Pacific Graphics (2012)
C. Bregler, P. Sander, and M. Wimmer (Editors)

Real-time Particle Fluid Simulation with WCSPH

Fengquan Zhang[1] and Xukun Shen[1] and Xiang Long[1] and Bin Zhao [1] and Lei Hu [1]

[1] State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China

Abstract

In this paper, based on weakly compressible smoothed particles hydrodynamics (WCSPH), we present a method for simulating high quality fluid. We propose a new pressure equation to reduce time overhead, speed up convergence towards small density fluctuations and allow for a larger time step than that of traditional approaches. And based on the new equation, we introduce a new adaptive time-stepping approach that automatically adapts the time step according to each particle current individual time step, thus improving the efficiency of computation and contributing much to a real-time particle fluid simulation. For real-time simulation, we make full use of CPU and GPU in a combined way to build a parallel platform, namely Multi-GPU platform. And optimization of the platform design is made to run a simulation with realistic visual effects in real time.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism —Animation

1. Introduction

Incompressibility is an important factor protecting the details of fluid, such as splashes, droplets and turbulent motion. SPH [MG92] was originally designed for compressible fluid, but it can be extended to nearly incompressible fluid. The most commonly used approach of SPH tries to ensure the uniformity of particle density by introducing a stiff equation of state, however, the time step of this method is only one tenth as long as that in the normal state. As a result, people have to find a balance between the time step and the efficiency of the simulation. Using the ideal gas equation to relate pressure and density results in high compressibility and large density fluctuation. Using the Poisson equation takes a great amount of computation to solve the velocity field dissipation. It is difficult to guarantee the real-time of simulation with the increase in the number of particles. Weakly compressible SPH method, however, can handle similar computation with less time.

In this paper we present a new efficient pressure equation for weakly compressible fluid simulation, which avoids excessive time-consumption and speeds up convergence towards small density fluctuations. Furthermore, it permits a much larger time step than the methods in [Mon94] [BT07]. Based on this, an adaptive time-stepping method is proposed to reduce the waste on computing resources. It can automatically estimate appropriate time steps independently of the scenario. Compared with simulation using constant time step, using adaptive time step method needs much shorter simulation time.

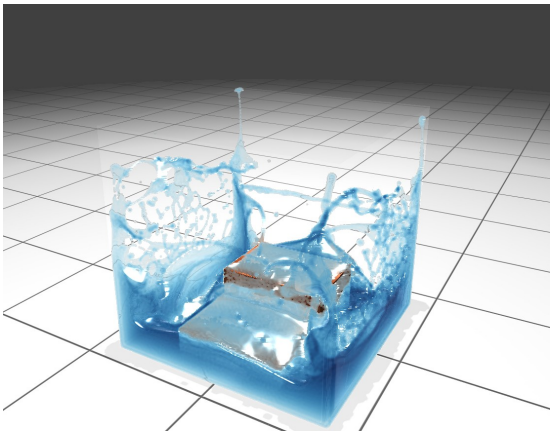

Figure 1: A dam breaking around a stair, 220k particles. Simulatied using our method

The whole process of WCSPH simulation and rendering, including neighborhood searching, physical computing, particles surface extracting and rendering, is implemented on the Multi-GPU platform.

DOI: 10.2312/PE/PG/PG2012short/029-034

2. SPH fluid model

2.1. Computation model

In this paper, we simulate all of our example using the weakly compressible SPH framework [BT07] [MCG03]. The SPH method for solving the Navier-Stokes equation is a Lagrangian mesh-free model that allows one to solve the continuum hydrodynamic equation with a set of interacting fluid particles. It is an interpolation method with a kernel function. SPH carries out computation in a local neighborhood of each particle using radial symmetrical smoothing kernels. For more understanding of SPH, please refer to [Mon05].

2.2. The new pressure equation

There are several ways to compute pressure for WCSPH [Mon94] [BT07]. But these methods either require smaller time steps due to big exponential compared to the time-consuming Poisson method or cannot assure the stability in complex interaction scene. Furthermore, stability and convergence towards uniform density tend to recede. For incompressibility fluid, the Poisson equation $\nabla^2 P = \rho \frac{\nabla v}{\nabla t}$ is a commonly solved method in Eulerian approaches and Lagrangian approaches. But, it is time-consuming. Using the traditional ideal gas equation of Eq. 1 results in a rather high compressibility. The ideal gas equation and Tait's equation have the form

$$P = k(\rho - \rho_0) \tag{1}$$

$$P = B((\frac{\rho}{\rho_0})^{\gamma} - 1) \tag{2}$$

where k is stiffness parameter, $\gamma = 7$ and $B = \frac{\rho_0 c_s^2}{\gamma}$ where c_s is chosen so that the speed of sound is large enough to keep the density fluctuation small.

Our pressure equation is inspired by the model of Lennard-Jones and Soft-Sphere and the work of [TS07] [XHT03] [ZZW07]. They adapt the Lennard-Jones model to simulate fluid in the SPH framework, and their primary interest is in simulation accuracy. In SPH, the Lennard-Jones potential is represented by:

$$P_{ij} = D[(\frac{r_0}{r_{ij}})^{12} - (\frac{r_0}{r_{ij}})^{6}]\frac{\mathbf{r}}{r^2} \tag{3}$$

where D is taken as the square of the largest velocity and r_0 is the smoothing distance. r_{ij} is the distance of between the particles and $\mathbf{r}$ is the particle distance vector.

In Eq.1, the simple proportional increase leads to the large fluctuations of density. The relationship between pressure and density is similar with what between force and molecules as described in Eq.3. When the density of fluid gradually increases, the pressure would decrease at first, and then increase. For towards small density fluctuations and positive pressure, considering the value of twice reference density, we propose a new pressure equation based on Eq. 2 and Eq. 3, which can avoid the negative influence from traditional high exponent equation and high compressibility from ideal gas equation. It can be written as a segmented function as follows

$$P = K \begin{cases} (\frac{\rho - \rho_0}{\rho_0})^{\alpha} - (\frac{\rho - \rho_0}{\rho_0})^{\beta} & \rho \geq 2\rho_0 \\ (\frac{\rho}{\rho_0})^{\alpha} - (\frac{\rho}{\rho_0})^{\beta} & \rho_0 \leq \rho < 2\rho_0 \\ 0 & otherwise \end{cases} \tag{4}$$

where ρ_0 is a reference density, K is the pressure constant to keep the relative density fluctuation $(\rho\text{-}\rho_0)/\rho_0$ small. α and β, the exponent constant is similar to γ in Eq. 2. But we can use smaller exponent than γ. Since compressibility effects are $O(M^2)$ with M denoting the Mach number, we can derive the K from

$$\frac{|\rho - \rho_0|}{\rho_0} \sim \frac{v^2}{c_s^2} \tag{5}$$

where v is the maximum speed of the fluid and c_s is the speed of sound in the fluid. We set $\frac{|\rho-\rho_0|}{\rho_0} \sim 0.01$, if $\frac{v}{c_s} < 0.1$; that is, density fluctuation η is 1%. The K can be defined as

$$K = \frac{\rho_0 c_s^2}{\alpha + \beta} \tag{6}$$

For example, if H, g are constant, we can compute the $v \sim \sqrt{2gH}$ for the maximum velocity, $c_s = \sqrt{200gH}$. After a large number of experiments, we find that the best value $\alpha = 3$, $\beta = 2$ are used for low viscosity fluids, $\alpha = 2$, $\beta = 1$ are used for high viscosity fluids. Experiments indicate that the new pressure is less time-consuming, and low exponent can get rid of the constraints of time step. It speeds up convergence towards small density fluctuations, allows for a larger time step than traditional approaches. Furthermore, the new pressure equation can be applied to various kinds of fluids such as newtonian and viscoelastic fluids only by adjusting the exponents of α and β. A detailed quantitative discussion will be summarized on Sec. 5.

2.3. Adaptive time-stepping

During the simulation, the time step of a particle should be small enough to avoid instabilities, but a too small time step will affect the efficiency. In constant time stepping schemes, the minimum time step is used for each particle during the whole simulation time, as shown by the red circles in Fig. 2. Such a time step may be too restrictive for some particles. As a consequence, a large number of reiterated computations have to be made, thus costing a lot of time. In this consideration, time steps of Δt_{min}, $1.5\Delta t_{min}$ and $2.0\Delta t_{min}$ are tested. Experiment shows that we can use a large time step by adding some extra conditions. So we integrate the equation of particle motion according to current individual time step.

An important criterion for numerical stability and

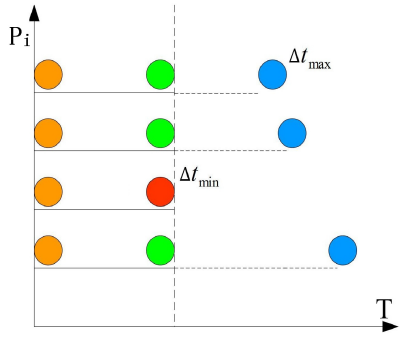

Figure 2: Orange is the initial time of particle. Red is the updating time of the shortest time step. Blue is the updating time of the longest time step. In the adaptive time-stepping simulation, it firstly iterates simulation using a constant time step computed by Eq. 7 and Eq. 8 as red and green denoting, and then increases time step by below certain conditions.

convergence is the CFL condition. For SPH, the criterion is that a particle i can not move a distance longer than its smoothing length h in one time step. We have

$$\Delta t_i \leq \lambda_v(\frac{h}{v^{max}}) \tag{7}$$

where Δt is the time step, h is the smoothing length, and $v^{max} = (\| V_i(t) \|)$ is the maximum velocity. Moreover, the simulation using particle-based model depends largely on good approximations of speed and density sampled from particles. The time step is derived from the CFL condition of the viscous term and the force term. Therefore, two extra conditions are added to make the time step also satisfy

$$\Delta t_i \leq \lambda_f(\sqrt{\frac{h}{\| a_i \|}}) \ \ and \ \Delta t_i \leq \lambda_\nu(\frac{h}{c_s + 0.6(c_s\mu)}) \tag{8}$$

Following [MG92], we set $\lambda_v = 0.1$, $\lambda_f = 0.25$, $\lambda_\nu = 0.3$ and let a_i be the current acceleration of particle i. c_s is the speed of sound in the fluid and μ is the viscosity coefficient. The adaptive time-stepping algorithm for WCSPH can be summed up as follows:

1) Use the minimum time step ΔT_0 to enforce the global iteration, $\Delta T_0 = min(\Delta t_i)$, where ΔT_0 meets the three criteria above and each iteration step adds one more ΔT_0 to the time step.

2) Compute the individual time step of particle T_i and set the length of time since the last updating T_{passed}, $T_{passed} = 0$.

3) During each ΔT_0 iteration, compute the $\Delta T_0 + T_{passed}$, where the $T_{passed} = T_{i+1} - T_i$. If $\Delta T_0 + T_{passed} \geq T_i$, update the current velocity and position using T_i. If $\Delta T_0 + T_{passed} < T_i$, then the density term and force term are computed by

$$A_{now} = (1 - \frac{T_{passed}}{T_i - \Delta T_0})A_0 + \frac{T_{passed}}{T_i - \Delta T_0}A_i \tag{9}$$

and the current velocity and position are computed using $\Delta T_0 + T_{passed}$, where A denotes the physical parameters of density or force term.

The proposed method can solve two problems. One is to reduce the number of iterations and save the simulation time. The other is to make the time step independent of the scene, i.e. we no longer need to reset the time step when there is a change in scenario, because it can adaptively find an appropriate time step. Therefore, a large time step rather than a constant time step should be used in the simulation. More detailed experimental results are available in Sec.5.

3. Surface Extraction

The surface of the fluid is defined as the scalar field of the following function:

$$\phi(x) = \sum_j \frac{m_j}{\rho_j} W(\boldsymbol{r}_{ij}, h^*) \tag{10}$$

where $\boldsymbol{r}_{ij} = r_i - r_j$, and the smoothing length is replaced by an adaptive smoothing length h^*. The standard SPH density kernel has a variable smoothing length, which can only reflect the interparticle distance variation in time and space. It will not show any feature of the directional effects. If it is used for simulating scenes with anisotropic deformations, a good deal of details will be lost. Besides, the spherical smoothing function is not fit to illustrate the density distribution near a surface. Thus, we choose to use the cubic spline based one-dimensional smoothing function W_1 of ellipsoid shape is devised by [ML85], which can be arbitrarily oriented as follows:

$$W(\boldsymbol{r}_{ij}, h) = \alpha_d \times \begin{cases} \frac{2}{3} - k^2 + \frac{1}{2}k^3 & 0 \leq k < 1 \\ \frac{1}{6}(2-k)^3 & 1 \leq k < 2 \\ 0 & k \geq 2 \end{cases} \tag{11}$$

where $\alpha_d = \frac{1}{h}$ in one-dimensional space and $k = \frac{r}{h}$, k is the distance between two particles normalized by the smoothing length h, and r is the real distance between two particles. The 3D cubic spline kernel is given by

$$W(\boldsymbol{r}_{ij}, h^*) = W_3(\boldsymbol{r}_{ij}, h^*_{1,2,3}) = W_1(x', h_1)W_1(y', h_2)W_1(z', h_3) \tag{12}$$

where W_1 and W_3 are the 1D and 3D cubic spline kernels. x', y' and $z'$are the projections of $\boldsymbol{r}_{ij}$ along each of these axes. h_1, h_2 and h_3 are the three components of semimajor axes of the ellipsoid. Having obtained this list of nearest neighbors, particle r_i's smoothing length of $h^i_{1,2,3}$ is defined to be

$$h^i_{1,2,3} = r_{imax} - r_i \tag{13}$$

where r_{imax} is the position vector of particle r_i's most distant nearest neighbor. $h_1 = |h_1|$, $h_2 = |h_2|$ and $h_3 = |h_3|$ are the projections of vector of $h^i_{1,2,3}$ along each of these axes. We then use the marching cubes method to extract the triangle mesh, corresponding to the fluid surface.

4. Multi-GPU Implementation

WCSPH can be parallelized well since there are almost no data dependencies. In our work WCSPH is implemented on the Multi-GPU, including neighborhood search, physical simulation, surface extraction and rendering. We improve the traditional implementations to enable the data to be computed on 4 GPUs preferably.

5. Results

This section shows the results of applying the proposed method to several test cases. All results are generated on a computer equipped with Windows7 OS, Intel Xeon Eight Core E5620 @2.40GHZ and NVIDIA GeForce GTX 480 GPU. The algorithm is implemented on four GPUs by using CUDA architecture.

5.1. Volume preservation

In order to illustrate the effect of new pressure equation, we compare it with three equations of Eq. 1, Eq. 2 and Eq. 4 respectively under the same particle conditions. The experiment scene of bunny landing has 300K particles. For the gas equation 1, the stiffness parameter k is set to 1000 and the reference density ρ_0 is set to 1. For the Tait equation 2 and new equation 4, the density fluctuation $\eta = 0.01$. The height $H = 3m$, $g = 9.80m/s^2$, $\rho_0 = 1000$. The B and K can be computed from these parameters. For the equation 4, the $\alpha = 3$ and $\beta = 2$ are applied. First of all, the simulation space is divided into uniform grids which width is equal to the smooth radius. Then the total volume of fluid in each time step is approximated as the number of grids involving particles. For each pressure equation, we select 500 time steps in the same simulation scene, and the standard deviation (STDEV) is computed for comparison. The situation is as shown in Figure 3. The black, blue and red lines show the volume changes using three equations of ideal gas equation, Tait equation and our equation, respectively. It can be seen that our method has the smallest volume fluctuations. The Tait and our method have similar results of simulation, and tend to balance. The ideal gas method, however, continues to show a dramatic change. Moreover, to contrast the three STDEV values, our method is better than the other two methods. The visual result is shown in the accompanying video and Figure 5.

For comparison with PCISPH, we choose two scenes with varying particle sets from 220K to 504K. We set the density fluctuation $\eta = 1\%$. The adaptive time step method is used in our WCSPH. But the adaptive method does not design for PCISPH framework, we use CFL condition to update the time step for PCISPH. The performance measurements and simualtion data are shown in Table 1. Because the time step of per simulation depends on the number of executed convergence iterations, the overhead per physics time step is higher using PCISPH than our WCSPH. From Table 1, we can see the speed up of dam scene is 1.29. However, for high impact two boxes scene, the overall simualtion time is less using PCISPH than our method. The main reason is the adaptive time step limits the length of time step for the stability in solving high impact scenes.

The physical behavior and visual results of WCSPH and PCISPH are compared in Figure 6. It can be seen our WCSPH simulations are in good agreement with the PCISPH method.

The above two experiments show that our method can ensure small density ratios and protect fluid volume. Furthermore, it is applicable for the simulation of simple and complex scenes with different particles sizes.

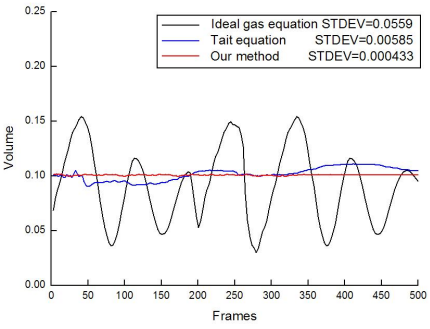

Figure 3: The relative volume change using the different equations

5.2. Adaptive time-stepping

In order to illustrate the evolution of adaptive time-stepping, we design a scene of two dams collapsing and colliding with 504k particles. In this simulation, the basic time step is set at 0.0020, and then updates by Sec.2.3 criterions. As is shown in Figure 4, the time steps per frame are changing with the evolution of simulation. Because of the collapse and collision during the simulation, the time steps have two obvious turning points, and tend to be stable until the water has small fluctuations. From this experiment it can be seen our adaptive time-stepping method can handle the scene with high impact velocities.

In order to compare the performance of the method using constant step and adaptive time step, we execute different simulation runs with varying particles sizes from 220K to 1940K as shown in Table 2. For the two boxes scene with 504k particles, the overall computaion time t_{min} is 5 minutes 40 seconds with constant time step and 3 minutes 10 seconds with our adaptive time step method, and the speed up is 1.31. Due to the high impact in two boxes scene, its speed up is smaller than dam scene, but samll details are well preserved. Furthermore, we simulate an armadillo scene with 1940K particles, and its the speed up is 2.58.

Table 1: Comparison of WCSPH and PCISPH

Model	Scene	Particle Numbers	η	Δ t	t_{sim}	Speedup
WCSPH	Dam	220k	1%	see Table 2	1min 10s	1.29
PCISPH	Dam	220k	1%	0.001	1min30s	-
WCSPH	Two boxes	504K	1%	see Table 2	3min 10s	0.74
PCISPH	Two boxes	504K	1%	0.002	2min 20s	-

Table 2: Comparison of constant time step and adaptive time step performance

Scene	Particle Numbers	GPUs	t_{sim} Adaptive	t_{sim} Constant	Speedup
Dam	220k	4	1min 20s $\Delta t_{min} = 0.000726$, $\Delta t_{max} = 0.004132$	3min 35s $\Delta t = 0.001002$	2.69
Two boxes	504K	4	4min 20s $\Delta t_{min} = 0.00176$, $\Delta t_{max} = 0.00288$	5min 40s $\Delta t = 0.002$	1.31
Armadillo	1940K	4	7min 30s $\Delta t_{min} = 0.0025$, $\Delta t_{max} = 0.0085$	19min 23s $\Delta t = 0.0030$	2.58

Table 3: The Simulation performance for four different sizes of particles sets

Particle Count	Physical Simulation	Surface Reconstruction	Overall
298K	192.34	64.15	53.15
504K	112.85	32.18	28.36
1,940K	44.18	17.12	14.28
3,049k	29.78	12.55	9.86

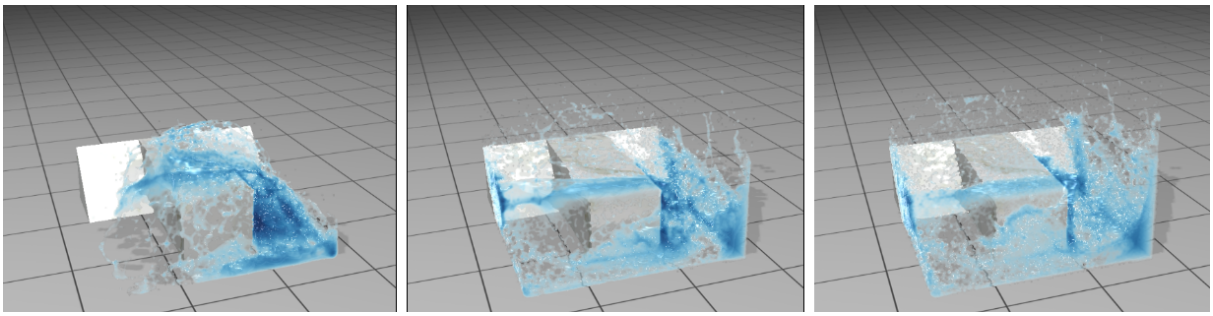

Figure 5: Comparison of gas equation(left), Tait equation(middle) and our equation(right).

5.3. Simulation Performance

In Table 3, we summarize statistics of the time-consuming of our WCSPH simulation using different sizes of particle sets. The performance of our WCSPH is measured by the number of time steps that can be executed in one second. The fourth colum lists the number of time steps that can be executed in one second. Obviously, our method is faster than the simulation on single GPU. Therefore, the platform of Multi-GPU has dominant advantages over others.

6. Conclusions and Future Work

We have presented a method for simulating and rendering interactive WCSPH fluid on the Multi-GPU platform. Instead of using the ideal gas pressure equation, we introduce a new pressure equation to ensure small density ratios and protect fluid volume. In the iterative simulation, we adapt an adaptive time-step approach to keep the numerical convergence and save the fluid simulation time. Using our highly optimized framework, the performance of the WCSPH model is greatly improved, and better results than those of other state-of-the-art implementations are obtained.

Nevertheless, there are still has some issues to be solved in the future. We will carry out a further research on CPU-GPU hybrid cluster platform architecture to optimize overall performance. In addition, to speed up the calculations

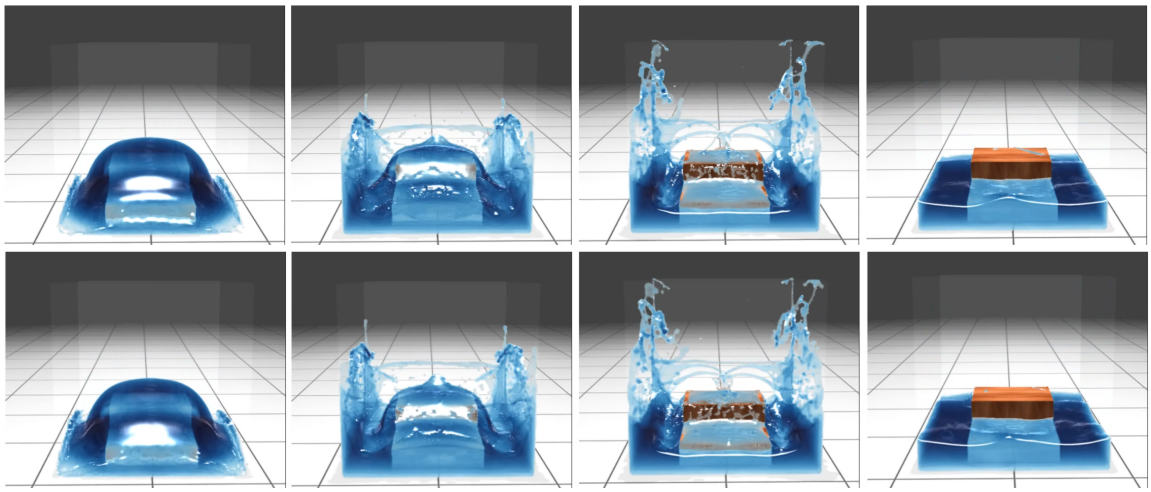

Figure 6: Comparison of breaking dam with 220K particles simulated with PCISPH(upper row) and our WCSPH(lower row).

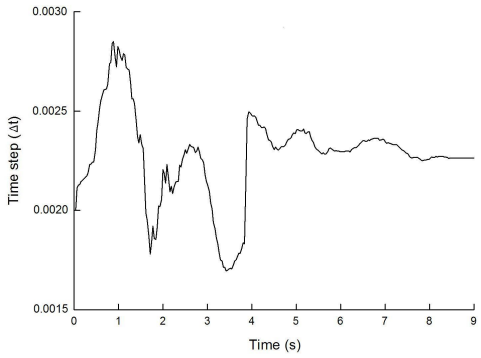

Figure 4: Time step evolution for the collision scene

further, we can use a more efficient algorithm for the simulation, such as the adaptive sampling [APK07], and multi-scale particle simulation [BM11]. For fast rendering, we will use a more efficient algorithm for the surface reconstruction, such as screen space meshes [HS09].

7. Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments, Xu Liuqing for correction. This work is supported by National Basic Research Program of China(973 Program: 2010CB734104) and Beijing Natural Science Foundation (Support ID:4102033).

References

[APK07] ADAMS B., PAULY M., KEISER R.: Adaptively sampled particle fluids. *ACM Transactions on Graphics 26*, 3 (2007), 48:1–48:7. 6

[BM11] BARBARA S., MARKUS G.: Two-scale particle simulation. *ACM Transactions on Graphics 30*, 4 (2011), 81. 6

[BT07] BECKER M., TESCHNER M.: Weakly compressible sph for free surface flows. *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation* (2007), 66–72. 1, 2

[HS09] H. C., STAADT. O. G.: Interactive screen-space surface rendering of dynamic particle clouds. *Journal of Graphics, GPU and Game Tools*, 3 (2009), 137–146. 6

[MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. *In Symposium on Computer Animation* (2003), 154–159. 2

[MG92] MONAGHAN J., GINGOLD R.: Smoothed particle hydrodynamics. *Annu.Rev.Astron Astrophys 30* (1992), 543–574. 1, 3

[ML85] MONAGHAN J., LATTANZIO J.: A refined particle method for astrophysical problems. *Astron. Astrophys 149* (1985), 135–143. 3

[Mon94] MONAGHAN J.: Simulating free surface flows with sph. *Journal of Computational Physics 110*, 2 (1994), 399–409. 1, 2

[Mon05] MONAGHAN J.: Smoothed particle hydrodynamics. *Report on Progress in Physics 68* (2005), 1703–1759. 2

[TS07] THOMAS W., SUDHANSHU K.: Acoustic modeling of reverberation using smoothed particle hydrodynamics. *University of Colorado at Colorado Springs* (2007). 2

[XHT03] XU C., HE Y., TAO W.: Comparisons of transport property of lennard-jones fluid with soft-sphere fluid: A molecular dynamics simulation study. *J. University of Shanghai for Science and Technology 25*, 4 (2003). 2

[ZZW07] ZHENG J., ZHANG J., WEI Y.: The method for choosing the parameters in the lennard-jones force model in sph and numerical verification. *Sciencepaper online:http://www.paper.edu.cn/index.php/default/releasepaper /content/200712-115* (2007). 2