Volume 24 ('*05*), *Number 3*

Physically-Based Simulation of Tornadoes

Abstract

In this paper, we present a physically-based model to perform 3D visual simulation of tornadoes. Simulating natural phenomena based on physically realistic models has been an active research area in computer graphics. The tornado phenomenon is a spectacular one as it demonstrates the magnificent strength of forces of nature and its shear size. Previous research on tornadoes have focused either on the velocity, pressure and temperature profiles obtained by satellite images and tornado chasing, or on generating visually appealing images without taking into account the underlying physics. In this paper, the tornado dynamics are modeled by the incompressible Navier-Stokes equations with appropriately defined boundary conditions which will result in rotating and uplifting flow movements as found in real tornadoes. To model the complex irregular tornado shapes, a particle system is incorporated with the model equation solutions. Images and simulation sequences generated by volume rendering are presented and compared qualitatively with real tornado photographs.

Keywords: Physically-based simulation, natural phenomena modeling, Navier-Stokes equations, tornado dynamics

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

1. Introduction

As one of the severe phenomena in the world, "A tornado (from the Latin *tornare*, 'to turn') is the most violent storm nature produces" [Flo53]. They are most frequent in the United States; about a thousand tornadoes are reported each year, but they are also seen in many other countries. A tornado may appear anywhere at any time of the year and last only a few minutes. And yet, tornadoes cause incredible amounts of damage and significant numbers of fatalities every year, thus making this phenomenon a major research topic. In order to gather more information about tornadoes, tornado-chasing has become an important activity for many scientists and weather enthusiasts to observe the wind dynamics and collect the real data. The tornado videos recorded offer us further understandings about its appearance and formation. Unfortunately, due to the difficulties to put the recording equipments into the tornado path, the real data collection for tornado's internal dynamics study is still not much of a success today [BG93]. Another alternative is to examine the satellite and radar images. But to build sufficient number of satellites and radars to monitor tornadoes can be very costly. Some scientists create their own "twisters" in laboratories [Lay79]. Tornadoes with different rotation speed can be simulated by varying the apparatus' condition, such as fan speed. However, it is still costly to build or rebuild the apparatus.

In addition to the field and laboratory observations [BG93, Bra69, EJ01, EBSJ01, Flo53, Lay79, NG74], mathematical modeling [Hat03, JP78, Lew93, NF99, Rot93] and numerical simulation [Fie93, KW78, LL97, Rot77, SC02, TF93] have been attractive alternatives for their low risk and relatively low costs. Various models have been proposed to study tornado's physical properties. Some models [NF99] are in cylindrical coordinates (2D) and assume that tornadoes are in axisymmetric internal structure. Although such an assumption seems to be valid for funnel shape tornadoes, many real life tornadoes have arbitrary shapes and hence these models may not be applied. Some others [TF93] are in the cube domain (3D) with the emphasis on asymmetric flow study. To generate the characteristic rotating tornado flow, however, nonphysical boundary conditions are used. Moreover, all the above results focus primarily on the velocity, pressure and temperature profiles and no simulated tornado image is given to show its shape, color, and movement. On the other hand, visually appealing tornado images and animations are often seen on TV and movies. These images are often generated by pure graphics techniques without taking into account the physical characteristics in the tornado evolution process. For instance, in the movie *Twister*, particle systems are used to simulate the tornado movement and shapes changes. In their particle systems, randomness based functions are used to update the particle attributes such as position and velocity.

In this paper, a physically-based tornado simulation is presented to combine the advantages of both numerical modeling and graphics rendering to produce tornadoes with interesting and realistic shapes and motion. To the best of our knowledge, this has never been done in tornado studies. In our simulation, three-dimensional Navier-Stokes equations for incompressible viscous fluid flows are used to model the tornado dynamics; i.e. rotating on the horizontal planes and uplifting on the vertical planes. A particle system is introduced to define tornado volume and a volume rendering algorithm is used to visualize the flow movement based on the particle density. The final images are rendered using OpenGL. In addition, another focus in this tornado simulation is to study tornado formation environment; i.e. how numerical boundary conditions generate tornado shape evolution. We note that the simulation of natural phenomena involving fluid flow is an active area of research in computer graphics (for example [FF01, FSJ01, HBSL03, NFJ02]). A comprehensive review of fluid flow simulations in computer graphics is beyond the scope of this paper. However, we remark that boundary condition study has not been emphasized in natural phenomena simulation, but it is essential to generate interesting shapes of tornadoes.

The paper is organized as follows. Section 2 provides a brief overview about the tornado phenomenon, especially its physical characteristics relevant to the simulation. In Section 3, the modeling framework and dynamics simulation are presented. In particular, the boundary conditions that will lead to rotating and uplifting flow movement will be discussed. Moreover, a particle system is introduced to model the irregular tornado shapes as well as to simulate the debris swirling effect at the bottom. The rendering issues are addressed in Section 4. In Section 5, validation of our model is discussed as well as the presentation of the simulation results. The comparison between the simulations and real tornadoes will also be presented. Finally concluding remarks are made in Section 6.

2. Tornado Phenomena

Tornado is defined as a "violently rotating column of air, in contact with the ground, either pendant from a cumuliform cloud or underneath a cumuliform cloud" [Gli00]. The wind rotation speed can be as low as 40 mph to as high as 300 mph, and the forward motion speed of a tornado averages around 40 mph, but can be as high as 68 mph. In addition to the strong vortical motion in a tornado, the "up-rushing currents of lifting strength" [Flo53, SPZ*01] is also an important characteristic of it. The typical tornado flow structure is described in [LL97] as "the flow spirals radially inward into the center of the vortex that is basically a swirling and rising jet".

Compared to other storms in nature, tornado is the most violent one [EJ01, Lay79]. Some important physical characteristics directly related to the simulation of tornado dynamics are summarized in the following categories:

- *rotation*: the rotation direction of a tornado is counter-clockwise in the northern hemisphere and clockwise in the southern hemisphere [Flo53]. Our simulation results show that the counter-clockwise rotation in the northern hemisphere can be successfully produced by using our model and specific boundary conditions. With the corresponding changes on boundary conditions, our model can be easily applied for the simulation of the clockwise rotation in the southern hemisphere.
- *shape*: a tornado generally takes the shape of a cone, a cylinder, or a funnel in which tornado is equal or greater in diameter at the top than at the bottom [Flo53]. But a tornado with funnel shape is the most common one; see Figure 1(a) for an example of funnel-shape tornadoes. Such funnel shape is the result of the "rotary motion of extremely high speed" [Bra69]. Tornadoes, however, can have some other shapes as well. For example, a tornado can be greater at the bottom [BG93, Flo53, SR93]. And a tornado can also look like a snake or rope [BG93, XWL93] and even make a "right-angle bend" as it nears the ground [BG93, Flo53]. Figure 1(b) shows an example of bended rope-like tornadoes. The animation frames demonstrated in Section 5 show the simulation results of a funnel-shape tornado and a bended tornado. In theory, with modifications on the boundary conditions and variations of particle distribution, our model can produce other shapes as well, such as the cylinder shape or the form that the bottom is greater than the top.

(a) (b)

Figure 1: *(a) A tornado with a funnel shape on the plains of North Dakota, USA (http://ww2010.atmos.uiuc.edu/(Gh)/guides/mtr/svr/torn/home.rxml) (Courtesy of National Severe Storms Laboratory (NSSL), USA). (b) A tornado with an unusual shape making a right-angle bend in the middle (http://www.weatherstock.com/tornadocat3.html) (Courtesy of the Weatherstock, USA).*

- *color*: Laycock [Lay79] describes that a tornado is generally gray or dark. When we see a tornado, what we are really seeing is the water vapor inside the wind or the sucked objects by the swirling strength because the wind itself is invisible. If there is only water vapor in the wind, the color will be gray. But if some objects such as dirt, leaves and bark, are drawn to the wind, the tornado will be dark. It should be noted that not all tornadoes are visible, for example, when a tornado moves through comparatively dry and clean air, it is invisible [Flo53]. But the visible tornadoes are more interesting to viewers, so to produce a tornado with color is one of our goals in the simulation for the best realistic simulation results. The volume rendering algorithm used in this simulation produces visual response with interesting colors as the observed real tornadoes. As shown in Section 5, the rendering algorithm applied can produce gray color and darker tornadoes. Tornadoes with other colors can also be simulated by choosing light source of different colors.

The study on tornado formation and its dynamics has always been a challenging topic. In general, scientists understand that tornadoes form when a "rotating thunderstorm updraft" and a "gyratory wind" meet at the same place [Lag02]. But many other uncertain factors still exist about how a tornado is formed [EBSJ01]. Also, the physical data in a tornado, which is important for tornado dynamics study, is difficult to obtain [BG93]. One reason is that it is hard to place equipment on the right positions since tornadoes are only minutes long and they move with unpredictable paths. Since in the middle of 1980's, researchers at the National Severe Storms Laboratory (NSSL) tried for several years to put the equipment, called TOTO (TOtable Tornado Observatory), into the "coming" path of a tornado, but did not succeed. Samaras and National Geographic [Ves04] tried the similar approach but only five measuring instruments have been deployed successfully into the funnel paths after more than a decade of attempts.

In addition to the field observations, other tornado research methods have also been used in the understanding of the internal structure and dynamics of tornadoes, such as the development of radars, satellites and laboratory experiments. With the improvement in modeling and more accurate numerical simulations, computer animation has played an active role and made significant advances in understanding tornado formation and its dynamics. In the following, we shall explain how we combine physically-based modeling and graphics techniques to simulate realistic tornadoes.

3. Modeling and Dynamics Simulation

The use of mathematical model and numerical simulation in this paper is to model and simulate the tornado flow movement, and then provide physical data at some extent for graphics rendering in order to combine the physically-based modeling and graphics in this tornado simulation. The modeling of tornado can generally be divided into two categories: thunderstrom-scale simulations and tornado-scale simulations [NF99]. The former category is first proposed by Klemp and Wilhelmson [KW78, NF99] who present a model originally for cloud simulation known as KW model, and then this model is applied in tornado simulation to numerically simulate the formation and dynamics of the thunderstorms that are responsible for tornado formation. On the other hand, tornado-scale simulation, first studied by Rotunno [Rot77], assumes that a particular rotating environment caused by the convergence of rotating flow along boundaries leads to the tornado-like vortices, and the models applied in this approach are intended to provide the details of the wind field in the tornado and an understanding of the dynamics that lead to the flow rotation structure. Because the domain boundary and initial conditions are easy to control in the tornado-scale simulation, this approach has been used in laboratories [CS93, Lad93, Rot77, War72] to produce miniature tornadoes. Tornado scientists [Fie93, Lew93, SC02, TF93] apply the tornado-scale approach in their numerical simulations to study the tornado dynamics.

The actual physical models and boundary conditions used in the tornado-scale simulations depend on the process of tornado formation. For example, most tornadoes have the characteristic funnel shape in real life. Thus, the tornado flow structure is often formulated to be axisymmetric [Fie93, Lew93, NF99, SC02]. Based on this assumption, many numerical models adopt 2D Polar coordinate system as given by Sinkevich *et al.* [SC02] and Fiedler [Fie93]. They all assume the tornado funnel rotating about a vertical axis with a known constant angular frequency. The model of Sinkevich *et al.* [SC02] involves the two-phase (liquid and gas) turbulent heat and mass transfer while Fiedler [Fie93] uses an incompressible Navier-Stokes equations model applied in a cylinder domain. It is an axisymmetric convection model with a defined thermodynamic speed limit (the rotation speed) in a closed domain [Fie93]. Navier-Stokes equations are also applied in Trapp *et al.*'s [TF93] simulation, but the model is applied in 3D cube domain and they assume that the flow is compressible. Their goal is to study how tornadoes, which appear "at least locally axisymmetric", are born out of the asymmetric surrounding flow with initial horizontal velocity only [TF93]. As will be discussed in Section 3.1, their simulation is by simulating a tornado-producing environment, and they do not simulate a rotating environment initially as other tornado-scale simulations generally do.

In our simulation, the tornado-scale simulation approach is applied. The physical model used is the governing equations of fluid mechanics, the Navier-Stokes equations, which was originally developed to simulate the viscous and unsteady fluid flow dynamics[Bat67, GDN98]. But recently, it has also been used in computer graphics for the simulation of physical phenomena such as water [FF01], smoke [FSJ01], cloud [HBSL03] and fire [NFJ02]. In the modeling of our simulation, the equations employed are for incompressible fluid flow [Fie93, LL97, NF99]. In contrast with previous models, the axisymmetric tornado internal structure is not assumed here, thus this simulation is more general and allows the tornado to form any shape. Moreover, this model is applied in 3D Cartesian coordinate system instead of the 2D Polar system. The 3D cube model coordinate environment is shown in Figure 2.

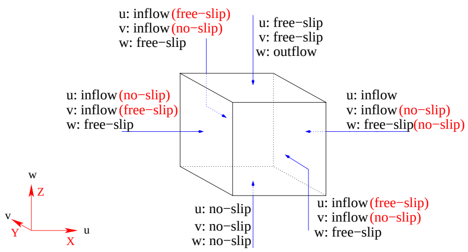


Figure 2: *Coordinate environment and boundary conditions of our model. The boundary conditions inside the parentheses are used in Trapp et al.'s [TF93] simulation.*

The standard Navier-Stokes equations applied in our simulation can be written in dimensionless form as:

$$\vec{u}_t + (\vec{u} \cdot \nabla)\vec{u} = \frac{1}{Re}\Delta\vec{u} - \nabla p + f$$
$$\nabla \cdot \vec{u} = 0,$$

where $\vec{u}$ is the fluid velocity vector, p is the pressure, f denotes body forces such as gravity, and Re is the Reynolds number which represents the viscosity of the fluids. In our mathematical model, the motion of the liquid is described by the evolution of the two dynamic field variables, velocity and pressure. In three dimensions, the velocity vector is denoted by (u, v, w), where u, v and w are velocity components along the X, Y, Z coordinate axes, respectively.

3.1. Boundary Conditions

Unlike many other natural phenomena simulations, the boundary conditions are essential in tornado-scale simulation to achieve the rotation and uplifting movement [NF99, Rot77]. Our boundary conditions are based on Trapp *et al.*'s model; (see Figure 2). The outflow condition on the top is used to simulate the fact, as discussed by Rotunno [Rot93], that the low pressure on the top of tornado forces the air upward which in turn generates the updraft. But the boundary conditions on the vertical planes are different from their model, in which no-slip conditions are used for u and v. They call these boundaries as "stagnation walls". Thus, there is no flow in and out of those vertical planes, forcing the flow to go either up or down. It seems quite awkward and unrealistic. To simulate a more reasonable tornado formation environment, inflow condition is used here for both u and v on the vertical planes. It assumes that flow can go in or out of the vertical boundaries at a flow speed explicitly given. In addition, u and v are assigned specific directions along the vertical boundaries as shown in Figure 3. This rotation implementation is to simulate the counter-clockwise rotation as the tornadoes have in the northern hemisphere.

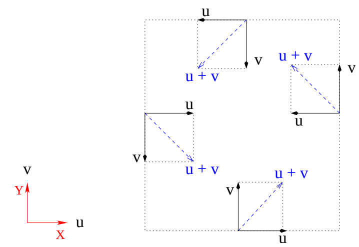


Figure 3: *Velocity directions on X-Y plane to simulate the counter-clockwise rotation in the northern hemisphere.*

But how to appropriately implement such inflow conditions of u and v on those boundaries leads to more subtle questions. Previous results [Rot93, TF93] indicate that applying a vertical shear flow with varied velocity magnitude on one of the vertical boundaries is a reasonable choice. For example, in Trapp *et al.*'s [TF93] simulation, a vertical shear flow enters the domain through one boundary. Also as pointed out by Rotunno [Rot93] and Houze [Hou93], such vertical shear flow generally leads to the horizontal rotation at the end. Based on our specific boundary conditions applied on the vertical boundaries, our experiments show that only applying the vertical shear flow on one boundary can produce rotation, but it cannot generate a nice tornado shape at reasonable simulation time. Our simulation results presented in Section 5 demonstrate that by applying shear flow on all 4 vertical boundaries with certain varied velocity magitudes can lead to different shapes at the end. For example, to simulate the funnel shape tornadoes, on the 4 vertical boundaries, we can set the velocity magitude of u and v decreasing from the bottom to the top. One implementation is to set u and v equal to $2 \times \frac{nz-k}{nz}$, where nz is the number of segments in the numerical discretization along the Z direction, k is the index along the Z coordinate and it is from 0 to nz (bottom to top). Other tornado shapes such as a tornado having an angle can be generated by varying the velocity magitude function accordingly.

3.2. Numerical Solution

The $nx \times ny \times nz$ domain in Figure 2 is discretized as a staggered grid, in which the different unknown variables are not located at the same grid points [GDN98]. More precisely, for a grid cell in 3D, pressure p is located in the cell center while u, v and w are at the center of the right, back and top planes of that cell respectively. This staggered arrangement of the unknowns prevents possible pressure oscillations which could occur had we evaluated all unknown values of u, v, w and p at the same grid points. The values of the unknowns near the boundaries are defined using the boundary conditions as described in Section 3.1.

The equations are solved by a finite difference method. The velocity field is updated using the projection method [Cho68]. Conjugate gradient is applied to solve the Possion equation for the solutions of pressure p at time t given the velocities u, v and w at the previous time step.

3.3. Particle System

A particle system is used to model the shape changes and flow dynamics of the tornado simulation. Such a technique has also been used in other natural phenomena simulation (e.g. [FSJ01, HBSL03]). In our simulation, a particle has velocity, position and lifetime attributes. The initial position of a particle depends on the purpose of such particle. For example, at the beginning of the simulation, certain number of particles are added into the system at random positions to describe how the cube shape converges to a particular tornado shape at the end. But some other particles are added at later steps at specific locations to simulate some other effects, such as color variation and bottom debris swirling effect. The lifetime attribute is defined by the position instead of time as a particle system usually does. Specifically, during the simulation, a particle dies if it is out of the system boundaries; otherwise, its position is updated based on its velocity (speed and direction). As mentioned before, this particle system is coupled with the model's numerical solutions. So at each time step, the position attribute is updated based on the velocity solutions instead of using random process as used in [Fan96]. Therefore, to simulate a particular tornado shape, the particle positions can be controlled accordingly by applying different boundary conditions.

Moreover, to simulate the debris swirling effect at tornado bottom, some special particles are generated at each time step. All these particles have mass attribute. In the simulation, the mass value is randomly initialized when the particle enters the system. Then, by applying gravity and centrifugal force, the velocity attribute of these particles is modified differently from the numerical solutions. More precisely, consider the particle as shown in Figure 4. By Newton's law of motion, the magnitude of the centrifugal force of this particle can be computed by

$$F = m\frac{u^2+v^2}{\sqrt{(x-x_c)^2+(y-y_c)^2}},$$

where m=mass, (u,v)=velocity on the horizontal plane, (x,y)=particle position, and (x_c,y_c)=approximate position of the center of the tornado. The force components F_x and F_y can be computed using the angle β of the particle direction. Since acceleration $a = F/m$, the velocity of this particle is updated by

$$u^{new} = u + \delta t\tilde{a}_x$$
$$v^{new} = v + \delta t\tilde{a}_y,$$

where $a = (a_x, a_y)$, $\tilde{a}_x = a_x\nu$, $\tilde{a}_y = a_y\nu$, and ν is a control parameter. As a consequence of applying the centrifugal force, the particle will move away from the circular path. How far away the particle moves depends on the value of ν, and this value varies for different simulation goals. For example, ν is randomly selected between 0.3 and 0.9 in our simulations. Finally, on the Z coordinate, w is updated after gravity is applied.

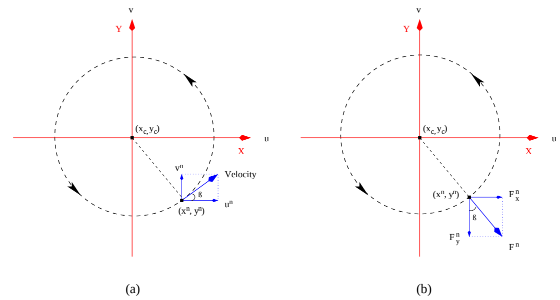


Figure 4: *The decomposition of velocity and centrifugal force for a particle in the bottom debris swirling simulation. (a) Velocity decomposiiton. (b) Centrifugal force decomposition.*

After applying the centrifugal force and gravity to the particles specifically for the debris swirling simulation at the bottom, these particles rotate, go up slowly and move gradually away from the tornado center.

4. Rendering Issues

To visualize the dynamics of the particles and hence the tornado, volume rendering is used. In this tornado simulation, the cube domain in Figure 2 is discretized into cells. Depending on the position attribute, each particle is mapped to a particular cell. Then a modified metaball method is used to calculate and smooth the particle density for each cell. The particle density of a cell is calculated using the metaball functions proposed by Wyvill *et al.* [WT90]. To simplify implementation, the metaball used here is actually not a ball, but a cube. That is, the metaball with radius R for a cell C is a cube with length $2R+1$. The density function ρ_C for cell C is then given by [DKY*00, WT90]:

$$\rho_C = \sum_{i,j,k\in\Omega(C,R)} K(i,j,k)\,f(r)$$
$$f(r) = \begin{cases} -\frac{4}{9}(\frac{r}{R})^6 + \frac{17}{9}(\frac{r}{R})^4 - \frac{22}{9}(\frac{r}{R})^2 + 1, & r \le R \\ 0, & r > R, \end{cases}$$

where $\Omega(C,R)$ is the set of cells whose centers are included in cell C's metaball, $K(i,j,k)$ is the number of particles of a particular cell in $\Omega(C,R)$, r is the distance of a cell to cell C, and f is the density contribution function which has Gaussian-like shape. The metaball size determines the smoothing result and the computing time. In our simulations, metaball size of $R = 3$ is used.

After texture mapping and antialising are applied, each cell color is rendered in OpenGL. The volume rendering step is based on Dobashi *et al*'s algorithm for cloud simulation [DKY*00]. The algorithm employs the standard forward mapping (or splatting) techniques [Wes91]. As shown in Figure 5, the camera is placed at the viewer's position and the perspective view is applied. Then, all the cells are projected onto the image plane. The calculation order is based on the distance of a cell to the viewer from the back to the front. The color used for each call is set by the user and the cell's density is used to control the extent of opacity.

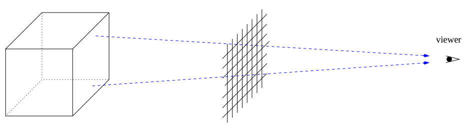


Figure 5: *The surface brightness of a tornado is determined by the cell particle densities projected onto the image plane along the viewing direction.*

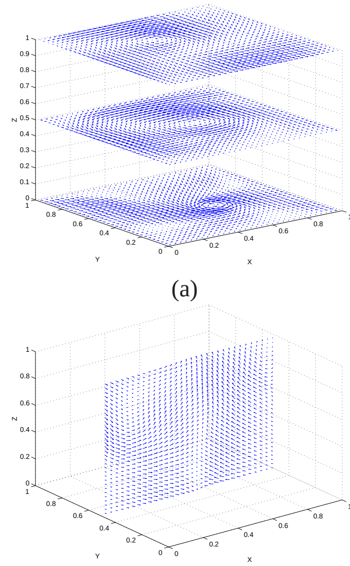
We remark that the blending factors used in [DKY*00] cannot be simply reapplied here due to the fact that blending percentage of the next rendered pixel with the value already rendered depends on the visual representation of the simulated phenomena. Hence, instead of using (1,*GL_SRC_ALPHA*) as in [DKY*00], we use the blending factors of (*GL_SRC_ALPHA*, $1 - $ *GL_SRC_ALPHA*) to obtain the realistic visual results in tornado simulation.

5. Results

In this section, we present examples of simulations of tornadoes. Due to stochastic nature of tornadoes, we qualitatively analyze the simulations in comparison with photographs of real tornadoes. The computations were performed on a PC with a 1GHz Pentium 3 processor.

First, we compare the simulated tornado flow with previous scientific simulation result. By applying the specific boundary conditions presented in Section 3.1, our simulation can produce tornado-like rotation and uplifting effect. In Figure 6(a), the velocity vectors on the horizontal planes close to the bottom, at the middle, and near the top are shown. As one can see, the vortex intensity close to the bottom boundary is higher, lower on the middle plane and lowest on the top boundary. This simulated tornado flow profile is in good agreement with the results in [TF93]. The uplifting effect is illustrated in Figure 6(b) where the velocity vectors on the middle slice of the *X*-*Z* plane are shown. The wind flow converges at the center and goes up. This result is also consistent with the discussion in [TF93].

We now present tornadoes with different shapes. In Figure 7, images of a funnel-shape tornado and a tornado with an angle are presented. The boundary conditions and velocity magnitudes on the boundaries used in Figure 7(a) are



(a) (b)

Figure 6: *(a) Velocity vectors of u and v close to the bottom boundary, at the middle and near the top of the X-Y planes. (b) Velocity vectors of u, v and w on the X-Z plane sliced in the middle.*

described in Section 3.1. In Figure 7(b), we use a smaller time step size and larger velocity magnitudes on the vertical boundaries close to the bottom to simulate the visual effect of small funnel diameter ratio of the bottom to the top. Compared with the real photographs of tornadoes shown in Figure 1, qualitatively speaking, our simulations can generate tornadoes with interesting and realistic shapes. Another shape comparison with a real tornado is given in Figure 8 (photograph of a tornado) and Figure 9 (simulation of a tornado). The funnel shape near the bottom and the bend in the middle are clearly observed in the simulated image. We note that these tornadoes are not axisymmetric and hence previous 2D tornado models would not be able to simulate such shapes.

(a) (b)

Figure 7: *Simulated tornado shapes. (a) A funnel-shape tornado. (b) A tornado with an angle.*

Figure 8: *A funnel-shape tornado in Neal, Kansas, USA on May 30, 1982 (http://www.chaseday.com/tornado-neal.htm) (Courtesy of G. Moore).*

Figure 9: *(b) Simulation of a tornado showing a bend funnel shape.*

The debris swirling effect at the bottom is simulated after centrifugal force and gravity are applied to some of the particles. The simulated tornado is compared with a real one in Figure 10. The simulated tornado shape resembles the arrow-like shape of the real tornado created by the swirling of debris from the ground; see Figure 11. We note that the particles in this simulation do not interact with the environment, and so collision detection is not performed in this simulation.

Figure 10: *An arrow-shaped tornado due to swirling of debris from the ground occurred near the town of Woonsocket, USA on June 24, 2003 (http://magma.nationalgeographic.com/ngm/0404/feature1/zoom3.html) (Courtesy of National Geographic).*

In real life, tornadoes often change shape continuously over time. Figure 12 presents four frames of a real tornado video, illustrating different (funnel) shapes at different times. Figure 13 presents eight frames from a simulation sequence

Figure 11: *A simulated tornado with particles simulating the debris swirling effect.*

showing dynamic shape changes of a tornado. In this sequence, one can observe the ratio of the diameter of the bottom to that of the top changes over time.

The parameters used in the implementation are as follows: Reynolds number $Re = 1000$, grid size $nx \times ny \times nz$ for solving the Navier-Stokes equations is $32 \times 32 \times 32$. For the particle density calculation, the grid size of the cube domain is $250 \times 250 \times 250$. Other parameters used in the simulations are given in Table 1.

6. Conclusion

In this paper, we have proposed a physically-based tornado simulation method which produces realistic animation results. The model applied is based on the 3D Navier-Stokes equations. A particle system is incorporated with the equation solutions to model the irregular tornado shape, and a modified metaball scheme is used in the density distribution calculation. Finally, the images are generated using OpenGL based on volume rendering techniques.

The tornado studies in this simulation are useful to understand the tornado flow movement and facilitate the understanding of tornado formation environment. The simulation method used here has the following advantages:

- The combination of physically-based modeling and graphics is used to simulate tornado flow dynamics.
- Using the model in this simulation, appropriate boundary conditions lead to realistic tornado flow generation. Also, different and interesting tornado shapes can be easily simulated by varying the number of particles generated at each time step.
- The density smoothing scheme can be modified by choosing different metaball radius and density contribution function.

We also note that our research is just an early step in developing a fully physical simulation of tornadoes and hence there are numerous limitations in our model. The main difficulty in studying this problem stems from the lack of measurement data (e.g. wind speed, air pressure, temperature). In this

Figure 12: *Frames of a real tornado animation which dynamically changes shape during its lifetime (http://iwin.nws.noaa.gov/iwin/videos/tv5.avi) (Courtesy of the National Weather Service (NWS) of National Oceanic and Atmospheric Administration (NOAA), USA).*

Figure 13: *Frames from a simulation sequence showing the dynamic change of tornado shape.*

paper, we have combined the modeling and graphics visualization of realistic tornado flows without knowledge of the physical data. In the future, more physical factors need to be integrated into the model in order to simulate a more realistic tornado formation environment. Also, the particle system does not take into account the interaction with the environment. Besides, the rendering does not consider the lighting physics, such as scattering. More physics and graphics need to be incorporated in the simulation and rendering process in the future.

References

[Bat67] BATCHELOR G.: *Introduction to Fluid Dynamics*, first ed. Cambridge University Press, Cambridge, UK, 1967. 3

[BG93] BLUESTEIN H., GOLDEN J.: A review of tornado observations. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 319–352. 1, 2, 3

[Bra69] BRADFORD S.: *A Study of Tornado and Tornado Producing Synoptic Systems*. Master's thesis, University of Wisconsin, USA, 1969. 1, 2

[Cho68] CHORIN A.: A numerical solution of the Navier-Stokes equations. *Math. Comp. 22* (1968), 745–762. 4

[CS93] CHURCH C., SNOW J.: Laboratory models of tornadoes. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 277–295. 3

[DKY*00] DOBASHI Y., KANEDA K., YAMASHITA H., OKITA T., NISHITA T.: A simple, efficient method for realistic animation of clouds. In *Pro-*

Image	Metaball Radius	Time step size	# of particles ($t = 0$)	# of particles added ($t > 0$)	Frame(s) order	Total # of particles
Figure 7(a)	3	0.005	500,000	1,200/step	344th	996,800
Figure 7(b),9	3	0.00166	450,000	5,000/9 steps	932nd	970,000
Figure 11	3	0.005	600,000	1,200/step, 40 of 1,200 for bottom simulation	344th	1,096,800
Figure 13	4	0.00166	450,000	5,000/9 steps	702nd - 902nd	950,000

Table 1: *The parameters used in the simulations.*

ceedings of SIGGRAPH (2000), pp. 19–28. 5, 6

[EBSJ01] ETKIN D., BRUN S., SHABBAR A., JOE P.: Tornado climatology of canada revisited: Tornado activity during different phases of enso. *International Journal of Climatology 21* (2001), 915–938. 1, 3

[EJ01] EASTLUND B., JENKINS L.: Taming tornadoes: Storm abatement from space. *IEEE Aerospace Conference Proceedings 1* (2001), 389–395. 1, 2

[Fan96] FANGMEIER S.: Designing digital tornadoes. *Computer Graphics World 8* (1996), 65–73. 5

[FF01] FOSTER N., FEDKIW R.: Practical animation of liquids. In *Proceedings of SIGGRAPH* (2001), pp. 23–30. 2, 3

[Fie93] FIEDLER B.: Numerical simulation of axisymmetric tornadogenesis in forced convection. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 41–48. 1, 3

[Flo53] FLORA S. (Ed.): *Tornadoes of the United States*. University of Oklahoma Press, Norman, Oklahoma, USA, 1953. 1, 2

[FSJ01] FEDKIW R., STAM J., JENSEN H.: Visual simulation of smoke. In *Proceedings of SIGGRAPH* (2001), pp. 15–22. 2, 3, 4

[GDN98] GRIEBEL M., DORNSEIFER T., NEUNHOEFFER T.: *Numerical simulation in fluid dynamics: A practical introduction*, first ed. Society for Industrial and Applied Mathematics, Philadelphia, USA, 1998. 3, 4

[Gli00] GLICKMAN T. (Ed.): *Glossary of Meteorology*, second ed. American Meteorological Society, Boston, USA, 2000. 2

[Hat03] HATTON L.: Coordinate transformations and two exact solutions of the axisymmetric navier-stokes equations. Computing Laboratory, Univeisity of Kent, UK, 2003. 1

[HBSL03] HARRIS M., BAXTER W., SCHEUERMANN T., LASTRA A.: Simulation of cloud dynamics on graphics hardware. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware* (2003), pp. 92–101. 2, 3, 4

[Hou93] HOUZE R.: *Cloud Dynamics*, first ed., vol. 53 of *International Geophysics Series*. Academic Press, Inc., San Diego, USA, 1993. 4

[JP78] JISCHKE M., PARANG M.: On tornado funnels. In *Oklahoma Academy of Science* (1978), pp. 81–87. 1

[KW78] KLEMP J., WILHELMSON R.: The simulation of three-dimensional convective storm dynamics. *Journal of the Atmospheric Sciences 35* (1978), 1070–1095. 1, 3

[Lad93] LADUE J.: Vortex formation from a helical inflow tornado vortex simulator. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 307–316. 3

[Lag02] LAGASSE P. (Ed.): *The Columbia Encyclopedia*, sixth ed. Columbia University Press, New York, USA, 2002. 3

[Lay79] LAYCOCK G.: *Tornadoes, killer storms*, first ed. David McKay Company, Inc., New York, USA, 1979. 1, 2

[Lew93] LEWELLEN W.: Tornado vortex theory. In

The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79) (1993), pp. 19–39. 1, 3

[LL97] LEWELLEN W., LEWELLEN D.: Large-eddy simulation of a tornado's interaction with the surface. *Journal of the Atmospheric Sciences 54*, 5 (1997), 581–605. 1, 2, 3

[NF99] NOLAN D., FARRELL B.: The structure and dynamics of tornado-like vortices. *Journal of the Atmospheric Sciences 56* (1999), 2908–2936. 1, 3, 4

[NFJ02] NGUYEN D., FEDKIW R., JENSEN H.: Physically based modeling and animation of fire. In *Proceedings of SIGGRAPH* (2002), pp. 721–728. 2, 3

[NG74] NOVLAN D., GRAY W.: Hurricane-spawned tornadoes. *Monthly Weather Review*, 102 (1974), 476–488. 1

[Rot77] ROTUNNO R.: Numerical simulation of a laboratory vortex. *Journal of the Atmospheric Sciences 34* (1977), 1942–1956. 1, 3, 4

[Rot93] ROTUNNO R.: Supercell tunderstorm modeling and theory. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 57–73. 1, 4

[SC02] SINKEVICH O., CHIKUNOV S.: Numerical simulation of two-phase flow in a tornado funnel. *High Temperature 40*, 4 (2002), 604–612. 1, 3

[SPZ*01] SAVORY E., PARKE G., ZEINODDINI M., TOY N., DISNEY P.: Modelling of tornado and microburst-induced wind loading and failure of a lattice transmission tower. *Engineering Structures 23* (2001), 365–375. 2

[SR93] SZOKE E., ROTUNNO R.: A comparison of surface observations and visual tornado characteristics for the june 15, 1988, denver tornado outbreak. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 353–366. 2

[TF93] TRAPP R., FIEDLER B.: Numerical simulation of tornadolike vortices in asymmetric flow. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 49–54. 1, 3, 4, 6

[Ves04] VESILIND P.: Chasing tornadoes. *National Geographic 4* (2004), 2–38. 3

[War72] WARD N.: The exploration of certain features of tornado dynamics using a laboratory model. *Journal of the Atmospheric Sciences 29* (1972), 1194–1204. 3

[Wes91] WESTOVER L.: *Splatting: a Parallel, Feed-Forward Volume Rendering Algorithm*. PhD thesis, University of North Carolina, Chapel Hill, USA, 1991. 5

[WT90] WYVILL G., TROTMAN A.: Ray-tracing soft objects. In *Proceedings of the eighth international conference of the Computer Graphics Society on CG International '90: computer graphics around the world* (1990), pp. 469–476. 5

[XWL93] XU Z., WANG P., LIN X.: Tornadoes of china. In *The Tornado: its structure, dynamics, prediction, and hazards (Geophysical Monograph 79)* (1993), pp. 435–444. 2