Inequality Cloth

Ning Jin
Stanford University
njin19@stanford.edu

Wenlong Lu
Stanford University
wenlongl@stanford.edu

Zhenglin Geng
Stanford University
zhenglin@stanford.edu

Ronald P. Fedkiw
Stanford University, Industrial Light & Magic
rfedkiw@stanford.edu

ABSTRACT

As has been noted and discussed by various authors, numerical simulations of deformable bodies often adversely suffer from so-called "locking" artifacts. We illustrate that the "locking" of out-of-plane bending motion that results from even an edge-spring-only cloth simulation can be quite severe, noting that the typical remedy of softening the elastic model leads to an unwanted rubbery look. We demonstrate that this "locking" is due to the well-accepted notion that edge springs in the cloth mesh should preserve their lengths, and instead propose an inequality constraint that stops edges from stretching while allowing for edge compression as a surrogate for bending. Notably, this also allows for the capturing of bending modes at scales smaller than those which could typically be represented by the mesh. Various authors have recently begun to explore optimization frameworks for deformable body simulation, which is particularly germane to our inequality cloth framework. After exploring such approaches, we choose a particular approach and illustrate its feasibility in a number of scenarios including contact, collision, and self-collision. Our results demonstrate the efficacy of the inequality approach when it comes to folding, bending, and wrinkling, especially on coarser meshes, thus opening up a plethora of interesting possibilities.

CCS CONCEPTS

• **Computing methodologies → Animation**; **Physical simulation**;

KEYWORDS

cloth, inequality, folds, wrinkles, optimization

ACM Reference format:
Ning Jin, Wenlong Lu, Zhenglin Geng, and Ronald P. Fedkiw. 2017. Inequality Cloth. In *Proceedings of SCA '17, Los Angeles, CA, USA, July 28-30, 2017,* 10 pages.
https://doi.org/10.1145/3099564.3099568

SCA '17, July 28-30, 2017, Los Angeles, CA, USA

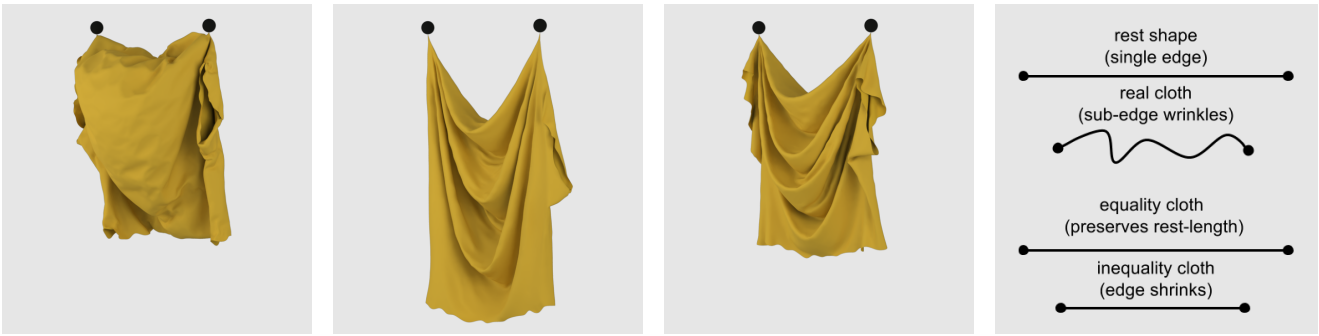
ACM ISBN 978-1-4503-5091-4/17/07…$15.00
https://doi.org/10.1145/3099564.3099568

1 INTRODUCTION

Cloth simulation is prevalent in the field of computer graphics and has received widespread attention from a large cross-section of researchers, see for example [Baraff and Witkin 1998; Baraff et al. 2003; Breen et al. 1994; Bridson et al. 2002; Provot 1995, 1997; Terzopoulos and Fleischer 1988; Terzopoulos et al. 1987; Volino et al. 1995]. As computational resources increased over the years, researchers were able to simulate more and more detailed cloth meshes, see for example [Selle et al. 2009] which simulates over 2 million triangles. Recently, with the increased interest in simulations on the GPU, interactivity, virtual and augmented reality, and video games, researchers have regained quite some interest in simulating coarser cloth meshes once again, see for example [Gillette et al. 2015; Kavan et al. 2011; Koh et al. 2014; Liu et al. 2013; Müller and Chentanez 2010; Müller et al. 2007].

The return of cloth simulation on coarser meshes brings with it the inherent difficulties associated with representing continuous nonlinear phenomena by a small discrete number of degrees of freedom. One of the main difficulties is associated with the general phenomenon of "locking," where a mesh that lacks sufficient degrees of freedom behaves more rigidly than desired. Indeed, "locking" problems are not specific to cloth and are common in many simulation setups involving mass-spring systems, finite elements, grid-based fluids, etc., see for example [Ando et al. 2013; Irving et al. 2007; Misztal et al. 2010; Patterson et al. 2012; Twigg and Kačić-Alesić 2011], and in the modeling of any developable surfaces or thin shells, see for example [Bucalem and Bathe 1997; English and Bridson 2008; Narain et al. 2013; Solomon et al. 2012; Wang et al. 1999]. It is worth noting that "locking" can typically be alleviated by increasing mesh resolution and thereby adding more degrees of freedom; for instance [Koh et al. 2014, 2015; Narain et al. 2012] adaptively remeshes the cloth in order to capture more details in some regions. However, the extra expense may not be affordable in many applications.

[Choi and Ko 2002] discussed these kinds of issues, focusing on the implications of real and numerical/fictitious damping on the so-called buckling instability. By now, it is well-accepted that a numerical approach can lead to unwanted "locking" and stiffness when it comes to bending. However, we show that cloth meshes can "lock" and subsequently fail to bend even with non-dissipative fully explicit time integration and even when only edge springs (and no bending springs) are used. Furthermore, we illustrate that a mesh can lock into a stationary improperly unfolded position with zero contribution from damping, see e.g. Figures 3b and 3c. In fact, as discussed in Section 2, one can trivially see via side-side-side



(a) *stiff edge springs (not folding)* **(b)** *soft edge springs (rubbery stretching)* **(c)** *inequality cloth* **(d)** *compression "captures" bending modes*

Figure 1: *In cloth simulation, if each edge of the triangle mesh is constrained to preserve its length, then the resulting articulated rigid structure would have difficulty bending. Unfortunately, a large edge spring stiffness (a) is typically required to stop cloth from stretching too much and appearing rubbery (b), meaning that the cloth spuriously "locks" and cannot bend even when using only edge springs (a). Using inequality constraints on cloth edges (c) restricts the cloth from stretching and appearing rubbery as in (b), while allowing for edge compression properly captures bending modes, preventing the "locking" seen in (a). Furthermore, compressed edges provide the best surrogate for bending, folding, and wrinkling at scales smaller than the mesh edges (d).*

congruence that an edge equality constraint enforces rigidity of each triangle, and thus certain cloth mesh topologies can be seen to spuriously "lock," losing bending modes, based only on edge spring stiffness.

We avoid the "locking" of bending modes by changing the basic underlying formulation and equations governing our cloth's degrees of freedom. Instead of targeting an equality constraint on each edge (or element) of the mesh and enforcing this constraint via a penalty-based spring approach where the spring stiffness has to be tuned between stiffer values which "lock" bending modes and weaker values which admit a soft rubbery behavior, we propose an inequality constraint that simply prevents edges (or elements) from inappropriately stretching while allowing for edge (or element) compression as a surrogate model for bending modes that would not otherwise be representable by the mesh. Remarkably, it has also been observed in some different contexts that lowering compression stiffness can be quite beneficial. For instance, [Müller and Chentanez 2010] noted that weakening compression resistance in their coarse simulation mesh was important for recovering wrinkles on their finer wrinkle mesh. Similarly, [Koh et al. 2015] observed that reducing compressive strain helped avoid undesirable buckling. Interestingly, [Conti and Maggi 2008] also showed the link between compression relaxation and the formation of fine-scale oscillations in the context of folding paper. However, we additionally provide formal arguments that demonstrate the flaws of equality-based cloth and propose a new model based on inequality constraints that allows us to transform the problem into an optimization-based formulation. While enforcing our inequality constraints may appear somewhat similar to strain limiting [Bridson et al. 2002; Goldenthal et al. 2007; Provot 1995; Thomaszewski et al. 2009; Wang et al. 2010b], important differences remain in the model. Moreover, normally applying strain limiting to restore rest-lengths can lead to "locking." Notably, though [Kim et al. 2012] also employed one-sided constraints, their goal is to enforce distance constraints from clothing to character body.

We discuss a number of optimization-based approaches to our *inequality cloth* in Section 4, noting that there is obvious serendipitous leverage that can be gleaned from the recent works addressing optimization-based frameworks for *equality cloth* and deformable bodies, see for example [Bouaziz et al. 2014; Fratarcangeli et al. 2016; Gast and Schroeder 2014; Goldenthal et al. 2007; Müller et al. 2007; Narain et al. 2016; Wang and Yang 2016]. The particular approach we use is presented in Section 5 along with various implementation details, and its efficacy is demonstrated on a number of examples including bending, folding, and wrinkling as well as contact, collision, and self-collision. The simulation results entail a high degree of optimism with regard to future research in this direction.

2 MOTIVATION

Consider a square cloth mesh cut into N^2 sub-squares where each is triangulated by a diagonal producing $2N^2$ triangles. There are $(N+1)^2$ vertices and $3(N+1)^2$ degrees of freedom. Applying a length constraint on each of the $2N(N+1)+N^2$ edges reduces the degrees of freedom to $4N+3$. Although this already seems problematic since there is not even one degree of freedom for each of the $(N+1)^2$ vertices, it turns out that the situation can be worse.

Consider Figure 2a where the 12 degrees of freedom reduce to 6 when vertices 1 and 4 are constrained to not move. Constraining edges a, b, c, and d to each preserve their length removes 4 more

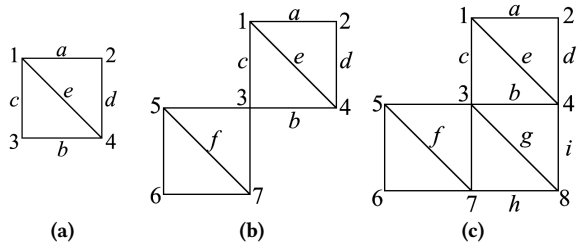
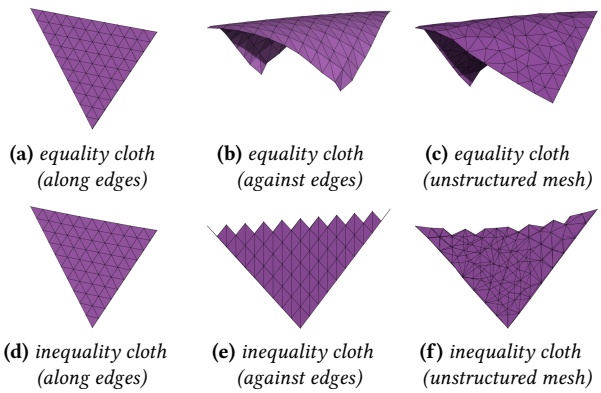


Figure 2: *The simple square cloth meshes above can easily "lock" when two diagonal vertices (e.g., 2 & 3 or 2 & 6) are held fixed.*

degrees of freedom, and vertices 2 and 3 each retain only 1 degree of freedom as they rotate around edge e while each triangle preserves its shape due to side-side-side congruence. Alternatively, constraining vertices 2 and 3 allows edge e to remove an additional degree of freedom, leaving only 1 degree of freedom corresponding to the square rotating rigidly in a circle about the line segment connecting vertices 2 and 3. Here, everything works as expected based on the constraints and the degrees of freedom. It turns out that the same is true for a large cloth mesh with the corresponding corners fixed, easily bending when the fixed diagonal is along the edges (Figure 3a), but struggling to bend when the fixed diagonal is against the edges (Figure 3b).

Consider Figure 2b where we constrain vertices 2 and 6, and the other 5 vertices suggest a potential 15 degrees of freedom which drop to 5 when one considers the 10 edge constraints. However, it turns out that the standard process of counting of constraints and degrees of freedom is overly optimistic. From side-side-side, all 4 triangles remain congruent to their rest-shapes, and thus as triangle 1-2-4 rotates around vertex 2, triangle 1-3-4 can merely rotate around edge e. Thus, vertex 3 will never be more than $||e||$ away from vertex 2. Similarly, vertex 3 must lie within the sphere of radius $||f||$ from vertex 6. Since vertices 2, 3, and 6 are collinear, these two spheres are tangent at vertex 3, and thus vertex 3 cannot move. This removes 3 degrees of freedom, dropping the total number from 5 to 2 as each square independently rigidly rotates about diagonal 2-6. Furthermore, adding an edge between vertices 4 and 7 creates a new triangle constraining the mesh, so that it can only rigidly rotate around diagonal 2-6, i.e., it removes the independent rotations of the two squares. Similar arguments applied to more than 2 squares connected in this same configuration show that each square independently rotates only when there are no herring bone connections (i.e., 1-5 or 4-7) that remove the independent rotations.

Consider Figure 2c where the new triangles are added in a non-herring bone configuration. Bending along edge g allows vertices 4 and 7 to be closer together but never farther apart than $||g||$. However, any independent rotation of square 1-2-3-4 from square 3-5-6-7 make vertices 4 and 7 farther apart than $||g||$. Thus the addition of these two new triangles removes the independent rotation of the two original squares, constraining them to rotate together. Furthermore, edges g, h, and i then constrain away the 3 degrees of freedom of vertex 8, so all 6 triangles rigidly rotate about diagonal 2-6. In summary, the expected 5 degrees of freedom reduce to only 1, illustrating how overly optimistic the bad news in the first paragraph of this section was.



(a) *equality cloth (along edges)* **(b)** *equality cloth (against edges)* **(c)** *equality cloth (unstructured mesh)*

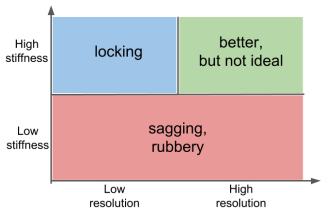
(d) *inequality cloth (along edges)* **(e)** *inequality cloth (against edges)* **(f)** *inequality cloth (unstructured mesh)*

Figure 3: *A piece of square cloth is hanging with two vertices fixed. If the cloth is hanging along the edges of the mesh, both the equality cloth (a) and the inequality cloth (d) fold properly. If the cloth is instead hanging against the edges of the mesh, then a stiff piece of equality cloth has trouble bending (b), whereas inequality cloth still folds properly (e) by compressing some of its edges. Similar results are obtained for unstructured mesh as shown in (c) and (f).*

(a) *equality cloth (low resolution)* **(b)** *inequality cloth (low resolution)* **(c)** *equality cloth (high resolution)* **(d)** *inequality cloth (high resolution)*

Figure 4: *The inequality cloth curtain (2.7K triangles) folds well (b), while the equality cloth curtain struggles (a). Increasing the resolution (11.1K triangles) helps the equality cloth curtain (c), but it still struggles comparing to its inequality counterpart (d). Note that one might prefer the look of the curtains in (a) and (c) to that in (b) and (d), but it is preferrable to obtain (a) and (c) by increasing bending resistance on (b) and (d) so that the user has control over the degree of bending. Furthermore, it is obviously undesirable to be stuck with only (a) and (c), unable to obtain (b) and (d) even with bending springs/resistance set to zero.*

Obviously, relaxing equality constraints on edges to a penalty-based spring formulation does open up the ability for the mesh to bend, but the stiffer the springs, the more "locking" one would expect. Figures 3a and 3b show how stiff edge springs (with no explicit bending springs/resistance) allow cloth to bend when constrained along the edges, but incorrectly "lock" when constrained against the edges. Figure 3c shows that this poor behavior is not limited to structured meshes, although the analysis applied to Figure 2 would obviously be more complicated in the case of an unstructured mesh.



In practical scenarios, we have observed that the more edge constraints are relaxed, the less "locking" occurs. However, this low stiffness in turn makes the mesh appear too rubbery (the red region in the accompanying figure). Meanwhile, increasing the stiffness inappropriately creates "locking" (blue region), which appears to be ameliorated to some degree by higher

Figure 5: *Equality cloth (top) and inequality cloth (bottom) both with 10K triangles for a dancing sequence with a skirt. Note that the equality cloth not only has significantly less detail but also suffers from the "locking" phenomenon when the cloth is clustered closely together and appears crumbly as in columns 1 and 3. Note the much more natural and smooth shapes obtained by the inequality cloth in those columns.*

mesh resolution (green region), as illustrated in Figure 4. In summary, one is left tuning parameters to try to avoid poor behavior ("locking" or rubberiness) especially on lower resolution meshes.

The benefits of inequality cloth are easy to see even in Figure 2a. Although none of the edges are allowed to increase beyond their rest-lengths, they are allowed to decrease, and so vertices 1 and 4 retain many degrees of freedom. For example, they can move closer together and downward as a surrogate for bending about diagonal 2-3. Figure 1d also illustrates how inequality cloth uses compression to capture bending. Here a single edge wrinkles at a scale too small to be tracked by a single straight line segment. Whereas equality cloth simply remains at its rest-length, inequality cloth can at least compress to the correct Euclidean distance even though it misses the geodesic detail. This is similar in spirit to the way shocks are captured in compressible flow, correctly modeling the pre-shock and post-shock states while ignoring the sub-shock details that the grid is not fine enough to resolve, see e.g. [LeVeque 2002].

3 INEQUALITY CLOTH

For each edge with rest-length $\bar{l}$, we define the maximum allowable length as $l^{\max} = (1+\alpha)\bar{l}$ where $\alpha > 0$ is chosen by the user. We then write our inequality constraints as

$$\frac{l_i(\mathbf{x})}{\bar{l}_i} - 1 \leq \alpha, \quad i = 1, \ldots, m, \tag{1}$$

where $\mathbf{x}$ are the particle positions and $l_i(\mathbf{x})$ is the current length of an edge. Here, the left hand side of Equation 1 is chosen to resemble a typical non-dimensionalized strain measure, which is quite often used for springs, see e.g. [Selle et al. 2009]. This can be rewritten in the form

$$c_i(\mathbf{x}) = \frac{l_i(\mathbf{x})}{l_i^{\max}} - 1 \leq 0, \quad i = 1, \ldots, m, \tag{2}$$

which can be written more succinctly as $\mathbf{c}(\mathbf{x}) \leq \mathbf{0}$.

We approach the resulting constrained dynamics problem via Lagrange multipliers in maximal coordinates, since the inequality constraints pose inherent difficulties for reduced/generalized coordinates. See for example [Baraff 1996; Goldstein 1950]. Thus, we differentiate our constraints to obtain a Jacobian matrix $\mathbf{J}(\mathbf{x}) = \partial \mathbf{c}(\mathbf{x})/\partial \mathbf{x}$ of size $m \times 3n$, where n is the number of particles, noting that the constraint force then has the form $-\mathbf{J}^T(\mathbf{x})\boldsymbol{\lambda}$ where $\boldsymbol{\lambda}$ is a vector of m scalar Lagrange multipliers. Notably, when $c_i(\mathbf{x}) < 0$, the constraint should be inactive with $\lambda_i = 0$, and when $c_i(\mathbf{x}) = 0$, the constraint should be active with $\lambda_i \geq 0$. See [Gill et al. 1981]. Thus, we solve the following set of equations for our inequality cloth

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}, \tag{3a}$$

$$\mathbf{M}\frac{d\mathbf{v}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{v}) - \mathbf{J}^T(\mathbf{x})\boldsymbol{\lambda}, \tag{3b}$$

$$\mathbf{c}(\mathbf{x}) \leq \mathbf{0}, \tag{3c}$$

$$\lambda_i = 0 \quad \text{when} \quad c_i(\mathbf{x}) < 0, \quad i = 1, \ldots, m, \tag{3d}$$

$$\lambda_i \geq 0 \quad \text{when} \quad c_i(\mathbf{x}) = 0, \quad i = 1, \ldots, m, \tag{3e}$$

where $\mathbf{v}$ is the velocity vector, $\mathbf{M}$ is the mass matrix, and $\mathbf{f}(\mathbf{x}, \mathbf{v})$ includes all non-constraint forces. Basically, if any constraint $c_i(\mathbf{x})$

Figure 6: *Left five figures show an equality cloth T-shirt (7.5K triangles) with increasing edge spring stiffness and no bending springs. Low stiffness leads to exaggerated sagging and high stiffness leads to "locking." Second figure from the right shows the equality cloth T-shirt with optimal stiffness for edge springs augmented with weak bending springs to remove some noise artifacts. Right figure shows that our inequality cloth T-shirt does not suffer from sagging while preserving nice details and a natural shape.*

is about to be violated with $c_i(\mathbf{x}) = 0$, then the associated Lagrange multiplier λ_i applies a force fighting the violation of the constraint by scaling the column of $\mathbf{J}^T$ corresponding to $(\partial c_i(\mathbf{x})/\partial \mathbf{x})^T$; otherwise this column of $\mathbf{J}^T$ makes no contribution to the forces which proceed as usual.

There are a few issues one must consider regarding Equation 3. First, even though $-(\partial c_i(\mathbf{x})/\partial \mathbf{x})^T \lambda_i$ is in the correct direction to maintain $c_i(\mathbf{x}) \leq 0$ by cancelling all unwanted forces in $\mathbf{f}(\mathbf{x}, \mathbf{v})$ in Equation 3b, it also must cancel any part of $\mathbf{v}$ that would cause a change in $\mathbf{x}$ in Equation 3a that would violate the constraint. Thus, the application of $\mathbf{M}^{-1}$ to the right hand side of Equation 3b had better not skew the constraint forces, which is trivially accomplished with a diagonal mass matrix. Secondly, when one first reaches the constraint boundary with $c_i(\mathbf{x}) = 0$, $\mathbf{v}$ would typically have a finite non-zero value approaching the constraint manifold, meaning that the constraint would be violated instantaneously according to Equation 3a unless there is an infinite constraint force in Equation 3b that cancels this offending velocity in infinitesimal time. Thus, one would require a constraint impulse to instantaneously cancel any offending velocity (after which a finite constraint force would be used as usual). Moreover, one should use the smallest such impulse (and force) that works. Alternatively, in a discrete setting, we could integrate Equation 3b forward in time with a finite constraint force to achieve a velocity that could subsequently be used in Equation 3a. Although this makes a first-order error in time, conveniently, it allows us to use a constraint force instead of a constraint impulse, and Equation 3a takes on the usual backward Euler form of $\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t \mathbf{v}^{n+1}$.

To simplify our approach, we ignore the velocity-based forces in Equation 3b, addressing damping later in Section 5. We then discretize Equation 3b with backward Euler,

$$\mathbf{M}\frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} = \mathbf{f}(\mathbf{x}^{n+1}) - \mathbf{J}^T(\mathbf{x}^{n+1})\boldsymbol{\lambda}, \tag{4}$$

and rearrange to obtain

$$\mathbf{M}\Delta\mathbf{x} = \Delta t \mathbf{M}\mathbf{v}^n + \Delta t^2 \left[\mathbf{f}(\mathbf{x}^{n+1}) - \mathbf{J}^T(\mathbf{x}^{n+1})\boldsymbol{\lambda}\right], \tag{5}$$

where $\Delta\mathbf{x} = \mathbf{x}^{n+1} - \mathbf{x}^n$.

4 OPTIMIZATION FRAMEWORK

Following [Gast and Schroeder 2014], one can define a potential $\phi(\mathbf{x})$ corresponding to conservative forces (minus gravity) so that $\mathbf{f}(\mathbf{x}) = -\nabla\phi(\mathbf{x}) + \mathbf{M}\mathbf{g}$. The energy

$$E(\Delta\mathbf{x}) = \frac{1}{2}\Delta\mathbf{x}^T\mathbf{M}\Delta\mathbf{x} - \Delta t \Delta\mathbf{x}^T\mathbf{M}\mathbf{v}^n + \Delta t^2\phi(\mathbf{x}^{n+1}) - \Delta t^2 \Delta\mathbf{x}^T\mathbf{M}\mathbf{g} \tag{6}$$

then has critical points that satisfy Equation 5 in the absence of constraints (i.e., when $\boldsymbol{\lambda} = \mathbf{0}$). Thus, we solve the following constrained optimization problem

$$\text{minimize} \qquad E(\Delta\mathbf{x}) \tag{7a}$$

$$\text{subject to} \qquad \tilde{\mathbf{c}}(\Delta\mathbf{x}) \leq \mathbf{0}, \tag{7b}$$

where $\tilde{\mathbf{c}}(\Delta\mathbf{x}) \equiv \mathbf{c}(\Delta\mathbf{x} + \mathbf{x}^n) = \mathbf{c}(\mathbf{x}^{n+1})$, and similarly $\tilde{\mathbf{f}}(\Delta\mathbf{x}) \equiv \mathbf{f}(\mathbf{x}^{n+1})$ and $\tilde{\mathbf{J}}(\Delta\mathbf{x}) = \partial\tilde{\mathbf{c}}(\Delta\mathbf{x})/\partial\Delta\mathbf{x} = \mathbf{J}(\mathbf{x}^{n+1})$.

Solving this constrained optimization problem automatically enforces Equation 2 via Equation 7b. Under appropriate assumptions, the KKT conditions state that the solutions to Equation 7, i.e. $\Delta\mathbf{x}^*$ and $\boldsymbol{\lambda}^*$, also satisfy

$$\nabla E(\Delta\mathbf{x}^*) + \tilde{\mathbf{J}}^T(\Delta\mathbf{x}^*)\boldsymbol{\lambda}^* = \mathbf{0}, \tag{8}$$

which (after noting that $\boldsymbol{\lambda}^* = \Delta t^2\boldsymbol{\lambda}$) is equivalent to Equation 5, which in turn is our discrete version of Equations 3a and 3b. However, $\Delta\mathbf{x}^*$ and $\boldsymbol{\lambda}^*$ may not identically solve Equation 5 if one only approximates the solution to Equation 7 or various assumptions are not met. In those scenarios, it is safer to use $\Delta\mathbf{x}^*$ and $\boldsymbol{\lambda}^*$ on the right hand side of Equation 5 to approximate the forces, subsequently updating $\mathbf{x}^{n+1}$ on the left hand side of Equation 5 with this approximation to the backward Euler forces. See e.g. [Shinar et al. 2008] and the discussions and citations therein. The KKT conditions also assert that there is no constraint force when $\tilde{c}_i(\Delta\mathbf{x}^*) < 0$, satisfying Equation 3d, and that $\boldsymbol{\lambda}^* \geq \mathbf{0}$, satisfying Equation 3e. Unfortunately, the KKT conditions are subject to various assumptions on regularity and the nature of the solutions, and they are also necessary but not sufficient. Therefore, when solving Equation 7, one should take additional care to ensure that $\boldsymbol{\lambda}$ is chosen properly in regards to Equations 3d and 3e and that $\boldsymbol{\lambda}$ is as small as possible.

Interior point methods such as the barrier method can be straightforwardly applied to solve Equation 7. This turns the constrained optimization problem into an unconstrained optimization problem with the addition of a barrier potential term for the inequality constraints. A weighting factor is successively reduced as one follows the central path toward the optimal solution while always

maintaining feasibility. This transformed unconstrained optimization problem can then be solved by taking Newton steps with line search, where the resulting linear system is sparse and symmetric positive definite. Under this barrier method formulation, one can intuitively picture the inequality constraints as generating force fields which are scaled to be successively smaller as long as they still counterbalance the gradient of the unconstrained energy term. This mechanism provides a heuristic for choosing the smallest necessary constraint force to enforce the inequality constraints. See [Boyd and Vandenberghe 2004] for more details. A major issue with this approach is that an initially large barrier potential can drive the state far enough from a nearby local minimum such that the subsequent reduction of the strength of the barrier potential force incorrectly leads to a different local minimum. In our inequality cloth, this could result in the change of a wrinkled state or the removal of wrinkles altogether. Similar to the barrier method, the primal-dual method also uses Newton's method to find the search directions by solving a linear system that closely resembles the one for the barrier method, except that the system is now augmented with the Lagrange multipliers.

Instead of initially applying a strong barrier potential force and then slowly relaxing it away, risking perturbation into nearby local minima, we implemented a complementary approach where we first relax the constraints until they are satisfied and then slowly tighten the constraints activating constraint forces as necessary. That is, instead of initially over-tightening constraint forces and perturbing the current state of the cloth potentially towards incorrect local minima, we initially allow the cloth to remain stationary by loosening the constraints. Either way, the method starts in a so-called feasible configuration and moves towards a local minimum, but our approach favors more temporal consistency.

To accomplish this, we choose a sequence of scalar target constraint values starting from the currently worst violated constraint $c^{(0)}$ and ending with $c^{(k_{\max})}$ which should be zero to satisfy Equation 2. Then given iteration $(k-1)$ we proceed to iteration (k) as following. First, we identify the subset of the m constraints in Equation 2 which do not have $c_i(\mathbf{x}^{(k-1)}) < c^{(k)}$, and then compute a subset of the Jacobian matrix $\hat{\mathbf{J}}$ using only these constraints, evaluating $\hat{\mathbf{J}}$ using the current configuration, i.e., $\mathbf{x}^{(k-1)}$. The full accumulated constraint force $(\mathbf{J}^T\boldsymbol{\lambda})^{(k-1)}$ for all m constraints will have a subset of it corresponding to the active constraints updated using $\hat{\mathbf{J}}^T\hat{\boldsymbol{\lambda}}$. Focusing only on the constraint forces in Equation 5 gives $\mathbf{M}(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}) = -\Delta t^2\hat{\mathbf{J}}^T\hat{\boldsymbol{\lambda}}$, which allows us to compute $\mathbf{x}^{(k)}$ from $\hat{\boldsymbol{\lambda}}$ and complete the iteration. To compute $\hat{\boldsymbol{\lambda}}$, we assume a linearization $\Delta\hat{\mathbf{c}} = \hat{\mathbf{J}}\Delta\mathbf{x}$, allowing us to write

$$c^{(k)}\mathbf{1} - \hat{\mathbf{c}}(\mathbf{x}^{(k-1)}) = -\Delta t^2\left(\hat{\mathbf{J}}\mathbf{M}^{-1}\hat{\mathbf{J}}^T\right)\hat{\boldsymbol{\lambda}} \tag{9}$$

where $c^{(k)}\mathbf{1}$ is a column vector of scalar values $c^{(k)}$. Since $\hat{\mathbf{J}}\mathbf{M}^{-1}\hat{\mathbf{J}}^T$ is sparse and symmetric positive definite, the system can be efficiently solved by Krylov subspace solvers such as conjugate gradients. Noting that Equation 9 is only a guide, we use line search, safe sets, etc. as is typical. Equation 9 bears significant resemblance to the method presented in [Goldenthal et al. 2007], except of course for the repeated applications due to a changing $c^{(k)}$, the incremental additions of groups of particles to the active set, line search, etc. Similar related methods are discussed in [Bridson et al. 2002; Provot 1995; Wang et al. 2010b]; however, the equality-based approaches suffer from the "locking" artifacts as illustrated in Figure 7.

(a) *uni-directional constraints* **(b)** *two uni-directional constraints*

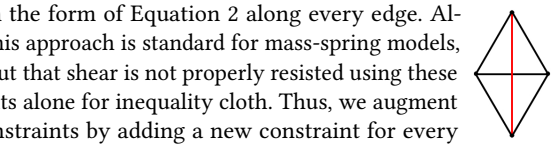
Figure 7: *Left: Inequality cloth using Equation 9. Right: Applying similar strain limiting to compression as well as expansion leads to "locking" artifacts where the cloth fails to fold when hanging diagonally on two corners.*

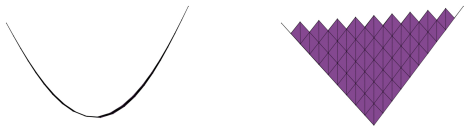
5 IMPLEMENTATION DETAILS

After testing a number of methods including those discussed in Section 4 and achieving various degrees of success, in the end we chose a single method that we felt would be most accessible for researchers and practitioners to incorporate into existing codebases, and subsequently ran all of the examples shown in the paper and the video using that method. There very well may be much more efficient and successful numerical approaches to our inequality cloth framework, but our focus here is more on the efficacy of any inequality cloth approach as compared to equality cloth. Our goal is to establish a framework for the mathematical formulation similar in spirit to and motivated by [Kajiya 1986]'s rendering equation.

5.1 Mass-Spring Model

We model the cloth with a triangular mesh using a constraint in the form of Equation 2 along every edge. Although this approach is standard for mass-spring models, it turns out that shear is not properly resisted using these constraints alone for inequality cloth. Thus, we augment these constraints by adding a new constraint for every pair of triangles in order to connect their non-shared vertices (shown in red in the figure). These cross-edge connections are often used to add bending resistance to an equality cloth mesh, but their utilization via Equation 2 does not resist bending but only resists shear as shown in Figure 8. When bending resistance is in fact desired, we simply add very weak springs to every edge that has a constraint, using the same spring stiffness on both edges and cross-edges. The compression of these springs provides a surrogate for bending as shown in Figure 9.





(a) *no "cross" springs/constraints* **(b)** *with "cross" springs/constraints*

Figure 8: *Left: Without "cross" springs/constraints, a square cloth can collapse due to missing shear resistance. Right: The addition of "cross" springs/constraints allows the cloth to behave as expected.*

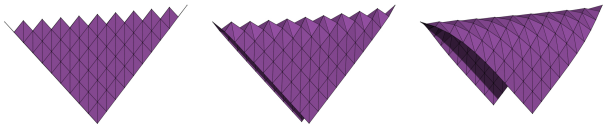


Figure 9: *The addition of weak springs allows inequality cloth to resist bending as desired. Unlike equality cloth which has some fixed minimal resistance to bending due to "locking," inequality cloth can obtain any degree of desired bending continuously approaching zero. (Stiffness increases from left to right.)*

5.2 Time Integration

Loosely speaking we use a Newmark style scheme, see e.g. [Bridson et al. 2003]:

(1) $\mathbf{v}^{*n+\frac{1}{2}} = \mathbf{v}^n + \frac{\Delta t}{2}\mathbf{M}^{-1}\mathbf{f}(\mathbf{x}^n, \mathbf{v}^{*n+\frac{1}{2}})$
(2) $\mathbf{x}^{*n+1} = \mathbf{x}^n + \Delta t \mathbf{v}^{*n+\frac{1}{2}}$
(3) Resolve collision with $\mathbf{x}^{*n+1}, \mathbf{v}^n$ to get $\hat{\mathbf{x}}^{n+1}, \hat{\mathbf{v}}^n$
(4) Resolve inequality constraints violation in $\hat{\mathbf{x}}^{n+1}, \hat{\mathbf{v}}^n$ to get $\tilde{\mathbf{v}}^n$
(5) $\tilde{\mathbf{v}}^{n+\frac{1}{2}} = \tilde{\mathbf{v}}^n + \frac{\Delta t}{2}\mathbf{M}^{-1}\mathbf{f}(\mathbf{x}^n, \mathbf{v}^{*n+\frac{1}{2}})$ by applying the same amount of forces as in step 1
(6) $\tilde{\mathbf{x}}^{n+1} = \mathbf{x}^n + \Delta t \tilde{\mathbf{v}}^{n+\frac{1}{2}}$
(7) Resolve contact with $\tilde{\mathbf{x}}^{n+1}, \tilde{\mathbf{v}}^{n+\frac{1}{2}}$ to get $\mathbf{x}^{n+1}$
(8) $\mathbf{v}^{n+1} = \tilde{\mathbf{v}}^n + \Delta t \mathbf{M}^{-1}\mathbf{f}(\mathbf{x}^{n+1}, \mathbf{v}^{n+1})$ with contact

Steps 1 and 2 use semi-implicit integration in order to obtain a position for use in collision detection and resolution in step 3. The resultant post-collision state will be constraint violating, and we correct for this using the method proposed in Section 4. However, these constraint satisfying positions are not retained; instead, we turn the full accumulated constraint force into an impulse which is used to augment the post-collision velocity at the end of step 4. Subsequently, this velocity is used to obtain the final position in steps 5 and 6, which is further processed for contact in step 7. Finally, the velocity is updated in step 8, including the effects of collisions and inequality constraints. Note that damping forces are included as usual in steps 1, 5, and 8.

5.3 Collisions

In order to enforce inequality constraints without causing collision violating interpenetrations with rigid objects, we include collision constraints as an additional set of inequality constraints when solving for our inequality constraint forces. Representing the collision bodies as level set surfaces (see [Bridson et al. 2003]) admits simple constraints in the form of $d(\mathbf{x}) \geq 0$. Importantly, the $d(\mathbf{x}) \geq 0$ constraints are scaled to be commensurate with Equation 2 so that they work nicely with our sequence of $c^{(k)}$ in Section 4. Since both $l_i(\mathbf{x})$ in Equation 2 and $d(\mathbf{x})$ are Euclidean distances, we simply divide d by the average rest-length of the edges in a mesh as a substitute for division by $l_i^{\max}$. The gradients can be computed exactly for analytic shapes and interpolated from a background grid for non-regular geometries and subsequently used to move vertices currently inside an object towards its surface, serving the same role as the right hand side of Equation 9 used in order to obtain the desired change in constraint violation given on the left hand side of that equation. See Figure 10. For self-collisions we use the method from [Bridson et al. 2002].

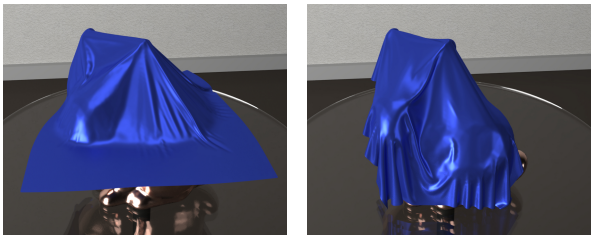


Figure 10: *A piece of inequality cloth (8.8K triangles) falls (left) and drapes (right) over a copper bunny model, illustrating that our collision constraints work well with complex geometries.*

6 EXAMPLES

For the sake of comparison with equality cloth, we chose a semi-implicit Newmark method, see e.g. [Bridson et al. 2002; Selle et al. 2009], since it has been shown to provide more visual details than the typical backward Euler approach, see e.g. [Baraff and Witkin 1998]; that is, our inequality cloth fares even better when compared with backward Euler integration. In the comparison examples with equality cloth, we first test a range of stiffness values with edge springs only and select the smallest stiffness that does not lead to exaggerated sagging. Then weak "cross" springs are added to provide the minimal bending resistance required to smooth away any sharp angles or noise. See Figure 6 for an illustration of this process. Even though we are quite experienced in cloth simulation and worked laboriously to tune parameters, we were unable to

Figure 11: *Top: An equality cloth skirt (10K triangles) looks rigid and heavy. Bottom: In contrast, our inequality cloth skirt (same mesh) looks lighter and produces more natural folds and wrinkles following the body's movement.*

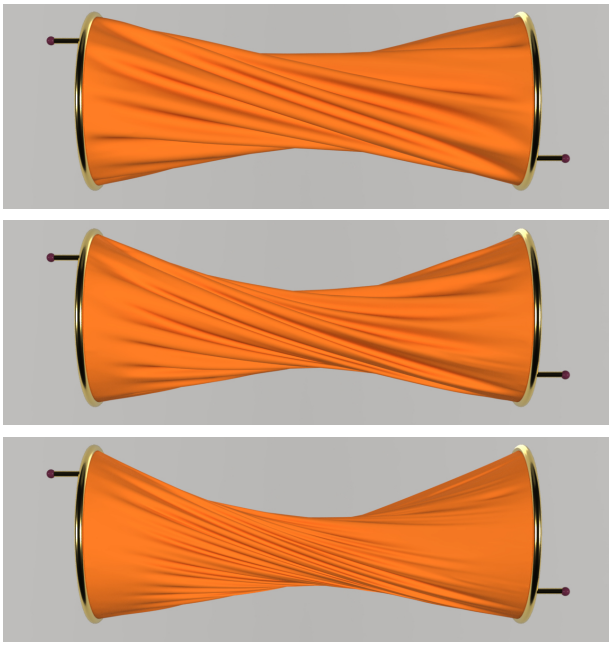
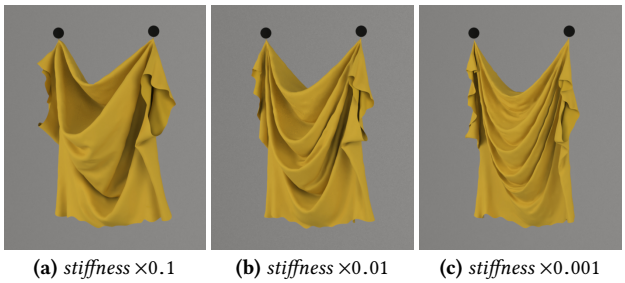


Figure 12: *Each cylinder-shaped cloth (12.4K triangles) is twisted around its axis by 81 degrees. They are all simulated with inequality cloth allowing a maximum of 5% stretching of each edge. Decreasing spring stiffnesses are used here to demonstrate the ability of our method to control bending resistance independently from the stretching constraints, leading to wrinkles at different frequencies.*

get equality cloth to achieve the same level of visual details that our first attempt at implementing inequality cloth gave, see for example Figure 11. For more examples of the visual details obtained by the inequality cloth, and comparisons to equality cloth, see Figure 5 and the video. In many examples, we render the simulation meshes. In a few of the examples, where we meant to show higher visual fidelity, and efficacy of our method, we use a small amount of subdivision illustrating that our method does not preclude this popular post-processing techniques. The runtime per rendering frame for our inequality cloth examples is roughly comparable to their best equality cloth counterparts.

Although α can be chosen individually for each constraint, we simply set a global value in all of our examples, and we typically recommend values in the range of 0.1% to 1% (larger values can



(a) *stiffness* $\times 0.1$ **(b)** *stiffness* $\times 0.01$ **(c)** *stiffness* $\times 0.001$

Figure 13: *Our inequality cloth armadillo "T-shirt" (14.7K triangles) exhibits different patterns depending on the desired stiffness.*

lead to undesirable stretchiness). To further illustrate the flexibility and artist-directability of our inequality cloth model demonstrated in Figure 9, we ran an armadillo "T-shirt" example in Figure 13 and a twist example in Figure 12 with varying spring stiffness to show that one can attain different patterns in the cloth and nicely control wrinkle frequency. In addition, we also performed quantitative analysis on our inequality cloth model. The histogram plots in Figure 14 show the edge strain of the equality cloth (edge springs only) and our inequality cloth (edge and "cross" springs/constraints) for the armadillo "T-shirt" final resting pose. Note that we pick typical (different) ranges of stiffness for both models respectively.

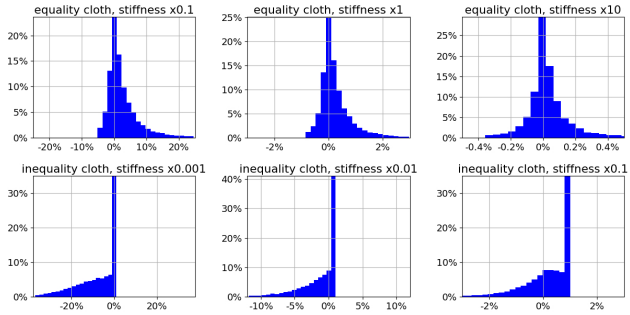


Figure 14: *Histograms of edge strain are computed for the final pose of a simulated armadillo "T-shirt" (shown in Figure 1 and Figure 13). The top row shows the result for equality cloth of increasing spring stiffness (*$\times 0.1, \times 1, \times 10$*), and the bottom row shows the result for inequality cloth of increasing spring stiffness (*$\times 0.001, \times 0.01, \times 0.1$*) with* $\alpha = 1\%$ *stretching allowed. The inequality cloth allows the armadillo "T-shirt" to enter a completely different configuration that does not "lock" or sag by letting the edges compress. One can also see that there are indeed a large number of edges in the inequality cloth (over 30%) stretched to the maximum.*

7 CONCLUSION AND FUTURE WORK

We introduce a new inequality cloth paradigm based on the observation that simulated cloth requires low compression stiffness in order to bend and fold properly and high expansion resistance to prevent sagging. We demonstrated the significant benefits of inequality cloth over equality cloth through a wide range of examples. Just as there are many ways to simulate cloth, for example [Baraff and Witkin 1998; Bridson et al. 2002, 2003; Choi and Ko 2002; Provot 1995; Volino et al. 1995] etc., there will undoubtably be many ways to simulate inequality cloth. Although our approach illustrates the efficacy and feasibility of the inequality cloth formulation, admittedly, there are bound to be superior approaches devised/proposed by us and others in the future.

In particular, the specific constraint-enforcing approach we used can introduce some noise/oscillation into the mesh, especially since it is a bit of a post-process to the other physical forces as opposed to being fully two-way implicitly coupled. Although axial bending springs or increased damping could alleviate these issues, these measures would also smooth away desirable details. Approaching

the inequality constraints with a force-based instead of impulse-based method could potentially improve the results, albeit this would require more careful examination of the active set and likely be more costly. Perhaps a bit surprisingly, collapsing the cloth to a single point is an allowed solution under our model, but if the path from the current state to that state requires constraints to be violated, then it does not happen; and in most practical scenarios, external physical forces, kinematic constraints, and collision constraints counteract any tendency to collapse.

Even though our inequality cloth framework maximizes the capability of a mesh to represent rich patterns, it obviously cannot capture sub-edge details. To approximate these missing details, one could employ high-order parametric models or learn models from real-world cloth. [Rohmer et al. 2010; Wang et al. 2010a] added nice higher resolution wrinkles to a coarser equality cloth mesh, but it is important to exercise caution so that one does not erroneously increase the geodesic length of the material by adding out-of-plane detail without compressing the planar representation to compensate. Interestingly, [Narita et al. 2016] proposed a solution to this problem where they compressed a two-dimensional flat image before using video projection to view it on a three-dimensional non-planar wrinkled surface in a manner that preserves geodesic distances for the original uncompressed image. Finally, another area for future work would be to investigate the potential benefits of applying an inequality cloth framework to woven/knitted cloth models, see [Cirio et al. 2014, 2017; Kaldor et al. 2008].

ACKNOWLEDGEMENTS

Research supported in part by ONR N00014-13-1-0346, ONR N00014-17-1-2174, ARL AHPCRC W911NF-07-0027 and NSF CNS1409847. Computing resources were provided in part by ONR N00014-05-1-0479. N. Jin. was supported by the Stanford Graduate Fellowship. The examples are rendered with RenderMan, and the authors thank Pixar for their generous donation of licenses. The authors would especially like to thank Matthew Cong for showing numerically that a square piece of cloth supported by 2 corners doesn't deform at all if the spring stiffness is too strong, even with explicit time stepping and no damping. The authors would also like to thank Professor S. Boyd and Nicholas Moehle for their valuable input on optimization.

REFERENCES

Ryoichi Ando, Nils Thürey, and Chris Wojtan. 2013. Highly Adaptive Liquid Simulations on Tetrahedral Meshes. *ACM Trans. Graph. (SIGGRAPH Proc.)* 32, 4 (2013), 103:1–103:10.

D. Baraff. 1996. Linear-time dynamics using Lagrange multipliers. In *Proc. of ACM SIGGRAPH 1996*. 137–146.

D. Baraff and A. Witkin. 1998. Large steps in cloth simulation. In *Proc. SIGGRAPH 1998*. 43–54.

D. Baraff, A. Witkin, and M. Kass. 2003. Untangling cloth. *ACM Trans. Graph. (SIGGRAPH Proc.)* 22 (2003), 862–870.

Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. 2014. Projective Dynamics: Fusing Constraint Projections for Fast Simulation. *ACM Trans. Graph.* 33, 4, Article 154 (July 2014), 11 pages.

Stephen Boyd and Lieven Vandenberghe. 2004. *Convex Optimization.* Cambridge University Press.

D. E. Breen, D. H. House, and M. J. Wozny. 1994. Predicting the drape of woven cloth using interacting particles. *Comput. Graph. (SIGGRAPH Proc.)* (1994), 365–372.

R. Bridson, R. Fedkiw, and J. Anderson. 2002. Robust Treatment of Collisions, Contact and Friction for Cloth Animation. *ACM Trans. Graph. (SIGGRAPH Proc.)* 21, 3 (2002), 594–603.

R. Bridson, S. Marino, and R. Fedkiw. 2003. Simulation of clothing with folds and wrinkles. In *Proc. of the 2003 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.* 28–36.

M. L. Bucalem and K. J. Bathe. 1997. Finite element analysis of shell structures. *Archives of Computational Methods in Engineering* 4, 1 (1997), 3–61.

K.-J. Choi and H.-S. Ko. 2002. Stable but responsive cloth. *ACM Trans. Graph. (SIGGRAPH Proc.)* 21 (2002), 604–611.

Gabriel Cirio, Jorge Lopez-Moreno, David Miraut, and Miguel A. Otaduy. 2014. Yarn-level Simulation of Woven Cloth. *ACM Trans. Graph.* 33, 6, Article 207 (Nov. 2014), 11 pages.

G. Cirio, J. Lopez-Moreno, and M. A. Otaduy. 2017. Yarn-Level Cloth Simulation with Sliding Persistent Contacts. *IEEE Trans. on Vis. and Comput. Graph.* 23, 2 (Feb 2017), 1152–1162.

Sergio Conti and Francesco Maggi. 2008. Confining Thin Elastic Sheets and Folding Paper. *Arch. Rational Mech. Anal.* 187 (2008), 1–48.

Elliot English and Robert Bridson. 2008. Animating Developable Surfaces Using Nonconforming Elements. In *ACM SIGGRAPH 2008 Papers (SIGGRAPH '08)*. ACM, Article 66, 5 pages.

Marco Fratarcangeli, Valentina Tibaldo, and Fabio Pellacini. 2016. Vivace: A Practical Gauss-seidel Method for Stable Soft Body Dynamics. *ACM Trans. Graph.* 35, 6, Article 214 (Nov. 2016), 9 pages.

T. F. Gast and C. Schroeder. 2014. Optimization Integrator for Large Time Steps. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '14)*. Eurographics Association, 31–40.

P. E. Gill, W. Murray, and M. H. Wright. 1981. *Practical Optimization.* Academic Press, San Diego, USA.

Russell Gillette, Craig Peters, Nicholas Vining, Essex Edwards, and Alla Sheffer. 2015. Real-time Dynamic Wrinkling of Coarse Animated Cloth. In *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA '15)*. ACM, 17–26.

R. Goldenthal, D. Harmon, R. Fattal, M. Bercovier, and E. Grinspun. 2007. Efficient Simulation of Inextensible Cloth. *ACM Trans. Graph.* 26, 3 (2007), 49.

H. Goldstein. 1950. *Classical Mechanics.* Addison-Wesley Publishing Company.

G. Irving, C. Schroeder, and R. Fedkiw. 2007. Volume conserving finite element simulations of deformable models. *ACM Trans. Graph. (SIGGRAPH Proc.)* 26, 3 (2007), 13.1–13.6.

James T. Kajiya. 1986. The Rendering Equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '86)*. ACM, 143–150.

Jonathan M. Kaldor, Doug L. James, and Steve Marschner. 2008. Simulating Knitted Cloth at the Yarn Level. In *ACM SIGGRAPH 2008 Papers (SIGGRAPH '08)*.

Ladislav Kavan, Dan Gerszewski, Adam Bargteil, and Peter-Pike Sloan. 2011. Physics-inspired Upsampling for Cloth Simulation in Games. *ACM Trans. Graph.* 30 (2011).

Tae-Yong Kim, Nuttapong Chentanez, and Matthias Müller-Fischer. 2012. Long Range Attachments - a Method to Simulate Inextensible Clothing in Computer Games. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '12)*. 305–310.

Woojong Koh, Rahul Narain, and James F. O'Brien. 2014. View-dependent Adaptive Cloth Simulation. In *Proc. of the ACM SIGGRAPH/Eurographics Symp. on Comput. Anim. (SCA '14)*. 159–166.

W. Koh, R. Narain, and J. F. O'Brien. 2015. View-Dependent Adaptive Cloth Simulation with Buckling Compensation. *IEEE Transactions on Visualization and Computer Graphics* 21, 10 (Oct 2015), 1138–1145.

Randall J. LeVeque. 2002. *Finite volume methods for hyperbolic problems.* Cambridge University Press.

Tiantian Liu, Adam W. Bargteil, James F. O'Brien, and Ladislav Kavan. 2013. Fast Simulation of Mass-spring Systems. *ACM Trans. Graph.* 32, 6, Article 214 (Nov. 2013), 7 pages.

Marek Krzysztof Misztal, Robert Bridson, Kenny Erleben, Jakob Andreas Bærentzen, and François Anton. 2010. Optimization-based Fluid Simulation on Unstructured Meshes. In *Proc. of the 7th Workshop on Virtual Reality Interactions and Physical Simulations (VRIPhys2010)*. 11–20.

Matthias Müller and Nuttapong Chentanez. 2010. Wrinkle Meshes. In *Proc. of the 2010 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim. (SCA '10)*.

Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. 2007. Position based dynamics. *J. Visual Comm. and Image Repr.* 18, 2 (2007), 109–118.

Rahul Narain, Matthew Overby, and George E. Brown. 2016. ADMM ⊇ Projective Dynamics: Fast Simulation of General Constitutive Models. In *Proc. of the ACM SIGGRAPH/Eurographics Symp. on Comput. Anim. (SCA '16)*.

Rahul Narain, Tobias Pfaff, and James F. O'Brien. 2013. Folding and Crumpling Adaptive Sheets. *ACM Trans. Graph.* 32, 4 (2013), 51:1–51:8.

Rahul Narain, Armin Samii, and James F. O'Brien. 2012. Adaptive Anisotropic Remeshing for Cloth Simulation. *ACM Transactions on Graphics* (2012), 147:1–10. Proceedings of ACM SIGGRAPH Asia 2012, Singapore.

G. Narita, Y. Watanabe, and M. Ishikawa. 2016. Dynamic Projection Mapping onto Deforming Non-rigid Surface using Deformable Dot Cluster Marker. *IEEE Transactions on Visualization and Computer Graphics* PP, 99 (2016), 1–1.

Taylor Patterson, Nathan Mitchell, and Eftychios Sifakis. 2012. Simulation of Complex Nonlinear Elastic Bodies Using Lattice Deformers. *ACM Trans. Graph.* 31, 6, Article 197 (Nov. 2012), 10 pages.

X. Provot. 1995. Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *Graph. Interface*. 147–154.

X. Provot. 1997. Collision and self-collision handling in cloth model dedicated to design garment. *Graph. Interface* (1997), 177–89.

Damien Rohmer, Tiberiu Popa, Marie-Paule Cani, Stefanie Hahmann, and Alla Sheffer. 2010. Animation Wrinkling: Augmenting Coarse Cloth Simulations with Realistic-looking Wrinkles. In *ACM SIGGRAPH Asia 2010 papers (SIGGRAPH ASIA '10)*.

A. Selle, J. Su, G. Irving, and R. Fedkiw. 2009. Robust High-Resolution Cloth Using Parallelism, History-Based Collisions, and Accurate Friction. *IEEE Trans. on Vis. and Comput. Graph.* (2009), 339–350.

T. Shinar, C. Schroeder, and R. Fedkiw. 2008. Two-way Coupling of Rigid and Deformable Bodies. In *SCA '08: Proceedings of the 2008 ACM SIGGRAPH/Eurographics symposium on Computer animation (SCA '08)*. 95–103.

Justin Solomon, Etienne Vouga, Max Wardetzky, and Eitan Grinspun. 2012. Flexible Developable Surfaces. *Computer Graphics Forum* 31, 5 (2012), 1567–1576.

D. Terzopoulos and K. Fleischer. 1988. Deformable models. *The Vis. Comput.* 4, 6 (1988), 306–331.

D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. 1987. Elastically deformable models. *Comput. Graph. (Proc. SIGGRAPH 87)* 21, 4 (1987), 205–214.

Bernhard Thomaszewski, Simon Pabst, and Wolfgang Strasser. 2009. Continuum-based Strain Limiting. *Computer Graphics Forum* (2009).

Christopher D Twigg and Zoran Kačić-Alesić. 2011. Optimization for sag-free simulations. In *Proc. of the 2011 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.* ACM, 225–236.

P. Volino, M. Courchesne, and N. Magnenat-Thalmann. 1995. Versatile and efficient techniques for simulating cloth and other deformable objects. *Comput. Graph. (SIGGRAPH Proc.)* (1995), 137–144.

Huamin Wang, Florian Hecht, Ravi Ramamoorthi, and James F. O'Brien. 2010a. Example-based Wrinkle Synthesis for Clothing Animation. In *ACM SIGGRAPH 2010 Papers (SIGGRAPH '10)*.

Huamin Wang, James F. O'Brien, and Ravi Ramamoorthi. 2010b. Multi-Resolution Isotropic Strain Limiting. *ACM Trans. Graph.* 29, 6 (Dec. 2010), 156:1–10.

Huamin Wang and Yin Yang. 2016. Descent Methods for Elastic Body Simulation on the GPU. *ACM Trans. Graph.* 35, 6, Article 212 (Nov. 2016), 10 pages.

J. Wang, R. Paton, and J.R. Page. 1999. The draping of woven fabric preforms and prepregs for production of polymer composite components. *Composites Part A: Applied Science and Manufacturing* 30, 6 (1999), 757 – 765.