

# Tetrahedral Embedded Boundary Methods for Accurate and Flexible Adaptive Fluids

Christopher Batty<sup>1</sup>, Stefan Xenos<sup>2</sup>, and Ben Houston<sup>2</sup>

<sup>1</sup>University of British Columbia, Canada

<sup>2</sup>Exocortex Technologies, Inc., Canada

---

## Abstract

*When simulating fluids, tetrahedral methods provide flexibility and ease of adaptivity that Cartesian grids find difficult to match. However, this approach has so far been limited by two conflicting requirements. First, accurate simulation requires quality Delaunay meshes and the use of circumcentric pressures. Second, meshes must align with potentially complex moving surfaces and boundaries, necessitating continuous remeshing. Unfortunately, sacrificing mesh quality in favour of speed yields inaccurate velocities and simulation artifacts. We describe how to eliminate the boundary-matching constraint by adapting recent embedded boundary techniques to tetrahedra, so that neither air nor solid boundaries need to align with mesh geometry. This enables the use of high quality, arbitrarily graded, non-conforming Delaunay meshes, which are simpler and faster to generate. Temporal coherence can also be exploited by reusing meshes over adjacent timesteps to further reduce meshing costs. Lastly, our free surface boundary condition eliminates the spurious currents that previous methods exhibited for slow or static scenarios. We provide several examples demonstrating that our efficient tetrahedral embedded boundary method can substantially increase the flexibility and accuracy of adaptive Eulerian fluid simulation.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

---

## 1. Introduction

Adaptivity is a key feature for the efficient animation of fluids because it can focus computational resources on visually significant details. Examples include regions of greater vorticity, inside the viewing frustum, and flow near free surfaces and solid objects [LGF04, KFCO06, KIC06]. Tetrahedral meshes, octrees, elongated Cartesian cells [IGLF06], and general nested Cartesian (AMR) grids (eg. [BO84]) have all been used for this purpose. These methods' primary drawback is that domain boundaries must align with the underlying voxels or tetrahedra, which either limits the variety of boundaries that can be simulated or greatly increases the expense and difficulty of high quality mesh generation.

Embedded boundary methods that account for sub-grid geometry of free surfaces and solid objects on Cartesian grids have recently gained acceptance in graphics [ENGF03, BBB07], by offering improved resolution of irregular shapes with only minor modifications to existing solvers. These

methods yield equivalent or better results than conforming tetrahedral meshes, because the regularity of the grid structure affords more accurate finite difference operators.

In terms of adaptivity however, tetrahedra offer substantial advantages over Cartesian grids. Octrees and nested grids provide only discrete jumps in grid size, which artificially prevent smooth grading between resolutions. At these "T-junction" faces where jumps in resolution occur, a single lower resolution face is shared by four or more higher resolution faces. Besides added implementation complexity, it is so far unclear how to apply embedded methods across such faces; at present grading must be disallowed along air and solid boundaries. T-junctions also require care to avoid losing accuracy and causing simulation artifacts [LGF04, CFL\*07, LFO05]. In contrast, tetrahedral meshes need no T-junctions and allow elements of arbitrary size, thus providing greater flexibility.

Our work seeks to hybridize tetrahedral methods with em-

bedded boundary techniques. In this manner, we achieve the high quality results associated with embedded boundary Cartesian grid methods, while simultaneously providing the best combination of speed, flexibility, and adaptivity of state-of-the-art tetrahedral schemes. Our specific contributions are the following:

**Embedded Free Surfaces:** We extend the ghost fluid sub-grid free surface pressure projection [ENGF03] to tetrahedra, which improves the accuracy of free surfaces and removes their boundary alignment restriction.

**Embedded Solid Boundaries:** We adapt the sub-grid solid boundary pressure projection [BBB07] to tetrahedra, to provide accurate support for non-mesh-aligned solids.

The elimination of boundary alignment constraints provides several vital benefits:

**Fast High Quality Delaunay Meshing:** We can exploit high quality *non-conforming* Delaunay meshes with circumcentric pressures. Such meshes guarantee consistent finite difference estimates of at least first order, and are faster and easier to generate, particularly for complex domains.

**Improved Surface Motion:** Enforcing the free surface boundary condition precisely at the air-liquid interface improves the resulting fluid motion, especially for slow or still fluid.

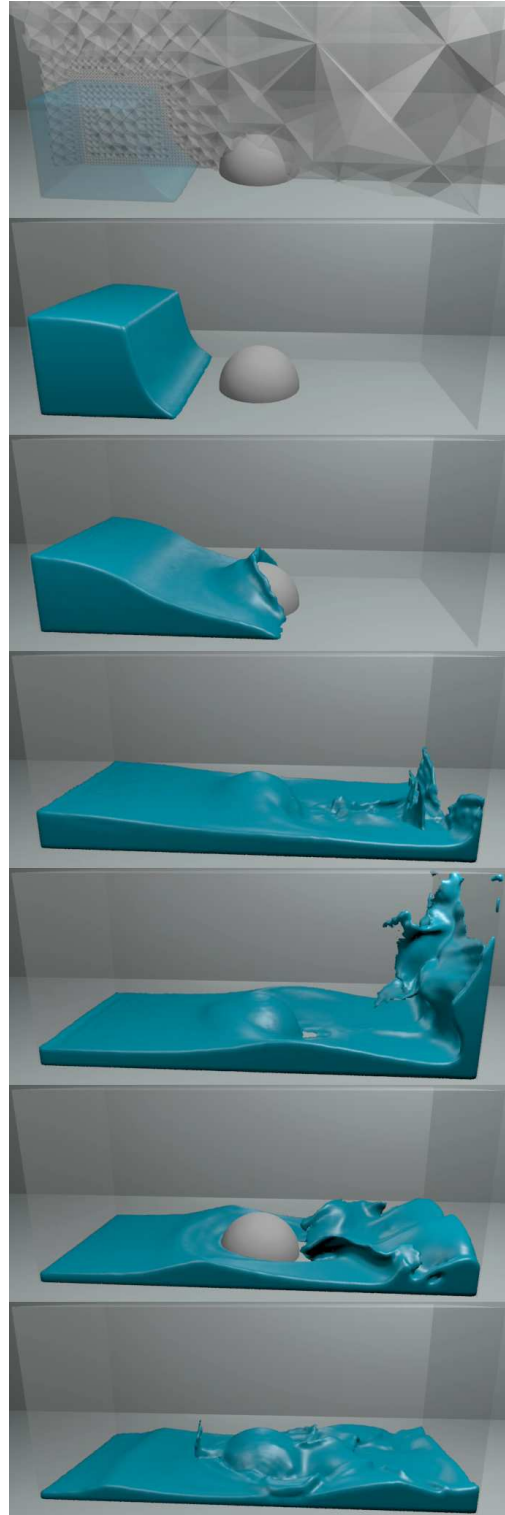
**Reduced Remeshing Frequency:** Since adaptivity requirements typically exhibit high temporal coherence, we can reuse entire meshes over several timesteps.

**Increased Flexibility:** It becomes possible to easily grade along air or solid boundaries without simulation artifacts or more complex mesh generation, allowing effectively arbitrary mesh adaptivity.

## 2. Related Work

### 2.1. Adaptive Fluids and Tetrahedral Meshes

Tetrahedral meshes have become popular within the computer graphics fluid simulation community because they provide straightforward adaptivity and until recently were the only simple method for accurately incorporating non-axis-aligned boundary geometry into Eulerian schemes. Feldman et al. [FOK05] first mapped the basic Stable Fluids method [Sta99] to tetrahedra, using finite volume techniques (as in the octree work of Losasso et al. [LGF04]). This was extended to mildly deforming meshes [FOKG05], and then to rigid and deformable body coupling [KFCO06, CGFO06], by remeshing on every timestep. Klingner et al. [KFCO06] also specifically highlighted the adaptivity benefits of tetrahedra, where previous work had focused solely on matching irregular boundaries. Wendt et al. [WBOL07] used a slightly different discretization and a level set method to include viscosity and non-conforming free surfaces. Chentanez et al. [CFL\*07] presented an efficient algebraic multi-grid method, along with conforming free surfaces through



**Figure 1:** Our method yields quality results on a dam break example without matching air or solid boundaries. The top frame shows a cutaway of the mesh.

the use of isosurface stuffing for faster remeshing [LS07]. This method provides fast adaptive meshing with guaranteed angle bounds, at the cost of the Delaunay property. An alternate circulation-based approach was advocated by Elcott et al. [ETK\*07], and Mullen et al. [MCP\*09] demonstrated energy-preservation, using an Eulerian advection scheme combined with a unified, fully non-linear solver. Sin et al. [SBH09] recently presented a dual approach, solving the pressure projection on unstructured Voronoi meshes clipped against boundaries.

Tetrahedral methods achieve their best results when the meshes used possess the Delaunay property and pressures are stored at circumcentres; this ensures convex dual elements and consistent first order accurate finite difference approximations (see §4). If the tetrahedralization aligns with a particular domain boundary, it is referred to as a conforming Delaunay tetrahedralization [CSECY04]. Such meshes are generally difficult to compute, although allowing flexibility in the surface by adding Steiner points or approximating the boundary simplifies matters [CSECY04, ACSYD05]. Nonetheless, it remains substantially slower and more difficult than either *non-conforming* Delaunay meshing [WT08] or conforming *non-Delaunay* meshing [LS07]. For example, Alliez et al. [ACSYD05] performed Delaunay meshing up to 50 times during their iterative variational scheme, while requiring heuristic vertex jittering to discourage slivers near boundaries. Klingner et al. [KFCO06] found that the same method required 5 minutes per frame for 500K tetrahedra. Recently Tournois et al. [TWAD09] interleaved Delaunay refinement with optimization and improved the boundary treatment to produce even higher quality meshes, but had meshing times in the tens of minutes for 120K tetrahedra and still provided no guarantees against slivers.

Because guaranteed quality conforming Delaunay meshing remains challenging, research in computational physics has modified the finite volume method in hopes of achieving good accuracy on more general meshes. Perot and Subramanian [PS07] used an exact calculus discretization with improved interpolation to handle non-Delaunay meshes. Similarly, deferred correction approaches use geometric arguments to correct directional errors in gradients [TAL09]. Though effective, such methods are more expensive and complex, and it remains unclear how to enforce air and solid boundaries simultaneously. In effect, these approaches seek to relax the mesh quality requirements, whereas we will relax the boundary alignment requirement.

## 2.2. Embedded Boundaries

Two techniques have allowed embedded (ie. non-mesh-aligned) boundaries to be supported on Cartesian grids with relative ease. The first is the second order accurate ghost fluid free surface pressure condition of Enright et al. [ENGF03], which greatly improves the behaviour of liquid surfaces by modifying the finite difference gradient stencil.

The second is the sub-grid solid boundaries approach which achieves high quality solid interaction on non-conforming grids by adding face weights to the divergence stencil. Roble et al. [RZF05] first derived the latter idea in a finite volume manner, suggesting the use of face *area* weights. Batty et al. [BBB07] developed a related variational method to handle moving boundaries and full rigid body coupling. For the case of static objects this yields essentially the same discretization, except for the use of face *volume* weights. However, a recent paper by Ng et al. [NMG09] has shown that face area weights should be preferred, as this provides second order accuracy in pressure. This solid boundary discretization can also be combined with the free surface condition above, as demonstrated by Batty & Bridson [BB08]. With the addition of a few simple weights to the standard finite difference stencils, these two embedded boundary approaches naturally extend the MAC scheme to irregular domains.

There are also numerous embedded boundary methods in the computational physics literature. These include immersed boundary methods [Pes02], cut-cell methods (eg. [MKLU05, LKMU05, SBCL06]), ghost fluid variants (eg. [BF08, MDB\*08]) and many more. Our focus on the works of Enright et al. [ENGF03] and Batty et al. [BBB07] is motivated by these methods' simplicity and effectiveness. Both also have longstanding roots in computational fluid dynamics. Enforcing the Dirichlet pressure condition at the sub-grid free surface position was first suggested by Chan and Street [CS70], albeit in rudimentary form. Similarly, the basic face area-weighting scheme for static embedded solids can be traced to work by Purvis and Burkhalter [PB79].

While the majority of embedded boundary schemes use Cartesian grids, a few have recently highlighted the benefits of an underlying simplex mesh, as we do here (eg. [LCC\*07, FD07]). However, they employ fundamentally different and more complex cut-cell schemes than those we propose.

Another loosely related area of research is embedding methods for simulating deformable objects. These approaches embed a more detailed surface mesh into a lower resolution simulation mesh to reduce computational costs [FvdPT97, CGC\*02]. This family was extended to handle extreme plastic deformations by Wojtan & Turk [WT08], with a simple but effective remeshing strategy that generates high quality non-conforming Delaunay meshes from an adaptive BCC lattice. Because the simulation mesh need not conform to the object's surface geometry, they can guarantee very high quality elements while apparently remeshing one to two orders of magnitude faster than state-of-the-art conforming, non-Delaunay approaches. Our approach applies essentially the same premise to fluid simulation.

## 3. System Overview

Our approach builds most directly on the work of Chentanez et al. [CFL\*07], to which we refer the reader for implementation details and pointers to prior work. We will simply

highlight the differences unique to our approach. The specific steps in our algorithm are:

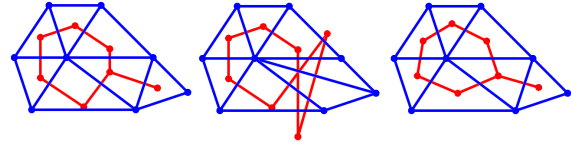
1. Advect the liquid surface, with a standard surface tracker.
2. *Optional*: Remesh to generate a new tetrahedral mesh enveloping the liquid domain (§4). We use a quality non-conforming adaptive Delaunay BCC lattice mesh generator in place of isosurface stuffing.
3. Apply semi-Lagrangian advection to mesh velocities. If remeshing occurred, this transfers velocities to the new mesh as usual (without extra smoothing [FOKG05]).
4. Add external forces.
5. Apply our tetrahedral embedded boundary pressure projection (§5 & §6). This replaces the standard conforming tetrahedral pressure projection [FOK05].

We have not applied any volume preservation strategies in our system. We solve the pressure projection with the conjugate gradient method, but expect that the algebraic multigrid (AMG) method proposed by Chentanez et al. [CFL\*07] would provide a useful speed-up. A final key difference is that we store pressures at tetrahedra circumcentres, and velocities at face circumcentres, as in most earlier work (eg. [KFCO06]).

#### 4. High Quality Meshes

Basic staggered mesh methods for tetrahedra require a Delaunay mesh with pressures stored at tetrahedra circumcentres [KFCO06]. These "covolume" meshes are a natural generalization of classic staggered grid (MAC) schemes [Nic92, NW97, ZSP02]. Connecting neighbouring circumcentres on a primal Delaunay mesh yields its circumcentric or Voronoi dual, a valid Voronoi tessellation possessing two very useful properties, as discussed by Perot and Subramaniam [PS07]. First, local orthogonality refers to the fact that the line through neighbouring pressure locations (circumcentres) is perpendicular to the shared face of their tetrahedra, and thus parallel to the velocity sample stored at the face (Figure 3, left). When we apply the standard fluid velocity update ( $\vec{u} = \vec{u}^* - \frac{\Delta t}{\rho} \nabla p$ ) the pressure gradient corrects for divergence in the appropriate velocity component, ensuring that the pressure gradient estimate is at least first order accurate. Perot and Subramaniam further note that while strict second order accuracy requires that dual edge midpoints between neighbouring circumcentres coincide exactly with primal face circumcentres (as for uniform grids), second order accurate convergence is frequently observed in practice for fairly well-behaved meshes. The second useful property of these meshes is the convexity and non-self-intersection of the dual mesh, which avoids dual elements with conceptually negative volumes and allows generalized barycentric interpolation of velocities [WSHD04, ETK\*07, KFCO06]. Klingner et al. [KFCO06] also used properties of Delaunay meshes to simplify these interpolations.

Relaxing these two constraints would simplify mesh gen-



**Figure 3:** Different choices of triangulations (blue) and dual meshes (red) in 2D. From left to right: 1) Delaunay triangulation with circumcentric dual. 2) Non-Delaunay triangulation with circumcentric dual. The dual mesh is self-intersecting. 3) Delaunay triangulation with barycentric dual. Primal/dual edge pairs lack orthogonality. Based on figures by Perot & Subramaniam [PS07].

eration, but unfortunately this causes problems for simulation. The dual mesh generated by connecting circumcentres of non-Delaunay meshes is self-intersecting, though it retains orthogonality (Figure 3, centre). Conversely, the barycentric or median dual generated by connecting mesh barycentres is comprised of valid convex elements, but sacrifices the crucial orthogonality property (Figure 3, right). This latter situation gives rise to the linear inconsistency mentioned by Chentanez et al. [CFL\*07] and partly accounts for the artifacts they observed in slow-moving fluids.

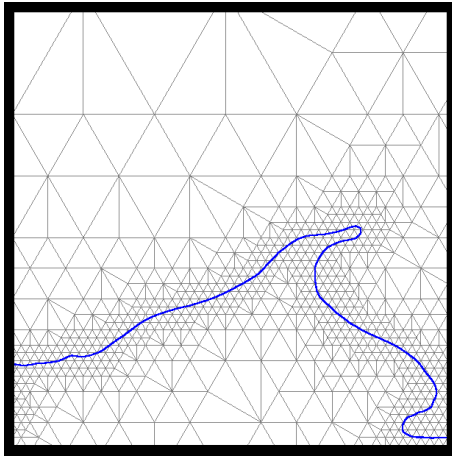
As an aside, we note that in the original octree method of Losasso et al. [LGF04] pressure gradients across T-junctions lose accuracy because they too give up orthogonality. This method places velocity samples on each small face incident on a T-junction, and estimates a non-orthogonal pressure gradient between the associated large and small cell pressures. However, Losasso et al. [LFO05] later corrected this to achieve second order accuracy with a slight modification. The solution was to use just a single velocity sample on the entire T-junction face and construct a fully orthogonal pressure gradient as an area-weighted combination of the small face pressure gradients. This further illustrates the importance of retaining orthogonality.

Instead of sacrificing mesh orthogonality or convexity, we will use embedded boundary methods to eliminate the restriction that meshes align with domain geometry. This has several practical consequences with respect to the meshes used for simulation. First, it allows us to more aggressively exploit temporal coherence. For example, in the case of a moving object the user might desire high resolution elements around the object to capture small flow details. With conforming methods, accommodating even slight motions of an object can require substantial changes to the mesh to maintain high quality. This holds true even if the mesher is warm-started with the previous mesh, as done by Klingner et al. [KFCO06]. With non-conforming methods, we remesh only when our mesh adaptivity criterion ceases to be satisfied (eg. in Figure 4 we might remesh when the liquid surface leaves the surrounding region of highest refinement). By tailoring such criteria appropriately, we can balance the



	Static test	Graded Boundaries	Remesh frequency	Remesh speed	Mesh quality
Embedded regular grids	<b>Pass</b>	No	N/A	N/A	<b>High</b>
Embedded octrees	<b>Pass</b>	No	<b>Low</b>	<b>Fast</b>	<b>High</b>
Conforming Delaunay	Fail	<b>Yes</b>	High	Slow	Low/Moderate
Chentanez et al. [CFL*07]	Fail	No	High	<b>Fast</b>	Low
Embedded Delaunay (ours)	<b>Pass</b>	<b>Yes</b>	<b>Low</b>	<b>Fast</b>	<b>High</b>

**Figure 2:** A qualitative comparison of different methods and simulation meshes. "Mesh quality" encompasses orthogonality, the Delaunay property, and angle bounds. "Static test" refers to still fluid where pressure should precisely balance gravity. Our embedded Delaunay method (using an underlying adaptive BCC lattice) provides a good combination of accuracy, speed, flexibility, and adaptivity.



**Figure 4:** An embedded fluid simulation on a high quality adaptive non-conforming lattice mesh. Since we need not match boundaries, we can reuse consecutive meshes to save on meshing costs.

benefits of frequent adaptivity updates against the costs of remeshing.

Second, because the true boundaries are allowed to cut through the mesh arbitrarily, mesh grading can occur even along free surfaces and solid boundaries. In contrast, the complexity of correctly handling octree T-junctions makes it unclear how to simultaneously combine them with embedded boundaries; a uniform resolution is thus required along all boundaries. Similarly, though isosurface stuffing is incredibly fast, it only provides effective angle guarantees if adaptivity is restricted to the interior of the mesh. Consider Figure 4, where high resolution is desired only near the liquid surface. If all boundary elements needed to be uniformly sized, many would be wasted resolving unnecessary details along the bottom wall.

Third and most vitally, using embedded boundaries accelerates and simplifies remeshing, and lets us easily guarantee that our simulator is provided with high quality Delaunay meshes with the properties necessary for maximum

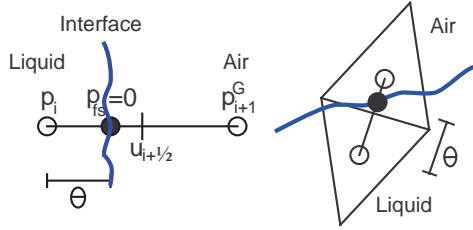
accuracy. This is particularly important because poor mesh quality and high meshing costs are two major drawbacks of tetrahedra as compared to grids. Any Delaunay mesh that fully envelops the domain may be used with our scheme. For maximum remeshing speed and regularity, we recommend the unmodified octree-graded BCC lattice as in the work of Wojtan and Turk [WT08]. As they pointed out, this results in high quality meshing that is effectively free compared to the remaining steps of the algorithm. Furthermore, its regularity may be exploited to accelerate point-location and save memory [CFL\*07]. Our single-threaded implementation generates over 500K tetrahedra/second for meshes up to three million elements.

## 5. Embedded Free Surfaces on Tetrahedra

The free surface (Dirichlet) pressure boundary condition presented by Enright et al. [ENGF03] allows free surfaces to lie between rather than strictly at grid cell centres. The discretization for a velocity update due to pressure in one dimension for a particular face at the boundary between liquid and air is:

$$u = u^* - \frac{\Delta t}{\rho} \cdot \frac{p_{fs} - p_i}{\theta \Delta x} \quad (1)$$

In this expression  $u$  is the final divergence-free face velocity,  $u^*$  is the velocity before projection,  $\Delta t$  is the time step,  $\rho$  is the liquid density,  $\Delta x$  is the grid cell size,  $p_i$  is the pressure variable in the liquid cell,  $p_{fs}$  is the specified boundary value for the free surface pressure (typically zero or standard atmosphere), and finally  $\theta$  is the fractional distance from the last internal pressure sample to the sub-grid liquid surface. In situations where  $\theta$  is at or near zero, it is perturbed to be slightly positive [GFCK02, Bri08].  $\theta$  is typically estimated from signed distance values stored at pressure samples, but more generally is extracted from the user's choice of surface tracker. Figure 5, left, illustrates this situation. This can most directly be understood as a shortened finite difference estimate of the pressure gradient from the last submerged pressure cell to the free surface position. Equivalently, it can be derived by placing a ghost pressure  $p_{i+1}^G$  in the adjacent air cell such that the linearly interpolated pressure value crosses  $p_{fs}$  precisely at the sub-grid interface location. In ei-



**Figure 5:** Left: A 1D example of the method of Enright et al. [ENGF03] for capturing the free surface position between pressure samples. Right: The same idea applied to an unstructured triangle mesh.

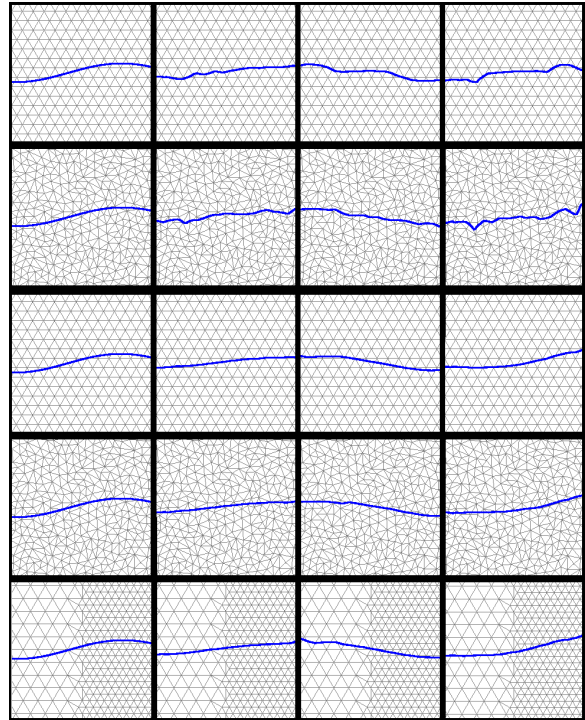
ther case, this expression for the velocity update is plugged into the divergence constraint  $\nabla \cdot \vec{u} = 0$ , yielding a symmetric positive definite Poisson system and a second order accurate pressure solution for smooth boundaries. Enright et al. [ENGF03] showed that this drastically improves the behaviour of free surfaces on regular grids.

This method can be readily adapted to tetrahedral meshes with minimal modification. If the interface lies between two tetrahedra circumcentres (ie. where pressure is stored), we replace  $\Delta x$  with the distance between the circumcentres and modify  $\theta$  to be an estimate of the fractional position of the interface along the line joining the two circumcentres (Figure 5, right). As in the Cartesian grid case this yields much improved small-scale behaviour, and eliminates the tetrahedral analog of free surface stairstep artifacts (See Figure 6). (This is the "aliasing" noted by Wendt et al. [WBOL07].)

This discretization can beneficially be applied even to existing conforming tetrahedral methods. The current standard approach to enforcing the Dirichlet boundary condition is to use a mirrored ghost pressure set to  $p_{fs}$  on the outside of the appropriate face [FOK05, CFL\*07, SBH09]. Considering again Figure 5, right, this incorrectly sets the pressure value at the exterior ghost point, rather than precisely at the face where the liquid interface lies. This is likely the second source of error which prevented hydrostatic balance (where pressure precisely cancels gravity forces) from being achieved by Feldman [Fel07] and Chentanez et al. [CFL\*07]. Our method easily corrects this; the denominator in the pressure gradient calculation should be the perpendicular distance from the circumcentre to the surface face, rather than twice that value.

## 6. Embedded Solid Boundaries on Tetrahedra

The variational projection technique presented by Batty et al. [BBB07] allows for sub-grid resolution of solid (Neumann) boundaries, by modifying the divergence operator with weights to account for partial cells. Expressed in a finite

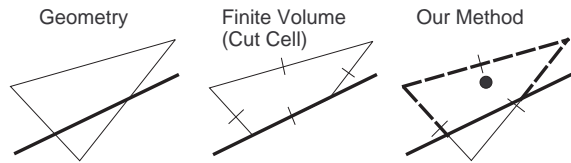


**Figure 6: Slashing tank** Top row: The standard free surface approach with non-conforming meshes yields bumpy artifacts on the scale of one triangle. Second row: The same approach on an irregular mesh illustrates the mesh-dependency of these artifacts. Third row: Embedded free surfaces with a regular mesh yields smooth sloshing without artifacts. Fourth row: Embedded free surfaces with an irregular mesh yields behaviour consistent with the regular mesh case, demonstrating mesh independence. Bottom row: Grading across free surfaces introduces no errors.

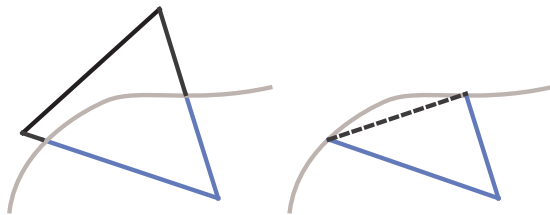
volume-like form their divergence operator is the following:

$$\nabla \cdot \vec{u} \approx \sum_{i \in \text{faces}} w_i (\vec{u}_i \cdot \vec{n}_i) \quad (2)$$

where the set of faces are those of the original cell.  $\vec{u}_i$  and  $\vec{n}_i$  are the fluid velocity and normal at the face, respectively. A few weight choices are reasonable, but following Ng et al. [NMG09] we choose the weights  $w_i$  to be the partial non-solid area of the face. As noted by Roble et al. [RZF05], this choice yields a slightly simplified cut-cell finite volume discretization, as illustrated in Figure 7. The simplification is that the standard cut-cell finite volume approach (eg. [SBCL06]) would interpolate velocity samples to the mid-points of the truncated faces and generate new faces along the boundary, arriving at a non-symmetric linear system. In contrast, our chosen approach weights the original faces by their partial areas, but creates no new faces and leaves the velocity samples at their original positions. This ensures that



**Figure 7:** Left: A 2D example of a solid boundary (thick line) cutting through a triangular element. Centre: A standard finite volume discretization clips the triangle and re-locates the velocity samples, requiring complex interpolation to accurately determine pressure gradients. Right: Our embedded boundary scheme uses finite volume face area weights (dashed lines) but leaves velocity positions unchanged, thereby retaining local orthogonality.

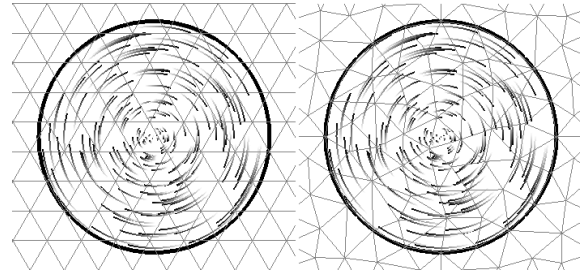


**Figure 8:** Left: A 2D example in which one triangle face is cut off by the solid boundary (curved line) and is assigned a zero area weight. Right: The approximated physical boundary (dashed) has a different average normal than the original triangle face.

we retain both the accuracy provided by the local orthogonality property and the symmetric positive definiteness of the standard discretization, without requiring explicit clipping of geometry or complex interpolation schemes to compute orthogonal pressure gradients. This approach can be applied to tetrahedra by simply estimating partial tetrahedra face areas, and extensions to one- and two-way solid coupling are also straightforward, following Batty et al. [BBB07].

In our implementation we store signed distance values for solid boundaries on the vertices of the tetrahedra. This allows face area fractions to be estimated with simple 2D marching triangles cases, and eliminates the need for mesh-based geometric clipping. (In exchange, this gives up a slight amount of resolution, since flows through cracks below the mesh resolution are disallowed.)

For use during advection, a full velocity for each tetrahedron's circumcentre is typically reconstructed from the normal components on its faces using a least-squares fit [ETK\*07, KFCO06, PVW06]. Given these circumcentre velocities, we perform interpolation and advection exactly following Chentanez et al. [CFL\*07]. However, in our scheme when a face is clipped entirely it has zero associated area weight (Figure 8, left). It therefore does not participate in the



**Figure 9:** Our non-conforming embedded solid boundary method (left) compared against the standard conforming method (right), for a low resolution rotating disk of fluid visualized with streamlines seeded at the same points. The two are essentially indistinguishable, illustrating that our method accurately handles boundaries and reconstructs the free slip velocity even near zero-area faces.

pressure solve and is not assigned a valid velocity. Naïvely using a zero velocity (or more generally, the wall velocity) for this face frequently destroys free-slip in the reconstruction. This is because the missing face's normal doesn't necessarily match the solid boundary normal (Figure 8, right), so it may actually require a non-zero velocity component to be consistent with the fluid velocity.

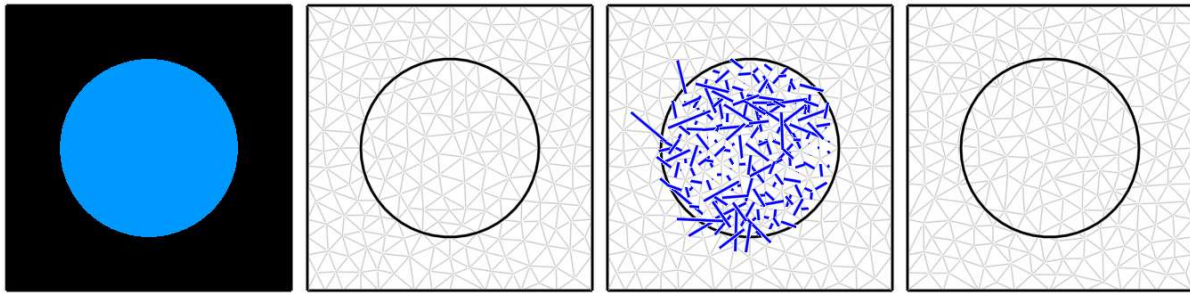
We handle this by simply dropping the zero-area face's row from the least-squares solve. The use of nodal signed distances to compute weights ensures that if a quality tetrahedron is not entirely solid, it can have only one such zero-area face, leaving three valid velocity components with linearly independent normal vectors. Three independent components are always sufficient to reconstruct a three-dimensional velocity vector, so the linear system is never underdetermined. Furthermore, the divergence-free condition ensures that this velocity is in fact unique [ETK\*07], and hence no information is lost in dropping the face. This approach robustly reconstructs the free slip fluid velocity.

Conveniently, the free surface and solid embedded boundary methods above are entirely complementary, as illustrated by Batty & Bridson [BB08]. Perhaps the simplest interpretation is that the free surface condition modifies the gradient operator near air, while the variational pressure projection modifies the divergence operator near solids. By plugging the modified velocity update into the modified divergence operator, we get a method that straightforwardly handles both boundaries without special cases.

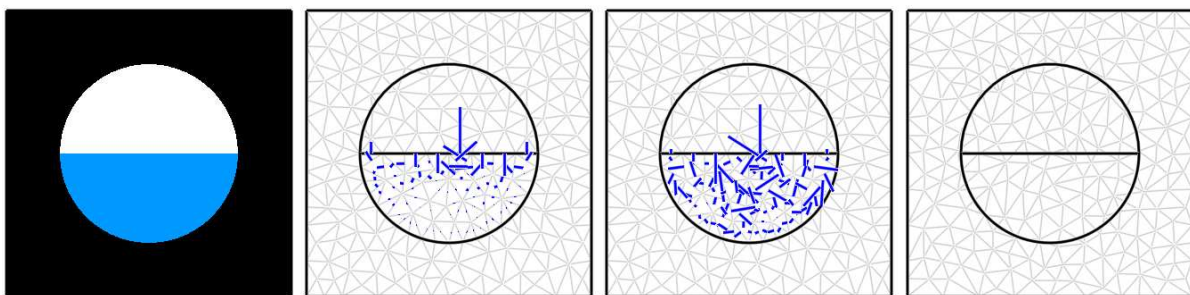
## 7. Results

We focus on two-dimensional examples to emphasize the relationship between the simulation and the underlying mesh, however we stress that all of our results extend straightforwardly to 3D. Furthermore, while we do not compare in detail the speed of our method to that of Chentanez et al.





**Figure 10:** From left to right: 1) Input geometry for a closed hydrostatic scene under uniform gravity. 2) A conforming Delaunay mesh with circumcentric pressures finds the exact solution (no motion.) 3) The same mesh using barycentric pressures yields substantial spurious velocities. 4) Using our embedded solid boundary method, the exact solution is found on a non-conforming Delaunay mesh with circumcentric pressures.



**Figure 11:** From left to right: 1) Input geometry for a free surface hydrostatic test under uniform gravity. 2) Standard free surface boundary conditions introduce spurious velocities near the surface, despite using a conforming circumcentric Delaunay mesh. 3) The use of barycentric pressures with the same mesh and boundary conditions worsens the errors. 4) A non-conforming Delaunay mesh with circumcentric pressures using our embedded solid and free surface boundary conditions finds the exact solution.

[CFL\*07], we expect that optimized implementations will exhibit approximately the same speed, assuming we replace our CG routine with AMG and remesh continuously for both. Our reasoning is that both our proposed method and isosurface stuffing first build a (potentially adaptive) BCC lattice. Isosurface stuffing then queries the boundary isosurface in order to stuff the tetrahedra inside it, whereas our method instead uses these queries to determine the solid and free surface weights used by the pressure discretization. The methods can otherwise be made nearly identical, so we will aim to demonstrate the key accuracy and flexibility benefits of our method.

Figures 10 and 11 illustrate the ability of our projection method to correctly handle the hydrostatic scenario, consisting of a vertical gravity force that should be precisely cancelled by the resulting pressure gradient, for a completely enclosed case and a free surface case. We compare against both circumcentric and barycentric conforming Delaunay meshes, confirming that circumcentres are preferred. Note that only our scheme is able to achieve the correct cancella-

tion in the difficult free surface scenario, and to our knowledge it is the first tetrahedral method in computer graphics able to do so.

To illustrate the improved motion provided by embedded free surfaces, Figure 6 compares several versions of a slow-moving sloshing scenario. The basic non-conforming method yields bumpy artifacts due to its inability to "see" waves below the scale of one triangle. These artifacts are also highly mesh-dependent; the irregular mesh produces different (and substantially worse) motion. The same example using embedded free surfaces yield smoother and more consistent results regardless of the underlying mesh, even when grading is performed across the free surface itself. The associated animations demonstrate that the embedded approach also exhibits less artificial damping.

An example in our accompanying video compares a 2D free surface flow simulated with no remeshing, continuous remeshing, and intermittent remeshing every 5 time steps, on a high quality non-conforming adaptive lattice, like that



in Figure 4. The non-remeshed example is chosen to have approximately the same number of triangles as an early frame of the adaptive case. The low resolution of the non-remeshed example performs comparatively poorly, however both intermittent and continuous remeshing provide much better and qualitatively similar results despite the former expending one-fifth as much effort on remeshing. Real applications might use more elaborate adaptivity criteria, but this already illustrates the flexibility and power of combining adaptivity with our embedding scheme: it enables an explicit trade-off between accuracy and remeshing costs. This also suggests an interesting potential optimization: at the cost of slightly lagged adaptivity updates, remeshing could be performed in parallel such that the simulator proceeds with the current mesh until a new one becomes available. This holds out the possibility of entirely hiding the costs of remeshing.

To illustrate that our non-conforming solid boundary approach gives qualitatively identical results to a conforming scheme and fully reconstructs free slip velocities near walls, we compare frames from a simple rotation inside a disk-shaped 2D domain. Visualized with streamlines in Figure 9, it is clear that the two methods are almost perfectly consistent. The accompanying video makes a similar comparison against a naïve rasterized non-conforming approach, which exhibits erroneous damping and inaccurate motion near walls.

Our video also includes a complex 2D animation consisting of liquid drops splashing against curved and angled solid boundaries. This emphasizes that our two boundary conditions can be used together without artifacts. Lastly, Figure 1 shows a 3D breaking dam example similar to one by Chentanez et al. [CFL\*07]. The simulation used adaptive BCC lattice meshes ranging between 400K and 1.1M non-conforming tetrahedra, yet achieves accurate and smooth liquid motion. Computation times averaged 31 seconds/frame (about 40% of which is our single-threaded particle level set surface tracker) on a 4-core Intel i7 860. The simulator was parallelized using Intel's Threading Building Blocks library.

## 8. Conclusions & Future Work

We have demonstrated a few simple modifications to tetrahedral mesh fluid simulation that can improve its accuracy, flexibility, and speed. The use of non-conforming Delaunay meshes together with embedded boundary techniques improves the liquid behaviour in many scenarios while substantially reducing the frequency, complexity, and costs of high quality meshing. This has the potential to make tetrahedral schemes more competitive with regular grid methods, while retaining the vital advantage of adaptivity.

There are several directions for future work. Studying the accuracy and convergence of our tetrahedral embedded boundary techniques would be valuable. A potential drawback of circumcentric pressures is that they are not necessarily

contained in their associated tetrahedra, and though orthogonality ensures first order accuracy, this might impact the magnitude of the approximation error on low quality meshes. Research into so-called well-centred meshes might prove useful in this respect [VHG08]. Relatedly, our method should adapt seamlessly onto unstructured Voronoi meshes (eg. [SBH09]), where Voronoi sites are guaranteed to be inside their associated cells. Extending unstructured meshes to support free surface viscosity, viscoelasticity, and surface tension are other interesting directions. Finally, given the ubiquity of Poisson problems in graphics research, these embedded boundary schemes could likely benefit applications outside of fluid animation.

**Acknowledgements** We thank Evan VanderZee and Dr. Anil Hirani for sharing their expertise on well-centred meshes, and Dr. Robert Bridson for his encouragement, advice, and public domain code. We also thank the anonymous reviewers whose comments lead to substantial improvements of this paper.

## References

- [ACSYD05] ALLIEZ P., COHEN-STEINER D., YVINEC M., DESBRUN M.: Variational tetrahedral meshing. In *Proc. SIGGRAPH* (2005), p. 10.
- [BB08] BATTY C., BRIDSON R.: Accurate viscous free surfaces for buckling, coiling, and rotating liquids. In *Proc. Symposium on Computer Animation 2008* (2008), pp. 219–228.
- [BB07] BATTY C., BERTAILS F., BRIDSON R.: A fast variational framework for accurate solid-fluid coupling. In *Proc. SIGGRAPH* (2007), p. 100.
- [BF08] BERTHELSEN P., FALTINSEN O.: A local directional ghost cell approach for incompressible viscous flow problems with irregular boundaries. *J. Comp. Phys.* 227 (2008), 4354–5397.
- [B084] BERGER M. J., OLIGER J.: Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comp. Phys.* 53, 3 (1984), 484–512.
- [Bri08] BRIDSON R.: *Fluid Simulation for Computer Graphics*. A.K. Peters, 2008.
- [CFL\*07] CHENTANEZ N., FELDMAN B. E., LABELLE F., O'BRIEN J. F., SHEWCHUK J. R.: Liquid simulation on lattice-based tetrahedral meshes. In *Proc. Symposium on Computer Animation 2007* (2007), pp. 219–228.
- [CGC\*02] CAPELL S., GREEN S., CURLESS B., DUCHAMP T., POPOVIĆ Z.: A multiresolution framework for dynamic deformations. In *Proc. Symposium on Computer Animation 2002* (2002), pp. 41–47.
- [CGFO06] CHENTANEZ N., GOKTEKIN T. G., FELDMAN B. E., O'BRIEN J. F.: Simultaneous coupling of fluids and deformable bodies. In *Proc. Symposium on Computer Animation 2006* (2006), pp. 83–89.
- [CS70] CHAN R. K.-C., STREET R. L.: A computer study of finite amplitude water waves. *J. Comp. Phys.* 6 (1970), 68–94.
- [CSECY04] COHEN-STEINER D., ÉRIC COLIN DE VERDIÈRE, YVINEC M.: Conforming Delaunay triangulations in 3D. *Computational Geometry* 28, 2-3 (2004), 217–233.
- [ENGF03] ENRIGHT D., NGUYEN D., GIBOU F., FEDKIW R.: Using the particle level set method and a second order accurate pressure boundary condition for free surface flows. In *Proc. of the 4th ASME-JSME Joint Fluids Engineering Conference* (2003).

- [ETK\*07] ELCOTT S., TONG Y., KANSO E., SCHRÖDER P., DESBRUN M.: Stable, circulation-preserving, simplicial fluids. *ACM Trans. Graph.* 26, 1 (2007), 4.
- [FD07] FIDKOWSKI K. J., DARMOFAL D. L.: A triangular cut-cell adaptive method for high-order discretizations of the compressible navier-stokes equations. *J. Comp. Phys.* 225 (2007), 1653–1672.
- [Fel07] FELDMAN B.: *Fluid Animation from Simulation on Tetrahedral Meshes*. PhD thesis, EECS Dept, UC Berkeley, 2007.
- [FOK05] FELDMAN B. E., O'BRIEN J. F., KLINGNER B. M.: Animating gases with hybrid meshes. In *Proc. SIGGRAPH* (2005), pp. 904–909.
- [FOKG05] FELDMAN B. E., O'BRIEN J. F., KLINGNER B. M., GOKTEKIN T. G.: Fluids in deforming meshes. In *Proc. Symposium on Computer Animation 2005* (2005), pp. 255–259.
- [FvdPT97] FALOUTSOS P., VAN DE PANNE M., TERZOPOULOS D.: Dynamic free-form deformations for animation synthesis. *IEEE TVCG* 3, 3 (1997), 201–214.
- [GFCK02] GIBOU F., FEDKIW R. P., CHENG L.-T., KANG M.: A second-order-accurate symmetric discretization of the Poisson equation on irregular domains. *J. Comp. Phys.* 176, 1 (2002), 205–227.
- [IGLF06] IRVING G., GUENDELMAN E., LOSASSO F., FEDKIW R.: Efficient simulation of large bodies of water by coupling two and three dimensional techniques. In *Proc SIGGRAPH* (2006), pp. 805–811.
- [KFCO06] KLINGNER B. M., FELDMAN B. E., CHENTANEZ N., O'BRIEN J. F.: Fluid animation with dynamic meshes. In *Proc. SIGGRAPH* (2006), pp. 820–825.
- [KIC06] KIM J., IHM I., CHA D.: View dependent adaptive animation of liquids. *ETRI Journal* 28, 6 (2006), 697–708.
- [LCC\*07] LÖHNER R., CEBRAL J. R., CAMELLI F. E., APANABOYINA S., BAUM J. D., MESTREAU E. L., SOTO O. A.: Adaptive embedded and immersed unstructured grid techniques. *Computer Methods in Applied Mechanics and Engineering* 197 (2007), 2173–2197.
- [LFO05] LOSASSO F., FEDKIW R., OSHER S.: Spatially adaptive techniques for level set methods and incompressible flow. *Computers and Fluids* 35 (2005), 995–1010.
- [LGF04] LOSASSO F., GIBOU F., FEDKIW R.: Simulating water and smoke with an octree data structure. In *Proc. SIGGRAPH* (2004), pp. 457–462.
- [LKMU05] LIU H., KRISHNAN S., MARELLA S., UDAYKUMAR H. S.: Sharp interface Cartesian grid method II: a technique for simulating droplet interactions with surfaces of arbitrary shape. *J. Comp. Phys.* 210, 1 (2005), 32–54.
- [LS07] LABELLE F., SHEWCHUK J. R.: Isosurface stuffing: fast tetrahedral meshes with good dihedral angles. In *Proc. SIGGRAPH* (2007), p. 57.
- [MCP\*09] MULLEN P., CRANE K., PAVLOV D., TONG Y., DESBRUN M.: Energy-preserving integrators for fluid animation. In *Proc. SIGGRAPH* (2009), p. 38.
- [MDB\*08] MITTAL R., DONG H., BOZKURTASA M., NAJAR F., VARGASA A., VON LOEBBECKEA A.: A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *J. Comp. Phys.* 10 (2008), 4825–4852.
- [MKLU05] MARELLA S., KRISHNAN S., LIU H., UDAYKUMAR H. S.: Sharp interface Cartesian grid method I: an easily implemented technique for 3D moving boundary computations. *J. Comp. Phys.* 210, 1 (2005), 1–31.
- [Nic92] NICOLAIDES R. A.: Direct discretization of planar div-curl problems. *SIAM J. Numer. Anal.* 29, 1 (1992), 32–56.
- [NMG09] NG Y. T., MIN C., GIBOU F.: An efficient fluid-solid coupling algorithm for single-phase flows. *J. Comp. Phys.* 228 (2009), 8807–8829.
- [NW97] NICOLAIDES R. A., WU X.: Covolume solutions of three-dimensional div-curl equations. *SIAM J. Numer. Anal.* 34, 6 (1997), 2195–2203.
- [PB79] PURVIS J. W., BURKHALTER J. E.: Prediction of critical mach number for store configurations. *AIAA Journal* 17 (1979), 1170–1177.
- [Pes02] PESKIN C.: The immersed boundary method. *Acta Numerica* 11 (2002), 479–517.
- [PS07] PEROT J. B., SUBRAMANIAN V.: Discrete calculus methods for diffusion. *J. Comp. Phys.* 224, 1 (2007), 59–81.
- [PVW06] PEROT B., VIDOVIC D., WESSELING P.: Mimetic reconstruction of vectors. In *Compatible Spatial Discretizations*, Arnold D. N., Bochev P. B., Lehoucq R. B., Nicolaides R. A., Shashkov M., (Eds.). 2006, pp. 173–188.
- [RZF05] ROBLE D., ZAFAR N. B., FALT H.: Cartesian grid fluid simulation with irregular boundary voxels. In *ACM SIGGRAPH 2005 Sketches* (2005), p. 138.
- [SBCL06] SCHWARTZ P., BARAD M., COLELLA P., LIGOCKI T.: A Cartesian grid embedded boundary method for the heat equation and Poisson's equation in three dimensions. *J. Comp. Phys.* 211, 2 (2006), 531–550.
- [SBH09] SIN F., BARGTEIL A., HODGINS J.: A point-based method for animating incompressible flow. In *Proc. Symposium on Computer Animation 2009* (2009), pp. 247–255.
- [Sta99] STAM J.: Stable fluids. In *Proc. SIGGRAPH* (1999), pp. 121–128.
- [TAL09] TRAORÉ P., AHIPO Y. M., LOUSTE C.: A robust and efficient finite volume scheme for the discretization of diffusive flux on extremely skewed meshes in complex geometries. *J. Comp. Phys.* 228, 14 (2009), 5148–5159.
- [TWAD09] TOURNOIS J., WORMSER C., ALLIEZ P., DESBRUN M.: Interleaving delaunay refinement and optimization for practical isotropic tetrahedron mesh generation. In *Proc. SIGGRAPH* (2009), p. 75.
- [VHG08] VANDERZEE E., HIRANI A. N., GUOY D.: Triangulation of simple 3D shapes with well-centered tetrahedra. In *Proceedings of the 17th International Meshing Roundtable* (2008).
- [WBOL07] WENDT J. D., BAXTER W., OGUZ I., LIN M. C.: Finite volume flow simulations on arbitrary domains. *Graph. Models* 69, 1 (2007), 19–32.
- [WSHD04] WARREN J., SCHAEFER S., HIRANI A. N., DESBRUN M.: Barycentric coordinates for convex sets. *Advances in computational mathematics* 27, 3 (2004), 319–338.
- [WT08] WOJTAN C., TURK G.: Fast viscoelastic behavior with thin features. In *Proc. SIGGRAPH* (2008), p. 47.
- [ZSP02] ZHANG X., SCHMIDT D., PEROT B.: Accuracy and conservation properties of a three-dimensional unstructured staggered mesh scheme for fluid dynamics. *J. Comp. Phys.* 175, 2 (2002), 764–791.