Boundary Condition and Constrained Method in Cloth Collision

Min Shi
North China Electric Power University
shi_min@ncepu.edu.cn

Tianlu Mao
Institute of Computing Technology, CAS
ltm@ict.ac.cn

Yabing Huang
Institute of Computing Technology, CAS
ybh@ict.ac.cn

Junzheng Wang
North China Electric Power University
jzw@ncepu.edu.cn

Mingdeng Zhu
Institute of Computing Technology, CAS
mdzhu@ict.ac.cn

Zhaoqi Wang
Institute of Computing Technology, CAS
zqwang@ict.ac.cn

Abstract

Collision detection and response is one of the bottlenecks that restrict the application of cloth animation. In order to prevent penetration of cloth-body and self-penetration, it is necessary to accurately extract all the collisions and implement stable response. However, collisions and self -collisions are likely to be undetected when they are in a critical contact state. Once a collision is missed, the cumulative numerical error will result in disastrous consequences. In this paper, various boundary conditions in cloth contact are analyzed and the solutions are given, which can overcome missing collision and ensure the robustness of detection. After that, collisions are regarded as constraints which are introduced to construct constrained dynamics equations. A semi-global constraint optimization method is used to solve the constraint equations which is based on collision dynamic influence. Using this method, complex and potential collision can be picked out, while invalid collision can be abandoned. It not only ensures the effectiveness of collision detection, but also simplifies the constraint set. The experiments show that the method improves the detection accuracy and response stability. At the same time, the computing efficiency is also improved.

Keywords: Cloth Animation; Collision; Constraints; Animation

1. Introduction

Fabric is a kind of thin and soft material, its collisions in motion are much more complex than the rigid body. During an animation, any one vertex on the cloth or garment may collide with the objects in the surrounding, or collide with itself. In order to prevent the cloth from penetrating the surrounding objects or itself, it is necessary to accurately detect all collisions that occur during the movement of the cloth, and stably respond them. Even though one collision being ignored releasing, the penetration will result in a disaster.

Compared with a rigid body or an ordinary elastomer, fabric is more flexible and easily deformed. Accordingly, collisions are more frequent to appear and more complex to deal with. Due to numerical error, collisions occurring between cloth surfaces or cloth and other objects are likely to be missed. Especially when an element or a face is in the critical contact state, missing is more likely to occur. In this paper, three contact states are analyzed and handling, which are combined with continuous detection.

A constraint method is usually used to deal with collision response, that is, collisions are regarded as the constraint, based on which constraint equations are constructed and further carried out. Local constrained optimization is used in some systems by solving each collision one by one, while global constraint optimization is used in other systems by spending more time solving large-scale constraint equations in exchange for stable response. Local constraints usually meet the efficiency requirements, but may lead to an unresolved death cycle. As shown in Figure 1, the distance between point P and surface S1 is less than the given threshold. So is that between point P and surface S2. In order to avoid colliding with the left contact surface, a right shift is needed for the point P, that results in colliding with the right contact surface, vice versa. This competing constraint will lead to an iterative process, which will result in a system crash.

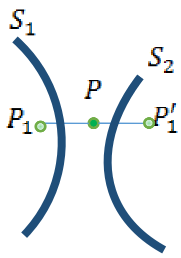


Figure 1: competing constraint

In this paper, the idea of combining momentum method with semi-global constraint optimization method is adopted in the process of collision response. In global constraint methods, all the collisions after each iteration are added to the constraint set and the collisions that do not appear again are abandoned. In the proposed method, a dynamic impact factor is labeled for each collision constraint, and according to which, the constraint is dynamically adjusted. When the influence size of the collision exceeds the specified threshold, the collision is added to the constraint set, and otherwise it will be discarded. This method makes some complex and potential collisions be picked out while others ineffective collisions be discarded, that not only ensures the effectiveness of collision set, but also quickens down the collision set and reduces the computational cost. At the same time, it keeps the simulation stable.

2. Related Work

In the field of computer animation, the research on collision detection and collision response has been going on for many years, and a lot of literatures have been reported. Some classical methods [1-3] have been the basis of subsequent research.

2.1 Collision Detection

Hierarchical bounding box method [2] is the most classical collision detection technology based on object space, which effectively reduces the number of intersecting test times by hierarchical data structure and cutting technique. It is mainly used for rigid body animation in the early time. Later, researchers constructed different levels of hierarchical bounding boxes, making it suitable for flexible objects such as fabric simulation. This kind of method needs to carry on the dynamic renewal to the pre-construction data, usually need a large amount of computation. In contrast, the other methods are based on collision detection of image space [4], which do not need time-consuming pretreatment. Since accurate collision cannot be obtained, so it is difficult to carry out accurate collision response.

On the basis of classical work, researchers put forward some new methods. The main strategy is to eliminate the unnecessary "Collision Pair" to reduce the number of collisions, so as to improve the efficiency of collision detection. Wong et al [5] filtered the repeatedly detected triangles by using “feature filter layer”. Curtis et al [6] used “representing triangles” to reduce the intersection of repeated geometric elements. Zhang et al [7] proposed an energy-based acceleration method that discarded self-collision detection when the energy in a bounding box was below a specified threshold. Other researchers focused on how to maintain the accuracy of collision detection. Tang et al [8] proposed an accurate continuous collision detection method based on Bernstein symbol classification. Wang et al [9] proposed a robust collision detection method based on the characteristics of high accuracy and sensitivity to numerical error of garment animation, which can accurately detect the collision in the system.

2.2 Collision Response

When collisions occur, it is necessary to immediately modify the speed and position of the collision elements in order to avoid the penetration. Repulsion method [1] avoids collision by calculating the depth the fabric penetrates to the scene and then applying a proportional repulsion. Geometric projection method [3] is a collision response method that was used in early time. Although this kind of method is simple, it is usually poor for response. The momentum method [10] describes collisions based on the momentum theorem and calculate the change of the momentum. Since it assumes that the contact of objects is instantaneous, so it cannot avoid colliding again after the response. Constrained collision response is another common method. Harmon et al [11] took the collision as the optimization condition, and the collision response was obtained by minimizing the energy change of the system. Otaduy et al [12] obtained robust and precise collision response by refining constraint. Since it needed to solve the large-scale linear equations, the computation was large.

In this paper, a semi-global constrained optimization method is presented. A dynamic impact factor is marked for each collision constraint, which is used to determine whether the collision will be added to the constraint set. Once the size of constraint impact is under the threshold, it will be discarded. Using this method, the efficiency of collision response can be improved by reducing the constraint set.

3. Collision Detection

In cloth animation, a continuous collision detection method is commonly used to detect whether a collision occurs over a period of time.

First, we should consider penetration between P and triangular ABC. Assuming that the current time period is $[t_i, t_{i+1}]$, and $t_{i+1} = t_i + \Delta t$. During this period, the necessary and sufficient conditions for the penetration between the point P and the triangle ABC are: P falls in the triangle, the equation (1) is satisfied.

$$\overrightarrow{AP}(t) = u\overrightarrow{AB}(t) + v\overrightarrow{AC}(t) \qquad (1)$$
$$t \in [t_i, t_{i+1}],\ u, v \in [0,1],\ u + v \le 1$$

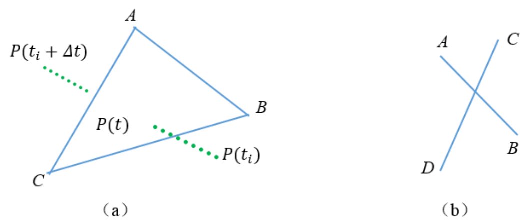


Figure 2: point-triangle intersection and edge-edge intersection

Since there is an error in the numerical computing, adverse effect will be caused if P is too close to the triangle ABC at the end of the period. So we introduce a close handling technique, combined with the above continuous detection, which can keep the robustness and stability.

There are three specific cases, in which the distance is less than the given threshold value $\grave{O}$ and a collision is considered to occur. First, the projection P', which the point P projects on the plane ABC, falls within the triangle ABC and $\|PP'\| < \grave{O}$, as shown in Figure 3 (a). Second, P' is not within the triangle ABC, but the projection P'', which P projects to any side of the triangle ABC, falls on the side and $\|PP''\| < \grave{O}$, as shown in Figure 3 (b); Third, P' is not in the triangle ABC, P'' does not fall on the corresponding edge, but the distance of point P to a vertex is less than $\grave{O}$, as shown in Figure 3 (c). In these three cases, although the geometric elements do not actually contact, but they will be added to the collision constraint set.

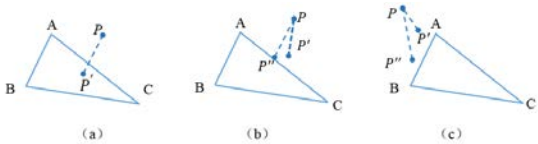


Figure 3: distance detection between P and ABC

Second, an edge-edge collision occurs only if the edge AB has a common intersection with the CD. That is to meet the formula (2).

$$\overrightarrow{AC}(t) = u\overrightarrow{AB}(t) - v\overrightarrow{CD}(t) \qquad (2)$$
$$t \in [t_i, t_{i+1}],\ u, v \in [0,1]$$

Similar to "point-triangle" critical collision, we can get three cases where edge AB and edge CD are critical close, as shown in Fig.4.

The dotted line is the projection of edge AB on the projection plane. First, the projection of the edge AB intersects with the edge CD, and the distance from the point of intersection M to line AB is $\|MN\|$, and $\|MN\| < \grave{O}$. Second, the projection of AB to the projection plane does not intersect the CD, but the distance from the point M (the intersection of the projection and the extension line of the edge CD)to any end of the CD is less than the given threshold $\grave{O}$. Third, the distance between any endpoint of the edge AB and any endpoint of the edge CD is less than the threshold $\grave{O}$. In these three cases, it is assumed that AB collide with CD.

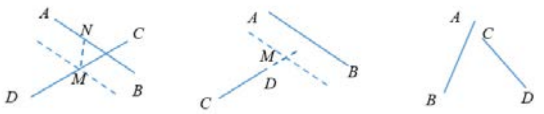


Figure 4: Edge AB and CD Distance Detection

4. Response based on momentum method

4.1 Momentum Method

Momentum method is a commonly used method, in that momentum theorem is used to describe the collision and the amount of momentum changes is calculated [13].

For "point-triangle" case, suppose that P' is a collision point that P falls in the triangle $x_1x_2x_3$, denote $P' = \omega_1 x_1 + \omega_2 x_2 + \omega_3 x_3$, where $\omega_1 + \omega_2 + \omega_3 = 1$. Assuming that the mass of the point P and the collision point P' are all *m*, the projection of the relative velocity in the plane normal $\hat{n}$ is V_N, then the momentum can be written as $I = \frac{m}{2} V_N$. After respons, the relative speed between P and P' should be 0, so the responding impulse is equation (3):

$$\tilde{I} = \frac{2I}{1+{\omega_1}^2+{\omega_2}^2+{\omega_3}^2} \quad (3)$$

The velocities of three vertexes in the triangle are shown in formula (4) , and the velocity of P is shown in formula (5).

$$\vec{v}_i^{new} = \vec{v}_i + \omega_i(\tilde{I}/m)\hat{n},\ \ i = 1,2,3 \quad (4)$$

$$\vec{v}_4^{new} = \vec{v}_4 - (\tilde{I}/m)\hat{n} \quad (5)$$

For edge-edge collision case, assuming that the cross product of the two edges is $\hat{n}$, the relative position of the collision point on the edge x_1x_2 is P, the relative position on the edge x_3x_4 is P', then denote $P = (1-a)x_1 + ax_2$, $P' = (1-b)x_1 + bx_2$. Assuming that the mass of P and P' are both m, and the momentum along direction $\hat{n}$ is I. To make the relative speed between P and be 0 P' after response, the required impulse is shown as equation (6):

$$\tilde{I} = \frac{2I}{a^2+(1-a)^2+b^2+(1-b)^2} \quad (6)$$

After responding, the velocities of each endpoint are:

$$\vec{v}_1^{new} = \vec{v}_1 + (1-a)(\tilde{I}/m)\hat{n} \quad (7)$$

$$\vec{v}_2^{new} = \vec{v}_2 + a(\tilde{I}/m)\hat{n} \quad (8)$$

$$\vec{v}_3^{new} = \vec{v}_3 - (1-b)(\tilde{I}/m)\hat{n} \quad (9)$$

$$\vec{v}_4^{new} = \vec{v}_4 - b(\tilde{I}/m)\hat{n} \quad (10)$$

When collisions occur, not only the normal velocity needs to be calculated, but also the tangential friction needs to be calculated. The magnitude of the friction force is proportional to the normal force acting in the opposite direction of the relative velocity. In order to avoid the relative movement of the object in the opposite direction, an upper limit for friction should be set. As shown in equation (11), where μ is the coulombic friction coefficient and $\vec{v}_T^{pre}$ is the tangential relative velocity before the friction is calculated.

$$\vec{v}_T = \max\left(1 - \mu\frac{\Delta V_N}{\vec{v}_T^{pre}}, 0\right)\vec{v}_T^{pre} \quad (11)$$

4.2 Secondary Collision

Although the momentum method is simple and fast, a secondary collision is easy to occur when the speed and location of a certain element on the fabric are modified. As shown in Figure 5 (a), assuming that the velocity of a certain point on the fabric is v, and this point movements from P to P_1; In order to avoid the penetration into surface S, the velocity is modified to v'. Unfortunately, the point moves from P to P_1', penetrating into the surface S again. In reality, the collision surface may be much more complex, and in the worst case the collision in the system cannot be resolved, which finally falls into the dead cycle.

In addition, when there are multiple collisions in the same direction, an excessive response is very possible, as shown in Fig. 5 (b). The edge AB moves to A_1B_1 in a simulation step, due to colliding with three edges L_1, L_2, L_3. The response required for each collision is superimposed, and so three times the amount of response will be applied to *AB*, which results in an over-response.

In this paper, a semi-global constrained optimization method is proposed and combined with a momentum method, which will be described in detail in section 5.

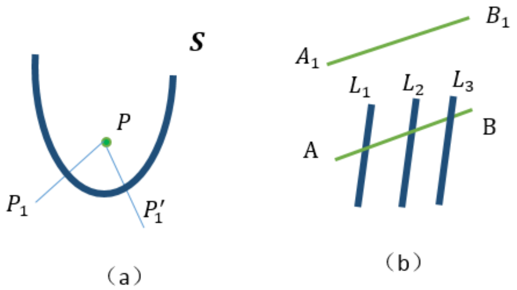


Figure 5: Problems in Response

5 Semi-Global Constrained Optimization

5.1 Constraint Equation

In order to solve collision response based on constraint, we first should establish constraint equations and then transform it to the extreme value problem under constraint, and finally use the Lagrangian Multiplier Method to solve it. As shown in Fig. 2 (a), if the point P falls on the triangle ABC at some moment, the following equation (12) is satisfied, where $N(t)$ is the normal vector of the triangle ABC.

$$N(t) \cdot AP(t) = 0 \qquad (12)$$

The coordinate of the point P is p, and the left side of the equation is denoted as $g(p)$. If there is no penetration, p should be outside the triangle ABC and meet the constraints of $g(p) \geq 0$. If P falls on the triangle ABC, $g(p) = 0$. If P falls inside the triangle ABC, a penetration occurs, $g(p) < 0$. The equation (12) can be further transformed into velocity constraint form (17). Where $J = \frac{\partial g}{\partial p}\frac{\partial p}{\partial v}$, $g_0 = g(p_0)$, p_0 is the initial position of the collision point.

$$J\Delta v \geq -\frac{1}{\Delta t} g_0 \qquad (13)$$

If an "edge-edge" collision occurs at some moment, then $N(t) \cdot AD(t) = 0$ should be satisfied. Where $N(t)$ is the result of the cross product of AB and CD. Similarly, the position constraint can be transformed into the velocity constraint form.

In the Signorini model [14], if pressure generates between two objects (denoted by $\lambda > 0$) by squeezing each other, they must be affixed together, that is, $g(p) = 0$. Conversely, if they are not in contact, there must be no pressure between them, that is, $\lambda = 0$. According to this, a complementary constraint equation (14) on the normal to the contact surface can be established. Where v^* is the velocity without constraint, and $\Delta v = v - v^*$.

$$0 \leq \lambda \perp J\Delta v \geq -\frac{1}{\Delta t} g_0 - Jv^* \qquad (14)$$

Next, we will consider the tangent constraints on the contact surface. Let v_{t_i} is the relative velocity along the tangential direction on the contact surface, and $t_i (\mathrm{i} = 1 \mathrm{or}\ \mathrm{i} = 2)$ is the direction of the two orthogonal directions on the contact surface. Under the friction, objects on the contact surface will keep the original movement direction or keep relatively static, and will not move in the opposite direction, and so $v_{t_i} \geq 0$. The friction force satisfies the Coulomb friction cone constraint, $\|\gamma_i\| \leq \mu\lambda$, where γ_i is the friction force in the direction of t_i and μ is the friction coefficient. The dynamic friction coefficient is approximately equal to the static friction coefficient. If the two objects have relative movement ($v_{t_i} > 0$), there must be sliding friction ($\|\gamma_i\| = \mu\lambda$). If they are relatively stationary ($v_{t_i} = 0$), there must be static or non-existent friction. The tangential constraint on the contact surface can be expressed:

$$\|\gamma_i\| \leq \mu\lambda \perp H\Delta v \geq -Hv^* \qquad (15)$$

$$\lambda \equiv \begin{pmatrix}\gamma_1 \\ \gamma_2\end{pmatrix}, \quad H = \begin{pmatrix}t_1 \\ t_2\end{pmatrix} \qquad (16)$$

Good collision response not only should avoid secondary collision, but also should ensure that the fabric deformation looks natural. The former requires the constraints on the contact surface should be met, while the latter requires that the state of motion of the particles do not change too violently. According to the above principle, the collision response problem can be transformed into the extreme value problem under the constraint condition:

$$\min \quad \frac{1}{2}\Delta v^T A \Delta v \qquad (17)$$

$$\text{s.t.} \quad J\Delta v \geq -\frac{1}{\Delta t} g_0 \qquad (18)$$

The speed under the constraint can be expressed as (19), which is combined with the complementary constraint conditions to establish linear equations. Moreover, the amount of velocity change can be calculated after the collision.

$$A\Delta v = J^T\lambda + H^T\gamma, \Delta v = v - v^* \quad (19)$$

5.2 Jacobi Iterative Method

In this paper, we use Jacobi Iterative Method to solve equations due to its simplicity. Shown as (20) and (21), D_A is the block diagonal matrix, L_A is the lower triangular matrix, U_A is the upper triangular matrix.

$$D_A\Delta v(i) = (L_A + U_A)\Delta v(i-1) + J^T\lambda(i) + H^T\gamma(i) \quad (20)$$

$$A = D_A - L_A - U_A \quad (21)$$

If the elastic force $\lambda(i)$ and the friction force $\gamma(i)$ are known, $\Delta v(i)$ can be solved by the formula iteration. The normal constraint condition represented by the above equation (14) can be converted into the equation (22), where $B_\lambda = JD_A^{-1}J^T$, $B_{\lambda\gamma} = JD_A^{-1}H^T$, $0 \le \lambda \perp B_\lambda\lambda \ge C_\lambda(i) - B_{\lambda\gamma}\gamma$

$$0 \le \lambda \perp B_\lambda\lambda \ge C_\lambda(i) - B_{\lambda\gamma}\gamma \quad (22)$$

The tangential constraint can be transformed into (23), where $B_\gamma = HD_A^{-1}H^T$, $C_\gamma(i) = -Hv^* - HD_A^{-1}(L_A + U_A)\Delta v(i-1)$

$$\|\gamma\| \le \mu\lambda \perp B_\gamma\gamma \ge C_\gamma(i) - {B_{\lambda\gamma}}^T\lambda \quad (23)$$

In equations (22) and (23), two quantities are unknown, which are the elastic force λ and the friction forceγ. When calculating, the friction γ is initialized to 0, using equation (22) to solve elasticλ; Then λ is substituted into Eq. (23) to solve γ; Then γ is substituted into Eq. (22) to solve the elastic force λ. This solution is iterated until the iteration results converge. Then the result is substituted into equation (20), and Δv is solved iteratively.

5.3 Semi-Global Constrained Optimization

In the global constraint method, it is necessary to continuously add the new detected collisions to the existing constraint set and then do an iterative solution. As the constraints increase, the computation time will increase. Therefore, in order to improve the simulation speed of the system, we should try to reduce the number of constraints. One of the improvements is to keep only the most recent collisions at each iteration, that is, only the newly detected collision will be as a constraint to solve the constraint system. Although this method can rapidly reduce the size of the collision constraint set, there are obvious limitations. Since each iteration in constraint calculation is only equivalent to the local collision response, it is easy to cause the case shown in Figure 1. It is not possible to get a response that satisfies the global condition.

Different from the above method, we adopt a new semi-global constraint optimization method. First, a collision impact factor is set for each collision. The larger the value, the greater the influence of the collision. In the "detection - response" of the iterative calculation process, the influence of collision is dynamically changing. When the impact exceeds the specified response threshold, the marked collision is added to the constraint set. On the contrary, when the influence decays to less than the specified threshold, the marked collision is discarded from the constraint set. Because the constraint set generated by this method is not exactly equivalent to the classical global constraint method, it is called semi-global constraint optimization method.

The constraint condition of the semi-global constraint method are based on the dynamic change of influence. Suppose the constraint set size is N, the initial value of the j-th impact is E_j^0, and the influence is expressed as E_j^i in the i-th iteration, the influence of the collision in the $i+1$ -th iteration is calculated as follows. If the collision does not occur in the $i+1$ -th iteration, its influence decays to the previous α times, that is, $E_j^{i+1} = \alpha \cdot E_j^i$, where α is the attenuation coefficient and $\alpha \in (0,1)$. If the collision still occurs in the $i+1$ iteration, its influence is $E_j^{i+1} = \alpha \cdot E_j^i + E_j^0$. When $E_j^{i+1} \ge E_j^0$, it remains in the constraint set.

Suppose the initial value E_j^0 of the collision influence is 1 and the response threshold is also 1, the specific steps of a frame animation calculation are as follows:

Step1: $i = 0$, initial collision impact of the initial value of $E_j^0(j = 1,2,\dots,N)$, set the response threshold is 1;

Step2: $i = i + 1$;

Step3: optimization computing;

Table 1 New collision update strategy

		1	2	3	4
Number of iterations		1	2	3	4
Whether the collision is detected		Y	Y	N	N
Dynamic influence factor		1	1*0.7+1=1.7	1.7*0.7=1.19	1.19*0.7=0.833
Whether to respond to this collision	Semi - global constraints		√	√	×
	Only consider new constraints		√	×	×
	Global constraint		√	√	√

Step5: For the original collision r, if it continues to occur, then $E_r^{i+1} = \alpha \cdot E_r^i + E_r^0$, otherwise, let $E_r^{i+1} = \alpha \cdot E_r^i$. If $E_r^{i+1} \geq E_r^0$, it is kept in the constraint set, otherwise it is discarded;

Step6: For the first occurrence of a new collision k, let $E_k^{i+1} = E_k^0$ and add it to the constraint set;

Step7: Reset constraint set size N. Turn to Step2;

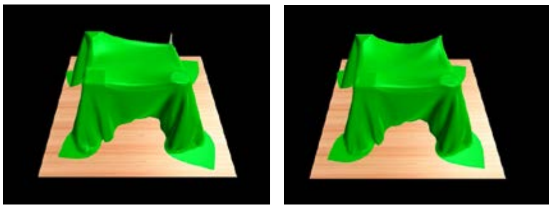
Step8: End;

To explain this strategy better, give a simple example. Assume that the attenuation factor $\alpha = 0.7$. In the first iteration, a new collision occurs, its influence is set to 1, which meet the response condition and so this collision is added to the constraint set. In the second iteration, if the collision occurs again, its influence becomes 1.7, which still meet the response condition and so continue to remain in the constraint set. In the third iteration, the influence decay to 1.19. Although the collision does not occur, the response is still needed. In the fourth iteration, the influence attenuation decay to 0.833 <1. The collision still does not occur and no response is needed. It follows that if a collision continues to occur in the first few iterations, it will be still kept in the constraint set and responded even if it does not appear in the last iteration. This approach allows some complex and potential collisions to be picked out, avoiding collision situations that cannot be solved in an infinite loop. Table 1 lists the collision set update strategy under three different constraint methods.

6 Experimental Results and Analysis

In this paper, all experimental environments are Intel Core 2 Duo E8400, 3GHz CPU, 4GB of memory, NVIDIA GeForce9800GTS GPU.

The first set of experiments are square cloth animations, which is used to verify the effectiveness of critical contact processing in collision detection. The first experiment is a simulation of a square cloth falling on four pillars, which are cylindrical, thin rectangular, triangular prism and cone. Collisions occur mainly between the cloth and the four pillars and between the cloth and the floor. If the close state in the collision detection is neglected, the square cloth is penetrated by the circular vertex and the animation is distorted. The animation result is shown in Figure 6(a). After critical contact handling using the method in section 3 is adding into the system, the penetration is effectively avoided, shown in Figure 6(b).



(a) (b)

Figure 6: Square Cloth falling on the pillars

The second experiment is twisting motion of square cloth. A piece of cloth is rotated while fixed on the upper. During this piece of cloth continues to rotate and twist, the twisted areas overlap and close fit together. If critical contact processing is not added in the system, the animation diverges severely due to too close of element-pair in twisted region. After close state handling technique is added in the system, the divergence is avoided. The animation result is shown in Fig 7, which makes the collision detection more robust and the final animation is more realistic.

The second set of experiments is to compare response efficiency. The global constraint method and the improved semi-global constraint method are used to simulate

Table2 Cloth Falling on the Rotating Ball

Fabric model	The method used	Collision detection time /ms	Collision response time/ms
50×50 square cloth	Proposed method	309	168
	Global constraint method	319	184
100×100 square cloth	Proposed method	1734	1179
	Global constraint method	1938	2966

different sizes of square cloths falling on a rotating sphere. The collision time is recorded. The sizes of two piece of cloth are 50×50 and 100×100 respectively. For the same size of

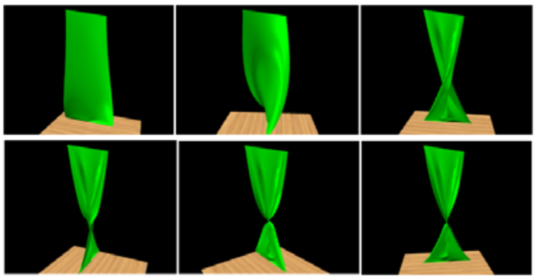
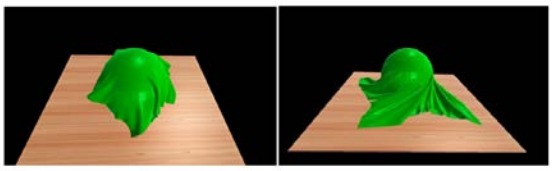


Figure 7: Square Cloth Twist

cloth, two methods can get almost the same effect, but the simulation time is different. For the 50×50 cloth, semi-constraint method takes 477ms, which is less than 503ms of the global constraint method. For the 100×100 cloth, the method of this paper costs 2913ms to deal with the collision in each frame, while the global constraint method costs 4904ms. The efficiency of collision processing is improved by 40%, and the efficiency of the response is increased by about 60%. It shows that the higher the precision of the cloth model, the more improvement of the response efficiency. The final animation results are shown in Figure 8.



(a) 50×50 of the square cloth (b) 100×100 of the square cloth

Figure 8: Square Cloth on a Rotating Ball

The third set of experiments is dressing human animation. In this experiment, a standard female model dressed in skirt walked on the T-stage, and human animation time duration is 8.5s. The multi-layer skirt includes 59,000 vertices. The entire animation not only contains the collision processing between the human body and the skirt, but also a large number of self-collision processing. The second skirt includes 12,150 vertices. The collisions mainly occur between the human body and skirt. The two animation results are shown in Figure 9 and Figure 10.

7 Conclusion

In this paper, the critical contact state including "vertex-surface" and the "edge-edge" on the cloth is analyzed in details. The solutions of three most common boundary states are given, which effectively overcomes the adverse effect caused by collision missing.

Figure 9: Multi-Skirt Dress Animation

Figure 10: Gown Dress Animation

The robustness of collision detection is improved. In addition, a semi-global constraint optimization method for fabric collision response is proposed. By imposing a dynamic influence factor on each collision constraint, some complex and potential collisions are picked out, while other invalid collisions are abandoned. It not only ensures that the collision is not missing, but also simplifies the constraint set and so effectively improves the system efficiency.

Based on the proposed method, square cloth animation and the dressed human animation were simulated. Both of them can effectively avoid the collision missing. Moreover serious animation distortion due to divergence or penetration is avoided, which ensures the realistic and stability of the cloth.

Reference

[1] M. Moore and J.Wilhelms. Collision Detection and Response for Computer Animation.ACM SIGGRAPH Computer Graphics, 22(4), 1988.

[2] S.J.Adelson and L.F.Hodges. Generating exact ray-traced animation frames by reprojection[J]. IEEE Computer Graphics and Applications, 1995, 15(3):43-52.

[3] P.Volino and N. M.Thalmann.Developing Simulation Techniques for an Interactive Clothing System. Proc. of Virtual Systems and Multimedia, pages 109-118, 1997.

[4] M.Teschner, S.Kimmerle, B.Heidelberger, G.Zactunann, Laks RaghuPathi, A.Fuhrmalin, Marie-Paule Cani, F.Faure, N.Magnetat Thalmann, W.Strasser, and P.Volino. Collision Detection for Deformable objects[J]. Computer Graphics Forum,2005,24(l):61-81.

[5] W.S.-K. Wong and G. Baciu.A Randomized Marking Scheme for Continuous Collision Detection in Simulation of Deformable Surfaces. In ACM International Conference on Virtual Reality Continuum and Its Applications (VRCIA),pages 181-188, 2006.

[6] S. Curtis, R. Tamstorf and D. Manochal. Fast Collision Detection for Deformable Models using Representative-Triangles. In Proceedings of the 2008 symposium on Interactive 3D graphics and games, pages 15-17, 2008.

[7] Changxi Zheng and Doug L. James, Energy-based Self-Collision Culling for Arbitrary Mesh Deformations, ACM Transaction on Graphics (SIGGRAPH 2012), 31(4), August, 2012.

[8] Min Tang, Ruofeng Tong, Zhendong Wang, and Dinesh Manocha. Fast and Exact Continuous Collision Detection with Bernstein Sign Classification, ACM Transactions on Graphics, 33(6), Article 186 (November 2014), 8 pages (Proc. of ACM SIGGRAPH Asia), 2014.

[9] Huamin Wang. 2014. Defending Continuous Collision Detection against Errors. ACM Transactions on Graphics (SIGGRAPH), vol. 33, no. 4.

[10] B. Mirtich, and J. Canny. Impulse based Simulation of Rigid Bodies. Proceedings of the 1995 symposium on Interactive 3D graphics, pages 181-ff,1995.

[11] Harmon D, Vouga E, Tamstorf R, et al. Robust treatment of simultaneous collisions[J]. ACM Trans. Graph. 2008, 27(3): 1-4.

[12] M.A. Otaduy, R. Tamstorf, D. Steinemann, and M. Gross. Implicit Contact Handling for Deformable Objects[J]. Computer Graphics Forum,2009,28(2):559-568.

[13] R.Bridson, R.Fedkiw and J. Anderson. Robust Treatment of Collisions, Contact and Friction for Cloth Animation. Proc.of ACM SIGGRAPH, pages 594-603, 2002.

[14] Duriez C, Dubois F, Kheddar A, Andriot C.Realistic haptic rendering of interacting deformable objects in virtual environments, IEEE Trans Vis Comput Graph. 2006 Jan-Feb;12(1):36-47