Recovering Turbulence Details using Velocity Correction for SPH Fluids

Xiaokun Wang*
Sinuo Liu*
Beijing Advanced Innovation Center for Materials Genome Engineering, University of Science and Technology Beijing

Xiaojuan Ban†
Yanrui Xu
banxj@ustb.edu.cn
Beijing Advanced Innovation Center for Materials Genome Engineering, University of Science and Technology Beijing

Jing Zhou
Cong Wang
School of Computer and Communication Engineering, University of Science and Technology Beijing

ABSTRACT

In general, kinetic energy of water molecules at translational rotational degree of freedoms (DOFs) occupies the dominant position. However, coarse space discretization always results in severe numerical dissipation if only the linear kinetic energy is considered. Therefore, we proposed a novel turbulence refinement method using velocity correction for SPH simulation. In this method, surface details were enhanced by recovering the energy lost in DOFs for SPH particles. We used a free vortex model to convert particles' diffused and stretched angular kinetic energy to its neighbours' linear kinetic energy. Turbulence details would be efficiently generated using the shear between slices. Compared with previous methods, our method can generate turbulence and vortex more vividly and stably.

CCS CONCEPTS

• **Computing methodologies → Physical simulation**.

KEYWORDS

turbulence simulation, SPH, vortex-based method

ACM Reference Format:
Xiaokun Wang, Sinuo Liu, Xiaojuan Ban, Yanrui Xu, Jing Zhou, and Cong Wang. 2019. Recovering Turbulence Details using Velocity Correction for SPH Fluids. In *SIGGRAPH Asia 2019 Technical Briefs (SA '19 Technical Briefs), November 17–20, 2019, Brisbane, QLD, Australia.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3355088.3365145

1 INTRODUCTION

Fluid simulation is a hot topic in computer graphics, which was first introduced by Stam in 1999 [Stam 1999]. As one of the most popular approaches for fluid simulation in computer graphics, Smoothed Particle Hydrodynamics(SPH) has been widely used to generate fluid animation with lively details and vivid motions [Koschier et al. 2019]. Though many researchers have proposed novel models for animating various materials and enforcing incompressibility, there still has much work to do to enhance the realistic visual effects of complex phenomena. The simulation of turbulent details, for example, due to numerical dissipation[Fernando et al. 2015] or coarse sampling of grids[Kim et al. 2013], is still a tough nut.

Vortex-based methods aim at creating and preserving turbulence through vorticity field, which include vorticity confinement methods and Lagrangian vortex methods. Vorticity confinement (VC) methods recover existing vortexes and enhanced them by adding a new force [Fedkiw et al. 2001; Lentine et al. 2011; Steinhoff and Underhill 1994; Zhu et al. 2010]. Lagrangian vortex (LV) methods build on the vorticity representation of Navier-Stokes equations [Angelidis and Neyret 2005; Park and Kim 2005; Pfaff et al. 2012]. However, VC methods always add more energy than dissipation, and only existing vortices can be amplified. This results in an inefficiency in liquid scenes, even ones where there are great possibilities to generate visible vortices structures. Recently Chu and Thuerey [Chu and Thuerey 2017] proposed a deep-learning based synthesis method and succeeded in enhancing the quality of fluid animation.

To solve the problems mentioned above, we recover the linear velocity from missing angular velocity to enhance turbulent detail. In the ideal state of SPH approach, particles used to discretize space is small enough so the energy dissipation of angular kinetic could be safely ignored without affecting the overall performance. However, when high efficiency is desired and the particle size is large, the inertia tensor absent from the equation will result in severe numerical dissipation. Therefore, we use the particle as the core due to the sheer and refine the velocity field with potential flow model [Batchelor and Batchelor 1967] to affect the velocity of neighboring particles. Though the velocity of the center in a free vortex is extremely large, actually the sheer makes the core rotates like a rigid body. The turbulence and vortex effects can be restored without causing unstable results.

2 TURBULENCE REFINEMENT FOR SPH FLUIDS

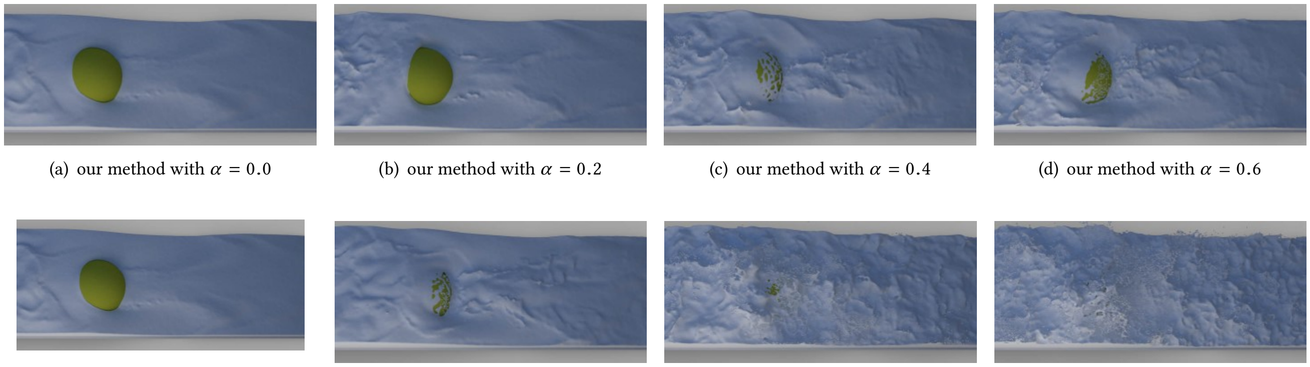
2.1 SPH-based Fluid Simulation

SPH is a popular method for simulating continuum like incompressible fluids. Physical attributes A can be derived using a user-defined

*Co-first author, contributed equally
†Corresponding author

SA '19 Technical Briefs, November 17–20, 2019, Brisbane, QLD, Australia

ACM ISBN 978-1-4503-6945-9/19/11…$15.00
https://doi.org/10.1145/3355088.3365145



(a) our method with $\alpha = 0.0$ (b) our method with $\alpha = 0.2$ (c) our method with $\alpha = 0.4$ (d) our method with $\alpha = 0.6$

(e) MP method with $v_t = 0.0$ (f) MP method with $v_t = 0.2$ (g) MP method with $v_t = 0.4$ (h) MP method with $v_t = 0.6$

Figure 1: Simulation of 94k/s turbulent fluid particles (with radius $r = 0.1m$) with a hemisphere using our method(first row) and MP method(second row) with different parameters. In the first row, from left to right is our method with control parameter α = 0.0, 0.2, 0.4, 0.6 respectively. In the second row, from left to right is MP method with transfer coefficient v_t = 0.0, 0.2, 0.4, 0.6 respectively. The turbulence effects increase as the parameter augmented.

kernel W with its neighbor information. An arbitrary field A can be expressed as:

$$A(\boldsymbol{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\boldsymbol{r} - \boldsymbol{r}_j, h) \tag{1}$$

where j represents the neighbour particle of i within the supporting radius h. And m_j is the mass, ρ_j is the density, $\boldsymbol{r}_j$ is the position.

Using SPH equation, the Navier-Stokes equations for incompressible fluids can be solved[Müller et al. 2003]:

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2}$$

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\nabla p + \rho \boldsymbol{g} + \mu \nabla^2 \boldsymbol{u} \tag{3}$$

where $\boldsymbol{u}$ the linear velocity, p is the pressure, $\boldsymbol{g}$ is the gravity and μ is dynamic viscosity coefficient.

2.2 Energy Dissipation in SPH

Typically, particles in SPH simulation only have three degrees of freedom (DOF) to reflect their translational motion. But severe energy dissipation would occur if rotational DOFs is not considered. For instance, a fixed axis rotation on a disk around the mass center in two dimensional scenario, as shown in Fig 2 (left). The total kinetic energy of this disk can be expressed as: $E_r = \frac{1}{2} J\omega^2$, where J is rotational inertia and ω is angular velocity. In this case, the rotational kinetic energy is equal to the total kinetic energy. Then consider another situation, discretizing the disk into particles, as shown in Fig 2 (right). The total kinetic energy could be split to two part:

$$E_r = \sum E_{r_i} + \sum E_{k_i} \tag{4}$$

where E_{r_i} and E_{k_i} are the rotational kinetic energy and linear kinetic energy of particle i respectively. If the rotational DOFs are neglected in this discretization, the total rotational energy $\sum E_{r_i}$ would be lost. Because $E_{r_i} = \frac{1}{2} J_i \omega^2$, and $J_i = m_i r_i^2 = \rho \pi r_i^4$, the rotation kinetic energy is related to the radius r_i:

$$\sum E_{r_i} \propto r_i^2 \tag{5}$$

This indicates that energy from rotational DOFs can be ignored only when the particles are infinitely small. While using macroscopic particles like SPH, energy is severely dissipated without considering rotational DOFs.

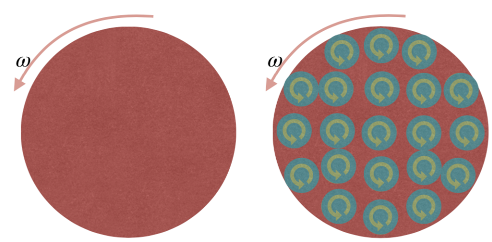


Figure 2: A two dimensional disk spinning around the mass center with an angular velocity of ω. On the right side the disk is discretized with small particles as that in SPH approach.

Therefore, we use vorticity field to approximate the angular velocity of each particle, and adopt the potential flow model to exert the diffusion of angular velocity to neighbor particles. Since the adjustment of velocity field would possibly result in slight divergence fluctuation, we integrate our method with Implicit incompressible SPH [Ihmsen et al. 2014] and Divergence-free SPH [Bender and Koschier 2015] to keep divergence-free and incompressible.

2.3 Velocity Adjustment using vorticity

Linear velocity refinement using vorticity has been widely used in lots of previous Lagrangian vortex methods. The shear between slices and the stretching term of the vortex generate the chaotic motion of the fluid. However, many of them tend to be unstable, especially when the movements are intense, even though they are theoretically stable in the continuous model. To recover the missing details, we regard each particle as the rigid core, and refine the linear velocity of its neighbors inversely proportional to the distance between them.

For computing the curl of a field $\boldsymbol{A}_i$ in SPH, we apply the difference curl formulation,

$$(\nabla \times \boldsymbol{A}_i)^{diff} = \frac{1}{\rho_i} \sum_j m_j (\boldsymbol{A}_i - \boldsymbol{A}_j) \times \nabla_i W_{ij} \quad (6)$$

where ρ_i is the density at the location of particle i, m is the mass of each particle, and W is the smoothed kernel in SPH approach, we use spline kernel in our experiments. We use this equation to derive angular velocity $\boldsymbol{\omega}$, and add an extra relax factor α which enable users to decide how rough the turbulence they desire. So the angular velocity for particle i at k_{th} time step is:

$$\boldsymbol{\omega}_i^{(k)} = \boldsymbol{\omega}_i^{(k-1)} - \alpha(\boldsymbol{\omega}_i^{(k-1)} - \frac{(\nabla \times \boldsymbol{v}_i^{(k)})^{diff}}{2}) \quad (7)$$

where α can be set between 0 to 1. When $\alpha = 1$, angular velocity will be fully determined by the vorticity field. According to this equation, the angular velocity that be used to refine linear velocity field is:

$$\delta\boldsymbol{\omega}_i^{(k)} = \boldsymbol{\omega}_i^{(k)} - \boldsymbol{\omega}_i^{(k-1)} = \alpha(\boldsymbol{\omega}_i^{(k-1)} - \frac{(\nabla \times \boldsymbol{v}_i^{(k)})^{diff}}{2}) \quad (8)$$

Using Eqn 8 we can successfully recover the rotational kinetic energy and convert it to angular velocity.

To refine linear velocity using the difference of angular velocity, we need to convert $\delta\boldsymbol{\omega}_i^{(k)}$ into $\delta\boldsymbol{v}_{i\to j}^{(k)}$ using irrotational refinement model. In our model, we take the space inside particle radius as rigid body rotation, and other space inside support radius as irrotational flow (left diagram in figure 2). Since we can get $\delta\boldsymbol{\omega}_i^{(k)}$ for each particle, we treat each particle as a rigid sphere, and make them as turbulence generators for the diffusion and stretching of its local vorticity field (see Fig 3). In this case we refine velocity for each

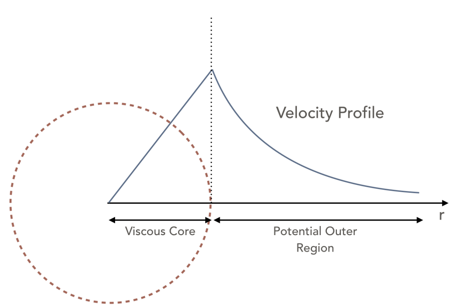


Figure 3: A solid core will be formed in the free vortex. We regard the fluid particle as the core. And outside the core the velocity is decreased proportional to the length to the vortex center.

point at the particle surface (right diagram figure 2):

$$\delta\boldsymbol{v}_{surface}^{(k)} = \delta\boldsymbol{\omega}_i^{(k)} \times \boldsymbol{r} \quad (9)$$

By inversely refining the neighbor particles within the supporting radius, we can adjust the linear velocity for every particle:

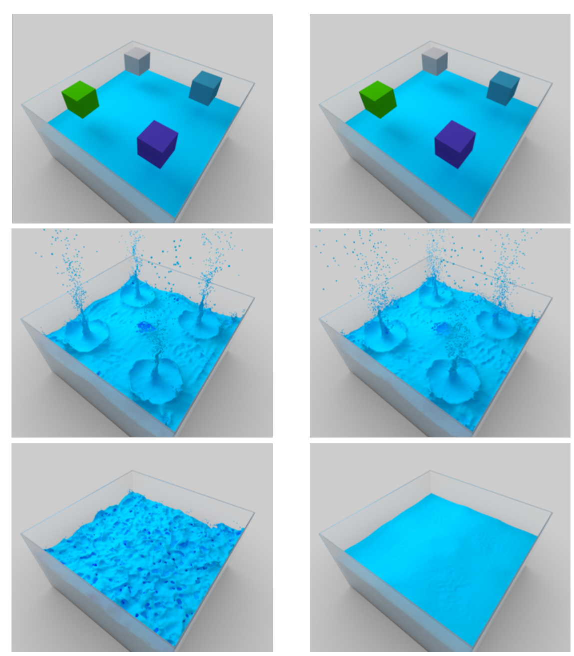
$$\delta\boldsymbol{v}_{i\to j}^{(k)} = \frac{\|\boldsymbol{x}_{ij}\|}{\|\boldsymbol{r}\|} \delta\boldsymbol{v}_{surface}^{(k)} \quad (10)$$

Consequently, we obtain the velocity refinement for all particles in a linear fashion and thus with little computation overhead.

3 EXPERIMENTS AND DISCUSSIONS

In this section, we tested our turbulence refinement in several scenes and compared it with standard fluid (SSPH) and state of the art turbulence method (please see the accompanying video for more results). We use the cubic spline kernel for our experiments.

As shown in Fig1, 94k fluid particles flush through a hemisphere in a tunnel per second. In this classical scenario, rich turbulence details are expected to be observed. We compared our method with Bender’s MP method [Bender et al. 2018] from various coefficient settings in this scene. When $\alpha = 0$ or $v_t = 0$, it is equivalent to standard SPH method. Under this coefficient, some vague and feeble turbulence is produced when the fluid is injected into the tunnel. Though the fluid then is partially blocked by the hemisphere, it forms a level difference as it flows through this obstacle, no significant interaction details are visible around the hemisphere. Behind the obstacle, two regular shallow traces gradually disappear along the flow. At the right end, the current restores calm.



(a) MP method (b) our method

Figure 4: Four blocks fall into the fluid simultaneously (with radius $r = 0.08m$). Gravitational potential energy of blocks is transformed into kinetic energy of the fluid.

In this experiment, both our method and MP method is capable of increasing turbulence details. As α increases, flow is more violent at the injection port and the traces become more irregular and apparent. Fluids are stacked in front of the obstacle using MP method and our method, and the hemisphere is almost submerged. Compared with our method, MP method increases the turbulence in a fiercer way. Fluid performance of $\alpha = 0.4$ looks as furious as that of $v_t = 0.2$. One thing should be noticed that MP increases turbulence by producing lots of small fragments, but our method tends to preserve the original form macroscopically and intensify dissipated curves. Besides, the flow at the right end seems too intense when using MP method.

Fig4 shows the disturbance caused by four blocks falling into the water at the same time. We compared our method with MP method to discuss the convergence capacity under extreme conditions. Compared to the fluid, the density of these blocks is considered large enough that the acceleration of blocks should be maintained as gravity acceleration throughout the experiment. Significant turbulence can be observed in both methods when blocks falling down. But when they reach the bottom, effects start to vary. One significant difference between these two is that turbulence in our method smooths out over time, yet unstable movements appear and don't stop when using the MP method.

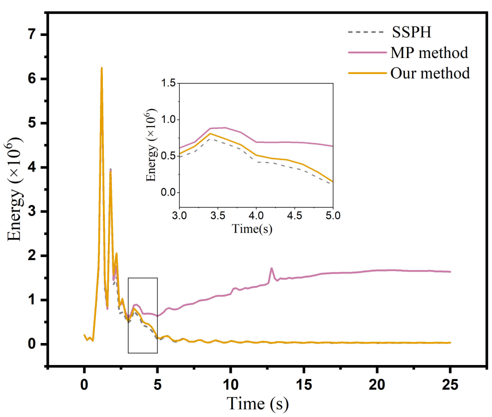


Figure 5: Total energy comparison of three methods.

Fig5 exhibits the total kinetic energy of blocks falling into water, both MP method and our method accumulate more kinetic energy than SSPH. Typically our method carries a lot kinetic energy at 1.6s. In MP method, positive feedback effect eventually overcomes the damping effect and starts to grow until a dynamic equilibrium is reached instead of a static state. Meanwhile our method converges to the stable state as SSPH does.

4 CONCLUSION

We present an SPH based method of recovering turbulence details for low viscosity incompressible fluids. Built on Lagrangian system, our method can be easily integrated with any SPH method with negligible computational overhead. It consults the fluid dynamic interpretation for the turbulence that is the performance of multiple unstable vortices interactions. By granting angular velocity that calculated through vorticity to each particle, unstable vortices are regarded as the difference of angular velocity over time steps. According to the unconditional stability of potential flow in SPH simulation process, free vortex model is preferable to adjust linear velocity and generates small vortices in the vicinity of each particle. Further, we add a coefficient α to control angular velocity in the refinement which determines the intensity of turbulence.

Numerical dissipation is an inherent defect of SPH methods, especially in simulating turbulence. It is difficult to find an intuitive way to eliminate it completely. Our method successfully solved the instability effects when the positive feedback exceeds the convergence limit of viscous damping. And it can create abundant turbulence effects while maintain stable in extreme scenes.

ACKNOWLEDGMENTS

The authors acknowledge the financial support from the National Key Research and Development Program of China (No. 2016YFB0700 500), and the National Science Foundation of China (No. 61702036, No. 61873299, No. 61572075), and Key Research Plan of Hainan Province (No. ZDYF2018139).

REFERENCES

Alexis Angelidis and Fabrice Neyret. 2005. Simulation of smoke based on vortex filament primitives. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. ACM, 87–96.

Cx K Batchelor and GK Batchelor. 1967. *An introduction to fluid dynamics*. Cambridge university press.

Jan Bender and Dan Koschier. 2015. Divergence-free smoothed particle hydrodynamics. In *Proceedings of the 14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 147–155.

Jan Bender, Dan Koschier, Tassilo Kugelstadt, and Marcel Weiler. 2018. Turbulent Micropolar SPH Fluids with Foam. *IEEE transactions on visualization and computer graphics* (2018).

Mengyu Chu and Nils Thuerey. 2017. Data-driven synthesis of smoke flows with CNN-based feature descriptors. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 69.

Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. 2001. Visual simulation of smoke. (2001), 15–22.

de Goes Fernando, Wallez Corentin, and Huang Jin. 2015. Power particles:an incompressible fluid solver based on power diagrams. *ACM Transactions on Graphics* 34, 4 (2015), 1–11.

Markus Ihmsen, Jens Cornelis, Barbara Solenthaler, Christopher Horvath, and Matthias Teschner. 2014. Implicit incompressible SPH. *IEEE Transactions on Visualization and Computer Graphics* 20, 3 (2014), 426–435.

Theodore Kim, Jerry Tessendorf, and Nils Thuerey. 2013. Closest point turbulence for liquid surfaces. *ACM Transactions on Graphics (TOG)* 32, 2 (2013), 15.

Dan Koschier, Jan Bender, Barbara Solenthaler, and Matthias Teschner. 2019. Smoothed Particle Hydrodynamics Techniques for the Physics Based Simulation of Fluids and Solids. (2019).

Michael Lentine, Mridul Aanjaneya, and Ronald Fedkiw. 2011. Mass and momentum conservation for fluid simulation. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 91–100.

Matthias Müller, David Charypar, and Markus Gross. 2003. Particle-based fluid simulation for interactive applications. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Eurographics Association, 154–159.

Sang Il Park and Myoung Jun Kim. 2005. Vortex fluid for gaseous phenomena. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. ACM, 261–270.

Tobias Pfaff, Nils Thuerey, and Markus Gross. 2012. Lagrangian vortex sheets for animating fluids. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 112.

Jos. Stam. 1999. Stable fluids. *ACM Transactions on Graphics* (1999), 121–128.

John Steinhoff and David Underhill. 1994. Modification of the Euler equations for "vorticity confinement": Application to the computation of interacting vortex rings. *Physics of Fluids - PHYS FLUIDS* 6 (08 1994), 2738–2744. https://doi.org/10.1063/1.868164

Bo Zhu, Xubo Yang, and Ye Fan. 2010. Creating and preserving vortical details in sph fluid. In *Computer Graphics Forum*, Vol. 29. Wiley Online Library, 2207–2214.