A Practical Guide to Thin Film and Drips Simulation

Alexey Stomakhin
Weta Digital
astomakhin@wetafx.co.nz

Andrew Moffat
Weta Digital
amoffat@wetafx.co.nz

Gary Boyle
Weta Digital
gboyle@wetafx.co.nz

Figure 1: Shot from *Alita: Battle Angel* demonstrating the use of our technique to capture subtle water drips and thin film dynamics on the character's skin. ©2018 Twentieth Century Fox Film Corporation. All rights reserved.

ABSTRACT

We present a practical approach to model close-up water interaction with characters. We specifically focus on high-fidelity surface tension and adhesion effects as water sheds off skin. We show that an existing particle-in-cell (FLIP/APIC) solver can be adapted to capture small-scale water-solid interaction dynamics and discuss the role and implementation details of the relevant key components: treatment of surface tension and viscosity, enforcement of contact angle, and maintenance of contact with fast-moving collision geometry. The method allows for resolution of effects on a scale of a fraction of a millimeter and is performant enough to be able to cover a whole human body with a layer of water. We demonstrate successful use of the approach in a shot from *Alita: Battle Angel*.

CCS CONCEPTS

• **Computing methodologies → Physical simulation**;

KEYWORDS

physical simulation, drips, thin film, surface tension

ACM Reference format:
Alexey Stomakhin, Andrew Moffat, and Gary Boyle. 2019. A Practical Guide to Thin Film and Drips Simulation. In *Proceedings of SIGGRAPH '19 Talks, Los Angeles, CA, USA, July 28 - August 01, 2019,* 2 pages.
DOI: 10.1145/3306307.3328141

SIGGRAPH '19 Talks, Los Angeles, CA, USA

DOI: 10.1145/3306307.3328141

1 INTRODUCTION

Water surface behavior on a sub-centimeter scale is fascinating. Surface tension forces, dominating the dynamics, lead to formation of peculiar tendrils, oscillating droplets, and thin film. On contact with solid objects, one may observe formation of characteristic fluid channels and capillary wave patterns.

In a shot from *Alita: Battle Angel* (Figure 1) where the main character emerges from under water and moves right into the camera, capturing those small-scale effects was of utmost importance in order to sell the performance in an otherwise photo-real environment. The particular requirement was to accurately capture water dynamics as it sheds off the exposed skin parts and armor pieces. It was understood early on that the surface flow simulation, due to its computational intensity, would need to be decoupled from the bulk fluid in the pool and limited to parts of the character only.

There are a number of simulation techniques in the literature that handle surface tension. Mesh-based approaches [Da et al. 2016; Zhu et al. 2015, 2014] looked promising as they can capture thin features directly. However, they appeared challenging from artistic control and pipeline integration standpoints. SPH methods can produce convincing results, see e.g. [Akinci et al. 2013], but we were concerned about their scalability. A major limitation of Eulerian-based techniques is that they are primarily designed for bulk fluid simulation and require high resolution in order to capture thin features. Even so, we were inspired by the visual results demonstrated in [Wang et al. 2005], and chose to take a similar route, though using a particle-in-cell rather than a particle levelset representation.

2 APPROACH

We started with a standard variational FLIP solver implementation [Batty et al. 2007]. We treated surface tension explicitly via an inhomogeneous boundary condition in the pressure projection step equal to mean curvature of the fluid SDF multiplied by the surface tension coefficient (0.072N/m) as in [Wang et al. 2005]. Interestingly, even though dynamic viscosity of water is relatively small (10^{-3}Pa·s), it starts to play an important dampening role at a millimeter scale, so we have implemented the variational approach of [Batty and Bridson 2008] to account for that. Adding APIC transfer [Jiang et al. 2015] may also be desirable to improve the overall accuracy and stability of the simulation. This turned out to be enough to get the proper dynamics on the water-air interface, but unfortunately resulted in hydrophobic behavior with respect to collision objects, as shown in Figure 2a.

In order to make the water stick, the fluid SDF needed to be extrapolated into solid[1] prior to mean curvature computation at a *contact angle*, see [Wang et al. 2005], which is a physical parameter representing the amount of adhesion for a specific solid/fluid

[1] A typical FLIP implementation always has some sort of extrapolation of fluid into the solid for enforcing solid boundary condition. The results, however, may not be accurate enough for producing stable adhesion effects through mean curvature computation.

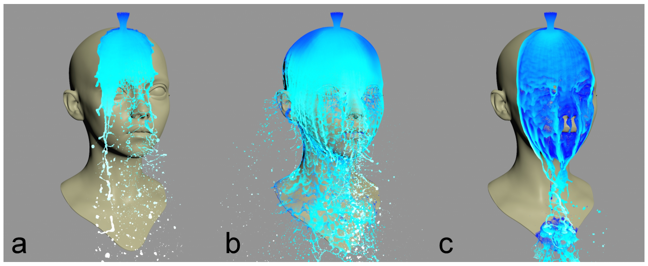

Figure 2: Pouring water over a character's face with: **(a)** no fluid SDF extrapolation; **(b)** 45° extrapolation; **(c)** 45° extrapolation and 10x viscosity multiplier in 1 voxel band near collision. Here we used a simulation voxel size of 0.4mm. ©Weta Digital Ltd 2019.

combination. Lower contact angles correspond to hydrophilic, and higher contact angles correspond to hydrophobic behaviors, with 180° being the highest value corresponding to no extrapolation. Figure 2b shows the result for a 45° contact angle.

Water interacts with rough surfaces and surfaces with tiny features such as pores and peach fuzz on skin in quite a different way compared to smooth ones such as porcelain. The tiny features are typically too small to be captured by the simulation grid and have a macroscopic friction effect. As a result, fluid does not slide off as readily. We modeled the effect by increasing fluid viscosity in a thin band around collisions objects. This also resulted in a more stable flow structure and yielded characteristic capillary wave patterns, as shown in Figure 2c. In general, viscosity adjustments can be made spatially varying to represent surfaces with varying roughness.

3 NOTES ON IMPLEMENTATION

[Wang et al. 2005] proposes a way for accurate extrapolation of fluid into solid with a given contact angle. Their approach, however, relies on the presence of a fairly high-quality fluid SDF representation due to tracking via particle levelsets. In our case, the SDF gets reconstructed from FLIP particles on each simulation step, and consequently exhibits poorer temporal coherence and contact with solid surfaces. As a result, the method of [Wang et al. 2005] was giving noisy results and subsequent inaccurate curvature estimations. We therefore developed a more robust extrapolation Algorithm 1 that runs over all simulation grid voxels, though we found that it suffices to only extrapolate four voxels deep into the solid. We additionally employed redistancing and then one iteration of Gaussian smoothing near the solid surface to further improve the result. Note that the algorithm only works for angles from 0° to 90°, which was sufficient for our application requiring strong adhesion.

Without careful handling of *moving* collision objects, the simulated water may gradually separate from the character and collapse due to surface tension, giving the character an undesirably hydrophobic look. We correct for this by ensuring that the velocity field enforcing the boundary conditions in the pressure and viscosity solves, and hence advecting fluid at the fluid-solid interface, corresponds exactly to the motion of the solid. Thus, after advection both are in sync. One way to guarantee this is to treat the motion of solids as piecewise linear from step to step (or frame to frame if sub-frame data is unavailable) and use forward differencing to calculate the velocity for boundary conditions.

Sample renders are shown in Figure 3. For the shot in Figure 1 we also trailed the simulation particles to create wetness maps and scattered stationary drops to further enhance the look.

4 LIMITATIONS AND FUTURE WORK

We found that running simulations at 0.4mm-1mm voxel sizes with 10-100 substeps per frame was needed to achieve believable and stable results. An average simulation of water covering a face would have 5-20M particles and needed to be run overnight. In the future we would like to explore performance optimizations and the possibility of using techniques such as [Stomakhin and Selle 2017] for limiting emission and simulation to areas of interest only.

ACKNOWLEDGMENTS

We would like to thank simulation, FX and leadership of Weta Digital for their support with the submission, and also Lightstorm Entertainment and Twentieth Century Fox Film Corporation for their permission to use the character model and the shot from *Alita: Battle Angel*.

REFERENCES

N. Akinci, G. Akinci, and M. Teschner. 2013. Versatile Surface Tension and Adhesion for SPH Fluids. *ACM Trans. Graph.* 32, 6, Article 182 (Nov. 2013), 8 pages.

C. Batty, F. Bertails, and R. Bridson. 2007. A Fast Variational Framework for Accurate Solid-fluid Coupling. *ACM Trans. Graph.* 26, 3, Article 100 (July 2007).

C. Batty and R. Bridson. 2008. Accurate Viscous Free Surfaces for Buckling, Coiling, and Rotating Liquids. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '08)*. Eurographics Association, 219–228.

F. Da, D. Hahn, C. Batty, C. Wojtan, and E. Grinspun. 2016. Surface-only Liquids. *ACM Trans. Graph.* 35, 4, Article 78 (July 2016), 12 pages.

C. Jiang, C. Schroeder, A. Selle, J. Teran, and A. Stomakhin. 2015. The Affine Particle-in-cell Method. *ACM Trans. Graph.* 34, 4, Article 51 (July 2015), 10 pages.

A. Stomakhin and A. Selle. 2017. Fluxed Animated Boundary Method. *ACM Trans. Graph.* 36, 4, Article 68 (July 2017), 8 pages.

H. Wang, P. Mucha, and G. Turk. 2005. Water Drops on Surfaces. *ACM Trans. Graph.* 24, 3 (July 2005), 921–929.

B. Zhu, M. Lee, E. Quigley, and R. Fedkiw. 2015. Codimensional non-Newtonian Fluids. *ACM Trans. Graph.* 34, 4, Article 115 (July 2015), 9 pages.

B. Zhu, E. Quigley, M. Cong, J. Solomon, and R. Fedkiw. 2014. Codimensional Surface Tension Flow on Simplicial Complexes. *ACM Trans. Graph.* 33, 4, Article 111 (July 2014), 11 pages.

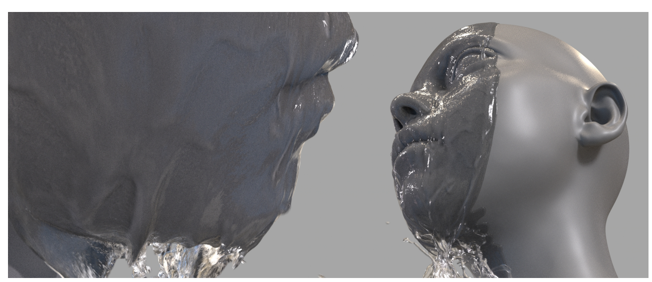

Figure 3: Close-up renders of the simulation shown in Figure 2c. ©Weta Digital Ltd 2019.

Algorithm 1 Extrapolation corresponding to contact angle θ

procedure EXTRAPOLATE(**float&** fluidSDF, **float** solidSDF)
  if solidSDF ≤ 0 **and** fluidSDF $< 4 *$ voxelSize **then**
    if $-$solidSDF $>$ fluidSDF **then** fluidSDF $\leftarrow -$solidSDF
    $d \leftarrow \sqrt{\text{fluidSDF}^2 - \text{solidSDF}^2}$
    fluidSDF $\leftarrow \sin\theta * d + \cos\theta *$ solidSDF