Real-Time Visual Animation of Explosions

Wenhui Zhang[1], Feng Guo[1], Zhian Lin[1], Yanhao Zhang[1], Jiming Lin[2*], Xinxiang Wei[1]

[1] Guangxi Colleges and Universities Key Laboratory of Intelligent Processing of Computer Images and Graphics, Guilin University of Electronic Technology, Guilin, 541004, China.
[2] Guangxi Experiment Center of Information Science, Guilin University of Electronic Technology, Guilin, 541004, China.

* Corresponding author. Email: linjm@guet.edu.cn.
Manuscript submitted November 11, 2014; accepted January 4, 2015.
doi: 10.17706/jsw.10.3.331-343

Abstract: Realistic explosion effects can enhance the realism and immersion in virtual battlefield. The real-time explosion effects simulation of the explosion products and their interactions, such as explosive fireball, explosive smoke and explosive ash, will be exploded into two phases: explosions and fuel modeling. The explosions modeling describe the formation and spreading of the explosive fireball. The fuel modeling depicts rising of the explosive ash accompanied by the explosive smoke. The paper describes an Eulerian approach for animating explosions in the first phase. The divergence-free velocity field is computed on a grid. In order to ensure the quality of conservation, the velocity divergence in the local is joined to describe smoke diffusion. The particles with the respective attributes are updated in both two phases. Simplified Navier-Stokes (NS) equations modified according to the non-conservation of explosion energy is used to determine the fluid motion in whole simulation process. In order to further improve the fidelity and real-time of the simulation effect, small-scale swirl effect of explosion smoke with vorticity confinement is introduced to compensate for the loss of finite difference methods to solve the NS equations. The whole simulation process of explosion effects are realized on Visual Studio 2010 and OpenGL platform, including multiple explosions, explosions near obstacles. The temperature and velocity in the explosion center, explosive diameter and height, are analyzed to verify the high fidelity of this new simulation method. The high fidelity and computational efficiency of this new modified approach are verified with several experimental simulation results.

Key words: Fluid simulation, NS equations, particle system, virtual battlefield, vorticity confinement.

1. Introduction

Visual explosion models have many obvious applications in the computer and mobile games industry including special effects and interactive games. Generally, a chemical explosion comprise of severely combustion of the explosive materials and a sudden release of energy. The explosion's effect include not only the blast wave, explosive fireball, explosive smoke, explosive ash and other effects, but also a huge sound phenomenon. Explosion process is typically divided into two stages: the first stage is the formation of explosive fireball which is produced by burning explosion materials and increase its explosive radius rapidly ranged from several to several dozen times. The second stage is the diffusion and rising process of the explosive smoke and the accompanied explosive ash. A particle system proposed by Liang *et al*. [1] can simulate the whole process of explosion on the ground and under the water, and texture technology are used to simulate the explosion fireball. This method is simple but lack of physical realism. Feldman *et al*. [2] combined the fluid model and particle model to describe the fireball, but the following smoke have not kidney shaped structure and only the gravity is included in internal force which cannot accurately describe the vortex motion of smoke particles effected by

thermal buoyancy. Brunelli and Marchioro [3] simulated the small-scale swirl effect of the explosive smoke, which is a uniquely two-dimensional characteristic feature of smoke. Tang *et al*. [4] improved the confinement method of vorticity. They added an adaptive vorticity confinement into the external force in Navier-Stokes (NS) equations, then used B-spline interpolation to pre-sample the velocity field, finally integrated the pre-sampling velocity field and the adaptive vorticity confinement into NS equations. High resolution swirl details and enhanced realism can be obtained after smoothing and de-noising with this method. Chen *et al*. [5] and Gao *et al*. [6] simulated smoke near obstacles by setting differential boundary conditions in the fluid space. The experimental results show that smoke will not pass through obstacles but move along the boundary, which was in well agreement with the actual situation.

Explosion of explosive materials and smoke movement will be combined in this paper to implement a real-time visual animation of whole chemical explosion process. The Eulerian model and the particle model are used to simulate all the transformations and interactions between explosive fireball, smoke and ash after quickly combustion of the explosive materials. The particle model describes numerical renewal calculation of each physical quantity in both two phases. The Eulerian model clearly determines the interactions between various explosive components in second phase. The particle system in [2] is used to model the explosion, the smoke particle density is set as Gaussian random distribution, and, the flow velocity of smoke particles is modeled by Navier-Stokes (NS) equations which are modified with addition of velocity divergence according to the non-conservation of explosion energy. We discrete the computational domain into spatially homogeneous cubic lattices and finite difference method is used to numerically solve the modified NS equations. Formation and diffusion of the explosive smoke is simulated by the velocity divergence. Finally, in order to enhance the small-scale swirl effect of explosive smoke, the vorticity confinement in [3] is used to compensate for the numerical loss owing to discretization error of the finite difference method. GPU acceleration techniques such as display list and depth sorting are also used to accelerate the rendering speed in the algorithm. The tests were carried out on Microsoft Visual Studio 2010 and OpenGL platform. The experimental results show that the algorithm is simple and effective to achieve real-time, realistic explosions, and is adapt to complex scenes with complex boundary conditions and obstacles.

2. Fluid Model for Explosion

Navier-Stokes (NS) equations are used to determine the motion of the fluid in computational fluid dynamics

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u}\cdot\nabla)\mathbf{u} + \upsilon\nabla^2\mathbf{u} - \nabla P\big/\rho + \mathbf{F} \tag{1}$$

$$\frac{\partial \rho}{\partial t} + \rho\nabla\cdot\mathbf{u} = 0 \tag{2}$$

where (1) describes the momentum conservation, (2) states the mass conservation. $\mathbf{u}$ is the fluid velocity vector, υ the viscosity, ρ the fluid density, P pressure, $\mathbf{F}$ any external forces acting on the fluid, and ∇ denotes the gradient operator. In two-dimensional Cartesian coordinate system, $\nabla = \left(\frac{\partial}{\partial x} \quad \frac{\partial}{\partial y}\right)$.

2.1. Eulerian Model

According to the explosion theory, the instant gaseous fluid speed of explosion is well below the speed of sound then the compressibility effects of gas are negligible in this simulation algorithm. Consequently, ρ is constant and can be normalized to 1 for dimensionless simulation. Gaseous fluid viscosity coefficient is very small and can be neglected in explosion simulation, $\upsilon \approx 0$. Therefore (1) can be simplified as Eulerian Model

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u}\cdot\nabla)\mathbf{u} - \nabla P + \mathbf{F} \tag{3}$$

where $-(\mathbf{u}\cdot\nabla)\mathbf{u}$ is the advection term, $-\nabla P$ the diffusion term, $\mathbf{F}$ is external force term.

Gaseous fluid density is constant, i.e. $\partial\rho/\partial t = 0$, then (2) can be simplified further as

$$\nabla\cdot\mathbf{u} = 0 \tag{4}$$

In the process of blasting, the explosive particle obtains the combustion energy and transforms to gaseous fluid, which makes the velocity divergence of the corresponding fluid grid increases, greater than zero. So we slightly modify the mass conservation (4) by adding a velocity divergence to reflect function of the external injected energy come from the combustion [2]

$$\nabla\cdot\mathbf{u} = \varphi \tag{5}$$

The velocity divergence of fluid grid φ mainly effects the pressure term for solution of the mass and momentum conservation equations.

So the gaseous fluid model of explosion is determined by (3) and (5) in this algorithm used in this paper.

2.2. Velocity Update in Fluid Model

In order to solve the gaseous fluid model of the explosion, the simulated fluid space is discretized into structured uniform micro-grid. The projection of the discretized space into XOY Cartesian coordinate plane is two-dimensional grid plane as shown in Fig. 1, where i and j identify the number of fluid grid, grid nodes $[i,j]$, $[i,j+1]$, $[i,j-1]$ and $[i-1,j]$ define a scalar grid. The scalar parameters of the fluid grid, such as temperature T and velocity divergence φ, are stored and calculated in the grid nodes. A vector grid plane is defined with staggered grid arrangement, which is half-interval away from the scalar grid. Vector parameters including vertical velocity u_y and horizontal speed u_x of fluid grid are stored and calculated in the second grid..

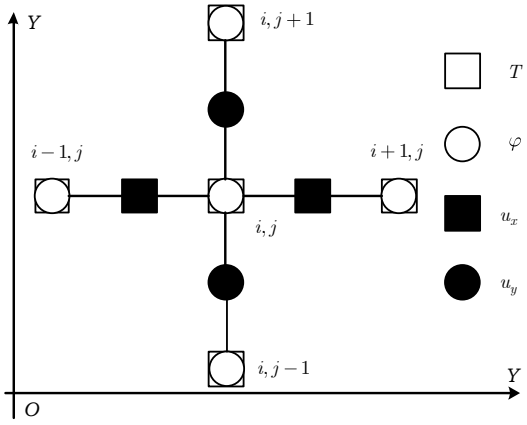

Fig. 1. Definition of the discretized fluid grid and its parameters.

The velocity of each grid in the staggered grid system is calculated with finite difference method, which improves the accuracy of numerical methods. Explosion energy governs the renewal of the temperature and velocity divergence. The velocity update must be satisfied the momentum conservation (3), but the numerical dissipation introduced by finite difference method and practical numerical computation will cause the calculation results does not satisfy the mass conservation. We add the vorticity confinement and the iterative projection to the momentum equations to reduce the numerical dissipation. The calculation of the momentum conservation equation is done in three steps. First, we add the external force term $\mathbf{F}$ to the velocity grid in (3) with neglection of advection and diffusion terms. Next we solve the advection term $-(\mathbf{u}\cdot\nabla)\mathbf{u}$ in (3) by building a new velocity grid from the ones already computed in first step. Finally, we consider the diffusion term $-\nabla P$

in (3) with the similar method and obtain the final horizontal and vertical velocities.

2.2.1. Velocity update with the external force term only

Neglecting the advection and diffusion terms in (3), we can obtain the differential equation

$$\frac{\mathbf{u}(t+\Delta t)-\mathbf{u}(t)}{\Delta t}=\mathbf{F} \tag{6}$$

where we advance our simulation by updating grids over a fixed time step Δt. So we can compute an intermediate velocity field with finite difference method for the fluid mesh,

$$\begin{cases} u_{Fx}[i,j]=u_x[i,j]+F_x[i,j]\cdot\Delta t \\ u_{Fy}[i,j]=u_y[i,j]+F_y[i,j]\cdot\Delta t \end{cases} \tag{7}$$

where $u_x[i,j]$ and $u_y[i,j]$ are the initial horizontal and vertical velocities of the fluid mesh node $[i,j]$ respectively, $F_x[i,j]$ and $F_y[i,j]$ are the horizontal and vertical external force component respectively, $u_{Fx}[i,j]$ and $u_{Fy}[i,j]$ are the corresponding intermediate horizontal and vertical velocities influenced by external force.

The external forces $\mathbf{F}$ include gravity G, thermal buoyancy F_b and the vortex force $\mathbf{F}_v$ used to compensate the numerical dissipation in the finite difference equation. The buoyancy F_b and gravity G are a pair of opposite forces in Y direction, which affect only the vertical velocity $u_y[i,j]$ of the fluid grid. Heavy and cold smoke tends to fall downwards due to gravity while light and hot smoke tends to rise due to buoyancy. Therefore, the sum of buoyancy and gravity is directly proportional to the temperature

$$F_{bg}=F_b-G=-\beta g(T_0-T_k) \tag{8}$$

where g is the acceleration of gravity, T_0 is the ambient temperature of the air, T_k is the temperature of the current grid, $T_k=\left(T[i,j]+T[i,j+1]\right)\big/2$, which is the temperature interpolation of the current grid node and the next grid node. Thermodynamic coefficient β is a positive constant.

From (7), the finite difference calculation for fluid velocity under the action of F_{bg} is given by

$$\begin{cases} u_{bx}[i,j]=u_x[i,j] \\ u_{by}[i,j]=u_y[i,j]+F_{bg}\left[i,j\right]\Delta t \end{cases} \tag{8}$$

where $u_x[i,j]$ and $u_y[i,j]$ are horizontal and vertical velocities of the original fluid grid, $u_{bx}[i,j]$ and $u_{by}[i,j]$ are horizontal and vertical velocities of fluid grid when the gravity and buoyancy are taken into account.

Our solver for the fluid velocity in consideration of vortex force term [7] is done in three steps. First, the vorticity density $\omega[i,j]$ at node $[i,j]$ is calculated with bilinear interpolation of the horizontal and vertical velocities,

$$\omega[i,j]=\left|\frac{u_{by}[i,j+1]-u_{by}[i,j-1]}{2}-\frac{u_{bx}[i+1,j]-u_{bx}[i-1,j]}{2}\right| \tag{10}$$

Then, the external force obtained by linear interpolation of the vorticity density

$$\begin{aligned}\omega_{\text{in}} \equiv\ & \omega[i+1,j]-\omega[i-1,j]^{2} \\ & +\ \omega[i,j+1]-\omega[i,j-1]^{2} \\ F_{vx}[i,j] &= \varepsilon\cdot\omega[i,j]\frac{\omega[i+1,j]-\omega[i-1,j]}{\sqrt{\omega_{\text{in}}}} \\ F_{vy}[i,j] &= \varepsilon\cdot\omega[i,j]\frac{\omega[i,j+1]-\omega[i,j-1]}{\sqrt{\omega_{\text{in}}}}\end{aligned} \tag{11}$$

where ε is the vorticity confinement coefficient, $F_{vx}[i,j]$ and $F_{vy}[i,j]$ are the horizontal and vertical components of the vortex force $\mathbf{F}_v$ at node $[i,j]$ respectively. Finally, we obtain an intermediate velocity vector $\mathbf{u}_F$ under the action of external force $\mathbf{F}$ by substituting the vortex force given in the (11) and velocity vector defined by (9) into (7)

$$\begin{cases}u_{Fx}[i,j]=u_{bx}[i,j]+F_{vx}[i,j]\cdot\Delta t \\ u_{Fy}[i,j]=u_{by}[i,j]+F_{vy}[i,j]\cdot\Delta t\end{cases} \tag{12}$$

2.2.2. Velocity renewal with addition of the advection term

According to the Euler method, (3) can be expressed as a differential form of the velocity field in each fluid grid when consider advection term only

$$\begin{aligned}\mathbf{u}(t+\Delta t) &= \mathbf{u}(t)-(\mathbf{u}\cdot\nabla)\mathbf{u}\cdot\Delta t \\ &= \mathbf{u}(t)+\mathbf{a}\cdot\Delta t\end{aligned} \tag{13}$$

where $\mathbf{a}=-(\mathbf{u}\cdot\nabla)\mathbf{u}$ is the advection term. $\mathbf{u}(t+\Delta t)$, $\mathbf{u}(t)$ are velocity vectors of fluid grid at time $t+\Delta t$ and t respectively.

Equation (13) is numerical solved by finite difference method

$$\begin{aligned}u_{ax}[i,j] = u_{Fx}[i,j]+\ & 4u_{Fx}[i,j]-u_{Fx}[i-1,j] \\ & -u_{Fx}[i+1,j]-u_{Fx}[i,j-1] \\ & -u_{Fx}[i,j+1]\ \ \Delta/4 \\ u_{ay}[i,j] = u_{Fy}[i,j]+\ & 4u_{Fy}[i,j]-u_{Fy}[i-1,j] \\ & -u_{Fy}[i+1,j]-u_{Fy}[i,j-1] \\ & -u_{Fy}[i,j+1]\ \ \Delta/4\end{aligned} \tag{14}$$

where $u_{ax}[i,j]$, $u_{ay}[i,j]$ are horizontal velocity and vertical velocity in grid $[i,j]$ after adding the advection term.

2.2.3. Pressure projection renewal

The pressure in explosive materials is equal everywhere before the explosion. When the explosion occurres, pressure in the explosion area increases rapidly to a maximum value and then explosion smoke diffuse to the surrounding. When pressure in the explosion area is slightly less than the pressure surrounding, explosion smoke will concentrate slightly. When the pressure returns to stable equilibrium state, the explosion is finished. In this algorithm, we add a velocity divergence to velocity vector in the grid before calculation of each diffusion term to represents velocity changes after energy injection.

First, velocity vector field $\mathbf{u}=(u_{dx},u_{dy})$ in the fluid mesh can be decomposed into sum of the velocity vector

field $\mathbf{u}_a = (u_{ax}, u_{ay})$ given by (14) and the gradient of the pressure $-\nabla P$ according to the Helmholtz-Hodge decomposition rule [8],

$$\mathbf{u} = \mathbf{u}_a - \nabla P \tag{15}$$

Applying divergence to both sides of (15), this leads to

$$\nabla \cdot \mathbf{u} = \nabla \cdot \mathbf{u}_a - \nabla^2 P \tag{16}$$

Substituting (5) into (16) yields a slightly modified version of pressure Poisson equation with injection of the velocity divergence

$$\nabla^2 P = \nabla \cdot \mathbf{u}_a - \varphi \tag{17}$$

After solving the pressure from (17), the velocity vector updates according to (15).

According to the analysis above, the numerical updating process about the velocity considering the impact of pressure is as follows. First of all, the initial pressure values $P^{(1)}[i, j]$ of fluid grid $[i, j]$ is determined by

$$\begin{aligned} P^{(1)}[i,j] = \frac{1}{2} \; & u_{ax}[i+1,j] - u_{ax}[i-1,j] + \\ & u_{ay}[i,j+1] - u_{ay}[i,j-1] \; - \varphi[i,j] \end{aligned} \tag{18}$$

Then, discretize (16) and solve it by Jacobi iterative method

$$\begin{aligned} P^{(k+1)}[i,j] = & -(\delta x)^2 \nabla \cdot \; u_{ax}[i,j], \, u_{ay}[i,j] \\ & + \frac{P^{(k)}[i+1,j] + P^{(k)}[i-1,j]}{4} \\ & + \frac{P^{(k)}[i,j+1] + P^{(k)}[i,j-1]}{4} \end{aligned} \tag{19}$$

where $P^{(k+1)}[i, j]$ is the pressure value of fluid mesh $[i, j]$ in the k-th iteration, δx is the horizontal distance of the adjacent fluid grid.

After N iterations (19) achieve accurate results, the final velocity vector can be calculated with linear interpolation,

$$\begin{aligned} u_{dx}[i,j] = u_{ax}[i,j] - \; & P^{(N+1)}[i+1,j] \\ & - P^{(N+1)}[i-1,j] \; \frac{\Delta t}{2} \\ u_{dy}[i,j] = u_{ay}[i,j] - \; & P^{(N+1)}[i,j+1] \\ & - P^{(N+1)}[i,j-1] \; \frac{\Delta t}{2} \end{aligned} \tag{20}$$

where $u_{dx}[i, j]$ and $u_{dy}[i, j]$ are the final horizontal and vertical velocities in fluid grid respectively.

2.3. Temperature Update

Before the explosion, particle temperature inside reaction zone of explosive materials and fluid temperature are equal to ambient temperature. When the explosive materials are igniting, temperature of reaction zone is increased to a temperature above the autoignition point, explosion particle begin to burning, generate high temperature deliver to neighboring particle and make it burning quickly too. So explosion is start with the generation of high temperature and the release of energy. The change of temperature is affected by thermal

convection, natural cooling of fluid, diffusion, and the heat energy transferred into the fluid from an external source.

The temperature T in fluid grid evolves according to [2]

$$\begin{aligned} T^{n+1}[i,j] = \Delta t \Bigg\{ & \frac{4T^{n}[i,j] - T^{n}[i-1,j]}{4} \\ & - \frac{T^{n}[i+1,j] + T^{n}[i,j-1]}{4} \Bigg\} \\ & - c_r \frac{2T_0 - T^{n}[i,j] + T^{n}[i,j+1]}{2T_{\max} - T^{n}[i,j] + T^{n}[i,j+1]} \\ & + c_v H + c_k \ T^{n}[i-1,j] + T^{n}[i+1,j] \\ & + T^{n}[i,j-1] + T^{n}[i,j+1] \ + T^{n}[i,j] \end{aligned} \tag{21}$$

where T_0 denotes ambient temperature, $T_{\max}$ the maximum temperature of the system, $T^{(n)}[i,j]$, $T^{(n+1)}[i,j]$ temperatures at the node $[i,j]$ of the n-th and $n+1$-th iterations, H the heat increment injected into fluid grid, c_r, c_k, c_v the constant adjustment coefficients. The first four right-hand terms in (21) represent respectively the advection by the fluid, the natural cooling of fluid, diffusion, heat energy transferred into the fluid from an external source.

2.4. Update of the Velocity Divergence

The velocity divergence reflects the local mass conservation of fluid grid. The energy of combustion particle propagates to the adjacent fluid grid and increases the temperature at local fluid grid. The renewal for the velocity divergence is given by

$$\varphi = b_g u_{bs} \cdot \Delta t \tag{22}$$

Where u_{bs} is the speed of the particle combustion, b_g the adjustment factor, determined by the experiment.

3. Particle Model

We model the combustion products using a particle system. Particle descriptions consist of position, mass, velocity, type identifier and the renewal of all the aforementioned parameters.

3.1. Update for the Particle Mass

The particles express three states at differential phase of explosion: stable, combustion and ash particle. Before the explosion, all particles are stable, namely the normal state of unburned particle. When some stable particles are heated, the particle temperature is rise. When the temperature reaches autoignition point, stable particles start rapid combustion to explode, and transform from stable state to combustion state, becoming the combustion particles. Mass of combustion particles gradually reduced and the particles progressively transfer to ash state because of the combustion. Mass of the ash particles increases gradually with time. When the mass of combustion particle decreases to zero, all the particles transfer to ash particles. The mass update process for stable particles and combustion particles can be expressed as

$$\text{mass}^{(k+1)} = \text{mass}^{(k)} - b_z u_{bs} \cdot \Delta t \tag{23}$$

where $\text{mass}^{(k)}$ and $\text{mass}^{(k+1)}$ are the particle mass at present and the next moment, b_z the adjustment coefficient determined by the experiment.

The mass renewal for the ash particles is defined as

$$\text{mash}^{(k+1)} = \text{mash}^{(k)} + b_a u_{bs} \cdot \Delta t \tag{24}$$

where $\text{mash}^{(k)}$ and $\text{mash}^{(k+1)}$ are the mass of ash particles at present and next moment, b_a the adjustment coefficient.

3.2. Simulation of the Particle Velocity

Here, simple, fast Euler method is used to update the velocity of each particle in the system, which depends on gravity and motion inside the fluid.

The velocity changes of each particle caused by gravity is

$$\begin{cases} u_{gx} = u_{gx0} \\ u_{gy} = u_{gy0} - g \cdot \Delta t \end{cases} \tag{25}$$

where u_{gx0}, u_{gy0} and u_{gx}, u_{gy} are the horizontal and vertical velocities of particle before and after consideration of gravity.

The motion and force inside the fluid affect the particle velocity, it will be specifically explained in detail in next section.

3.3. Simulation of the Particle Position

Particle position update equation is

$$\begin{cases} x^{(k+1)} = x^{(k)} + u_{px} \cdot \Delta t \\ y^{(k+1)} = y^{(k)} + u_{py} \cdot \Delta t \end{cases} \tag{26}$$

where $x^{(k)}$, $y^{(k)}$ and $x^{(k+1)}$, $y^{(k+1)}$ are horizontal position and vertical position of a particle at present and next moment, u_{px}, u_{py} renewal horizontal and vertical velocities.

4. Interaction between Particle and Fluid Models

The particle and fluid models interact with each other through the transfer of momentum and heat energy. The combustion of explosive particle usually generates a lot of heat energy injected into the fluid grid. Temperature of the fluid grid increases and spreads around, finally the adjacent explosive particles are burning too, whole explosion process complete. The interaction between particle and fluid models is composed with two parts: one is that the explosion particle combustion wields considerable influence on the fluid grid properties, another is the impact that the internal motion of the fluid grid has on the explosion particle velocity.

4.1. Influence on the Attributes of Fluid Mesh

The combustion of explosion particle has an effect on the velocity divergence and temperature of the fluid mesh. When the fluid grid temperature rises, heat spreads around to make the surrounding grid temperature rise too, and makes the neighboring particles of the fluid grid start burning. The update for heat increment H in (21) is

$$H = b_h u_{bs} \cdot \Delta t \tag{27}$$

where b_h is adjustment coefficient determined by the experiment.

4.2. Effects of Fluid Movement Inside the Grid

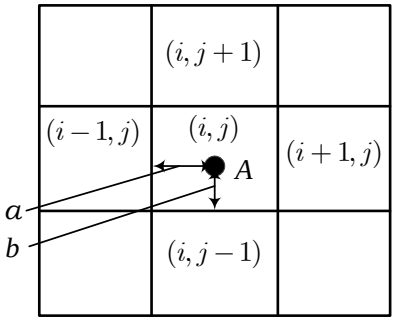

Fig. 2. Position arrangement of the particles in the fluid grid.

The particle motion inside the fluid grid will affect particle velocity. In Fig. 2 the position of particles in the fluid grid is marked A, where the grid unit is 1. a and b represent the horizontal and vertical distances from the lower left corner respectively. The effects of advection term on the particles can be distinguished as ash and non-ash cases.

The velocity of ash particle affected by gravity only is given by

$$\begin{aligned} u_{px} &= (1-a)(1-b)u_{gx}[i+1,j] + abu_{gx}[i-1,j] \\ &\quad + a(1-b)u_{gx}[i,j+1] + (1-a)bu_{gx}[i,j-1] \\ u_{py} &= (1-a)(1-b)u_{gy}[i+1,j] + abu_{gy}[i-1,j] \\ &\quad + a(1-b)u_{gy}[i,j+1] + (1-a)bu_{gy}[i,j-1] \end{aligned} \tag{28}$$

The velocity of non-ash particle is related to the fluid velocity and its own speed. The horizontal velocity increment u'_{px} and vertical velocity increment u'_{py} affected by particle fluid are

$$\begin{aligned} u'_{px} &= (1-a)(1-b)u_{dx}[i+1,j] + a(1-b)u_{dx}[i,j+1] \\ &\quad + abu_{dx}[i-1,j] + (1-a)bu_{dx}[i,j-1] \\ u'_{py} &= (1-a)(1-b)u_{dy}[i+1,j] + a(1-b)u_{dy}[i,j+1] \\ &\quad + abu_{dy}[i-1,j] + (1-a)bu_{dy}[i,j-1] \end{aligned} \tag{29}$$

Then the particle velocity of non-ash particle is

$$\begin{cases} u_{px} = u_{gx} + \eta \dfrac{1}{\text{mass}^{(k)}} r^2 u'_{px} \Delta t \\ u_{py} = u_{gy} + \eta \dfrac{1}{\text{mass}^{(k)}} r^2 u'_{py} \Delta t \end{cases} \tag{30}$$

where η the constant coefficients, $\text{mass}^{(k)}$ the particle mass, r the particles radius.

5. Experimental Results and Discussion

All numerical simulations are performed on a PC platform with an Intel Core Duo2 T6500 CPU, 2 GB memory, and GeForce FX5700 video graphics card. In order to run the calculation with 32-bit floating point precision, the OPENGL extension and Visual Studio 2010 development tools are used in the experimental simulations.

5.1. Simulation of Explosion Process

5.1.1. Independent explosion

Fig. 3. The simulated example of the whole process of independent explosion.

A simulated example of the whole explosion process is illustrated in Fig. 3. The first one shows that the explosion is initialized in the scene. The second one shows that the explosive materials are lighted up, explosive materials start burning to produce fireball. The temperature in middle of the fireball is the highest and the color becomes yellow. The fireball expands larger and rises up. Thirdly, due to the inconsistent density of the explosive products, diffusion rate of the fireball is not the same at anywhere, a smoke plume is gradually formed in the lower part of the fireball. Fourthly explosives materials are completely burned out and explosive fireball is disappeared. Explosive fireball fully transfers into explosive smoke and rises up to shape as a vortex. The vortex ring rolls from inside to outside around the vortex core that forms the familiar mushroom cloud. The diameter of vortex ring gradually stretches stably until reach to the top boundary. Then, the explosion smoke is progressively dissipated and the explosion ash goes down due to the gravity. In this simulation example, the generation and rising process of the vortex in the explosion smoke can be clearly observed in Fig. 3. A kidney shaped pattern in the upper cross-section of smoke flow shown in Fig. 3 is confirmed by many experimental observations done by other researchers. This kidney shape is composed by two vortexes with anti-circulation. Vortex structure causes a velocity field move upward, it leads to stretching and distortion of the gas blobs surface.

5.1.2. Simulation results of the continuous explosion

Fig. 4. Simulated example of the continuous explosion.

The example shown in Fig. 4 demonstrates a two-dimensional continuous explosion. Two combustible particles clusters are initialized in this scene. The left-side particles clusters reach the autoignition and begin to explode after injection of energy. The heat energy produced by left-side explosive particles spread from left to right, and leads to explosion of the right-hand particles cluster. Then, right-hand explosive smoke and left-hand explosive smoke interact with each other.

5.1.3. Explosion with a set of obstacles

Fig. 5. Explosion example with a relatively open rectangular obstacles surrounds the source.

In Fig. 5, there is a relatively open set of obstacles surrounds the explosive source in two-dimensional space. After the explosion, explosive smoke is spread and bypasses the obstacles. The vortex effect can be clearly observed in Fig. 5, suggesting that the smoke movement model in this algorithm is consistent with physical reality, and do not go through obstacles or disappear, but moves according to the true physics laws when encountering an obstacle.

5.1.4. Explosion with the horizontal wind field

Fig. 6. The explosive smoke distribution with horizontal wind field.

Table 1. Simulation Parameters

Size of grid	Number of particles	Average frame rate(FPS)
128×128	16654	46
256×256	16654	27
$32\times32\times32$	6184	26

In Fig. 6, the morphological changes of the explosion affected by the horizontal wind field are illustrated. It can be see that the horizontal wind field has seriously impact on the explosive smoke.

5.2. Analysis of the Explosive Parameters

In this simulations, the number of particles and grid size have direct impact on the real-time performance of the system. Those data used in different simulation environments is shown in Table 1. The average frame rate is up to 46FPS in the two-dimensional simulation and 26FPS in three dimensional simulation. Augmented fidelity and real-time interaction will be achieved by using the simulation algorithm in this paper.

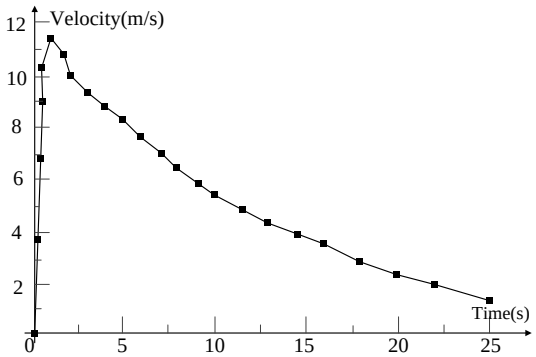

Fig. 7. Curve of explosive temperature changing over the time.

From Fig. 7 we can see that the explosion temperature goes down to dozens of centigrade higher than the ambient temperature in 5 to 10 seconds, which is matched with infrared thermal imaging records of explosion smoke in reference [9].

Fig. 8 illustrates the particle velocity varied along with time. At the beginning of explosion, buoyancy is greater than gravity and the air resistance, the explosive particles rise rapidly and reach maximum velocity in a few

seconds. Then explosive burning complete quickly, external energy injection is terminated and the heat of explosion smoke diffuses outward. The buoyancy and particle velocity of the explosion smoke decrease slowly.

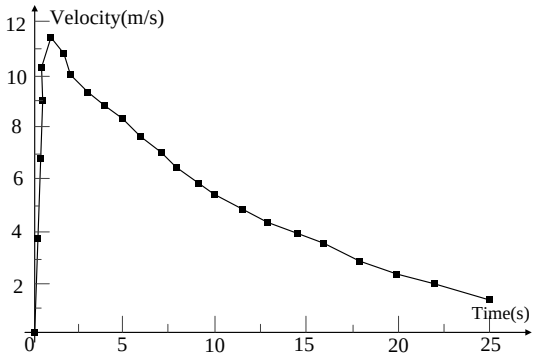

Fig. 8. Curve of particle velocity varying over time.

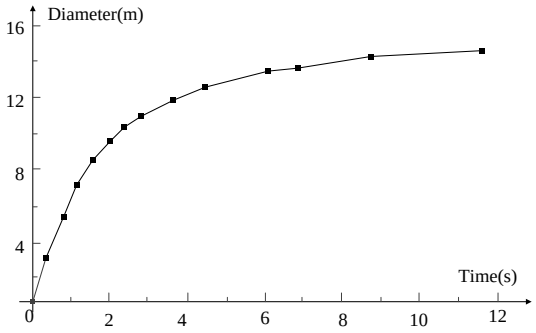

Fig. 9. Height of explosion smoke changing over time.

Explosion radius and explosion height of explosive smoke increase exponentially at the beginning when the explosive materials combust rapidly. The growth rate of radius and height of explosive smoke is slowed down gradually when the explosive combustion finish. This is verified in Fig. 9 and Fig. 10, and is consistent with the results given in reference [10]. From Fig. 7, Fig. 8, Fig. 9, and Fig. 10, we can see that the accompanied sound disappear at 5 to 10 seconds after the explosion.

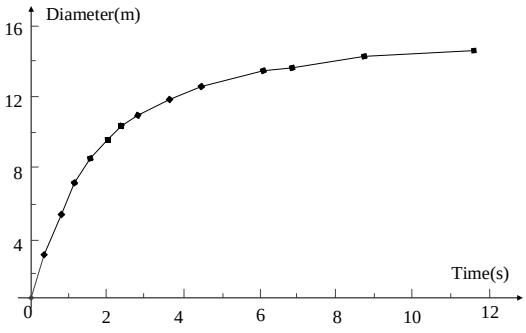

Fig. 10. Diameter of explosive product.

6. Conclusions

The fluid model and particle model are combined to simulate completely the whole process of explosion in this paper. The external shape, movement process and interaction of explosion fireball and smoke are real-time simulated. The change of trajectory of the explosive smoke when there are obstacles in the scene is realized and verified with simulation.

Acknowledgment

This work was partially supported by National Natural Science Foundation of China (61362006), Guangxi Natural Science Foundation (2014GXNSF AA118387), Project of Guangxi Education Department (YB2014121), Guangxi Colleges and Universities Key Laboratory of Intelligent Processing of Computer Images and Graphics (No.GIIP201407), and Innovation Project of Guangxi Graduate Education, China (2014105950812M25).

References

[1] Liang, Q., Xu, R. J., Du, J., & Yao, F. F. (2012). Study on dynamic modeling of explosion effect. *Journal of System Simulation*, *24(9)*, 1945-1951.

[2] Feldman, B. E., O'Brien, J. F., & Arikan, O. (2003). Animating suspended particle explosions. *ACM Transactions on Graphics*, *22(3)*, 708-715.

[3] Brunelli, E., & Marchioro, C. (2011). Vanishing viscosity limit for a smoke ring with concentrated vorticity. *Journal of Mathematical Fluid Mechanics*, *13(3)*, 421-428.

[4] Tang, Y.， Wu, Y., Lv, M. Y., & Song Z. J., (2012). Real-time smoke simulation using the improved adaptive vorticity confinement. *Journal of Chinese Computer Systems*, *33(12)*, 2676-2680.

[5] Chen, Z. Y., Tang, Y., Ma, K. Q., Lv, M. Y., & Zhai, S. Q. (2011). Simulation for smoke and obstacles interaction using the overlapping grids. *Journal of Chinese Computer Systems*, *32(6)*, 1118-1121.

[6] Gao, Y., Li, C. F., Ren, B., & Hu, S. M. (2013). View- dependent multiscale fluid simulation. *IEEE Transactions on Visualization and Computer Graphics*, *19(2)*, 178-188.

[7] Fedkiw, R., Stamy, J., & Jensenz, H. W. (2001). Visual simulation of smoke. *Proceedings of the 28th annual Conference on Computer Graphics and Interactive Techniques*, *Siggraph '01* (pp.15-22). Los Angeles, CA: Association for Computing Machinery (ACM).

[8] Maria, D. F. (2003). On the application of the helmholtz–hodge decomposition in projection methods for incompressible flows with general boundary conditions. *International Journal for Numerical Methods in Fluids*, *43(1)*, 43-69.

[9] Wang, S. Q. (2008). *Radiation monitoring technique and equipment used for "dirty bomb" attack*. Ph.D. dissertation, Antichemical Warfare Institute, Beijing.

[10] Duan, Z.-S., Luo, Y. F., Yuan, W., & Shang, A. G. (2013). Experiment and simulation of explosion cloud diffusion. *Chinese Journal of High Pressure Physics*, *27(2)*, 305-312.

Wenhui Zhang received the B.S. degree in mechanical engineering from Guilin Institute of Electronic Technology, Guilin, China, in 1992, and the M.Sc. degree in mechanical engineering from Nanjing University of Science and Technology, Nanjing, China, in 2000.

She is currently an associate professor with the School of Computer Science and Engineering, Guilin University of Electronic Technology, Guilin, China. Her research interests include virtual reality technology and realistic graphics.

Jiming Lin received the B.S. degree in electronic engineering from Harbin Shipbuilding Engineering Institute (HSEI), Harbin, China, in 1992, the M.Sc. degree in communication and information engineering from University of Electronic Science and Technology of China, Chengdu, China, in 1995, and the Ph.D. degree in ultrasonic electronics from the Department of Electronic Science and Engineering, Nanjing University, Nanjing, China, in 2002.

He is currently a professor with the School of Information and Communication, Guilin University of Electronic Technology, Guilin, China. His research interests include ultra wideband communications and location techniques in wireless communication system.