Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior

Xavier Provot

Institut National de Recherche en Informatique et Automatique (INRIA)
B.P. 105
78153 Le Chesnay Cedex
France

Xavier.Provot@inria.fr
http://www-rocq.inria.fr/syntim/research/provot/

Abstract

This paper describes a physically-based model for animating cloth objects, derived from elastically deformable models, and improved in order to take into account the non-elastic properties of woven fabrics. A cloth object is first approximated to a deformable surface composed of a network of masses and springs, the movement of which is evaluated using the numerical integration of the fundamental law of dynamics. We show that when a concentration of high stresses occurs in a small region of the surface, the local deformation becomes unrealistic compared to real deformations of textiles. With such an elastic model, the only solution to decrease these deformations has been so far to increase the stiffness of the deformed springs, but we show that it dramatically increases the cost of the algorithm. We present therefore a new method to adapt our model to the particularly stiff properties of textiles, inspired from dynamic inverse procedures.

Résumé

Cet article décrit un modèle physique d'animation des tissus, variante des modèles élastiques déformables, et amélioré de façon à prendre en compte les propriétés non élastiques des textiles. Nous modélisons tout d'abord une pièce de tissu par une surface déformable, constituée d'un réseau de masses et de ressorts. Son mouvement est évalué grâce à l'intégration numérique de la loi fondamentale de la dynamique. Nous montrons que lorsqu'une forte concentration de contraintes apparaît à certains endroits de la surface, la déformation locale y devient irréaliste comparée aux déformations rencontrées dans les tissus réels. Avec un tel modèle élastique, la seule solution permettant d'atténuer cette déformation était jusqu'à présent d'augmenter la raideur des ressorts déformés, mais nous montrons que ceci faisait croître dramatiquement le coût de l'algorithme. Nous présentons donc ici une nouvelle méthode permettant d'adapter notre modèle aux propriétés particulièrement rigides des textiles, inspirée des procédures de dynamique inverse.

Keywords: Physically-based models, deformable surfaces, cloth animation, rigid behavior.

1 Introduction

1.1 Background

Woven fabrics have been widely studied in computer graphics in order to find appropriate models describing their particular properties, namely their static behavior (e.g. drape) and their dynamic behavior (e.g. buckling propagation). Early studies can be found in [1, 2, 3, 4], but regarding cloth animation with which we were mostly concerned, physically-based models have proved to be both the most efficient and realistic. Among the physically-based models used in cloth animation, *elastically deformable models* have been used successfully in order to give a representation of the behavior of various cloth objects such as flags, tablecloths, or even garments dressing synthetic actors [5, 6, 7].

1.2 Realism

However, one of the problems encountered in this kind of modelization is that woven fabrics are far from having ideal elastic properties. This is why, under certain conditions and stresses, these elastic models behave more like sheets of rubber, or gum, than like textiles. This behavior occurs especially when the elastic model is subject to high constraints, and therefore to *high "super-elastic" deformation rates*. Such high constraints do not appear in

many of the normal uses of textiles (tablecloth, ample clothes,...), and this is why the use of an elastic model for cloth animation is valid. But this is not the case for flags, or hanging sheets[1], where constraints are often concentrated at the hanging point(s) of the piece of cloth.

In these cases, when high contraints should lead to lower deformation rates, cloth animation models should be much *stiffer* than the elastic models that have been implemented so far. But the numerical solution of these models is evaluated by the means of a sampling rate in time which increases when stiffness increases—the higher the stiffness, the higher the cost of the algorithm.

This is why some attempts have been made to give up elastically deformable models and to use instead a network of rigid rods of fixed length [9], the movement of which can be computed thanks to the advances in constraint problems. This computation remains nevertheless costly when the number of rods increases, and the method gives free course to uncontrolled shear deformations unlikely to occur in woven fabrics. This could be handled by adding springs, but then the same "super-elastic" problem would remain, though this time only for shear deformation. From a different viewpoint, D. E. Breen also proposes, in [8], to use particle systems. But he only applies this method to the static behavior (the drape) of cloth objects, and not to cloth animation. Besides the computational speed of his algorithm is very slow.

1.3 Preview

In our approach, we start in section 2 with a model composed of masses and springs, which can be considered as a variant of elastic models. The specificity of this approach is that, unlike in [5, 6, 7], the model is not considered as a continuous surface that will have to be discretized, but rather as a discrete structure of elements where each mass-point and each spring can be handled individually. In sections 3 and 4, we describe the inconvenient "super-elastic" property of this model and why this problem cannot be solved by increasing stiffness. In section 5, we then introduce *constraints* on the deformation rates of the springs in order to avoid this "super-elastic" effect, and we take these constraints into account using a low-cost heuristic method inspired from classical *dynamic inverse* procedures. Results are discussed in section 6.

---
[1]also: coat hanging on a peg, banner, curtain,...

2 The Mass-Spring Model

2.1 The Mesh

Our elastic model is a mesh of $m \times n$ virtual masses, each mass being linked to its neighbors by massless springs of natural length non equal to zero. The linkage inbetween neighbors is achieved in three different ways (figure 1):

- springs linking masses $[i, j]$ and $[i + 1, j]$, and masses $[i, j]$ and $[i, j + 1]$, will be referred to as "structural springs";

- springs linking masses $[i, j]$ and $[i+1, j+1]$, and masses $[i + 1, j]$ and $[i, j + 1]$, will be referred to as "shear springs";

- springs linking masses $[i, j]$ and $[i + 2, j]$, and masses $[i, j]$ and $[i, j + 2]$, will be referred to as "flexion springs".

Indeed, under pure shear stresses, only the "shear springs" are constrained; under pure flexion stresses (i.e. bending), only the "flexion springs" are constrained; whereas under pure compression or traction stresses (i.e. stretching), only the "structural springs" are constrained.
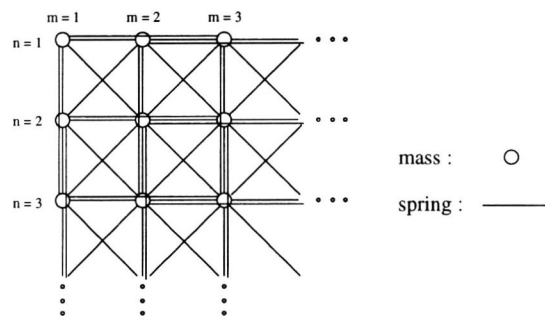

Figure 1: Regular mesh of masses and springs used for our model.

2.2 Dynamics and Forces

The system under study is the mesh of the $m \times n$ masses, each mass being positioned at time t on the point $P_{i,j}(t)$, where $i = 1, \ldots, m$ and $j = 1, \ldots, n$. The evolution of the system is governed by the fundamental law of dynamics:

$$\mathbf{F}_{i,j} = \mu \mathbf{a}_{i,j}$$

where μ is the mass of each point $P_{i,j}$ and $\mathbf{a}_{i,j}$ is its acceleration caused by the force $\mathbf{F}_{i,j}$. $\mathbf{F}_{i,j}$ can be divided between the internal and external forces.

The internal force is the resultant of the tensions of the springs linking $P_{i,j}$ to its neighbors:

$$\mathbf{F}_{int}(P_{i,j}) =$$
$$-\sum_{(k,l)\in\mathcal{R}} K_{i,j,k,l}\left[\mathbf{l}_{i,j,k,l} - l^0_{i,j,k,l}\frac{\mathbf{l}_{i,j,k,l}}{\|\mathbf{l}_{i,j,k,l}\|}\right] \quad (1)$$

where:

- $\mathcal{R}$ is the set regrouping all couples (k,l) such as $P_{k,l}$ is linked by a spring to $P_{i,j}$,

- $\mathbf{l}_{i,j,k,l} = \overrightarrow{P_{i,j}P_{k,l}}$,

- $l^0_{i,j,k,l}$ is the natural length of the spring linking $P_{i,j}$ and $P_{k,l}$,

- $K_{i,j,k,l}$ is the stiffness of the spring linking $P_{i,j}$ and $P_{k,l}$.

The external force is of various nature according to the kind of load to which we wish the model to be exposed. Omnipresent loads will be gravity, a viscous damping and a viscous interaction with an air stream (or wind). Let $\mathbf{g}$ be the acceleration of gravity, the weight of $P_{i,j}$ is given by:

$$\mathbf{F}_{gr}(P_{i,j}) = \mu\mathbf{g}$$

The viscous damping will be given by:

$$\mathbf{F}_{dis}(P_{i,j}) = -C_{dis}\mathbf{v}_{i,j}$$

where C_{dis} is a damping coefficient, and $\mathbf{v}_{i,j}$ is the velocity of point $P_{i,j}$. The role of this damping is in fact to model in first approximation the dissipation of the mechanical energy of our model. It is introduced as an external force, but could actually be considered as an internal force as well. Finally, a viscous fluid moving at a uniform velocity $\mathbf{u}_{fluid}$ exerts, on the surface of a body moving at a velocity $\mathbf{v}$, a force $\mathbf{F}_{vi} = C_{vi}[\mathbf{n} \cdot (\mathbf{u}_{fluid} - \mathbf{v})]\mathbf{n}$, where $\mathbf{n}$ is the unit normal on the surface. In our case:

$$\mathbf{F}_{vi}(P_{i,j}) = C_{vi}[\mathbf{n}_{i,j} \cdot (\mathbf{u}_{fluid} - \mathbf{v}_{i,j})]\mathbf{n}_{i,j}$$

where $\mathbf{n}_{i,j}$ is the unit normal on the surface at point $P_{i,j}$.

2.3 Integration

All these considerations allow us to compute the force $\mathbf{F}_{i,j}(t)$ applied on point $P_{i,j}$ at any time t. The fundamental equation of dynamics can therefore be explicitly integrated through time by a simple Euler method:

$$\begin{cases} \mathbf{a}_{i,j}(t + \Delta t) = \frac{1}{\mu}\mathbf{F}_{i,j}(t) \\ \\ \mathbf{v}_{i,j}(t + \Delta t) = \mathbf{v}_{i,j}(t) + \Delta t\,\mathbf{a}_{i,j}(t + \Delta t) \quad (2) \\ \\ P_{i,j}(t + \Delta t) = P_{i,j}(t) + \Delta t\,\mathbf{v}_{i,j}(t + \Delta t) \end{cases}$$

where Δt is a chosen time-step[2].

2.4 Dynamic Inverse Procedures

There are some cases where the movement of a cloth object is not entirely caused by analytically computable forces. This occurs for all *contact problems*. A simple example of this situation is given by the case of a hanging curtain. Each hanging point of a mass-spring curtain is subject to internal forces and gravity which all tend to pull it downward. But the rod from which the curtain hangs is exerting a counter-balancing force which is not directly computable as a function of the positions and the velocities of the vertices of the mesh. Nevertheless, this counter-balancing force can be indirectly determined.

The integration of the fundamental law of dynamics allows us to compute the displacement of a point from the knowledge of the force applied to it. But in our case, we can solve the inverse problem: we know the displacement of the hanging point (it is equal to $\vec{0}$ if the rod is fixed), and hence we can compute its actual velocity and the actual resulting force applied to it (also equal to $\vec{0}$ if the rod is fixed). All happens as if we did not take into account the results of the integration of equation (2) for the hanging points. Whatever the result found, the displacement of the hanging points is re-set to its *a priori* known value.

In cloth animation, dynamic inverse procedures are also used to deal with collisions of the cloth object with other objects, and to deal with self-collisions of the cloth itself. Such procedures are described by Carignan *et al.* in [7], but we did not focus on this point in this paper.

3 The "Super-Elastic" Effect

After having implemented this algorithm, we have tested it in various situations, which led to more or less realistic results. We have been particularly interested in finding the manifestations of the lack of realism of our model, and then in understanding its causes. We will study here the case of a sheet hanging by two adjacent corners, subject to gravity, but in a scene where there is no wind ($\mathbf{u}_{fluid} = \vec{0}$). The sheet is modeled by a mesh composed of 17×17 vertices. The two corners are immobile and held into place by a dynamic inverse procedure, as described in section 2.4. Figure 2(a) shows the initial position of the "structural" springs (represented by cylinders) of the sheet model. Figure 2(b) shows the resulting

---

[2]More details about how Δt must be chosen are presented in section 4.

150

deformation computed by the algorithm after 200 steps.

This example clearly shows one of the problems—the elongation of the springs directly tied to the corners is very high compared to all the other springs. The deformation of the sheet is therefore very locally concentrated around the corners, and the deformation rate[3] decreases very rapidly with the distance between the vertices and the corners. The value of the deformation rate of the most elongated springs exceeds 100 % !
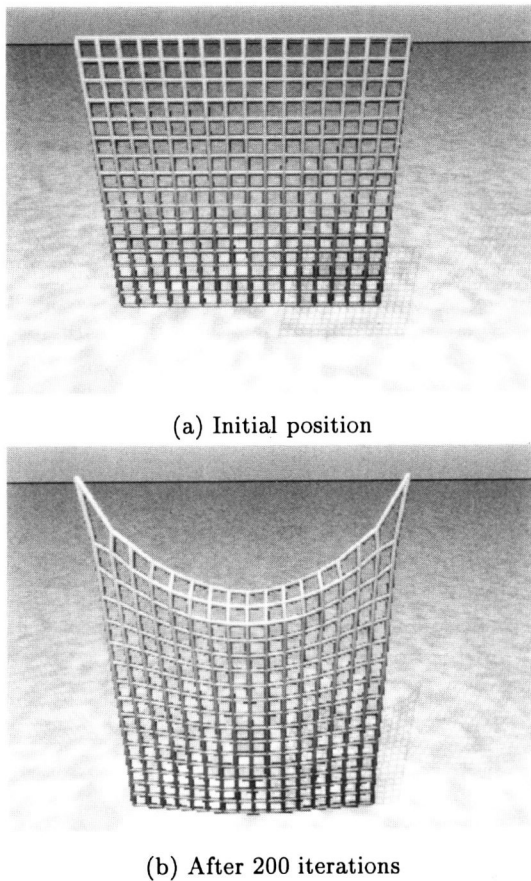

(a) Initial position

(b) After 200 iterations

Figure 2: Deformation of the elastic model of a sheet hanging by two adjacent corners.

Such a local deformation never occurs in woven fabrics. In fact it can be sometimes encountered in very loose knitted fabrics, but it is rather infrequent. In the field of the physics of polymers, a similar phe-

nomenon can be observed and it is called "super-elasticity", since it concerns materials which can be subject to very high elastic deformation rates.

The reason for this difference between our model and real woven fabrics is that woven fabrics are not super-elastic at all. Their elasticity is non-linear, and their "stiffness" increases very rapidly when the deformation rate increases. The deformation rate is thus always limited to a very low amount, and when very high loads are applied, rupture occurs before any large deformation can take place. However, our goal was not to model the phenomenon of rupture, but only to limit unlikely large deformations. The maximum deformation rate of most woven fabrics is around 10 %, and it is even lower for some linen cloths, calicos or denims, for instance.

Another lack of realism can be seen during the animation of the sheet: this "super-elongation" does not come to stabilization easily, and leads to a high amplitude *oscillation* around the equilibrium position of the sheet. To avoid this oscillation, it is therefore necessary to increase the damping coefficient C_{dis}. Though this operation can indeed suppress any oscillation, one of its shortcomings is that the sheet then looks like it had been immersed in some oily fluid and its movement loses realism.

4 Increasing Stiffness

To avoid the "super-elastic" effect, we have tried to adjust the parameters of the model. This consists in increasing the stiffness of the springs. For a same level of constraints (same gravity in our case), the deformation rate should be lower for stiff springs. This result can indeed be attained, but not as simply as it seems.

Experience shows that, for a given time-step Δt and a given mass μ, there is a critical stiffness value K_c above which the numerical resolution of the system is divergent. In fact, this result is well known in the case of linear differential equations[4]. The mathematical results concerning such linear equations show that their numerical solving is ill-conditioned if Δt is greater than the natural period of the system [11], given by:

$$T_0 \approx \pi \sqrt{\frac{\mu}{K}} \qquad (3)$$

$$\Longrightarrow K_c \approx m \frac{T_0^2}{\pi^2}$$

Therefore, if we want to increase stiffness, we have to decrease Δt below the new decreased value of T_0.

---

[3]In the following, we will speak of the "deformation rate" of the springs, which will be defined as: $\tau = \frac{l-l_0}{l_0}$ where l_0 is the natural length of a spring, and l is its length at any time t.

[4]Our model can be reduced to this case provided the natural lengths of the springs are supposed equal to zero [12].

For a same animation time, the number of iterations needed will then be greater, and the algorithm will be more costly[5].

In our model, all springs have a natural length non equal to zero, but they remain nevertheless intrinsically linear springs. However, these springs once coupled lead our model to lose its linearity: it cannot be reduced to linear matricial differential equations. However, experience shows that the result given by equation (3) is still (at least qualitatively) true in our case.

This is why we have tried to find a new method to avoid the super-elastic effect, without having to decrease Δt.

5 Dynamic Inverse Constraints on Deformation Rates

Our idea has been to apply an ad hoc dynamic inverse procedure to the "super-elongated" springs (see below for their characterization) so as to reduce their elongation. At each given time-step, the numerical integration is achieved using equation (2). Then the deformation rates of all springs are computed. If, and only if, the deformation rate of a spring is greater than a critical deformation rate τ_c, then a dynamic inverse procedure is applied to the two ends of the spring so that its deformation rate exactly equals τ_c. This means that, if we choose $\tau_c = 0.1$, we want the length of the springs not to exceed their natural length by more than 10 % (for many fabrics, it could even be less than that).

The underlying reasoning which helped us build this procedure was the following: we assume that the position of the spring computed using equation (2) is correct regarding its *direction*, but not regarding the distance between the two ends of the spring; the only thing to do is then to reduce this distance so that the deformation rate does not exceed τ_c while keeping the computed direction of the spring unchanged.

The distance reduction is done differently whether the ends of the spring are *loose* or are *fixed* by a new dynamic inverse procedure. If both ends are loose (figure 3(a)), both are evenly "brought closer" to their middle so that the "shrunk" spring reaches τ_c. If only one end is loose (figure 3(b)), then it is "brought closer" to the fixed end so as to reach τ_c. If both are fixed, they are left unchanged.

Thus, in a single computation, all the springs with

(a) Adjustment of a "super-elongated" spring linking two loose masses.

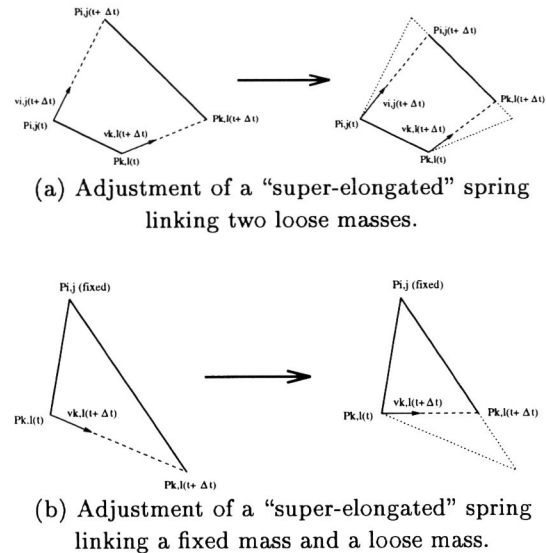(b) Adjustment of a "super-elongated" spring linking a fixed mass and a loose mass.

Figure 3: Principle of our ad hoc dynamic inverse procedure: adjustment of the "super-elongated" springs.

a deformation rate exceeding τ_c after the numerical integration are adjusted to a more "reasonable" deformation rate. Of course, at this point, this operation has modified the position of many vertices, and may have over-elongated other springs. But, if the deformation is very locally concentrated, the springs affected by the operation should be less elongated than the ones which had been detected before the operation. One of the effects of the procedure is to help the deformation propagate through the structure, instead of remaining in a concentrated area.
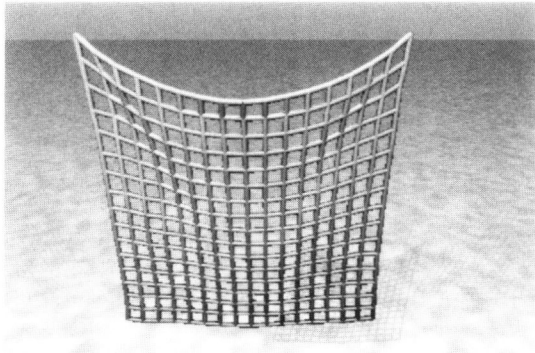
We did not take into account the order in which the super-elongated springs are adjusted at each step. In our procedure, this order depends entirely on our data structure. This is acceptable only because we restricted our method to situations in which constraints are *locally* distributed, that is situations in which only a few springs are super-elongated at each step. If high constraints were globally extending to the whole cloth object, then the adjustment order of the springs would probably have more importance, and should be studied.

There are still mathematical investigations to be carried out in order to prove the properties of our procedure. However, we decided to test it, and our results show that it is valid in all the situations we have implemented so far.
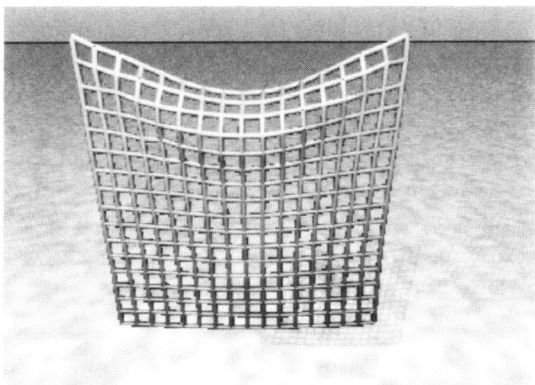
---

[5]It must be mentioned that even if only the stiffness of a few springs (e.g. the most elongated ones) is increased, the time-step Δt must be decreased: the lowest value of μ and the highest value of K have to be used in equation (3).

6 Results and Discussion

We applied this procedure to the case of the hanging sheet described in section 3. A critical deformation rate $\tau_c = 10$ % has been chosen, first only for "structural" springs, and then for both "structural" and "shear" springs. We never put any constraint on "flexion" springs because flexion, unlike elongation and shear deformation, is almost not limited at all in real cloth objects—they are easily folding.

(a) Our method applied to "structural" springs

(b) Our method applied
to "structural" and "shear" springs

Figure 4: Deformation of our model of a sheet hanging by two adjacent corners (after 200 iterations).

Figures 4(a) and 4(b) show the results we have obtained, and are to be compared with figure 2(b). On figure 4(a), the "super-elastic" effect is totally suppressed, and all springs have a reasonable deformation rate. The sheet does not seem to be made out of some kind of rubber anymore. On figure 4(b), the shear deformation is also successfully controlled, and the sheet model is forced to undergo a greater folding deformation in order to reach a low energetic position, since now its shear deformation is limited.

Finally, everything happens as if the springs were classical springs up to a certain deformation rate, and quasi rigid rods above this deformation rate. But this quasi rigid behavior is performed using a low-cost dynamic inverse procedure instead of a more costly solving of a constraint problem. At a given stiffness, the computational cost of the new algorithm is indeed only 15 % greater than that of the classic elastic model[6]. But to reach an equivalently stiff behavior with the latter, stiffness and time sampling have to be increased. Therefore, for two equivalently stiff behaviors, we have measured that our algorithm is *90 % faster* that the classic elastic algorithm.
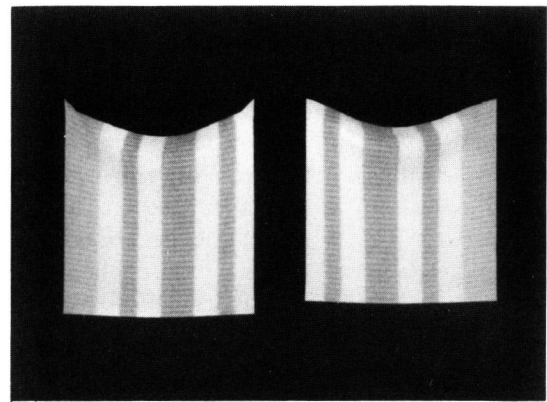

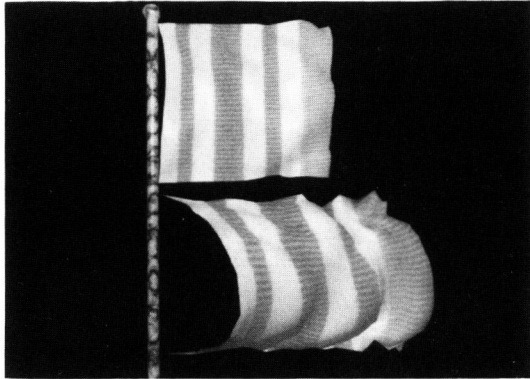Figure 5: On the left: a stiff elastic model computed in 9 mn. On the right: our model computed in 1 mn.

Regarding the animation sequence, the amplitude of the oscillation of the falling sheet is also much lower than in the initial model, and it disappears more quickly since many springs are kept from behaving like ideal elastic springs. The damping coefficient can be lowered, and the movement is less "oily".

The same results have been obtained with a 33×33 mesh. We have noticed in some tests that the results were even better when we performed the dynamic inverse procedure twice in a row at each iteration, and this makes us think that the procedure, when repeated, is converging.
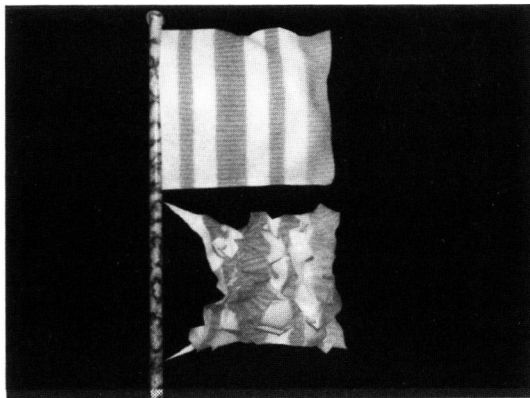
---

[6]Here, the two models do not include any procedure designed to avoid self-collision. If such a procedure was added and since it increases the computational cost a lot, the cost added by our procedure would be even less significant in proportion.

We have also successfully applied the method to a flag in a strong wind. Besides leading to extravagant elongations of the springs, increasing the velocity of the wind is indeed another factor which makes the purely elastic algorithm unstable, and hence which requires lower time-steps. But with our model, the elongations were totally controlled, and the move-

(a) Elastic flag and our "semi-rigid" flag when stiffness is low.

(b) Increasing stiffness: elastic flag becomes chaotic and "semi-rigid" flag remains stable.

Figure 6: Comparison for the case of a flag.

ment of the flag was very realistic. With the same wind velocity, time-step and stiffness, the purely elastic model was totally unstable and chaotic (figure 6(b)). We must also mention that this time, a critical deformation rate of only 5 % had been chosen for our model. Though this leads to a more "rigid" behavior, it does not decrease neither the efficiency of our method, nor its cost.

At a lower stiffness[7], we have been able to make

_____
[7]The value of the acceleration of gravity was also lowered.

a comparison between the purely elastic model and our model. As is shown on figure 6(a), the qualitative observations mentioned above for the hanging sheet are still significantly true. Though the flag was designed to be a square of cloth, its elastic model deforms to an elongated rectangle shape, and the shear deformation at the attaching points is high. On the contrary, the square shape of our controlled model is much less deformed and there is hardly no shear deformation.

In these examples, local super-elongations were concentrated around only two hanging points. But our method is also efficient for the case of the sail of the "drakkar" in figure 7 which hangs by 8 points on the upper rod and is tied to 2 points on the lower rod.
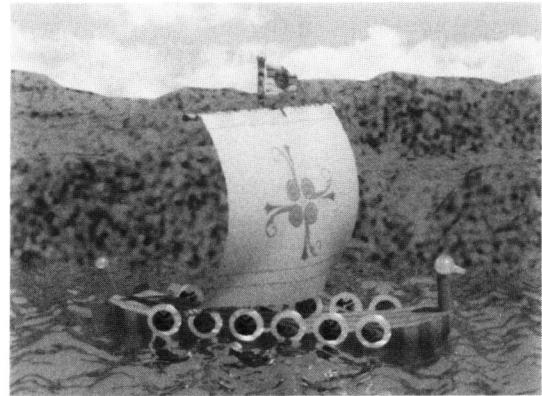

Figure 7: Wind blowing in a sail.

7 Conclusion

We have presented an elastically deformable model of cloth objects and we have studied its unrealistic behavior compared to that of real woven fabrics. We have shown that this behavior, known as "super-elasticity", could not be improved by merely adjusting the model's parameters without significantly increasing the cost of the algorithm. We have therefore proposed another method to avoid the "super-elastic" effect, based on a low-cost dynamic inverse procedure. We have shown how this heuristic method could help us model a realistic cloth object in situations in which constraints had not been adequately handled before. Using this more realistic model, it is likely that our next step will be to determine our model's parameters (including maximum deformation rate τ_c) so that our model's behavior

identifies with that of real cloth samples of various nature. This could be done using the image processing of an animation sequence of real cloth samples. We hope that this work will contribute to the field of cloth animation in computer graphics so that cloth models stick closer to reality. This is one of the conditions necessary to further consider the potential application of these cloth models to garment industry.

8 Acknowledgements

References

[1] Amirbayat J. and Hearle J. W. S. The Complex Buckling of Flexible Sheet Materials—Part I. Theoretical Approach. In *Int. J. Mech. Sci.*, Great Britain, 1986, Vol. 28, No. 6, pp. 339–358.

[2] Aono Masaki. A Wrinkle Propagation Model for Cloth. In *Proc. Computer Graphics International'90*, Springer, Tokyo, 1990, pp. 96–115.

[3] Weil Jerry. The Synthesis of Cloth Objects. In *Proc. SIGGRAPH'86, Computer Graphics*, 1986, Vol. 20, No. 4, pp. 49–54.

[4] Taillefer Frédéric. Modélisation du Rendu et du Comportement des Surfaces Tissées. *Ph.D. Thesis*, University of Paris 6, 1993.

[5] Terzopoulos Demetri, Platt John, Barr Alan, Fleischer Kurt. Elastically Deformable Models. In *Proc. SIGGRAPH'87, Computer Graphics*, 1987, Vol. 21, No. 4, pp. 205–214.

[6] Lafleur Benoit, Magnenat Thalmann Nadia, Thalmann Daniel. Cloth Animation with Self-Collision Detection. In *Proc. IFIP Conference on Modeling in Computer Graphics*, Springer, Tokyo, 1991, pp. 179–187.

[7] Carignan Michel, Yang Ying, Magnenat Thalmann Nadia, Thalmann Daniel. Dressing Animated Synthetic Actors with Complex Deformable Clothes. In *Proc. SIGGRAPH'92, Computer Graphics*, 1992, Vol. 26, No. 2, pp. 99–104.

[8] Breen David E., House Donald H., Wozny Michael J. Predicting the Drape of Woven Cloth Using Interacting Particles. In *Proc. SIGGRAPH'94, Computer Graphics Proceedings*, 1994, pp. 365–372.

[9] Van Overveld C. W. A. M., Van Loon Erik. Hanging Cloths and Dangling Rods: a Unified Approach to Constraints in Computer Animation. In *The Journal of Visualization and Computer Animation*, 1992, Vol. 3, pp. 45–60.

[10] Platt John C., Barr Alan H. Constraint Methods for Flexible Models. In *Proc. SIGGRAPH'88, Computer Graphics*, 1988, Vol. 22, No. 4, pp. 279–288.

[11] Klaus-Jurgen Bathe. In *Finite Element Procedures in Engineering Analysis*. Prentice-Hall, 1982.

[12] Nastar Chahab, Ayache Nicholas. Fast Segmentation, Tracking, and Analysis of Deformable Objects. In *Proceedings of the Fourth International Conference on Computer Vision (ICCV '93)*, Berlin, 1993.