Realism in Cloth Draping: Modeling Air Resistance in Particle Systems

N. Damodharan
Sharat Chandran
Department of Computer Science and Engineering
Indian Institute of Technology
Bombay, India, 400076.
{damu,sharat}@cse.iitb.ernet.in

Abstract

Modeling the behavior of cloth drape has been under intense research in the area of interactive garment design and dressing animated synthetic actors. Recently, a particle system *based model was proposed in [BHW94a] to visualize drape. This model — bolstered with empirical data — describes the behavior of cloth in terms of its energies that characterize repelling, stretching, bending, and trellising. In this work, we enhance this model to handle resistance offered by air. This is incorporated into the energy functions under the norms of* physically-based modeling techniques. *We analyze the existing model and suggest a few strategies to enhance its computational efficiency.*

Keywords: Physically-based modeling, particle methods, cloth, air-resistance.

1 Background and prior work

The need for better techniques in modeling and visualization has been felt in the field of computer graphics to model common but complex materials. This is mainly due to the fact that existing models are biased in their objectives towards either computational efficiency or realism. Cloth is one such common material where existing models often perform less satisfactorily than desired in both these aspects.

Existing approaches can be broadly classified into three categories [NG96]: *geometric*, *physically-based*, or *hybrid*. Initial work in the field of geometric deformations using Jacobian matrix to deform solids locally or globally was done by [Bar84]. One of the earliest work on cloth deformation was done by [Wei86]. He proposed a method for modeling cloth hanging in 3-D and supported by a number of constraint points. The method consists of a two-stage process; approximation followed by relaxation. In the first step all the interior points which lie within the convex hull formed by the constraint points are positioned. In the second step, relaxation is performed on all points to arrive at a close approximation of the surface. Both these models suffer from a few deficiencies; there is no concept of time, only a few kinds of drape can be visualized, and physical properties such as elasticity is lacking.

In [DMTM93], cloth is described as a swept surface defined by means of a genetrix curve and a directrix. These geometric parameters are controlled using the principles of applied mechanics. In [KG90], a hybrid approach is proposed to model wrinkles. A global analysis gives a hint about the modeling primitives which consists of all the characteristic points on the surface along with contours associated with them. These primitives are then deformed for modeling wrinkles on the surface.

Elasticity theory is used to describe the deformation of material in [TPFB87]. The draping process is described using a set of differential equations in terms of curvature and torsion, and finite differences methods are applied to discretize and solve them. In [Aon90] stress-strain relationship from the theory of elasticity is used to simulate drape.

1.1 Our focus

Other physically-based methods include *particle methods*, but we turn to an interesting variation.

The nature of cloth is particularly intriguing when we study *external* effects due to the viscous media surrounding it. In [LDG93], a concerted effort was made to see effects of the wind that might potentially disturb cloth drape. Low speed aerodynamics theory is used to represent the distributed force loading on the cloth offered by air flow. Displacement of various points on the surface is obtained by numerically solving the model based on the theory of elasticity.

Recently, [OTMS95] used a simpler model (with consequent lower realism) to achieve similar effects. A completely different approach was taken by [CIMM92] where use of motion flow from computer vision theory was used to calculate wind forces.

These models are appropriate in strong wind conditions but, due to the very nature, are not applicable to model effect of air resistance when there is little or *no wind*. Our main goal is to attack this situation, instead of concentrating on versatile techniques for simulating cloth in general. In this paper, we use a particle systems approach to add this effect for increased realism, as well for computational efficiency. In a nutshell, during simulation, due to resistance offered by air, the cloth behaves differently. Due to a pressure distribution profile on the surface, particles tend to settle at different speeds, thus giving a more realistic appearance to the drape. The net effect of this is that the animation of the drape tends to be more natural; particles in the center of the cloth will move slower than the particles towards the extremity.

1.2 Particle Systems

In many simulations, cloth is modeled as a set of surface patches stitched to form the surface. Representation of cloth in terms of such primitive surface elements appears to be insufficient [BHW94a, CYTT92], as cloth is a flexible object. Instead, the surface is further broken down to represent a particle as the smallest primitive [Ree83]. Every intersection of weft and warp[1] constitutes a particle. This gives us a 2-D array of 3-D points as a straight forward, yet powerful, representation of cloth.

We imbibe the concepts of particle systems originally used to model cloud and fire [Ree83] into our model. Particles in this model are generated, have a life span, and die. For modeling cloth, all the particles are generated right at the beginning of simulation, but they never die.

1.2.1 Dormancy

Instead, and in view of the computational aspect of the model, we explicitly introduce *dormancy* as a feature. Initially all particles are active. A particle may be brought down to a dormant state due to certain constraints and is made active only on satisfaction of some other set of constraints. Since only active particles contribute to the computations, computational efficiency may be achieved. A typical example of dormancy is when an active particle comes to rest on any object beneath it; the particle is declared dormant. This particle will be turned into an *active* state only under some other constraints; examples include the case when neighbors have their energy level above a certain threshold. In many situations when simulations progress iteratively, most of the cloth particles are likely to be dormant and the number of active particles few; the simulation is consequently fast. The test for making the dormant particle active is less expensive than the unnecessary computation done on them, especially, when increased realism is the desiderata.

[1] The horizontal and vertical threads, respectively.

1.2.2 Attributes

Other particle attributes [Ree83] like velocity, acceleration and color are used in the simulation and rendering of cloth. The simulation is analogous to distributed behavior modeling [Rey87], where the particles are subjected to collision avoidance, velocity matching, and centering.

1.3 Overview of this paper

Using the above concepts (Section 1.2), this paper analyzes the particle based method to model drape introduced in [BHW94a]. We first summarize this approach in Section 2. The model is introduced in greater detail and is enhanced to handle resistance offered by air in Section 3. Since the simulation is based on energy minimization, energy due to air resistance is directly integrated into the current model. Methods to reduce computations using dormancy, along with efficient and accurate calculation of the repel energy and curvature for the bending energy are also presented. Finally, some sample results are shown in the last section to visually present the enhanced realism.

2 The model

The model in [BHW94a] tries to capture the various interactions occurring at the thread level using energy equations. Solutions for these can be described using Geometrically Deformable Models (GDMs) [MBL$^+$91]. Following the same norms our model has been extended to handle resistance offered by air.

2.1 Geometrically Deformable Models

Geometrically Deformable Models or GDM's [MBL$^+$91] start with a simple model that is topologically closed and deform the model based on a set of constraints, that grow or shrink to fit the object within the volume while maintaining their closed and non-intersecting nature. The effect of the process is similar to that of inflating a balloon within a container or draping a piece of cloth around an object beneath it.

Once the behavior and characteristics of the object are known, constraints are developed to achieve this behavior.

Generally the behavior is modeled using cost functions, and as they are minimized, the model deforms searching for boundaries. A very important aspect of GDM is that each cost function must be able to show its response significantly when its constraint is violated and be insignificant when its constraints are met. The total cost is expressed as a linear combination of individual cost functions:

$$C_i(x, y, z) = a_0 D(x, y, z) + a_1 I(x, y, z) + a_2 T_i$$

where $C_i(x, y, z)$ is the cost associated with the current position of the particle, $D(x, y, z)$ is the potential field that drives the model towards the boundary, $I(x, y, z)$ is the image term that identifies events, T_i is a measure of local configuration, a_0, a_1, a_2 are individual weighting coefficients. These constraints, as defined by the authors, may be elaborated as follows:

- Deformation potential: This potential must monotonically increase or decrease from the frame of reference and will repel or attract the model point away from or towards its frame of reference.

- Image events: This constraint counterbalances the deformation potential and restricts progress, e.g., a surface already in contact with the boundary. This function is used to declare a particle dormant.

- Maintaining topology: Although it appears that the first two terms are enough, maintenance of topology is an important constraint. Typically, this function will prevent the distance between any two neighbors from crossing a threshold and thus prevent spurious thread elongations.

3 Energy functions

The model in [BHW94a] utilizes concepts from particle-systems to derive the energy functions and describes solutions for them based on GDM for simulating cloth drape. The constraints are intuitive and modeled using the physical characteristics of the object. There are five features that the model captures and expresses as potential energy functions. These are repelling or non-penetration, stretching, bending, trellising and gravity [BHW94a]. Thus,

$$U_{total_i} = U_{repel_i} + U_{stretch_i} + U_{bend_i} + U_{trellis_i} + U_{gravity_i}$$

where all these are independent constraints that contribute to the total energy. In the above equation U_{total_i} is the cost function, $U_{gravity_i}$ is the deformation potential and the remaining are used as image events and for maintaining topology.

We propose to express resistance offered by air when the particle is in motion as an energy function $U_{airResistance_i}$, which acts as an image event to the cost function. Thus,

$$U_{total_i} = U_{repel_i} + U_{stretch_i} + U_{bend_i} + U_{trellis_i} + U_{gravity_i} + U_{airResistance_i}$$

$U_{airResistance_i}$ is evaluated under the norms of physically-based modeling techniques in terms of change in pressure and projected area. In this section, we also present our methods to compute the existing energy functions efficiently and accurately.

3.1 Repelling, Stretching and Gravitational Energy

Repel energy is an artificial energy that effectively keeps particles away from each other and thus is responsible for non-penetration [BHW94a] and self-collisions. This method, by its very nature, is not as general as the one suggested in [VCT95, VT94]. Intuitively this function must have the property that the constraint is violated when the distance between the point and its neighbor collapses below a threshold σ as shown in Figure 1, and so the energy minimization function must exhibit its presence by returning a high value. One such function is:

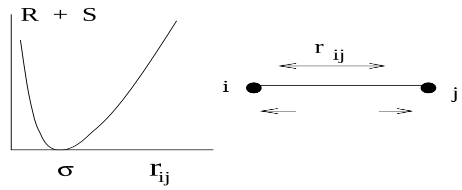

Figure 1: Nature of the sum of stretching energy $S(r_{ij})$ & repelling energy $R(r_{ij})$ as a function of distance r_{ij}.

$$R(r_{ij}) = \begin{cases} C_0[(\sigma - r_{ij})^5 / r_{ij}] & r_{ij} \leq \sigma \\ 0 & r_{ij} > \sigma \end{cases} \quad (1)$$

where C_0 is the scaling constant. U_{repel_i} is calculated by summing over all particles and is given by

$$U_{repel_i} = \sum_{j \neq i}^{n} R(r_{ij})$$

The stretch function is responsible for the stretch that a particle can experience during interactions with its neighbors [BHW94a]. Intuitively this function must be exactly opposite in functionality to that of the repelling energy.

$$S(r_{ij}) = \begin{cases} 0 & r_{ij} \leq \sigma \\ C_0[((r_{ij} - \sigma)/\sigma)^5] & r_{ij} > \sigma \end{cases} \qquad (2)$$

$U_{stretch_i}$ is now calculated by summing over all its four-connected neighbors and is given by

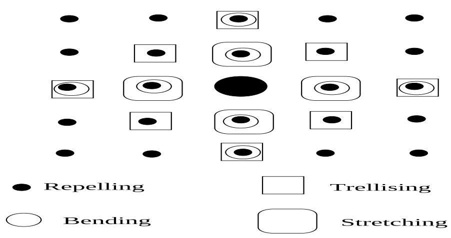

Figure 2: Particle dependencies.

$$U_{stretch_i} = \sum_{j \in \mathcal{N}_i}^{n} S(r_{ij})$$

where $\mathcal{N}_i$ are the four-connected neighbors of the particle i. Graphically the stretch function mentioned above can be represented as shown in Figure 1. Particles that contribute to the stretch energy are as shown in Figure 2.

The energy due to gravity can be directly obtained from classical physics:

$$U_{gravity_i} = m_i g h_i$$

where m_i is the mass, h_i is the height and g is the acceleration due to gravity.

3.1.1 First modification: Optimizations

Computationally, this is the most expensive operation because the distance between the particle and every other particle on the cloth has to be calculated. Since distance calculation is an expensive operation, it would be better if we could eliminate unnecessary computations before the actual distance calculation.

The bounding-volume test is one of the cheapest and the most efficient. We enclose a small cube which is parallel to the XYZ-planes, with the particle p whose repel energy is being calculated as the center, and at a distance σ from every side. With six comparisons — and thus $O(n)$ computations in total — it can be inferred whether any other particle has fallen below the nominal distance from the particle in the center and then proceed with the distance test to calculate repel energy. Empirical analysis shows mostly particles from a window 4×4 around the particle are the ones that participate in the repel energy. Note that we do not exclude any particle per se in the repel energy calculation; it *is* mandatory to consider the repel energy for p with every other particle on the surface to avoid self-collisions. However, since a number of distance computations is avoided, the net result is an implementation that runs fast. By the very definition of repel energy, self intersections never take place.

A related empirical observation that we made also showed that the energy function in Equations 1 and 2 above may be raised to powers lower than 5 and still largely similar results were obtained. However, this is a function of the test input.

3.2 The Kawabata evaluation system

Both bending and trellising energy make vital contributions to the energy function. These energy equations can be derived from empirical data produced by the Kawabata evaluation system [BHW94b]. The system has standard methods to measure bending, shearing and tensile properties of cloth. Bending measurements are carried out on a standard sample that is clamped and bent in the middle and bending moment M versus curvature K is plotted. Shearing measurements are done by applying shearing force along one of the edges and plotting force F_s versus shear angle ϕ. The process of developing the energy equations consists of three main steps:

- Determine functions that approximate the Kawabata curves.
- Relate the approximating functions and the model's energy functions.
- Scale the resulting equations to produce energy values in standard physical units.

3.3 Bending Energy

Bending has to be modeled with the help of angles formed between each set of three particles on a single thread [BHW94a].

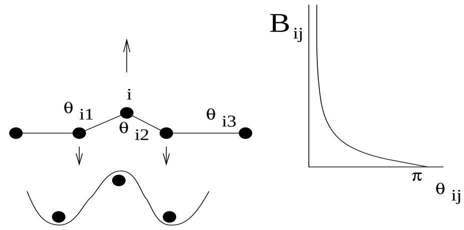

Figure 3: On the left, a piece of bent thread is modeled in terms of angles θ_{ij}. The nature of the bending energy as a function of angle appears on the right.

Intuitively this function has to record a violation of constraint when the bending angle between the three particles is nearing zero and return infinity as shown in Figure 3. So the total bending energy is given by

$$U_{bend_i} = \sum_{j \in M_i}^{n} B(\theta_{ij})$$

where M_i is the set of six angles θ_{ij} formed by the segments connecting particle i and its eight neighbors as in Figure 2. Bending energy from the theory of elastic bending [MS91] is $dU = \frac{M}{2\rho}ds$ where $\rho = \frac{1}{K}$, K is the curvature and ρ is the radius of curvature. Integrating on both sides, we get $U = \int_0^{\sigma} \frac{MK}{2} ds = \frac{MK}{2}\sigma$. Since M is the bending moment per unit width of sample, the summed up energy contribution for a patch $\sigma \times \sigma$ is obtained by just multiplying the equation by σ where σ is the equilibrium distance from its 4-connected neighbors. So the bending energy is given by $B = \frac{MK}{2}\sigma^2$. The Kawabata plot for bending give us a direct relation between K and M and if we plug in the value of K we can find M, and hence B.

3.3.1 Second modification: Accuracy in curvature

To relate K to θ, which is the geometry of the bending angle formed by the three particles, an **approximate** formula is suggested in [BHW94a]: $K(\theta) \approx \frac{2}{\sigma}\cos(\frac{\theta}{2})$.

We note, however, that curvature for a given set of three particles can be **accurately** expressed as a function of the radius of curvature instead of relating with θ, even though the original intuitive formulation is in terms of θ.. The center of curvature can be computed for the three points that form the bending angle. It is the intersection of the two perpendicular bisector planes of any two sides of the triangle formed by the three points and the three point plane itself. The intersection is obtained by inverting a 3×3 matrix, and thus ρ and K can be easily calculated. The solution of a 3×3 can be expanded out into code. This method is computationally as efficient as the previous method, if not better, and is more accurate. The two special cases where the three points are collinear or the angle between them has collapsed to zero can be easily taken care of (in both these cases the determinant of the three points will be zero).

3.4 Trellising Energy

Trellising occurs when threads are held fast at their crossings and bent to create S-curves in the local plane of the cloth. Intuitively this energy prevents thread crossing. It prevents the collapse of angle between a weft and a warp, which does not happen in real life because of the weave as shown in Figure 4. The trellis angle is defined as the angle formed as one of the line segments moves away from the equilibrium angle that is pre-defined [BHW94a]. The total energy contribution from trellising is

$$U_{trellis_i} = \sum_{k \in K_i} T(\phi_{ij})$$

where K_i is a set of four trellising angles ϕ_{ij} formed around the four-connected neighbors of particle i as in Figure 2.

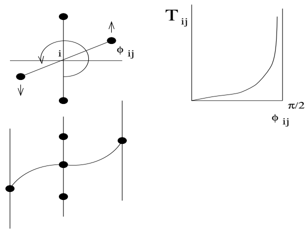

Figure 4: On the left, a piece of thread is modeled for the purposes of trellising. The nature of energy as a function of angle appears on the right.

In the Kawabata shear tester the work W produced by a force F acting over a displacement dS is given by, $W = \int FdS$. If we assume the width of the sample l to remain constant then the length of the path traveled by a point at which the shearing force is applied is $S = l\phi$ where ϕ is the shearing angle. So the trellis energy in the direction of the motion is $T = \int F\cos(\phi) l d\phi$. The Kawabata plots for shear tester gives a relationship between F and ϕ, from which we directly find the trellis energy.

No change was made in our computation of the energy due to trellis.

3.5 Third modification: Increased realism with air-resistance

When cloth falls freely over objects beneath it, it encounters resistance from air between the objects and itself. We calculate the amount of force applied on the surface due to this resistance, and express this force in terms of energy. This allows us to embed the extension directly into our new model.

For a freely falling panel, the energy contribution due to resistance from air is given by

$$U_{airResistance_i} = (P_{up_i} - P_{down_i}) A_{xy_i} dX_i$$

where P_{down_i} and P_{up_i} are the pressure distributions in the downstream and upstream sides of the panel i respectively and A_{xy_i} is the projected area of the panel i and dX_i is the displacement of the panel i. For a given plate as shown in Figure 5, due to the flow of the plate against the fluid, the pressure on down stream side is slightly less than that of the up stream. So the equation is $U_{airResistance_i} = \triangle P_i A_{xy_i} dX_i$. Since both A_{xy_i} and dX_i are known, it remains to compute $\triangle P_i$ for the given panel.

Concepts from fluid mechanics are applied to calculate the value of $\triangle P_i$. Consider an infinitely long plate held perpendicular to the direction of the flow of fluid in a stream of infinite extent. For an ideal fluid flowing past the plate the flow pattern will be symmetrical on the upstream and the downstream sides of the plate as in Figure 6. Further, because of the symmetrical flow pattern the pressure distribution will also be symmetrical and hence there will be no drag. However, for a real fluid like air, flowing past a flat plate the flow pattern as well as the pressure distribution are markedly different from those for an ideal fluid [MS91].

As shown in Figure 5 the flow separates at the top and the bottom edges of the plate, thereby forming a fairly wide *wake* on the downstream side. Figure 7 shows the pressure distribution on the upstream and the downstream faces of the plate as obtained by actual measurement. As expected, the pressure on the upstream face of the plate is positive for the entire width, with a maximum at the middle section along which the stagnation points are located. On the downstream face of the plate in the wake zone a constant pressure drop takes place as indicated by a more or less constant negative pressure over the entire width of the plate [MS91].

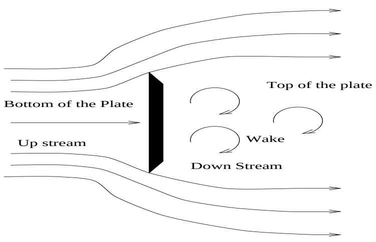

Figure 5: Real fluid flow

Since cloth is not an infinitely long plate, but has a finite area, the intersection of the pressure distribution in both the directions is obtained in order to calculate the pressure at a small panel on the surface. To calculate the pressure distribution for a panel xy on the cloth surface we can approximate the pressure as the geometric mean of the pressures in the two directions. So the net pressure is approximated as $P_{xy} = \sqrt{P_x \times P_y}$ where P_x and P_y are the pressure distributions of panel xy in the two directions respectively. The pressure distribution is now a paraboloid as shown in Figure 8. It can be clearly seen that the pressure on the interior panels of the cloth is more than those bordering the surface. P_x is given by $P_x = P_{up_x} - P_{down_x}$ so the parabolic pressure distribution as shown in Figure 7 can be approximated as

$$P_x = P_{up_x} - (-1.4\frac{\rho V_i^2}{2}),$$

$$P_{up_x} = \frac{\rho {V_i}^2}{2}[1 - 4(x_i - \frac{1}{2})^2]$$

where V_i is the velocity, ρ is the density of air and x_i is the normalized coordinate of the panel in the X direction. The pressure in the Y direction can be similarly calculated. And $P_{xy} = \sqrt{P_x \times P_y}$.

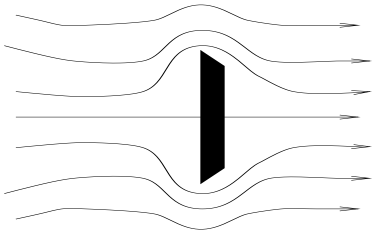

Figure 6: A symmetrical flow pattern is observed for an ideal fluid

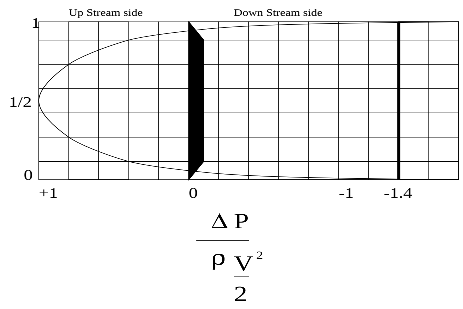

Figure 7: Pressure distribution on an infinite flat plate. The ordinate indicates (normalized) position of points on plate. Along the abscissa sample values are indicated to show parabolic distribution.

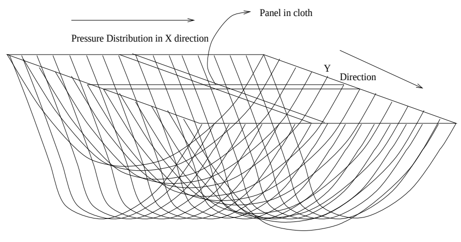

Figure 8: The pressure distribution on an finite plate

Since the force acting on the panel is resolved at the centroid of the panel, we select a panel for the particle P_{xy} such that the particle is at the centroid of the panel. This is shown in Figure 9 where panel $ABCD$ is the panel for the particle P_{xy}. Thus every panel is made of four patches surrounding the particle P_{xy}. To find the projected area we subdivide the panel into triangles as shown in Figure 9 and for each triangle in 2-D with vertices $(x_1, y_1)(x_2, y_2)$ and (x_3, y_3), the area of the triangle is given by $A = \frac{1}{2}|(y_2 - y_3) \times (x_1 - x_3) - (y_1 - y_3) \times (x_2 - x_3)|$

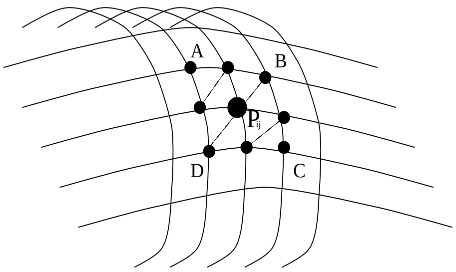

Figure 9: The panel for P_{xy}

Our extension is clearly different from the [LDG93] model. Here concepts from fluid mechanics which describe pressure distribution on a finite plate are used instead of low speed aerodynamics. The later is more suitable to model cloth draping in strong winds, while our extension is applicable to handle resistance offered by air when the cloth is falling on objects beneath due to gravity.

4 Evaluation method

Now that we have framed the potential energy functions based on the characteristics of the object, we frame an evaluation method to model the dynamics of the model. The energy constraints were formulated as a non-negative smooth function $E(\psi)$ such that $E(\psi)$ is minimum at all and only values of ψ for which the constraints are satisfied [WFB87]. So from an initial condition ψ_0 the energy E is minimized numerically solving the differential equation

$$\frac{dF(t)}{dt} = \bigtriangledown E, F(0) = \psi_0.$$

The solution is not guaranteed to converge to a global minimum. A variety of numerical solutions can be used for this purpose like the Euler's method or the Runge-Kutta's method.

5 Visualization of the Model

Once the dynamics of the model has been solved for the cloth using the above technique, the model has to be visualized for observing the drape. A simple, but powerful enough B-Spline model with the Painters algorithm to eliminated hidden surfaces can be used for this.

B-Spline surfaces have the undesired property that the control points do not lie on the surface. To eliminate this, the given set of control points have to be converted into another set whose B-Spline passes through these points [BG80]. So the points $P_{i,j}$ have to be converted into new control points $V_{i,j}$. The corner points are give by $V_{0,0} = P_{0,0}, V_{0,n} = P_{0,n}, V_{m,0} = P_{m,0}$ and $V_{m,n} = P_{m,n}$. The one set border points are given by $P_{0i} = \frac{(V_{0,i-1} + 4V_{0,i} + V_{0,i+1})}{6}, 1 \leq i \leq n-1$, and similarly the remaining border points can be obtained. The interior points are given by

$$\begin{aligned} P_{i,j} = \frac{1}{36}(&V_{i-1,j-1} + 4V_{i-1,j} + \\ &V_{i-1,j+1} + 4V_{i,j-1} + 16V_{i,j} + 4V_{i,j+1} + \\ &V_{i+1,j-1} + 4V_{i+1,j} + V_{i+1,j+1}) \end{aligned}$$

for $i = 1, \ldots, m-1$ and $j = 1, \ldots, n-1$. Standard Gauss-Elimination is applied to solve for the new control points.

6 Final remarks and representative results

Cloth modeling on the computer is a difficult problem, but research in recent years have gravitated to physically-based models such as the one proposed by [VTJT96, BHW94a].

In this paper, we further investigate such methods to allow for external forces that change the geometry of cloth drape. We conclude that particle-system based modeling of cloth drape is a viable alternative to the surface panel based models in many, but not all, situations. Introduction of dormancy as an attribute of the particle allows for increased realism using reduced computations. In particular, effects of viscous media outside cloth that change its behavior has been modeled in this work which builds on prior work in this area [BHW94a].

Resistance offered by air was embedded into the energy functions using concepts from fluid mechanics under the norms of physically-based modeling techniques. Figure 10 and Figure 11 — representative results for the purposes of this paper — show the drape of cloth without and with the introduction of the air-resistance model under our implementation. Since the particles are under constraints, even before they are in contact with the object beneath, they travel at different speeds. Because of the parabolic approximation of pressure distribution, a more realistic animation of falling drape can be obtained, as particles near the center travel slower than those bordering the cloth.

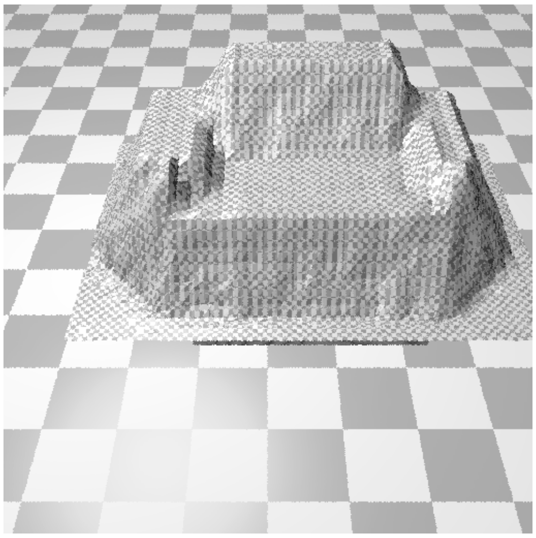

Figure 10: Drape using prior models.

Figure 11: Drape with our enhanced model.

References

[Aon90] Masaki Aono. Wrinkle propogation model for cloth. In T.L. Kunii and T.S. Chua, editors, *Computer Graphics Around the World (Proc. International Conference on Computer Graphics)*, pages 95–115. Springer-Verlag, Tokyo-Japan, 1990.

[Bar84] Alan H. Barr. Global and local deformation of solid primitives. *Computer Graphics*, 18(3):21–30, July 1984.

[BG80] Brain A. Barsky and Donald P. Greenberg. Determining a set of B-spline control vertices to geneterate an interpolating surface. *Computer Graphics and Image Processing*, 14:227–248, 1980.

[BHW94a] David E. Breen, Donald H. House, and Michael J. Wonzy. A particle-based model for simulating the draping behavior of woven cloth. *Textile Research Journal*, 64(11):663–685, November 1994.

[BHW94b] David E. Breen, Donald H. House, and Michael J. Wonzy. Predicting the drape of woven cloth using interacting particles. *Conference Proceedings SIGGRAPH*, pages 365–372, July 1994. http://www.ecrc.de/staff/david.

[CIMM92] S. Chandran, J. Inagawa, S. Miyazaki, and T. Maejima. Animating non-rigid objects: Computer vision technique applied to physical-based models. In Tosiyasu L. Kunii, editor, *Visual Computing*, chapter 15, pages 843–853. Springer-Verlag, Tokyo, Japan, 1992.

[CYTT92] Michel Carignan, Ying Yang, Nadia Magnenat Thalmann, and Daniel Thalmann. Dressing animated synthetic actors with complex deformable clothes. In Edwin E. Catmull, editor, *Computer Graphics (SIGGRAPH '92 Proceedings)*, volume 26, pages 99–104, July 1992.

[DMTM93] Sanjay G. Dhande, Rao P. V. M., Shahariar Tavakkoli, and Carolyn L. Moore. Geometric modeling of draped fabric surfaces. In Mudur S. P. and Pattanail S. N., editors, *Graphics, Design and Visualization (Proc. International Conference on Computer Graphics)*, pages 173–180. Jaico Publishing House, Bombay-India, February 1993.

[KG90] Tosiyasu L. Kunii and Hironobu Gotoda. Singularity theoretical modeling and animation of garment wrinkle formation process. *The Visual Computer*, 6(6):326–336, 1990.

[LDG93] Li Ling, M. Damodaran, and Robert K. L. Gay. A quasi-steady force model for animating cloth motion. In Mudur S. P. and Pattanaik S. N., editors, *Graphics, Design and Visualization (Proc. International Conference on Computer Graphics)*, pages 181–188. Jaico Publishing House, Bombay-India, February 1993.

[MBL+91] James V. Miller, David E. Breen, William E. Lorensen, Robert M. O'Bara, and Michael J. Wonzy. Geometrically deformed models: A method for extracting closed geometric models from volume data. *Computer Graphics (Proc. SIGGRAPH)*, 25(4):217–226, July 1991.

[MS91] P.N. Modi and S.M. Seth. *Hydraulics and Fluid Mechanics*. Standard Book House, Post Box 1074, 1705-A, Nai Sarak, Delhi-110006, tenth edition, 1991.

[NG96] H. Ng and R. Grimsdale. Computer graphics techniques for modeling cloth. *IEEE Computer Graphics and Applications*, 16(5):15–24, 1996.

[OTMS95] Kenichi Ohta, Hideyuki Tanaka, Fujio Miyawaki, and Mitzuya Saeki. Dynamic deformation of textile design. In *Asian Conference on Computer Vision*, volume 2, page 220. Nanyang Technological University, Singapore, Dec 1995.

[Ree83] William T. Reeves. Particle system-a technique for modeling a class of fuzzy objects. *ACM Transaction on Graphics*, 2(2):91–108, April 1983.

[Rey87] Craig W. Reynolds. Flocks, herds, and school: A distributed behavioral model. *Computer Graphics (Proc. SIGGRAPH)*, 21(4):25–34, July 1987.

[TPFB87] Demetri Terzopoulos, John Platt, Kurt Fleischer, and Alan Barr. Elastically deformable models. *Computer Graphics (Proc. SIGGRAPH)*, 21(4):205–214, 1987.

[VCT95] Pascal Volino, Martin Courshesnes, and Nadia Magnenat Thalmann. Versatile and efficient techniques for simulating cloth and other deformable objects. In Robert Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 137–144. ACM SIGGRAPH, Addison Wesley, August 1995. held in Los Angeles, California, 06-11 August 1995.

[VT94] P. Volino and N. M. Thalmann. Efficient self-collision detection on smoothly discretized surface animations using geometrical shape regularity. In *Computer Graphics Forum*, volume 13, pages 155–166. Eurographics, Basil Blackwell Ltd, 1994. Eurographics '94 Conference issue.

[VTJT96] P. Volino, N. Thalmann, S. Jianhua, and D. Thalmann. An evolving system for simulating clothes on virtual actors. *IEEE Computer Graphics and Applications*, 16(5):42–51, 1996.

[Wei86] Jerry Weil. The synthesis of cloth objects. *Computer Graphics (Proc. SIGGRAPH)*, 20(4):49–54, August 1986.

[WFB87] Andrew Witkin, Kurt Fleischer, and Alan Barr. Energy constraints on parameterized models. *Computer Graphics (Proc. SIGGRAPH)*, 21(4):225–232, July 1987.