Procedural Synthesis using Vortex Particle Method for Fluid Simulation

Jong-Chul Yoon\textsuperscript{1}, Hyeong Ryeol Kam\textsuperscript{2}, Jeong-Mo Hong\textsuperscript{3}, Shin Jin Kang\textsuperscript{4} and Chang-Hun Kim\textsuperscript{2}

\textsuperscript{1} FXGear Inc. \textsuperscript{2} Korea University \textsuperscript{3} Dongguk University \textsuperscript{4} Hongik University

Abstract
We propose a fast and effective technique to improve sub-grid visual details of the grid based fluid simulation. Our method procedurally synthesizes the flow fields coming from the incompressible Navier-Stokes solver and the vorticity fields generated by vortex particle method for sub-grid turbulence. We are able to efficiently animate smoke which is highly turbulent and swirling with small scale details. Since this technique does not solve the linear system in high-resolution grids, it can perform fluid simulation more rapidly. We can easily estimate the influence of turbulent and swirling effect to the fluid flow. 

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction
As the use of computer graphics is increasing in movies, commercials and other media, the work of animating fluids such as explosion and fire has become an important process in that area.

Recently fluid dynamics technologies are applied to a range of special effects, yet achieving precise calculations is still a time-consuming process, which is why various technologies are being studied for better balance between productivity and the reality of graphics. Therefore, in the area of graphics, enhancing productivity while maintaining an adequate level of visual quality - even at the modest expense of physical/mathematical accuracy - presents a challenge to fluid animation technologies.

While the vortex particle method \cite{SRF05} is useful for generating turbulent effects, its drawback is that the grid is too coarse to accommodate the vorticity of all particles. Although it is possible to use high-resolution grids in simulation, it would require much more time and memory capacity. Recently introduced techniques \cite{KTJG08,SB08,NSCL08} have improved details in grids by using noise.

We propose a new technique for improving details in grids in fluid simulation. This technique depicts the uncomplicated motion of fluid through flow fields calculated by solving the incompressible Navier-Stokes equations, and detailed motion of fluid through vorticity field calculated by the vortex particle method. This technique does not solve the linear system in high-resolution grids, thus it performs large-scale grid simulation efficiently. Another benefit is that it generates turbulent effects of explosion or rough motion of smoke as the vorticity of vortex particles is transferred to the high-resolution grid.

Our approach has the following characteristics:

- A degree of resolution equivalent to the result of simulation using the \((k \times n)^3\) grid is achieved by solving fluid equations of the \(n^3\) grid in a three-dimensional space. \(k\) is a constant making desired resolution.
- Since this method has the benefit that the result can be quickly drawn through low-resolution simulation, if the result is satisfactory, a high-resolution result can be obtained without running simulation with adjusted parameters.
- Highly turbulent and swirling effects that cannot be generated easily by using noise for simulating explosion, smoke, etc. can be achieved since it is hard to anticipate how they will affect flow when using noise.
- Vortex particle carries the vorticity calculated from low resolution and we create the high resolution vorticity field using the vorticity. This field uses the information from the base simulation.
• We can easily control the size of turbulence depending on the frequency by adjusting the particle radius.

2. Previous Work

Stam and Fiume [SF93] proposed a method to depict gaseous phenomena through the ambient turbulence that is used to add new eddies by noise. Smoke simulation using the three dimensional Navier-Stokes equations began with Foster and Metaxas [FM97]. Stam [Sta99] presented a Semi-Lagrangian advection model which allows stable simulation with large time step. Fedkiw et al. [FSJ01] introduced a simulation model for generating a vorticity effect which is hard to depict in coarse grids. This vorticity confinement method reduced the loss of small-scale details and increased the amount of swirling motion. Felici and Drela [FD90, FD93a, FD93b] coupled an Eulerian and a Lagrangian Solver and tried to reduce numerical diffusion.

Selle et al. [SRF05] presented vortex particle method which allowed effective simulation by incorporating localized vorticity confinement in grid. It directly computes the velocity for particles by trilinear interpolation. It reduces the numerical loss because each vortex particle stores a vorticity value. But it is slow for high-resolution simulation. And Park and Kim [PK05] presented a Lagrangian method for gaseous phenomena simulation based on the vortex method. Hong and Kim [HK05] proposed the method for vortex advection based on vorticity confinement. Angelidis et al. [ANSN06] used the vortex filament methods based on the vorticity formulation of the Navier-Stokes equations. They defined a vorticity preserving flow field around a set of vortex primitives. Previous research results were only useful in small-scale simulation due to the high computational cost. Although the use of fluid technologies is increasing, the improvement of computing environment still falls short of the demand of developers, which has led to a number of attempts to find ways to enhance the quality as well as reality of simulation. Neyret [Ney03] increased the details of fluid by advecting texture to the velocity field. Rasmussen et al. [RNGF03] proposed an efficient method for large-scale fluid simulation, and Losasso et al. [LGF04] introduced a method to reduce simulation time by limiting the refinement to the fluid surface part of the grid using octree data structure instead of using uniform grid. Another approach is to employ error correction schemes such as BFECC [KLLR07] or a MacCormack method [SFK*08], or less dissipative advection schemes such as USCIP [KSK08], to reduce the diffusion in the numerical method directly.

Recently, methods for adding further details in sub-grid by using noise have been introduced. These methods are similar to our method’s schematic outline, adding the small-scaled detail to the upsampled field gained by interpolation in coarse-grid simulation for making a new velocity field. Kim et al. [KTJG08] used wavelet method and synthesized missing turbulent flow components with band-limited wavelet noise. This helps to show small-scaled detail in a coarse grid. Schechter and Bridson [SB08] presented sub-grid turbulence evolution model for fluid simulation. They tracked bands of turbulent energy using a simple linear model and created the turbulent velocity using flow noise. They added a predictor step to the usual time splitting of the incompressible Euler equation and corrected the additional vorticity dissipation due to time splitting of the pressure and
advection. Narain et al. [NSCL08] introduced a technique coupling a procedural turbulence model with a numerical fluid solver. They used an energy function and a Lagrangian approach to advect noise when synthesizing an incompressible turbulent velocity field. They applied the method to simulation of liquids with free surfaces.

Turbulence can be edited easily using these methods because the turbulence is independent from the large-scale flow and able to be added like a post process. But we don’t include a noise and energy function but vortex particle method, i.e. vortex particle carries the vorticity calculated from low resolution and we create the high resolution vorticity field using the vorticity. This field uses the information from the base simulation while the noise addition is unrelated to the base simulation. In addition, our method is easy to implement and able to show highly turbulent effects such as an explosion or rocket smoke that is hard to be generated from a noise function.

3. High resolution fluid synthesis

This section discusses the high-resolution method of synthesizing vortex particles in fluid simulation. The notation used is as follows. Bold denotes a vector, and non-bold represents a scalar. Parameters used in low-resolution simulation are marked with lower-case letters and parameters used in high-resolution simulation are marked with upper-case letters, while \( x \) represents the location. \( f(u, X) \) is the function for interpolation of velocity field \( u \) by high-resolution location \( X \). \( A(U, D) \) is the function for advection of density \( D \) by \( U \).

3.1. Base fluid simulation

We simulate the motion of fluid by solving the incompressible Navier-Stokes equations commonly used to animate fluid in graphics. The Navier-Stokes equations describing inviscid incompressible fluid motion are

\[
\begin{align*}
\mathbf{u}_t + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p/\rho &= f \\
\nabla \cdot \mathbf{u} &= 0,
\end{align*}
\]

where \( \mathbf{u} = (u, v, w) \) is the fluid’s velocity, \( p \) is the pressure, \( \rho \) is the density, and \( f \) is the external forces such as gravity and vorticity confinement. Since upsampled velocity field depends on that of base simulation, using a vorticity confinement results in more dynamic flow than not using it. Since numerical methods of solving equations (1) and (2) are well known, we refer readers to [Sta99, FSJ01] for details.

3.2. Saving grid vorticity to vortex particle

We use vortex particle method [SRF05] in this paper. The Navier-Stokes equations can be put into vorticity form by
3.3. Generating vorticity field

The vorticity of vortex particles obtained in 3.2 is transferred to high-resolution vorticity field $U_{\text{vorticity}}$, which is different from Selle et al. [SRF05]. We get the vorticity of a particle, $\omega^*_p(X) = \xi_p(X-X_p)\omega_p$ by using the distribution kernel, $\xi_p(X-X_p)$ which is a function of the distance between the particle position and the high-resolution grid point. Then we calculate the sum of all contributions from all particles by the normalized location vector from grid point to vortex particle, $N_p(X) = X_p - X / ||X_p-X||$ and the confinement force, $F_p(X) = \epsilon_p(N_p \times \omega^*_p)$ to update the vorticity field element which is in the influence of the particles.

**Discussion:** The vorticity field generated by above method is not divergence-free. To make vorticity field divergence-free, we can calculate curl of particle’s vorticity taken by a ramp function instead of using $(N_p \times \omega^*_p)$. Our result came from the method of 3.3. The non divergence-free status in vorticity field at sub-grid level is ignorable since the vorticity field is newly created at each step instead of being advected. Our result shows little difference as compared with divergence-free case and it operates a little faster.

3.4. Blending vorticity field with upsampled velocity field

We create a new field $U_{\text{advection}}$ for the advection of density $D$, as follows:

$$U_{\text{advection}}(X,t) = kU_{\text{vorticity}}(X,t) + U_{\text{interpolation}}(X,t)$$

where $k$ is a constant that controls influence of the vorticity field and $U_{\text{interpolation}} = I(u,X)$. We use $k = 1.0$ in our experiment and simple linear interpolation for $U_{\text{interpolation}} = I(u,X)$.

---

**Figure 3:** 2D simulation comparison: The low resolution simulation $(30 \times 90)$ is shown in the left. An effective resolution of $480 \times 1440$ using our method is shown in the right.
3.5. Pseudo-code

1. Advection\( (u) \)
2. Vorticity Confinement\( (u) \)
3. Projection\( (u) \)
4. \( U_{\text{interpolation}} = I(u, X) \)
5. \( U_{\text{vorticity}} = \text{Vortex Particle Method}(u) \)
6. \( U_{\text{advection}} = kU_{\text{vorticity}} + U_{\text{interpolation}} \)
7. \( A(U_{\text{advection}}, D) \)

4. Examples

We used the following kernel function [MCG03] to generate vorticity field from vortex particles.

\[
\xi_p(x - x_p) = \begin{cases} 
  (r^2 - (x - x_p)^2)^3 & 0 \leq (x - x_p) \leq r \\
  0 & \text{otherwise}
\end{cases}
\]

where \( r \) is the radius of the particles, determines the influence range when the force of the vortex particle is transferred to \( U_{\text{vorticity}} \). We defined a particle radius that is long enough to cover about two to three \( U \) cells.

Figure 1 shows smoke rising from a spherical density source due to buoyancy forces. In this example, vortex particles are randomly seeded where the density is sourced. Vortex particle method(approximately 300 vortex particles for low-resolution field and 20,000 vortex particles for high-resolution field, respectively) and Vorticity Confinement were used for base simulation and the BFECC advection method was used for density and velocity advection.

Our method is prominently faster than an equivalent full-resolution simulation and performs around seven times faster than the full solver as shown in table 1.

All the experiments were performed on an Intel Quad Core CPU 2.4GHz processor with 2GB RAM.

<table>
<thead>
<tr>
<th></th>
<th>Our method</th>
<th>Traditional NS solver</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution</td>
<td>120 × 360 × 120</td>
<td>120 × 360 × 120</td>
</tr>
<tr>
<td>Time per frame</td>
<td>9.2 sec</td>
<td>59.0 sec</td>
</tr>
</tbody>
</table>

Table 1: Comparison of computation time: Performance comparison of our method with the traditional Navier-Stokes Solver needed to generate the same visual detail. Our method used the result of base simulation(30 × 90 × 30, 0.58 seconds per frame).

**Limitations:** Our method has following limitations. We calculate the sum of all contributions from every vortex particle when generating vorticity field. So vorticity force is added to velocity field in proportion to the number of vortex particles gathering together in the particle radius. And then strong vorticity force can cause unnaturalness in fluid’s motion. Our method cannot reproduce physically correct high-resolution simulation. And boundary condition depends on
low resolution. So, there could be a velocity going into a obstacle at sub-grid level.

5. Conclusions and Future Work

This paper proposed a synthesizing method using vortex particle method to improve sub-grid visuals of fluid simulation. We modified the technique to apply the vortex particle method instead of using a noise function. We created flow field by solving the incompressible Navier-Stokes equations. It is upsampled through linear interpolation and blended with vorticity field which is calculated by vortex particle method. High-resolution results were achieved by advecting the density with using the calculated velocity field, and made it possible to simulate highly-turbulent swirling motion of fluid.

Comparing with previous noise based work [KTJG08, NSCL08], we use vortex particle method. Our work can show more sub-grid details with high turbulences and swirls. And the generation of sub-grid details using our approach can be expected more easily than noise based work. Overall process of ours and theirs is similar but we have different approach in generating turbulence, so the computational time of our work is little different from that of their work. And this also depends on the number of vortex particles and the radius of the particles.

In the future, we will try to apply our method to animation of liquids with free surfaces and make proper boundary conditions at the sub-grid level for more precise simulation.

Acknowledgements

This research is supported by Ministry of Culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology (CT) Research & Development Program 2009. This work was supported by the Korea Science and Engineering Foundation(KOSEF) grant funded by the Korea government(R01-2008-000-20855-0). This work was supported by the Second Brain Korea 21 Project. J. H. was supported by the Korea Science and Engineering Foundation(KOSEF) grant funded by the Korea government(MEST) (No. 2009-0058641)

References


Figure 5: Side by side comparison: 50 × 100 × 50 base simulation (top), upsampled simulation on a 200 × 400 × 200 resolution (middle), our method of 200 × 400 × 200 resolution (bottom)