

A Spatially Adaptive Morphological Filter For Dual-Resolution Interface Tracking Of Fluids

Andreas Söderström¹ and Ken Museth^{2,1}

¹Linköping University

²DreamWorks Animation

Abstract

We present a novel surface-tracking technique for free-surface fluid animations. Unlike the semi-implicit Particle Level Set method (PLS) our interface-tracking approach is purely implicit and hence avoids some of the well-known issues like surface noise and inflated memory footprints. Where PLS augments the interface with Lagrangian tracker-particles, we instead employ a higher resolution level set represented as a DT-Grid. The synchronization of our dual-resolution level sets is facilitated by a novel Spatially Adaptive Morphological (SAM) filter that attempts to preserve fine details while still avoiding spurious topology changes and boundary violations. We demonstrate that our approach can achieve comparative results to the PLS, but with a fraction of the memory footprint. We also show how our technique can be used to effectively enhance thin interface sheets at the cost of volume gain.

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

1. Introduction

Interface tracking lies at the very heart of any free-surface fluid animation. Some methods, notably SPH [MCG03] and FLIP [ZB05], use particles (*i.e.* explicit geometry), to track the deforming fluid/air interface. These Lagrangian approaches have the advantage that it is relatively easy to accurately advect the particles in the fluid field. A major disadvantage is however that it can be very complicated to reconstruct the actual fluid/air interface from the tracker particles. Consequently implicit representations like level sets, that can readily change topology, have become very popular for fluid animations over the past decade. However, the advection of level sets can introduce numerical dissipation due to limited grid resolution which in turn can lead to mass-loss and erosion of surface details. To address this problem the semi-implicit Particle Level Set (PLS) method [EFFM02] was proposed. It effectively combines Lagrangian tracker particles with an Eulerian level set and greatly reduces the mass-loss associated with pure level sets. However, PLS is known to introduce a large memory overhead due to the use of typically 64 particles per voxel, can be surprisingly hard to implement (*i.e.* minor variations in the parameters of the particle-correction step of the PLS algorithm can introduce dramatic changes) and has a tendency to introduce surface noise.

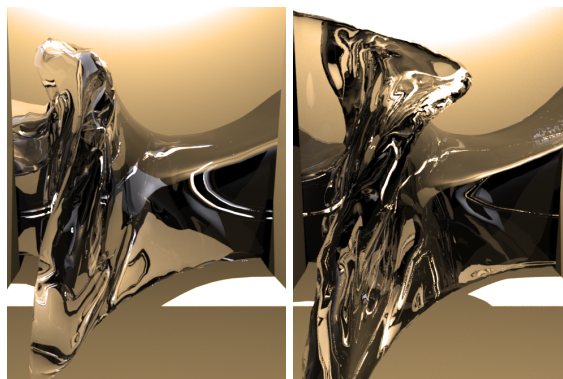


Figure 1: Water flooding into a corridor. The water surface to the left is tracked using a regular level set, the surface to the right uses our dual resolution surface tracking method.

To redeem these issues we propose the use of a dual resolution level set representation. This purely implicit approach effectively replaces the tracker particles by another higher-resolution level set which in turn is represented by the very compact narrow-band level set data-structure DT-Grid [NM06]. This allows us to achieve comparable results to the PLS, while dramatically reducing the memory footprint. Additionally the fact that our method is purely implicit makes it relatively robust and easy to implement. The key component of this dual-resolution level set approach is a Spatially Adaptive Morphology (SAM) filter that effectively establishes the relation between the two level sets at different resolution. An example of the additional level of detail that can be obtained with our method is shown in figure 1.

2. Contributions

The contributions of this paper are twofold. Firstly we present a spatially adaptive morphological (SAM) filter for dual-resolution level set tracking of free surfaces in fluid animations. Secondly we present a sheet preserving extension of this filter (SP-SAM) that effectively prevents thin sheets of fluid from dissipating due to limited sampling resolution. Our work can be considered a purely implicit alternative to the semi-explicit particle level set method.

3. The Spatially Adaptive Morphological Filter

At the core of our interface tracking algorithm is a dual-resolution level set representation: Let ϕ_h denote the level set representation of the fluid-air interface on a high-resolution grid, and ϕ_l a filtered (i.e. down-sampled) version of ϕ_h on a lower-resolution grid where the actual Navier-Stokes equation is solved for the velocity field \mathbf{v}_l . Let α denote the sampling rate of ϕ_h divided by ϕ_l . The challenge is now to derive ϕ_l from ϕ_h in a way that is feature-preserving and yet does not violate boundary conditions and introduce unnecessary topology changes. Once ϕ_l is defined we can compute \mathbf{v}_l which in turn can be up-sampled to \mathbf{v}_h using simple interpolation. This finally allow us to advect ϕ_h through the level set equation

$$\frac{\partial \phi_h}{\partial t} = \mathbf{v}_h \cdot \nabla \phi_h \quad (1)$$

Throughout this paper we will assume that ϕ_h and ϕ_l are signed Euclidean distance fields with positive distances defined on the outside of the interface and negative inside.

Let us start by noting that simply defining ϕ_l from a naive down-sampling of ϕ_h can cause severe aliasing problems. Although we are not explicitly rendering ϕ_l a poorly defined low-resolution interface can cause problems when solving the Navier-Stokes equations for \mathbf{v}_l which in turn will affect ϕ_h through Eq. 1. The Eulerian fluid solver will only be “aware” of the fluid region enclosed by ϕ_l and may thus fail to take into account regions of fluid that can be represented by ϕ_h but not by ϕ_l . This problem has recently been studied in [KSK09] by means of a so-called “liquid-biased filter”. This filter simply consists of a uniform dilation of ϕ_h by $0.5\Delta x_l$, i.e. half the width of a low resolution voxel, thus resulting in a ϕ_l where the thinnest possible feature is still greater than one such voxel. As a result all features in ϕ_h will be present in ϕ_l . However, the global nature of the liquid biased filter can cause several problems. First of all, if ϕ_h is in contact with a solid surface we may create a ϕ_l that penetrates this surface by as much as $0.5\Delta x_l$ potentially causing boundary problems. Furthermore, this dilation may cause topology changes in ϕ_l that are not present in ϕ_h . This includes premature merging (e.g. colliding droplets) and non-physical merging of, for example, separate surface details moving in parallel close to each other without touch-

ing. Finally, we observe that dilation is not necessary for any features of ϕ_h already large enough to be resolved on ϕ_l .

Based on these observations we have devised a Spatially Adaptive Morphological (SAM) filter that is able to mitigate aliasing problems during downsampling while still avoiding the topology and boundary problems described above. Our filtering algorithm consists of the following steps:

1. For every sample point \mathbf{x} : $|\phi(\mathbf{x})_h| < 1.5\Delta x_h$ calculate the surface normal \mathbf{n}_{ϕ_h} using an upwinding scheme [LOC94].
2. For every sample point \mathbf{x} : $|\phi(\mathbf{x})_h| < 1.5\Delta x_h$ gather the neighboring $2N$ sample points along the direction of the surface normal \mathbf{n}_{ϕ_h} into a linear stencil $S = \{\phi_h(\mathbf{x} - N\Delta x_h), \phi_h(\mathbf{x} - (N-1)\Delta x_h), \dots, \phi_h(\mathbf{x} + N\Delta x_h)\}$. Where we suggest $N \geq \alpha + 1$. See figure 2 for an illustration of the 7 point stencil, recommended when $\alpha = 2$.
3. Along every stencil S calculate the thickness $\epsilon_{\phi_h}^n$ of ϕ_h . Observe that ϕ_h may intersect the stencil multiple times, in which case the thickness is calculated from the part of ϕ_h that is closest to the center of the stencil. If ϕ_h crosses the stencil multiple times, and the distance between two such interfaces is less than some desired threshold ϵ_{fluid} , make note of this. Also make a note of any cases where the closest part of ϕ_h is in contact with a solid object. If the stencil is too small to estimate the thickness of any part of ϕ_h we set the estimated thickness to “infinity”.
4. For every grid point \mathbf{x} close to the interface of ϕ_h calculate the dilation function $g_{offset} = \max(\epsilon_{target} - \epsilon_{\phi_h}^n, 0)$ where ϵ_{target} is the desired minimum geometric thickness. At this point we need to decide what to do if the interface is close to either a solid or another part of the fluid as detected in step 3.. In regions where ϕ_h has a solid or fluid on both sides we do not need to widen the surface, i.e. $g_{offset} = 0$. If on the other hand ϕ_h only has fluid or solid on one side, and the current grid point \mathbf{x} is close to the side of the surface that is *not* in contact with the offending geometry, we keep g_{offset} as initially calculated. This will in effect only widen the surface in the direction away from the offending geometry, avoiding undesired penetration or merging. If the relevant part of ϕ_h is free we set $g_{offset} = 0.5g_{offset}$ in order to compensate for the fact that the surface will be widened on both sides.
5. The dilation, g_{offset} , estimated above will only be a good approximation close to the interface and only when the fluid is thin. Thus we extrapolation the surface estimation of g_{offset} to the rest of the level set function ϕ_h by means of the hyperbolic transport equation $\frac{\partial g_{offset}(\mathbf{x})}{\partial t} = -\mathbf{n}_{\phi_h} \cdot \nabla g_{offset}(\mathbf{x})$, with the interface as boundary condition. This extrapolation will be such that the values of g_{offset} are constant along the normals of ϕ_h thus generating a valid field for dilating the level set surface while still maintaining the Euclidean signed distance property.
6. We now create a temporary widened level set function $\phi(\mathbf{x})^{temp} = \phi_h(\mathbf{x}) - g(\mathbf{x})_{offset}, \forall \mathbf{x}$.
7. Finally $\phi_l(\mathbf{x})$ can be obtained by downsampling $\phi(\mathbf{x})^{temp}$ using for example trilinear interpolation.

Note that the algorithm outlined above *only* modifies ϕ_l . Consequently, even though we are adaptively dilating the surface, this operation will not accumulate mass over time.

4. Sheet Preserving Adaptive Filter

The adaptive nature of the algorithm described in section 3 also allows us to explicitly preserve thin sheets of fluid if desired. This can be done by applying our adaptive morphological filter to regions of ϕ_h that are deemed thin and free. In order to accomplish this we extend the algorithm in section 3 as follows:

1. Calculate $\epsilon_{\phi_h}^n$ as described in section 3 step 1. - 3.
2. For every sample point \mathbf{x} close to the interface of ϕ_h calculate the surface offset function $g_{offset}^{widen} = \max(\epsilon_{target}^{widen} - \epsilon_{\phi_h}^n, 0)$ where $\epsilon_{target}^{widen}$ is the desired minimum sheet thickness. If the surface closest to \mathbf{x} is deemed *not* free then $g_{offset}^{widen} = 0$ for that point, allowing the fluid to break apart when contacting solid geometry and preventing undesired merging. To suppress undesired amplification of surface noise we can optionally set $g_{offset}^{widen} = 0$ in regions where the mean curvature is larger than a user-defined threshold.
3. Extrapolate g_{offset}^{widen} as described in section 3 step 5.
4. Perform the dilation $\phi(\mathbf{x})_h \rightarrow \phi(\mathbf{x})_h + g(\mathbf{x})_{offset}^{widen}, \forall \mathbf{x}$.

For sufficiently large $\epsilon_{target}^{widen}$ this Sheet Preserving Spatially Adaptive Morphological (SP-SAM) filter will guarantee that free parts of ϕ_h are always wide enough not to dissipate due to limited grid resolution. This approach may of course add mass to the simulation, however the amount added will decrease with increasing resolution of ϕ_h since typically $\epsilon_{target} = \beta \Delta x_h$ where β is constant. Still, we deem the SP-SAM filter most useful when significant sources are already present in the simulation, or when persistent thin sheets of fluid are more desirable than volume conservation.

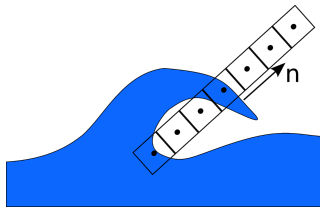


Figure 2: Example of a normal-aligned 7 point stencil.

5. Results

The goal of this paper is to present an alternative to the fast but memory demanding and sometimes noisy PLS method. As such our main focus is on surface fidelity and low memory footprint. However, we also put emphasis on low computational complexity and for this reason we will limit ourselves to $\alpha = 2$ throughout the rest of this paper. As can be seen from the examples below this is sufficient to allow for surface details comparative to the PLS. However, if more surface detail is desired a larger α may be chosen. For all

examples below we use the parameters $\epsilon_{fluid} = 1.05\Delta x_l$, $\epsilon_{target} = 1.05\Delta x_l$ and $\epsilon_{target}^{widen} = 2.1\Delta x_h$. In order to maximize the advantage of our higher resolution surface we solve equation (1) through high order explicit integration. For all examples the HJ WENO [LOC94] and TVD Runge-Kutta [SO88] methods were employed.

5.1. Breaking Dam Simulation

This example is intended to show the additional level of detail that can be obtained by employing our method. A comparison between our surface tracking approach and several other methods is shown in figure 3. Most notably we see that our dual interface approach provides comparative results to the PLS method without any surface noise. We also note that the SAM and SP-SAM filters behave very similarly. This is to be expected since the SP-SAM filter only widens surfaces when needed and thus leaves the simulation untouched in well behaved cases such as this.

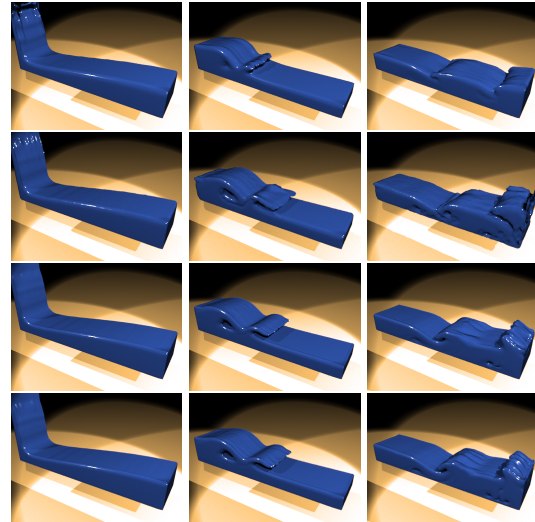


Figure 3: Each row of images above show three snapshots (frame 40, 73 and 110) of the breaking dam simulation using different surface tracking methods. From top to bottom these methods are: Single level set at 100^3 resolution, PLS at 100^3 , Dual level sets and SAM filter at $100^3/200^3$. Dual level sets and SP-SAM filter at $100^3/200^3$.

5.2. Splash Simulation

This example tests the behavior of our method during extreme stretching of the fluid. A comparison of frames from this simulation using different surface tracking methods can be found in figure 4. In this example we see the real value of the SP-SAM filter. Where both a high resolution single level set and a PLS fail to keep the sheets of fluid, our method kicks in and creates a smooth, noiseless surface. Note that as expected the surface still breaks apart on impact with the pedestal.

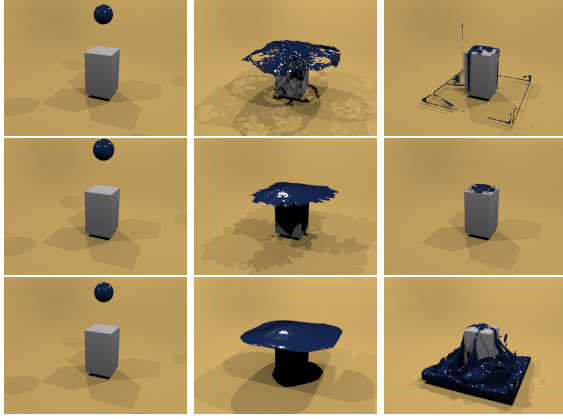


Figure 4: Each row of images above show three snapshots (frames 2, 30 and 69) of the Splash simulation using different surface tracking methods. From top to bottom the methods are: PLS at 100^3 resolution, dual level sets & SAM at $100^3/200^3$ and dual level sets & SP-SAM at $100^3/200^3$.

	100^3	200^3	300^3	400^3
Single level set	1.9	12.1	45.6	153
PLS	2.8	20.1	67.7	181
Dual + SAM	6.2	28.8	78.8	207
Dual + SP-SAM	6.3	31.8	96.6	212
Single level set 2x res	12.1	153	689	2261

Table 1: Benchmark results showing average time in seconds per frame for the initial frames of the Braking Dam simulation.

5.3. Computational cost

Table 1 shows the average computation time per frame for the first couple of frames of the breaking dam simulation. In table 2 we show the memory footprint associated with representing the fluid surface of the breaking dam simulation using different methods and resolutions. All methods use the DT-Grid data structure. As can be seen our method is slightly slower than the PLS, but requires more than an order of magnitude less memory.

6. Conclusions

We have presented a dual resolution purely implicit interface tracking algorithm based on a spatially adaptive morphological filter. We have also shown how this filter can be extended to preserve thin sheets at the cost of volume gain. As can be seen in figure 3 our algorithm allows us to capture additional fluid surface details without increasing the resolution of the fluid solver itself. From figures 3 and 4 we also conclude that the visual appearance is similar to the level of detail obtained by the common PLS method. In figure 4 we see that our algorithm can be used to preserve thin sheets of fluid even under extreme conditions. In all our tests we have also observed that the resulting fluid surfaces are smooth and free from the subtle noise often introduced by the PLS method. In

	100^3	200^3	300^3	400^3
Single level set	0.21	0.80	1.79	3.18
SAM & SP-SAM	0.90	3.63	8.19	14.58
PLS	14.81	59.99	135.61	241.56

Table 2: Table showing initial memory footprints (in MB) for the fluid surface of the breaking dam simulation. The PLS uses a particle density of 64.

table 1 we show that the computational cost of our algorithm is significantly less than if the full simulation was run at the higher resolution. This emphasizes that our method provides additional detail without introducing a significant computational overhead. In table 2 we also show that the memory footprint of our method is more than an order of magnitude less than that of the PLS - typically 16 times smaller.

The main disadvantage of our method is that it is slightly more computationally expensive than the PLS method as can be seen from table 1. Also, the difference in resolution between \mathbf{v}_l and ϕ_h could potentially lead to parts of ϕ_h moving in an unphysical manner. However, our adaptive filter takes several steps to mitigate this by attempting to avoid unintentional merging of ϕ_h as it is downsampled.

All things considered we conclude that the method described in this paper constitutes an effective implicit alternative to the PLS method. However we do not claim to have invented the “silver-bullet” for fluid surface tracking, and we acknowledge that the PLS is still an attractive method. Nevertheless, we hope and expect that our method will become yet another useful addition to the ever growing fluid animation toolbox.

References

- [EFFM02] ENRIGHT D., FEDKIW R., FERZIGER J., MITCHELL I.: A hybrid particle level set method for improved interface capturing. *J. Comput. Phys.* 183, 1 (2002), 83–116.
- [KSK09] KIM D., SONG O.-Y., KO H.-S.: Stretching and wiggling liquid. *ACM Transactions on Graphics* 28, 5 (2009), 120.
- [LOC94] LIU X., OSHER S., CHAN T.: Weighted essentially nonoscillatory schemes. *J. Comput. Phys.* 115 (1994), 200–212.
- [MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 154–159.
- [NM06] NIELSEN M. B., MUSETH K.: Dynamic tubular grid: An efficient data structure and algorithms for high resolution level sets. *J. Sci. Comput.* 26, 3 (2006), 261–299.
- [SO88] SHU C., OSHER S.: Efficient implementation of essentially non-oscillatory shock capturing schemes. *J. Comput. Phys.* 77 (1988), 439–471.
- [ZB05] ZHU Y., BRIDSON R.: Animating sand as a fluid. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), ACM, pp. 965–972.