Modelling of Smoke Flow Taking Obstacles into Account

Satoru Yoshida Tomoyuki Nishita
Dept. of Information Science, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, 113-0033, Tokyo, Japan
{yoshida, nis}@nis-lab.is.s.u-tokyo.ac.jp

Abstract

Recently, dsplaying natural phenomena such as smoke has become a topic of interest in computer graphics. The ability to simulate the complex shapes and motion of the smoke particles is not only important but is also a difficult problem to solve. In this paper, we propose a method of displaying swirling smoke, including the consideration of its passage round obstacles. By using the idea of metaballs, we can easily represent the 3-dimensional density distribution of smoke. We solve the physical equation of the flow, and represent the vortices by using the vorticity vector. Therefore, we can make a model of the smoke flow even if there are obstacles in it's path.

Keywords and phrases: *Smoke, Metaballs, Vorticity, Computational Fluid Dynamics, Natural Phenomena.*

1. Introduction

Methods of displaying smoke are used in various fields, such as visual simulation, entertainment, etc. Therefore, there is quite a demand in several industries to mimic the appearance and motion of smoke. The problem of modelling it's complex behaviour from it's interactions with the surrounding air and obstacles is a topic of interest. We can see such motion as a part of so many everyday scenes.

We can handle smoke as a set of small smoke particles, but the interaction between the smoke and obstacles is difficult to evaluate when we simply consider the forces that act upon these smoke particles. We can deal with smoke as a type of fluid, and when we deal with it in this way we can simulate more accurately smoke flow around obstacles. Therefore, when we want to describe a realistic flow and to consider its passage around obstacles, we have to take into account some of the fluid properties of the smoke.

Firstly, we have to represent the non-uniform density distribution of smoke, which changes as time goes by. Since smoke is one kind of the fluid, we can consider the smoke as a set of smoke clusters. We can then consider a model of the smoke's density distribution. In this model we deal with some of the smoke particles as a set, and hence deal the smoke in it's entirety as a combination of these sets. The main problem is how we define these sets.

Secondly, smoke has a complex motion because it exhibits turbulent flow. In turbulence there are random velocity components, which make the calculation of the flow complex. When we consider the introduction of obstacles into the flow path, the problem becomes even more difficult, because the obstacles clearly affect the flow. An animator takes pains to depict such flows by only inputting the velocity by hand. The modelling and simulation of such flows are, of course, important problems in the fields of science and engineering. Fluid simulations are based on the standard mathematical framework. Many methods have been proposed in various areas regarding how we can compute the fluid's flow numerically. We can choose one of these methods to calculate the flow of smoke around obstacles.

Thirdly, there are a lot of vortices in turbulence. We have to consider vortices which cause swirling within the smoke. The strength of the rotational velocity and the direction of the rotational axis depend on the condition of the velocity distribution. Therefore, we need to model these properties of the vortices. It is one of the fundamental problems regarding how we deal with vortices in the fluid simulation system.

Therefore, we propose a method of displaying smoke with these three properties. A summary of our method is as follows.

(1) We use metaballs to simulate the density distribution of smoke. Each metaball has a 3-dimensional density distribution, and we consider that the smoke is a combination of metaballs.

(2) We calculate the velocity field, which can be expressed by the physical equations of the flow. The result of the equations represents a realistic flow around obstacles.

(3) We generate small vortices by a concept called 'vorticity'. The vorticity vector defines the strength and direction of the vortices. By using these vortices, we can display the swirling smoke.

2. Previous Work

Many methods have been proposed to display the simulated flow of smoke or clouds. Clouds have some properties similar to those of smoke, so we can often apply the method used for clouds to also display smoke.

It seems to be natural to think of modelling the smoke as a set of smoke particles. The particle system is popular for natural phenomena. In this method, gas is dealt with as a set of particles. Reeves [1], Ebert, Carlson, and Parent [2], and Reeves and Blau [3] displayed smoke by using the particle system. However, to model the density distribution of the smoke requires a large number of particles.

One of the methods to model the density distribution of smoke is to define a function that shows the density distribution of smoke or clouds. Musgrave created a typhoon by a procedural approach using 2-dimensional fractals [4]. Gardner [5] and Nishita et al. [6] displayed smoke by using a Fourier series. However, these three methods use mathematical functions to define the smoke flow, so it is difficult to add physical parameters to represent a more realistic smoke flow.

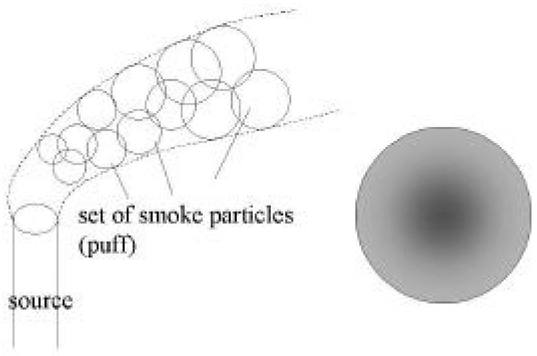
There have been several approaches to modelling the flow of the smoke or clouds. For depicting clouds, Kajiya et al. simulated cloud formation by considering the phase transition effects from vapor to water [7]. However, this method is complex and time-consuming. Sakas [8], Stam et. al. [9], and Shinya et. al. [10] have modeled the turbulent motion by taking a stochastic approach. The user simulates the flow by stochastic parameters, so it is not easy to add the physical parameters which the user requires to these systems.

One of the approaches to representing the turbulence is to consider the wind field or vortex field. Fluid's flow satisfies the physical equations such as Navier-Stokes equations [11, 12], and we can solve these equations on a computer for the wind field or vortex field. Wejchert and Haumann [13] modelled the motion of gas by using a wind field. Sims [14] also used a wind field, and introduced the concept of random turbulence. However, in these two methods users have to set the velocity or vortices manually. To model the smoke automatically, we need to simulate the smoke flow, and many methods have been proposed to achieve this. Yaeger, Upson, and Myers [15] first used 2-dimensional vortices coupled with a Poisson solver to calculate turbulence flow, and generated an animation of the surface of the planet Jupiter. Later, Chen et al. solved the 2-dimensional Navier-Stokes equations and animated water surfaces by utilising the pressure term [16]. Chiba et al. also generated the 2-dimensional vortex field [17]. Results using these methods are limited to using a fixed viewpoint to get realistic images.

Foster et al. modelled the 3-dimensional smoke flow around obstacles by solving the Navier-Stokes equations [18], although the calculation used in this method can be unstable according to the parameters used. Since the allowable parameters in this method are restricted for achieving stable calculation, it is troublesome to display swirling smoke. Stam et al. proposed a method using the same equations [19]. They solved the equations in a faster and more stable way for real-time animation, but this method cannot be applied for the other equations of flow, which include the random velocity components in the turbulence.

In our method, we use metaballs to model smoke based on a physical-based fluid model, so that we can represent the 3-dimensional density distribution of smoke, and it requires less data than the particle system. We use equations that take into account random velocity in order to represent a more realistic flow. Furthermore, to represent the swirling smoke, vortices are evaluated using the calculated flow.

3. Modelling the Smoke by Using Metaballs



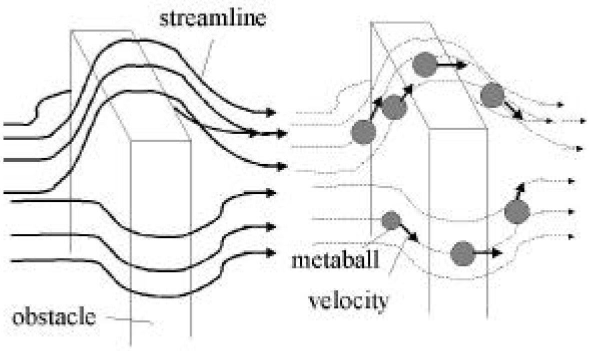
(a)Set of smoke particles (b)Example of density distribution

Figure 1: The smoke puffs and the density distribution.

Smoke has a complex density distribution due to the nature of smoke particles. To model this distribution by using the particle system, a lot of particles are needed. To model the density distribution of the smoke with less data, we consider a 'set' of smoke particles.

As shown in Figure 1, we can deal with smoke as a combination of smoke clusters. This cluster is called a "puff." The puff is ball-shaped, and has a density distribution, which satisfies the following condition. The density of the puff is highest at the center, and the farther from the center, the lower it becomes. In general, the Gaussian function can be applied to represent the density distribution of the smoke puff, but it has infinite domain. We use a metaball to define the density distribution of the puff. In our method, we use a sixth degree function for the density distribution of the metaball [20].

We calculate the flow of metaballs in the velocity field by the method described in Chapter 4. This velocity field is calculated in advance. We make metaballs move by considering the calculated velocity field. As shown in Figure 2(a), the velocity field represents the flow avoiding obstacles, and so the flow of metaballs also avoids obstacles as shown in Figure 2(b).



(a)Velocity field (b)Smoke flow by using velocity field

Figure 2: The velocity field and the metaballs' flow.

The radius of the smoke puff is not constant, because smoke particles diffuse in the puff. To simplify this, we assume that the radius of the metaball increases proportionally as time passes, and the larger the radius of the metaball is, the less dense the metaball becomes.

4. Calculation of the Flow by Using the Velocity Field Theory

In this section, we derive a numerical model for the smoke flow. The space around the surface of the earth is called the "boundary turbulence layer," which has random velocity components. Let the velocity vector to be $\mathbf{u} = (\overline{u},\overline{v},\overline{w})$, the average velocity vector to be $\overline{\mathbf{u}} = (\overline{u},\overline{v},\overline{w})$, and the random velocity vector to be $\mathbf{u}' = (u',v',w')$. Each velocity component is the sum of the average component and random component (i.e. $\mathbf{u} = \overline{\mathbf{u}} + \mathbf{u}'$). A fluid is represented as a combination of a temperature field and velocity field. The rotational, buoyancy and convective components of the smoke motion are modeled by the Navier-Stokes equations. Foster et al. solved the Navier-Stokes equations to simulate the smoke flow. These equations are popular, but they don't consider the random velocity components of the turbulence. There are better equations to describe the forces acting in the smoke flow. We use the Reynolds equations, which consider the random velocity components of the flow.

4.1 The equation of the flow

The velocity of the fluid is affected by a number of factors such as convection, buoyancy, pressure, and so on. The Reynolds equations fully describe the forces acting within the smoke flow, and consider the effects of random velocity components. The Reynolds equations are:

$$\frac{\partial \overline{u}}{\partial t} = -(\frac{\partial \overline{u}^2}{\partial x} + \frac{\partial \overline{uv}}{\partial y} + \frac{\partial \overline{uw}}{\partial z}) + n(\frac{\partial^2 \overline{u}}{\partial x^2} + \frac{\partial^2 \overline{u}}{\partial y^2} + \frac{\partial^2 \overline{u}}{\partial z^2}) - n_t(\frac{\partial \overline{u'^2}}{\partial x} + \frac{\partial \overline{u'v'}}{\partial y} + \frac{\partial \overline{u'w'}}{\partial z}) - \frac{1}{r}\frac{\partial p}{\partial x},$$

$$\frac{\partial \overline{v}}{\partial t} = -(\frac{\partial \overline{v}^2}{\partial y} + \frac{\partial \overline{vu}}{\partial x} + \frac{\partial \overline{vw}}{\partial z}) + n(\frac{\partial^2 \overline{v}}{\partial x^2} + \frac{\partial^2 \overline{v}}{\partial y^2} + \frac{\partial^2 \overline{v}}{\partial z^2}) - n_t(\frac{\partial \overline{v'^2}}{\partial y} + \frac{\partial \overline{v'u'}}{\partial x} + \frac{\partial \overline{v'w'}}{\partial z}) - \frac{1}{r}\frac{\partial p}{\partial y},$$

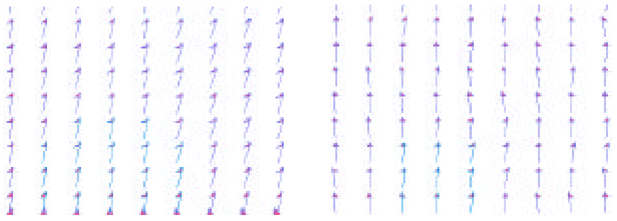
$$\frac{\partial \overline{w}}{\partial t} = -(\frac{\partial \overline{w}^2}{\partial z} + \frac{\partial \overline{wu}}{\partial y} + \frac{\partial \overline{wv}}{\partial z}) + n(\frac{\partial^2 \overline{w}}{\partial x^2} + \frac{\partial^2 \overline{w}}{\partial y^2} + \frac{\partial^2 \overline{w}}{\partial z^2}) - n_t(\frac{\partial \overline{w'^2}}{\partial z} + \frac{\partial \overline{w'u'}}{\partial x} + \frac{\partial \overline{w'v'}}{\partial y}) - \frac{1}{r}\frac{\partial p}{\partial z}$$

$$+ F_{bv}, \qquad (1)$$

where r is the density, p the pressure, n the coefficient of the viscosity, and n_t the coefficient of eddy viscosity.

To calculate Equation (1), we subdivide the whole space that contains the smoke flow into voxels (see Section 4.2). By using Equation (1), we can calculate the average velocity components in each voxel.

In the right sides of Equation (1), the third terms are the terms incorporating the random velocity components, which the Navier-Stokes equations do not have. These terms generate the random flow, which cannot be represented by using the Navier-Stokes equations. It is known that the strength of the random flow is several percent of the average flow. We can determine the random velocity component as the random value whose strength is several percent of the average velocity components. Figure 3 is a comparison of the velocity field with two differential equations. By using the Navier-Stokes equations we can only consider a simple flow, but the random components of the flow can be represented by the Reynolds equations.



(a)Navier-Stokes equations (b)Reynolds equations

Figure 3: The comparison of the velocity field with two different equations.

Also in the right side of Equation (1), the fourth terms are the pressure terms. When we calculate Equation (1), we divide the pressure term and the other terms. Such a division could minimize the error (see Section 4.4).

In Equation (1), there is a buoyancy term F_{bv} only used by the third equation. To calculate it, we have to consider the temperature field. Temperature T is defined as the sum of the average component $\overline{T}$ and the random component T'. To calculate the temperature field, we solve the equations of heat flow, which have some analogy with the Reynolds equations:

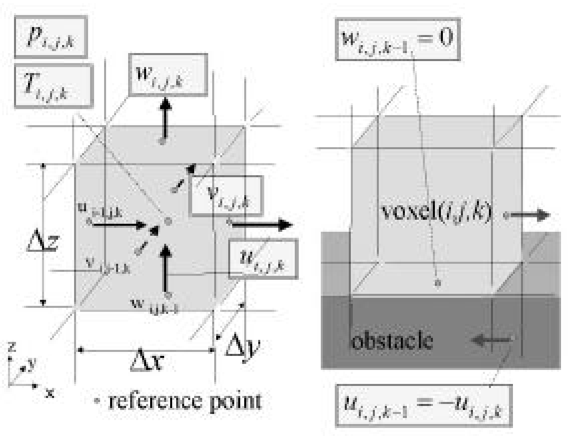
$$\frac{\partial \overline{T}}{\partial t} = -(\overline{u}\frac{\partial \overline{T}}{\partial x} + \overline{v}\frac{\partial \overline{T}}{\partial y} + \overline{w}\frac{\partial \overline{T}}{\partial z}) + a(\frac{\partial^2 \overline{T}}{\partial x^2} + \frac{\partial^2 \overline{T}}{\partial y^2} + \frac{\partial^2 \overline{T}}{\partial z^2}) + a_t(\frac{\partial^2 T'}{\partial x^2} + \frac{\partial^2 T'}{\partial y^2} + \frac{\partial^2 T'}{\partial z^2}), \quad (2)$$

where a are the coefficient of the heat diffusivity, and a_t is the coefficient of eddy and heat diffusivity.

Before calculating the buoyancy term, we calculate the temperature field, and F_{bv} can be calculated as [18].

There are some conditions to calculate Equation (1), called "boundary conditions". In our method, we consider the conditions described in Section 4.3.

4.2 Definition of the voxel



(a)Voxel (i, j, k) (b)Example of boundary conditions

Figure 4: The definition of voxel and boundary conditions.

To calculate Equation (1) in Section 4.1, we subdivide the whole space that contains the smoke flow into voxels. We can use other elements instead of voxels, but the regular voxel grid makes it easy to understand how to calculate the physical parameters such as velocity, pressure, and temperature. We deal with the coordinate systems as 3-dimensional Euclid space in which the z-axis represents the vertical direction and the x- and y-axes represent the horizontal directions. The number of subdivisions is N_x, N_y, and N_z in the x, y, and z directions, respectively.

Within each voxel, there are some physical variables of the smoke. The voxel can be labeled (i, j, k) $(0 \le i < N_x$, $0 \le j < N_y$, $0 \le k < N_z)$ the velocity vector is $(u_{i,j,k}$, $v_{i,j,k}$, $w_{i,j,k})$, the temperature is $T_{i,j,k}$, and the pressure is $p_{i,j,k}$. To define these variables, we set some "reference points" on the surface of the voxel, and calculate the value of the variables at these points. As shown in Figure 4(a), at the center of each face of the voxel, we define a velocity component perpendicular to the face, and at the center of the voxel, we define variables of temperature and pressure.

4.3 Boundary conditions

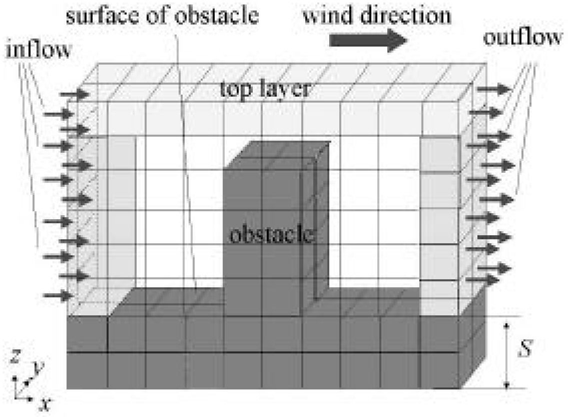


Figure 5: The boundary conditions.

To calculate Equation (1), we have to consider some conditions called "boundary conditions." For the boundary conditions, we consider the velocity on the surface of the obstacle, the velocity of inflow, the velocity of outflow, and the velocity at the top layer. We consider these conditions for the calculation of Equation (1).

(1) We consider the velocity on the surface of the obstacle. The velocity on the surface of the obstacle is defined to be zero. For example, we consider the situation as shown in Figure 4(b). In this situation, voxel $(i, j, k-1)$ is in the obstacle and voxel (i, j, k) is out of it. We can set the velocity component $w_{i,j,k-1}$ in the vertical direction to be zero. In the horizontal direction, since there is no reference point on the surface, it is difficult to define the velocity component in this direction. We can set the velocity component $u_{i,j,k-1}$ inside the obstacle to be $-u_{i,j,k}$ to approximate the velocity on the surface of the obstacle.

(2) We have to define the flow at the boundary of the analytical space. We call such a flow the "inflow" (see Figure 5). Simply speaking, we assume that there is a flow only in the x direction. For the voxels in which label in the x direction is zero (voxel $(0, j, k)$, $0 \le j < N_y$, $0 \le k < N_z$), there is inflow. At a location close to the surface of the obstacle, the velocity is zero, and the higher the position becomes, the stronger the wind flows. To model such a velocity distribution, we use one of the models of velocity distribution (see Appendix A).

(3) We assume again that there is a flow in the x direction.

In the voxels whose labels in the x direction are N_x-1 (voxel ($N_x - 1, j, k$), $0 \le j < N_y$, $0 \le k < N_z$), there is the flow at the boundary of the analytical space. We call such a flow the "outflow" (see Figure 5). Since the flow in the subdivided space becomes stable, the differential term of the velocity in the horizontal direction in the voxel ($N_x - 1, j, k$) is zero (i.e. $u_{N_x-1,j,k} = 2u_{N_x-2,j,k} - u_{N_x-3,j,k}$.)

(4) In the top voxel (voxel $(i, j, N_z - 1)$, $0 \le i < N_x$, $0 \le j < N_y$), the velocity in the horizontal direction is constant (i.e. $u_{i,j,N_z-1} = u_{0,j,N_z-1}$, $v_{i,j,N_z-1} = v_{i,0,N_z-1}$).

4.4 The calculation of the pressure term

To calculate the pressure terms, we can directly calculate the variables p and r (see Section 4.1), however, this calculation is unstable. In our method, we assume that the fluid is incompressible, and we calculate pressure terms based on this assumption. To minimize the error, we divide the calculation of the pressure term and the calculation of the other terms in the Reynolds equations. The pressure term can be calculated to solve the Poisson equation derived from the continuity (the preservation of mass)

$$\nabla \cdot \mathbf{u} = 0. \qquad (3)$$

There are several methods of calculating the pressure term, and we apply one of them, which is simple to calculate. First, in each voxel (i, j, k), we calculate the difference $div\mathbf{u}$ between the inflow and outflow in the voxel. Then we calculate the change of the pressure Δp in a time-step. Δp in the voxel (i, j, k) is expressed by:

$$\begin{aligned} del &= \Delta t\{(\frac{2}{\Delta x})^2 + (\frac{2}{\Delta y})^2 + (\frac{2}{\Delta z})^2\}, \\ div\mathbf{u} &= \frac{u_{i,j,k} - u_{i-1,j,k}}{\Delta x} + \frac{v_{i,j,k} - v_{i,j-1,k}}{\Delta y} + \frac{w_{i,j,k} - w_{i,j,k-1}}{\Delta z}, \\ \Delta p &= -\frac{\Omega}{del} div\mathbf{u}, \end{aligned} \qquad (4)$$

where $\Delta x, \Delta y$, and Δz are the voxel sizes in the x, y, and z directions, respectively (see Figure 4(a)). And Ω is a constant. The user can control Ω to accelerate the calculation. After that, we calculate the velocity components by using Δp. We update $u_{i,j,k}$ to $u_{i,j,k} + \frac{\Delta p}{\Delta x}$, and $u_{i-1,j,k}$ to $u_{i-1,j,k} - \frac{\Delta p}{\Delta x}$. We update the other velocity components in the same way. We have to calculate Equation (4) recursively. The calculation terminates when the Δp value of each voxel is smaller than the specified value.

4.5 The boundary layer equation

In Section 4.1, we applied the Reynolds equations as the equations of flow, but these do not always achieve the correct answer, because there are unstable terms in them. Moreover, as we have seen in Section 4.4, to calculate the pressure term recursively in each time step is time-consuming. Hence we tried to use other equations which define the force in the turbulence flow to decrease the unstable terms or the number of iterations necessary.

In most cases smoke flows in the "boundary turbulence layer," which is a very thick layer on the surface of the earth. In the boundary layer, we have to assume that the differential term of the pressure in the horizontal direction, which is defined in Equation (1), is zero. Of course, this condition is not always satisfied, but we can locally assume it to be the case in order to optimize the calculation. To apply equations that represent the flow in this layer, we consider the effects of the random velocity components. But, to calculate these effects, we apply $k - e$ model, and we have to add the following two parameters, that is, turbulence energy k, and range $\overline{t^2}$ of the random component of the temperature. On the surface of the obstacles, these parameters are zero. The equations representing the flow in the turbulence layer are as follows.

$$\begin{aligned}
&\frac{\partial}{\partial x}(uu) + \frac{\partial}{\partial y}(vu) + \frac{\partial}{\partial z}(wu) = \frac{\partial}{\partial z}[(n + n_t)\frac{\partial u}{\partial z}], \\
&\frac{\partial}{\partial x}(uv) + \frac{\partial}{\partial y}(vv) + \frac{\partial}{\partial z}(wv) = \frac{\partial}{\partial z}[(n + n_t)\frac{\partial v}{\partial z}], \\
&\frac{\partial}{\partial x}(uT) + \frac{\partial}{\partial y}(vT) + \frac{\partial}{\partial z}(wT) = \frac{\partial}{\partial z}[(a + a_t)\frac{\partial T}{\partial z}], \\
&\frac{\partial}{\partial x}(uk) + \frac{\partial}{\partial y}(vk) + \frac{\partial}{\partial z}(wk) = \frac{\partial}{\partial z}[(n + \frac{n_t}{1.4})\frac{\partial k}{\partial z}] + n_t[(\frac{\partial u}{\partial z})^2 + (\frac{\partial v}{\partial z})^2] - e, \\
&\frac{\partial}{\partial x}(u\tilde{e}) + \frac{\partial}{\partial y}(v\tilde{e}) + \frac{\partial}{\partial z}(w\tilde{e}) = \\
&\frac{\partial}{\partial z}[(n + n_t)\frac{\partial \tilde{e}}{\partial z}] + 1.45\frac{\tilde{e}}{k}n_t[(\frac{\partial u}{\partial z})^2 + (\frac{\partial v}{\partial z})^2] - 1.9\frac{\tilde{e}}{k}\tilde{e} + E, \\
&\frac{\partial}{\partial x}(u\overline{t^2}) + \frac{\partial}{\partial y}(v\overline{t^2}) + \frac{\partial}{\partial z}(w\overline{t^2}) = \frac{\partial}{\partial z}[(a + a_t)\frac{\partial \overline{t^2}}{\partial z}] + a_t(\frac{\partial T}{\partial z})^2 - 2e_t \qquad (5) \\
&\frac{\partial}{\partial x}(u\tilde{e}_t) + \frac{\partial}{\partial y}(v\tilde{e}_t) + \frac{\partial}{\partial z}(w\tilde{e}_t) = \\
&\frac{\partial}{\partial z}[(a + a_t)\frac{\partial \tilde{e}_t}{\partial z}] + 1.7\frac{\tilde{e}_t}{\overline{t^2}}a_t(\frac{\partial T}{\partial z})^2 + 0.64\frac{\tilde{e}_t}{k}n_t[(\frac{\partial u}{\partial z})^2 + (\frac{\partial v}{\partial z})^2] \\
&- 2\frac{\tilde{e}_t}{\overline{t^2}}\tilde{e}_t - 0.9 f_{D2}\frac{\tilde{e}_t}{\overline{t^2}}\tilde{e}_t + E_t,
\end{aligned}$$

where

$$R_t = 44{,}000, f_2 = 1 - 0.3\exp(-R_t^2)), f_{w1} = [1 - \exp(-\Delta z/30)]^2,$$

$$f_{w2} = [1 - \exp(-\Delta z \boldsymbol{n}^{\frac{1}{3}}/30)]^2, f_{D2} = \frac{1}{0.9}(1.9 f_2 - 1),$$

$$E = \boldsymbol{nn}_t(1 - f_{w1})\{(\frac{\partial^2 u}{\partial z^2})^2 + (\frac{\partial^2 v}{\partial z^2})^2\}, E_t = \boldsymbol{aa}_t(1 - f_{w1})\{(\frac{\partial^2 u}{\partial z^2})^2 + (\frac{\partial^2 v}{\partial z^2})^2\},$$

$$\boldsymbol{e} = \tilde{\boldsymbol{e}} + 2\boldsymbol{n}(\frac{\partial\sqrt{k}}{\partial z})^2.$$

We can represent each of these equations as:

$$\frac{\partial}{\partial x}(uf)+\frac{\partial}{\partial y}(vf)+\frac{\partial}{\partial z}(wf-\Gamma\frac{\partial f}{\partial z})=S_c+S_k f, \qquad (6)$$

where f is the physical variable $(=u,v,T,k,\tilde{e},\overline{t^2},\tilde{e}_t)$, Γ is the coefficient of the effective diffusion, and S_c, S_k are the other terms. To calculate f, we can use the following equation, which is derived from Equations (3) and (6).

$$a_k f_{i,j,k} = a^+ f_{i,j,k+1} + a^- f_{i,j,k-1} + b_k \ , \qquad (7)$$

where

$$a^+ = DA(|Pe^+|) + \max(-w_{i,j,k}\Delta x - w_{i,j,k}\Delta y, 0),$$
$$a^- = DA(|Pe^-|) + \max(w_{i,j,k-1}\Delta x + w_{i,j,k-1}\Delta y, 0),$$
$$a_k = a^+ + a^- + u_{i-1,j,k}\Delta z - S_{i,j,k}\Delta x\Delta z + v_{i,j-1,k}\Delta z - S_{i,j,k}\Delta y\Delta z,$$
$$b_k = S_c\Delta x\Delta z + u_{i-1,j,k} f_{i-1,j,k}\Delta z + S_c\Delta y\Delta z + v_{i,j-1,k} f_{i,j-1,k}\Delta z,$$
$$D = \Gamma_{i,j,k}\Delta x/\Delta z + \Gamma_{i,j,k}\Delta y/\Delta z,$$
$$Pe^+ = w_{i,j,k}\Delta x/D + w_{i,j,k}\Delta y/D,$$
$$Pe^- = w_{i,j,k-1}\Delta x/D + w_{i,j,k}\Delta y/D,$$
$$A(|Pe|) = \max[0,(1-0.1|Pe|)^5].$$

We calculate the physical variables in Equation (7) from the voxels (0, j,k) ($0\le j < N_y$, $0\le k < N_z$) to the voxels (N_x-1, j, k) ($0\le j < N_y$, $0\le k < N_z$).

Equation (7) has some advantages that make it better than the Reynolds equations.

1) The number of the unstable terms is decreased, so the calculation becomes more stable.

2) When we calculate the physical variable $f_{i,j,k}$ of the voxel (i,j,k), the physical variables of the voxel $(i+1,j,k)$ and voxel $(i,j+1,k)$ are not needed, and so we can calculate the equations sequentially.

3) We assume that the pressure in the horizontal direction is constant, therefore we do not need the recursive calculation of the pressure term, which is needed by the Reynolds equations, and we can calculate the velocity field faster.

4.6 The smoke flow algorithm

To calculate the smoke flow, we use the Reynolds equations, which consider the random velocity components. The calculations necessary for one frame of the animation are as follows.

(1) Calculate the distribution of the temperature (i.e. the temperature field). We need it for the calculation of the buoyancy term in the Reynolds equations.

(2) Calculate the velocity field. In this calculation, we perform the following.

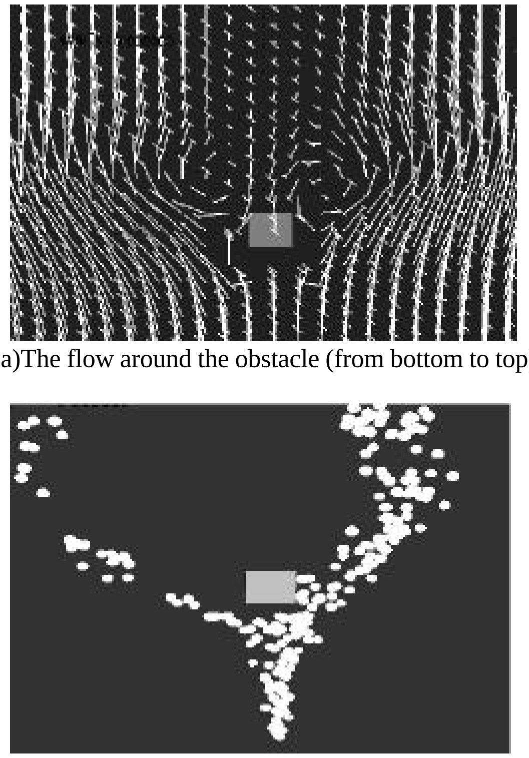
(2.1) Set the value of the inflow, which is one of the boundary conditions (see Section 4.4).

(2.2) Calculate terms in the Reynolds equations except for the pressure term.

(2.3) Calculate the pressure term in the Reynolds equations recursively.

(2.4) Apply the boundary conditions besides the inflow.

(3) Move metaballs according to the calculated velocity field.



(a)The flow around the obstacle (from bottom to top)

(b) Markers' movement in the flow (a)

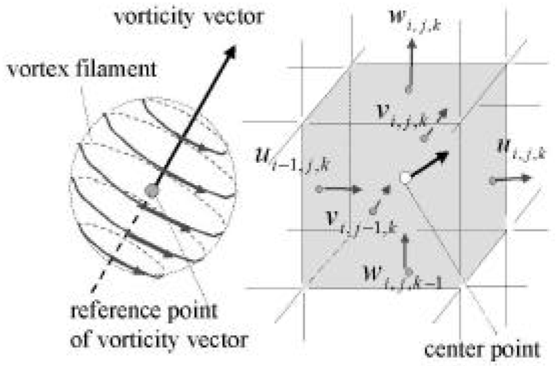
Figure 6: Examples of the calculation of velocity field.

When the field satisfies some conditions described in Section 4.5, we can locally use the boundary layer equations to optimize the calculation. When we use them instead of the Reynolds equations, we don't need the recursive calculation.

By using the proposed method, we can calculate the velocity field as shown in Figure 6(a), and markers in the flow avoid the obstacle as shown in Figure 6(b).

5 Modelling of Vortices Using Vorticity

In turbulent flow, there are many vortices generated by random velocity components. Foster et al. [18] tried to show the swirling smoke flow by using only the velocity field, but to calculate the vortices, we have to subdivide the space into a large number of voxels, which makes the calculation unstable. To model the vortex, there are three parameters that need to be considered, the rotational axis, the rotational velocity, and the convective velocity of the vortex. These three parameters depend on the random velocity components.



(a)Vorticity vector and vortex filaments (b)Vorticity vector at the center of the voxel(*i*,*j*,*k*)

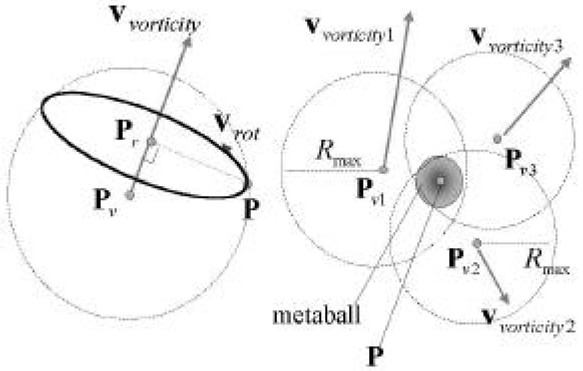
Figure 7: The vorticity vector.

To represent the vortices simply, we use the vorticity vector (see Figure 7(a)). The vorticity vector is a vector that defines the direction of the rotational axis and strength of the rotational velocity of the vortex. The vorticity vector is given by:

$$(\frac{\partial w}{\partial y}-\frac{\partial v}{\partial z},\frac{\partial u}{\partial z}-\frac{\partial w}{\partial x},\frac{\partial v}{\partial x}-\frac{\partial u}{\partial y}). \qquad (8)$$

Each voxel has six reference points of the velocity components on the surface of it (see Figure 7(b)). By using these velocity components, we can easily calculate the vorticity vector at the center of the voxel. And the vortices are defined by the vorticity vector. We call these vortices as vortex filaments. To model the velocity distribution of the vortex filaments, we use the Rankinn' vortex [16].

To calculate the rotational velocity of the vortex, we define a point to refer the vorticity vector, and we call it "the reference point of the vorticity vector". To explain the vorticity vector, we define the positional vector of the reference point as $\mathbf{P}_v$, the vorticity vector at $\mathbf{P}_v$ as $\mathbf{v}_{vorticity}$, and the time-step as Δt (see Figure 8(a)). The rotational axis at $\mathbf{P}_v$ is $\mathbf{v}_{vorticity}$, the direction of the rotational velocity $\mathbf{v}_{rot}$ is perpendicular to $\mathbf{v}_{vorticity}$, and the strength of the rotational velocity is the length of $\mathbf{v}_{vorticity}$. Since we model the velocity distribution of the vortex using the Rankinn vortex, we can calculate the rotational velocity $\mathbf{v}_{rot}$ at the point $\mathbf{P}$ around $\mathbf{P}_v$ as follows:



(a)Rotational velocity $\mathbf{v}_{rot}$ at **P** (b)Reference points around **P**

Figure 8: The calculation of the rotational velocity.

$$\mathbf{P}_r=\mathbf{P}_v+\{(\mathbf{P}-\mathbf{P}_v)\cdot\frac{\mathbf{v}_{vorticity}}{\|\mathbf{v}_{vorticity}\|}\}\mathbf{v}_{vorticity},$$
$$\mathbf{v}_{rot}=\frac{\mathbf{P}-\mathbf{P}_r}{\|\mathbf{P}-\mathbf{P}_r\|}\times\mathbf{v}_{vorticity}\cdot v(\mathbf{P}-\mathbf{P}_v), \qquad (9)$$

where $v(\mathbf{P}-\mathbf{P}_v)$ is a function to represent the velocity distribution of the Rankinn' vortex [16].

The reference points of the vorticity vector are movable, while the other reference points of the physical variables defined in Section 4.1 are fixed. The movement of the reference point of the vorticity vector is as follows. First, we define the initial position of the vorticity vector's reference point randomly within the analytical space. Then, using the vorticity vector at this point, the point is convective in each animation frame (i.e. the positional vector of $\mathbf{P}_v$ at the next frame is $\mathbf{P}_v+\Delta t\cdot\mathbf{v}_{vorticity}$). We can define the vorticity vector $\mathbf{v}_{vorticity}$ at $\mathbf{P}_v$ by using the tri-linear interpolation of the vorticity vector at the center points of the voxels.

We can calculate the swirling motion of the metaball by the rotational velocity $\mathbf{v}_{rot}$. The rotational velocity of the vortex filament at the center of metaball $\mathbf{P}$ is calculated as follows.

(1) Calculate the vorticity vector at the center of the voxel.

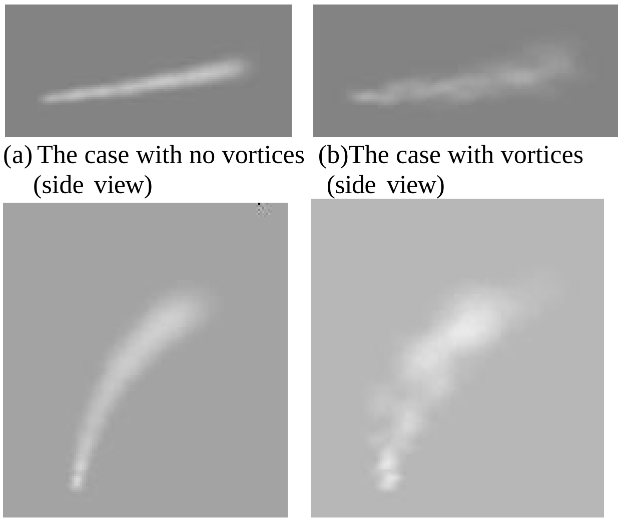
(2) Find reference points which have a smaller distance to $\mathbf{P}$ than given value $R_{\max}$, which is the radius of the Rankinn' vortex. In most case, there are many points which satisfy such a condition (see Figure 8(b)). We define the number of points that satisfy this condition as n, and define the i-th point as $\mathbf{P}_{vi}$.

(3) For each $\mathbf{P}_{vi}$ found in step (2), calculate vorticity vector $\mathbf{v}_{vorticity-i}$ at $\mathbf{P}_{vi}$.

(4) For each $\mathbf{P}_{vi}$, calculate rotational velocity $\mathbf{v}_{rot-i}$.

(5) When the velocity at $\mathbf{P}$ in the calculated velocity field

is $\mathbf{v}_{vf}$, moving the center of the metaball from $\mathbf{P}$ to $\mathbf{P}+\Delta t\cdot(\mathbf{v}_{vf}+\sum_{i=1}^{n}\mathbf{v}_{rot-i})$. i.e. the velocity of the metaball is the sum of the velocity, which the velocity field represents, and the rotational velocity calculated in step (4). Figure 9 shows the comparison of the smoke with and without vorticity.



(a) The case with no vortices (side view)

(b)The case with vortices (side view)

(c)The case with no vortices (top view)

(d)The case with vortices (top view)

Figure 9: The comparison of the vorticity.

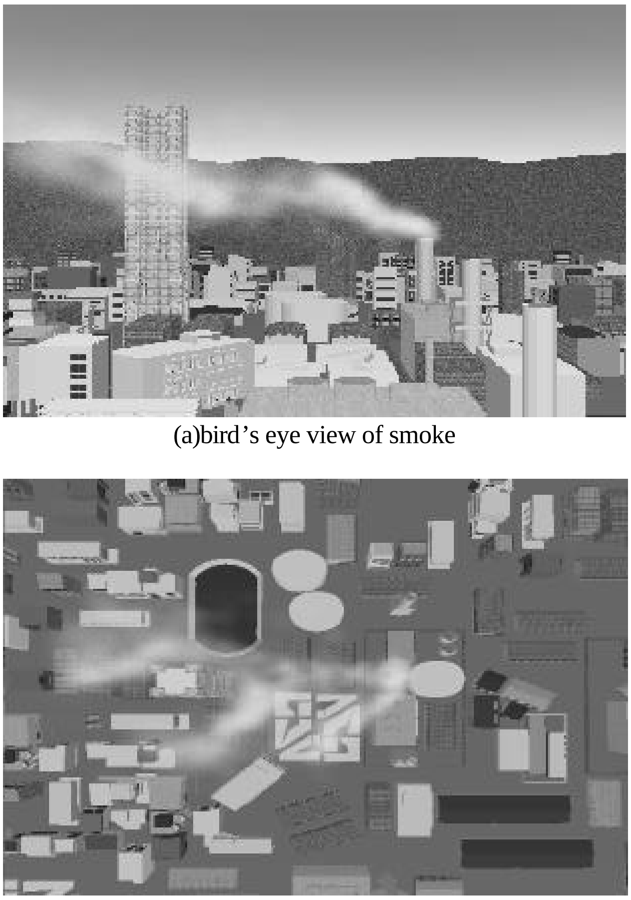
6 Examples

We have variated our method with several examples. To render metaballs, we use Dobashi's method, which uses OpenGL hardware rendering [21]. By using this method, we can render realistic smoke faster than [9].

Figure 10 shows the smoke flow around a tower building. A strong wind flows from the factory to the tower. Also Figures 12(a) and (b) (see color plate) show the smoke in the strong wind. As we can see in Figure 12(b), the smoke flowing to the tower makes a branch. Figures 12(c) and (d) (see color plate) show the smoke flow in the weak wind. They are generated at the same time step, but we can control the color of the smoke by the attenuation coefficient of the smoke particles. Figure 11 and Figure 13 (see color plate) are the smoke from a chimney with a roof. The smoke avoids the chimney's roof. In Figure 11, there is swirling smoke in the large vortices above the chimney's roof.

We calculated these examples on Intergraph TDZ 2000 GX1 (PentiumIII 550MHz). The image size is 720× 480. In Figure 10, the number of voxels is 20×20×20. There are 2,533 metaballs, and the calculation time is 52.735 seconds. In Figure 11, the number of voxels is 40×40×20. The number of metaballs is 3,212, and the calculation time is 57,841 seconds.

As shown in these examples, we have realized realistic smoke animation.



(a)bird's eye view of smoke

(b) top view of smoke

Figure 10: Example (1).

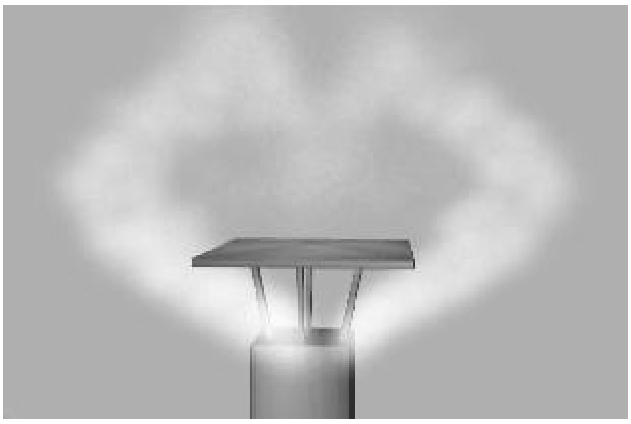


Figure 11: Example (2).

7 Conclusion

In this paper, we have proposed a method of displaying swirling smoke that has a 3-dimensional density distribution, while considering its passage round obstacles. The advantages of the proposed method are as follows.

(1) We deal with smoke as a combination of sets of smoke particles, called puffs. We model the puff by using a metaball. By using metaballs, we can represent a 3-dimensional density distribution for the smoke.

(2) To represent the motion of smoke flow allowing for obstacles, we use a velocity field method. The velocity field is generated by solving physical equations of flow, the Reynolds equations. This takes into account random velocity in the turbulence. The result of the model using these equations is a realistic flow that takes obstacles into consideration. To optimize the calculation of the velocity field, we can locally use the boundary turbulence layer equations instead of the Reynolds equations.

(3) The small vortices in the flow are generated from the vorticity vector and vortex filaments. The vorticity vector represents the strength and direction of the vortices, and we can calculate the vorticity vector simply. The vortices in the turbulence are dealt as vortex filaments.

References

[1] W. T. Reeves, “Particle systems. A Technique for Modeling a Class of Fuzzy Objects,” Proc. of SIGGRAPH’83, 1983, pp.359-376.

[2] D.S. Ebert, and W.E. Carlson, and R.E. Parent, “Solid Spacesand Inverse Particle Systems for Controlling the Animation of Gases and Fluids,” The Visual Computer, Vol.10, 1994, pp.179-190.

[3] W.T. Reeves, and R. Blau, “Approximate and Probabilistic Algorithms for Shading and Rendering Structed Particle Systems, ” Proc. of SIGGRAPH’85, 1985, pp.313-322.

[4] D.S. Ebert, F.K. Musgrave, P. Peachey, K. Perlin, and S. Worley, “Texturing and Modelling: A Procedual Approach,” AP Professional, 1994, pp. 267-294.

[5] Y. Gardner, “Modelling Amorphous Natural Features,” Course Notes of SIGGRAPH’94, 1994, pp.1-24.

[6] T. Nishita, “A Shading Model for Atmospheric Scattering Considering Luminous Intensity Distribution of Light Sources,” Proc. of SIGGRAPH’87, 1987, pp.303-310.

[7] J.T. Kajiya, and B.P.V. Herzen, “Ray Tracing Volume Densities,” Proc. of SIGGRAPH ‘84, 1984, pp.165-174.

[8] G. Sakas, “Modelling and Animating Turbulent Gaseous Phenomena Using Spectral Synthesis,” The Visual Computer, Vol.9, 1993, pp.200-212.

[9] J. Stam and E. Fiume, "Turbulent Wind Fields for Gaseous Phenomena", Proc. of SIGGRAPH ’93, 1993, pp.369-376.

[10] M. Shinya and A. Fournier, "Stochastic Motion-Motion Under the Influence of Wind", Proc. of Eurographics ’92, 1992, pp.119-128.

[11] H.Tennekes, J.L.Lumley, “A First Course in Turbulence,” The MIT Press, 1972.

[12] G.K.Batchelor, “An Introduction to Fluid Dynamics,” Cambridge At The University Press, 1967.

[13] J. Wejchert, and D. Haumann, “Animation Aerodynamics,” Proc. of SIGGRAPH ’91, 1991, pp.19-22.

[14] K. Sims, “Particle Animation and Rendering Using Data Parallel Computation,” Proc. of SIGGRAPH ’90, 1990, pp.405-413.

[15] L. Yaeger, C. Upson, R. Myers, “Combining Physical and Visual Simulation – Creation of the Planet Jupiter for the Film “2010”,” Proc. of SIGGRAPH’86, 1986, pp.85-93.

[16] J.X. Chen, N. da. Vittoria Lobo, C. E. Hughes, and J. M. Moshell, “Real-Time Fluid Simulation in a Dynamic Visual Environment,” IEEE Computer Graphics and Applications, Vol.3, 1997, pp.52-61.

[17] N. Chiba, S. Ohkawa, K. Mureoka, and M. Miura, “Two -dimensional Simulation of Flames, Smoke and the Spread of Fire,” The Journal of Visualization And Computing Animation, Vol.5, 1994, pp.37-54.

[18] N. Foster, D. Metaxas “Modelling the Motion of a Hot, Turbulent Gas,” Proc. of SIGGRAPH’97, 1997, pp.181-188.

[19] J. Stam, ”Stable Fluids,” Proc. of SIGGRAPH’99, 1999, pp.121-128.

[20] G. Wyvill and A. Trotman, “Ray-Tracing Soft Objects,” Proc. of CG International, 1990, pp.439-475.

[21] Y. Dobashi, K. Kaneda, H. Hamashima, T. Okita, T. Nishita, “A Simple, Efficient Method for Realistic Animation of Clouds,” Proc. of SIGGRAPH2000, pp.19-28.

APPENDIX

A. Model of Inflow

In the turbulence, the velocity close to the ground is zero, and the higher we go up, the stronger the wind flows. We can set such a flow $u_{0,j,k}$ in the voxel (i,j,k) by using the following equation.

$$u_{0,j,k} = \frac{3}{1-\frac{S}{N_z}} \frac{k\Delta z - \frac{S}{N_z}}{1-\frac{S}{N_z}} \left(1 - \frac{1}{2} \frac{k\Delta z - \frac{S}{N_z}}{1-\frac{S}{N_z}}\right),$$

where S is the height between the voxel which represents the surface of the obstacle and the bottom layer (see Figure 5), N_z is the number of the subdivision in the vertical direction, and Δz is the size of the voxel (see Figure 4(a)).