Collision Resolutions in Cloth Simulation

Suejung Huh, Dimitris N. Metaxas, and Norman I. Badler
Center of Human Modeling and Simulation Lab, 109 Moore Bldg.
University of Pennsylvania
Philadelphia, PA 19104-6839, USA
(215) 898-1976, fax (215) 573-7453
{suejung,dnm,badler}@graphics.cis.upenn.edu

Abstract

We present a new collision resolution scheme for cloth colli-sions. Our main concern is to find dynamically convincing resolu-tions, i.e. positions and velocities of cloth elements, for any kinds of collisions occuring in cloth simulation (cloth-cloth, cloth-rigid, and cloth-cloth-rigid). We define our cloth surface as connected faces of mass particles where each particle is controlled by its in-ternal energy functions. Our collision resolution method finds ap-propriate next positions and velocities of particles by conserving the particles' momentums as accurately as possible.

Cloth-cloth collision resolution is a special case of deformable N-body collision resolution. So to solve deformable N-body col-lision resolutions, we propose a new collision resolution method, which groups cloth particles into parts and resolves collisions be-tween parts using the law of momentum conservation. To resolve collisions, we solve a system of linear equations derived from the collision relationships. A system of linear equations is built us-ing a scheme adapted from the simultaneous resolution method for rigid N-body collisions [1]. For the special case where we can find cyclic relationships in collisions, we solve a system of linear inequalities derived from the collision relationships.

1. Introduction

Collision handling in Computer Graphics has two phases. One is to detect collisions and the other is to resolve collisions.

In cloth collision detection, the computation time to detect col-lisions is not negligible because the number of geometrical entities (nodes, faces, edges) the collision detection algorithm has to han-dle is considerable (over 10,000 particles for regular attire). For this reason, several approaches have tried to expedite the collision detection processes [16, 3].

Collision resolution is to find the correct next positions and velocities of colliding objects. Cloth resolution methods so far have found non-penetrating positions, velocities and accelerations of cloth surface particles [14, 3, 15]. This scheme works fine for cloth-rigid collisions and for the special case of cloth-cloth col-lisions where the dynamic interactions between cloth surfaces in cloth-cloth collisions do not have to be noticeable.

Volino *et. al.* [15] applied the conjugate gradient method to find the actual particles' positions where a group of particles are colliding into each other. By preserving barycentric relationships of collision entities, their method resolves collisions where nu-merous cloth surfaces are colliding together as a group, which is a novel way to resolve multiple collisions at once. However it does not conserve the momentum of cloth surfaces in cloth-cloth collisions. Another method for cloth-cloth collisions has been pro-posed by Provot [12], which resolves collisions by giving an av-erage velocity to all the particles of collisions. Provot's method is easy to implement but it cannot give proper visual effect of colli-sions since we cannot get dynamic interactions between particles once the particles collide into each other.

Cloth-cloth collision resolution is a special case of deformable N-body collision resolution. To solve deformable N-body colli-sion resolutions, we propose a new collision resolution method which gives a visually reasonable response by ensuring the conser-vation of N-body momentums. Our cloth system is particle-based, as many systems are in other cloth research groups [14, 6, 12, 3]. To resolve collisions, we first divide the colliding particles into parts and build a system of linear equations based on the collision relationships between these parts. Then we solve the whole system using the law of N-body momentum conservation. The system of equalities is based on the scheme adapted from the simultaneous resolution method for rigid N-body collisions proposed by Baraff [1]. However his original inequality relationships between relative velocities before and after collisions are purely heuristic, which may not be physically correct. Though this physical inaccuracy has been an inherent problem of simultaneous collision resolu-tions, it appears to give graphically agreeable results. Hence with the help of the law of N-body momentum conservation, we found the results of our resolutions are visually acceptable.

In the next section, we discuss the issues involved in collision detection, especially to define a way to store collision information. In Sec. 3, we propose our collision resolution scheme. In Sec. 3.1 and 3.2, we examine some special cases of collisions which need some prior processes before we actually resolve collisions. From Sec. 3.3 to Sec. 3.5, the actual collision resolution method is presented. Finally Sec. 4 discusses results.

2 Collision Detection

A widely used method for detecting cloth collisions is to put small repellent proximity forces between the cloth surface and the rigid or cloth surface [4, 14, 3] while the actual collisions are tested with pairs of particle-face or face-face of the current po-sitions. When objects are moving fast, however, these preventive proximity forces cannot prevent collisions since a particle can pass through the proximity violation region during one integration time step. This problem can be negligible when the integration time step is very small, so we rarely have those pass-through cases.

122

However since implicit integration methods [3, 5, 7] have been used for cloth simulations, the time step has become larger, sometimes as large as to meet the frame rate of $\frac{1}{30}$ sec/frame. Consequently this penetration problem is no longer negligible, so we use the swept volume approach [13, 8] which will be described in the next section.

2.1 Collision detection using swept-volumes

A *swept volume* is a volume made by two sets of positional entities of a face - one at time t and one at time $t + \Delta t$. Connecting these old and new positions of all particles in a face gives us a volume. Any collision happening within an integration time step always can be detected by this swept volume method, unless the motions of faces are highly rotational. An interesting case is where the faces are not actually intersecting but two swept volumes report a intersection anyway. Though this case is not an actual collision, it happens only when two faces are very close. Hence we resort to the collision report of this case, since we consider this case as a violation of the proximity law. We use classical edge-polygon detection algorithms to detect collisions among swept volumes. We use this swept volume approach for cloth and the dynamic rigid body alike, but for the non-moving rigid body only the surface faces are used for collision detection. In addition, we add proximity regions to the normal directions of faces of a swept volume to add proximity violation regions.
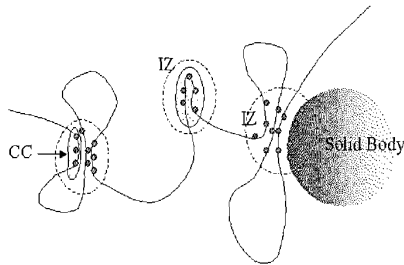
2.2 Saving detected collisions

Figure 1. Impact Zones

Though detected collisions are reported as pairs of face-face, we cannot respond to each collision individually since these individual responses may introduce another new collision or one face may possibly be related to several other collisions. So we save all detected collisions in a data structure, i.e. a set of *zones of impact* [12] during the collision detection phase. All stored detected collisions will be resolved comprehensively by the rule described in the next section. Originally a zone of impact (IZ) is an area where multiple self-collisions occur [12]. We extend Provot's definition of an IZ to an area where collisions happen, including collisions against bodies and self-collisions.

An object O is a set of particles, faces, and edges, where faces and edges are defined based on the positions of particles by the rule comprising cloth surface. An area A is a subset of O such that all the particles and edges constituting a face ϕ in A are members of A. An area A is called *separable* when, for all faces $\phi \notin A$, ϕ does not collide with any face in A. An area A is called *colliding* when, for all faces $\phi \in A$, ϕ collides with at least one face in A. An IZ

is a separable colliding area. Fig. 1 shows a cross section view of a collision situation where three cases of IZs are shown by dotted circles. An area A is called *visitable* when, for all particles $P \in A$, P can be encountered by traversing from any other particle in A using edges in A. Otherwise, the area A is *non-visitable*. We call a visitable subset area of an IZ a *collision cluster* (CC). When a face-face collision is detected, the entities of each colliding face ϕ (the particle and the edges of ϕ, and ϕ itself) are inserted into a CC, where the CC can be encountered by traversing from the particles in ϕ using only edges in ϕ. When there is no such CC, ϕ becomes a CC. When two or more such CCs are found, these CCs are merged into one CC connected by ϕ. In Fig. 1, a CC is denoted as a shaded area.

3 Collision Resolutions

Numerous approaches [14, 12, 3, 15] have been introduced for cloth collision resolution: the correct next positions and velocities of colliding cloth particles. So far, however, no cloth collision resolution method which considers cloth-cloth momentum conservation has been introduced, while we cannot achieve realistic cloth interactions in cloth animation without conserving cloth-cloth momentums. Having this characteristic is visually distinctive when cloth surfaces are moving fast and interact with each other.

For rigid N-body collisions by graphics and robotics groups [11, 1, 10, 9] and for flexible-rigid collision resolution [2], several approaches have been suggested. But they are not directly applicable for deformable N-body collision resolutions, which is the case cloth requires.

Cloth resolution methods so far compute non-penetrating positions, velocities or accelerations of particles [14, 3, 15], which work fine for the collisions against fixed bodies. Using these methods, however, we cannot achieve visually satisfying dynamics of cloth-cloth collisions. Adjusting particle orientations after collision resolution as suggested by [14] to sustain the geometrical consistency of colliding faces also does not warrant reasonable dynamic movements of cloth-cloth collisions.

Handling collisions in an IZ as a bundle, proposed by Provot [12], also does not give a proper visual effect. Since all particles are given the same velocities after collision resolution, particles tend to get stuck once they collide into each other in an IZ.

We propose our cloth collision resolution method which resolves simultaneous collisions while ensuring conservation of momentum as accurately as possible. Since simultaneous resolution does not blindly resolve a collision without considering neighboring collisions within an IZ, we do not introduce any new collisions while resolving a collision.

The collision resolution module receives collision information as a set of IZs as described in Sec. 2.2. Within an IZ, where possibly multiple collisions reside, collisions are resolved simultaneously as described in Sec. 3.4. After we handle each IZ separately, we check whether any new collisions between IZs are introduced by collision resolutions, and handle them if there are any. In an IZ, we first check whether it has CCs from rigid bodies. In case we do not find any CCs from rigid bodies, the collisions in that IZ are categorized as cloth-cloth collisions. If we find CCs from rigid bodies in an IZ, we extract them temporarily from the IZ so that only cloth-cloth collisions remain in the IZ. After resolving these cloth-cloth collisions, we take care of cloth collisions against rigid bodies so that the resolutions against rigid bodies will be done on top of the result of self-collision resolutions. This sequence of resolutions is chosen to avoid the case where self-collisions are ignored while collisions against rigid bodies are handled.

123

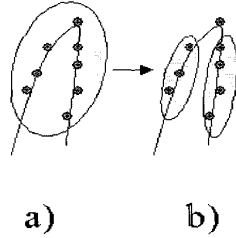## 3.1 Single Collision Cluster in an IZ

Figure 2. Segmenting a Collision Cluster into parts

For some cases, an IZ has only one CC (for example, in the case of extreme bending). When an IZ has only one CC, we cannot handle the particles in that CC as a bundle as usual. Since the particles in that CC will stick together after resolution, the movements of cloth would not be natural and satisfactory. Hence we divide one CC into parts so that we can find proper collision responses within these parts. Segmenting one CC into parts is performed by identifying border edges. A border edge is an edge where we identify a "significant" bending between two faces adjoining in that edge. Empirically a bending between two faces with face normals N_1 and N_2 is identified as significant when $N_1 \cdot N_2 < 0$. Eliminating the particles in the border edges segments a CC into several sub-CCs as shown Fig. 2.

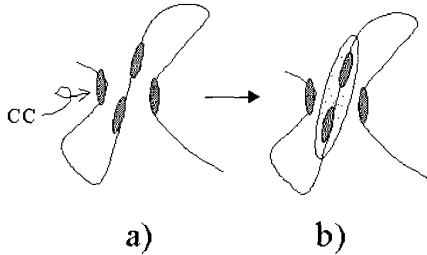## 3.2 Merging Multiple CCs in an IZ

Figure 3. Merging Collision Clusters

When an IZ has three or more CCs, we reduce the total number of CCs by merging closely located CCs. This merger is performed to prevent undesirable collision resolution. If CCs are closely located, it means the cloth patches represented by these CCs are closely located. We do not want to handle closely located CCs separately since it might instantly introduce instabilities to the system by allowing closely located CCs to have different velocities. However, there is an exception. When we find a significant bending between these closely located CCs, we have to resolve collisions between these CCs by handling them separately. Bending between CCs is considered *significant* in the same way as in the case of bending between faces. We do not want to handle closely

located CCs separately except for the case where the bending is significant (CCs are considered to be closely located heuristically when they can be connected using at most two edges which are not members of both CCs). Hence the candidates of the CC merger are the CCs closely located, where we do not witness any significant bending between the CCs. Fig. 3 shows this merging process. After merging, we still possibly have more than two CCs. Multiple CC collision resolution is discussed at length in Sec. 3.4.
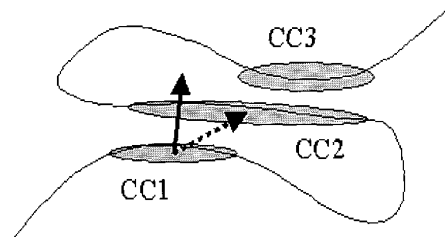
3.3 Collision resolution for two CCs

Figure 4. Determining the colliding direction. Showing two different colliding directions where the solid arrow shows the direction computed by CC face normals and the dotted arrow shows the direction computed by connecting centers of masses of CCs.

By definition, an IZ is a set of CCs. Since we pre-processed a single CC IZ previously, we assume an IZ always has two or more CCs. The important part of the collision resolution of these multiple CCs is to find the proper directions of collisions. Collision direction is a direction to which two CCs collide into each other. Since the velocities of CCs after collision are computed based on this collision direction, finding the correct collision direction is important to achieve proper visual effect of collisions. In the case of the two billiard ball collision, the collision direction is computed by connecting the two ball centers of mass. But in cloth-cloth collisions, connecting two centers of CC masses is not a proper way to decide the collision direction. We choose the collision direction to be the average direction of the two face normals of colliding CCs. To have the proper average direction, the CC face normals, N_1 and N_2, have to be properly signed as $N_1 \cdot N_2 > 0$. The face normal of a CC is the average normal of all faces in the CC. Fig. 4 shows two different colliding directions between two collision clusters CC1 and CC2; one by the average face normal and the other by connecting centers of masses.

We handle a CC as a sphere mass where the diameter of the sphere reflects the minimum proximity region. This approach serves us well empirically. The velocity of a CC is defined as the average velocity of all particles in that CC.

3.4 Simultaneous collisions

When we have collisions of three or more CCs in an IZ, it is not straightforward to resolve the collisions.

As has been discussed in multiple collisions of rigid bodies, we can think of two ways to solve this multiple cloth collision problem. One way of resolving these multiple collisions is to handle

124

them as staggered collisions [11, 10]; the other way is to handle them as simultaneous collisions [1]. The staggered collision approach handles multiple collisions as a series of single collisions [11] or a group of desynchronized collisions [10]. The simultaneous collision approach treats multiple collisions as simultaneous collisions within a time-step. The staggered collision approach gives us a more physically correct solution than the other. In the synchronized staggered collision method, we have to find the first collision among multiple collisions. After we resolve it, we march the time step until we find the next collision. Then we repeat the same procedure. This whole process is not only computationally expensive but also we have to consider the possibility that the resolution of a collision can create new multiple collisions, which we have to employ another strategy to resolve. In the desynchronized staggered collision method, we identify groups of collisions, and redefine the integration front-end by allowing time desynchronization. In addition to the substantial computational expense and complexity, the visual advantage of those staggered methods is not considered significant compared to that of the simultaneous collision method.

The simultaneous collision handling method, proposed by Baraff [1], resolves multiple rigid body collisions by solving a system of linear inequalities, where the system of linear inequalities is based on the colliding relationships between rigid objects. As we treat CCs as mass balls, we adapt this approach for the cloth collision resolution problem. A CC is considered as a body.

If N is the number of colliding bodies, v_i^- is the velocity of the body i before collision, and v_i^+ is the velocity of the same body after collision, given v_i^-, we have to find v_i^+ for all i $1 \leq i \leq N$.

When CC bodies are considered as vertices, an edge exists between two vertices where the bodies represented by those two vertices collide. We call the resultant graph a *collision graph*. When the collision graph of an IZ has a loop, we call the collisions in the IZ *cyclic*. When N bodies are colliding without having any cyclic collision, we can build a system of $N - 1$ linear equations based on the changes of relative velocities of each colliding pair, such as

$$v_{i,j}^+ = -C_e v_{i,j}^- \tag{1}$$

where CC_i is the ith CC, $v_{i,j}^-$ is the relative velocity of the CC_i and CC_j before collision, $v_{i,j}^+$ is the relative velocity of the same pair after collision, and C_e is an elastic coefficient.

In addition, the law of N-body momentum conservation says,

$$m_1 v_1^+ + m_2 v_2^+ + \cdots + m_N v_N^+ = m_1 v_1^- + m_2 v_2^- + \cdots + m_N v_N^-. \tag{2}$$

Using eq. 1 and 2, we can deterministically find the solution of all v_i^+ for $1 \leq i \leq N$.

Since we use swept-volumes of faces to detect collisions, we handle fast moving cloth and rigid body objects. However a problem arises when collision resolutions of an IZ create new collisions against objects around the IZ. This happens when objects do not move fast enough to penetrate objects outside an IZ, but just fast enough to make the result of collision resolutions penetrate the proximity region of objects outside the IZ. To our relief, this case appears to be very rare. However we can resolve this case by maintaining the barycentric relationship between cloth surfaces and the newly introduced colliding entities. Apparently, in the worst case, this involves repetitious processes as we may introduce other new collisions when we resolve the current collisions.

3.4.1 Cyclic Collisions

For the special case where we observe cyclic collisions in an IZ, we build a system of linear inequalities based on the collision rela-

tionships between grouped particle parts. We find the feasible solution of the linear inequality system, while trying to minimize the energy we introduce into the simulation artificially. If an IZ has cyclic colliding relationships between CCs, we have N or more linear inequalities such that

$$v_{i,j}^+ \geq -C_e v_{i,j}^- \tag{3}$$

where the notations are as in eq. 1. This inequality relationship between the relative velocities before and after collision is an artificial relationship set up heuristically, not based on physics. This inequality relationship, first used for rigid body multiple collisions [1], appears to serve the graphical purpose well.

When we solve this system of inequalities, we want to minimize the difference between $v_{i,j}^+$ and $-C_e v_{i,j}^-$, which would represent the extra energy we introduce into the system. Hence, we define an objective function,

$$X = \sum_{i,j} v_{i,j}^+ - (-C_e v_{i,j}^-) \tag{4}$$

to be minimized when we solve the system. The system of inequalities with an objective function can be solved using a Linear Programming Method.

3.5 Moving and Fixed Rigid Bodies

If an IZ has CCs from rigid bodies (rigid CCs) along with CCs from cloth (cloth CCs), the collision resolutions against rigid bodies are performed after cloth-cloth collisions are resolved.

When an IZ has rigid CCs, collision responses are different based on whether rigid CCs are moving or fixed or a mixture of both.

If the rigid CCs in an IZ are all fixed, we handle particles in that IZ individually. Where N_{face} is the normal of a rigid face, V is a particle velocity, V_{normal} and $V_{tangential}$ are the normal and tangential components of V with respect to the rigid face, particles are considered separating if $V_{normal} \cdot N_{face} > 0$. Particles are ignored if they are not in the vicinity of a face in fixed rigid CCs, where the size of vicinity is the thickness of cloth. Furthermore, particles separating from the rigid bodies are also ignored. The new particle velocity V_{new} is $-C_e V_{normal} + C_f V_{tangential}$, where C_e is an elastic coefficient and C_f is a frictional coefficient.

If the rigid CCs in an IZ are all moving, we handle particles as a bundle as long as particles are in the vicinity of moving rigid CCs. We find the Δx and the velocity V_{rigid} of a moving rigid CC, where V_{rigid} is defined as the translational velocity of the center of mass of the moving rigid CC. Then the positions of all particles we have to handle will be incremented by Δx and the velocities of the particles will be updated as V_{rigid}.

If an IZ has both moving and fixed rigid CCs along with cloth CCs, collision resolutions against rigid bodies are done based on the proximities of particles to the rigid CCs. Cloth collision resolutions against rigid CCs will be computed based on the closest rigid CC. Collisions between rigid bodies (rigid-rigid) have to be handled independently from cloth collisions.

4 Results

Fig. 5 shows cloth-rigid and cloth-cloth collisions, where no friction was assumed between the rod and the cloth. Fig. 6 shows various kinds of collisions; cloth-rigid(fixed), cloth-rigid(moving), and cloth-cloth. All our simulations were done on SGI Octane

125

with R10000 CPU and R10010 FPU. For numerical integration, we used the CG method proposed by Baraff [3]. Details of simulations are noted in Tab. 1. Notably Δt in Fig. 6 (above) is 5 times bigger than that of Fig. 6 (below). This mainly comes from the fact that the fabric in Fig. 6 (below) is stiffer than the one in Fig. 6 (above).

5 Acknowledgments

References

[1] D. Baraff. Analytical methods for dynamic simulation of non-penetrating rigid bodies. *Computer Graphics (Proc. SIGGRAPH)*, 23(3):223–232, 1989.

[2] D. Baraff and A. Witkin. Dynamic simulation of non-penetrating flexible body simulation. *Computer Graphics (Proc. SIGGRAPH)*, 26(2):303–308, 1992.

[3] D. Baraff and A. Witkin. Large steps in cloth simulation. *Computer Graphics (Proc. SIGGRAPH)*, pages 43–53, 1998.

[4] D. Breen, D. House, and M. Wozny. Predicting the drape of woven cloth using interacting particles. *Computer Graphics (Proc. SIGGRAPH)*, pages 365–372, 1994.

[5] M. Desbrun, P. Schroder, and A. Barr. Interactive animation of structured deformable objects. *Graphics Interface, Kingston, Canada*, June 1999.

[6] B. Eberhardt, A. Weber, and W. Strasser. A fast, flexible particle-system model for cloth draping. *IEEE Computer Graphics and Applications*, 1996.

[7] Y.-M. Kang, J.-H. Choi, and H.-G. Cho. Fast and stable animation of cloth with an approximated implicit method. *Computer Graphics International, Geneva, Switzerland*, June 2000.

[8] E.-A. Karabassi, G. Papaioannou, T. Theoharis, and A. Boehm. Interaction test for collision detection in particle systems. *ACM Journal of Graphics Tools, AK Peters, Ltd.*, 4(1):25–37, 1999.

[9] D. Marhefka and D. Orin. Simulation of contact using a nonlinear damping model. *Proc. IEEE Int. Conf. On Robotics and Automation, Minneapolis, MN*, 1996.

[10] B. Mirtich and J. Canny. Impulse-based real-time dynamic simulation. *Proc. Workshop on Algorithmic Foundations of Robotics, K. Goldberg et al. (eds.), AK Peters*, 1994.

[11] M. Moore and J. Wilhelms. Collision detection and response for computer animation. *Computer Graphics (Proc. SIGGRAPH)*, 22:289–298, 1988.

[12] X. Provot. Collision and self-collision handling in cloth model dedicated to design garments. *Proc. Graphics Interface*, pages 177–189, 1997.

[13] S. Raab. Controlled perturbation for arrangements of polyhedral surfaces with application to swept volumes. *Annual Symposium of Computational Geometry, Miami, FL*, pages 163–172, June 1999.

[14] P. Volino, M. Courchesne, and N. Magnenat-Thalmann. Versatile and efficient techniques for simulating cloth and other deformable objects. *Computer Graphics (Proc. SIGGRAPH)*, pages 137–144, 1995.

[15] P. Volino, M. Courchesne, and N. Magnenat-Thalmann. Accurate collision response on polygonal meshes. *Proc. of Computer Graphics*, pages 179–188, 2000.

[16] P. Volino and N. Magnenat-Thalmann. Efficient self-collision detection on smoothly discretized surface animations using geometrical shape regularity. *Eurographics*, 13(3):155–166, 1994.

Simulation	Δt	ST	CD(%)	CH(%)	Int	N_p	l_{edge}(cm)
Fig. 5(above)	0.0005	0.36	.20(56)	0.07(19)	0.07(20)	961	1
Fig. 5(below)	0.0001	0.22	.099(45)	0.039(18)	0.07(32)	961	1
Fig. 6(above)	0.0005	1.22	.98(80)	0.12(10)	0.1(8)	1681	1
Fig. 6(below)	0.0001	1.15	.90(78)	0.10(9)	0.1(9)	1681	1

Table 1. Simulation logs - Units are in seconds unless noted otherwise. ST is the average Simulation Time for one time step, CD is the average Collision Detection time, CH is the average Collision Handling time(including re-integration), Int is the average Integration time, Leftover time is used for displaying and saving frames, etc., N_p is the number of particles used. l_{edge} is the length of a warp or weft directional edge.
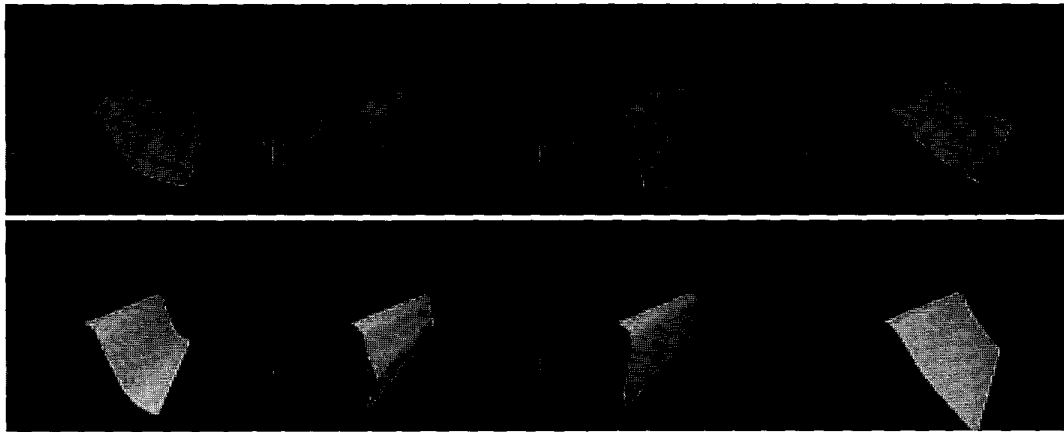

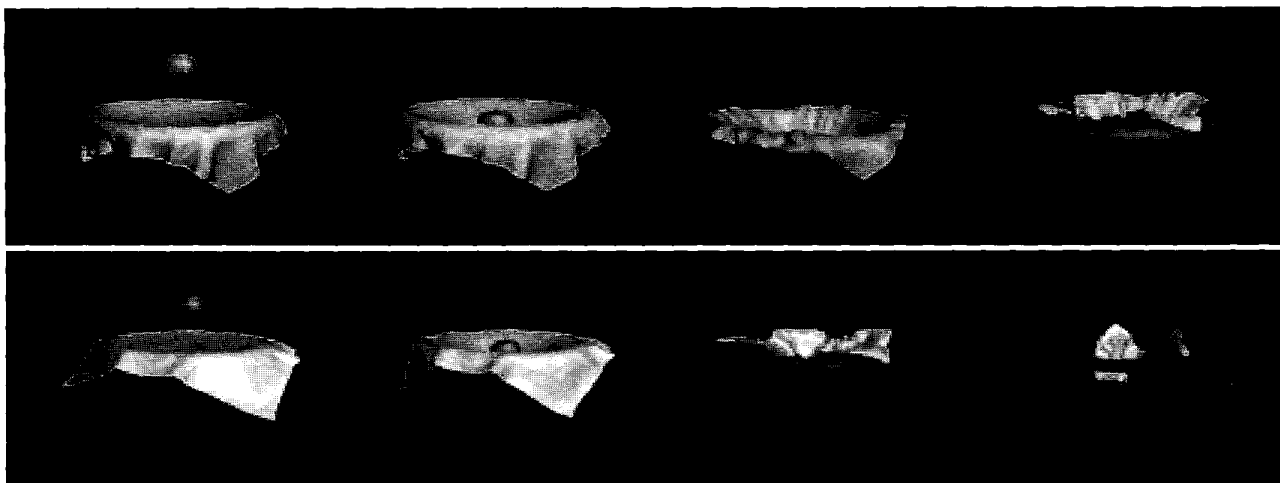Figure 5. Movements of two different types of fabrics

Figure 6. Dropping balls onto two different types of fabrics

127