Integrated Surface Dynamics for SPH-based Fluid Simulation

(a) $t_{sim} = 3\,s$ (b) $t_{sim} = 16\,s$ (c) $t_{sim} = 16\,s$ (detergent in blue)

Figure 1: *The pan's surface is cleansed from grease (orange) due to detergent concentration (blue in 1(c)) on the fluid's surface.*

Abstract

Surface dynamics play an essential role in fluid simulations. A vast number of effects including wetting of surfaces, cleansing, and foam dynamics are based on surface-surface and surface-bulk interactions, which in turn rely on a robust surfaces computation.
In this paper we introduce a conservative Lagrangian simulation of surface dynamics based upon incompressible smoothed particle hydrodynamics (SPH). The key concept of our approach is to realize surface dynamics using an implicit definition of the fluid's (free) surface. Our algorithm assigns a scalar particle value estimating the surface area of each particle. Based on this implicit surface definition, surface dynamics are incorporated using scalar particle quantity concentrations in the bulk and at the surface. Transport of quantities between bulk and surface is simulated by introducing a symmetric, conservative SPH-based formulation of the Langmuir-Hinshelwood adsorption mechanism.
We demonstrate two different types of surface-relevant effects: diffusion on surfaces and reaction kinetics. Reaction kinetics model the dynamics at fluid interfaces based on transition rules between different quantities in order to realize effects like cleansing.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: fluid simulation, SPH, surfaces dynamics, surfactants

1 Introduction

Lagrangian fluid simulation using smoothed particle hydrodynamics (SPH) is a well established method. Major advantages of this approach are mass conservation and the simple handling of free surfaces. There is a vast SPH literature, e.g. providing extensions of the SPH simulation model in order to realize complex dynamic boundaries [Akinci et al. 2012b], bulk transport of soluble substances [Kristof et al. 2009], multi-phase simulations [Solenthaler and Pajarola 2008], tracking [Adams et al. 2007] and capturing of surfaces [Akinci et al. 2012a].

Effects on complex fluid surfaces and interfaces are a basic element of many applications in Computer Graphics. Several SPH-based methods have been introduced in this respect, e.g. the convection of diffuse materials [Losasso et al. 2008], dynamics of foam [Ihmsen et al. 2012], and multiple-phase scenarios [Losasso et al. 2006; Solenthaler and Pajarola 2008; Akinci et al. 2012b].

Surface dynamics supplement fluid dynamics, e.g. on small scales the can even dominate convective flux. However, considering conservative Lagrangian surface dynamics, there is little research until now. Particle level set (PLS), for example, use particles to track flow details, e.g. Losasso et al. [2008] use a PLS-SPH combination to model surface effects like foam and spray. However, PLS in general cannot fully conserve mass at free surfaces and Losasso et al. [2008] did not incorporate transport mechanisms between surface and bulk. Semi-Lagrangian contouring methods, on the other hand, are used for tracking of surface characteristic, e.g. color and texture coordinates, like Bargteil et al. [2006], but maintaining an explicit polygonal mesh is expensive and surface dynamics are complicated to integrate.

Thus, to the best of our knowledge, for SPH-based fluid simulations there has been no formulation for conservative transport mechanisms within a fluid's surface and between surface and fluid volume (bulk), which, for example, are required to model the dynamics of detergents as shown in Fig. 1.

Detergents themselves belong to the wider class of materials called surfactants (surface active agents), which exhibit different behavior on surfaces and in the bulk, i.e. often they get attracted to the surface, rather than remaining in the bulk. Surfactants are widely used in chemical and biological applications, including wetting phenomena [Fell et al. 2011], cleansing action [Rosen and Kunjappu 2004], and drug delivery [Dekker 2002].

Note, that we will use the notion of surface dynamics and surfactants dynamics interchangeably throughout this paper, even though surface dynamics could involve further concepts.

Contribution: In this paper we present a novel SPH-based fluid simulation approach with integrated surface dynamics. The main contributions of our work can be summarized as follows:

- We introduce a reliable computation of surface areas for individual particles, which is the essential foundation for any surface dynamics (Sec. 5).
- We propose a consistent coupling of bulk and (free) surface based on physical entities (Sec.4).
- Based on the coupling, a conservative formulation of surface dynamics is presented (Sec. 6).
- We apply our SPH-based model to simulate dynamics of diffusion effects and of a specific example of reaction kinetics, i.e. detergents in combination with a cleansing action on fluid-rigid interfaces (Sec. 7).

In Sec. 8 we show the versatility and the performance of our fully GPU-based implementation in various simulation scenarios. Sec. 9 draws some final conclusions.

2 Related Work

Since its introduction [Gingold and Monaghan 1977; Lucy 1977], SPH has been heavily used for simulation of fluids and we refer the reader to the surveys of Monaghan [2005] and Koumoutsakos et al. [2008] for a general overview.

2.1 Convection-Diffusion

In Computer Graphics, compact smoothing kernels [Müller et al. 2003], adaptive sampling [Adams et al. 2007; Solenthaler and Gross 2011; Orthmann and Kolb 2012] and parallelization [Kolb and Cuntz 2005; Goswami et al. 2010; Ihmsen et al. 2011b] are essential in order to reduce overall simulation times. Incompressibility is enforced by using a predictive-corrective formulation [Solenthaler and Pajarola 2009] in combination with adaptive time steps [Ihmsen et al. 2010]. In order to prevent undesirable pressure variations in combination with large density contrasts [Solenthaler and Pajarola 2008] or with rigid bodies [Akinci et al. 2012b] it is preferable to replace the density summation [Müller et al. 2003] by computation of a particle's number density [Hu and Adams 2006]. Beside convection, a diffusive flux simulates thermodynamics [Müller et al. 2005] and defines transport of sediments [Kristof et al. 2009] or soluble substances [Cleary and Monaghan 1999; Monaghan 2005].

2.2 Surface Definitions

Interfaces can be identified using color gradient methods [Müller et al. 2003; Keiser et al. 2006; Akinci et al. 2012a]. In order to stabilize the formulation of surface tension, Solenthaler et al. [2008] employed an additional normalization. However, a second phase is required for gradient-based methods. As stated by Becker and Teschner [2007], derivatives of the color-gradient are sensitive to irregular particle structures. Thus, using ghost particles [Schechter and Bridson 2012] in order to emulate a second phase would result in unstable surface definitions. Our approach describes a stable surface definition which works on both, interfaces and free surfaces. For a more stable representation of free surfaces, signed distance functions are used [Solenthaler et al. 2007b; Adams et al. 2007]. These functions are employed to render smooth surfaces [Zhu and Bridson 2005; Solenthaler et al. 2007a; Onderik et al. 2011] and enable the computation of anisotropy information [Yu and Turk 2010]. However, these methods aim for surface reconstruction or capturing [Hieber and Koumoutsakos 2005]. In contrast, we define an implicit surface which is capable of simulating surface dynamics. Losasso et al. [2008] propose particle level-set methods in order to track the surface of a fluid. However, our approach has the advantage that the surface definition is directly deduced from SPH particles, avoiding additional surface representations and particle re-seeding.

2.3 Multi-Phase Effects

SPH has been used in a variety of multi-phase effects, ranging from temperature modulated viscosity for lava flows [Stora et al. 1999], melting and solidification of objects [Solenthaler et al. 2007a] and melting and freezing of ice objects including a heat transfer steering the phase transition [Iwasaki et al. 2010]. Beside phase transitions, simulations of trapped air [Müller et al. 2005], bubbles [Hong et al. 2008; Ihmsen et al. 2011a] and diffuse materials [Losasso et al. 2008; Ihmsen et al. 2012] have enriched SPH effect simulations which circumvent problems induced by large density contrasts [Solenthaler and Pajarola 2008]. Other research deals with wetting effects for granular materials [Rungjiratananon et al. 2008; Lenaerts and Dutré 2009] and physically-based erosion of terrains [Kristof et al. 2009]. A fluid-rigid coupling has been achieved by using distance fields [Harada et al. 2007], via direct forcing [Becker et al. 2009], or more consistently by considering relative contributions of inhomogeneously sampled rigid particles [Akinci et al. 2012b]. However, none of the proposed methods simulates a quantity transport on phase interfaces. In computational physics, Adami et al. [2010] presented an SPH-based approach for surfactant dynamics, which however, is not applicable to free-surface scenarios and requires complex corrections of interface dynamics. In contrast, we utilize an implicit signed distance function [Zhu and Bridson 2005] in order to explicitly integrate a surface delta-function, which leads to consistent bulk and surface representations and supports simple dynamics on free surfaces as well as phase interfaces.

3 SPH Background

In this section we give a short introduction to the fundamental building blocks as necessary for a quantity transport on fluid surfaces.

3.1 Field Reconstruction

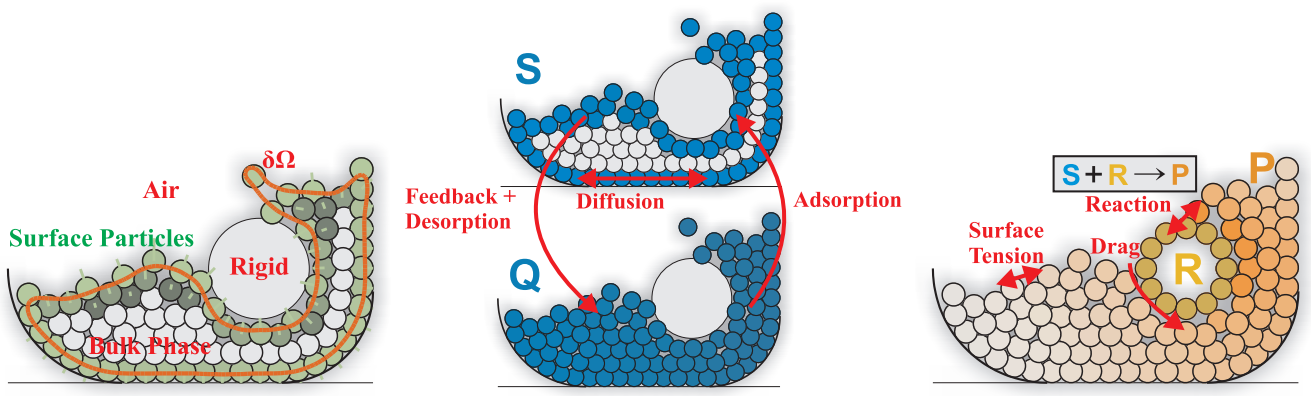
In SPH, as introduced by Gingold and Monaghan [1977] and Lucy [1977], a quantity field S is reconstructed over all particles j in the local neighborhood of sampling position $\mathbf{x}$:

$$S(\mathbf{x}) = \sum_j S_j \, V_i \, W_j(\mathbf{x}) \tag{1}$$

where $W_j(\mathbf{x}) = W(r,h) = W(||\mathbf{x} - \mathbf{x}_j||, h)$ (or in short W_{ij} for $S_i = S(\mathbf{x}_i)$) is a radial symmetric smoothing kernel [Müller et al. 2003] with compact support h and V_i is a particle's dynamic volume. Assuming constant rest density ρ_0 [Akinci et al. 2012b; Solenthaler and Pajarola 2008] leads to the concept of particle number density $n_i = \sum_j v_j \, W_{ij}$. As a result, $V_i = \frac{m_i}{\rho_i} = \frac{\rho_0 \, v_i}{\rho_0 \, n_i} = \frac{v_i}{\rho_i}$, where v_i is a particle's constant rest volume. Usually, a particle's number density is $n_i \approx 1$ for incompressible particle configurations. Thus, to correct the neighbor deficiency close to free surfaces we enforce $n_i \geq 1$. However, in order to establish quantity fields on a surface, we still need to model the singularity at the surface.

3.2 Implicit Surface Definition

In general, the free fluid-air interface $\partial\Omega := \{\mathbf{x} \,|\, \phi(\mathbf{x}) = 0\}$ is defined as the zero iso-contour of the following signed distance func-



(a) Computation of Surface Area (b) Integrated Surface Dynamics (c) Surface Effects

Figure 2: *Simulation overview: fluid particles compute the surface area a_i (green) as defined by $\delta\Omega$ (see Fig. 2(a)). Surface particles, i.e. particles for which $a_i > a_{\min}$, constitute a surfactant field S on the surface (see Fig.2(b)), which changes due to diffusion along the surface and which interacts with a surfactant field Q in a fluid's bulk due to adsorption and desorption. Surface particles also feedback surfactant concentration when they return to the bulk. As shown in Fig. 2(c), in our simulation, surfactants S react with substances R (yellow) carried by rigid particles resulting in products P (orange) which dissolve into the bulk. Additionally, surfactant concentration influences surface tension and fluid drag.*

tion [Zhu and Bridson 2005]:

$$\phi(\mathbf{x}) = ||\mathbf{x} - \mathbf{m}(\mathbf{x})|| - d, \tag{2}$$

where d is the desired distance between interface and particles, which in our simulation equals the particle's radius, and $\mathbf{m}(\mathbf{x})$ is the center of the local iso-density distribution [Onderik et al. 2011]:

$$\mathbf{m}(\mathbf{x}) = \frac{\sum_j \mathbf{x}_j V_j W_j(\mathbf{x})}{\sum_j V_j W_j(\mathbf{x})}.$$

Still, we need to find a representation which is capable of simulating dynamics on the surface and of robust surface computation.

4 Coupling of Bulk and Surface

Some substances, like surface active agents, i.e. surfactants, which are immersed in a fluid's bulk adhere to a fluid's surface. Consequently, to simulate surface dynamics, as summarized in Fig. 2, we need a robust and reliable estimation of the surface area. According to Bertalmio et al. [2001], it is advantageous to embed a lower dimensional surface in a higher dimensional volume. Hence, our idea is to sample the iso-contour as defined in the last section with SPH particles resulting in surface areas a_i. Similar to Adami et al. [2010], the underlying Lagrangian SPH simulation then acts as a "particle-pool" but for quantity fields defined on surfaces, as shown in Fig. 2(a).

Particles are divided into two groups. Particles within a small layer around the surface, which we call *surface particles* (green) and all other particles, called *bulk particles* (white). Surface particles have a double role during simulation (Fig. 2(b)): they constitute a quantity field S on the surface and at the same time, like bulk particles, they contribute to a quantity field Q. Even if our method is not restricted to scalar quantities, we consider $Q_i = \frac{\overline{Q}_i}{v_i}$ given in $\left[\frac{mol}{m^3}\right]$ and $S_i = \frac{\overline{S}_i}{a_i}$ given in $\left[\frac{mol}{m^2}\right]$ as concentrations of a soluble substance, e.g. concentration of detergent with a constant total molar mass of $\sum_i \overline{Q}_i + \overline{S}_i$. Please note the difference between mass $\overline{S}_i$ and concentration S_i.

While $\overline{Q}$ is trivially conserved in Lagrangian systems, including matter diffusion [Monaghan 2005] and passive transport with fluid particles [Solenthaler and Pajarola 2009; Akinci et al. 2012b], a quantity flux from Q to S and inside S requires special symmetrization mechanisms (see Sec. 6). Beside surface dynamics, our method is applicable for dynamics on phase interfaces as well, e.g. enabling a cleansing action of S with a substance R given in $\left[\frac{mol}{m^2}\right]$ on rigid surfaces (see Fig.2(c)). Additionally, a surfactant concentration on the surface may alter fluid drag as induced by rigid objects or surface tension, thus directly affects the underlying flow field.

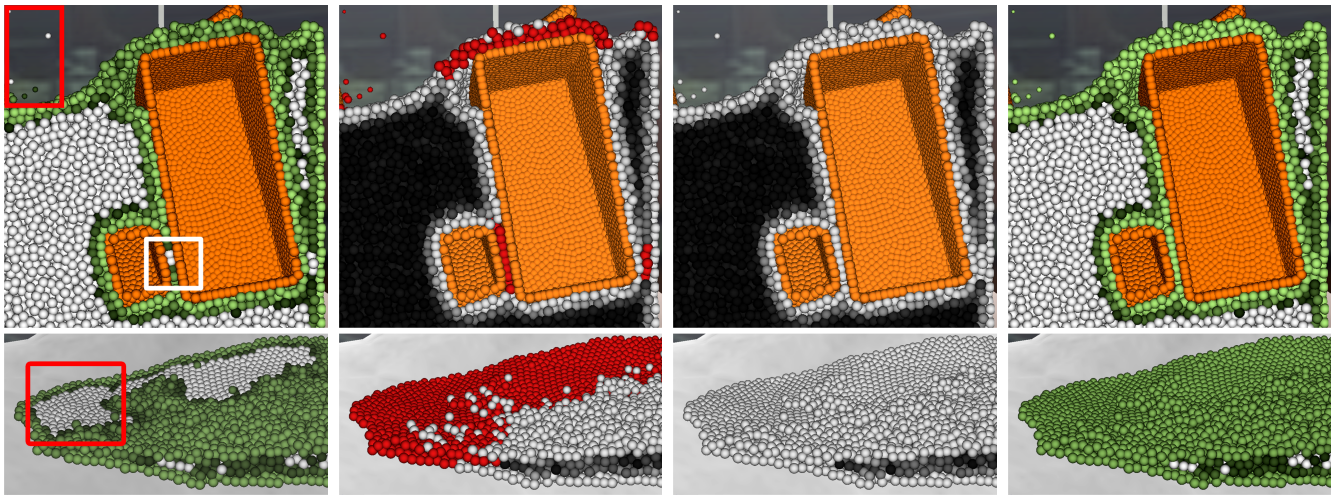
Unlike particle volumes v_i, a particle's fraction of surface area a_i is not constant due to internal and external forces changing the surface topology i.e. particles may move closer to or further away from the surface. Consequently, in order to reconstruct concentration fields S, we need to compute the surface area at particle locations as described in the following section.

5 Computation of Surface Area

The strength of our implicit surface definition is that quantity fields S on a surface can be reconstructed over the fluid volume using Eq. (1) leading to a consistent representation of bulk and surface. However, relative concentrations S_i are defined with respect to a particle's fraction of the surface area. In SPH, a surface integral over the fluid volume Ω reads

$$a(\mathbf{x}) = \int_\Omega \delta(\phi(\mathbf{x}'))\, W(||\mathbf{x} - \mathbf{x}'||, h) d\,\mathbf{x}', \tag{3}$$

where $\delta(\phi(\mathbf{x}'))$ is the surface delta function and where we apply the assumption that $||\nabla\phi(\mathbf{x})|| = 1$ common to level-set approaches [Osher and Fedkiw 2002], which for a regularly sampled and incompressible SPH is a true assumption as well. As shown in Fig. 3, a discretization of the delta function as described in Sec. 5.1 results in a narrow band of contributing particles. Each such particle represents a fraction of surface area, depending on its distance to the surface as described in Sec. 5.2. In order to enable stable dynamics (see Sec. 6) we correct delta values in singular regions as described in Sec. 5.3 and make use of temporal coherence as described in Sec. 5.4.



(a) Erroneous a_i (b) Singularity (c) Corrected δ_i (d) Corrected a_i

Figure 3: *Computation of surface area for two particle configurations showing surface particles (green), bulk particles (white) and rigid particles (orange). No check for singularity and no normalization lead to erroneous area values a_i (highlighted regions in 3(a)). Instead, a detection of singular regions (red in 3(b)) enables a correction of delta values δ_i (increasing from black to grey in 3(b) and 3(c)), which are integrated to area values a_i (3(d)) using a corrected SPH interpolation (Eq.* (5)*).*

5.1 Approximation of the Surface Delta Function

In order to determine the contribution of particles to the (free) surface, we smear out the iso-contour $\delta\Omega$ over the local fluid volume using SPH smoothing kernels. However, using a standard SPH kernel to approximate a surface delta function imposes several issues:

1. In combination with small support radii, Eq. (2), may return slightly fluctuating distance values ϕ_i. The resulting time-variance would then additionally be emphasized by higher order kernels.
2. Unlike level set methods, the surface $\partial\Omega$ is only sampled in the fluid $\Omega-$ as shown in Fig. 4.

Regarding stable distance and quantity fields, we aim at low-order kernels, additionally reducing computation times. The lack of particles in $\Omega+$ requires an adjustment of the smoothing kernel in order to satisfy delta-function properties. Assuming that the surface is locally planar, which is a true approximation in the limit $\lim_{h\to 0}$, the contribution of fluid particles simply doubles, in order to compensate contributions from non-existing particles in Ω_+. Thus, we approximate the surface delta function by a half-sided tent-kernel:

$$\delta(\phi) = \begin{cases} 2\,(1+\frac{\phi}{d}) & \text{if } \phi < 0 \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

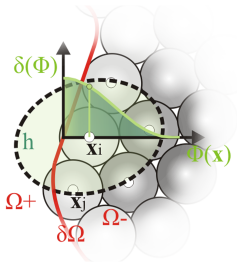


Figure 4: *Sampling of the surface using a half-sided smeared out delta function approximation.*

which depends only on the distance to the iso-contour at $\phi(\mathbf{x}) = 0$ for which by definition $\phi(\mathbf{x}) \geq -d$ holds. During simulation we integrate discrete values $\delta_i = \delta(\phi(\mathbf{x}_i))$ evaluated at particle locations in order to compute a surface area.

5.2 Interpolation of Particle Areas

Particles close to the surface have a high contribution. The closer a particle, the more it contributes to the surface (color coded from dark to light green in Fig. 3(d)). In order to determine how much of the surface area is represented by a particle, we have to integrate the surface delta function as derived in the last section over a particle's support h (see Fig. 4). In detail, the corresponding SPH summation for Eq. (3) yields

$$a_i = \sum_j \delta_j \, V_j \, \overline{W}_{ij}, \tag{5}$$

where $\delta_i = \delta(\phi(\mathbf{x}_i))$ is the delta function evaluated at position $\mathbf{x}_i \in \Omega-$. Due to the empty neighborhood in $\Omega+$ we have to correct the resulting volume deficiency by using an adapted weighting kernel [Bonet and Kulasegaram 2002]:

$$\overline{W}_{ij} = \frac{W_{ij}}{\sum_j V_j \, W_{ij}}.$$

Such a normalization also ensures that area values smoothly change between neighboring particles. Due to the definition of the distance function (Eq.(2)), a disturbance of regular particle structures, as caused by the flow-field, leads to small area values in the fluid bulk. Thus, we consider the particle to be a bulk particle (white particles in Fig. 3(d)) as long as a particle's area is below or equal $a_{\min}$. However, values δ_i may also be incorrect in under-resolved regions as described in the next section.

5.3 Thin Fluid Sheets

Quantity fields on the surface require stable area values. However, within thin fluid sheets regular but deficient particle neighborhoods can emerge which might lead to erroneous area values as shown in

Fig. 3(a). Singular cases can be identified by setting a lower threshold for the number of neighboring particles, but dense and strongly anisotropic distributions remain unidentified. A better description for local particle distributions is the weighted covariance matrix of a particle i, given as [Yu and Turk 2010]

$$\mathbf{C}_i = \frac{\sum_j (\mathbf{x}_j - \mathbf{m}_i)\,(\mathbf{x}_j - \mathbf{m}_i)^T\, W_{ij}}{\sum_j W_{ij}} \in \mathbb{R}^{3x3}. \quad (6)$$

Since we want to detect singular particle structures, the determinant of a particle's covariance matrix naturally reveals flat or planar particle neighborhoods. Accordingly, if $|\mathbf{C}_i| < \epsilon_c$ or if a particle i has less than 30 neighbors (red particles in Fig. 3(b)) we will set $\delta_i = \delta_{\max}$ leading to a corrected approximation of delta values as shown in Fig. 3(c).

5.4 Temporal Coherence

In order to stabilize quantity fields, particle areas should change smoothly over time. Fortunately, due to the temporal coherent nature of particle systems, we can apply a temporal smoothing of particle areas, which is computed as

$$\delta_i(t+1) = (1-\kappa)\delta_i(t) + \kappa\delta_i(t+1)^*, \quad (7)$$

where $\delta_i(t+1)^*$ is the approximated surface delta function according to Eq. 4 and where κ defines the rate of adjustment to new values $\delta_i(t+1)^*$. In all our examples $\kappa = 0.1$ resulted in smooth transitions.

In regions with little velocity divergence, i.e. where $\nabla \cdot \mathbf{u}_i < \epsilon_u$, we recompute $\delta_i(t+1)^*$ only every 10-th frame. This is valid, since in these regions $\frac{\partial}{\partial t}\delta_i = 0$ holds (see App. A).

6 Conservative Surface Dynamics

While mass transport in a fluid's bulk has been subject of many applications, a coupling of the bulk with a surface and dynamics on fluid surfaces have not been investigated yet. Depending on the simulated physics, interfacial quantity fields change due to a mass flux between neighboring particles:

$$\frac{\partial}{\partial t} S_i = \frac{1}{a_i} \sum_j \Gamma_{i \leftarrow j},$$

where $\Gamma_{i\leftarrow j}$ represents a mass flux from particle j to particle i. Most importantly, such pairwise contributions must be symmetric in order to satisfy mass conservation, i.e. $\Gamma_{i\leftarrow j} = -\Gamma_{j\leftarrow i}$. In this section we therefore develop the building blocks for a conservative transport between bulk and surface and for dynamics on surfaces as well as phase interfaces for which we give examples in the context of surfactant dynamics.

6.1 Consistent Transport between Bulk and Surface

Dynamics of a fluid's bulk and surface are strongly interconnected. For example, surfactants like detergents, which adhere to a fluid's surface, usually originate from a fluid's bulk and may return to it. Thus, mass is transferred between two different kinds of particle fields, surface particles i and bulk particles j (which might be one and the same particle, i.e. $i = j$, but in different roles). Consequently, a transport mechanism between bulk and surface is not symmetric per se. To ensure conservation, we compute a quantity flux with focus on surface particles i and assume the opposite flux for bulk particles j:

$$\frac{\partial}{\partial t} S_i = \frac{1}{a_i} \sum_j \Gamma_{i\leftarrow j}, \quad \text{and} \quad \frac{\partial}{\partial t} Q_j = \frac{1}{v_j} \sum_i -\Gamma_{i\leftarrow j}. \quad (8)$$

where $\Gamma_{i\leftarrow j}$ are defined with respect to absolute quantities, i.e. $\overline{S}_i = a_i\, S_i$ and $\overline{Q}_i = v_i\, Q_i$. That way we take as much of a substance from the fluid bulk as the surface receives and vice versa.

6.1.1 Adsorption and Desorption

The driving mechanism for our simulation of surface dynamics is the well known Langmuir-Hinshelwood mechanism [Rosen and Kunjappu 2004] which describes how much of a so-called "free" substrate or target Q_j in a carrier fluid is adsorbed on a surface as shown in Fig. 5. The net rate of adsorbed substrates at surface particle i mainly depends on the presence of targets at neighboring bulk particles j:

$$\Gamma^a_{i\leftarrow j} = \left[\sigma_a\, \overline{Q}_j\, (\overline{S}_0 - \overline{S}_i) \;-\; \sigma_f \overline{S}_i\right] V_j W_{ij}, \quad (9)$$

where σ_a defines the speed of adsorption and where the total number of available capture sites $\overline{S}_0$ limits the adsorption process. Over time adsorbed targets constantly dissociate from the surface as controlled by σ_f. The desorption rate usually is much smaller than the adsorption rate, i.e $\sigma_a \gg \sigma_f$. The larger σ_f, the less saturated the surface becomes. Fig. 5 shows the importance of adsorption for a falling drop of detergent.

6.1.2 Feedback

Beside physically-based mass transport between bulk and surface, we account for particles which move away from the surface, i.e. for which a_i drops below $a_{\min}$. Such (former) surface particles have to transfer mass back to the bulk in order to conserve the total number of surfactant molecules in the system. Naturally, such a feedback from surface particles i to bulk particles j is a much faster desorption process. Thus, we return mass by increasing σ_f and setting $\sigma_a = 0$ for particles returning to the bulk. In order to avoid numerical problems due to decreasing area values during feedback, we keep $a_i = a_{\min}$ until all surface quantity is transfered back to nearby bulk particles. Particles returning to the bulk are excluded from surface dynamics.

6.2 Unified Free-Surface and Interface Dynamics

In SPH, contributions between bulk particles usually are symmetrized using gradient approximations [Colin et al. 2006] or integral approximations [Cleary and Monaghan 1999] making use of constant rest volumes. Due to non-constant particle areas, standard symmetrization techniques do not lead to symmetric formulations for surface dynamics. Therefore, we enforce conservation explicitly by averaging pairwise contributions between interface particles, i.e. using the identity $\Gamma_{i\leftarrow j} = -\Gamma_{j\leftarrow i}$:

$$\frac{\partial}{\partial t} S_i = \frac{1}{a_i} \sum_j \Gamma_{i\leftarrow j} = \frac{1}{a_i} \sum_j \frac{1}{2} [\Gamma_{i\leftarrow j} - \Gamma_{j\leftarrow i}], \quad (10)$$

where i and j might also be part of different phases. As specific kinds of surface dynamics we introduce diffusion on a surface (Sec. 6.2.1) and a reaction kinetics (Sec. 6.2.2).

6.2.1 Diffusion

Once surfactants have been adsorbed at the surface, they are transported along the tangential direction of the surface. Since S is nonzero only at the surface, we can approximate a diffusion of surfactants as (see App.B)

$$\Gamma^d_{i\leftarrow j} = \sigma_d\, (S_i - S_j)\; (a_i V_i + a_j V_j)\; \frac{\partial}{\partial r} W_{ij}, \quad (11)$$

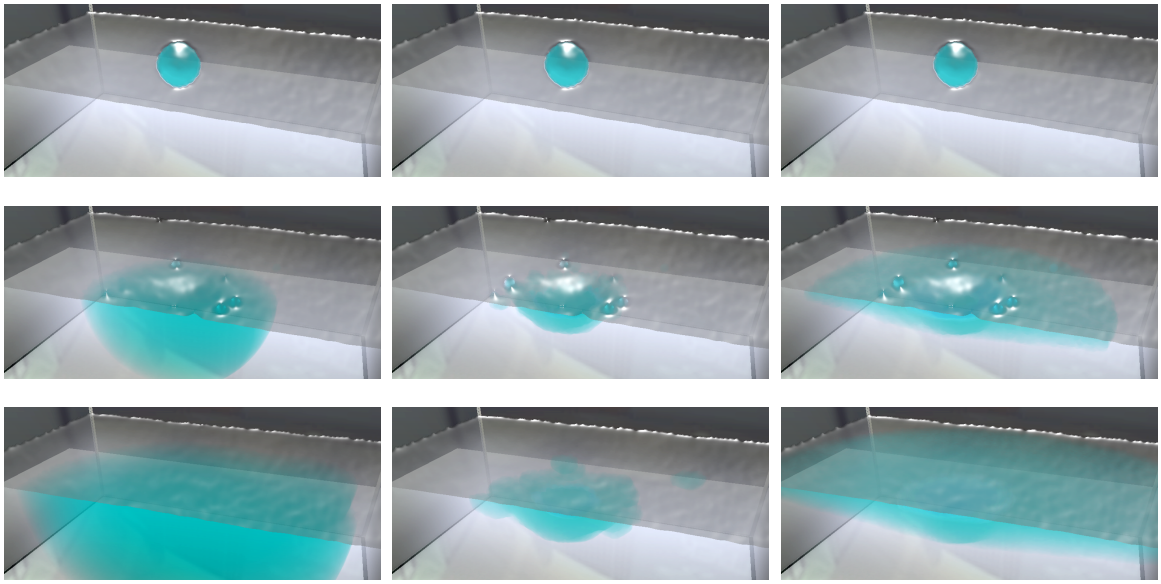


Figure 5: *A falling drop of detergent shown for three different time steps (rows) and for different adsorption σ_a and diffusion constants σ_d (columns). No adsorption $\sigma_a = 0$ leads to diffusion in the bulk (left column), while $\sigma_a > 0 \wedge \sigma_d = 0$ leads to a local concentration on the surface (middle column). In contrast, $\sigma_a > 0 \wedge \sigma_d > 0$ results in homogeneous concentration of surfactant on the surface (right column).*

where σ_d is an isotropic diffusion constant. Please note that i, j need to be restricted to surface particles in order to avoid diffusion into the fluid bulk, leading to a homogeneous surfactant concentration on the surface (see Fig. 5).

6.2.2 Reaction Kinetics

In reaction kinetics like the cleansing action of detergents, as shown in Fig. 1, surfactants S react with a substance R on rigid surfaces given in $\left[\frac{mol}{m^2}\right]$ forming products or micelles P which directly dissolve into the bulk given in $\left[\frac{mol}{m^3}\right]$. We describe reaction kinetics such a cleansing processes, as a simple forward reaction [Berthier and Silberzan 2009], consisting of a single bimolecular reaction $\alpha_S \overline{\overline{S}} + \alpha_R \overline{\overline{R}} \rightarrow \alpha_P \overline{\overline{P}}$, where α_S, α_R and α_P are stoichiometric coefficients, which define how much reactant molecules $\overline{S}$ and $\overline{R}$ are irreversibly consumed in order to form product molecules $\overline{P}$, e.g. micelles in case of cleansing. Here, $\overline{\overline{S}}, \overline{\overline{R}}, \overline{\overline{P}}$ stand for a constant molar mass of the respective substance. By enforcing symmetry as defined via Eq. (8), the reaction rate for fluid particles i and rigid particles k is

$$\frac{1}{\alpha_P}\frac{\partial}{\partial t}\overline{P}_i = -\frac{1}{\alpha_S}\frac{\partial}{\partial t}\overline{S}_i = -\frac{1}{\alpha_R}\frac{\partial}{\partial t}\overline{R}_k = \sum_k \Gamma^r_{i \leftarrow k}.$$

In our examples we set $\alpha_S = \alpha_R = \alpha_P = 1$. Please note that, since the product mass, e.g. micelles, directly dissolve into the fluid bulk, $\overline{P}_i = v_i\, P_i$ holds, whereas $\overline{S}_i = a_i\, S_i$ and $\overline{R}_k = a_k\, R_k$ are defined with respect to areas of fluid particles a_i and areas of rigid particles a_k. Since rigid particles directly sample a rigid's surface, we pre-compute a_k with Eq.(5) using $\delta_k = \delta_{\max}$.

The speed of the reaction is defined via a simple rate law and is proportional to $\sigma_r \overline{S}\,\overline{R}$, i.e. it linearly depends on the rate constant σ_r and the total number of molecules of both reactants. Thus, the reaction is formulated as

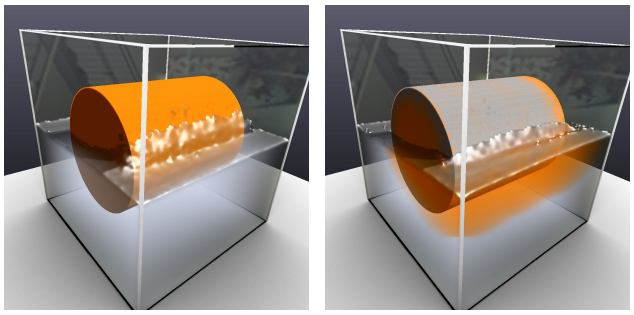
$$\Gamma^r_{i \leftarrow k} = \sigma_r \overline{S}_i \overline{R}_k \, \frac{1}{2}(V_i + V_k)\, W_{ij}, \qquad (12)$$

by using Eq. (10).

Please note that for adjacent phases iso-contours do in general not match exactly (see Eq.(2)). However, pairwise averaging of contributions as done in Eq. (12) naturally corrects such a discontinuity and leads to a unified simulation capable of computing dynamics for free surfaces as well as phase interfaces.

7 Implementation Details

For clarity we outline our fully GPU-based implementation of surface dynamics in Alg. 1 and refer to SPH literature for mass transport of Q in the bulk, including neighbor finding [Green 2009], an incompressible convection [Solenthaler and Pajarola 2009] coupled with rigid objects [Akinci et al. 2012b] and matter diffusion [Monaghan 2005]. Surfactant concentration linearly alters the surface tension coefficient as proposed by Becker and



(a) no detergent (b) with detergent

Figure 6: *Rotating cylinder before (6(a)) and after (6(b)) drops of detergent have been added. No detergent concentration leads to strong drag and surface tension effects corresponding to a real surface effect, as described by Fell [2011], while detergents clean the cylinder's surface and reduce fluid drag.*

Algorithm 1: Data-Parallel Surface Dynamics

Delta Function

foreach *fluid particle* i *where* $\nabla \cdot \boldsymbol{v}_i(t) > \epsilon_u$ ***inparallel*** **do**
  compute $\phi_i(t)$ using a support of $2.5h$ (Eq. (2))
  estimate $\delta_i(t)^*$ (Eq. (4))

foreach *fluid particle* i *where* $\nabla \cdot \boldsymbol{v}_i(t) > \epsilon_u$ ***inparallel*** **do**
  compute $\mathbf{C}_i(t)$ using a support of $2.5h$ (Eq. (6))
  if ($|\mathbf{C}_i(t)| < \epsilon_c ||$ NumNeighs < 30) $\delta_i(t)^* = \delta_{\max}$

foreach *fluid particle* i ***inparallel*** **do**
  update $\delta_i(t)$ (Eq. (7))

Surface Area

foreach *fluid particle* i ***inparallel*** **do**
  compute $a_i(t)$ (Eq. (5))
  if ($a_i(t) > a_{\min}$) mark i as *surface particle*

Adsorption and **Feedback** (Eq. (9))

foreach *surface particle* i ***inparallel*** **do**
  $\frac{\partial}{\partial t} S_i = \frac{1}{a_i} \sum_j \Gamma^a_{i \leftarrow j}$

foreach *fluid particle* i ***inparallel*** **do**
  $\frac{\partial}{\partial t} Q_i = \frac{1}{v_i} \sum_j -\Gamma^a_{j \leftarrow i}$

Diffusion (Eq. (11))

foreach *surface particle* i ***inparallel*** **do**
  $\frac{\partial}{\partial t} S_i \mathrel{+}= \frac{1}{a_i} \sum_j \Gamma^d_{i \leftarrow j}$

Reaction (Eq. (12))

foreach *surface particle* i ***inparallel*** **do**
  $\frac{\partial}{\partial t} S_i \mathrel{+}= \alpha_S \frac{1}{a_i} \sum_k \Gamma^r_{i \leftarrow k}$

foreach *rigid particle* k ***inparallel*** **do**
  $\frac{\partial}{\partial t} R_k = \alpha_R \frac{1}{a_k} \sum_k \Gamma^r_{i \leftarrow k}$

foreach *fluid particle* i ***inparallel*** **do**
  $\frac{\partial}{\partial t} P_i = \alpha_P \frac{1}{v_i} \sum_k -\Gamma^r_{i \leftarrow k}$

Scene	**"Pan"**	**"Flooding"**	**"Cylinder"**	**"Kinect®"**
σ_r/σ_d	10/10	70/10	40/10	40/10
σ_a/σ_f	10/0.1	10/0.1	10/1.0	10/0.1
Snapshot t_{sim}	23 sec.	30 sec	3 sec	20 sec
Avg.δt [ms]	2	1.5	2	1.5
# fluid ptcls	541k	1.1M	891k	750k
# surface ptcls	142k	200k	152k	300k
Delta Func. [ms]	14.2	30.2	21.7	60.0
Surface Area [ms]	7.4	14.8	12.8	9.2
Adsorption [ms]	10.1	19.0	16.1	14.9
Diffusion [ms]	1.9	2.8	2.2	3.6
Reaction [ms]	9.0	17.5	15.2	12.0
Time per Frame	**160**	**261**	**227**	**252**

Table 1: *GPU timings (in msec) for operations as given in Alg.1.*

Teschner [2007] and the drag coefficient similar to Akinci et al. [2012b] resulting in wetting effects as shown in Fig.6. We integrate a surfactant flux using an explicit Euler integration with adaptive time-steps [Ihmsen et al. 2010] which is sufficient for surface dynamics, since in scales of meters, diffusion always takes place at much smaller scale than convection. In case surfactant concentration does not influence convection, one may as well pre-compute a particle movement and use it as a base simulation for surface dynamics.

Beside attributes required for convection, we store surfactant masses $\overline{Q}$ and $\overline{S}$, delta values δ_i and areas a_i as additional attributes per fluid particle. To avoid numerical problems with 32-bit floating point numbers we normalize the actual molecular mass of a particle with respect to a maximum mass $\overline{S}_{\max}$. Consequently, for $\overline{S}_{\max} = e^8$ a concentration of $S = 1.0$ represents a molecular mass of $e^8[\frac{mol}{m^2}]$.

In all our examples, we set $h = 2d$, where d is a particle's radius, which results in a rest distance of $0.92h$ between particles and

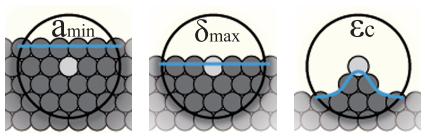


Figure 7: *Constants are computed for reference particles (white) in user-defined particle configurations.*

around 30 neighbors per particle in combination with smoothing kernels as proposed by Müller et al. [2003]. However, for the computation of the distance information and the covariance matrix we increase the smoothing radius to $2.5h$ to obtain reliable results. Except the velocity divergence threshold ϵ_u, all constants necessary to compute surface areas, i.e. the minimal surface area $a_{\min}$, the max delta function value $\delta_{\max}$, and the threshold for the determinant of the covariance matrix ϵ_c, are pre-computed from pre-defined incompressible particle configurations as shown in Fig. 7. Thus, a users get an intuitive way to define constants. For more details, see the accompanying 2D Matlab® implementation which has been used to generate Fig. 2.

8 Results and Discussion

We have tested our method in various scenarios in order to show its versatility, including the washing of dishes (Fig. 1), the wetting of a rotating cylinder (Fig. 6), the flooding of a valley (Fig. 10,3), a complex Kinect® scene (Fig. 8) and an evolution of a single drop of detergent (Fig. 5). For visualization of surfactant concentration we utilized an SPH-based volume rendering approach [Fraedrich et al. 2010] in combination with an image-based smoothing of particle surfaces [van der Laan et al. 2009]. The scenarios were simulated on an Intel Dual-Core 3.3 GHz with an NVidia GTX 580 with 1.5 GB VRAM, for which sample simulation timings are given in Tab. 1.

Due to increased smoothing radius during distance computation, our surface layer is two particles thick and computation of distance values becomes more time consuming. However, as described in Sec. 5.4, in regions with small velocity divergence we recompute the distance values only every 10 frames, which reduces the computational overhead.

Fig. 9 compares our surface area to the color gradient based interfaces used by Adami et al. [2010] (note, that we apply $2.5h$ sampling radius in contrast to $6h$ by Adami et al. [2010], which would lead to even thicker surface layers). Most importantly, our approach supports free surfaces. But even at phase interfaces our approach results in a much thiner band of contributing particles. Thus, our method avoids error-prone normal projection for surface dynamics as used by Bertalmio et al. [2001]; we do not require any normal computation. Additionally, color gradients are prone to irregular particle structures which requires a correction of differentials as also discussed by Becker and Teschner [2007].

In the teaser example in Fig. 1, dishes with a grease cover get exposed to incoming fluid with a constant detergent concentration of $Q = 1.0$. The underlying reaction kinetics transforms the grease R and the detergent S concentrated at the surface (Fig. 1, right), into reaction product P, i.e. solved grease / micelles in this case. The solved micelles diffuse in the bulk (Fig. 1, center).

Figure 8: *Complex room-scene which has been acquired with a Kinect® camera. Multiple material colors are washed out over time resulting in dirty water.*

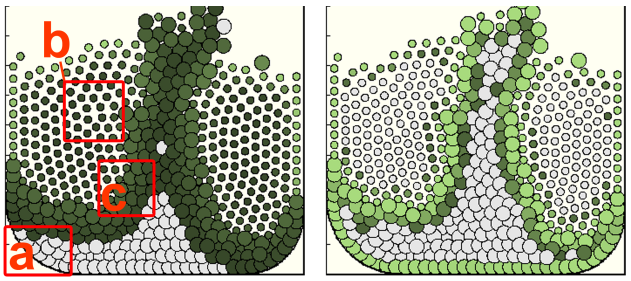
In the rotating cylinder example in Fig 6, three drops of detergent with $Q = 1.0$ are added after three seconds of simulation. Again, reaction kinetics take place at the cylinder surface, i.e. the added detergent leads to an wash-out effect. Additionally, detergent concentration at the fluid-cylinder interface steers the dragging force implied to the fluid. Here, the high detergent concentration lowers the fluid's drag.

In Fig. 8, we apply our SPH-based surface dynamics to a complex scene with highly irregular point distribution originating from a Kinect® senor. We use the acquired point colors as substance R to be washed out from the fluid poured into the scene with a detergent concentration of $Q = 1.0$.

The flooding valley example in Fig. 10 illustrates the influence of surface dynamics on dynamic rigid objects in the fluid. Again, two different detergent concentrations are used in the incoming fluid, $Q = 0.05$ (Fig. 10, top row) and $Q = 1.0$ (Fig. 10, bottom row). As in the cylinder example in Fig. 6, detergent concentration at the fluid-rigid interface take part in the reaction kinetics, i.e. the wash-out effect, and on the dragging at this interface. Whereas in an early state, the scene layout is very similar (Fig. 10, left column), the dynamic dragging and surface tension leads to a very different outcomes later on (Fig. 10, right column).

Currently, the main limitation of our implementation is that the surface resolution is directly linked with the bulk resolution, since we use a homogeneous particle size. However, in specific situations, the accuracy for surface area measures and surface thickness should be higher than the bulk resolution, as can be seen in the rotating cylinder example in Fig. 6. This limitation can be solved by adaptive particle resolutions at fluid surfaces as introduced by Orthmann and Kolb [2012] or by utilizing a multi-scale approach [Solenthaler and Gross 2011]. Another limitation is the simplicity of the reaction kinetics, which does not account for temperature dependencies or more complex, non-linear reactions like explosions.

We also refer to the additional video for better impression on the dynamics of our method.



(a) Adami et al. (b) Our approach

Figure 9: *Comparison to color-gradient based interface approximation by Adami et al. [2010] (sampling radius $2.5h$): Our approach handles free-surface (e.g. region a), irregular particle structures (e.g. region b), and it results in thin interfaces (e.g. region c).*

9 Conclusion

We presented a novel approach for robust computation of fluid (free) surfaces, which is the core concept for any physically based simulation of surface dynamics. We introduced a consistent and conservative formulation for interaction within a fluid surface and for adsorption-based interaction between fluid surface and bulk. Based on this formulation, we presented a diffusion model which incorporates different diffusion behavior for surface and bulk, as well as a model for reaction kinetics taking place at fluid-rigid surfaces. A specific example for reaction kinetics, i.e. surface cleaning is introduced as well. We evaluated our approach using various example scenarios.

Our surface dynamics approach can easily be enriched with other effects like foam dynamics, temperature dependant or non-linear reactions.

Acknowledgements

Suppressed for double blind reviewing purposes.

References

ADAMI, S., HU, X. Y., AND ADAMS, N. A. 2010. A conservative SPH method for surfactant dynamics. *J. Comput. Phys. 229*, 5, 1909–1926.

ADAMS, B., PAULY, M., KEISER, R., AND GUIBAS, L. J. 2007. Adaptively sampled particle fluids. *ACM Trans. Graph. 26*, 48:1–48:7.

AKINCI, G., IHMSEN, M., AKINCI, N., AND TESCHNER, M. 2012. Parallel surface reconstruction for particle-based fluids. *Computer Graphics Forum 31*, 1797–1809.

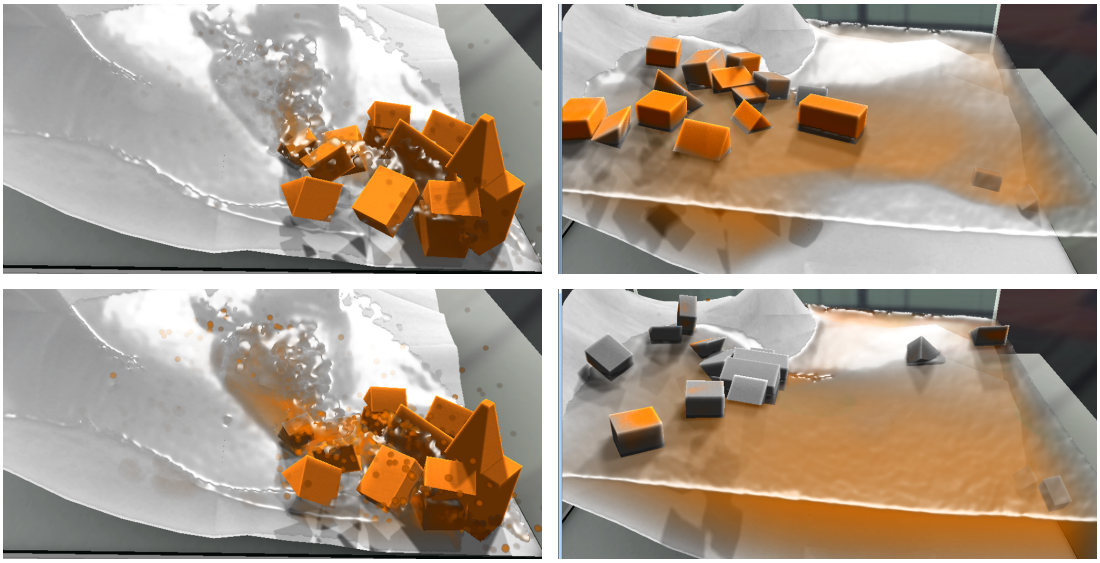


Figure 10: *Flooding of a valley with dynamics rigid objects shown for two time-steps $t_{sim} = 6\,s$ (left column) and $t_{sim} = 30\,s$ (right column). In the upper row a fluid with detergent concentration $Q = 0.1$ is emitted while in the lower row the fluid is initialized with a concentration of $Q = 1.0$. Less detergent not only results in less cleansing but also results in a different convection of rigid objects.*

AKINCI, N., IHMSEN, M., AKINCI, G., SOLENTHALER, B., AND TESCHNER, M. 2012. Versatile rigid-fluid coupling for incompressible SPH. *ACM Trans. Graph.*, 62:1–62:8.

BARGTEIL, A. W., GOKTEKIN, T. G., O'BRIEN, J. F., AND STRAIN, J. A. 2006. A semi-lagrangian contouring method for fluid simulation. *ACM Trans. Graph. 25*, 1, 19–38.

BECKER, M., AND TESCHNER, M. 2007. Weakly compressible SPH for free surface flows. In *Proc. Symp. Comp. Anim.*, 209–217.

BECKER, M., TESSENDORF, H., AND TESCHNER, M. 2009. Direct forcing for lagrangian rigid-fluid coupling. *IEEE Trans. Vis. & Comp. Graph. 15*, 3, 493–503.

BERTALMIO, M., CHENG, L.-T., OSHER, S., AND SAPIRO, G. 2001. Variational problems and partial differential equations on implicit surfaces: The framework and examples in image processing and pattern formation. *J. Comput. Phys. 174*, 759–780.

BERTHIER, J., AND SILBERZAN, P. 2009. *Microfluidics for biotechnology*. Artech House integrated microsystems series. Artech House.

BONET, J., AND KULASEGARAM, S. 2002. A simplified approach to enhance the performance of smooth particle hydrodynamics methods. *J. Applied Mathematics & Computation 126*, 2-3, 133–155.

CLEARY, P. W., AND MONAGHAN, J. J. 1999. Conduction modelling using smoothed particle hydrodynamics. *J. Comput. Phys. 148*, 227–264.

COLIN, F., EGLI, R., AND LIN, F. Y. 2006. Computing a null divergence velocity field using smoothed particle hydrodynamics. *J. Comput. Phys. 217*, 2, 680–692.

DEKKER, M. 2002. *Surfactants and Polymers in Drug Delivery*. Marcel Dekker, Inc., New York.

FELL, D., AUERNHAMMER, G. K., BONACCURSO, E., LIU, C., SOKULER, R., AND BUTT, H.-J. 2011. Influence of surfactant concentration and background salt on forced dynamic wetting and dewetting. *Langmuir 27*, 6, 2112.

FRAEDRICH, R., AUER, S., AND WESTERMANN, R. 2010. Efficient high-quality volume rendering of SPH data. *IEEE Trans. Vis. & Comp. Graph. 16*, 6, 1533–1540.

GINGOLD, R., AND MONAGHAN, J. 1977. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Notices of the Royal Astronomical Society 181*, 375–389.

GOSWAMI, P., SCHLEGEL, P., SOLENTHALER, B., AND PAJAROLA, R. 2010. Interactive SPH simulation and rendering on the GPU. In *Proc. Symp. Comp. Anim.*, 55–64.

GREEN, S. 2009. Particle simulation using CUDA. Tech. rep., NVIDIA.

HARADA, T., KOSHIZUKA, S., AND KAWAGUCHI, Y. 2007. Smoothed particle hydrodynamics on GPUs. *Proc. Computer Graphics International*, 63–70.

HIEBER, S. E., AND KOUMOUTSAKOS, P. 2005. A lagrangian particle level set method. *J. Comput. Phys. 210*, 342–367.

HONG, J.-M., LEE, H.-Y., YOON, J.-C., AND KIM, C.-H. 2008. Bubbles alive. *ACM Trans. Graph. 27*, 3, 48:1–48:4.

HU, X. Y., AND ADAMS, N. A. 2006. A multi-phase SPH method for macroscopic and mesoscopic flows. *J. Comput. Phys. 213*, 2, 844–861.

IHMSEN, M., AKINCI, N., GISSLER, M., AND TESCHNER, M. 2010. Boundary handling and adaptive time-stepping for PCISPH. In *Proc. Virt. Reality Interaction & Phys. Sim.*, 79–88.

IHMSEN, M., BADER, J., AKINCI, G., AND TESCHNER, M. 2011. Animation of air bubbles with SPH. In *Proc. Int. Conf. Comp. Graph. Theory & App. (GRAPP)*, 225–234.

IHMSEN, M., AKINCI, N., BECKER, M., AND TESCHNER, M. 2011. A parallel SPH implementation on multi-core CPUs. *Computer Graphics Forum 30*, 99–112.

IHMSEN, M., AKINCI, N., AKINCI, G., AND TESCHNER, M. 2012. Unified spray, foam and bubbles for particle-based fluids. *Vis. Comput. 28*, 669–677.

IWASAKI, K., UCHIDA, H., Y.DOBASHI, AND T. NISHITA, " (PG 2010), T. A. 2010. Fast particle-based visual simulation of ice melting. *Computer Graphics Forum 29*, 7, 2215–2223.

KEISER, R., ADAMS, B., GUIBAS, L. J., DUTR, P., AND PAULY, M. 2006. Multiresolution particle-based fluids. Tech. rep., ETH.

KOLB, A., AND CUNTZ, N. 2005. Dynamic particle coupling for GPU-based fluid simulation. In *Proc. Symp. Sim. Technique*, 722–727.

KOUMOUTSAKOS, P., COTTET, G.-H., AND ROSSINELLI, D. 2008. Flow simulations using particles: bridging computer graphics and CFD. In *Proc. SIGGRAPH classes*, 25:1–25:73.

KRISTOF, P., BENES, B., KRIVÁNEK, J., AND STAVA, O. 2009. Hydraulic erosion using smoothed particle hydrodynamics. *Computer Graphics Forum 28*, 2, 219–228.

LENAERTS, T., AND DUTRÉ, P. 2009. Mixing fluids and granular materials. *Computer Graphics Forum 28*, 2, 213–218.

LOSASSO, F., SHINAR, T., SELLE, A., AND FEDKIW, R. 2006. Multiple interacting liquids. *ACM Trans. Graph. 25*, 3, 812–819.

LOSASSO, F., TALTON, J., KWATRA, N., AND FEDKIW, R. 2008. Two-way coupled SPH and particle level set fluid simulation. *IEEE Trans. Vis. & Comput. Graph. 14*, 4, 797–804.

LUCY, L. B. 1977. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal 82*, 1013–1024.

MONAGHAN, J. J. 2005. Smoothed particle hydrodynamics. *Reports on Progress in Physics 68*, 1703–1759.

MÜLLER, M., CHARYPAR, D., AND GROSS, M. 2003. Particle-based fluid simulation for interactive applications. In *Proc. Symp. Comp. Anim.*, 154–159.

MÜLLER, M., SOLENTHALER, B., KEISER, R., AND GROSS, M. 2005. Particle-based fluid-fluid interaction. In *Proc. Symp. Comp. Anim.*, 237–244.

ONDERIK, J., CHLADEK, M., AND DURIKOVIC, R. 2011. SPH with small scale details and improved surface reconstruction. In *Proc. Spring Conf. Computer Graphics*.

ORTHMANN, J., AND KOLB, A. 2012. Temporal blending for adaptive SPH. *Computer Graphics Forum 31*, 8, 2436–2449.

OSHER, S., AND FEDKIW, R. 2002. *Level Set Methods and Dynamic Implicit Surfaces*. Springer-Verlag.

ROSEN, M. J., AND KUNJAPPU, J. T. 2004. *Surfactants and Interfacial Phenomena*. John Wiley & Sons, Inc.

RUNGJIRATANANON, W., SZEGO, Z., KANAMORI, Y., AND NISHITA, T. 2008. Real-time animation of sand-water interaction. *Computer Graphics Forum 27*, 1887–1893.

SCHECHTER, H., AND BRIDSON, R. 2012. Ghost SPH for animating water. *ACM Trans. Graph. 31*, 4, 61:1–61:8.

SOLENTHALER, B., AND GROSS, M. 2011. Two-scale particle simulation. *ACM Trans. Graph. 30*, 81:1–81:8.

SOLENTHALER, B., AND PAJAROLA, R. 2008. Density contrast SPH interfaces. In *Proc. Symp. Comp. Anim.*, 211–218.

SOLENTHALER, B., AND PAJAROLA, R. 2009. Predictive-corrective incompressible SPH. *ACM Trans. Graph. 28*, 40:1–40:6.

SOLENTHALER, B., SCHLÄFLI, J., AND PAJAROLA, R. 2007. A unified particle model for fluid solid interactions: Research articles. *Comput. Animat. Virtual Worlds 18*, 1, 69–82.

SOLENTHALER, B., ZHANG, Y., AND PAJAROLA, R. 2007. Efficient refinement of dynamic point data. In *Proc. Symp. Point-Based Graphics*, 65–72.

STORA, D., AGLIATI, P.-O., CANI, M.-P., NEYRET, F., AND GASCUEL, J.-D. 1999. Animating lava flows. In *Graphics Interface*, 203–210.

VAN DER LAAN, W. J., GREEN, S., AND SAINZ, M. 2009. Screen space fluid rendering with curvature flow. In *Proc. Symp. Interactive 3D Graphics & Games*, 91–98.

YU, J., AND TURK, G. 2010. Reconstructing surfaces of particle-based fluids using anisotropic kernels. In *Proc. Symp. Comp. Anim.*, 217–225.

ZHU, Y., AND BRIDSON, R. 2005. Animating sand as a fluid. *ACM Trans. Graph. 24*, 3, 965–972.

A Appendix: Temporal Coherence

A non-divergent and temporal coherent particle neighborhood, i.e. $\nabla \cdot \mathbf{u}_i = 0$ where velocities $\mathbf{u}_{ij} = \mathbf{u}_i - \mathbf{u}_j \approx \mathbf{0}$, leads to $\frac{\partial}{\partial t} W_{ij} = \mathbf{u}_{ij} \cdot \hat{\mathbf{r}}_{ij} \frac{\partial}{\partial r} W_{ij} \approx 0$. By applying this identity in combination with the quotient rule we get

$$\frac{\partial}{\partial t}\mathbf{m}_i = \frac{[\sum_j V_j \frac{\partial}{\partial t}(\mathbf{x}_j W_{ij})][\sum_j V_j W_{ij}] - [\sum_j V_j \frac{\partial}{\partial t} W_{ij}][\sum_j \mathbf{x}_j V_j W_{ij}]}{[\sum_j V_j W_{ij}]^2}$$

where $\frac{\partial}{\partial t}(\mathbf{x}_j W_{ij}) = \mathbf{u}_j W_{ij} + \mathbf{x}_j \frac{\partial}{\partial t} W_{ij} \approx \mathbf{u}_j W_{ij}$, which leads to

$$\frac{\partial}{\partial t}\mathbf{m}_i = \frac{[\sum_j \mathbf{u}_j V_j W_{ij}][\sum_j V_j W_{ij}]}{[\sum_j V_j W_{ij}]^2} \approx \mathbf{u}_i \frac{[\sum_j V_j W_{ij}][\sum_j V_j W_{ij}]}{[\sum_j V_j W_{ij}]^2} = \mathbf{u}_i.$$

for which directly follows that $\frac{\partial}{\partial t}\phi_i = \frac{\mathbf{x}_i - \mathbf{m}_i}{\|\mathbf{x}_i - \mathbf{m}_i\|}(\mathbf{u}_i - \frac{\partial}{\partial t}\mathbf{m}_i) = 0$. Consequently, $\frac{\partial}{\partial t}\delta_i = \frac{\partial}{\partial \phi}\delta(\phi_i)\frac{\partial}{\partial t}\phi_i = 0$ as long as $\nabla \cdot \mathbf{u}_i = 0$.

B Appendix: Surface Diffusion

Starting with the integral approximation of second derivatives according to Cleary et al. [Cleary and Monaghan 1999]:

$$\nabla^2 S(\mathbf{x}_i) \approx 2 \int_\Omega (S(\mathbf{x}_j) - S(\mathbf{x}_i)) \frac{\partial}{\partial r} W_{ij} \, \partial \mathbf{x}_j,$$

we can approximate the Laplacian by integrating first order derivatives over the fluid volume Ω. Since S is non-zero only at the surface, we can approximate a diffusive mass flux on the surface as

$$\sigma_d \nabla^2 S_i \approx \frac{2}{a_i} \sum_j \Gamma_{i \leftarrow j} = \frac{2}{a_i} \sum_j \sigma_d \, (S_j - S_i) \, a_j V_j \frac{\partial}{\partial r} W_{ij},$$

where σ_d is an isotropic diffusion constant. However, since particle areas are not equal we do not get an conservative SPH approximation directly and need to enforce symmetry $\Gamma_{i \leftarrow j} = -\Gamma_{j \leftarrow i}$ explicitly by using Eq. (10):

$$\sigma_d \nabla^2 S_i \approx \frac{1}{a_i} \sum_j \sigma_d \, (S_i - S_j) \, [a_j \, V_j \, + a_i \, V_i] \frac{\partial}{\partial r} W_{ij},$$

which effectively averages a mass flux between surface particles.