To appear in the ACM SIGGRAPH conference proceedings

Simulating Knitted Cloth at the Yarn Level

Jonathan M. Kaldor Doug L. James Steve Marschner

Cornell University

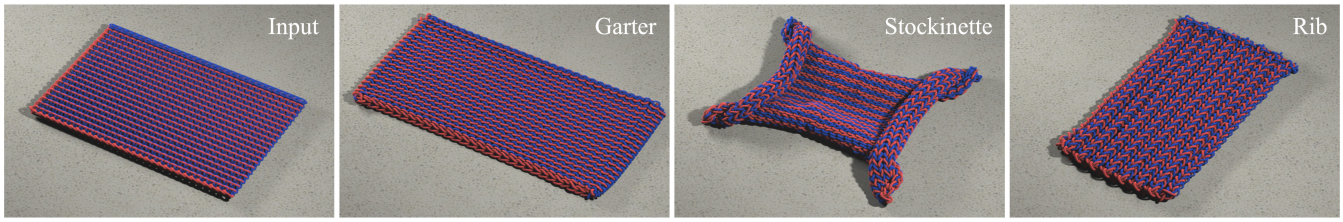


Figure 1: Yarn-level cloth: *Three knitting patterns relaxed to rest in our simulator. Each starts from a flat input configuration like at left; they differ only in their interlocking patterns. Characteristic shapes and textures of each knit emerge from our yarn-level physical model.*

Abstract

Knitted fabric is widely used in clothing because of its unique and stretchy behavior, which is fundamentally different from the behavior of woven cloth. The properties of knits come from the nonlinear, three-dimensional kinematics of long, inter-looping yarns, and despite significant advances in cloth animation we still do not know how to simulate knitted fabric faithfully. Existing cloth simulators mainly adopt elastic-sheet mechanical models inspired by woven materials, focusing less on the model itself than on important simulation challenges such as efficiency, stability, and robustness. We define a new computational model for knits in terms of the motion of yarns, rather than the motion of a sheet. Each yarn is modeled as an inextensible, yet otherwise flexible, B-spline tube. To simulate complex knitted garments, we propose an implicit-explicit integrator, with yarn inextensibility constraints imposed using efficient projections. Friction among yarns is approximated using rigid-body velocity filters, and key yarn-yarn interactions are mediated by stiff penalty forces. Our results show that this simple model predicts the key mechanical properties of different knits, as demonstrated by qualitative comparisons to observed deformations of actual samples in the laboratory, and that the simulator can scale up to substantial animations with complex dynamic motion.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling I.6.8 [Simulation and Modeling]: Types of Simulation—Animation,

Keywords: Cloth, simulation, knits, knitwear, yarn, constraints

1 Introduction

Most research on cloth mechanics, both in computer graphics and in other fields, has focused on woven cloth, both for its simplicity and because many fabrics used in engineering applications are woven. In computer graphics, however, clothing is the predominant application for cloth simulators—and in clothing, knit fabrics are as commonly used as wovens. Many very common garments, such as T-shirts or leggings, owe their existence to knits and cannot be made from woven material.

The distinction between knits and wovens is important for simulation because their mechanical structures are entirely dissimilar, and as a result they behave differently at all scales. The yarns in woven fabric are nearly immobile, leading to an almost inextensible sheet with limited deformations in the yarn structure. In contrast, the interlocked loops in a knit material deform and slide readily, leading to a highly extensible sheet with dramatic changes in small-scale structure as the material stretches.

Cloth simulation generally uses models that approximate the mechanics of linear elastic sheets. Because of the small in-plane deformations of woven materials, acceptable realism can often be achieved for woven fabric using these models. But linear-elastic sheet models inevitably look "rubbery" if they are allowed to stretch as much as a typical knit fabric does. This is unsurprising, since the mechanics of interlocking loops in a knit fabric bears little resemblance to the mechanics of a continuous elastic material—a fundamentally different kind of model is required. The small-scale behavior of knits is also important because many knits are made with large yarns, meaning that the yarn structure is clearly visible and must behave correctly for realistic results.

The simulation we propose in this paper meets the challenge of knits head-on, by directly solving for the motion of the yarn that makes up the fabric and the interactions between loops that determine its behavior. Our physical model is concisely described by the behavior of a single yarn: an inextensible curve with resistance to bending, a collision force that resists interpenetration, and damping for stability and to stand in for the effects of friction. Based on that model, we demonstrate the first practical yarn-level simulations of significant yarn structures, producing rich, complex deformations that are impossible to achieve using any kind of sheet-based simulation. Many of the properties of knit structures emerge naturally from the simulation, including the characteristic shapes and textures produced by different knitting patterns and the varying extensibility of the knit sheets. Evaluation of repetitive yarn-level computations can exploit multicore architectures naturally, allowing for large knits to be simulated in practice.

A simulation at this level of detail is required for realistic results with coarse-knit garments like sweaters, scarves, or socks, because of their visible yarn structure. Furthermore, yarn-level simulation is

a fundamental tool for studying the large-scale properties of finely knit fabric, in order to develop continuum models that can realistically describe knits under large deformations. The same approach can also lead to models for wovens that are able to capture the material-dependent subtleties missed by current models.

In the following sections we detail our model, the methods used to simulate it, and the results of our simulations, including qualitative laboratory validation of the deformations of similar structures.

2 Prior Work

Cloth has been modeled in a variety of different ways in the literature. Perhaps the most straightforward approach, and the one primarily used in the computer graphics community, is to treat the cloth as an elastic sheet, typically one that is linearly elastic and isotropic. These models are either explicitly continuous [Terzopoulos et al. 1987] or a discrete approximation to some continuous surface [Baraff and Witkin 1998]. Extensions to these models have focused on speeding up the computation time [Volino and Thalmann 2000], simulating stable behavior under compression [Choi and Ko 2002], revised models of bending [Bridson et al. 2003; Grinspun et al. 2003], or stable collision processing [Volino and Thalmann 2000; Bridson et al. 2002; Baraff et al. 2003]. Focus has also gone into limiting the amount of stretching fabric can undergo, either by a strain-limiting iterative process [Provot 1995; Bridson et al. 2002] or a constraint satisfaction phase [Goldenthal et al. 2007]. While simulation speeds are relatively fast, there is in general a problem of mapping physical cloth properties to the parameter space of the elastic model. Jojic and Huang [1997] used range scan data of static cloth configurations to estimate the elastic parameters. Bhat et al. [2003] used video data of moving cloth to estimate the elastic and damping parameters, with the experimentally determined parameters for the knit sample varying noticeably; this suggests that the elastic model may not be a good fit for knitted materials.

Several models have attempted to address the fact that cloth is comprised of a discrete set of yarns. Geometric modeling of yarns arguably began with Peirce [1937], who derived a set of parameters and equations for modeling the crossing of yarns in a woven fabric as inextensible curves. Kawabata et al. [1973] proposed a beam and truss model for yarn crossings in a woven fabric, as well as a system for measuring the physical force curves resulting from stretch, shear, and bend motions of cloth. Variations of the beam-and-truss model have been used in the textile community to simulate the behavior of plain-weave fabrics like Kevlar [King et al. 2005; Zeng et al. 2006]. Breen et al. [1994] and Eberhardt et al. [1996] modeled woven fabric as a particle system, where a particle ideally represented a yarn crossing.

Woven yarn crossings have also been modeled as a pair of curves [Warren 1990]. In particular, Nadler et al. [2006] employed a two-scale model, treating cloth at the high level as a continuous sheet, and at the fine level as a collection of pairs of orthogonal curves in contact with each other, with feedback from the fine scale driving the simulation of the large scale. Yarns have also been modeled as splines, with Rémion [1999] developing the basic equations for using splines in a knit; however, they used springs between the control points to preserve length. Jiang and Chen [2005] used a spline-based yarn model to generate plausible static woven fabric configurations. Similar to this, there has also been work done in computer graphics on modeling and simulating thin flexible rods [Pai 2002; Bertails et al. 2006; Theetten et al. 2007; Spillmann and Teschner 2008], although the simulated rods are typically much shorter than the spline curves used in our cloth.

The work of Chu [2005] is similar to the current work in that both use B-splines to simulate fabric, with similar terms for collisions, but the former is focused on woven fabrics and allows the yarns to stretch, which requires a much smaller timestep than our simulator for stable results. In addition, the integral for the contact model is approximated by using closest points in a set of contacts predetermined by the cloth structure at initialization time. Our collision evaluation makes no assumptions about cloth topology, allowing arbitrary collision regions typical of cloth-cloth contact, while scaling to support the simulation of large, arbitrary knits.

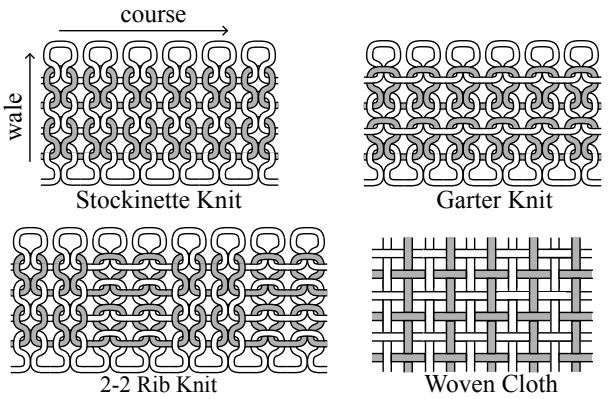


Figure 2: Interlocking loop structures *of three knitting patterns used in our examples compared to standard woven cloth.*

Because of their relative complexity compared to woven fabrics, knits are not as well-studied. Eberhardt et al. [2000] model knits as a continuous sheet with force curves derived from Kawabata measurements. Nocent et al. [2001] deform a thin sheet and project the changes to the sheet to an underlying spline-based geometry. Several works in the textile community have focused on generating knit geometry using spline curves, typically assuming incompressible yarns and specific geometric constraints [Demiroz and Dias 2000; Göktepe and Harlock 2002; Choi and Lo 2003]; only Choi [2003] attempts to simulate the resulting geometry. Chen et al. [2003] are primarily concerned with rendering knit geometry, and use as their base model a system of key points mapped to a mass-spring mesh. In contrast, we allow for compressible yarns through our collision model and have no geometric constraints on the curves, allowing them to take a natural instead of a prescribed shape and supporting arbitrary knit structures. Finally, we are able to scale up our simulations to model large knit structures like loose scarves and leg warmers; to our knowledge, ours is the first practical implementation capable of this.

3 Structure of Knitted Cloth

The yarns that comprise cloth are themselves formed from fibers, either long filaments like silk or shorter fibers like cotton, which are twisted so that friction holds the yarns together. As a result, the fibers at the core of the yarn tend to have little relative movement, and the yarn as a whole resists stretching. Much research has been done in the textile community on fiber-level models of yarn and how the interactions at the fiber level lead to yarn-level behavior, particularly bending rigidity [Park and Oh 2003; Park and Oh 2006]. In particular, Choi and Tandon [2006] develop a model of a multi-ply yarn, showing that their model reasonably approximates experimental results and predicts the strain energy of bending to be approximately quadratic with respect to the curvature.

3.1 A Brief Primer on Knits and Knitting

Cloth can in general be divided into two broad categories: woven and knit fabrics. Woven fabrics are comprised of two sets of yarns,

Figure 3: Views of three knitted samples: *Note the large differences in dimensions for the same number of stitches and length of yarn. The predictions of our model (Figure 1) are qualitatively similar.*

the warp and the weft, organized into two perpendicular directions on the cloth surface. In contrast, the yarns in a knitted fabric are organized into a regular set of loops in horizontal rows. The loops from each horizontal row of a knit are pulled through the loops of the previous row, either in a "knit" stitch (up through the previous loop) or a "purl" stitch (down through the previous loop). The two primary directions in a knit are called the *course* and the *wale*, with the course traveling in the direction of a single row of loops and the wale traveling in the direction of the stack of loops. Typically, when the yarn reaches the end of a row of a knit, it then doubles backs and forms the next row as well. As a result, knits consist of only a few yarns, which is in contrast to woven fabrics which consist of many yarns. The first and last row of stitches are special stitches known as bind-ons and bind-offs, respectively, which serve to keep the knit from unraveling, while the beginning and end of the yarn is either pulled back through the fabric several times and held in place by friction or simply knotted off.

Alternating between knit and purl stitches results in much of the variety in knitted fabrics, with three of the most common varieties being the stockinette (all "knit" stitches), the garter (alternating rows of "knits" and "purls"), and the 2-2 rib (each row consists of repetitions of 2 "knit" stitches followed by 2 "purl" stitches)[1]. In Figure 2, we show samples of the three styles of knitting from above. The garter is the simplest of the three, and has the same overall pattern on both sides of the fabric. In comparison, the stockinette is different on the front and the back, which leads to some dramatic curling behavior on the edges. The rib is much shorter in the course direction than either of the other two, again because of this same curling behavior. As can be seen from the pattern, it is essentially 2 columns of the front side of a stockinette followed by 2 columns of the back side of a stockinette. These columns curl like the regular stockinette, with adjacent columns curling in opposite directions, compressing the rib knit greatly and giving it a tremendous degree of stretchiness; typically, the cuffs of shirts and sweaters and the ankles of socks are made out of ribbed stitching. Examples of these fabric are shown in overview in Figure 3.

3.2 Qualitative Mechanical Behavior

As a result of its construction, the deformations of cloth are multiphasic, particularly when being stretched. When under tension, the cloth first begins unrolling from any compression caused by curling. This is particularly evident in ribbed knits, where the columns of the front-facing stockinette stitch are pulled apart, revealing the columns of back-facing stockinette stitch. After that, the cloth then begins deforming its woven or knit structure. In the case of a woven fabric, the warp-weft intersections become compressed, while in a knit fabric the loops are stretched in one dimension while being compressed in the other. Because the loops are typically free to undergo much larger deformations than the compression of the intersections in a woven fabric, knits tend to be much stretchier than their woven counterparts. At some point, however, the cloth is unable to deform much more in this fashion and so additional load causes the yarns themselves to stretch. As noted above, though, yarns are very resistant to stretching, which results in a sharp increase in the load curve at this point.

It is also important to note that stretching behavior in one dimension affects the characteristics of the other dimension as well. For instance, in a knit, as the loops are stretched in one dimension they compress in the other dimension, sometimes quite noticeably; as a result, capturing these couplings is important for visual accuracy. Although some current cloth simulators are capable of expressing these types of relationships, it can be difficult to tune the parameters correctly, and oftentimes they are ignored.

4 A Yarn-Level Cloth Model

Our cloth is modeled using the individual yarns that comprise it (see Figure 4 for an overview). Without loss of generality, we will assume that our knits are constructed using a single yarn. The yarn is an open cubic B-spline curve with a constant radius of r described by the control points $\mathbf{q} \in \mathbb{R}^{3m}$. In general, indices i, j range over spline segments, while indices k, l range over control points. The curve is described by $\mathbf{y}(s) = \sum b_k(s)\mathbf{q}_k$, $s \in [0, N]$ for a yarn with $N = m - 3$ spline segments, where $b_k(s)$ is the cubic B-spline basis function associated with control point k. Similarly, the velocity of the yarn at parametric point s is $\mathbf{v}(s) = \sum b_k(s)\dot{\mathbf{q}}_k$. For convenience, the curve restricted to a particular spline segment i is denoted $\mathbf{y}_i(s)$, $s \in [0, 1]$ (and $\mathbf{v}_i(s)$ for the velocity). Each spline segment has a fixed arclength ℓ_i. The yarn has a mass per unit length of m_{unit}, and mass is spread along the curve according to the function $m(s) = m_{\text{unit}}\ell_{\lfloor s \rfloor}$, a piecewise constant function that assigns mass to segments according to their arclength and then spreads the mass uniformly in parameter space.

We model the yarn's time evolution using the equations of motion of constrained Lagrangian dynamics; see Goldstein et al. [2002] for a further description of Lagrangian mechanics, and Rémion et al. [1999] for its application to spline curves. In addition, some of the stiffer properties of the cloth are enforced via constraints. The result is a differential algebraic equation (DAE) of the form,

$$\mathbf{M}\,\ddot{\mathbf{q}} = -\nabla_{\mathbf{q}} E(\mathbf{q}) - \nabla_{\dot{\mathbf{q}}} D(\dot{\mathbf{q}}) + \mathbf{f} \quad (1)$$

$$\mathbf{C}(\mathbf{q}) = \mathbf{0}, \quad (2)$$

[1] When hand knitting, the work is typically turned over after each row, which reverses the notation of stitches (i.e., a "purl" stitch when flipped looks like a "knit" stitch from the front). As a result, these definitions of a stockinette and a garter are reversed from standard hand-knitting definitions.

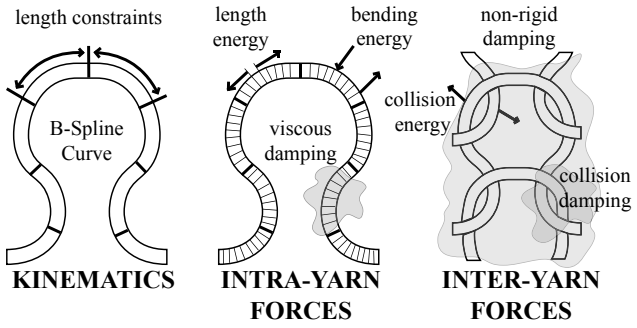


Figure 4: Summary of yarn-level model: *Yarns are splines with constraints fixing the arc length of each segment. Internal forces resist bending and intra-segment stretching, and external forces repel colliding lengths of spline. Damping forces are applied to the yarns, at the collisions, and to areas of cloth.*

where $\mathbf{M}$ is the mass matrix, $E(\mathbf{q})$ is the sum of all positional energy terms, $D(\dot{\mathbf{q}})$ is the sum of all "damping energy" terms, $\mathbf{f}$ are external forces, and $\mathbf{C}(\mathbf{q})$ is a vector of constraint functions.

4.1 Intra-Yarn Properties

Mass: The kinetic energy of the yarn is:

$$T(\dot{\mathbf{q}}) = \sum_{i=1}^{N} m_{\text{unit}}\, \ell_i \int_0^1 \mathbf{v}_i(s)^T \mathbf{v}_i(s)\, ds \tag{3}$$

In order to apply Lagrangian mechanics, we must compute $\frac{d}{dt}\left(\nabla_{\dot{\mathbf{q}}} T(\dot{\mathbf{q}})\right)$. By expanding the integral on the right hand side of Equation 3, $\nabla_{\dot{\mathbf{q}}} T(\dot{\mathbf{q}})$ can be rewritten as $\mathbf{M}\dot{\mathbf{q}}$, where $\mathbf{M}_{k,l} = \int_0^N m(s) b_k(s) b_l(s) ds$. Because this depends only on the arclengths, m_{unit}, and basis functions, all of which remain constant during simulation, this matrix can be precomputed, and because the cubic B-spline functions have local support, the matrix is sparse with upper and lower bandwidth of 12. Taking the derivative with respect to t yields $\mathbf{M}\ddot{\mathbf{q}}$, the left hand side of (1)

Bending: Bending resistance is modeled by a bend energy density functional which is quadratic in curvature:

$$E_i^{\text{bend}} = k_{\text{bend}}\, \ell_i \int_0^1 \kappa_i(s)^2\, ds, \tag{4}$$

where $\kappa_i(s) = \frac{\|\mathbf{y}_i'(s) \times \mathbf{y}_i''(s)\|}{\|\mathbf{y}_i'(s)\|^3}$ is the unsigned curvature of spline segment i at s.

Inextensibility: Because of their high resistance to stretching relative to the cloth, we model yarns as inextensible. Ideally this would be a constraint at the infinitesimal level; however, we need to ensure that we do not lock up the system by removing too many degrees of freedom. As a result, we define one length constraint on each spline segment

$$C_i^{\text{len}} = 1 - \frac{1}{\ell_i} \int_0^1 \|\mathbf{y}_i'(s)\|\, ds. \tag{5}$$

This constraint ensures that the total length of the segment remains constant, but it does not necessarily keep the mass of the spline from sliding around inside the curve as the parameterization speed changes; as long as the overall length is constant, the infinitesimal length can change without penalty. To prevent this, we also introduce a set of energy terms,

$$E_i^{\text{len}} = k_{\text{len}} \int_0^1 \left(1 - \frac{\|\mathbf{y}_i'(s)\|}{\ell_i}\right)^2 ds, \tag{6}$$

where k_{len} is a stiffness coefficient. It should be noted that this term does not have to be particularly stiff due to the use of length constraints, since it only needs to resist the stretching or compression of mass in a local area. For instance, in a piece of yarn hanging vertically k_{len} only needs to be stiff enough to support the weight of a single spline segment, while without the constraint term k_{len} needs to be stiff enough so that the first segment can support the weight of the *entire* yarn.

4.2 Yarn-Yarn Collisions

Yarn collision forces are modeled with an energy term,

$$E_{(i,j)}^{\text{contact}} = k_{\text{contact}}\, \ell_i \ell_j \int_0^1 \int_0^1 f\left(\frac{\|\mathbf{y}_j(s') - \mathbf{y}_i(s)\|}{2r}\right) ds\, ds', \tag{7}$$

for i, j such that $|i-j|>1$, where $f(d)$ is defined such that $f(d) \to 0$ as $d \to 1$, $f'(d) \to 0$ as $d \to 1$, and $f(d) \to \infty$ as $d \to 0$. In our implementation, we use

$$f(d) = \begin{cases} \frac{1}{d^2} + d^2 - 2, & d < 1 \\ 0, & \text{otherwise} \end{cases}. \tag{8}$$

We found this collision model to be physically and computationally more robust than ones based on closest point distances. In addition, this approach by definition also handles arbitrary cloth self-collisions such as those seen in folding and bunching.

4.3 Damping and Friction

Damping and friction in knitted cloth structures are complex, with significant hysteresis effects. The interlooped structure of knits creates large contact regions, and for yarns made of short fibers the direct contact between yarns combines with the mass of intertwined stray fibers, or "fuzz," that resists relative motion between nearby yarns. Accurate yarn-level modeling of such phenomena is beyond the scope of this paper. Instead, we employ three damping models of practical importance:

Mass-proportional damping is a classic way to dissipate any motion, and it is our most basic damping force component. Damping is applied uniformly to the yarn according to the following damping energy term:

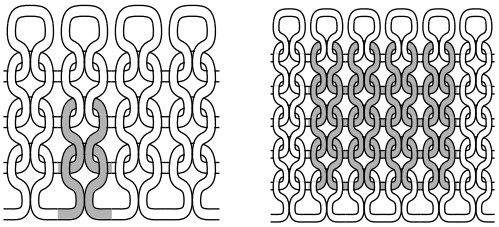
$$D_i^{\text{global}} = k_{\text{global}} \int_0^1 \mathbf{v}_i(s)^T \mathbf{v}_i(s) ds. \tag{9}$$

Because the density of our yarns is constant, the mass dependence is effectively pushed into k_{global}. We make extensive use of inherently stable mass-proportional damping model during knit structure initialization (§6). During actual simulation, however, k_{global} is typically turned off, and we rely on the following two damping models to damp the motion of the cloth.

Contact damping: A yarn-yarn collision damping term $D_{(i,j)}^{\text{collision}}$ is used both to damp the stiff yarn-yarn contact forces and to approximate sliding friction, defined as:

$$\ell_i \ell_j \int_0^1 \int_0^1 \Big(k_{dt} \|\Delta \mathbf{v}_{ij}\|^2 - (k_{dt} - k_{dn})(\hat{\mathbf{n}}_{ij}^T \Delta \mathbf{v}_{ij})^2\Big) ds\, ds', \tag{10}$$

where $k_{dt} \geq 0$ controls damping in the tangential direction, and $k_{dn} \geq 0$ controls damping in the normal direction; $\Delta \mathbf{v}_{ij} = \Delta \mathbf{v}_{ij}(s, s') = \mathbf{v}_j(s') - \mathbf{v}_i(s)$ is the relative velocity; and $\hat{\mathbf{n}}_{ij} = \hat{\mathbf{n}}_{ij}(s, s')$ is the normalized value of the collision direction, $\mathbf{n}_{ij}(s, s') = \mathbf{y}_j(s') - \mathbf{y}_i(s)$. This integral is only evaluated where the yarns are determined to be in contact with each other according to Equation 8



Small rigid damping region (repeat every row / column)

Large rigid damping region (repeat every 2 rows / 2 columns)

Figure 5: Regions used for the non-rigid damping velocity filter

Non-rigid motion damping: Modeling the dissipative effects of "fuzz" properly is a rather difficult problem. While we do not wish to model it explicitly, we also do not want to ignore it, since we believe that it is an important characteristic of cloth to capture. Here we present a simple model that works well in practice, and we leave the development of a more advanced model for future work.

In order to resist relative motion between nearby sections of yarn, we damp non-rigid motion [Müller et al. 2006]. The cloth is broken up into fixed overlapping regions as in Rivers and James [2007], and at each step the center of mass, angular momentum, and inertia tensor of each region are computed. The yarn in each region is then damped according to $\frac{\alpha}{r(s)}(\mathbf{v}_{\text{rigid}}(s) - \mathbf{v}(s))$, where $\mathbf{v}_{rigid}(s)$ is the expected rigid motion of point s. The parameter $\alpha \in [0, 1]$ controls how strong the damping is, and $r(s)$ is the number of fixed regions containing the point s.

There are several ways to break up the cloth into regions. We use a two-pass filter over parametrically static regions defined during yarn initialization, first heavily damping small regions of two yarn loops and then damping the motion of larger regions (see Figure 5). The first pass is designed to damp out motion locally where the yarns loop around each other, and the second pass damps stretching, shearing, and bending modes.

4.4 Additional Constraints and Contacts

In order to prevent knitted cloths from unraveling, the end of the yarn is typically either knotted off or pulled through several loops and held in place by friction. We accomplish the same effect by "gluing" the end of the yarn to another piece of yarn via a constraint of the form $\mathbf{C}^{\text{glue}} = \mathbf{y}(s_1) - \mathbf{y}(s_2)$ for particular choices of s_1 and s_2. Similarly, when the cloth needs to be pinned in place, vector constraints of the form $\mathbf{C}_i^{\text{pin}} = \mathbf{y}(s_i) - \mathbf{p}_i$ are inserted. It is important to avoid introducing too many hard constraints, which can lead to overconstrained or near-singular systems, or degraded quality in yarn dynamics.

We approximate object collisions in one of two ways. We resolve contact with implicit surfaces (such as our leg warmer example) using penalty forces modeled via an energy term,

$$\mathbf{E}_i^{\text{obj}} = k_{\text{obj}} \int_0^1 \left\{ \begin{matrix} (f(\mathbf{y}_i(s)) - f_0)^2, & f(\mathbf{y}_i(s)) < f_0 \\ 0, & \text{otherwise} \end{matrix} \right\} ds, \quad (11)$$

along with corresponding damping forces (analogous to yarn-yarn collision damping (10)). Alternately, for objects with distance fields (such as the scarf falling on a plane), we employ a velocity filter: a normal impulse is applied that resolves interpenetration, along with an approximate frictional impulse (c.f. [Bridson et al. 2002]).

```
For each timestep, h
    [q, v] = unconstrained_step(q, v, t)
    [q, v] = satisfy_constraints(q, v)
    v      = filter_velocity(q, v)
    t      = t + h
end
```

Figure 6: Overview of time-stepping scheme

5 Integrating Yarn Dynamics

Implementing our yarn-level model requires careful choice of simulation methods and attention to several crucial details in evaluating the terms presented in (§4). Figure 6 contains an overview of the steps in our simulator.

5.1 Integration Method

The DAE is stepped forward in time using the Implicit Constraint Direction (ICD) method [Goldenthal et al. 2007], using Algorithm 1 in that paper. We use an explicit midpoint step as our unconstrained step. The algorithm iterates until convergence, with each iteration requiring a sparse linear system solve of a matrix which depends on the inverse mass matrix and the Jacobian of the constraint function. To speed up the simulation and simplify inverse mass computation, we lump the mass matrix along the diagonal. We found that at most 4-5 iterations were usually needed for acceptable convergence, and oftentimes only 1 or 2.

Although the integrals for the mass matrix and for global damping are readily solved, the others lack efficient closed forms. As a result, we used Simpson's quadrature at fixed positions in parameter space to evaluate the integrals in (4), (5), (6), (7), (10), and (11), typically using 11 quadrature points per spline segment. We plan to explore the use of efficient adaptive quadrature rules for some of the integrals, in particular the collision integral.

5.2 Yarn Collisions

In practice, expanding the integrals in (7) and (10) is the bottleneck for the simulation, so they must be computed efficiently. Naive evaluation is exceedingly slow, because it involves a double integral over the entire yarn. However, it should be noted that the integrand is zero over the vast majority of the integration domain, since yarn segments typically contact only a few neighbors. To compute this integral effectively, we exploit spatial culling: we generate bounding spheres with a radius equal to the radius of the yarn at fixed quadrature points in parameter space, and insert them into a static AABB hierarchy that is generated at the beginning of the simulation. To evaluate the integral, we intersect the hierarchy with itself to determine the quadrature point-pairs requiring evaluation. In addition, both the tree traversal and contact force evaluations can be parallelized across multiple cores.

5.3 Velocity Filters

We also allow velocity filters to update control point velocities directly. Most previous velocity filters are used on discrete particle systems, however, our system models a continuous curve. In order to easily generate filters for the spline's control points, we split the curve into a set of disjoint sample segments, typically using 6–10 sample segments per spline segment. The desired impulse $\Delta\mathbf{v}(s)$ is computed for each sample segment, and the resultant impulse applied to the k^{th} control point is then $b_k(s)\Delta\mathbf{v}(s)$. All of the control point impulses are accumulated and then multiplied by the lumped $\mathbf{M}^{-1}$ to produce the actual change in velocity $\Delta\dot{\mathbf{q}}$ for the control

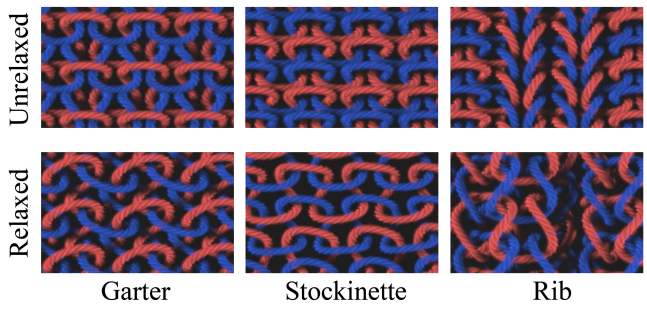


Figure 7: Small-scale structure *of the three knitting patterns before and after relaxation (yarn radius shrunk for clarity).*

points. Finally, $\dot{\mathbf{q}}_{\text{new}} = \dot{\mathbf{q}} + \Delta\dot{\mathbf{q}}$. To prevent impulses from affecting each other, all impulses for a particular velocity filter are computed first, then applied together. The non-rigid damping and non-penalty-based object collisions (for objects with distance fields) are both handled with a velocity filter.

6 Initializing Yarn Configuration

There are a variety of ways to generate initial knit configurations. For instance, methods have been proposed for generating knit geometry by simulating the knitting process itself [Eberhardt et al. 2000]. In order to generate an initial configuration, our algorithm takes as input a knit pattern, the number of spline segments k to generate per stitch, and a set of curves which describe the basic shape of the various kinds of stitches. In particular, it expects a model of a general loop (which can be flipped along the z-axis to form either a knit or a purl stitch) as well as models of the various types of stitches which can occur on the boundaries of the cloth. It then forms a single spline curve to describe the fabric by laying down one stitch at a time and finding the best least-squares cubic B-spline approximation of that stitch using k segments and given the control points already added by the previous stitch. The first (last) stitch are modeled with a special type of stitch where the beginning (end) of the yarn connects to another part of the stitch, forming a loop; these endpoints are then "glued" to the closest point using the glue constraints from (§4.4)

The goal of the model generation is to obtain a configuration where all of the loops are properly interconnected according to the specified pattern, but it is not necessarily the rest state. To find the rest state, we simulate the pattern using our model, but without the hard constraint on length and with a high length energy coefficient and high viscous damping. In addition, the yarn is shrunk by setting the desired arclength ℓ_i of each spline segment to $c\ell_i^0$, where ℓ_i^0 is the starting arclength and $c<1$ is a shrinking factor, e.g., in our simulations, $c=0.935$. This causes the entire cloth to compress and settle into a general rest state, which we can then use as a cloth sample in our simulations.

7 Results

Our simulator is implemented in Java and was run on machines with two 4-core Intel Xeon processors clocked at 2.66GHz. Simulation parameters common to all scenes are in Table 1, while scene-specific parameters and details are available in Table 2. The renderings in the paper and video were made using a software implementation of the Lumislice method [Chen et al. 2003] in a ray tracer. Our implementation follows the original except that it uses volume ray tracing rather than alpha blending to accumulate light through the volume; it uses first-order spherical harmonics, rather than a directional table, to store the Volume Reflectance Function; and it uses distribution ray tracing, rather than a shadow map, to compute shadows from area sources. The expensive shadow computations are performed at regularly spaced points throughout the yarn volume, then interpolated as the yarn volume is traversed. Rendering times range from 4 to 15 minutes per frame, on the same hardware as used for simulation.

	Relaxation	Simulation
r	0.125 cm	0.125 cm
m	0.006 $\frac{\text{g}}{\text{cm}}$	0.006 $\frac{\text{g}}{\text{cm}}$
k_{len}	10000 $\frac{\text{g cm}^2}{\text{s}^2}$	2000 $\frac{\text{g cm}^2}{\text{s}^2}$
k_{bend}	0.005 $\frac{\text{g cm}^2}{\text{s}^2}$	5 $\frac{\text{g cm}^2}{\text{s}^2}$
k_{global}	1.5 $\frac{\text{g}}{\text{s}}$	0 $\frac{\text{g}}{\text{s}}$
k_{contact}	3250 $\frac{\text{g}}{\text{s}^2}$	3250 $\frac{\text{g}}{\text{s}^2}$
k_{dt}	0.001–0.005 $\frac{\text{g}}{\text{cm}^2\text{s}}$	0.001–0.005 $\frac{\text{g}}{\text{cm}^2\text{s}}$
k_{dn}	0.01–0.05 $\frac{\text{g}}{\text{cm}^2\text{s}}$	0.01–0.05 $\frac{\text{g}}{\text{cm}^2\text{s}}$
α_{small}	0.3	0.6–0.75
α_{large}	0.3	0.2–0.4

Table 1: Parameters used during relaxation and simulation.

	Stretch	Scarf	Legwarmer
h	1/11800 s	1/22500 s	1/12000 s
Avg # segs/knit loop	8	8	8
# of spline segs	11264	26240	35200
# collision quadrature pts per seg	20	10	10
Avg time per frame	6.8 min	10.7 min	10.8 min
Yarn Collisions	58%	57%	52%
Other energy	7%	12%	23%
Constraints	7%	9 %	19%
Velocity filters	28%	22 %	6%

Table 2: Scene statistics.

The first stage of computation, before we begin simulating motion, is relaxing the models from their initial configuration to a rest state. In this process the models, which initially differ only in the orientation of the loops, take on the characteristic shapes and textures associated with these patterns in real knits. The relaxed models are shown in Figure 1, and the small-scale structures of the three models before and after relaxation are shown in Figure 7.

All of the real samples were knitted (by the first author) using wool worsted size 8 yarn, with each row knitted using alternating colors so that the knit structure is more readily apparent. The weight and diameter of this yarn was used as input parameters to our yarn model. Each sample consists of 42 rows, with each row containing 32 stitches. Figure 1 illustrates the results of using our model to relax an initial configuration into a default rest state for the three samples, while Figure 3 shows their real-life equivalents. Other than the placement of knit and purl stitches according to the model's knit pattern, all of the other parameters for the three models were identical. Our yarn-based model accurately predicts the curling on the edges of the stockinette, as well as the compression of the rib knit in the course direction and the garter knit in the wale direction. These properties arose naturally from the interactions of the yarn in our model; in comparison, to achieve the same effect in an elastic model would require careful manual tweaking of rest angles customized for each particular knit.

Figure 8 shows a three-way comparison of the stretching behavior of our sample knits along various directions. We compare the measured results to the output of our yarn-based model and an elastic sheet model [Baraff and Witkin 1998]. For all tests, one end of the cloth was held fixed while the other end was clamped and moved. In

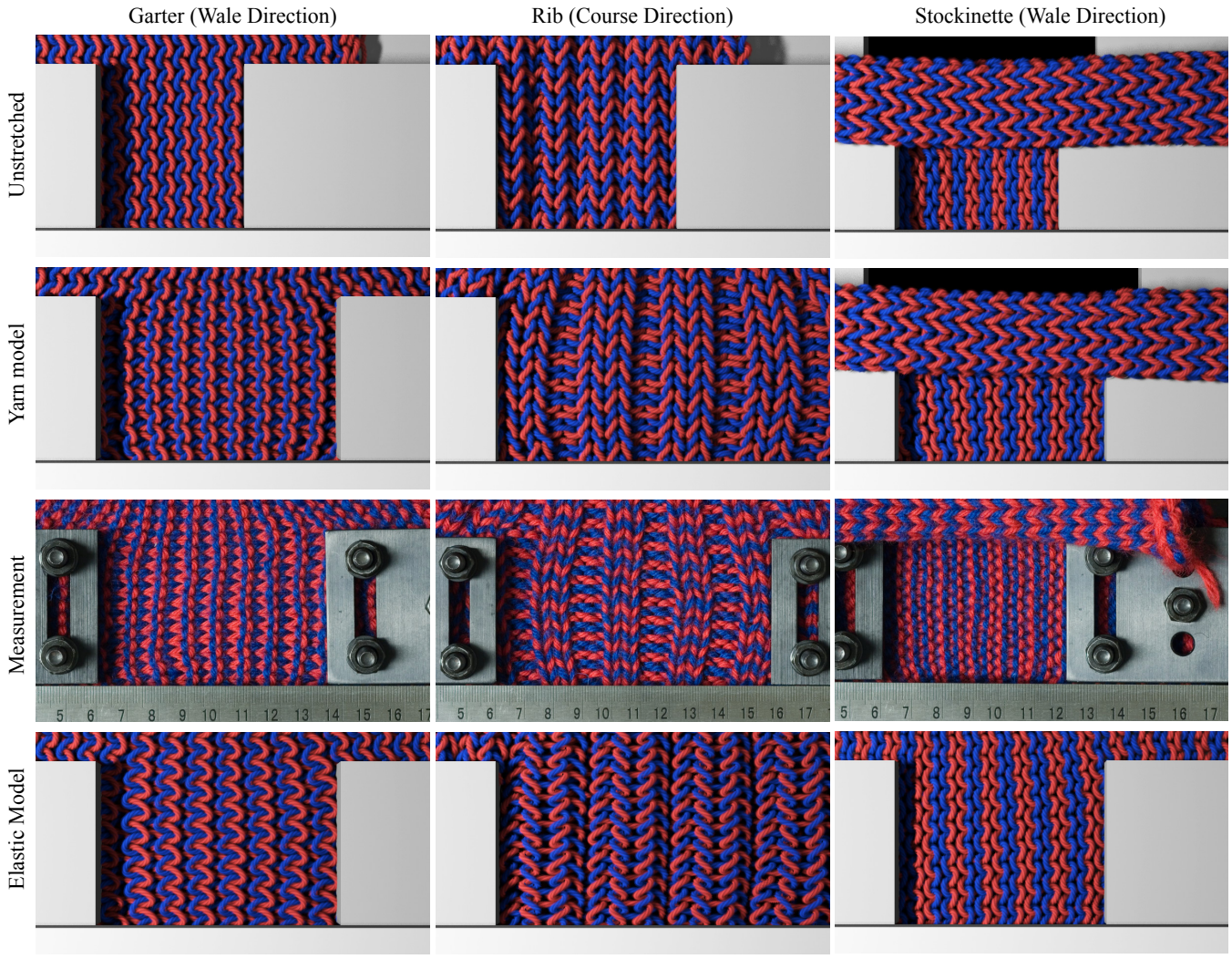


Figure 8: Stretch-test comparisons *for various knit types.*

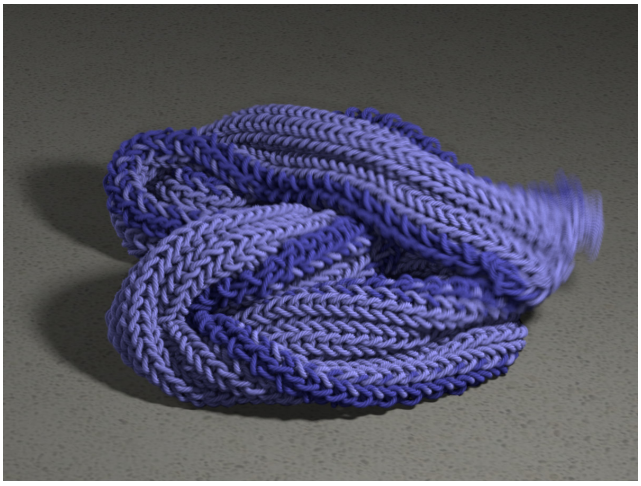


Figure 9: Scarf: *Our contact model scales to support the complex contact and folding that occurs in cloth.*

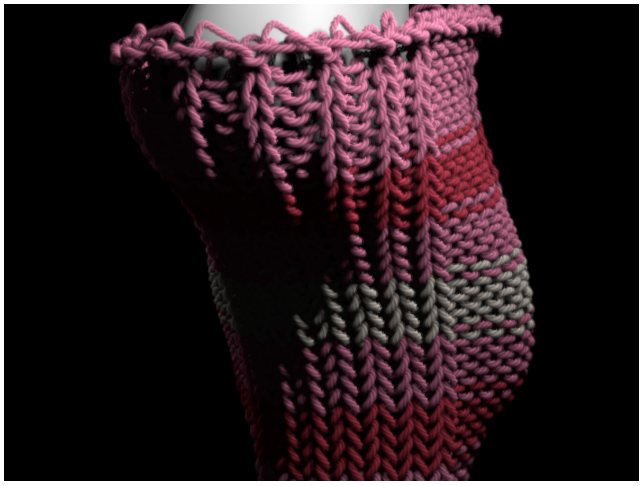


Figure 10: Leg warmer: *Simulating at the yarn level captures the nonlinear stretching of the knit ribs around the heel of the foot.*

order to generate the yarn geometry from the elastic sheet mesh, we project the control points of our knitted cloth while at rest into the mesh, determining for each point its barycentric coordinates with respect to the nearest triangle. We then use those to deform the control points when the mesh is deformed.

Our model predicts the characteristic shape of the knit while being stretched, in particular the tightening of the yarn loops in the garter, the separation of the ridges in the rib, and the rapid curling of the ends of the stockinette. The elastic model, due to its assumptions of infinitesimal continuity, predicts an unrealistic and inaccurate shape for the garter and the rib as the entire cloth stretches instead of the yarns deforming. For the stockinette, it does a reasonable job deforming the yarn structure; however, it fails to curl at the ends, which happens in both the real sample and our yarn model. Our model is in fact overeager to curl compared to the sample, although we believe this is due to the lack of an appropriate friction model. As a supplement to this paper, we provide the samples for the other 6 tests (in total, 3 samples and 3 directions each) for both the real sample and our model. We note that this is a rather strenuous test resulting in stiff but stable contacts, with some of the final states depending largely on frictional forces, which we do not model; as a result, some of the configurations reached are unstable and tend to rapidly shift to lower energy ones. However, despite this, we still capture the overall deformation of the yarns in the cloth.

Figure 9 shows the robustness of our collision model when applied to a 20×160 knitted scarf falling onto a plane. Our model is able to resolve the collisions resulting from contact with both the plane and itself. The average time per frame was about 10.7 minutes due to the small timestep used, which is comparable to the rendering time of about 9 minutes per frame for video-quality renderings.

Figure 10 shows a 44×96 knitted leg warmer being pulled over a foot. Because we simulate the yarn contacts directly, we are able to resolve the complicated stretching pattern as it slides over the heel. Due to the size of the model, there are over 100 billion pairs of quadrature points that potentially need to be evaluated for the collision integral at each step. Using our bounding box hierarchy, however, we are able to quickly find the 3.7 million pairs on average that are in contact, using only 12 million bounding box traversals and 12 million sphere-sphere evaluations on average.

Finally, Figure 11 shows our largest example, a 20×400 scarf composed of 64,690 spline segments falling on an inclined plane. The full movie of the sequence appears in the SIGGRAPH 2008 Computer Animation Festival [Kaldor et al. 2008].

Figure 11: A longer scarf on an inclined plane *"bunches up" as it falls and slides down.*

8 Conclusion

We have demonstrated a robust and scalable technique for simulating knitted cloth at the yarn level. Our simulation approach allows for significant increases in yarn-level knitted cloth complexity over previous research, while achieving practical offline simulation rates. In addition, we qualitatively verified our simulation both against knitted samples and against the predictions of a standard elastic cloth model, showing that there are interesting and visually noticeable nonlinear effects occurring in knits that are captured by our model but not by the elastic sheet approximation. In particular, our model is able to capture the salient mechanical features of garter, stockinette, and rib knits at rest without any parameter tuning or special cases–it follows directly from yarn interactions.

We anticipate that this work will be especially valuable to the textile community, particularly in the rapid design of clothing by allowing designers to see how knitted materials will drape and react without having to actually create the material. It is also applicable anywhere visual accuracy is of the utmost importance, such as for large, loose knits in computer animation, where individual yarns are visible in the frame and incorrect motion may be visually distracting. Finally, we think that models such as ours can provide a computational ground truth for future model comparisons. Approximations to yarn-level and/or continuous models can be compared against the output of a yarn-level simulator to see where and how they differ, and whether those differences are visually noticeable.

For future work, we see a wide variety of opportunities available for extending and building upon this research. Although our model stably handles yarns that are in constant, low-stiffness contact, as well as transient stiff contact between two colliding yarns, it is not as stable in handling constant, high-stiffness contact, such as those caused by stretching a knit excessively. In addition, our model does not treat friction, which is a critical component of yarn-yarn interactions and a driving factor in cloth hysteresis. Handling friction in yarn-level cloth is exceedingly difficult due to the large numbers of interrelated and distributed contacts. Beyond that, in addition to the qualitative evaluation performed here, we plan to do quantitative comparisons against real samples. Finally, as noted above, we believe that yarn-level cloth computations provide a starting point for the comparison of approximate models of knitted cloth to reality, and we plan to explore using this model to validate faster approximations of knit behavior.

Acknowledgments: This research was supported in part by funding from the following: the National Science Foundation (CCF-0702490), the NSF CAREER program (CCF-0347303, CCF-0652597), two Alfred P. Sloan Research Fellowships, and additional support from Intel, The Boeing Company, Pixar, Autodesk and NVIDIA.

References

BARAFF, D., AND WITKIN, A. 1998. Large steps in cloth simulation. In *Proc. SIGGRAPH '98*, ACM Press / ACM SIGGRAPH, 43–54.

BARAFF, D., WITKIN, A., AND KASS, M. 2003. Untangling cloth. *ACM Trans. Graph. 22*, 3, 862–870.

BERTAILS, F., AUDOLY, B., CANI, M.-P., QUERLEUX, B., LEROY, F., AND LÉVÊQUE, J.-L. 2006. Super-helices for pre-

dicting the dynamics of natural hair. *ACM Trans. Graph. 25*, 3 (August), 1180–1187.

BHAT, K., TWIGG, C., HODGINS, J., KHOSLA, P., POPOVIC, Z., AND SEITZ, S. 2003. Estimating cloth simulation parameters from video. In *Proc. SCA '03*, Eurographics Association, 37–51.

BREEN, D., HOUSE, D., AND WOZN, M. 1994. A particle-based model for simulating the draping behavior of woven cloth. *Textile Research Journal 64*, 11 (November), 663–685.

BRIDSON, R., FEDKIW, R., AND JOHN ANDERSON. 2002. Robust treatment of collisions, contact and friction for cloth animation. In *Proc. SIGGRAPH '02*, ACM Press / ACM SIGGRAPH, 594–603.

BRIDSON, R., MARINO, S., AND FEDKIW, R. 2003. Simulation of clothing with folds and wrinkles. In *Proc. SCA '03*, Eurographics Association, vol. 32, 28–36.

CHEN, Y., LIN, S., ZHONG, H., XU, Y.-Q., GUO, B., AND SHUM, H.-Y. 2003. Realistic rendering and animation of knitwear. *IEEE Transactions on Visualizations and Computer Graphics 9*, 43–55.

CHOI, K., AND KO, H. 2002. Stable but responsive cloth. In *Proc. SIGGRAPH '02*, ACM Press / ACM SIGGRAPH, 604–611.

CHOI, K., AND LO, T. 2003. An energy model of plain knitted fabric. *Textile Research Journal 73*, 739–748.

CHOI, K., AND TANDON, S. 2006. An energy model of yarn bending. *Journal of the Textile Institute 97*, 49–56.

CHU, L. 2005. *A Framework for Extracting Cloth Descriptors from the Underlying yarn Structure*. PhD thesis, University of California, Berkeley.

DEMIROZ, A., AND DIAS, T. 2000. A study of the graphical representation of plain-knitted structures part I: Stitch model for the graphical representation of plain-knitted structures. *Journal of the Textile Institute 91*, 463–480.

EBERHARDT, B., WEBER, A., AND STRASSER, W. 1996. A fast, flexible, particle-system model for cloth draping. *IEEE Computer Graphics and Applications 16*, 5, 52–59.

EBERHARDT, B., MEISSNER, M., AND STRASSER, W. 2000. Knit fabrics. In *Cloth Modeling and Animation*, D. House and D. Breen, Eds. A K Peters, ch. 5, 123–144.

GÖKTEPE, O., AND HARLOCK, S. C. 2002. Three-dimensional computer modeling of warp knitted structures. *Textile Research Journal 72*, 266–272.

GOLDENTHAL, R., HARMON, D., FATTAL, R., BERCOVIER, M., AND GRINSPUN, E. 2007. Efficient simulation of inextensible cloth. In *Proc. SIGGRAPH '07*, vol. 26.

GOLDSTEIN, H., POOLE, C., AND JOHN SAFKO. 2002. *Classical Mechanics*, 3rd ed. Addison Wesley.

GRINSPUN, E., HIRANI, A., DESBRUN, M., AND SCHRÖDER, P. 2003. Discrete shells. In *Proc. SCA '03*, Eurographics Association, 62–67.

JIANG, Y., AND CHEN, X. 2005. Geometric and algebraic algorithms for modelling yarn in woven fabrics. *Journal of the Textile Institute 96*, 237–245.

JOJIC, N., AND HUANG, T. 1997. Estimating cloth draping parameters from range data. In *Proc. Intl Workshop on Synthetic-Natural Hybrid Coding and Three Dimensional Imaging*, 73–76.

KALDOR, J., JAMES, D., AND MARSCHNER, S., 2008. Simulating cloth at the yarn level. Accepted to SIGGRAPH 2008 Computer Animation Festival, August.

KAWABATA, S., NIWA, M., AND KAWAI, H. 1973. The finite deformation theory of plain-weave fabrics part I: The biaxial-deformation theory. *Journal of the Textile Institute 64*, 21–46.

KING, M., JEARANAISILAWONG, P., AND SCORATE, S. 2005. A continuum constitutive model for the mechanical behavior of woven fabrics. *International Journal of Solids and Structures 42*, 3867–3896.

MÜLLER, M., HEIDELBERGER, B., HENNIX, M., AND RATCLIFF, J. 2006. Position based dynamics. In *Proc. Virtual Reality Interactions and Physical Simulations (VRIPhys)*, Eurographics, 71–80.

NADLER, B., PAPADOPOULOS, P., AND STEIGMANN, D. J. 2006. Multiscale constitutive modeling and numerical simulation of fabric material. *International Journal of Solids and Structures 43*, 206–221.

NOCENT, O., NOURRIT, J.-M., AND REMION, Y. 2001. Towards mechanical level of detail for knitwear simulation. In *WSCG 2001 Conference Proceedings*, V. Skala, Ed.

PAI, D. 2002. STRANDS: Interactive simulation of thin solids using Cosserat models. In *Proc. Eurographics*, vol. 21, 347–352.

PARK, J.-W., AND OH, A.-G. 2003. Bending mechanics of ply yarns. *Textile Research Journal 73*, 473–479.

PARK, J.-W., AND OH, A.-G. 2006. Bending rigidity of yarns. *Textile Research Journal 76*, 478–485.

PEIRCE, F. 1937. The geometry of cloth structure. *Journal of the Textile Institute 28*, T45–T97.

PROVOT, X. 1995. Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *Proc. Graphics Interface '95*, Canadian Human-Computer Communications Society, W. A. Davis and P. Prusinkiewicz, Eds., 147–154.

RÉMION, Y., NOURRIT, J.-M., AND GILLARD, D. 1999. Dynamic animation of spline like objects. In *Proc. WSCG'99*, V. Skala, Ed.

RIVERS, A. R., AND JAMES, D. L. 2007. FastLSM: Fast lattice shape matching for robust real-time deformation. *ACM Trans. Graph. 26*, 3, 82.

SPILLMANN, J., AND TESCHNER, M. 2008. An adaptive contact model for the robust simulation of knots. In *Proc. Eurographics 2008*, vol. 27, 497–506.

TERZOPOULOS, D., PLATT, J., BARR, A., AND FLEISCHER, K. 1987. Elastically deformable models. *Computer Graphics 21*, 205–214.

THEETTEN, A., GRISONI, L., DURIEZ, C., AND MERLHIOT, X. 2007. Quasi-dynamic splines. In *Proc. ACM Symposium on Solid and Physical Modeling '07*.

VOLINO, P., AND THALMANN, N. M. 2000. Implementing fast cloth simulation with collision response. In *Proc. Computer Graphics International*, 257–266.

WARREN, W. 1990. The elastic properties of woven polymeric fabric. *Polymer Engineering and Science 30*, 1309–1313.

ZENG, X., TAN, V. B. C., AND SHIN, V. P. W. 2006. Modelling inter-yarn friction in woven fabric armor. *International Journal for Numerical Methods in Engineering 66*, 1309–1330.