TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.5, May 2014, pp. 3811 ~ 3817
DOI: http://dx.doi.org/10.11591/telkomnika.v12i5.5124

Cloth Simulation Based on Simplified Mass-Spring Model

Wenqing Huang*, Jing Hu, Keqiang Yu, Yaming Wang, Mingfeng Jiang
School of Information Science and Technology, Zhejiang Sci-Tech University
Zhejiang, Hangzhou, 310018, China
*Corresponding author, e-mail: patternrecog@163.com

Abstract

The technology of cloth simulation can be used in many fields. Based on physical model of cloth simulation, we established the simulation system with a simplified mass-spring model. A modified implicit method was proposed in order to produces realistic animation. This method can increase the computational efficiency to a large extent and is easy to be realized with a stable and good real-time performance. Furthermore, the approach of AABB (Axis-Aligned Bounding Boxes) is adopted for the detection of cloth collision. Experimental results show that our system can give excellent real-time effect of cloth simulation.

Keywords*: cloth simulation, simplified mass-spring model, implicit method, collision detection.*

1. Introduction

Since the 1980s, with the development of computer technology, cloth simulation technology has been applied to many fields, such as computer animation, 3D game, garment and so on. Due to the characteristics of cloth [1], for instance, large deformation, non-linear stress and non-linear constraints, the simulation of cloth is complex. In the past two or three decades, many effective methods have been suggested by people. Nowadays these methods can be roughly divided into three classes: geometrically-based methods, physically-based methods and hybrid methods.

In the method based on geometric, we do not consider the physical factors of cloth. For example, the mass of cloth and the elastic coefficient of cloth are not considered. So we do not need to calculate a lot of complex equations about cloth's physical state. The advantages of this method are that it is simple and the calculating speed is fast. At the same time, this method has shortages. It is difficult to produce dynamic animation realistically. According to the principles of kinematics, physically-based method gets partial differential equations based on deformational relationship, which is produced by the interaction between particles. Solving the equations by using numerical integration method, the movement trail of particles in time sequence is known. So we can obtain 3d spatial dynamic simulation of cloth. This method can be used for the simulation of cloth prone to large deformation. It also can reflect many different physical properties of the cloth. Considering the advantages and disadvantages of geometrically-based method and physically-based method, we combine these two methods to produce a new hybrid method. But the existing hybrid method can only simulate some simple, special and regular shape of clothe, it has certain limitations in the practical applications. Collision detection and response have a great influence on the speed of cloth simulation. So an efficient collision detection algorithm is one of the key issues for cloth's dynamic real-time simulation. It largely affects the authenticity and accuracy of cloth simulation.

2. Simplified Mass-spring Model

Mass-spring model [2], the typical physically-based method, is widely used for representing cloth. It is a simple technique and the algorithm is easy to be implemented, high computational efficiency is obtained with mass-spring model. A typical model based on mass-spring model is proposed by Provot [3]. In this model, cloth is represented as a grid system

Received November 11, 2013; Revised December 30, 2013; Accepted January 12, 2014

constituted by mass-points and springs, the link between mass-points are mainly through springs. In order to according with cloth's characteristics of non-linear stress, springs are divided into three kinds: shear springs, structure springs and bending springs. The basic mass-spring model is shown in Figure 1.

Realistic simulation usually along with complex algorithm and high cost of computing, in order to improve the efficiency of computing, the basic mass-spring model has been simplified in this paper. Experiment shows that it does not affect the system significantly whether it is with two shear springs or one shear spring. Therefore, we discard one of the two shear springs to simplify the model. The simplified mass-spring model is shown in Figure 2.

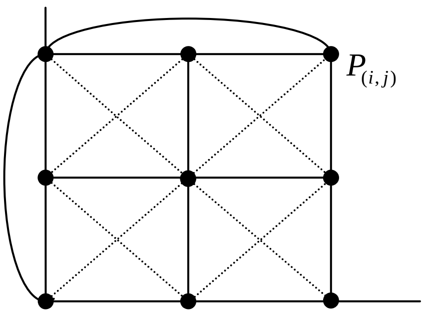


Figure 1. Basic Mass-spring Model

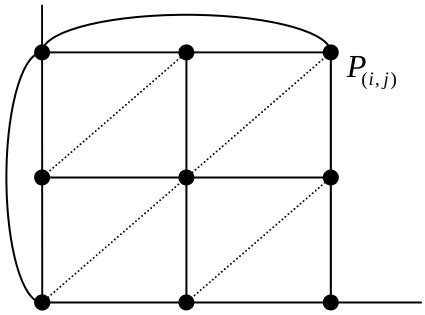


Figure 2. Simplified Mass-spring Model

3. Force Analysis of the Model

In mass-spring model, the motion of each mass-point depends on both internal forces and external forces that the mass-point suffered. And the motion of all mass-points reflects the deformation of cloth system. Internal forces include elasticity and damping force, etc. External forces include gravity and wind force, etc.

3.1. Elasticity

Assuming that mass-point *i* and mass-point *j* are connected by a spring, according to Hooke's law, the change in length of spring is proportional to forces that the mass-points suffer [4]. The elastic force is defined as follows:

$$F_{si}^{t} = k_{i,j}\left(| x_j - x_i | - l_{i,j}^{0}\right)\frac{\left(x_j - x_i\right)}{| x_j - x_i |} \tag{1}$$

Where, $k_{i,j}$ is the elastic coefficient of spring between mass-point *i* and *j*, x_i and x_j represent the positions of mass-point *i* and *j*, $l_{i,j}^{0}$ represents the initial length of the spring.

3.2. Damping Force

Damping force is necessary to maintain the stability of the system. For mass-point *i*, which is adjacent to mass-point *j*, the damping force is defined as follows:

$$F_{di}^{t} = \sum_{j=0}^{n} d_{ij}\left(v_i - v_j\right) \tag{2}$$

Where, $d_{i,j}$ represents the elastic coefficient of spring between mass-point *i* and *j*, v_i and v_j are the velocities of mass-point *i* and *j*.

3.3. Gravity

The gravity that mass-point *i* suffer is defined as follows:

$$F_{gi}^{t} = m_i g \tag{3}$$

Where, m_i represents the mass of mass-point *i*, *g* is the acceleration of gravity, it has a constant value.

3.4. Wind Force

We define a wind force to make the simulation more realistic. At the same time, in order to simplify the calculation, we ignore the changes when wind encounters with the cloth. The wind force is defined as follows:

$$F_{wi}^{t} = k_w \left(v_w - v_i\right) \tag{4}$$

Where, k_w is the wind coefficient, v_w represents wind speed, and v_i is the velocity of mass-point *i*.

According to Newton's second law of motion, the force that mass-point *i* suffer is defined as follows:

$$F_i^{t} = F_{si}^{t} + F_{di}^{t} + F_{gi}^{t} + F_{wi}^{t} \tag{5}$$

4. Numerical Integration Methods

In the cloth simulation based on mass-spring model, numerical integration method is one of the core matters. Recently, the common numerical integral methods include explicit Euler method [5], implicit Euler method [6] and improved methods based on these two methods. Implicit integration method is generally used to solve the rigid problem of equations. It avoids large time steps in calculation and overcomes instability problem. But it also has a problem of low efficiency. Explicit Euler method is simple and has fast computing speed. But in this method, the time step must be small to obtain stability. Considering the merits and demerits of these two methods, a modified implicit method is proposed in this paper. It not only solves the instability problems and small time steps in explicit Euler method, but also avoids large amount of calculations in implicit integration method. The formula of implicit integration method is defined as follows:

$$v_i^{n+1} = v_i^{n} + h \bullet \frac{F_i^{n+1}}{m_i} \tag{6}$$

$$x_i^{n+1} = x_i^{n} + h \bullet v_i^{n+1} \tag{7}$$

In Equation (6), F_i^{n+1} is unknown at current time, we can use the following formula for approximate calculation of F_i^{n+1}.

$$F^{n+1} = F^{n} + H \bullet \Delta x^{n+1} \tag{8}$$

Where H is the Hessian matrix of the system, $\Delta x^{n+1} = x^{n+1} - x^{n} = (v^{n} + \Delta v^{n+1})h$. So Equation (6) and Equation (7) can be rewritten as follows:

$$\left(I - \frac{h^2}{m} H \right) \Delta v^{n+1} = \left(F^n + hHv^n \right) \frac{h}{m} \tag{9}$$

Where hHv^n represents the additional forces, these forces can be calculated as follows mentioned by Desbrun [7]:

$$\left(hHv^n \right)_i = h \sum_{\forall j | (i,j) \in E |} k_{ij} \left(v_j^n - v_i^n \right) \tag{10}$$

Desbrun splitted the spring force into two parts and considered only the linear part, then the approximated Hessian matrix is defined as follows:

$$\left\{ \begin{array}{ll} H_{ij} = k_{ij} & if \;\; i \neq j \\ H_{ii} = -\sum_{j \neq i} k_{ij} & \end{array} \right\} \tag{11}$$

Where H_{ij} denotes the value of H at *i*th row and *j*th column. The velocity change of mass-point *i* is only related to its adjacent mass-points. H_{ij} is zero when mass-point *i* and *j* are not linked with a spring. Therefore, Equation (9) can be rewritten as follows:

$$\left(1 - \frac{h^2 H_{ii}}{m_i} \right) \Delta v_i - \frac{h^2}{m_i} \sum_{\forall j | (i,j) \in E} \left(H_{ij} \Delta v_j \right) = \frac{F_i^n h}{m_i} \tag{12}$$

We adopted the approximated Hessian matrix proposed by Desbrun to simplify the calculation. Assuming that the spring constant of all springs is k, and the number of mass-points linked to mass-point *i* is n_i, then the approximated Hessian matrix can be rewritten as follows:

$$\left\{ \begin{array}{l} H_{ij} = k \\ H_{ii} = -kn_i \end{array} \right\} \tag{13}$$

Then the deformation formula of Equation (12) is defined as follows:

$$\Delta v_i^{n+1} = \frac{F_i^n h + kh^2 \sum_{(i,j) inE} \Delta v_j^{n+1}}{m_i + kh^2 n_i} \tag{14}$$

Where Δv_j^{n+1} is unknown. Because we can not calculate Δv_i^{n+1} directly, an approximated formula is adopted to calculate Δv_j^{n+1}.

$$\Delta v_j^{n+1} \approx \frac{F_j^n h}{m_j + h^2 \sum_{(j,l) \in E} k_{jl}} \tag{15}$$

Then Δv_i^{n+1} can be calculated as follows:

$$\Delta v_i^{n+1} = \frac{F_i^n h + h^2 k \sum_{(i,j) \in E} F_j^n h \Big/ \left(m_j + h^2 k n_j \right)}{m_i + h^2 k n_i} \tag{16}$$

Now we can calculate Δv_i^{n+1} directly since F_j^n is known. In this method, we do not need to solve a large linear system, which is a critical flaw in implicit integration method. It updates the state of mass-points in $O(n)$ time when the number of total springs are $O(n)$. Moreover, the mass of mass-point, time step and spring coefficient can be easily modified without any additional computations. So it is faster than most general implicit numerical integration methods.

5. Collision Detection and Response

Collision detection [8] and response are very important in cloth simulation. At each time step, It is must be tested whether cloth collides with the environment object or not. If the collision occurs, it is necessary to make a appropriate response. Collision detection algorithms can be roughly divided into two types: Hierarchical Bounding Bolumes [9] and Space Discretization techniques [10]. Hierarchical Bounding Bolumes was first put forward by Volino [11], the core idea of this approach is wrapping up the complex geometric object with a bigger bounding box. Comparing with the geometric object, the bounding box is with simpler geometric features. When performing collision detection, we first carry on intersection test between the bounding boxes. Only when the bounding boxes have intersected will further intersection calculation of the wrapped geometric object be processed. Hierarchical Bounding Bolumes can exclude basic geometric elements that will not intersect as soon as possible. Thereby, the speed of collision detection is increased.

AABB [12] is the most widely used collision detection algorithm. Calculating the AABB of a given object is simple, we just need to calculate the maximum and minimum values of the basic geometric elements' vertex coordinate. Intersection test between AABB is simple, two AABB are intersectant only when the projections in three coordinate axes are all overlap. We need at most six comparison operations to complete an intersection test between two AABB.

In the process of cloth simulation, it is not allowed that cloth is embedded itself or embedded in the objects in the scene. When a collision is detected, we need to perform a collision response at once.

6. Experimental Results

Based on OpenGL graphics library, we use c++ as a development environment for dynamic simulation of cloth. Figure 3 shows the sequence results of cloth simulation when cloth in horizontal suspension posture. In Figure 3, two mass-points are constrained to be fixed. Figure 3(a) shows the initial state of cloth simulation, and the intermediate state of cloth simulation is shown in Figure 3(b).

(a) (b)

Figure 3. Sequence of Cloth Simulation when Cloth in Horizontal Suspension Posture

The Sequence results of cloth simulation when cloth in vertical suspension posture are shown in Figure 4. In the same way, two mass-points are constrained to be fixed. Figure 4(a) and (b) show the initial state and intermediate state of cloth simulation.

Figure 5 shows the simulation results of cloth that falls under the effect of gravity. The preliminary state of cloth falling under the effect of gravity is shown in Figure 5(a), and the intermediate state of cloth falling is shown in Figure 5(b).

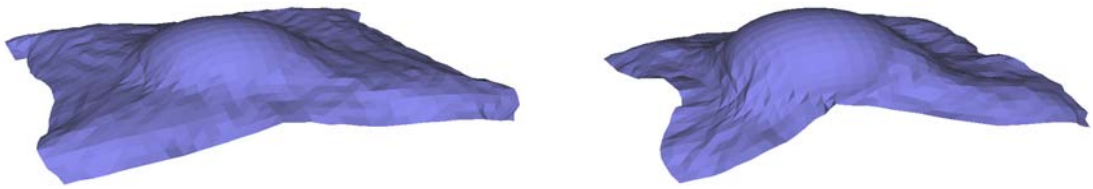
Figure 6 shows the sequence results of cloth that collides with a ball. The state of preliminary collision is shown in Figure 6(a), and Figure 6(b) shows the intermediate state of cloth collision.

(a) (b)

Figure 4. Sequence of Cloth Simulation when Cloth in Vertical Suspension Posture

(a) (b)

Figure 5. Sequence of Cloth Simulation when Cloth Falling under the Effect of Gravity



(a) (b)

Figure 6. Sequence of Cloth Simulation when Cloth Colliding with a Ball

From the experimental results, we find that our approach can rapidly gain realistic dynamic effect of cloth simulation.

7. Conclusion

In digital age, cloth simulation technology has broad application prospects. Its development will profoundly influence our lives. The realistic significances and practical values of this technology can not be ignored. In this paper, the computational efficiency is improved with the simplified mass-spring model. The modified implicit method is different from general implicit method, it does not involve solving a large linear system. And this method is very intuitive and easy to be implemented. The experiments show that this method can produce plausible animation results with large time steps. With the growing demands for fast cloth simulation applications, this method has some reference values.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China (Grant No. 50875245, 61070063, 61272311, 30900322) and the Zhejiang Province Natural Science Foundation (Grant No. Y107558).

References

[1] Li Qingfu, Yuan Ye, Chen Junxia. Segmentation for Fabric Weave Pattern using Empirical Mode Decomposition based Histogram. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(4): 1996-2001.
[2] A Luciani, S Jimenez, JL Florens, C Cadoz, O Raoult. Computational physics: a modeler simulator for animated physical objects. In Eurographics'91, Vienna, Austria, September 1991.
[3] Provot X. Deformation Constraints in A Mass-spring Model to Describe Rigid Cloth Behavior. Proceeding of Graphics Interface 95, Quebec. 1995; 147-155.
[4] Jaruwan M, Ratan G, Shafaq C. 3D Soft Body Simulation Using Mass-spring System with Internal Pressure Force and Simplified Implicit Integration. *Journal of Computers.* 2007; 2(8): 34-43.
[5] William Press, Saul Teukolsky, William Vetterling, Brian Flannery. Numerical Recipes in C, second edition. Cambridge University Press, New York, USA. 1992.
[6] David Baraff, Andrew Witkin. Large steps in cloth simulation. *In Michael Cohen, editor*, SIGGRAPH 98 Conference Proceedings, Annual Conference Series, pages 43-54. ACM SIGGRAPH, Addison Wesley. 1998.
[7] M Desbrun, P Schroder, A Barr. *Interactive animation of structured deformable objects*. Proc. of Graphics Interface. 1999.
[8] Zachmann G, Langetepe E. Geometric data structures for computer graphics. Natick, MA, USA: A.K. Peters, Ltd; 2006.
[9] Norhaida Mohd Suaib, Abdullah Bade, Dzulkifli Mohamad. Hybrid Collision Culling by Bounding Volumes Manipulation in Massive Rigid Body Simulation. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2013; 11(6): 3115-3122.
[10] Teschner M, Heidelberger B, Muller M, Pomeranets D, Gross M. *Optimized spatial hashing for collision detection of deformable objects.* Proceedings of vision, modeling, visualization. 2003; 47-54.
[11] Volino P, Magnenat-Thalmann N. Efficient Self Collision Detection on Smoothly Discretized Surface Animations using Geometrical Shape Regularity. *Computer Graphics Forum*. 1994; 13(3): 155-166.
[12] G.Vanden Bergen. Efficient Collision Detection of Complex Deformable Models using AABB Tree. Journal of Graphics Tools. 1997; 2(4): 1-13.