Eurographics Workshop on Natural Phenomena (2005)
E. Galin, P. Poulin (Editors)

FlowFixer: Using BFECC for Fluid Simulation

ByungMoon Kim Yingjie Liu Ignacio Llamas Jarek Rossignac

Georgia Institute of Technology

Abstract

Back and Forth Error Compensation and Correction (BFECC) was recently developed for interface computation by using the level set method. We show that it can be applied to reduce dissipation and diffusion encountered in various advection steps in fluid simulation such as velocity, smoke density and image advections. BFECC can be implemented easily on top of the first order upwinding or semi-Lagrangian integration of advection equations, while providing second order accuracy both in space and time. When applied to level set evolution, BFECC reduces volume loss significantly. We combine these techniques with variable density projection and show that they yield a realistic animations of two-phase flows. We demonstrate the benefits of this approach on the image advection and on the simulation of smoke, of bubbles in water, and of a highly dynamic interaction between water, a solid, and air.

1. Introduction

Simulation of incompressible fluid involves several computation steps including diffusion, advection and pressure projection. Advection steps transport some quantities from one region to another along the fluid's velocity field. In this paper, we explore four forms of advection encountered in fluid simulation: velocity, smoke density, image and level set advections. *Velocity advection* transports the velocity field along the velocity itself. This step is always needed in non-steady flow simulation based on Navier-Stokes equation. *Smoke density advection* transports smoke along the velocity field. Sometimes, we may want to advect a colored image, which may be considered as colored smoke. We call this process *image advection*. When one uses a level set method [OS88] to simulate a free surface or a two-phase flow, for example a water surface simulation, the level set must be transported as well. We call it *level set advection*.

Those advection steps can be computed by an upwind or a semi-Lagrangian method. The latter is often preferred due to its stability for large time step. The first order semi-Lagrangian method is popular in computer animation because of its simplicity. However, the first order semi-Lagrangian contains a significant amount of numerical diffusion and dissipation. In velocity advection, it yields damped fluid motion.

In smoke density advection, it leads to a premature dilution of smoke, and is not able to simulate pure advection. Therefore, higher order schemes, such as WENO or CIP [TFK*03], are desired. We show that the implementation complexity of these schemes may be easily avoided by adding a very simple Back and Forth Error Compensation and Correction (BFECC) to an existing first order semi-Lagrangian schemes, thus improving its space and time accuracy to second order. We show that this approach reduces velocity damping and smoke density dilution and demonstrate its benefits on the four forms of advections discussed previously.

BFECC was recently proposed in [DL03, DL04] as a level set interface computation method. As is mentioned in [ELF05], high order methods may not prevent volume loss much. However, the authors of [DL03] combined BFECC with their simple redistancing technique and applied it to the Zalesak's problem, showing significantly reduced the volume loss. In the level set framework, the one with smallest volume error would be the particle level set method [EMF02, ELF05] In this paper, we focus on applying BFECC to velocity and smoke advection steps rather than level set advection. We, however, use BFECC and the simple redistancing for level set advection of various fluid simulations and show that sufficiently realistic fluid animation can be obtained. It would be interesting to apply this to the level set advection part of the particle level set method [ELF05] for more demanding simulation.

2. Previous Work

The stability problems in the earlier works such as [FM96] were successfully remedied in [Sta99] by introducing the pressure projection scheme to enforce incompressibility of the fluid and the semi-Lagrangian treatment of the advection term in the Navier-Stoke equation. This solution is popular for the simulation of incompressible Fluids, such as smoke [FSJ01] and also for more challenging free surface flows [FF01, EMF02].

The semi-Lagrangian velocity advection [Sta99] comes with built-in dissipation, *i.e.*, the velocity is dissipated quickly since the linear interpolation in the first order semi-Lagrangian produces large error. While higher order interpolation can solve the problem, it involves more neighboring grid point values and increases the complexity, particularly when non-uniform mesh structures are used. In [FSJ01], vorticity is added to generate small scale fluid rolling motion. Recently, [SSK05] addressed this built-in dissipation by enhancing advection itself. They adopted the CIP [TFK*03] method that increases the order of accuracy in space by introducing the derivatives of velocity to build a sub-cell velocity profile. A nice feature of this CIP method is that it is *local* in the sense that only the grid point values of one cell are used in order to update a point value. However, in this CIP method, all components of velocity and their partial derivatives should be advected, increasing the implementation complexity and computation time, especially in 3D. It is also worth noting that CIP has higher order accuracy in space only. Therefore high order integration of characteristics is also necessary. In contrast, BFECC is easier to implement and exhibits second order accuracy both in space and time and is *local* during each of its operational steps.

Song et al [SSK05] focused on applying CIP to generate more dynamic water surface behavior. However, we believe that having less dissipative and diffusive advection provides significant benefits in smoke simulations as well. This is illustrated in the middle five images of Fig. 4, where large amount of dissipation makes the smoke look very dark. In contrast, when BFECC is used, the smoke keeps full brightness throughout the simulation as is shown in the last five images.

The introduction of the level set method to fluid animation in [FF01] allowed realistic simulation of fluids with complex free surfaces. The problem left here was the volume loss in the level set method and the solution, known as the particle level set method, proposed subsequently in [EMF02], turned out to be very successful in volume preservation.

The two phase fluid solver using variable density projection has been broadly studied in mathematics and fluid mechanics [SSO94, OKBG00, HKLS04]. It has been used in graphics applications by [HK03], where the authors simulated air bubbles rising and merging and by [TFK*03, SSK05], where splash style interactions between water surface and air are studied.

3. Fluid Simulation

Consider the following Navier-Stokes equation

$$\frac{\partial \mathbf{u}}{\partial t} = -\mathbf{u}\cdot\nabla\mathbf{u} + \nu\nabla\cdot(\nabla\mathbf{u}) - \frac{1}{\rho}\nabla P + \mathbf{f}. \tag{1}$$

We follow the operator splitting steps proposed in [Sta99] except for the advection step, where we use BFECC and for the projection step for which we use the variable density pressure projection. We use the standard staggered grid [FSJ01]. Suppose all terms in (1) except for $-\frac{1}{\rho}\nabla P$ are treated and let the velocity obtained so far be $\tilde{\mathbf{u}}$. The final step is applying the variable density pressure projection step to enforce the continuity equation $\nabla\cdot\mathbf{u} = 0$, i.e, solving the equation $\nabla\cdot\left(\frac{\Delta t}{\rho}\nabla P\right) = \nabla\cdot\tilde{\mathbf{u}}$. Its first order discretization is

$$\begin{aligned}\frac{\Delta t}{\Delta x^2}&\left(\frac{P_{i,j}-P_{i-1,j}}{\rho_{i-\frac{1}{2},j}} + \frac{P_{i,j}-P_{i+1,j}}{\rho_{i+\frac{1}{2},j}} + \frac{P_{i,j}-P_{i,j-1}}{\rho_{i,j-\frac{1}{2}}} + \frac{P_{i,j}-P_{i,j+1}}{\rho_{i,j+\frac{1}{2}}}\right)\\ &= \frac{1}{\Delta x}\left(\tilde{u}_{i+\frac{1}{2},j} - \tilde{u}_{i-\frac{1}{2},j} \;+\; \tilde{v}_{i,j+\frac{1}{2}} - \tilde{v}_{i,j-\frac{1}{2}}\right).\end{aligned} \tag{2}$$

We assume $\Delta x = \Delta y$ here and through the rest of the presentation. The extension to 3D is straightforward and hence omitted. This first order approximation is identical to [SSK05] and higher order formulations can be found in [ABS96, SAB*99]. Obviously, if ρ is constant, we have the pressure projection $\frac{\Delta t}{\rho}\nabla^2 P = \nabla\cdot\tilde{\mathbf{u}}$ introduced in [Sta99]. We also include a simple implementation of surface tension similar to [SAB*99].

4. The BFECC Method

In this section, we review the BFECC method. Since we want to apply it to various advections, we use φ to denote a quantity that is advected and reserve the symbol ϕ for the level set function through the presentation of this paper. This φ can be the velocity components u, v, w, smoke density, RGB color of an image or level set function ϕ, satisfying

$$\varphi_t + \mathbf{u}\cdot\nabla\varphi = 0. \tag{3}$$

We briefly describe the BFECC method here. Let L be the first order upwinding or semi-Lagrangian integration steps to integrate (3), such that

$$\varphi^{n+1} = L(\mathbf{u}, \varphi^n). \tag{4}$$

Using this notation, the BFECC can be written as the following three L-calling process

$$\begin{aligned}\varphi^{n+1} &= L\left(\mathbf{u},\quad \varphi^n + \frac{1}{2}\left(\varphi^n - \bar{\varphi}\right)\right)\\ &\text{where}\;\; \bar{\varphi} = L\left(-\mathbf{u},\;\; L\left(\mathbf{u}, \varphi^n\right)\right).\end{aligned} \tag{5}$$

One may understand this method intuitively as follows. If the advection step $L(\cdot,\cdot)$ is exact, the first two forward and backward steps should return the value exactly the same as the original one, *i.e.*, $\varphi^n = \bar{\varphi}$. However, this does not hold

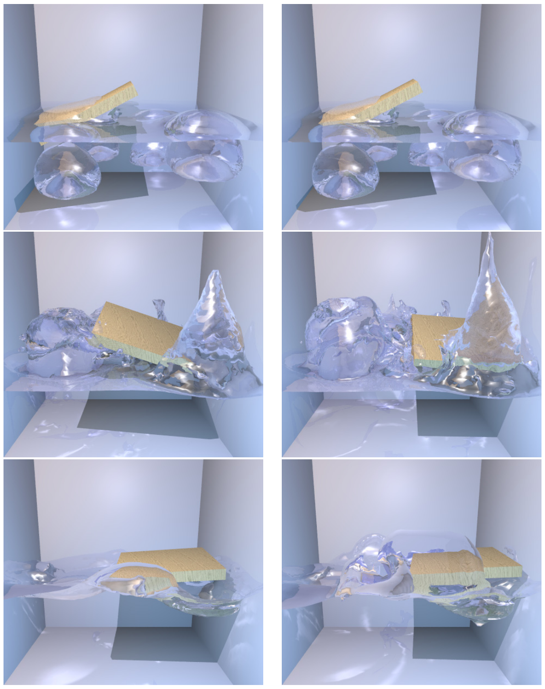

Figure 1: *On the right column, a highly dynamic behavior of water interaction with air, air bubbles, and solid is made possible by the two-phase formulation and the BFECC-based reduction of the dissipation in the velocity advection step. On the left column, the BFECC is turned off and the splash is lower.*

due to the error in the advection operation L. Suppose L contains an error e. Then the first two forward and backward steps will produce error $2e$, *i.e.*, $\bar{\varphi} = \phi^n + 2e$. Therefore, the error can be computed as $e = -\frac{1}{2}(\varphi^n - \bar{\varphi})$. We subtract this error e before the final forward advection step. Then the equation (5) becomes $\varphi^{n+1} = L(\mathbf{u}, \varphi^n - e)$. This step will add an additional e, which will be cancelled by the subtracted amount $-e$. This method is proven to be second order accurate both in space and time [DL03, DL04].

4.1. Implementation of BFECC

In this section, we provide a pseudo code to show the simplicity of the BFECC implementation. First let the function SingleStep($u, v, \varphi^n, \varphi^{n+1}$) implement upwind or semi-Lagrangian integration of the scalar field φ, which can be the velocity components u,v,w, the smoke density, RGB colors of an image or the level set function ϕ. Then BFECC is implemented as:

```
SingleStep($u, v, \varphi^n, \tilde{\varphi}$)
SingleStep($-u, -v, \tilde{\varphi}, \bar{\varphi}$)
$\tilde{\varphi} := \varphi^n + (\varphi^n - \bar{\varphi})/2$
SingleStep($u, v, \tilde{\varphi}, \varphi^{n+1}$)
```

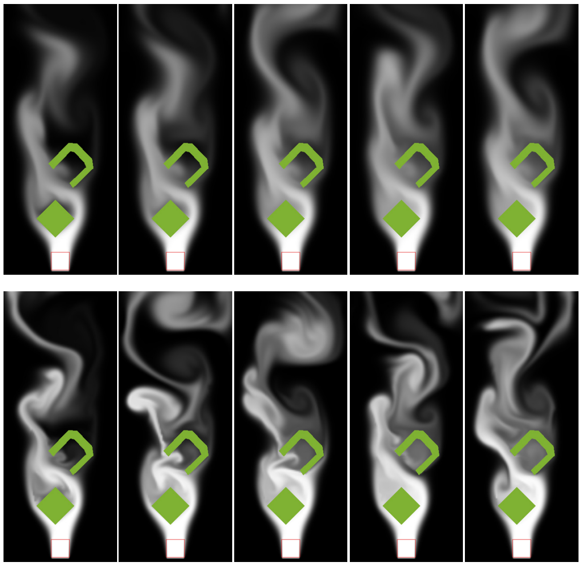

Figure 2: *Comparison of without (top) and with (bottom) BFECC in velocity advection on a 80×200 grid. Top images show damped fluid motion. Bottom ones show small scale details as well as large scale fluctuations thanks to BFECC applied to the velocity advection step.*

4.2. BFECC for Velocity Advection

We can use (5) to implement the velocity advection step in solving the Navier-Stokes Equation. In this case, φ becomes u, v and w. We show that BFECC can improve the damping in the first order semi-Lagrangian implementation of velocity advection, which is a well known drawback of [Sta99]. For multiphase flow, this BFECC needs to be turned off near the interface to prevent velocities of different fluids with different densities from being mixed, which creates momentum changes. We simply turn BFECC off, *i.e.*, use the first order semi-Lagrangian, for the grid points where $|\phi| < 5\Delta x$. We also turn it off near the boundary. Notice that reducing velocity dissipation is equally important in the entire fluid domain, not only near the interface. In other words, turning BFECC off near the interface has little effect since it is still turned on in most of the fluid domain.

As is shown in Fig. 2, applying BFECC adds details in smoke motion. Notice that these details cannot be obtained from the vorticity confinement method [FSJ01], which only adds small scale rolling motions. We also performed the same test in a coarser grid of 100×40. In this case, the flow did not fluctuate at all around the obstacles with the first order semi-Lagrangian advection. However, when BFECC was added, the flow fluctuated as in the refined grid. We conclude that BFECC creates a physically correct fluctuations in a coarser grid.

Velocity advection can also be important when rigid bodies are involved. In Fig. 7, the cup does not tumble due to

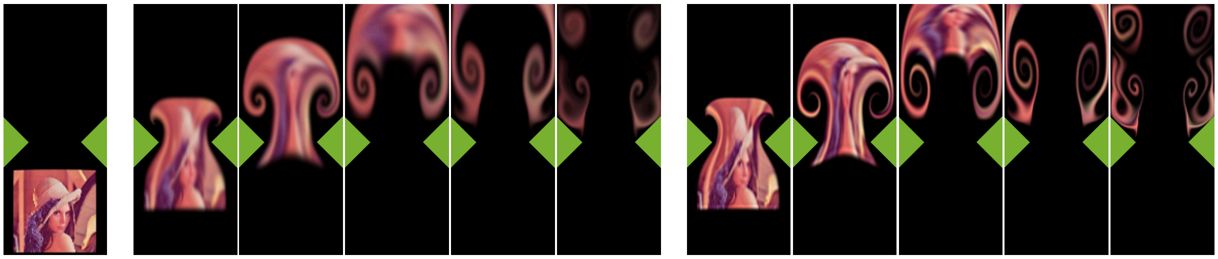

Figure 3: *Advection of an image along with the up-going flow field on 100×250 grid. The first image shows initial location of the image. The next six images are without BFECC where the dissipation/diffusion are significant. The last six images are with BFECC, where the dissipation is greatly reduced and the features of the image can be identified.*

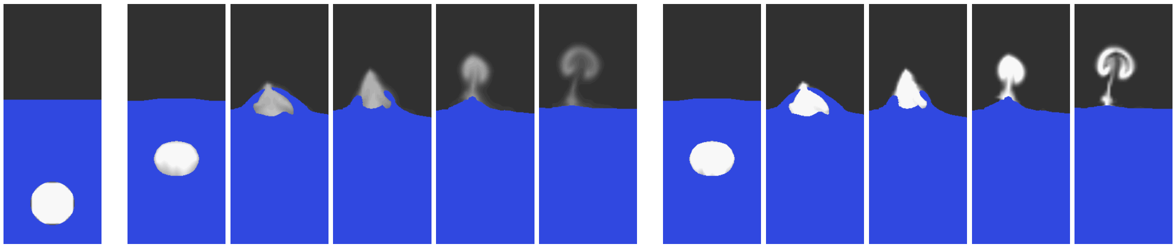

Figure 4: *Simulation of weakly diffusive and dissipative smoke in bubble rising and bursting situation. The far left image shows the initial bubble. The next five are without BFECC, where the dissipation/diffusion in the semi-Lagrangian step deteriorate the density of smoke. The last five images simulated with BFECC show minimal dissipation/diffusion and the smoke is in full density throughout the simulation. Notice that all simulation parameters are exactly same except for the usage of BFECC in smoke advection. Therefore the only difference is the density of smoke. Also, notice that the simulation time differs by less than 1% since it is dominated by the pressure projection step.*

the velocity dissipation in the first order semi-Lagrangian method, while the cup tumbles when BFECC is applied to the velocity advection step.

4.3. BFECC for Smoke Density and Image Advection

We also apply BFECC to the advection of smoke density for the smoke simulation. In Fig. 3 and 4, we show that BFECC can reduce dissipation and diffusion significantly. As is shown in [DL03], BFECC is linearly stable in l^2 sense, *i.e.*, $||a||_{l^2} = \sum |a_{ij}|^2$ is bounded, when the velocity field is constant, where a is the smoke density. However, density values a_{ij} can become negative or greater than 1.0 for some grid points. In our simulation, this problem was not significant and we simply clamped those values to stay in $[0,1]$.

To measure the diffusion/dissipation amount, we design a test problem similar to Zalesak's problem. Instead of the notched disk, we place a color image and rotate it 360 degree and then compare it with the original image as is shown in Fig. 5. As is shown in (d), the dissipation of the color is significantly reduced with BFECC. During the advection, the image is also diffused to neighboring region, even though it is not visible. To visualize the diffusion amount, we plot background pixels as blue to show the region where the image has been diffused into. As is shown in (d), the color of the object is little diffused into neighboring region when BFECC is used. Also notice that the size of the image looks smaller and its position is noticeably different from the original location in (c), which is again fixed in (d) where BFECC is used. The computation time was 0.156 sec (without BFECC) and 0.36 sec (with BFECC) per frame on a 3GHz Pentuim4.

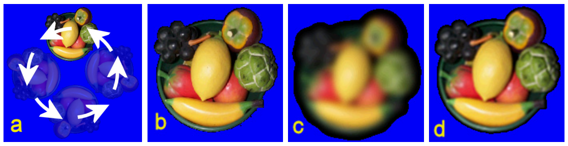

Figure 5: *Test of dissipation and diffusion with image advection problem on a rotating vector field (800×800 grid, CFL = 6.29). (b) is the top center portion of the original image (a). (c) is obtained by rotating it 360 degree using the first order semi-Lagrangian scheme, where one can see large amount of dissipation, diffusion, shrinkage of image and position error. These are significantly improved in (d) where BFECC is used. The blue background region is in fact in black color but it is rendered as blue to illustrate the region where the color is not diffused into.*

Advection is often used for scientific visualization, especially for various forms of flow visualization. For example, [JEH02] uses semi-Lagrangian advection of dye to visualize the vector field. [Wei04] applied level set method to advect dye without diffusion. Only one dye color is allowed

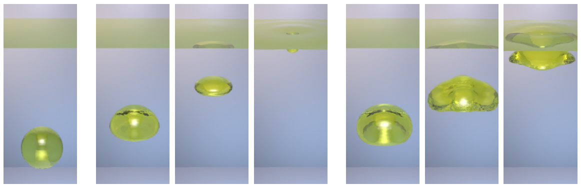

Figure 6: *The far left image shows an air bubble placed in olive oil at time zero. The next three images are first order semi-Lagrangian implementation of level set advection. The next three images are produced using BFECC and simple redistancing, showing significantly reduced volume loss.*

and the dye cannot be diffused at all. Also level set implementation is needed. In contrast, BFECC is trivial to implement and provides advection of fully colored pattern of dye, if necessary. As is shown in Fig. 3, the dissipation/diffusion is very small. Thus, we believe that it can be used in flow visualization as well. This remains as a future work.

4.4. BFECC for Level Set Advection

Even though, BFECC still has some volume loss in fluid simulation, especially for small droplets or thin filaments, it is still interesting to show how BFECC performs in the fluid simulation since it is trivial to implement and fast.

When we use the BFECC for level set advection, *i.e.*, $\varphi = \phi$, redistancing is needed to keep the level set function as a signed distance function. We use the following redistancing equation [SSO94]

$$\phi_\tau + \mathbf{w} \cdot \nabla\phi = \mathrm{sgn}(\phi) \quad \text{where } \mathbf{w} = \mathrm{sgn}(\phi)\frac{\nabla\phi}{|\nabla\phi|}. \qquad (6)$$

w is the velocity vector for redistancing. This equation can be solved by applying first order upwinding in discretizing the term $\mathbf{w} \cdot \nabla\phi$. An alternative is the semi-Lagrangian style integration, *i.e.*, $\phi^{n+1} = \phi^n(\mathbf{x} - \mathbf{w}\Delta\tau) + \mathrm{sgn}(\phi^n)\Delta\tau$, where $\mathbf{x}$ is the location of each grid point. Hence, $\phi^n(\mathbf{x} - \mathbf{w}\Delta\tau)$ is the ϕ value of previous location.

When these integration formulae for (6) are combined with BFECC, the redistancing tends to spoil good ϕ values computed from the second order accurate BFECC. This leads to the idea of turning redistancing off near the interface to keep good ϕ values there. The conditions to turn off redistancing is provided in [DL03], where the significant enhancement were shown for the Zalesak's problem. This simple redistancing is crucial in preserving volume [DL03]. It is also easy to implement since it simply requires to perform redistancing at the points where at least one of the following two conditions are met.

- When the grid point is not close to the interface, *i.e.*, when $\phi_{i,j}$ has the same sign with its eight neighbors.
- When the slope is sufficiently high, *i.e.*, when $|\phi_{i,j} - \phi_{i\pm1,j}|$ or $|\phi_{i,j} - \phi_{i,j\pm1}| \geq 1.1\Delta x$.

5. Results

We test BFECC in different fluid simulations. We simulate air-water and olive oil-air interactions. Properties of these fluids are provided in Table 1. Water is rendered as bluish surface and olive oil is rendered in yellowish color. We use PovRay (http://povray.org) to render images.

	$\rho[kg/m^3]$	$\nu[m^2/sec]$	Surf. Tension [N/m]
air	1.125	1.7×10^{-5}	
water	1000	1.0×10^{-6}	0.07
olive oil	910	9.2×10^{-5}	0.035

Table 1: *Properties of fluids used in simulations*

In Fig. 7, we simulated a cup, air and water interaction. The cup is released upside down near the water surface. Due to its weight, the cup sinks deep into water but it soon rise again because of the air in it. However, in the top, we turned BFECC off for velocity advection and hence the water became dissipative, preventing the cup from tumbling. In the bottom, we use BFECC for velocity advection, where the velocity dissipation is small and hence the cup can tumble 180 degree. This example indicates that reducing velocity dissipation could be important in simulating fluid and rigid body interaction. We implement the *rigid fluid* method [CMT04] to simulate rigid body and fluid interaction in Fig. 1 and 7. We use multiple pressure projections to address the seeping problem mentioned in [CMT04].

The computation time varies in situations such as the complexity of fluid motions. In simple bubble rising situation without rigid body, it took a few seconds per time step using a 50^3 mesh. The cup example in Fig. 7 has multiple pressure projections and it took about 30 to 130 seconds per time step on a 70^3 grid.

6. Conclusion

We have shown that the BFECC scheme can be used to improve the simulation of fluids. Once the simple first order upwinding or semi-Lagrangian steps for velocity, smoke density, image or level set advections are implemented, BFECC can be added with a trivial amount of code. We show that this simple extension yields significant enhancements in reducing diffusion and dissipation in velocity, smoke, image advection and in preserving volume under various situations including two-phase flows and rigid bodies. The benefits of the proposed approach are illustrated in the accompanying video.

7. Acknowledgement

This work was supported by the NSF under the ITR Digital clay grant 0121663.

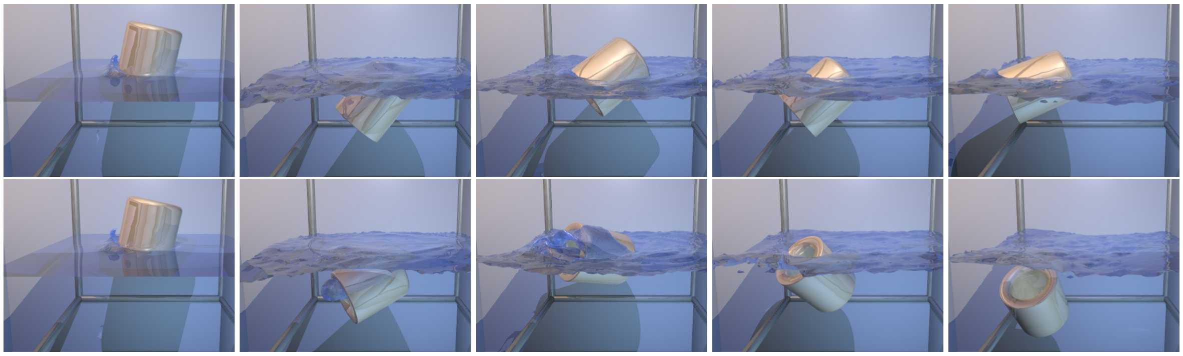

Figure 7: *Simulation of a sinking cup. The top row is simulated without the BFECC in velocity advection, where the cup fails to tumble. The bottom row is simulated with BFECC and the cup tumbles thanks to the reduced damping in velocity field.*

References

[ABS96] ALMGREN A. S., BELL J. B., SZYMCZAK W. G.: A numerical method for the incompressible navier-stokes equations based on an approximate projection. *SIAM Journal of Scientific Computing 17*, 2 (March 1996).

[CMT04] CARLSON M., MUCHA P. J., TURK G.: Rigid fluid: Animating the interplay between rigid bodies and fluid. In *SIGGRAPH* (2004), ACM.

[DL03] DUPONT T. F., LIU Y.: Back and forth error compensation and correction methods for removing errors induced by uneven gradients of the level set function. *Journal of Computational Physics 190*, 1 (2003), 311–324.

[DL04] DUPONT T. F., LIU Y.: Back and forth error compensation and correction methods for semi-lagrangian schemes with application to interface computation using level set method. In *CDSNS2004-399, School of Mathematics, Georgia Institute of Technology* (2004).

[ELF05] ENRIGHT D., LOSASSO F., FEDKIW R.: A fast and accurate semi-lagrangian particle level set method. *Computers and Structures 83* (2005), 479–490.

[EMF02] ENRIGHT D., MARSCHNER S., FEDKIW R.: Animation and rendering of complex water surfaces. In *SIGGRAPH* (2002), ACM.

[FF01] FOSTER N., FEDKIW R.: Practical animation of liquids. In *SIGGRAPH* (2001), ACM, pp. 15–22.

[FM96] FOSTER N., METAXAS D.: Realistic animation of liquids. *Graphical Models and Image Processing 58*, 5 (1996), 471–483.

[FSJ01] FEDKIW R., STAM J., JENSEN H.: Visual simulation of smoke. In *SIGGRAPH* (2001), ACM, pp. 23–30.

[HK03] HONG J.-M., KIM C.-H.: Animation of bubbles in liquid. In *EUROGRAPHICS* (2003), vol. 22.

[HKLS04] HAARIO H., KOROTKAYA Z., LUUKKA P., SMOLIANSKI A.: Computational modelling of complex phenomena in bubble dynamics: Vortex shedding and bubble swarms. In *Proceedings of ECCOMAS 2004* (2004).

[JEH02] JOBARD B., ERLEBACHER G., HUSSAINI M. Y.: Lagrangian-eulerian advection of noise and dye textures for unsteady flow visualization. *IEEE Transactions on Visualization and Computer Graphics 8*, 3 (2002).

[OKBG00] OEVERMANN M., KLEIN R., BERGER M., GOODMAN J.: *A Projection Method for Two-Phase Incompressible Flow with Surface Tension and Sharp Interface Resolution*. Tech. Rep. ZIB-Report 00-17, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2000.

[OS88] OSHER S., SETHIAN J. A.: Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics 79* (1988), 12–49.

[SAB*99] SUSSMAN M., ALMGREN A., BELL J., COLELLA P., HOWELL L., WELCOME M.: An adaptive level set approach for incompressible two-phase flow. *Journal of Computational Physics 148* (1999), 81–124.

[SSK05] SONG O., SHIN H., KO H.: Stable but nondissipative water. *ACM Transactions on Graphics 24*, 1 (2005), 81–97.

[SSO94] SUSSMAN M., SMEREKA P., OSHER S.: A levelset approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics 114*, 1 (1994), 146–159.

[Sta99] STAM J.: Stable fluids. In *SIGGRAPH* (1999), ACM, pp. 121–128.

[TFK*03] TAKAHASHI T., FUJII H., KUNIMATSU A., HIWADA K., SAITO T., TANAKA K., UEKI H.: Realistic animation of fluid with splash and foam. In *EUROGRAPHICS* (2003), vol. 22.

[Wei04] WEISKOPF D.: Dye advection without the blur: A level-set approach for texture-based visualization of unsteady flow. In *EUROGRAPHICS* (2004), vol. 23.