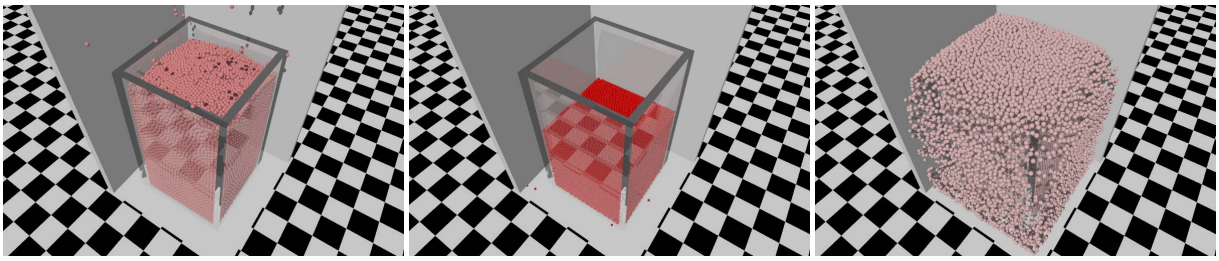
Volume Controllable SPH

Masanori Shinkawa and Yoichiro Kawaguchi

The University of Tokyo, Japan



(a) (b) (c)

Figure 1: *Our method can capture both contracting and expanding fluid behaviour. The vividnesses of the particle colours show the densities. (a) Initial state of a volume contraction and expansion test. The initial volume is 150%. (b) Fluid after volume contraction from 150% to 100%. (c) Fluid after volume expansion from 100% to 200% and overflowing out of the bucket.*

Abstract

We extend a Predictive Corrective Incompressible SPH (PCISPH) algorithm to a volume controllable SPH algorithm. In order to handle volume changes, we develop a method to control fluid volume. Our method precomputes a pressure correction factor of PCISPH at various particle volume and interpolates these factors to get a factor for arbitrary particle volume. Thanks to a PCISPH's high correctiveness, our method enables both stiff volume contraction and expansion. Additionally we derive computations for density and a force by pressure to prevent instabilities and unnatural flocks near interfaces between two fluids with different particle volume. Furthermore, granular simulators based on fluid simulators are also extensible to simulate volume change of granular materials.

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

1. Introduction

Fluid simulations for animations are one of the most tackled fields in computer graphics because they are hard to be captured and animated by human efforts. One subfield of the fluid simulations is to achieve incompressibility; fluid volume is intended to be constant. Fluid simulations without incompressibility would produce disappointing fluid animations with unnatural compressive and expansive behaviour.

In some cases, however, representing volume change is beneficial or required. For liquid with excessive number of small bubbles, naively simulating interactions of each bubble fosters expensive computational cost and forces the simulation to be infeasible. Furthermore, these bubbles contribute to volume rise, which has been ignored. For granular materials, loosely packed granular materials can be compressed by external forces. However, previous studies have assumed particle volumes to be constant and have failed to simulate volume-changing behaviour although there exist compressive granular materials such as snow. Another expanding example is sodium polyacrylate, sold as "Instant snow", which expands extremely after water absorption.

In order to break through these incapabilities, we present a preliminary but fundamental method to control fluid vol-

DOI: 10.2312/conf/EG2013/short/021-024

ume for the future simulations of compressive and expansive behaviour. Currently we have two contributions: a stiff volume controlling method, an modified evaluations of density and pressure force under heterogeneous particle volume settings to stabilize simulation and to prevent unnatural particle distribution near interfaces between multiple fluids with different density.

2. Related Work

Lagrangian fluid simulation for computer graphics was primarily initiated by Müller et al. [MCG03] based on Smoothed Particle Hydrodynamics (SPH), invented by Monaghan [Mon92]. Solenthaler et al. [SP09] present Predictive Corrective Incompressible SPH (PCISPH), which iteratively corrects particle positions to ensure incompressibility. Bubbling is challenged topic in fluid simulations. A large number of tiny bubbles is handled by Cleary et al. [CPPK07] on particle based fluid simulators, although their method separates liquid and air phases and does not contribute volume rise by these bubbles.

Instead of directly simulating every granular particle, Zhu et al. [ZB05] replace viscosity term of Navier-Stokes equation with frictional force term, producing fine-grained sand behaviour without excessive number of particles. Lenaerts et al. [LD09] derive the Lagrangian version from that by Zhu et al. Unilateral incompressibility by Narain et al. [NGL10] restricts compression above a certain density limit but allows free expansion, succeeding in free-flowing granular materials. However, these methods, based on limited-density fluid simulators, also can neither capture volume rise nor loss.

Porous flow simulation by Lenaerts et al. [LAD08] handles volume expansion by changing internal stress using SPH. However, this volume change is only applicable to elastic bodies, not to free-flowing materials such as fluids and granular materials. Geretshauser et al. [GSG*10] simulate porous dust collision and volumetric compression based on SPH. Although they can handle compressive volumetric change by external forces, their applications are trapped within passive volumetric change.

3. PCISPH and Extension for Granular Materials

We summarize PCISPH method by Solenthaler et al. [SP09] and its derivative work for granular materials by Alduán et al. [AO11]. The methods we present in this paper originate from their methods.

3.1. Original PCISPH

Fluid density is computed by the following summation.

$$\rho_i = \sum_j m_j W_{ij} \tag{1}$$

where $W_{ij} = W(\mathbf{x}_i - \mathbf{x}_j)$. At each simulation step, the following pressure correction is iteratively accumulated using density computed by Eq. (1).

$$\Delta p_i = \delta \cdot (\rho_i - \rho_{rest}) \tag{2}$$

where ρ, ρ_{rest} are a density and a rest density, respectively. δ is a pressure correction factor, which is precomputed among neighbouring particles at uniform rectangular grid points. δ is obtained with a squared summation of kernel gradients and a summation of squared kernel gradients.

$$\delta = \frac{1}{V^2\Delta t^2 \left(\left(\sum_j \nabla W_{ij}\right)^2 + \sum_j \left(\nabla W_{ij}\right)^2 \right)} \tag{3}$$

where $V = m/\rho$.

3.2. Unilateral Incompressibility, Friction and Cohesion

Unilateral incompressibility sets an upper limit of density and exhibits free-flowing granular behaviour. A pressure correction with a density limit is

$$\Delta p_i = \delta \cdot \max(0, \rho - \rho_{rest}) \tag{4}$$

Besides pressure, internal stress $\mathbf{s}$ is also corrected.

$$\Delta \mathbf{s} = \mathbf{D}^{-1} \nabla \mathbf{u}^T, \text{ with } \mathbf{D} = \frac{m\Delta t}{\rho} \sum_j \frac{1}{\rho^2} \nabla W_{ij} \nabla W_{ij}^T \tag{5}$$

where $\mathbf{D}^{-1}$ is a internal stress correction factor, which is precomputed, as with δ from PCISPH. Piling and cohesive behaviour is represented by forcing following yield criteria on internal stress.

$$\begin{array}{ll} |s_{ij}| < \alpha p & \text{for deviatoric elements} \\ |s_{ij}| < \beta^2 C & \text{for all elements} \end{array} \tag{6}$$

where α and C are controlling parameters. β is a cohesion intensity and is updated at each simulation step.

4. Controlling Volume

Because PCISPH aims to correct density to a fixed value under a constant particle mass, we modify pressure computation from Eq. (3,4), so that variable volume can be handled.

We represent volume change of fluid by volume change of particles. However, pressure correction factor is computed by virtual particles. Because pressure correction factor is dependent on particle volume and on the distances from the others, precomputing singular pressure correction factor contradicts heterogeneous particle volume settings.

In order to solve this problem, we interpolate precomputed pressure correction factor for various particle volume V. Our key idea is illustrated in Fig. 2. First, we precompute δ under different particle volume. Precomputation is performed among virtual particles at uniformly distanced grid points. Grid spacing l is computed by the cube root of

volume: $l = V^{1/3}$. While simulating, pressure correction requires δ at arbitrary particle volume. To obtain the δ at arbitrary particle volume V: $\delta(V)$, we interpolate these precomputed δ. The $\delta(V)$ computed by the interpolation is plugged into Eq. (3), yielding a pressure correction:

$$\Delta p = \delta(V) \cdot (\rho - \rho_{rest}(t)) \tag{7}$$

The rest density of each particle is given by $\rho_{rest}(t) = m/V(t)$.

For granular materials, we also interpolate precomputed $\mathbf{D}^{-1}$ at various particle volume, yielding $\mathbf{D}(V)^{-1}$.

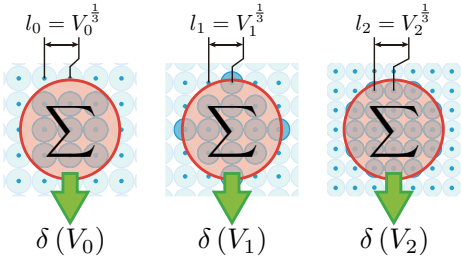


Figure 2: *$\delta(V)$ precomputation at various particle volume V_0, V_1, and V_2 ($V_0 > V_1 > V_2$), yielding $\delta(V_0)$, $\delta(V_1)$, and $\delta(V_2)$, respectively. Summations in Eq. (3) are computed with virtual filled neighbourhoods at uniform rectangular grid points. Particle spacing l is derived from the cube root of particle volume: $l = V^{1/3}$.*

5. Modified Evaluations of Density and Pressure Force

In this section we present some modifications required to prevent spurious particle distribution under heterogeneous particle volume simulations.

Solenthaler et al. [SP08] raise and resolve two problems under conditions of constant particle volume: spurious particle crowding and interfacial tension near interfaces with high density contrast. They define particle density as $\phi_i = \sum_j W_{ij}$ and evaluate density at each particle by $\rho_i = m_i \phi_i$. Monaghan's version of pressure force [Mon92] is also modified into $\mathbf{F}_{pressure} = -\sum_j \left(p_i/\phi_i^2 + p_j/\phi_j^2 \right) \nabla W_{ij}$ to avoid underivability of density near the interfaces. These modifications prevent unnatural surface tension and particle overlapping near fluid interfaces with a high density contrast.

However, particle volume also affects density. As shown in Fig. 4(b), particle crowds revive near interfaces with density contrast caused by particle volume. Furthermore, overlaps of particles lead to falsified decrease of entire volume, degrading volume controllability.

Thus we first extend their particle density model to that dependent on particle volume by weighting contributions of ϕ_i with volume ratio $V_{ij} = V_i/V_j$:

$$\phi_i = \sum_j V_{ji} W_{ij} \tag{8}$$

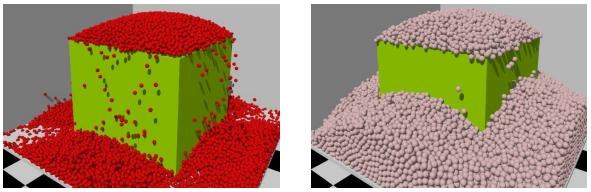


Figure 3: *An expanding Instant snow. (left) Initial pile. (right) After expansion.*

Scenario	Particles	Time (sec)	Iterations
Cont. and Exp.	39k	0.091	3.9
Instant snow	41k	0.076	3.0
Rayleigh-Taylor	48k	0.093	3.1

Table 1: *Performance results*

This extension prevents overestimation and underestimation of smaller and larger neighbourhoods, respectively.

We also modify $\mathbf{F}_{pressure}$ because ϕ_i is unsmooth and underivable near density interfaces caused by particle volume difference, which trigger numerical instability. As Solenthaler et al. [SP08] utilize a smooth and derivable density equivalent ϕ_i, we further make particle density smooth by multiplying particle density with particle volume to obtain a further density equivalent: $\varphi_i = V_i \phi_i$. Following the steps in which Solenthaler et al. [SP08] derive $\mathbf{F}_{pressure}$, we first discretise $\nabla p_i/\varphi_i = \nabla(p_i/\varphi_i) + \left(p_i/\varphi_i^2\right) \nabla \varphi_i$, yielding

$$\frac{\nabla p_i}{\varphi_i} = \sum_j \left(\frac{p_j}{\phi_j \varphi_j} + \frac{p_i \varphi_j}{\phi_j \varphi_i^2} \right) \nabla W_{ij} \tag{9}$$

Because $\mathbf{F}_{pressure} = -\nabla p/\phi$ and $\varphi_i = V_i \phi_i$,

$$\mathbf{F}_{pressure} = -\sum_j \left(V_{ji} \frac{p_i}{\phi_i^2} + V_{ij} \frac{p_j}{\phi_j^2} \right) \nabla W_{ij} \tag{10}$$

These derivations also apply to the force by internal stress

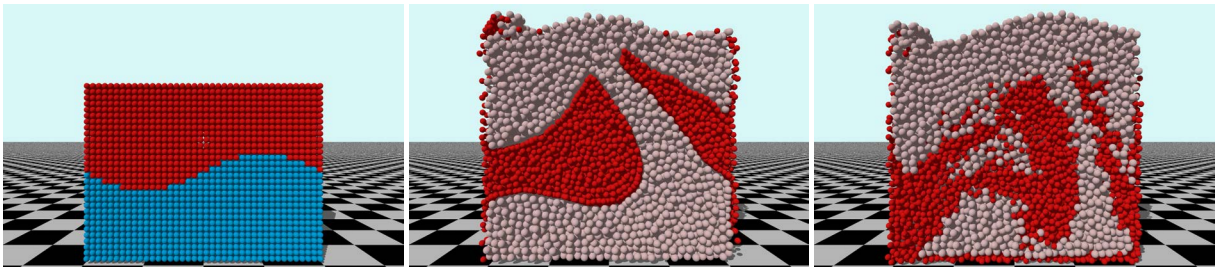
$$\mathbf{F}_{friction} = \sum_j \left(V_{ji} \frac{\mathbf{s}_i}{\phi_i^2} + V_{ij} \frac{\mathbf{s}_j}{\phi_j^2} \right) \nabla W_{ij} \tag{11}$$

Viscous and pressure force by Müller et al. [MCG03] are computed in the same manner as that by Solenthaler et al. [SP08] with adopting our Eq. (8) as their particle density.

6. Results and Discussion

All simulations are run on a personal computer with 3.20 GHz Quad-Core CPU, 24 GB RAM, and NVIDIA Geforce 560 Ti GPU. All the simulations are implemented on GPU. Animations are rendered by Pov-Ray. Timing and average PCISPH iteration results are shown in Table 1.

Fig. 1 shows the result of contracting and expanding fluid



(a) (b) (c)

Figure 4: *Cross sectional views of Rayleigh-Taylor instability simulation. (a) Initial state the simulation. The volume of blue particles at the lower layer double. (b) and (c) are the result at $t = 1.67s$ without and with our modified density and pressure force computations, respectively. Without our method, the smaller particles flock at the interface.*

simulation. Fluid is filled in a bucket and is forced to contract and expand serially. Initial condition is shown in Fig. 1(a). Fig. 1(b) shows volume contraction from 150% to 100%. After contraction, we force expansion from 100% to 200%, resulting in overflow out of the bucket, as shown in Fig. 1(c). Expanding granular material is also shown in Fig. 3.

In Fig. 4, Rayleigh-Taylor instability is simulated by heterogeneous particle volume. Fig. 4(a) shows the initial condition. All particles have the same volume and blue particles expand. The wavy interface is introduced to trigger the instability. Fig. 4(b) shows the result without our modified density and pressure force evaluations. Unnatural crowd of particles is observed near the interfaces. On the other hand, introducing our method adaptively distribute particles even on heterogeneous particle volume settings as shown in Fig. 4(c).

Our method would fail in extremely volume-changing scenarios. Too much volume rise pushes all neighbouring particles out of the kernel support, destabilizing the simulation. The more particles are within kernel support for volume decrease of particles, the more expensive the computational cost becomes. To overcome these drawbacks, adaptive sampling techniques could be a saviour; for large volume, particle splitting keep all neighbouring particles within kernel support; for small volume, particle merging limits the number of neighbouring particles.

7. Conclusion and Future Work

We obtain both expansive and contractive fluid animations through our volume controllable method. Our method also allows simulations of free-flowing granular materials by replacing viscosity with frictional stress. Furthermore, our modified density and pressure force evaluations prevent particles near interfaces from flocking even for heterogeneous particle volume settings.

Our method succeeds in transitional volume change while we do not take any account of reactive volume change, such as compaction by outer forces and volume-raising external stimuli, which is required in future challenges such as snow and liquid with excessive number of bubbles.

References

[AO11] ALDUÁN I., OTADUY M. A.: SPH granular flow with friction and cohesion. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2011), SCA '11, ACM, pp. 25–32. 2

[CPPK07] CLEARY P. W., PYO S. H., PRAKASH M., KOO B. K.: Bubbling and frothing liquids. In *ACM SIGGRAPH 2007 papers* (New York, NY, USA, 2007), SIGGRAPH '07, ACM. 2

[GSG*10] GERETSHAUSER R. J., SPEITH R., GÜTTLER C., KRAUSE M., BLUM J.: Numerical simulations of highly porous dust aggregates in the low-velocity collision regime*. *Astronomy & Astrophysics 513* (2010), A58. 2

[LAD08] LENAERTS T., ADAMS B., DUTRÉ P.: Porous flow in particle-based fluid simulations. In *ACM SIGGRAPH 2008 papers* (New York, NY, USA, 2008), SIGGRAPH '08, ACM, pp. 49:1–49:8. 2

[LD09] LENAERTS T., DUTRÉ P.: Mixing fluids and granular materials. *Computer Graphics Forum 28*, 2 (2009), 213 – 218. 2

[MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), SCA '03, Eurographics Association, pp. 154–159. 2, 3

[Mon92] MONAGHAN J. J.: Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics 30*, 1 (1992), 543–574. 2, 3

[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. 2

[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. 3

[SP09] SOLENTHALER B., PAJAROLA R.: Predictive-corrective incompressible SPH. In *ACM SIGGRAPH 2009 papers* (New York, NY, USA, 2009), SIGGRAPH '09, ACM, pp. 40:1–40:6. 2

[ZB05] ZHU Y., BRIDSON R.: Animating sand as a fluid. In *ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), SIGGRAPH '05, ACM, pp. 965–972. 2