

SPECIAL ISSUE PAPER

Real-time smoke simulation with improved turbulence by spatial adaptive vorticity confinement

Shengfeng He¹, Hon-Cheng Wong^{1,2*}, Wai-Man Pang³ and Un-Hong Wong²¹ Faculty of Information Technology, Macau University of Science and Technology, Macao, China² Space Science Institute, Macau University of Science and Technology, Macao, China³ Spatial Media Group, Computer Arts Laboratory, University of Aizu, Aizu-Wakamatsu, Japan

ABSTRACT

Turbulence modeling has recently drawn many attentions in fluid animation to generate small-scale rolling features. Being one of the widely adopted approaches, vorticity confinement method re-injects lost energy dissipation back to the flow. However, previous works suffer from deficiency when large vorticity coefficient ε is used, due to the fact that constant ε is applied all over the simulated domain. In this paper, we propose a novel approach to enhance the visual effect by employing an adaptive vorticity confinement which varies the strength with respect to the helicity instead of a user-defined constant. To further improve fine details in turbulent flows, we are not only applying our proposed vorticity confinement to low-resolution grid, but also on a finer grid to generate sub-grid level turbulence. Since the incompressible Navier–Stokes equations are solved only in low-resolution grid, this saves a significant amount of computation. Several experiments demonstrate that our method can produce realistic smoke animation with enhanced turbulence effects in real-time. Copyright © 2011 John Wiley & Sons, Ltd.

KEYWORDS

smoke animation; vorticity confinement; turbulence; fluid simulation

*Correspondence

Hon-Cheng Wong, Faculty of Information Technology, Macau University of Science and Technology, Macao, China.

E-mail: hcwong@ieee.org

1. INTRODUCTION

Realistic visual simulation of natural phenomena such as smoke, water, and fire is vital and commonly required in movies and games nowadays to enhance the visual enjoyment. In this paper, we are focusing on smoke simulation which usually exhibits significant turbulent flows. A common problem in capturing the visual details of turbulence is the requirement of a high-resolution grid, and this also means we need more time and memory for computation. In recent years, a number of works [1–3] were proposed to improve details by adding noise to those voxels that suffered from numerical dissipation. But these models do not account for the underlying dynamics of turbulence and they introduce a certain extent of computational complexity.

Fedkiw *et al.* [4] proposed a turbulence modeling method called vorticity confinement. Their method is simple and fast. However, in their vorticity confinement formulation, a simplified model is used in which the entire

grid utilizes the same coefficient to control the strength of augmented forces. This becomes a problem for cases with extreme helicity in grids. If the simulation grid is low in resolution, a moderate strength of vorticity confinement may not amplify the small-scale vortices significant enough. However, if we set the coefficient too large, it may cause artifacts and instabilities as demonstrated in Selle *et al.* [5].

As a result, we proposed a novel approach using an improved vorticity confinement which varies the strength spatially according to the helicity in the space instead of an arbitrary constant. Our approach starts the simulation at a low-resolution grid by solving Navier–Stokes (N–S) equations with the improved vorticity confinement as external force term. Then, we refine the grid size by up-sampling the velocity field with cubic B-spline interpolation instead of linear interpolation. Thus, we avoided the computational cost of handling high-resolution grid but still achieved a natural fluid flow of lower speed. Finally, we generate small-scale turbulence at this up-sampled grid

with improved vorticity confinement again, so that a high-resolution smoke animation with turbulence details can be obtained efficiently. In summary, our work has the following characteristics:

- An improved vorticity confinement method is introduced to enhance reality of turbulence effects in smoke. Our formulation takes helicity into account to vary the strength of confinement spatially, in contrast to a single coefficient for the whole space.
- Multi-scale scheme is used to generate a high-resolution flow from low-resolution simulation with cubic B-spline interpolation.
- A simple, fast, and high quality procedural turbulence model in real-time.

The rest of the paper is organized as follows: a brief overview of related work is presented in Section 2. The proposed vorticity confinement method is detailed in Section 3. The multi-scale scheme used is explained in Section 4. Experimental results are presented in Section 5. We conclude our work and give our future work in Section 6.

2. RELATED WORK

Numerous studies have been proposed since the seminal work on fluid simulation for computer graphics was carried out by Stam [6], who introduced an unconditionally stable semi-Lagrangian advection method. Considering there is a large number of papers on fluid simulation for computer graphics, only recent studies closely related to our work are reviewed here. We refer the reader to a recent book by Bridson [7] for a more comprehensive review.

The three-dimensional (3D) N–S equations were first used by Foster and Metaxas [8] for smoke simulation. Fedkiw *et al.* [4] modified the vorticity confinement method [9] to amplify small-scale vortices lost due to the numerical dissipation. Although these grid-based techniques [4,6,8] can provide interesting visual results, their development is prohibited by the amount computational power available. As a consequence, techniques [1–3] based on procedural synthesis were developed to add details to these simulations with noise.

Moreover, methods for generating a higher resolution result from a lower resolution simulation are getting considerable concern. Nielsen *et al.* [10] developed a method by using low-resolution input simulations to guide higher-resolution ones where details are added. Yoon *et al.* [11] utilized the vortex particle method [5] to generate the vorticity field, which are then procedurally synthesized with the flow fields obtained from the incompressible N–S solver to improve sub-grid visual details. Lentine *et al.* [12] presented a speedup technique for simulating detailed fluids by generating a divergence-free velocity field on a coarse grid. Zhao *et al.* [13] proposed a scheme utilizing random forcing in turbulence integration to enhance an

existing fluid simulation with controllable turbulence. Although the above mentioned methods achieved visually compelling results while keeping the performance, the dynamical nature of turbulent flows such as helicity [14,15], which is an important quantity in the physics of coherent structure and turbulence [16], was not considered. In this paper, we will propose a new algorithm considering helicity in the simulation process.

Our schematic pipeline shown in Figure 2 is similar to Yoon *et al.*'s [11] method in spirit, but we employ the proposed adaptive vorticity confinement instead of the vortex particle method [5] to generate the vorticity field and use the cubic B-spline interpolation [17] to up-sample the grid. Then the small-scale turbulence is generated with the proposed confinement from this up-sampled grid.

3. SPATIAL ADAPTIVE VORTICITY CONFINEMENT

In this section, we briefly review the incompressible N–S equations and the vorticity confinement first. Then, we introduce a spatial adaptive vorticity confinement method for incompressible flows.

4. INCOMPRESSIBLE NAVIER–STOKES EQUATIONS

For the sake of efficiency, fluids are commonly assumed to be inviscid and incompressible in computer animation, so most of the simulations are based on the incompressible N–S equations [7]. N–S equations describe the motion of fluid flow through the change of velocity \mathbf{u} over position and time step t .

$$\frac{\partial \mathbf{u}}{\partial t} = -\mathbf{u} \cdot \nabla \mathbf{u} - \frac{1}{\rho} \nabla p + \mathbf{f} \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (2)$$

where \mathbf{u} denotes the velocity and p is the pressure, ρ is the mass density, and \mathbf{f} represents the external forces such as gravity or user input forces. The numerical methods for solving the incompressible N–S equations can be found in Ref. [7].

5. VORTICITY CONFINEMENT

Fedkiw *et al.* [4] improved the vorticity confinement [9], by introducing the mesh size h , to guarantee the computational consistency with mesh refinement. Their method successfully recovered the small-scale rolling details of fluids on a coarse grid. The vorticity confinement is considered as the external force that adds to the velocity in Eq. (1) in every time step. For incompressible flows, the vorticity $\boldsymbol{\omega}$ can be obtained by calculating the curl of

velocity field \mathbf{u} :

$$\boldsymbol{\omega} = \nabla \times \mathbf{u} \quad (3)$$

The normalized vorticity location vector \mathbf{N} can be obtained by normalizing the gradient of $|\boldsymbol{\omega}|$:

$$\mathbf{N} = \frac{\nabla |\boldsymbol{\omega}|}{|\nabla |\boldsymbol{\omega}||} \quad (4)$$

The confinement force is computed as follows [4]:

$$\mathbf{f}_{\text{conf}} = \varepsilon h (\mathbf{N} \times \boldsymbol{\omega}) \quad (5)$$

where ε is the coefficient that controls the strength of the confinement by the user, and h is the mesh size.

6. SPATIAL ADAPTIVE VORTICITY CONFINEMENT FOR INCOMPRESSIBLE FLOW

As we mentioned in Section 1, large ε may cause artifacts and instabilities [5]. This is due to the fact that the important description of turbulence—helicity, was not considered in the simulation process.

The helicity is defined as [14,15]:

$$H = \int \mathbf{u} \cdot \boldsymbol{\omega} dV \quad (6)$$

It is a conserved quantity and unchanged in incompressible flows. It measures the amount of rotation of a fluid rotating around an axis that is parallel to the main flow direction. Helicity density is defined as the dot product of the velocity vector and the vorticity vector. Both helicity and helicity density are pseudoscalar quantities which admit different signs showing the direction of rotation of the fluid (either clockwise or counter-clockwise). By taking account the helicity, we expect to handle the flow rolling details better with a spatially varying confinement coefficient.

Inspired by Robinson's [18] work, which carried out dimensional analysis to get a modified vorticity confinement with the strength that varies with helicity for compressible flows. We adopted his method to develop an adaptive vorticity confinement for incompressible flows.

We factor $|\boldsymbol{\omega}|$ out from Eq. (5) and replace ε with $\varepsilon|\mathbf{u}|$:

$$\mathbf{f}_{\text{conf}} = \varepsilon h |\mathbf{u}| |\boldsymbol{\omega}| \left(\mathbf{N} \times \frac{\boldsymbol{\omega}}{|\boldsymbol{\omega}|} \right) \quad (7)$$

Since the unit of $|\mathbf{u}||\boldsymbol{\omega}|$ has the dimension of a helicity, after performing dimensionalization, we can arrange Eq. (7) to make the confinement proportional to the helicity:

$$\mathbf{f}_{\text{conf}} = \varepsilon_h h |\mathbf{u} \cdot \boldsymbol{\omega}| \left(\mathbf{N} \times \frac{\boldsymbol{\omega}}{|\boldsymbol{\omega}|} \right) \quad (8)$$

where ε_h is a true dimensionless parameter. Now the strength of the vorticity confinement depends not only on the mesh size h , but also on helicity. As a result, the effect of the rotation of a fluid is included when generating small rolling details. We demonstrate the effect of this adaptive vorticity confinement in Figure 4, from which we can find that the vorticity field generated with the vorticity confinement in Ref. [4] has a wider spreading while the one generated with the adaptive vorticity confinement gives a more realistic pattern.

7. MULTI-SCALE FLUID SYNTHESIS

A multi-scale scheme is used to synthesize a high-resolution fluid to ensure the quality of the visual effect while keeping the performance. This is achieved by solving the N-S equations in a low-resolution grid and enhance the sub-grid details by blending the high-resolution vorticity field obtained from the up-sampled velocity. The proposed adaptive vorticity confinement is applied when getting the velocity and vorticity fields.

8. UP-SAMPLING BY CUBIC B-SPLINE INTERPOLATION

To achieve a large-scale fluid flow efficiently, the velocity field is obtained by solving the N-S equations in a low-

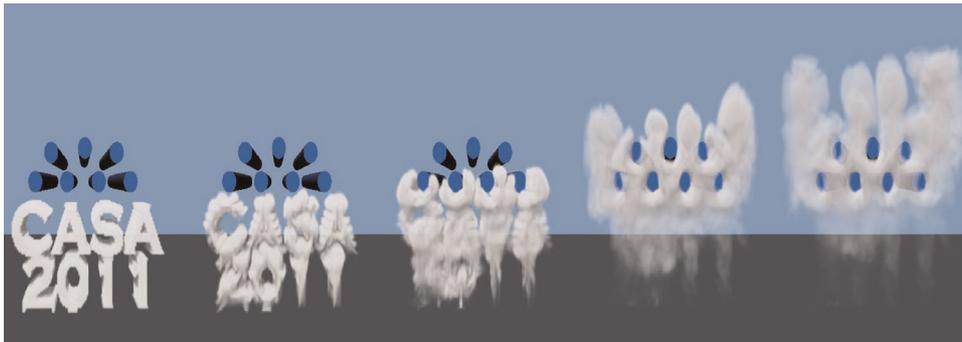


Figure 1. Rising smoke passing through cylinders: simulation sequence of a $128 \times 128 \times 128$ grid up-sampled from a $64 \times 64 \times 64$ coarse grid using the adaptive vorticity confinement proposed in this paper. The smoke was rendered by ray-marching at 39.3 frames per second.

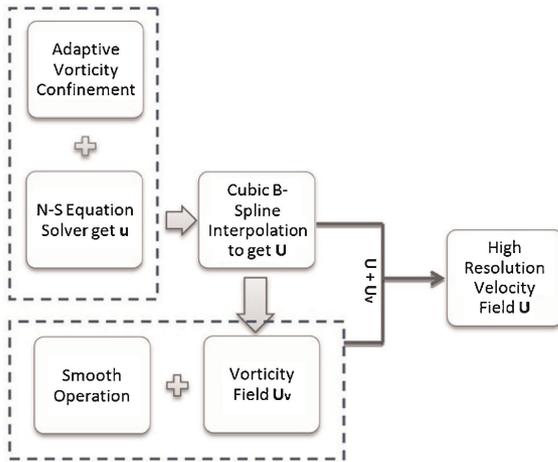


Figure 2. The schematic pipeline of our approach.

resolution grid. And then the low-resolution velocity field is up-sampled with a high-resolution grid. In this process, a simple interpolation method such as trilinear interpolation is not accurate enough to provide satisfactory visual results. For example, when we amplified the velocity field more than two times, the appearance of the up-sampled fluid became blurry (see Figure 5). As a result, we used the cubic B-spline interpolation [17] for the up-sampling process. Cubic B-spline interpolation is considered as one of the simplest interpolation methods to achieve true continuity values from the data points.

9. VORTICITY FIELD GENERATION

In practice we found that only up-sampled the low-resolution velocity field with the adaptive vorticity confinement still resulted in blurry and unrealistic smoke turbulence. Our method, therefore, starts with a low resolution simulation using the MacCormack approach [19] to solve the dynamics of laminar flow, then we apply the adaptive vorticity confinement the first time on this low resolution grid to re-inject turbulence into the main flow. Based on this, we up-sample the velocity field to increase its resolution and apply once again the adaptive vorticity confinement to extensively improve small-scale details. Notice that our approach will not down-sample the high-resolution flow back to low-resolution one. Therefore, this two-scale approach helps to minimize the computational cost while at the same time increases the visual quality of small-scale turbulence. In addition, blending the vorticity field directly to the up-sampled velocity field is also found to cause significant noise which is undesirable. Thus, in step 6 of the following list, a smooth procedure is introduced to average the vorticity magnitude for every grid by its neighbors to avoid this problem. The following lists the major steps of our approach to simulate and visualize the smoke dynamics:

- 1: Advection of \mathbf{u} using MacCormack method
- 2: Apply the adaptive vorticity confinement to \mathbf{u}
- 3: Pressure projection

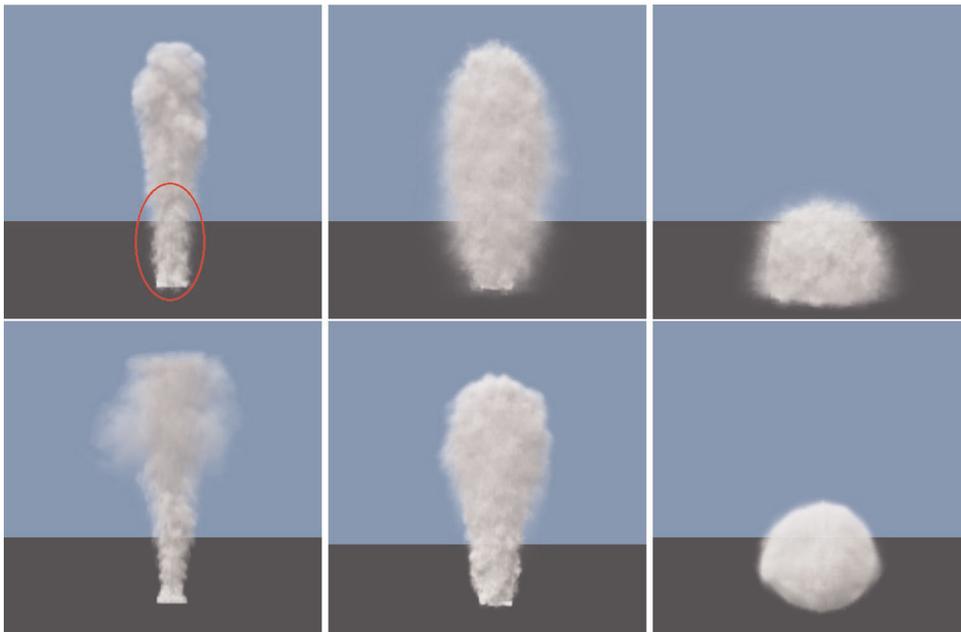


Figure 3. A comparison of (top) the vorticity confinement in Ref. [4] and (bottom) the adaptive vorticity confinement proposed in this paper. The figures at the top, from left to right were created with $\varepsilon = 0.4$, $\varepsilon = 0.7$, both with buoyancy force, and $\varepsilon = 0.7$ without buoyancy force. The figures at the bottom, from left to right were created with $\varepsilon_h = 0.04$, $\varepsilon_h = 0.07$, both with buoyancy force, and $\varepsilon_h = 0.07$ without buoyancy force.

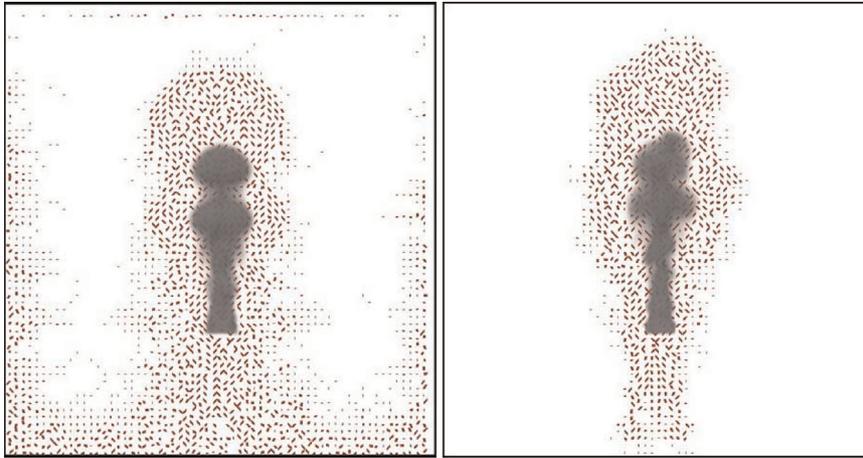


Figure 4. A quantitative comparison of the 2D vorticity fields of (left) the vorticity confinement in Ref. [4] and (right) the adaptive vorticity confinement proposed in this paper. Reddish brown lines indicate direction and their length representing magnitude.

- 4: Up-sample \mathbf{u} to \mathbf{U} by the cubic B-spline interpolation
- 5: Generate $\mathbf{U}_{\text{vorticity}}$ by the adaptive vorticity confinement
- 6: Smoothing operation of $\mathbf{U}_{\text{vorticity}}$
- 7: $\mathbf{U} = \mathbf{U} + \mathbf{U}_{\text{vorticity}}$
- 8: Advection of \mathbf{D} using MacCormack method
- 9: Render the density data \mathbf{D}
- 10: Go to step 1

Here, \mathbf{u} and \mathbf{U} are the velocity field in low and high resolution, respectively. $\mathbf{U}_{\text{vorticity}}$ is the vorticity field generated from \mathbf{U} , while \mathbf{D} represents the density field of the smoke.

10. EXPERIMENTAL RESULTS

Experiments were performed on a PC with Intel Core i7 CPU, 4GB RAM, and NVIDIA GeForce GTX 480 graphics card. Our simulation was implemented with CUDA working on GPU entirely. We have used the MacCormack method [19] for advection term, Jacobi iteration [20] for solving the linear system, and ray-marching [21] for rendering. Figure 1 shows the simulation sequence of a rising smoke passing through cylinders. The words ‘CASA 2011’ were composed by smoke which rises due to the buoyancy force and hits the cylinders. This simulation was executed in real-time. Figure 3 demonstrates the improvement of visual effects by our

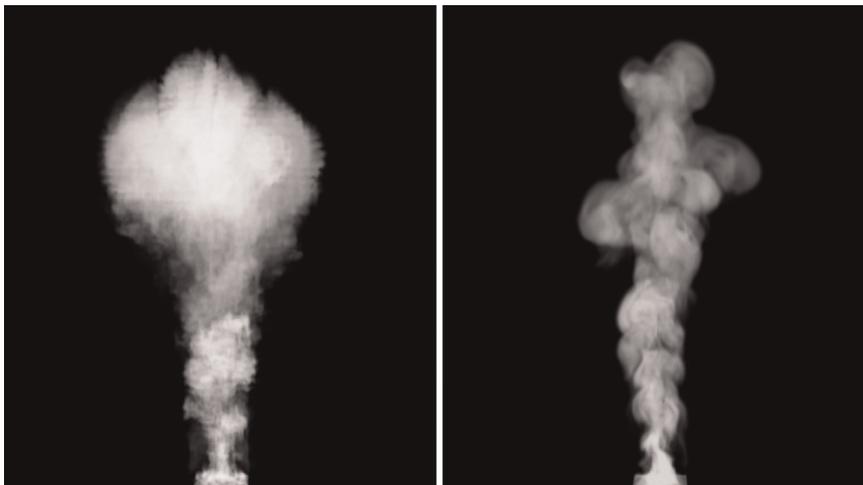


Figure 5. A comparison of (left) linear interpolation and (right) cubic B-spline interpolation. This example is a $256 \times 256 \times 256$ simulation with a $64 \times 64 \times 64$ coarse grid rendered by ray-casting.

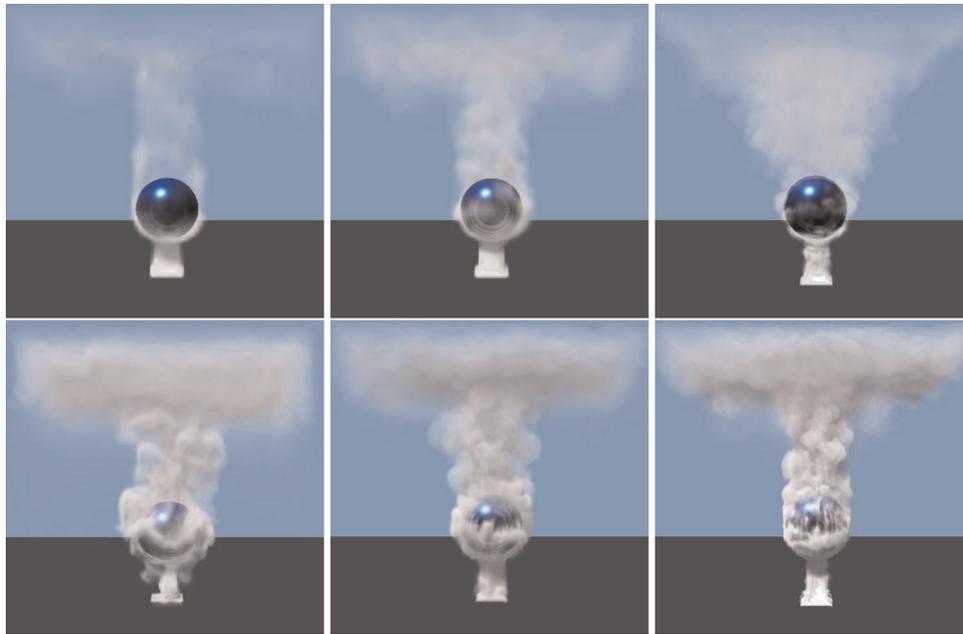


Figure 6. A comparison of (top) the simulation without multi-scale method and (bottom) with multi-scale method, using different grid resolutions. The figures at the top, from left to right: a $64 \times 64 \times 64$ base simulation without using vorticity confinement, a $64 \times 64 \times 64$ simulation, and a $128 \times 128 \times 128$ simulation, both using the adaptive vorticity confinement. The figures at the bottom, from left to right: a $128 \times 128 \times 128$ simulation with a $32 \times 32 \times 32$ coarse grid, a $128 \times 128 \times 128$ simulation with a $64 \times 64 \times 64$ coarse grid, and a $256 \times 256 \times 256$ simulation with a $128 \times 128 \times 128$ coarse grid.

method. Though the visual effects do not have large difference at small ε , when ε increases, the smoke using the vorticity confinement in Ref. [4] becomes falling apart as indicated at the red ellipse. The adaptive vorticity confinement can fix this problem quite well. In an extreme case, the figures at the last column show the difference of these two methods clearly. These two simulations are without buoyancy force but upward force at the smoke generation position. The simulation with the vorticity confinement spreads the smoke far apart, and the simulation with the adaptive vorticity confinement makes the smoke stay together. Figure 6 compares the results with

and without multi-scale method. Simulations without using the multi-scale method are not able to represent the chaotic effect of smoke when flowing around a sphere. In contrast, the smoke looks more chaotic and turbulent in the multi-scale case.

Our method can simulate fine smoke phenomena in real-time. Table 1 provides the performance information. Targeting on the same grid resolution, for example, a $128 \times 128 \times 128$ simulation, our method achieved around two times speedups than the one with the vorticity confinement in Ref. [4] and even provided better visual effects. The simulations using different grid resolutions are shown in Figure 6.

Table 1. A performance comparison of the vorticity confinement in Ref. [4] and the adaptive vorticity confinement using different grid resolutions.

Method	Coarse resolution	Target resolution	Time step (minutes)	Time (frames per second) (simulation)	Time (frames per second) (including rendering)
VC in Ref. [4]	—	$128 \times 128 \times 128$	40.1	24.9	20.7
Adaptive VC	$32 \times 32 \times 32$	$128 \times 128 \times 128$	15.6	63.9	43.4
Adaptive VC	$64 \times 64 \times 64$	$128 \times 128 \times 128$	18.5	54.0	39.3
VC in Ref. [4]	—	$256 \times 256 \times 256$	301.3	3.32	3.0
Adaptive VC	$64 \times 64 \times 64$	$256 \times 256 \times 256$	119.5	8.4	7.6
Adaptive VC	$128 \times 128 \times 128$	$256 \times 256 \times 256$	142.7	7.01	6.6

11. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a novel approach to produce highly turbulent flows in real-time smoke simulation using an adaptive vorticity confinement method considering the helicity. The strength of the confinement varies with respect to the helicity. Experimental results show that after up-sampling the velocity field from a low-resolution grid by cubic B-spline interpolation, we are able to achieve a high quality smoke by blending the vorticity field obtained by the adaptive vorticity confinement from a high-resolution grid. Compared with others noise based turbulence models [1–3], our method is fast, straightforward and easy to implement. Moreover, we can get highly turbulent visual effects. We can simulate smoke phenomena in a finer-resolution grid at interactive frame rates.

In our future work, we will try to extend our confinement approach to dynamic vorticity confinement where the strength of the confinement term is computed without any need for an arbitrary constant. We are also interested in improving performance of our simulation. Applying our model to a real interactive application is also one of our goals.

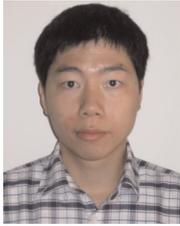
ACKNOWLEDGEMENTS

This work is supported by the Science and Technology Development Fund of Macao SAR (03/2008/A1) and the National High-Technology Research and Development Program of China (2010AA122205). Special thanks to anonymous reviewers for their constructive comments on the paper.

REFERENCES

- Schechter H, Bridson R. Evolving sub-grid turbulence for smoke animation. In *Proc. of the 2008 Eurographics/ACM SIGGRAPH Symp. on Comput. Anim.*, 2008, pages 1–8.
- Kim T, Thürey N, James D, Gross M. Wavelet turbulence for fluid simulation. *ACM Transactions on Graphics (SIGGRAPH Proceedings)* 2008; **27**: 50.
- Narain R, Sewall J, Carlson M, Lin MC. Fast animation of turbulence using energy transport and procedural synthesis. *ACM Transactions on Graphics (SIGGRAPH Asia Proceedings)* 2008; **27**: 166.
- Fedkiw R, Stam J, Jensen HW. Visual simulation of smoke. In *Proc. of ACM SIGGRAPH 2001*, 2001, pages 15–22.
- Selle A, Rasmussen R, Fedkiw N. A vortex particle method for smoke, water and explosions. *ACM Transactions on Graphics (SIGGRAPH Proceedings)* 2005; **24**: 910–914.
- Stam J. Stable fluids. In *Proc. of ACM SIGGRAPH 1999*, 1999, pages 121–128.
- Bridson R. *Fluid Simulation for Computer Graphics*, A K Peters: Wellesley, Massachusetts, 2008.
- Foster N, Metaxas D. Modeling the motion of a hot, turbulent gas. In *Proc. of ACM SIGGRAPH 1997*, 1997, pages 181–188.
- Steinhoff J, Underhill D. Modification of the euler equations for “vorticity confinement”: application to the computation of interacting vortex rings. *Physics of Fluids* 1994; **6**: 42738–42744.
- Nielsen NB, Christensen BB, Zafar NB, Roble D, Museth K. Guiding of smoke animations through variational coupling of simulations at different resolution. In *Proc. of the 2009 Eurographics/ACM SIGGRAPH Symp. on Comput. Anim.*, 2009, pages 217–226.
- Yoon J-C, Kam HR, Hong J-M, Kang SJ, Kim C-H. Procedural synthesis using vortex particle method for fluid simulation. *Computer Graphics Forum (Pacific Graphics Proceedings)* 2009; **28**: 1853–1859.
- Lentine M, Zheng W, Fedkiw R. A novel algorithm for incompressible flow using only a coarse grid projection. *ACM Transactions on Graphics (SIGGRAPH Proceedings)* 2010; **29**: 114.
- Zhao Y, Yuan Z, Chen F. Enhancing fluid animation with adaptive, controllable and intermittent turbulence. In *Proc. of the 2010 ACM SIGGRAPH/Eurographic Symp. of Comput. Anim.*, 2010, pages 75–84.
- Moffatt HK. The degree of knottedness of tangled vortex lines. *Journal of Fluid Mechanics* 1969; **36**: 117–129.
- Moffatt HK, Tsinober A. Helicity in laminar and turbulent flow. *Annual Review of Fluid Mechanics* 1992; **24**: 281–312.
- Moffatt HK. Some developments in the theory of turbulence. *Journal of Fluid Mechanics* 1981; **106**: 27–47.
- Sigg C, Hadwiger M. Fast third-order texture filtering. In *GPU Gems 2*, 2005, pages 313–329.
- Robinson M. Application of vorticity confinement to inviscid missile force and moment prediction. In *Proc. of AIAA 42nd Aerospace Sciences Meeting & Exhibit*, AIAA Paper 2004-0717, 2004.
- Selle A, Fedkiw R, Kim B, Liu L, Rossignac J. An unconditionally stable macCormack method. *Journal of Scientific Computing* 2008; **35**: 350–371.
- Harris M. Fast fluid dynamics simulation on the gpu. In *GPU Gems*, 2004, pages 637–665.
- Crane K, Tariq S, Llamas I. Real time simulation and rendering of 3d fluids. In *GPU Gems 3*, 2008, pages 633–675.

AUTHORS' BIOGRAPHIES



Shengfeng He received his BSc degree from the Faculty of Information Technology, Macau University of Science and Technology (MUST), Macao, China in 2009. He is currently a MSc candidate in the same university. His research interests include physically based computer animation, GPU techniques, and volume visualization.



tion, and GPU computing.

Hon-Cheng Wong received his PhD degree from the Faculty of Information Technology, Macau University of Science and Technology (MUST), Macao, China in 2005. He is currently the program coordinator and associate professor in the same university. His research interests include computer graphics, scientific visualization, and GPU computing.



Wai-Man Pang is currently an assistant professor in the Computer Arts Lab., University of Aizu, Japan. He finished his postdoctoral fellowship and PhD study related to Computer Graphics, Vision, Imaging, and Game related technologies in summer 2008 at the Department of Computer Science and Engineering at the Chinese University of Hong Kong, Hong Kong, China. His research interests include non-photorealistic rendering, image-based rendering, GPU programming, and physically based deformation. Pang is the author of several significant journals or books including ACM Transactions on Graphics, IEEE TVCG, IEEE CG&A, and ShaderX5.



Un-Hong Wong received his BSc and MSc degrees from the Faculty of Information Technology, Macau University of Science and Technology (MUST), Macao, China in 2004 and 2007, respectively. He is now a research assistant in the same university. His research interests include computer graphics, visualization, and physics simulations using GPU techniques.