DOI: 10.1111/cgf.13523
ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2018
T. Beeler and N. Thuerey
(Guest Editors)

Volume 37 (2018), Number 8

MPM simulation of interacting fluids and solids

X. Yan[1†], C-F. Li[2], X-S. Chen[1] and S-M. Hu[1]

[1]Department of Computer Science and Technology, Tsinghua University, Beijing, China
[2]College of Engineering, Swansea University, Swansea, U.K.

Abstract

The material point method (MPM) has attracted increasing attention from the graphics community, as it combines the strengths of both particle- and grid-based solvers. Like the smoothed particle hydrodynamics (SPH) scheme, MPM uses particles to discretize the simulation domain and represent the fundamental unknowns. This makes it insensitive to geometric and topological changes, and readily parallelizable on a GPU. Like grid-based solvers, MPM uses a background mesh for calculating spatial derivatives, providing more accurate and more stable results than a purely particle-based scheme. MPM has been very successful in simulating both fluid flow and solid deformation, but less so in dealing with multiple fluids and solids, where the dynamic fluid-solid interaction poses a major challenge. To address this shortcoming of MPM, we propose a new set of mathematical and computational schemes which enable efficient and robust fluid-solid interaction within the MPM framework. These versatile schemes support simulation of both multiphase flow and fully-coupled solid-fluid systems. A series of examples is presented to demonstrate their capabilities and performance in the presence of various interacting fluids and solids, including multiphase flow, fluid-solid interaction, and dissolution.

CCS Concepts

•***Computing methodologies*** → ***Physical simulation;***

1. Introduction

Simulations of fluid flow and solid deformation are important in the visual effects industry. Most popular simulation methods are based on either particle- or grid-based solvers. Particle methods such as smoothed particle hydrodynamics (SPH) are popular for simulating liquids [MSKG05], deformable solids [MKN*04], granular materials, etc. Particle solvers are often simpler to implement, more robust in coping with geometric and topological changes, and more suitable for GPU-based parallel acceleration. However, they are often vulnerable to stability issues, and for example, near-surface SPH particle configurations may become distorted and form clusters [DRI95, YJL*16]. In comparison, grid-based methods can often achieve higher simulation accuracy for the same number of unknowns, and provide better simulation stability. Grid methods are particularly popular for simulating smoke [Sta99], fire [NFJ02], and deformable solids [CGFO06]. However, they often have a higher computational cost, especially for large scenes, and it is harder to cope with complex free surfaces such as those found in splashes. Both particle- and grid-based methods are well established, and both support simulation of complex coupled systems, e.g. multiphase flow [RLY*14, YCR*15], and interacting and reacting fluids and solids [YJL*16, YCL*17].

Recently, hybrid approaches combining strengths of both kinds of method have attracted increasing attention from the graphics community. The PIC/FLIP method was introduced by Zhu and Bridson [ZB05] for simulating sand; mass conservation is solved by particle advection while the constitutive and momentum equations are solved using a grid. The material point method (MPM) performs better for history-dependent materials; it was introduced to computer graphics by Stomakhin et al. [SSC*13]. It solves the constitutive equations using particles and computes forces by mapping the divergence of the stress tensor to a grid. While both FLIP and MPM share a conceptual basis with PIC, historically they have been considered separately by the engineering community, with FLIP for fluid simulation and MPM for solid simulation. Taking advantage of both particle and grid schemes, the adoption of FLIP and MPM represent a milestone for visual simulation. However, compared to particle and grid solvers, such hybrid methods are less well equipped for coupled systems involving multiple interacting fluids and solids.

Our work extends the MPM framework to permit simulation of multiple interacting fluids and solids. It is compatible with standard MPM. The new hybrid method solves the constitutive equations using particles, and the momentum equations on a grid, handling multiple liquids, gases, fluid-solid interactions, dissolution, etc. in a uniform manner. Our approach has the following advantages:

† shiotoli@gmail.com

1. it provides a uniform MPM framework for simulating both fluids and solids,
2. it accurately models contact interactions between fluids and deformable solids,
3. it allows simulation of multiple fluids, gas and liquid, miscible and immiscible,
4. it can readily be extended to incorporate other interactions, such as the dissolution of solids in liquids.

2. Previous Work

Hybrid simulation methods have been extensively investigated, particularly FLIP and MPM methods; we briefly recap this work here. We also review the latest research on fluid-solid coupling and simulation of multiple fluids, and analyse its relationship to our work.

2.1. FLIP method

The FLIP method was introduced to the graphics community by Zhu and Bridson [ZB05] for sand simulation. Boyd and Bridson [BB12] extended it to multiFLIP to simulate two-phase fluids with glugging effects. Ando et al. [ATW15] used FLIP with stream functions to enforce incompressibility of fluids, and later they improved the computational efficiency of FLIP by use of narrow band calculations [FAW*16]. Cornelis et al. [CIPT14] proposed the IISPH-FLIP method for simulating incompressible fluids, in which IISPH and FLIP are combined to conserve mass and reduce the computational load. Yang et al. [YLHQ14] proposed a FLIP method for droplet and spray simulation. The FLIP approach has been very successful and popular, providing impressive fluid simulation results, but it is seldom used for simulating deforming solids, due to the difficulty of handling history-dependent materials when the constitutive equations are set up on a Eulerian grid.

2.2. Material point method

MPM was first introduced to computer graphics by Stomakhin et al. [SSC*13] for the simulation of snow; it was further extended to handle phase change and varied materials [SSJ*14]. MPM has also been used to simulate foams using a shear-dependent flow model [YSB*15]. Jiang et al. [JSS*15] proposed the affine particle-in-cell method, which avoids the noise encountered in FLIP. Klár et al. [KGP*16], and Daviet and Bertails-Descoubes [DBD16] simulated granular materials with MPM. Ram et al. [RGJ*15] proposed an MPM scheme for viscoelastic fluids, foams and sponges. More recently, Tampubolon et al. [TGK*17] proposed an MPM approach for simulating porous sand and water mixtures. Gao et al. [GTJS17] gave a generalized interpolating MPM method for elastoplastic materials, in which the specific region of interest is adaptively refined. Predominantly used for simulating solids, MPM is seldom used to model fluids or interactions of multiple fluids and solids.

2.3. Fluid-solid coupling

Various research works have considered fluid-solid coupling in different simulation contexts. Akinci et al. [AIA*12] presented an SPH method to simulate fluid interacting with rigid bodies; it was later extended to include elastic solids [ACAT13]. This method was also used by Macklin et al. [MMCK14] in a unified particle framework. Yang et al. [YLM*16] proposed a versatile approach for simulating interactions at solid-solid and fluid-solid interfaces. Batty et al. [BBB07] proposed a high performance variational framework for solid-fluid coupling. Guendelman et al. [GSLF05] presented a solid-fluid coupling method for thin solid materials such as cloth. Robinson-Mosher et al. [RMSG*08] proposed a coupling method in which fluid simulation uses a Eulerian scheme and solid simulation uses a Lagrangian scheme. Narain et al. [NGL10] simulated the coupling between sand and rigid bodies. Carlson et al. [CMT04] proposed the rigid fluid method, which treats a rigid body as a fluid to solve the coupling between the rigid body and fluid. Robinson-Mosher et al. [RMEF09] proposed a method to more accurately calculate velocities in fluid-solid coupling. In all these coupling techniques, either the fluid and solid are simultaneously simulated with the same particle method, or are separately simulated with a particle solver and a grid solver. As a result, these coupling techniques cannot be directly transferred to hybrid approaches like FLIP and MPM, in which fluids and solids are simultaneously modelled with both particles and a grid.

2.4. Multiple-fluids simulation

Multiple fluid simulation has attracted much attention in recent years. Based on SPH, Müller et al. [MSKG05] proposed a fluid-fluid interaction method for this purpose. Solenthaler and Pajarola [SP08] addressed the density discontinuity problem. Losasso et al. [LSSF06] tracked the interface between fluids using the level set method in a grid-based solver. Misztal et al. [MEB*12] used unstructured moving meshes to capture multiphase flow in immiscible fluids. Kang et al. [KPNS10] considered miscible and immiscible fluids using a diffusion model. Ren et al. [RLY*14] introduced the concept of volume fraction into SPH, using a mixture model to simulate various multi-fluid phenomena. This work was further extended for multiphase interaction by introduction of a Helmholtz energy [YCR*15]. More recently, within the SPH framework, Yan et al. [YJL*16] simulated multiple fluids and solids as well as their various interactions. Shin et al. [SRC10] proposed a hybrid method for simulating viscous fingering phenomena and intermolecular diffusion. Recently, Gao et al. [GPH*18] proposed an MPM approach for simulation of particle-laden flow and archived impressive results.

3. MPM simulation of fluids and solids

Fluids and solids both obey mass and momentum conservation laws. The major difference between them lies in the constitutive laws obeyed, i.e. the relationship between stress and deformation. The governing equations of a deformable solid can be expressed as:

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = 0, \qquad \rho\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} = \nabla\cdot\boldsymbol{\sigma} + \rho\boldsymbol{g} \tag{1}$$

where $\mathrm{D}(\cdot)/\mathrm{D}t$ denotes the material derivative, ρ the density, $\boldsymbol{u}$ the velocity, $\boldsymbol{g}$ the external body force, and $\boldsymbol{\sigma}$ the stress tensor. The governing equations of a non-viscous fluid are similar:

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = 0, \qquad \rho\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} = -\nabla p + \rho\boldsymbol{g} \tag{2}$$

where p is the fluid pressure.

We solve them for both fluids and solids using the MPM formulation; the solid solution involves the stress $\boldsymbol{\sigma}$, while the fluid solution involves the pressure p. The basic MPM formulation is summarized in §3.1, after which the solid and fluid solutions are considered in §3.2 and §3.3 respectively.

3.1. Basic MPM theory

In MPM, continua are sampled using Lagrangian particles, and space is covered by a Eulerian grid. The momentum equation is represented by particle advection. All derivative terms are computed on a grid and physical quantities are transferred between particles and the grid by an invertible particle-to-grid mapping. In the following description, we use A_I to represent the variable A on grid node I and A_i to represent the variable on particle i. The particle-to-grid mappings for mass m and momentum $\boldsymbol{p}$ are:

$$m_I = \sum_{j \in H_I} m_j N_I(\boldsymbol{x}_j), \tag{3}$$

$$\boldsymbol{p}_I = \sum_{j \in H_I} \boldsymbol{p}_j N_I(\boldsymbol{x}_j) = \sum_{j \in H_I} \boldsymbol{v}_j m_j N_I(\boldsymbol{x}_j), \tag{4}$$

where H_I represents all particles in the vicinity of grid node I and $N_I(\boldsymbol{x}_j)$ is the shape function. There are different options for the shape function, e.g. uniform GIMP [ZCL16], B-spline interpolation [SSJ*14,SKB08] and adaptive GIMP [GTJS17]. For simplicity, we use uniform GIMP, defined as:

$$\begin{aligned} N_I(\boldsymbol{x}_j) = & S_I\left(\frac{|\boldsymbol{x}_{j,x} - \boldsymbol{x}_{I,x}|}{h}\right) S_I\left(\frac{|\boldsymbol{x}_{j,y} - \boldsymbol{x}_{I,y}|}{h}\right) \times \\ & S_I\left(\frac{|\boldsymbol{x}_{j,z} - \boldsymbol{x}_{I,z}|}{h}\right) \end{aligned} \tag{5}$$

where $\boldsymbol{x}_{j,x}$, $\boldsymbol{x}_{j,y}$, $\boldsymbol{x}_{j,z}$ represent the coordinates of particle j, $\boldsymbol{x}_{I,x}$, $\boldsymbol{x}_{I,y}$, $\boldsymbol{x}_{I,z}$ represent the coordinates of grid node I, and h is the MPM grid spacing. Function $S_I(x)$ is given by:

$$S_I(x) = \begin{cases} \dfrac{7 - 16x^2}{8} & x \le 0.25 \\ 1 - x & 0.25 < x \le 0.75 \\ \dfrac{(5 - 4x)^2}{16} & 0.75 < x \le 1.25 \\ 0 & \text{otherwise} \end{cases}. \tag{6}$$

The velocity on grid node I is computed as:

$$\boldsymbol{u}_I = \frac{\boldsymbol{p}_I}{m_I}. \tag{7}$$

The velocity gradient is computed on the grid and mapped back to the particles, which allows the constitutive equations to be solved on the particles. Then, the pressure and the stress are mapped from the particles to the grid, which allows the computation of the pressure gradient and stress divergence on the grid. The details are explained in § 3.2 and § 3.3.

The velocity and force equations are solved on the grid, and then mapped back to the particles for advection. The grid-to-particle mapping for velocity is:

$$\boldsymbol{u}_i = \sum_{J \in H_i} \frac{\boldsymbol{p}_J N_J(\boldsymbol{x}_i)}{m_J} \tag{8}$$

where H_i represents all grid nodes in the vicinity of particle i. The grid-to-particle mapping for force is:

$$\boldsymbol{a}_i = \sum_{J \in H_i} \frac{\boldsymbol{f}_J N_J(\boldsymbol{x}_i)}{m_J} \tag{9}$$

where $\boldsymbol{a}_i$ denotes the acceleration of particle i, and $\boldsymbol{f}_J$ is the force on the grid node J.

To solve the momentum equation, the velocity term needs to be mapped to particles using Eqn. (8), which results in large dissipation. The common way of addressing this problem is to use FLIP/PIC interpolation [SSC*13]. We map both acceleration and velocity from the grid to particles, and update the particle velocities as follows:

$$\begin{aligned} \bar{\boldsymbol{u}}_i &= \boldsymbol{a}_i \mathrm{d}t + \boldsymbol{u}_i^{\text{old}}, \\ \boldsymbol{u}_i^{\text{new}} &= \alpha \boldsymbol{u}_i + (1 - \alpha)\bar{\boldsymbol{u}}_i, \end{aligned} \tag{10}$$

where α is a control coefficient. A small α value makes the material more active, which is more suitable for fluid simulation, while a large α makes the material more stable. We set $\alpha = 0.05$ for fluids and $\alpha = 1$ for solids. The position of each particle is then updated as $\boldsymbol{x}^{t+\Delta t} = \boldsymbol{x}^t + \Delta t \boldsymbol{u}^{\text{new}}$.

3.2. MPM solid simulation

The key to solving the solid equation Eqn. (1) is to compute the stress $\boldsymbol{\sigma}$. We take a stress rate approach. Specifically, the constitutive equation for a linear elastic solid can be expressed as:

$$\dot{\boldsymbol{\sigma}} = \boldsymbol{C} : \dot{\boldsymbol{\epsilon}} = K\mathrm{Tr}(\dot{\boldsymbol{\epsilon}})\boldsymbol{I}/3 + 2G[\dot{\boldsymbol{\epsilon}} - \mathrm{Tr}(\dot{\boldsymbol{\epsilon}})/3] + \dot{\boldsymbol{\omega}} \cdot \boldsymbol{\sigma} - \boldsymbol{\sigma} \cdot \dot{\boldsymbol{\omega}} \tag{11}$$

where $\dot{\boldsymbol{\sigma}}$ is the stress rate, $\boldsymbol{C}$ the stiffness tensor, K the volume bulk, G the shear modulus, $\boldsymbol{I}$ the identity tensor, $\dot{\boldsymbol{\epsilon}}$ the strain rate, and $\dot{\boldsymbol{\omega}}$ the rotation rate tensor. The relationship between shear modulus and bulk modulus is $G = E/[2(1+\nu)]$ and $K = E/[3(1-2\nu)]$ where ν is the Poisson ratio. The strain rate and rotation rate tensors are defined as:

$$\begin{aligned} \dot{\boldsymbol{\epsilon}}_i &= \frac{1}{2}(\nabla \boldsymbol{u}_i + \nabla \boldsymbol{u}_i^T), \\ \dot{\boldsymbol{\omega}}_i &= \frac{1}{2}(\nabla \boldsymbol{u}_i - \nabla \boldsymbol{u}_i^T), \end{aligned} \tag{12}$$

where

$$\nabla \boldsymbol{u}_i = \sum_{J \in H_i} \boldsymbol{u}_J \otimes \nabla N_J(\boldsymbol{x}_i). \tag{13}$$

After updating the stress rate for the particles, the stress can be computed as:

$$\boldsymbol{\sigma}^{\text{new}} = \boldsymbol{\sigma}^{\text{old}} + \dot{\boldsymbol{\sigma}} \mathrm{d}t. \tag{14}$$

The stress divergence in Eqn. (1) is computed as:

$$\boldsymbol{f}_{I,\text{stress}} = -\sum_{j \in H_I} \frac{m_j}{\rho_j} \boldsymbol{\sigma}_j \cdot \nabla N_I(x_j) \tag{15}$$

where ρ_i is the density of particle i.

MPM overcomes tensile instability that occurs commonly in SPH simulation; it is caused by the poor distribution of particles near the surface. Some research has been undertaken to address tensile instability [CBP05,SB12,MM13,HWZ*14], but these methods

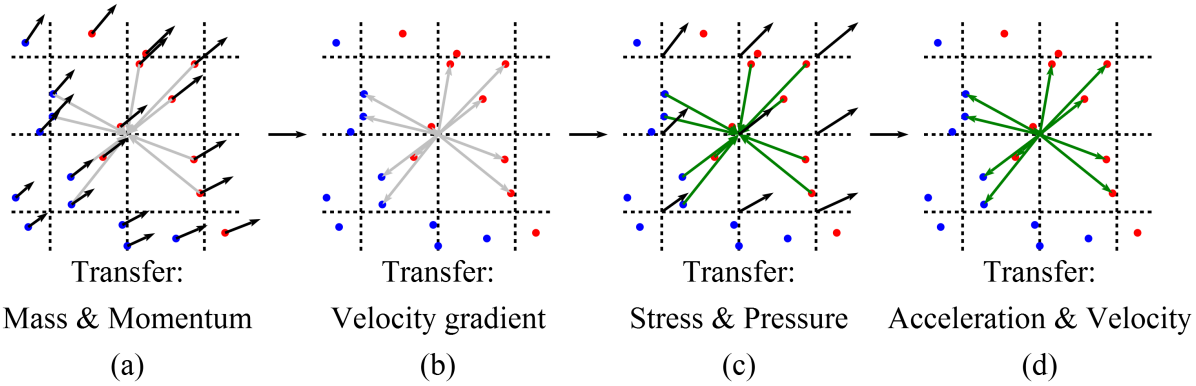

Figure 1: *MPM workflow. (a) Interpolate mass and momentum on the grid. (b) Map the velocity back to particles and solve the constitutive equation. (c) Map and compute force on the grid using particles. (d) Map velocity and acceleration back to particles.*

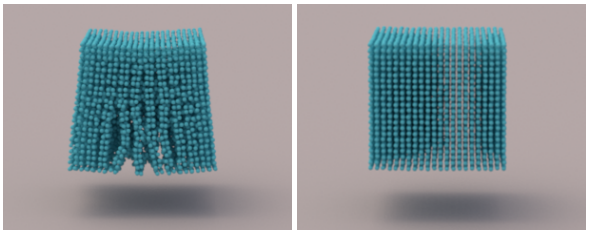

Figure 2: *Comparison between SPH and MPM methods for elastic material simulation. Left: the SPH method suffers from tensile instability. Right: The cube retains its shape during MPM simulation.*

require additional calculations. This issue is avoided in MPM as all velocity gradients are interpolated on grid nodes. Figure 2 illustrates the problem, and MPM's resolution of it.

3.3. MPM fluid simulation

Solving the fluid equation, Eqn. (2), may be done in a similar manner to solving for solids, but the key is now to compute the fluid pressure, which depends on the fluid density. There are different ways to do this, using different interpolation schemes. [SSC*13] and [ZCL16] estimate the fluid density at position $\boldsymbol{x}$ as:

$$\rho(\boldsymbol{x}) = \sum_{j \in H} \frac{m_j \theta(\boldsymbol{x} - \boldsymbol{x}_j)}{V_j}, \tag{16}$$

where $V_i^p = h^3$ is the particle volume, $\theta(\boldsymbol{x} - \boldsymbol{x}_i)$ can be the shape function or Dirichlet function, and H denotes the particle set in the vicinity of $\boldsymbol{x}$. An alternative approach is to use SPH interpolation:

$$\rho(\boldsymbol{x}) = \sum_{j \in H} m_j W(\boldsymbol{x} - \boldsymbol{x}_j, h_p) \tag{17}$$

where $W(\boldsymbol{x} - \boldsymbol{x}_j, h_p)$ is the kernel function and $h_p = 1.25h$ is the smoothing radius. Our limited experiments indicate that SPH interpolation performs better.

After obtaining the particle density, the particle pressure can be computed according to the weakly compressible equation [TGK*17] by:

$$p_i = k_s((\rho_i/\bar{\rho})^7 - 1) \tag{18}$$

where k_s is the stiffness coefficient, p pressure, and $\bar{\rho}$ rest density. Finally, the pressure gradient in Eqn. (2) can be calculated as:

$$\boldsymbol{f}_{I,\text{pressure}} = \sum_{j \in H_I} \frac{m_j}{\rho_j} p_j \nabla N_I(\boldsymbol{x}_j) \tag{19}$$

3.4. MPM work flow

ALGORITHM 1: MPM work flow

```
repeat
    for each grid node I do
        map mass and momentum to the grid using Eqns. (3,4) and compute velocity with Eqn. (7) (See Fig. 3(a))
    end for
    for each particle i do
        compute velocity gradient for solid particles with Eqn. (13) and compute pressure for fluid particles with Eqn. ( 18) (See Fig. 3(b))
    end for
    for each grid node I do
        compute the divergence or the gradient of the pressure and stress tensor on the grid with Eqns. (15,19) (See Fig. 3(c))
    end for
    for each particle i do
        map force and velocity from grids to particles using Eqns. (8,9) (See Fig. 3(d))
        update acceleration and velocity for each particle using Eqn. (10)
    end for
    t ← t + Δt
until end of simulation
```

The MPM work flow is summarized in Algorithm 1, and a graphical illustration is given in Figure 1. We use a leapfrog approach to

offset the velocity $\boldsymbol{u}$, $\boldsymbol{p}$, $\dot{\boldsymbol{\sigma}}$, $\dot{\boldsymbol{\epsilon}}$ and $\dot{\boldsymbol{\omega}}$ for $1/2$ time steps. Other variables such as $\boldsymbol{\epsilon}$, $\boldsymbol{\omega}$, $\boldsymbol{\sigma}$ and $\boldsymbol{f}$ are not offset. In detail, for the nth time step, the leapfrog computation can be summarized as follows:

1. The grid velocity $(\boldsymbol{u}_I)^{n-1/2}$ is updated via the grid momentum $(\boldsymbol{p}_I)^{n-1/2}$.
2. The velocity gradient $(\nabla \boldsymbol{u}_i)^{n-1/2}$ is updated by $(\boldsymbol{u}_J)^{n-1/2}$ to compute $(\dot{\boldsymbol{\epsilon}}_i)^{n-1/2}$ and $(\dot{\boldsymbol{\omega}}_i)^{n-1/2}$, and we then update $\dot{\boldsymbol{\sigma}}_i$.
3. The stress $(\boldsymbol{\sigma}_i)^n = (\boldsymbol{\sigma}_i)^{n-1} + \mathrm{d}t(\dot{\boldsymbol{\sigma}}_i)^{n-1/2}$ is computed to update the force term $\boldsymbol{f}_I^n$ and sequentially the velocity $(\boldsymbol{u}_i)^{n+1/2}$ and $(\boldsymbol{u}_I)^{n+1/2}$.

4. Coupling and interaction between fluids and solids

As explained in §3, MPM solid simulation and MPM fluid simulation share the same work flow, with the main difference lying in the treatment of the constitutive equations. As both solid and fluid equations are handled by particle advection, parallel computing may be readily used in both cases. As all spatial derivatives are computed on a grid, both solid and fluid simulations are potentially more accurate and more stable than those produced by conventional particle methods. However, the lack of efficient and robust coupling and interaction techniques remains a major hurdle hampering the wider application of MPM.

Two types of interactions are considered in this work: fluid-fluid interaction and fluid-solid interaction. The former deals with multiple fluids; both immiscible and miscible effects are considered. Fluid-solid interaction can also be considered as being immiscible or miscible. In the former case, their interaction is treated as coupling through contact forces. In the latter case, their interaction is treated as a mixing process, i.e. the solid dissolves. Solid-solid interaction in the MPM framework has been well studied and documented elsewhere [ZCL16, SSC*13], so is not considered in this work. Tampubolon et al. [TGK*17] used two background grids for sand and water simulation. In our approach, we use a single background grid for all different material phases.

4.1. Coupling between fluids and solids

Evaluating fluid-solid coupling involves two tasks:

1. computing the contact force between fluid and solid,
2. preventing the fluid from penetrating the solid.

Dealing with the contact force between fluid and solid is straightforward in particle methods, but how to do so is less clear for MPM because inter-particle forces are not considered as part of the computation. Instead, for a given grid node, we compute separately a force term from the fluid pressure and a force term from the solid stress, which are then combined to give the total force on grid node $\boldsymbol{f}_{I,\text{total}}$:

$$\boldsymbol{f}^f_{I,\text{total}} = -\boldsymbol{f}^s_{I,\text{total}} = \boldsymbol{f}_{I,\text{pressure}} + \boldsymbol{f}_{I,\text{stress}}. \tag{20}$$

After obtaining the total force, we update the velocity on the grid, and map the velocity and force from the grid to both fluid and solid particles. The fluid and solid velocities are driven individually by the fluid pressure and the solid stress respectively. By using the shared force term $\boldsymbol{f}_{I,\text{total}}$ at the fluid-solid interface, we ensure consistent velocities at the interface.

As Eqn. (10) is used to control dissipation in fluid simulation, it is difficult to prevent fluid-solid penetration. In SPH simulations, it is typically prevented by adding artificial forces to particles [AIA*12], but this simple treatment does not work here as inter-particle forces are not intrinsically incorporated in MPM. Stomakhin et al. [SSJ*14] proposed a collision detection method to handle collision, but it relies on interface tracking, which is rather inconvenient in the presence of multiple fluids and solids. Instead, we propose an anti-penetration model consistent with the MPM workflow that prevents fluid-solid penetration on the grid. Let $\boldsymbol{p}^f_I$ denote the momentum of the fluid at grid node I, and $\boldsymbol{p}^s_I$ the momentum of the solid. We map the fluid and solid velocities onto the grid, and allow the fluid and the solid to fully collide on the grid node and reach the same velocity. Based on momentum conservation, the interaction force during this collision process can be derived at the grid node:

$$\boldsymbol{f}_{I,collision} = \frac{\boldsymbol{p}^s_I m^s_I - \boldsymbol{p}^f_I m^f_I}{(m^f_I + m^s_I)\Delta t}, \tag{21}$$

where Δt is the time step. Here, we ignore the viscous force of the fluid, so the contact force between fluid and solid is merely a normal force at the interface; it may be estimated as below:

$$\boldsymbol{f}^f_{I,\text{ap}} = -\boldsymbol{f}^s_{I,\text{ap}} = \beta(\boldsymbol{f}_{I,\text{collision}} \cdot \boldsymbol{n}_I)\boldsymbol{n}_I, \tag{22}$$

where $\boldsymbol{n}_I$ is the normal direction at the interface and β is a control coefficient (set to 1.0 in our simulations for efficient prevention of penetration). The normal direction $\boldsymbol{n}_I$ is estimated from the mass distribution of fluid and solid:

$$\bar{\boldsymbol{n}}^f_I = \frac{\sum_{j\in H^f_I} m_j \nabla N_I(\boldsymbol{x}_j)}{||\sum_{j\in H^f_I} m_j \nabla N_I(\boldsymbol{x}_j)||}, \tag{23}$$

$$\bar{\boldsymbol{n}}^s_I = \frac{\sum_{j\in H^s_I} m_j \nabla N_I(\boldsymbol{x}_j)}{||\sum_{j\in H^s_I} m_j \nabla N_I(\boldsymbol{x}_j)||}, \tag{24}$$

$$\boldsymbol{n}^f_I = -\boldsymbol{n}^s_I = \frac{\bar{\boldsymbol{n}}^f_I - \bar{\boldsymbol{n}}^s_I}{||\bar{\boldsymbol{n}}^f_I - \bar{\boldsymbol{n}}^s_I||}, \tag{25}$$

where H^f_I and H^s_I are the fluid particle set and the solid particle set in the vicinity of grid node I. The anti-penetration force $\boldsymbol{f}_{I,ap}$ is mapped from the grid to fluid and solid particles separately in opposite directions $\boldsymbol{n}^f_I$ and $\boldsymbol{n}^s_I$. Note that this is only done when a collision is detected on the grid, i.e. when $(\hat{\boldsymbol{u}}^f_I - \hat{\boldsymbol{u}}^s_I)\cdot \boldsymbol{n}_I > 0$. The current velocity $\hat{\boldsymbol{u}}_I$ is updated on the grid using the grid force: $\hat{\boldsymbol{u}}_I = \boldsymbol{u}_I + \boldsymbol{f}_I \mathrm{d}t/m_I$.

Figure 3 illustrates the computation involved in fluid-solid coupling, while its workflow is summarized in Algorithm 2.

4.2. Multi-fluid simulation

Simulation of multiple fluids has already been considered using the SPH framework with impressive results: fluid-fluid interaction is modelled using a mixture model in [RLY*14] and a Helmholtz free

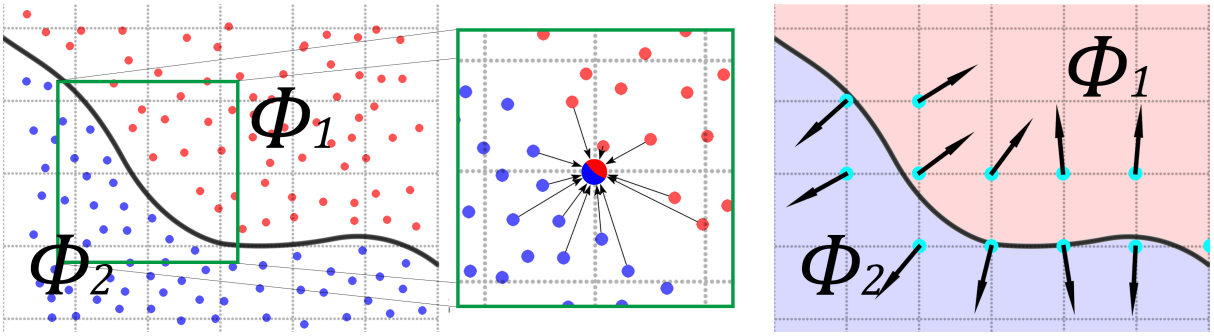

Figure 3: *Fluid-solid coupling in MPM. Left: Φ_1 and Φ_2 denote two different phases. Middle: All forces are added to the grid and the surface normal is computed on the grid. Right: The collision force is mapped from the grid to particles.*

ALGORITHM 2: Fluid-solid coupling

repeat
  for each grid node I **do**
    map mass and momentum to the grid using Eqns. (3,4) and compute velocity with Eqn. (7)
  end for
  for each particle i **do**
    compute velocity gradient for solid particles with Eqn. (13) and pressure for fluid particles with Eqn. (18)
  end for
  for each grid node I **do**
    compute the divergence or the gradient of the pressure and stress tensor on the grid with Eqns. (15,19)
  end for
  for each grid node I **do**
    sum fluid and solid forces in Eqn. (20)
  end for
  for each grid node I **do**
    predict fluid and solid velocities on the grid
    estimate the surface normal in Eqn. (25)
    if a collision is detected, compute the collision force in Eqn. (22) on the grid and map it to fluid and solid particles
  end for
  for each particle i **do**
    map force and velocity from grid nodes to particles using Eqns. (8,9)
    update acceleration and velocity for each particle using Eqn. (10)
  end for
  $t \leftarrow t + \Delta t$
until end of simulation

energy model in [YCR*15]. Here, we show how to include fluid-fluid interaction in MPM, including both miscible and immiscible effects.

Immiscible fluids If the fluids are immiscible, they can be treated in a similar way to the fluid-solid coupling described in § 4.1. Specifically, all pressure forces are added up on grid nodes and then mapped to particles. To prevent interpenetration of immiscible fluids, an anti-penetration force is introduced at the fluid-fluid interface:

$$\boldsymbol{f}_{I,\text{contact}}^{f_1} = -\boldsymbol{f}_{I,\text{contact}}^{f_2} = \beta \left(\frac{\boldsymbol{p}_I^{f_2} m_I^{f_2} - \boldsymbol{p}_I^{f_1} m_I^{f_1}}{(m_I^{f_1} + m_I^{f_2})\Delta t} \cdot \boldsymbol{n}_I \right) \boldsymbol{n}_I. \quad (26)$$

The superscripts f_1 and f_2 denote two immiscible fluids, while β is a control coefficient. Larger β prevents penetration more strictly (it is set to 0.3 in our simulations). The contact force is 0 if $(\hat{\boldsymbol{u}}_I^{f_1} - \hat{\boldsymbol{u}}_I^{f_2}) \cdot \boldsymbol{n}_I < 0$. For a fluid mix with three or more different fluids, we treat each fluid phase separately, with all other phases considered as one single phase.

Miscible fluids To model interaction between miscible fluids, we use a diffusion model:

$$\frac{\mathrm{D}\alpha^k}{\mathrm{D}t} = k_d \nabla^2 \alpha^k, \quad (27)$$

where k_d is the diffusion coefficient, α^k the concentration of the fluid phase k, and $\sum_k \alpha^k = 1$. In the MPM formulation, the diffusion equation is handled by concentration advection, where the associated spatial derivatives are computed on the grid using:

$$\nabla \alpha_I^k = \frac{\sum_j m_j \alpha_j^k \nabla N_I(\boldsymbol{x}_j)}{||\sum_j m_j \alpha_j^k \nabla N_I(\boldsymbol{x}_j)||}, \qquad \nabla^2 \alpha_i^k = \sum_J \nabla \alpha_J^k \cdot \nabla N_J(\boldsymbol{x}_i). \quad (28)$$

Then, we use $\nabla^2 \alpha_i^k$ to compute the material derivative of α^k in Eqn. (27) and update the fluid concentrations. After doing so on particles, the mass and density values are updated accordingly: the particle mass is computed as $m_i = \sum_k \alpha_i^k \bar{m}^k$, where $\bar{m}^k$ is the rest mass of phase k on each particle, and the particle rest density is computed as $\bar{\rho}_i = \sum_k \alpha_i^k \bar{\rho}^k$, where $\bar{\rho}^k$ is rest density of phase k.

4.3. Dissolution

If a solid is miscible with a fluid, the interaction takes the form of dissolution, instead of contact. Dissolution processes such as sugar dissolving in hot drinks are common in daily life, and they can be

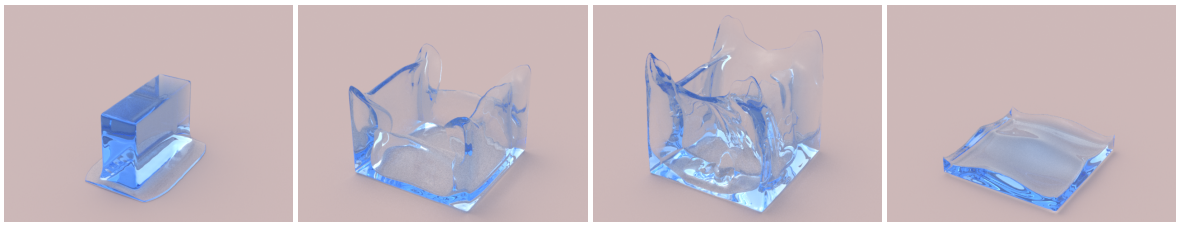

Figure 4: *Single-phase dam break flow.*

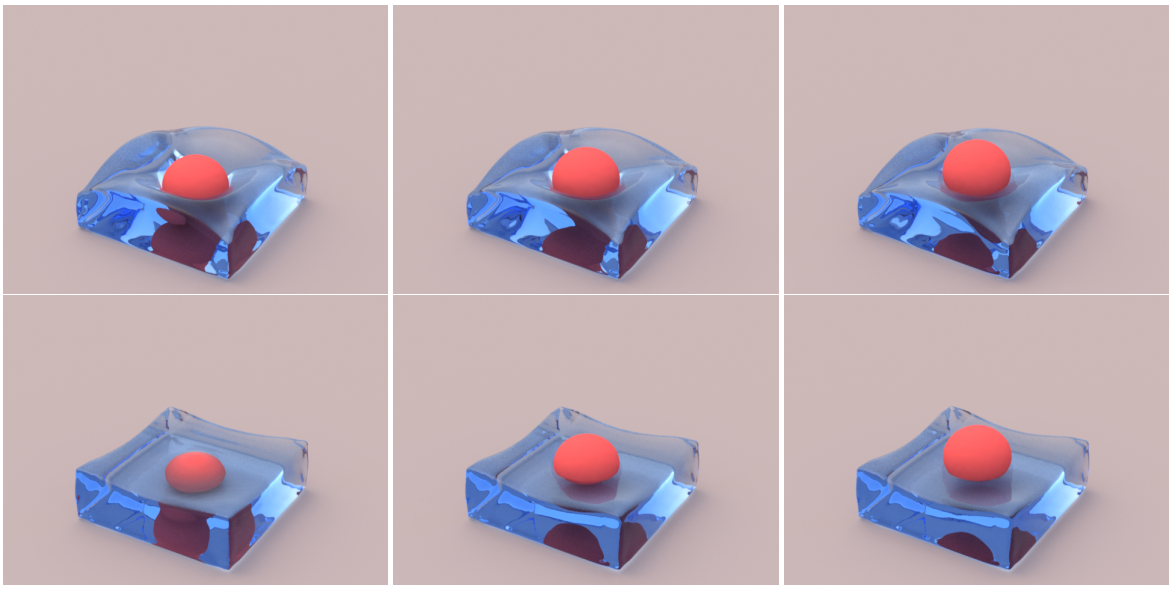

Figure 5: *A ball dropping into water. Left: Density ratio of ball:water is 1:1. Middle: Density ratio of ball:water is 1:2. Right: Density ratio of ball:water is 1:3*

described by the Noyes-Whitney equation [NW97]:

$$\frac{DC}{Dt} = \kappa(C_s - C), \tag{29}$$

where C represents the volume fraction of the dissolved solid phase, C_s is the saturation volume fraction, and κ is the dissolution coefficient.

Let s denote the solid phase that is soluble in the fluid phase f. To solve Eqn. (29) in the MPM framework, we first need to compute the concentration C at grid node I:

$$C_I = \frac{\sum_{j \in H_I^f} m_j \alpha_j^s, N_I(\boldsymbol{x}_j)}{m_I^f} \tag{30}$$

where H_I^f represents all fluid particles in the vicinity of node I, m_j the particle mass, α_j^s the solid volume fraction on the fluid particle j, and m_I^f the fluid mass at node I. Then the concentration change can be computed on the grid using:

$$\Delta C_I = \kappa(C_s - C_I)dt. \tag{31}$$

Finally, this concentration change is mapped from the grid to the particles, allowing updating of the particle volume fraction of the solid phase:

$$\begin{aligned} \Delta\alpha_i^s &= \sum_{J \in H_i} \frac{m_J^s}{m_J^s + m_J^f} \Delta C_J N_J(\boldsymbol{x}_i), \quad \text{for fluid particle } i, \\ \Delta\alpha_i^s &= -\sum_{J \in H_i} \frac{m_J^f}{m_J^s + m_J^f} \Delta C_J N_J(\boldsymbol{x}_i), \quad \text{for solid particle } i, \end{aligned} \tag{32}$$

where m_I^s is the solid mass at grid node I, m_I^f is the fluid mass there, and H_i represents all grid nodes in the vicinity of particle i. The volume fractions can be readily updated for particles using $\alpha_i^s = \alpha_i^s + \Delta\alpha_i^s$, $\alpha_i^f = 1 - \alpha_i^s$.

In our implementation, we label each particle as a fluid or solid particle. If the concentration α_i^s of solid s in the solid particle meets C_s, we relabel this solid particle as a fluid particle; the fluid particles never become solid particles in our approach. Both α and C can be computed independently from the standard MPM framework. In our approach, we update α and C after line 18 in Algorithm 2.

5. Results

We have implemented our approach on an NVIDIA GeForce GTX980 16GB GPU. A series of examples is presented to demon-

strate the simulation capacity of the proposed hybrid approach. Corresponding performance data are given in Table 1.

Fluid simulation Case 1 (see Fig. 4) shows a dam-break example, where a block of water collapses due to gravity, strikes the wall, splashes into the air, and gradually settles under gravity. This standard test confirms that the proposed MPM approach can successfully simulate single-phase fluid flow.

Coupling between fluid and solid Case 2 (see Fig. 5) is designed to test fluid-solid coupling. Three elastic balls are dropped into water separately; the geometry in all three tests remains constant. The three balls also have identical shear modulus and Poisson ratio, $G = 10^6$ and $\nu = 0.5$. The balls have different densities, with density ratios between ball and water of $1:1$, $1:2$ and $1:3$ respectively. The heaviest ball drops deeply into the water producing a large hole, and is then covered by water directly. The medium ball hits the water making a small hole, and bounces up and down before becoming partially immersed in the water. The lightest ball hits the water with little penetration, and then drifts on the water surface. This example confirms that our approach can correctly capture dynamic buoyancy effects. An efficiency comparison is made between simulations with and without the anti-penetration force. With the anti-penetration force the simulation runs at an average of 21 steps per second; it runs at 25 steps per second otherwise. There are 5.56×10^5 particles.

Multiple fluid simulation Case 3 (see Fig. 6) shows three dam breaks with different fluids. The red and green liquids are miscible with each other, but they are both immiscible with the blue liquid. The densities of the blue, red and green liquids are set to 3, 2 and 1 respectively, and the diffusion coefficient is set as $k_d = 0.0003$. Driven by gravity, the three liquids collapse and interact with each other, forming a complex mix. As time passes, the green and red liquids become fully mixed with each other, but the mix remains separate from the blue liquid, with a sharp interface. This example confirms that the proposed MPM approach can correctly handle fluid-fluid interaction, capturing both miscible and immiscible effects.

Smoke simulation Case 4 (see Fig. 7) shows an example of rising smoke, which is simulated as a two-phase flow, with smoke and air phases. The densities of smoke and air are set to 1 and 2 respectively, allowing the smoke to rise while retaining visible interaction with the surrounding air. To accelerate the rate of rising of the smoke, an additional temperature field is also introduced to add artificial buoyancy effects. Temperature evolution follows a diffusion model, which is solved in a similar way to Eqn. (32) using the MPM formulation. For clarity of demonstration, the smoke phase is assumed to be immiscible with the air phase. In Fig. 7, the results on the left are obtained using the fluid-fluid interaction Eqn. (26), while the results on the left are obtained without fluid-fluid interaction. Clearly, the fluid-fluid interaction is essential to keep the smoke in the form of a mushroom cloud, instead of it moving randomly under the artificial buoyancy.

Dissolution Case 5 (see Fig. 8) shows an elastic bunny dissolving into water. After a dam break, water crashes into the elastic bunny and causes it to drift around. During this time the bunny dissolves gradually into the water, turning it red. The densities of bunny and water are set to 1 and 2 respectively. The saturation concentration is 0.5 and the dissolution coefficient is 0.0003. This example confirms that the proposed MPM approach can deal with miscible solids and fluids.

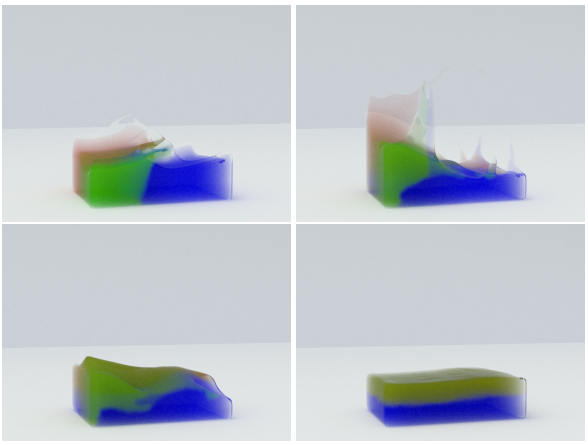

Figure 6: *Left to right, top to bottom: Three dam breaks. The red and the green liquids are miscible, but are both immiscible with the blue liquid.*

6. Conclusions

In this paper, we have developed a unified MPM framework with efficient and versatile coupling techniques for multiple fluid-fluid and fluid-solid interactions; we have demonstrated the new approach on a wide range of challenging phenomena. For fluid-solid interactions, we give a method to prevent the fluid from penetrating the solid, and an MPM-based dissolution simulation method based on the Noyes-Whitney equation. For fluid-fluid interactions, we proposed a multi-fluid scheme for both miscible and immiscible fluids in the MPM framework. Using our MPM-based multi-fluid simulation method, smoke can be simulated by incorporating air particles into the scene to interact with smoke particles.

The MPM scheme has the intrinsic advantage of readily coping with geometric and topological changes, and the convenience of ease of GPU implementation. Compared with the SPH method, MPM interpolates physical quantities using the background grid, thereby improving stability during solid simulation, and avoiding the tensile instability found in SPH (see Fig. 2). In SPH, fluid-solid coupling [AIA*12, ACAT13], multiple fluids [RLY*14], and dissolution [YJL*16] have been developed more thoroughly. However, wider applications of MPM have up to now been hampered by the lack of robust and flexible interaction and coupling techniques between multiple fluids and solids. We have extended the MPM framework with robust coupling schemes to fill this gap.

7. Limitations and future work

The main limitation of the proposed MPM approach lies in its method of fluid simulation, in particular for incompressible fluids. We use the weakly compressible equation to obtain the pressure

Table 1: *Performance data; the time step used in all cases is* 10^{-4} *s.*

Example	Description	Phases	Particles	Performance (steps/s)	Performance (frame/s)
1	Single phase dam break	1	3.42×10^5	55 steps/s	0.55 frame/s
2	Ball in water	2	5.56×10^5	21 steps/s	0.42 frame/s
3	Multiple fluids	3	3.94×10^5	56 steps/s	1.01 frame/s
4	Smoke	2	$1.66 \times 10^6 - 1.85 \times 10^6$	11 steps/s	0.44 frame/s
5	Bunny dissolution	2	1.63×10^5	67 steps/s	1.33 frame/s

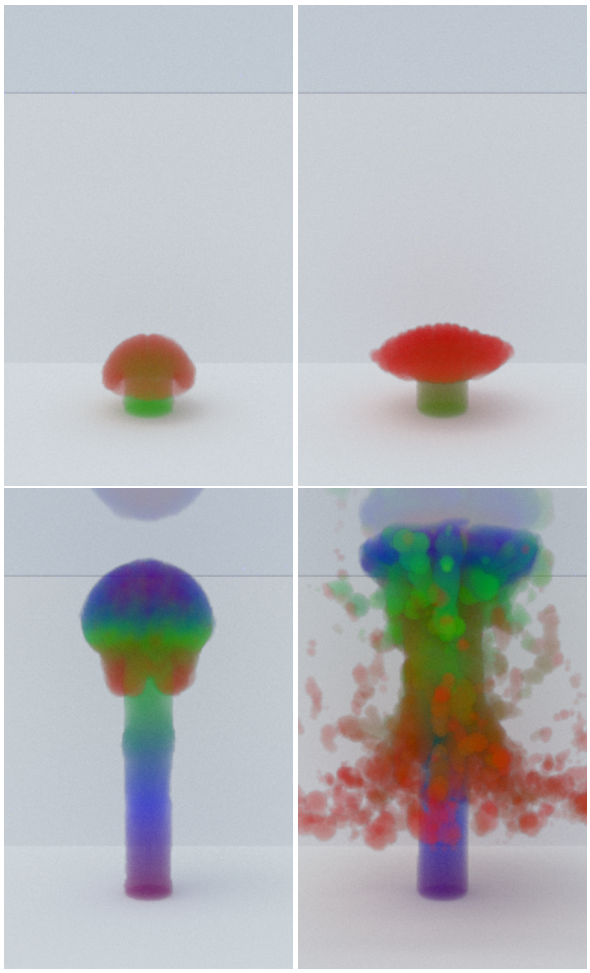

Figure 7: *Rising smoke; the smoke is injected from the ground. Left: Good results with fluid-fluid interaction. Right: Poor results without fluid-fluid interaction.*

from the fluid density. Pressure accuracy drops at the simulation boundary where the particle distribution is typically poorer. A more accurate approach would be to enforce the incompressibility condition and solve for the pressure implicitly. This is beyond the scope of this work, but is worth pursuing in future research.

For fluid-solid coupling, we use an anti-penetration force which results in a small gap between the solid and the fluid. When the solid is soluble, the gap will slow down its dissolution. If the force is very large, it may even prevent it. In order to maintain a normal dissolution rate, the larger the anti-penetration force is, the greater the control coefficient β in Eqn. (26) needs to be to compensate.

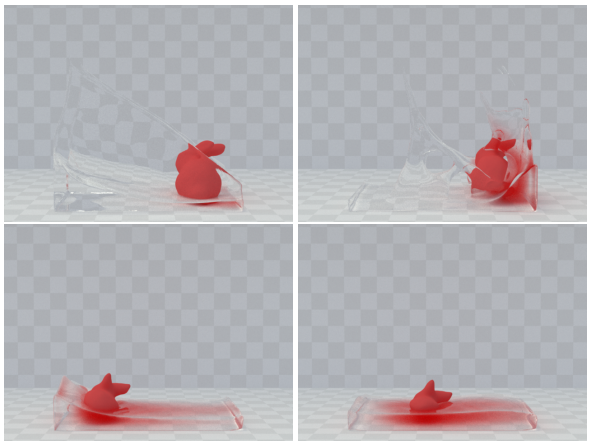

Figure 8: *Left to right, top to bottom: An elastic red bunny dissolves in water, turning it red.*

Since the coarse background grid smooths the flow field during grid-based interpolation, the simulated liquid will seem more viscous. One solution is to decrease α in Eqn. (10) so that the kinetic energies of particles increase. However, this will cause the particles to become more disordered. Activity and stability form a trade-off in MPM. Furthermore, a multi-fluid scheme is used to simulate gas in the MPM framework, but in order to achieve detailed smoke simulation, more air particles are needed. Refining the simulation can address these two issues, at the cost of increased simulation time. Some recent research promises to be relevant, e.g. using adaptive GIMP [GTJS17], and will be pursued in future work.

Acknowledgements

This work was supported by the National Key Technology R&D Program (Project Number 2017YFB1002701), the Natural Science Foundation of China (Project Number 61521002). The authors would like to thank Professor Ralph R. Martin at Cardiff University, UK, for his constructive comments on this work and the help on the English presentation.

References

[ACAT13] AKINCI N., CORNELIS J., AKINCI G., TESCHNER M.: Coupling elastic solids with smoothed particle hydrodynamics fluids. *Computer Animation and Virtual Worlds 24*, 3-4 (2013), 195–203. 2, 8

[AIA*12] AKINCI N., IHMSEN M., AKINCI G., SOLENTHALER B., TESCHNER M.: Versatile rigid-fluid coupling for incompressible sph. *ACM Trans. Graph. 31*, 4 (July 2012), 62:1–62:8. URL: http://doi.acm.org/10.1145/2185520.2185558, doi:10.1145/2185520.2185558. 2, 5, 8

[ATW15] ANDO R., THUEREY N., WOJTAN C.: A stream function solver for liquid simulations. *ACM Trans. Graph. 34*, 4 (July 2015), 53:1–53:9. URL: http://doi.acm.org/10.1145/2766935, doi:10.1145/2766935. 2

[BB12] BOYD L., BRIDSON R.: Multiflip for energetic two-phase fluid simulation. *ACM Trans. Graph. 31*, 2 (Apr. 2012), 16:1–16:12. URL: http://doi.acm.org/10.1145/2159516.2159522, doi:10.1145/2159516.2159522. 2

[BBB07] BATTY C., BERTAILS F., BRIDSON R.: A fast variational framework for accurate solid-fluid coupling. *ACM Trans. Graph. 26*, 3 (2007), 100. doi:http://doi.acm.org/10.1145/1276377.1276502. 2

[CBP05] CLAVET S., BEAUDOIN P., POULIN P.: Particle-based viscoelastic fluid simulation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), SCA '05, ACM, pp. 219–228. URL: http://doi.acm.org/10.1145/1073368.1073400, doi:10.1145/1073368.1073400. 3

[CGFO06] CHENTANEZ N., GOKTEKIN T. G., FELDMAN B. E., O'BRIEN J. F.: Simultaneous coupling of fluids and deformable bodies. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2006), SCA '06, Eurographics Association, pp. 83–89. URL: http://dl.acm.org/citation.cfm?id=1218064.1218075. 1

[CIPT14] CORNELIS J., IHMSEN M., PEER A., TESCHNER M.: Iisph-flip for incompressible fluids. *Comput. Graph. Forum 33*, 2 (May 2014), 255–262. URL: http://dx.doi.org/10.1111/cgf.12324, doi:10.1111/cgf.12324. 2

[CMT04] CARLSON M., MUCHA P. J., TURK G.: Rigid fluid: Animating the interplay between rigid bodies and fluid. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 377–384. URL: http://doi.acm.org/10.1145/1015706.1015733, doi:10.1145/1015706.1015733. 2

[DBD16] DAVIET G., BERTAILS-DESCOUBES F.: A semi-implicit material point method for the continuum simulation of granular materials. *ACM Trans. Graph. 35*, 4 (July 2016), 102:1–102:13. URL: http://doi.acm.org/10.1145/2897824.2925877, doi:10.1145/2897824.2925877. 2

[DRI95] DYKA C. T., RANDLES P. W., INGEL R. P.: Stress points for tension instability in sph. *International Journal for Numerical Methods in Engineering 40*, 13 (1995), 2325–2341. doi:10.1002/(SICI)1097-0207(19970715)40:13<2325::AID-NME161>3.0.CO;2-8. 1

[FAW*16] FERSTL F., ANDO R., WOJTAN C., WESTERMANN R., THUEREY N.: Narrow Band FLIP for Liquid Simulations. *Computer Graphics Forum* (2016). doi:10.1111/cgf.12825. 2

[GPH*18] GAO M., PRADHANA A., HAN X., GUO Q., KOT G., SIFAKIS E., JIANG C.: Animating fluid sediment mixture in particle-laden flows. *ACM Trans. Graph.* (2018). URL: http://doi.acm.org/10.1145/3197517.3201309, doi:10.1145/3197517.3201309. 2

[GSLF05] GUENDELMAN E., SELLE A., LOSASSO F., FEDKIW R.: Coupling water and smoke to thin deformable and rigid shells. *ACM Trans. Graph. 24*, 3 (July 2005), 973–981. URL: http://doi.acm.org/10.1145/1073204.1073299, doi:10.1145/1073204.1073299. 2

[GTJS17] GAO M., TAMPUBOLON A. P., JIANG C., SIFAKIS E.: An adaptive generalized interpolation material point method for simulating elastoplastic materials. *ACM Trans. Graph. 36*, 6 (Nov. 2017), 223:1–223:12. URL: http://doi.acm.org/10.1145/3130800.3130879, doi:10.1145/3130800.3130879. 2, 3, 9

[HWZ*14] HE X., WANG H., ZHANG F., WANG H., WANG G., ZHOU K.: Robust simulation of sparsely sampled thin features in sph-based free surface flows. *ACM Trans. Graph. 34*, 1 (Dec. 2014), 7:1–7:9. URL: http://doi.acm.org/10.1145/2682630, doi:10.1145/2682630. 3

[JSS*15] JIANG C., SCHROEDER C., SELLE A., TERAN J., STOMAKHIN A.: The affine particle-in-cell method. *ACM Trans. Graph. 34*, 4 (July 2015), 51:1–51:10. URL: http://doi.acm.org/10.1145/2766996, doi:10.1145/2766996. 2

[KGP*16] KLÁR G., GAST T., PRADHANA A., FU C., SCHROEDER C., JIANG C., TERAN J.: Drucker-prager elastoplasticity for sand animation. *ACM Trans. Graph. 35*, 4 (July 2016), 103:1–103:12. URL: http://doi.acm.org/10.1145/2897824.2925906, doi:10.1145/2897824.2925906. 2

[KPNS10] KANG N., PARK J., NOH J., SHIN S. Y.: A hybrid approach to multiple fluid simulation using volume fractions. *Computer Graphics Forum 29*, 2 (2010), 685–694. URL: http://dx.doi.org/10.1111/j.1467-8659.2009.01638.x, doi:10.1111/j.1467-8659.2009.01638.x. 2

[LSSF06] LOSASSO F., SHINAR T., SELLE A., FEDKIW R.: Multiple interacting liquids. *ACM Trans. Graph. 25*, 3 (July 2006), 812–819. URL: http://doi.acm.org/10.1145/1141911.1141960, doi:10.1145/1141911.1141960. 2

[MEB*12] MISZTAL M. K., ERLEBEN K., BARGTEIL A., FURSUND J., CHRISTENSEN B. B., BÆRENTZEN J. A., BRIDSON R.: Multiphase flow of immiscible fluids on unstructured moving meshes. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Goslar Germany, Germany, 2012), SCA '12, Eurographics Association, pp. 97–106. URL: http://dl.acm.org/citation.cfm?id=2422356.2422372. 2

[MKN*04] MÜLLER M., KEISER R., NEALEN A., PAULY M., GROSS M., ALEXA M.: Point based animation of elastic, plastic and melting objects. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2004), SCA '04, Eurographics Association, pp. 141–151. URL: http://dx.doi.org/10.1145/1028523.1028542, doi:10.1145/1028523.1028542. 1

[MM13] MACKLIN M., MÜLLER M.: Position based fluids. *ACM Trans. Graph. 32*, 4 (July 2013), 104:1–104:12. URL: http://doi.acm.org/10.1145/2461912.2461984, doi:10.1145/2461912.2461984. 3

[MMCK14] MACKLIN M., MÜLLER M., CHENTANEZ N., KIM T.-Y.: Unified particle physics for real-time applications. *ACM Trans. Graph. 33*, 4 (July 2014), 153:1–153:12. URL: http://doi.acm.org/10.1145/2601097.2601152, doi:10.1145/2601097.2601152. 2

[MSKG05] MÜLLER M., SOLENTHALER B., KEISER R., GROSS M.: Particle-based fluid-fluid interaction. 237–244. URL: http://doi.acm.org/10.1145/1073368.1073402, doi:10.1145/1073368.1073402. 1, 2

[NFJ02] NGUYEN D. Q., FEDKIW R., JENSEN H. W.: Physically based modeling and animation of fire. *ACM Trans. Graph. 21*, 3 (July 2002), 721–728. URL: http://doi.acm.org/10.1145/566654.566643, doi:10.1145/566654.566643. 1

[NGL10] NARAIN R., GOLAS A., LIN M. C.: Free-flowing granular materials with two-way solid coupling. *ACM Trans. Graph. 29*, 6 (Dec. 2010), 173:1–173:10. URL: http://doi.acm.org/10.1145/1882261.1866195, doi:10.1145/1882261.1866195. 2

[NW97] NOYES A. A., WHITNEY W. R.: The rate of solution of solid substances in their own solutions. *Journal of the American Chemical Society 19*, 12 (1897), 930–934. 7

[RGJ*15] RAM D., GAST T., JIANG C., SCHROEDER C., STOMAKHIN A., TERAN J., KAVEHPOUR P.: A material point method for viscoelastic fluids, foams and sponges. In *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (New York, NY, USA, 2015), SCA '15, ACM, pp. 157–163. URL: http://doi.acm.org/10.1145/2786784.2786798, doi:10.1145/2786784.2786798. 2

[RLY*14] REN B., LI C., YAN X., LIN M. C., BONET J., HU S.-M.: Multiple-fluid sph simulation using a mixture model. *ACM Trans. Graph. 33*, 5 (Sept. 2014), 171:1–171:11. URL: http://doi.acm.org/10.1145/2645703, doi:10.1145/2645703. 1, 2, 5, 8

[RMEF09] ROBINSON-MOSHER A., ENGLISH R. E., FEDKIW R.: Accurate tangential velocities for solid fluid coupling. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2009), SCA '09, ACM, pp. 227–236. URL: http://doi.acm.org/10.1145/1599470.1599500, doi:10.1145/1599470.1599500. 2

[RMSG*08] ROBINSON-MOSHER A., SHINAR T., GRETARSSON J., SU J., FEDKIW R.: Two-way coupling of fluids to rigid and deformable solids and shells. In *ACM SIGGRAPH 2008 Papers* (New York, NY, USA, 2008), SIGGRAPH '08, ACM, pp. 46:1–46:9. URL: http://doi.acm.org/10.1145/1399504.1360645, doi:10.1145/1399504.1360645. 2

[SB12] SCHECHTER H., BRIDSON R.: Ghost sph for animating water. *ACM Trans. Graph. 31*, 4 (July 2012), 61:1–61:8. URL: http://doi.acm.org/10.1145/2185520.2185557, doi:10.1145/2185520.2185557. 3

[SKB08] STEFFEN M., KIRBY R. M., BERZINS M.: Analysis and reduction of quadrature errors in the material point method (mpm). *International Journal for Numerical Methods in Engineering 76*, 6 (2008), 922–948. URL: http://dx.doi.org/10.1002/nme.2360, doi:10.1002/nme.2360. 3

[SP08] SOLENTHALER B., PAJAROLA R.: Density contrast sph interfaces. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2008), SCA '08, Eurographics Association, pp. 211–218. URL: http://dl.acm.org/citation.cfm?id=1632592.1632623. 2

[SRC10] SEUNG-HO S., RYEOL K. H., CHANG-HUN K.: Hybrid simulation of miscible mixing with viscous fingering. *Computer Graphics Forum 29*, 2 (2010), 675–683. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2009.01637.x, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8659.2009.01637.x, doi:10.1111/j.1467-8659.2009.01637.x. 2

[SSC*13] STOMAKHIN A., SCHROEDER C., CHAI L., TERAN J., SELLE A.: A material point method for snow simulation. *ACM Trans. Graph. 32*, 4 (July 2013), 102:1–102:10. URL: http://doi.acm.org/10.1145/2461912.2461948, doi:10.1145/2461912.2461948. 1, 2, 3, 4, 5

[SSJ*14] STOMAKHIN A., SCHROEDER C., JIANG C., CHAI L., TERAN J., SELLE A.: Augmented mpm for phase-change and varied materials. *ACM Trans. Graph. 33*, 4 (July 2014), 138:1–138:11. URL: http://doi.acm.org/10.1145/2601097.2601176, doi:10.1145/2601097.2601176. 2, 3, 5

[Sta99] STAM J.: Stable fluids. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1999), SIGGRAPH '99, ACM Press/Addison-Wesley Publishing Co., pp. 121–128. URL: http://dx.doi.org/10.1145/311535.311548, doi:10.1145/311535.311548. 1

[TGK*17] TAMPUBOLON A. P., GAST T., KLAR G., FU C., TERAN J., JIANG C., MUSETH K.: Multi-species simulation of porous sand and water mixtures. *ACM Trans. Graph.* (2017). URL: http://dx.doi.org/10.1145/3072959.3073651, doi:10.1145/3072959.3073651. 2, 4, 5

[YCL*17] YANG T., CHANG J., LIN M. C., MARTIN R., ZHANG J., HU S.-M.: A unified particle system framework for multi-phase, multi-material visual simulations. In *ACM SIGGRAPH Asia 2017 Papers* (2017), SIGGRAPH Asia '17. 1

[YCR*15] YANG T., CHANG J., REN B., LIN M. C., ZHANG J. J., HU S.-M.: Fast multiple-fluid simulation using helmholtz free energy. *ACM Trans. Graph. 34*, 6 (Oct. 2015), 201:1–201:11. URL: http://doi.acm.org/10.1145/2816795.2818117, doi:10.1145/2816795.2818117. 1, 2, 6

[YJL*16] YAN X., JIANG Y.-T., LI C.-F., MARTIN R. R., HU S.-M.: Multiphase sph simulation for interactive fluids and solids. *ACM Trans. Graph. 35*, 4 (July 2016), 79:1–79:11. URL: http://doi.acm.org/10.1145/2897824.2925897, doi:10.1145/2897824.2925897. 1, 2, 8

[YLHQ14] YANG L., LI S., HAO A., QIN H.: Hybrid particle-grid modeling for multi-scale droplet/spray simulation. *Computer Graphics Forum 33*, 7 (2014), 199–208. URL: http://dx.doi.org/10.1111/cgf.12488, doi:10.1111/cgf.12488. 2

[YLM*16] YANG T., LIN M. C., MARTIN R. R., CHANG J., HU S.-M.: Versatile interactions at interfaces for sph-based simulations. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2016), SCA '16, Eurographics Association, pp. 57–66. URL: http://dl.acm.org/citation.cfm?id=2982818.2982827. 2

[YSB*15] YUE Y., SMITH B., BATTY C., ZHENG C., GRINSPUN E.: Continuum foam: A material point method for shear-dependent flows. *ACM Transactions on Graphics* (2015). 2

[ZB05] ZHU Y., BRIDSON R.: Animating sand as a fluid. *ACM SIGGRAPH 2005 Papers* (2005), 965–972. URL: http://doi.acm.org/10.1145/1186822.1073298, doi:10.1145/1186822.1073298. 1, 2

[ZCL16] ZHANG X., CHEN Z., LIU Y.: *The Material Point Method: A Continuum-Based Particle Method for Extreme Loading Cases*. Elsevier Science, 2016. URL: https://books.google.com.tw/books?id=_O8QCgAAQBAJ. 3, 4, 5