Physically Based Real-Time Terrain Deformations: Extensions

Radim Dvorak, Martin Drahansky
Department of Intelligent Systems, Faculty of Information Technology
Brno University of Technology, Brno, Czech Republic
{idvorak, drahan}@fit.vutbr.cz

Abstract

This paper describes a model which deals with physically based real-time terrain deformations. The problem of terrain deformation modeling is wide, it includes the modeling of terrain purely “by hand”, the terrain modeling by hand with help of some physical laws (e.g. modeling forces or material properties) and modeling deformations caused by external, mostly nature, forces. The presented method belongs to the last case. More specifically, in our work we designed, implemented and tested simple physically based model which simulates the terrain deformations caused mainly by moving objects on its surface and by explosions of bomb above the terrain. The results show that our proposal is suitable for its purpose and has potential to further extension and investigation.

Keywords: *Terrain Deformations, Terrain Dynamics, Simulation, Modeling.*

1. INTRODUCTION

The problem of the dynamic terrain modifications is a part of many computer science disciplines, namely computer graphics, modeling and simulation. From the computer graphics perspective, it means mainly finding of the way for terrain representation and visualization. The representation ability of a modification is crucial during the time according to the actual needs. In the term of physically based modeling, the terrain is represented as a model that is a simple imitation of the reality and it has specific physical characteristics. The simulation shows model behavior in time.

Terrain dynamics could be understood as its modification during the time. This modification is controlled by an external system or directly by a user. Generally, for simple creation and manipulation of graphical objects, it is suitable to work with feasible tools, which facilitate or at least make these tasks partly automatically. For non-physical terrain deformation, it’s recommended to have a tool for adjusting of the terrain geometry to some new added objects, e.g. houses, routes, rivers, lakes etc. There exists the project called Terrain Deformation Software [3], which deals directly with these problems for the OpenSceneGraph toolkit [7].

The terrain, as a physical model, could be dynamically modified by the currently acting external loads, especially by acting forces. These forces can be of an artificial or a natural character. The artificial forces are caused by the external sources (e.g. person, computer system etc.) and their purpose is mostly physically based manipulation or modification of an existing object [10].

Natural forces are mainly caused by solid objects (bodies) interacting with the terrain, by massive forces originated from bomb explosion or they can be from a meteorological origin (e.g. winds, erosions etc). The presented method deals with the first two mentioned cases, where solid figures and bombs take an effect on the terrain and thus cause its deformation.

The paper is structured as follows. In the next section, the previous work of related problems is depicted with outlining of commonly used physically based deformable models and their usage. Hereafter, the main approach and its advantages and disadvantages with the respect to defined goals are described. Finally, the simulation results are presented and discussed at the end.

2. PREVIOUS WORK

There do not exist many works which deal with problematic of physically based terrain deformations. In many projects, their focus is laid to deformable model in general or in the area not close to our work, such as chirurgic operation simulation [8], clothes modeling [8, 2] or e.g. crash modeling [9]. In all these cases the model which is deformed in time does not correspond to large terrain which consists of thousands of triangles and is described by its surface only.

In the next chapters, different kinds of deformation models will be described including their attitude and possible usage for our purpose.

2.1 Continuum Models

Continuum models consider the permanent energy equilibrium of the body, on which the external forces are acting. The body deformation is then a function of these acting forces and material characteristics of the body. The equilibrium is reached, when the body potential energy is minimal.

Finite element method, FEM, is commonly used for the finding of an approximation for a continuous function that satisfies deformation equilibrium. In FEM, the object is divided into the elements, joined at discrete node points. For every node point, the algebraic equation based on elements material properties and equilibrium potential is determined. This system of equations is then solved and new elements characteristics are set.

Generally, the solution for finding of continuum models is very time-consuming, because of the need to solve a linear system. Moreover, when the large deformations have to be accounted, the computational complexity arises significantly [8].

2.2 Mass-Spring Models

Mass-spring models represent bodies as a collection of masses connected by springs in a regular lattice structure. The form of the structure is usually a hexahedron and the springs between particles are set permanently.

Each particle actuates on another via the defined spring, which is usually interpreted as a linearly elastic. The force f_{si} acting on mass i, formed by the spring between masses i and j is often defined as [8]:

$$f_{si} = k_s(|x_{ij}| - l_{ij})\frac{x_{ij}}{|x_{ij}|} \quad (1)$$

where x_{ij} is a vector from the mass i to the particle j, k_s is the spring stiffness and l_{ij} is its rest length.

When the position of one mass changes the forces caused by the distance difference between masses, it starts the acting on itself. When the distance between two masses is higher than the rest length, the resulted force is an attractive force and on the other hand, when the distance is below the rest length, the force is repulsive. In regards to this behavior, the mass-spring system tries to reach the energy equilibrium state, where the distances between masses are equal to the rest length.

Thus, after releasing of the forces acting on the mass-spring system, the masses tend to reach its original positions. Materials with such behavior are called elastic.

However, it is often needed to model plastic materials, by which the permanent deformation occurs, when they are pushed by an external force. This behavior can be achieved by adding of the damping forces into the deformation model [10]:

$$f_{vi} = k_d(v_j - v_i) \quad (2)$$

where f_{vi} is the *viscous damping force* acting on the mass i, k_d is the viscous damping coefficient, v_j and v_i are the velocities of the masses i and j. This force occurs whenever the relative velocity of any two masses is nonzero and is acting as the masses velocities, which have been synchronized. The result is that all masses tend to stop moving and there occurs the permanent deformation.

The mass-spring systems include another damping force. This is called *global damping force* and acts on every mass opposite of its velocity. Therefore, this force damps movement of every mass and its main purpose is the guarantee of a numerical stability. The global damping force f_{gi} exerting on mass i can be expressed as [8]:

$$f_{gi} = -\gamma_i v_i \quad (3)$$

where γ_i is the global damping coefficient of the mass i and v_i is the velocity of the mass i.

The behavior of a mass-spring model can be described by equations (1), (2) and (3) as a collection of n differential equations, where n is a number of the masses [5]:

$$m_i\ddot{x}_i = -\gamma_i\ddot{x}_i + \sum_j g_{ij} + f_i \quad (4)$$

where m_i is the mass, x_i is the position, γ_i is the global damping coefficient, g_{ij} is the force caused by a spring between particles i and j, including the viscous damping force (2), and f_i is the total external force acting on a particle i.

The mass-spring systems are widely used, because of its simple implementation and intuitiveness. Further, they can model large deformations and have low computational costs.

2.3 Loosely Coupled Particle Systems

As in classical particle systems (mass-spring models), there is usually used classical Newtonian physics for describing of particles motion. However, instead of the fixed springs between particles, there is defined the potential field, which holds the particles together.

Generally, the potential field can be defined free, but it has to accomplish some specific requirements. First, when two particles are closer to each other than the rest distance, the resulted force must be repulsive. On the other hand, when two particles are farther than the rest distance, the resulted force must be attractive or zero, when the particles distance exceeds a particular limit.

The potentials are mostly defined, similarly to Eq. (1), as a linear elastic [11]. However, there exist some particle system implementations, where the potential force is defined similarly as in the molecular level. In [10], the Lennard-Jones potential is used to model the inter-particle forces. The Lennard-Jones potential is an approximation of the inter-molecular forces and it was successfully used for such kind of simulations.

The general form of the Lennard-Jones potential is as follows:

$$\Phi_{LJ}(r) = \frac{B}{r^n} - \frac{A}{r^m} \quad (5)$$

where $\Phi_{LJ}(r)$ is the potential magnitude, r is the particles distance and n, m, A and B are constants.

Alternative and more suitable form of the Lennard-Jones potential is:

$$\Phi_{LJ}(r) = \frac{-e_0}{m-n}\left(m\left(\frac{r_0}{r}\right)^n - n\left(\frac{r_0}{r}\right)^m\right) \quad (6)$$

where $\Phi_{LJ}(r)$ is the potential magnitude, r is the particles distance, r_0 is the particles rest distance, e_0 is so called dissociative energy (energy needed for the separation of two particles) and n, m are constants. The derivation of (6) could be found in [10].

The change of material properties is done by a modification of parameters e_0, m and n. When e_0, m and n are getting greater, the material stiffness is rising and vice versa.

3. SOLUTION

The design and implementation was influenced by the fact of using OpenSceneGraph environment. However, the principle of this method is general and it can be easily implemented for any scene and terrain representation.

The terrain consists, in general, of thousands of triangles, thus it is too huge structure for processing at once. On the other hand deformations are of local character and therefore the dynamical particle system was chosen to simulate real-time terrain deformations.

3.1 Deformations Caused by Moving Objects

The solution in the situation of terrain deformation induced by moving objects on terrain surface is similar to [5]. In this article, the specific surface particle system was used to direct simulation of terrain deformations.

The surface particle system is generated dynamically when the object touches the ground. Then the forces which are based on object properties such as mass, inertia and restitution (elastic) parameters start to act on it. The result is that some particles start to move from their initial positions. For the purpose of terrain consistency, the Lennard-Jones potential was used (6).

For the object-terrain collision detection the sophisticated schema with bounding volume hierarchy was used. The structures are dynamically rebuilt according to arisen deformations.

3.2 Deformations Caused by Bomb Explosion

The terrain deformation caused by bomb explosion, which means that there arises a pressure force from point source above terrain, is the specific case. The elevation of bomb can be small or larger which is partially crucial because of numerical imprecision occurrence.

Every explosion is accompanied by birth of massive pressure wave, which is spreading out to all directions from its origin by the same starting speed. Under the influence of wave progress from the source point, large part of soil which is close to place of explosion is being removed. The crater appears.

For the possibility to describe the character of the terrain damage or deformation by the suitable way, the reference explosion with the force of one ton of trinitrotoluene (TNT) is defined [6]. All other explosion sizes are expressed on the base of this unit.

The shock wave, blast wave and the crater width are important parameters of the explosion which approximately describe its character. We use them to set the basic formula for force expression which is further used to simulate the detonation process.

3.3 Mathematical Expression

The pressure P is defined as an amount of force F effected to area unit:

$$P = \frac{F}{S} \qquad (7)$$

where S is the area size.

With an advantage, the pressure can be expressed in decibel units. The reason is a direct application into the next calculation terms:

$$P_{db} = 20 \log(P_{Newton} /(2 \cdot 10^{-5})) \qquad (8)$$

where P_{db} is the pressure in decibel and P_{Newton} is the pressure in N/m^2 units.

One ton of TNT has the starting shock pressure in the size of P_{s1TNT} = 210.6 *db* and the width of a crater is W_{1TNT} = 7.13232 *m*. Size of general strength shockwave of explosive P_{sNTNT}, whose size is N-multiple of one ton TNT size, can be calculated this way:

$$P_{sNTNT} = P_{s1TNT} + 6.67 \log(N) \,. \qquad (9)$$

Next, the crater width W_{NTNT} of generally strength explosive of N-multiple of one ton TNT can be calculated on the basis of reference explosive as:

$$W_{NTNT} = W_{1TNT} N^{1/3} \,. \qquad (10)$$

Close to the explosion shockwave of the bomb, there is another wave, called blast wave, which is registered particularly by the stronger explosions which have the strength of shockwave above 207.46 *db*. Within such huge explosions, the blast wave strength is higher than shockwave strength, thus it is necessary to take it to simulation progress.

The size of blast wave P_Q can be derived from shockwave $P_{sNewton}$ (in N/m^2 units) as:

$$P_Q = 2.5 \cdot P_{sNewton}^2 /(7 \cdot P_{atm} + P_{sNewton}) \qquad (11)$$

where P_{atm} is the size of atmospheric pressure (P_{atm} = 101325 N/m^2).

Total pressure P_{total} arisen in the point of explosion just after its detonation is simply the sum of shock pressure and blast pressure:

$$P_{total} = P_S + P_Q \,. \qquad (12)$$

The pressure size is dropping proportionally to third power of distance x in another place than explosion epicenter:

$$P_x = P_{total} /(\frac{4}{3} \pi \cdot x^3) \,. \qquad (13)$$

According to pressure definition (7) and the equation (13), it is possible to express the force F_{Pi} causing on each particle i immediately after the explosion:

$$F_{Pi} = P_{total} /(\frac{4}{3} \pi \cdot x^3) \cdot (\frac{\pi \cdot W^2}{4C}) \qquad (14)$$

where P_{total} is the total pressure in explosion epicenter, W is crater width and C is particle count of particle system.

3.4 Implementation Issues

In case of deformations caused by moving objects, the forces acts on terrain surface are approximately constant or with small deviations. Thus the stability of simulation is in almost all cases assured even with usage of simple Euler integration scheme. The application of global damping forces in the model has the main part on that, so the convergence of the calculation is satisfied.

The case of bomb explosion is different. In this scenario, the forces arisen during detonation are much more heterogeneous because of great force dropping according to distance from epicenter. Due to this fact, the sophisticated integration process must be considered.

In our simulation tests we used popular 4th order Runge-Kutta method. It might be omitted that the method is very time consuming to calculate one step that it is proportional to quadruple of time of simple Euler scheme. The answer is yes but there must be kept in mind that the Runge-Kutta scheme is much more accurate and especially more stable and thus longer time steps could be used. This is crucial to explosion simulation and the results show it as suitable for this purpose.

3.5 Optimizations

Further extension of physical deformation model by [5] is in advanced optimizations of the algorithm.

There are two basic kinds of optimizations according to particle system dynamical behavior. First, in the previous algorithm version, the kd-tree structure in its original and basic form was used. Because of further usability of proposed model, there were made few intentional changes.

The modifications are not in kd-tree itself but in live and dynamics of that structure. The main idea is that the rebuilding of kd-tree is not necessary every time step but can be adapted to expected amount of changes of terrain. This is possible due to dynamical character of used structure, however, the time between each two rebuilds cannot be raised forever, because of rising calculation error.

Second optimization is done by adapting terrain bounding volume hierarchy (BVH) actualizations. It is not necessary to recalculate the terrain BVH in every time step as in the case of kd-tree. Moreover, when we use the half-edge structure to object-terrain collision detection, the complex BVH structure rebuilding can be done after relatively long time distances.

The effects of both optimizations are shown in Figure 1. It is obvious that optimizations made to structure behaviors give significant improvements to simulation performance.

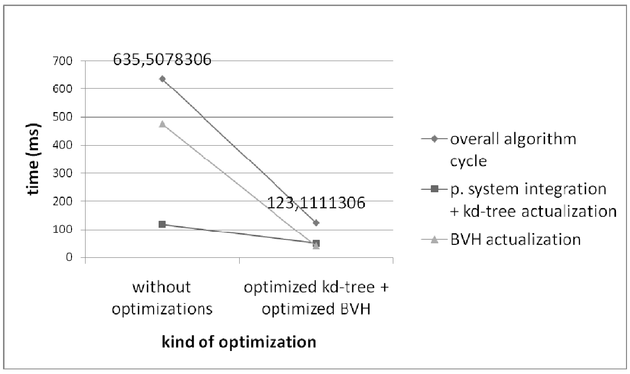


Figure 1: Time durations of the algorithm parts in 1000ms of simulation (98 particles).

4. RESULTS

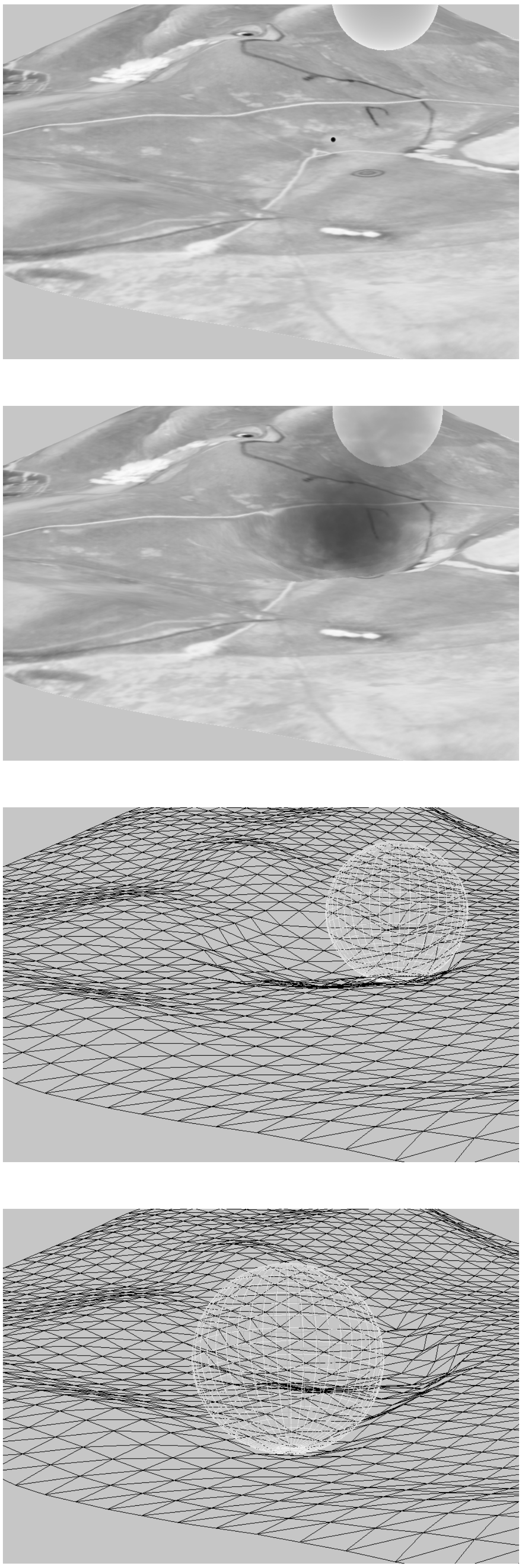
As an example and demonstration of proposed model, we have used simple and clear scene which consists of the part of terrain, small sphere object representing a bomb and larger sphere representing the object moving on the terrain surface.

The terrain consists of 2808 triangles and the body is defined as a rigid sphere. The terrain properties are set as follows: each particle mass m is 1.0 kg; dissociation energy e_0 is set to 300 J; constant m = 2; constant n = 4. The rest length r_0 is determined from the original surface form. The sphere mass m_s is 45 kg; the sphere radius r is 75 m. The Euler scheme has been selected for an integration process of the moving sphere. In opposite, for the bomb, the 4th order Runge-Kutta scheme has been chosen. The bomb strength was set to 5000 tons of TNT.

The simulation began by dropping the sphere and the bomb onto the terrain with non-zero sphere initial velocity. Simulation was running in real-time with approximately 1/60s simulation step. Figure 2 shows the simulation screenshot sequence.

The first screenshot in Figure 2 shows the initial conditions of the simulation, where the small black sphere marks the bomb above terrain and the bigger white one represents the interaction object with the terrain.

During the simulation process, the bomb touches the ground and in consequence the crater appears which shows the second screenshot. The crater causes the direction change of moving sphere what is demonstrated on the third, fourth and fifth screenshot (the wire models are used for clarity). The important fact in this experiment is that there were not any other forces acting on objects in spite of gravitation.



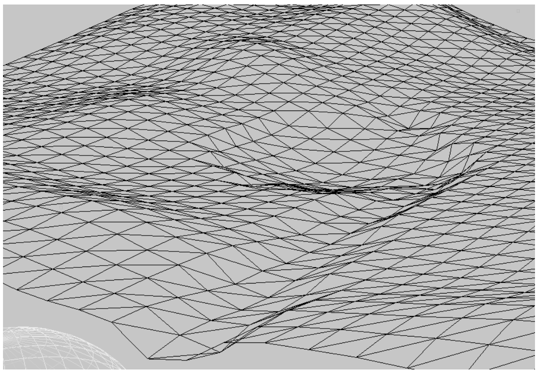


Figure 2: Simulation process.

5. CONCLUSION

The presented method was successfully used with OpenSceneGraph toolkit as a part of the scene. The usage of this method can improve visual quality and bring more realistic impression of the moving objects on the ground.

The physical part of this method is important from the option possibilities that can be set to both the terrain and the moving object or bomb, respectively. This can serve to definition of different kinds of object-terrain behavior, such as moving the object in the mud or in some wheat field on the one side, or minimal deforming the hard asphalt road on the other side. In the first example, there arises a big trail in the terrain whereas in the second case, the minimal, by eye not perceptible, deformation occurs. This could be seen, e.g., when the truck goes across the road from the surrounding field. This is the case of bomb explosions too. The method can distinguish between the width and depth of crater for different cases of terrain material and bomb properties.

The next project progress should concern the heterogeneous possibility of terrain deformation. This means that the surface particle system arisen during simulation should generate new terrain triangles and thus react even more realistic. This process is more time consuming, however, with optimizations, including these presented in this paper, the simulation will be able to run in real time.

6. ACKNOWLEDGMENTS

The research was done under the support of the Research Intention No. MSM 0021630528 “Security-Oriented Research in Information Technology”.

7. REFERENCES

[1] Barraff, D. 1997. An Introduction to Physically Based Modeling: Rigid Body Simulation. Siggraph Course Notes.

[2] Barraff, D. 1998. *Large Steps in Cloth Simulation*. Siggraph 98, Orlando.

[3] Computer Graphics Systems Development Corporation. 2007. *Terrain Deformation Software and osgTDS, User’s Manual*. Computer Graphics Systems Development Corporation.

[4] Dvorak, R. and M. Drahansky. 2007. Real-time terrain deformations. In *Proceedings of the Fourth International Conference on Image and Graphics*, 1020-1025, Los Alamitos.

[5] Gibson, S. F. F., and B. Mirtich. 1997. *A Survey of Deformable Modeling in Computer Graphics*. MERL, Cambridge.

[6] Hamby, W. 2004. *Bomb Shock Wave Estimation*. Available via <http://www.makeitlouder.com/document_bombshockwaveestimation.html> [acceseed 2008-03-11].

[7] Martz, P. 2007. OpenSceneGraph Quick Start Guide. lulu.com.

[8] Nealen, A. et al. 2006. *Physically Based Deformable Models in Computer Graphics*. Computer Graphics Forum, Vol. 25, issue 4, 809-836.

[9] O’Brien, J. F. and J. K. Hodgins. 1999. *Graphical Modeling and Animation of Brittle Fracture*. Siggraph 99. Los Angeles.

[10] Tonnesen, D. 1998. Dynamically Coupled Particle Systems for Geometric Modeling, Reconstruction, and Animation. Ph.D. thesis, University of Toronto, Canada.

[11] Witkin, A. 1997. Physically Based Modeling: Principles and Practice : Particle System Dynamics. Siggraph Course Notes.

About the authors

Radim Dvorak is a Ph.D. student in the Department of Intelligent Systems at the Faculty of Information Technology, Brno University of Technology, Czech Republic. He received a master degree in information technology program from the Brno University of Technology on subject of deformable bodies’ dynamics. He is interested in modeling, simulation and computer graphics. His contact email address is idvorak@fit.vutbr.cz.

Martin Drahansky is an assistant professor in the Department of Intelligent Systems at the Faculty of Information Technology, Brno University of Technology, Czech Republic. He received his Ph.D. degree in 2005 in the branch of Information technology and his master degree in 2001 in the branch of Informatics and Computer Science and Electrotechnics. He is interested in biometric systems, artificial intelligence, security, cryptography and sensoric systems. More information could be found here: http://www.fit.vutbr.cz/~drahan. His contact email address is drahan@fit.vutbr.cz.