$ (right) after the detonation front had reached the middle of the longitudinal slot. Lower row: schlieren plot of fluid density in planes perpendicular to x- and z-axis on different refinement levels (gray).

$x = 0$. The opening of the flaps and the gaseous venting are visualized in the upper row of Fig. 10.

In this computation, an SAMR base mesh of $104 \times 80 \times 242$ cells with 3 additional level and refinement factors $r_{1,2} = 2$ and $r_3 = 4$ is used. Additional to the refinement criteria in Sec. 6.1 that capture the detonation front reliably at the highest level, the walls of the test specimen are always fully refined. The effective resolution at the walls allows for an offset parameter of $h = 0.81\,\mathrm{mm}$, which is less than twice the accurate wall thickness of $0.445\,\mathrm{mm}$. The refinement is constrained for $x < 0$, $y > 15.0\,\mathrm{cm}$ and $|z| > 15.75\,\mathrm{cm}$. The dynamic evolution of the block-structured mesh hierarchy is depicted in the lower row of Fig. 10. The graphics show schlieren of the fluid density on the three refinement levels which are visualized by shading their domains in different gray tones. The images highlight the enormous efficiency gain from dynamic mesh adaptation. An equivalent unigrid CFD calculation would require $> 7.9 \cdot 10^9$ cells, but our SAMR computation uses



$2\,\mu s$ $0\,\mu s$ $32\,\mu s$ $30\,\mu s$

$92\,\mu s$ $90\,\mu s$ $212\,\mu s$ $210\,\mu s$

Figure 11: Simulated schlieren pictures of fluid density and side view of the deforming solid mesh compared to schlieren photo graphs taken in a corresponding experiment.

only $\approx 4.0 \cdot 10^7$ cells on average.

For the CSD sub-problem, we use a triangular input mesh of 17,056 elements, in which each base element is subdivided internally into four smaller elements. The initial mesh is shown in Fig. 12. We employ a J2 plasticity model for aluminum with power-law hardening and thermal softening as bulk material model [11]. The parameters for our particular model have been adapted from the Johnson-Cook material parameters reported by Lesuer et al. [18]. An alternative reference for Al 6061-T651 material parameters is Warren et al. [29].

The computation ran on 72 Opteron-2.2 GHz processors connected with Infiniband network for about 4300 h CPU to a final time of $t_e = 460\,\mu$s. 460 coupled time steps with fixed step size have been simulated, where 5 solid solver sub-steps were taken in each fluid time step. As the wall boundary is refined up to the highest level, the coupling level, cf. Sec. 5, is naturally set to $l_c = 3$. In Fig 11, a series of schlieren photo graphs from the experiment showing the fluid venting and flap opening 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

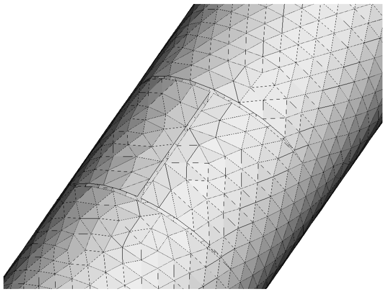


Figure 12: Initial mesh for the CSD shell solver in the coupled flap opening simulation.

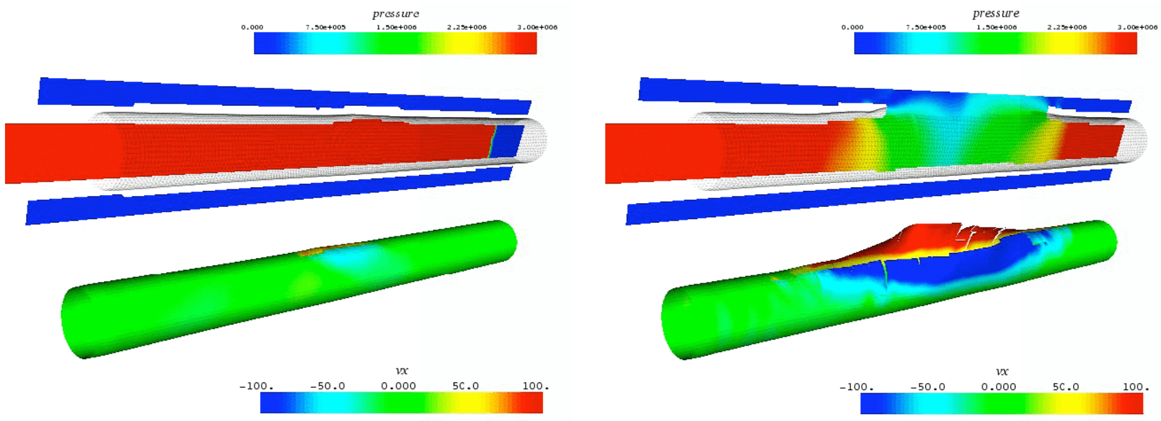


Figure 13: Coupled simulation of detonation-driven rupture of a thin aluminum tube. Two snapshots of the fracturing tube with velocity iso-contours (lower row); cuts through the fluid domain show the resulting hydrodynamic venting (upper row).

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 for the shown times steps, which confirms the correct function of the fluid-structure coupling method and the appropriateness of the chosen computational setup.

6.3 Detonation-driven fracture simulation

Finally, we present one exemplary fluid-structure interaction computation that involves the rupture of the test specimen. The initial pressure is $p_0 = 180\,\mathrm{kPa}$ and the specimen has a length of 45.7 cm. To ensure a reproducible fracture the specimen has a central longitudinal notch of 6.32 cm at the middle, which is modeled as an initial crack in the computations. The computational setting is similar as described above, but no fluid mesh adaptation has been employed yet: The simulation has preliminary character. Fracture simulations with a similar level of accuracy as the calculation shown in the previous sub-section are currently carried out.

The material model for the cohesive interface elements is a linearly decreasing envelope with a plane stress fracture toughness $K_{\mathrm{IC}} = 30MPa\sqrt{m}$ [9]. In accordance with Li et al.'s [19] numerical computations of thin-sheet ductile fracture, the crack initiation stress is chosen to be $\sigma_c = 2\sigma_y$, where σ_y denotes the yield stress of the bulk material.

Figure 13 visualizes the results for a shell mesh of 8665 elements and a uniform Cartesian fluid mesh of $40 \times 40 \times 725$ cells that required $\approx 900\,\mathrm{h}$ CPU 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 $t_e = 260\,\mu\mathrm{s}$ have been calculated (20 solid solver sub-steps in each fluid time step). The left graphic of Fig. 13 shows the beginning of the crack opening $\approx 150\,\mu\mathrm{s}$ after the detonation has passed the initial crack. The snapshot on the right shows the rupture at the final time $260\,\mu\mathrm{s}$. 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 and the update with the core Cartesian finite volume scheme sketched in Sec. 2.2, have about the same computational costs on each fluid processor. This result confirms that our approach utilizing distance functions for implicit geometry representation is sufficiently efficient for computing even complex FSI problems with large deformations and evolutions in the mesh topology with high computational efficiency.

7 CONCLUSIONS

A weakly coupled parallel level-set-based fluid-structure coupling methodology for the time-accurate simulation of thin flexible shells dynamically responding to gaseous detonation waves has been described. The approach has been demonstrated to handle arbitrary topology changes and large deformations in reasonable agreement with experimental results. 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 these components in the software framework “Virtual Test Facility” is freely available for research purposes (cf. http://www.cacr.caltech.edu/asc).

ACKNOWLEDGEMENTS

The authors would like to thank J. C. Krok, Z. Lianz, J. Karnesky, F. Pintgen, and J. E. Shepherd for providing the experimental validation results shown in the Figs. 8 and 11.

REFERENCES

[1] M. Aivazis, W. Goddard, D. I. Meiron, M. Ortiz M, J. Pool, J. Shepherd. A virtual test facility for simulating the dynamic response of materials. *Comput. Science & Engineering* 2(2):42–53, (2000).

[2] R. Deiterding. *AMROC - Blockstructured Adaptive Mesh Refinement in Object-oriented C++*. http://amroc.sourceforge.net.

[3] M. Arienti, P. Hung, E. Morano, J. Shepherd. A level set approach to Eulerian-Lagrangian coupling. *J. Comput. Phys.* 185:213–251, (2003).

[4] M. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.*, 82:64–84, (1988).

[5] M. J. Berger and C. Helzel. Grid aligned h-box methods for conservation laws in complex geometries. In *Proc. 3rd Intl. Symp. Finite Volumes for Complex Applications*, Porquerolles, June (2002).

[6] T.-W. Chao. *Gaseous detonation-driven fracture of tubes.* PhD thesis, California Institute of Technology, (2004).

[7] F. Cirak, M. Ortiz, P. Schröder. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *Int. J. Numer. Meth. Engineering*, 47:2039–2072, (2000).

[8] F. Cirak and M. Ortiz. Fully C^1-conforming subdivision elements for finite deformation thin-shell analysis *Int. J. Numer. Meth. Engineering*, 51:813–833, (2001).

[9] F. Cirak, M. Ortiz, and A. Pandolfi. A Cohesive Approach to Thin-Shell Fracture and Fragmentation. *Computer Methods in Appl. Mechanics and Engineering*, 194:2604–2618, (2005).

[10] F. Cirak and R. Radovitzky. A Lagrangian-Eulerian Shell-Fluid Coupling Algorithm Based on Level Sets. *Computers & Structures*, 83:491–498, (2005).

[11] A. Cuitino and M. Ortiz. 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).

[12] R. Deiterding. *Parallel adaptive simulation of multi-dimensional detonation structures.* PhD thesis, Brandenburgische Technische Universität Cottbus, (2003). Available at http://www.cacr.caltech.edu/˜ralf/html/pub.htm.

[13] R. Deiterding. Construction and application of an AMR algorithm for distributed memory computers. In T. Plewa, T. Linde, and V. G. Weirs, editors, *Adaptive Mesh Refinement - Theory and Applications*, volume 41 of *Lecture Notes in Computational Science and Engineering*, pages 361–372. Springer, (2005).

[14] R. Deiterding, R. Radovitzky, S. P. Mauch et al. A virtual test facility for the efficient simulation of solid materials under high energy shock-wave loading. *Engineering with Computers*, (2006). In press.

[15] R. P. Fedkiw. Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method. *J. Comput. Phys.* 175:200–224, (2002).

[16] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152:457–492, (1999).

[17] W. Fickett and W. C. Davis. *Detonation.* University of California Press, Berkeley and Los Angeles, California, 1979.

[18] D.R. Lesuer, G.J. Kay, and M.M. LeBlanc. Modeling large-strain, high rate deformation in metals. UCRL-JC-134118, Lawrence Livermore National Laboratory, 2001.

[19] W. Li and T. Siegmund. An analysis of crack growth in thin-sheet metal via a cohesive zone model. *Engineering Fracture Mechanics* 69:2073–2093, (2002).

[20] R. Löhner, J. Baum, C. Charman, D. Pelessone. Fluid-structure interaction simulations using parallel computers. *Lecture Notes in Computer Science 2565*, Springer, Berlin, pp. 3–23, (2003).

[21] R. Löhner, J. Cebral, C. Yang, J. Baum, E. Mestreau, C. Charman, D. Pelessone D. Large-scale fluid-structure interaction simulations. *Comput. Science & Engineering* 6(3):27–37, (2004).

[22] C. L. Mader. *Numerical modeling of detonations.* University of California Press, Berkeley and Los Angeles, California, (1979).

[23] S. P. Mauch. *Efficient Algorithms for Solving Static Hamilton-Jacobi Equations.* PhD thesis, California Institute of Technology, (2003).

[24] S. P. Mauch, D. .I. Meiron, R. Radovitzky, R. Samtaney R. *Coupled Eulerian-Lagrangian simulations using a level set method.* In Bathe K (ed). 2nd M.I.T. Conference on Computational Fluid and Solid Mechanics, Cambridge, MA, June 17-20, (2003).

[25] M. Parashar, J. C. Browne. *On Partitioning Dynamic Adaptive Grid Hierarchies.* In Proc. of the 29th Annual Hawaii Int. Conf. on System Sciences, (1996).

[26] J. J. Quirk. An alternative to unstructured grids for computing gas dynamics flows around arbitrarily complex two-dimensional bodies. *Computers Fluids*, 23:125–142, (1994).

[27] U. Specht. *Numerische Simulation mechnanischer Wellen an Fluid-Festkörper-Mediengrenzen.* Fortsch.-Ber. VDI Reihe 7 Nr. 398, VDU Verlag, Düsseldorf, (2000).

[28] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics.* Springer-Verlag, Berlin, Heidelberg, 2nd edition, (1999).

[29] T.L. Warren and M.J. Forrestal. Effects of strain hardening and strain-rate sensitivity on the penetration of aluminum targets with spherical-nosed rods. *Int. J. Solids Structures*, 35:3737–3753, (1998).