A Triangle Bending Constraint Model for Position-Based Dynamics.

Conference Paper · January 2010

DOI: 10.2312/PE/vriphys/vriphys10/031-037 · Source: DBLP

3 authors, including:

Sarah Niebe
University of Copenhagen
9 PUBLICATIONS **77** CITATIONS
SEE PROFILE

Kenny Erleben
University of Copenhagen
71 PUBLICATIONS **559** CITATIONS
SEE PROFILE

Workshop on Virtual Reality Interaction and Physical Simulation VRIPHYS (2010)
J. Bender, K. Erleben, and M. Teschner (Editors)

A Triangle Bending Constraint Model for Position-Based Dynamics

Micky Kelager [†] Sarah Niebe [‡] Kenny Erleben [§]

eScience Center, Department of Computer Science, University of Copenhagen, Denmark

Abstract

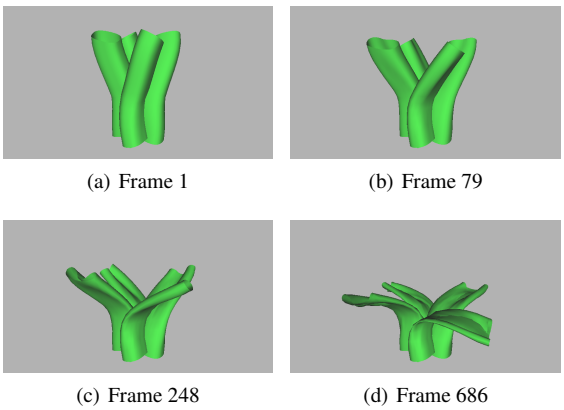
We present a novel bending model and constraint creation method for position-based dynamics. Our new bending model is introduced as an alternative to the current state-of-the-art dihedral bending model. Our model is motivated by geometric principles and operates on virtual triangles. It has the same cheap computational cost as the stick constraint model but with higher simulation quality and faster convergence than the dihedral bending model. Along with the model a new global bending parameter is introduced to control the curvature deformation at high precision compared to the traditional stiffness constant. Further, we propose a new constraint creation method that we believe is well suited for the triangle bending model and less affected by the underlying mesh tessellation.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.5]: Physically based modeling—Computer Graphics [I.3.7]: Animation—

Keywords: Bending Constraint, Curvature, Interactive Simulation, Position-based Dynamics

1. Introduction

Position-based dynamics [MHHR06] is the industry standard for interactive real-time applications [Jak01, Cou05, Cor10]. Maintaining only positions, and treating velocity as an implicit quantity, is favorable, as contacts and collisions can be handled by simple projections. The paradigm offers a certain amount of generality, as a wide variability of geometric constraints can be added without considering conservation laws, etc. With this paper, we address the issue of deformable models in interactive simulations. Our contribution is a new bending model that is inspired by simple geometric principles. Compared to competing bending models in position-based dynamics our model has a faster convergence rate that improves responsiveness and heightens fidelity. The proposed model is less sensitive to the choice of parameter values, and we expect that it will increase the ease of use. Our contribution addresses both equality and inequality constraints of the bending model.

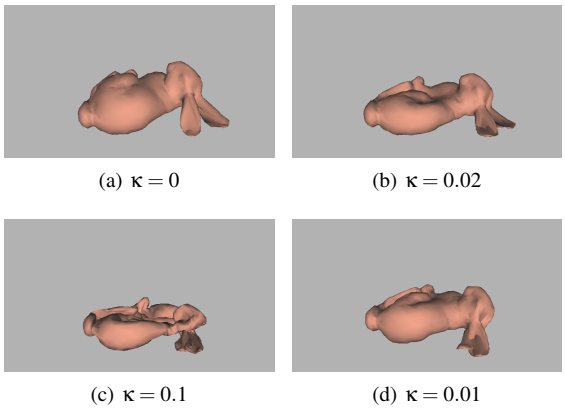


(a) Frame 1 (b) Frame 79

(c) Frame 248 (d) Frame 686

Figure 1: *A quasi plant is simulated using the physics-based dynamics paradigm and our triangle bending constraint model. The frames are grabbed from the supplementary movie.*

† e-mail: micky.kelager@gmail.com
‡ e-mail: sniebe@gmail.com
§ e-mail: kenny@diku.dk



(a) $\kappa = 0$ (b) $\kappa = 0.02$

(c) $\kappa = 0.1$ (d) $\kappa = 0.01$

Figure 2: *The triangle bending constraint model allows for fast response and high fidelity. The global parameter κ is increased to control the collapse of the Bunny and decreased to "inflate" it again.*

State-of-the-art interactive physics simulations suffer from linear convergence rates which result in damped elastic responses and cause low fidelity. Our work yields a fast response and increases the fidelity as illustrated in Figures 1 and 2. Current state-of-the-art is limited to specific contexts and does not support a wider range of deformation behaviors. The animation quality is highly dependent on the specific tessellation, the stability is limited, and parameter values are difficult to fine tune. Figure 3 shows results from rigid fabrics simulations and illustrates some of the issues with the competing bending methods.

Combined, the problems with the state-of-the-art method often imply the need of technical artists in e.g. the games industry. In this paper, we propose a method that is more stable and allows for a wider range of behaviors. Our hope is that our new method is more simple to use by modelers and animators.

2. Related Work on Bending Models

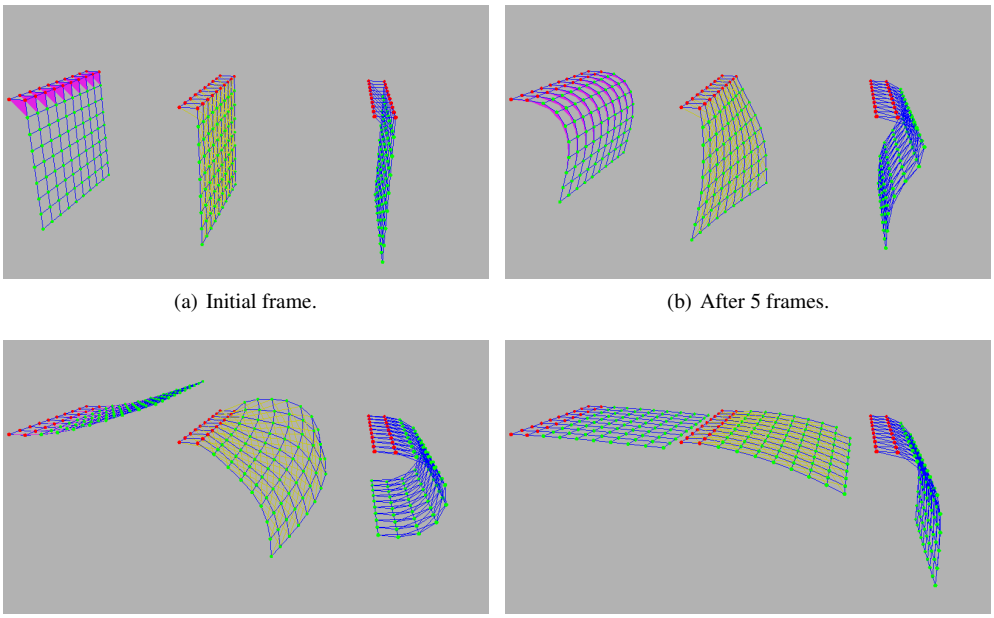
Many works in Computer Graphics have addressed bending models, in particular for cloth simulation. Here we will present a brief overview of the main approaches for bending models in the Computer Graphics community.

Mass-spring models have been used for cloth simulation [Pro95, BFA02] and have been generalized to other types of simulations like hair [SLF08]. Traditionally, a regular grid with springs that are connected to every other node is used to resist bending. These ideas have also been applied to stick models [Jak01]. Many variations exist for creating bending models on triangular meshes. The principle from regular grids is extended to triangular meshes by adding a bending connection between the tips of a winged triangle pair sharing an edge. In [CK02] a circular arc with constant arc length was used as the equilibrium shape of such a bending connection. This allowed for a numerically stable method for dealing with cloth buckling. Virtual nodes were exploited by [SLF08] to create an auxiliary spring from a virtual node on the bending connection to the shared edge of the two triangles. In [VMT06], the authors use a momentum conserving bending model, based on height differences, to evaluate the local curvature of a pair of winged triangles. In [THMG04] bending was added by taking a pair of winged triangles and use them to create a virtual tetrahedron and attaching a volume constraint to the tetrahedron.

A bending model, based on the dihedral angle between two triangles sharing an edge, was presented in [BW98]. The same model was used in position-based dynamics [MHHR06, MHHR07] and an improved quasi-multi-grid solver was later presented in [MÖ8]. A new improved bending force model, based on the dihedral angle, was introduced in [BMF03]. Here, the authors analyzed the deformation mode corresponding to only making changes in the dihedral angle and combining this single mode with a spring model, using coefficients weighted by the triangle mesh size. By construction, the model conserves momentum and only affects the dihedral bending. A related model was later used in [PKST08] to model seams.

Bending energy functionals belong to another paradigm. Here, an integral of some curvature measure serves as the mathematical foundation of the model. A curvature evaluation technique was used in [VCMT95] to compute an associated moment, which again could be turned into a force by considering triangle dimension and normals. A finite element model approach using co-rotational linear elasticity was taken in [EKS03]. Here, bending forces are evaluated by using a Laplace operator projected onto the surface normal direction. The average of the surface normal projections at the triangle vertices is used for each triangular element. A family of discrete curvature energies has been presented in [WBH*07]. One application of such a discrete curvature measure is found in [GHDS03]. In [Gri06], the circumcenters of a winged triangle pair are used to define a diamond shaped area, in which the mean curvature integral is evaluated. The approach is somewhat similar to a finite volume method. The resulting models [GHDS03, Gri06] of the bending energy is the squared dihedral angle, multiplied by the length of the shared edge, divided by the orthogonal distance between the circumcenters. Recently, elastic rods have been addressed in [BWR*08] and [ST07, ST09]. Here, both twist and bending modes are considered. In these models the bending energy functionals are used and discretization techniques similar to finite volume method are applied. To our knowledge, this work does not appear to have been used for interactive simulation.

Our work is based on position-based dynamics. Thus, none of the force-based approaches known from



(a) Initial frame. (b) After 5 frames.

(c) After 17 frames. (d) System at rest.

Figure 3: *Rigid fabric simulation meshes, comparing our method (to the left) against the state-of-the-art dihedral angle constraints (in the middle) and distance sticks (to the right). This test serves to stress how well the different bending constraints can recover from a highly violated state. In all cases, we use $k = 1$ and $\kappa = 0$ with no elasticity on the tension constraints.*

particle/mass-spring systems applies, nor would any of the bending energy driven approaches work. For interactive simulations the industrial state-of-the-art on bending models is based on dihedral angles. Our model is based on a distance measure between the centroid and the tip of a triangle.

3. A Triangle Bending Constraint Model

Position-based dynamics is current state-of-the-art in interactive simulation software such as PhysX and Bullet and makes use of the following bending model [MHHR06],

$$C_{\text{dihedral}}(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4) = \arccos(\mathbf{n}_1 \cdot \mathbf{n}_2) - \phi_0, \qquad (1)$$

where ϕ_0 is the initial dihedral angle and the unit normals $\mathbf{n}_1$ and $\mathbf{n}_2$ are given by

$$\mathbf{n}_1 = \frac{(\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1)}{\| (\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1) \|}, \qquad (2a)$$

$$\mathbf{n}_2 = \frac{(\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_4 - \mathbf{p}_1)}{\| (\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_4 - \mathbf{p}_1) \|}. \qquad (2b)$$

The advantage of this model, compared to distance or stick constraints, is that it is independent of stretching. Our model is semi-independent of stretching which means that on average there is no noticeable difference in tension lengths. The arccos-term in (1) poses a problem with the dihedral bending model, as it is only well defined in the range of $-1..1$ which corresponds to $0..\pi$ radians. This means that in a state of equilibrium a complete local reflection of the deformable object results in another equilibrium state.

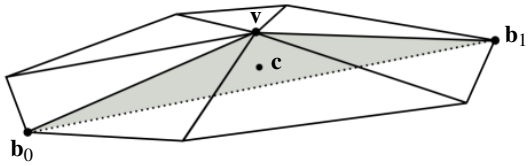
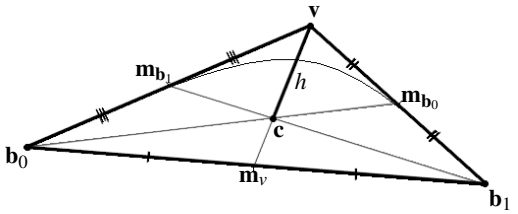
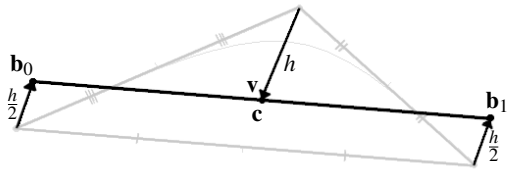


Figure 4: *The triangle bending constraint must span at least two triangles to model curvature.*

Working with position-based dynamics, it is essential to realize that many of the straight forward properties of the simulation paradigm come directly from the geometry itself. Thus, tension constraints are constructed from the edges and dihedral bending constraints are created between adjacent triangles that share an edge. Stiffness bending constraints are constructed from traversing the mesh topology and connecting every second, non-connected, vertex by the means of tension constraints. Our model is not directly dependent on topology in the sense that triangle bending constraints can be created between any three unique vertices as long as their contribution will model "curvature", i.e. for a triangle mesh any triangle bending constraint must span at least two tri-



(a) Triangle bending model notation.



(b) Triangle bending model principle.

Figure 5: *The triangle bending model. (a) The intersection point of the three medians is the centroid* **c**. *The radius of curvature is represented by the length h from the centroid to the tip of the triangle. By employing the length difference, a simple geometric constraint can be formulated. (b) The triangle in a collapsed state where the vertices have been displaced along the direction of the center median.*

angles as illustrated in Figure 4. Degenerated triangles are valid for our triangle bending constraints as it merely employs that the mesh does not initially curve in this case. Our algorithm for the creation of triangle bending constraints is detailed in Section 4.1.

The triangle bending model arises from the thought experiment of how to collapse a triangle with as few and as cheap operations as possible, while introducing as little potential energy as possible. The idea is to work with the centroid of the triangle which is illustrated in Figure 5(a). The centroid has a couple of neat properties for our purpose. The three medians intersect exactly at the centroid, which divides each median in a ratio of $2:1$, i.e. $\| \mathbf{v}-\mathbf{c} \| = 2 \| \mathbf{m}_v - \mathbf{c} \|$. To collapse the triangle while conserving both linear and angular momenta we displace the base line vertices ($\mathbf{b}_0$ and $\mathbf{b}_1$) with $\frac{1}{2}(\mathbf{v}-\mathbf{c})$ and update $\mathbf{v}=\mathbf{c}$ as illustrated in Figure 5(b).

The geometric principle in our method can be explained by computing the centroid **c** which is given by

$$\mathbf{c} = \frac{1}{3}(\mathbf{b}_0+\mathbf{b}_1+\mathbf{v}). \tag{3}$$

The distance between the tip of the triangle **v** and the centroid **c** is penalized as follows,

$$C_{\text{triangle}}(\mathbf{b}_0,\mathbf{b}_1,\mathbf{v}) \geq \| \mathbf{v}-\mathbf{c} \| - (\kappa + h_0), \tag{4}$$

where h_0 is the rest length (rest radius of curvature) and κ is a global bending parameter of the system (explained below). The expression in (4) is defined as an inequality constraint, $C_{\text{triangle}} \geq 0$, which will allow the underlying model to become more curved than its rest shape, e.g. plants and leaves. The corresponding equality bending constraint is defined as,

$$C_{\text{triangle}}(\mathbf{b}_0,\mathbf{b}_1,\mathbf{v}) = \| \mathbf{v}-\mathbf{c} \| - h_0, \tag{5}$$

where we implicitly assume that $\kappa = 0$. The global bending parameter κ is not confined from not being employed in (5) but this equality constraint is meant to model systems that want to stay true to their original curvatures, e.g. fabrics. By exerting that $\kappa > 0$, the underlined mesh can be forced into a shape it has not been designed for.

The material behavior in our triangle bending model is controlled by user defined parameters. These parameters include the global bending constant $\kappa \geq 0$ which can be thought of as a crude estimation of the mean curvature radius, and the stiffness constant $0 \geq k \geq 1$ which controls the rigidity of the bending constraints. We do not employ k directly in the positional projection corrections, as its contribution in general will be non-linear when we perform more than one iteration over the constraints. Instead, we employ a linear dependent stiffness constant as suggested by [MHHR06],

$$k' = 1-(1-k)^{\frac{1}{n}}, \tag{6}$$

where n is the number of solver iterations. Observe, our model suffers from the same deficiency as the dihedral model in not being able to discriminate between a symmetrical reflection of the model. However, our model will not let adjacent triangles collapse which can be provoked by the dihedral model due to symmetrical reflections. The bending constant κ controls the curvature explicitly whereas the stiffness constant k controls it implicitly. The reason for keeping both parameters is to support meshes with triangles that are significantly different in size. The value of κ is an absolute Euclidean distance where k can be seen as a percentage.

4. A Fast Method for Interactive Applications

We have implemented our bending constraint model into an existing position-based dynamics solver that allows us to simulate a variety of deformable objects, e.g. cloth, plants, pressure soft bodies, etc. When our triangle bending model is used as an equality constraint (5), we perform the positional projections every time. However, if the model is used as an inequality constraint (4) we evaluate the constraint and perform the projection corrections if and only if $C_{\text{triangle}} < 0$.

In the following, we show how to perform the positional projection corrections to the 3 vertices $\mathbf{b}_0,\mathbf{b}_1,\mathbf{v}$ for a given triangle bending constraint. The inverse mass w for a given vertex with mass $m > 0$ is

$$w = \frac{1}{m}, \tag{7}$$

where we employ $w = 0$ to indicate a fixed, or locked, vertex

```
01 : algorithm create-bending-constraints(Mesh M)
02 :   foreach vertex v in M do
03 :     V ← connected-vertices(v, M)
04 :     foreach vertex v_i in V do
05 :       cos(φ_best) ← 0
06 :       v_best ← v_i
07 :       foreach vertex v_j in V do
08 :         cos(φ) ← ((v_i − v)·(v_j − v)) / (||v_i − v|| ||v_j − v||)
09 :         if cos(φ) < cos(φ_best) then
10 :           cos(φ_best) ← cos φ
11 :           v_best ← v_j
12 :         end if
13 :       end foreach
14 :       if cons[index(v_best)] ≠ index(v_i) then
15 :         cons[index(v_i)] ← index(v_best)
16 :         create-bending-constraint(v_i, v_best, v)
17 :       end if
18 :     end foreach
19 :   end foreach
20 : end algorithm
```

Figure 6: *Pseudo-code to illustrate our triangle constraint creation algorithm.*

with infinite mass. We define the generalized inverse mass for the triangle as

$$W = w_{b_0} + w_{b_1} + 2w_v. \tag{8}$$

When a bending constraint of any type is being violated, the positional corrections to the 3 vertices yield

$$\Delta \mathbf{b}_0 = \frac{2w_{b_0}}{W}(\mathbf{v}-\mathbf{c})\left(1 - \frac{\kappa + h_0}{\| \mathbf{v}-\mathbf{c} \|}\right), \tag{9a}$$

$$\Delta \mathbf{b}_1 = \frac{2w_{b_1}}{W}(\mathbf{v}-\mathbf{c})\left(1 - \frac{\kappa + h_0}{\| \mathbf{v}-\mathbf{c} \|}\right), \tag{9b}$$

$$\Delta \mathbf{v} = -\frac{4w_v}{W}(\mathbf{v}-\mathbf{c})\left(1 - \frac{\kappa + h_0}{\| \mathbf{v}-\mathbf{c} \|}\right). \tag{9c}$$

Because the constraint evaluations can be squared on both sides of (4) and (5), and because the positional projections in (9a), (9b), and (9c) only require the evaluation of a single square root to compute the length of the difference vector, the triangle bending constraints can be solved very efficiently and very fast. This make them ideal for applications in interactive contexts.

4.1. Working with Triangle Meshes

It is possible to apply triangle bending constraints to an arbitrary triangle mesh, i.e. both an open and closed two manifold. One such approach would be to add a virtual vertex at the midpoint of every edge, and create triangle bending constraints over any two pair of triangles sharing an edge. Using the notation of the dihedral angle model we would use $\mathbf{b}_0 = \mathbf{p}_3$ and $\mathbf{b}_1 = \mathbf{p}_4$ and $\mathbf{v} = \frac{\mathbf{p}_1+\mathbf{p}_2}{2}$. This would correspond to the traditional edge-based bending connections used for triangle meshes where the projection of the virtual vertex is distributed to both edge vertices. In our experience, using the edge-based approach results in fewer bending constraints than our preferred approach, which will be explained below. However, the animation results suffer from the asymmetric topology of the mesh which is why we favor a different approach for creating the triangle bending constraints.

We have constructed an algorithm that in our experience works well for any two-dimensional triangle mesh. The algorithm operates on the mesh topology. For each vertex $\mathbf{v}$ in the mesh we look at the one ring neighborhood set $\mathcal{V}$. For each neighboring vertex $\mathbf{v}_i \in \mathcal{V}$, we seek to find another matching vertex $\mathbf{v}_j \neq \mathbf{v}_i$ in the neighborhood that is most in a straight direction through the vertex $\mathbf{v}$. Let us assume that each vertex is assigned an unique integer index $\mathbf{index}(\mathbf{v}_i)$ then our algorithm can be sketched in pseudo code as listed in Figure 6. In line 03, we assume that a method exists that returns all adjacent vertices connected to $\mathbf{v}$ through an edge. By resetting ϕ_{best} to zero in line 05, we ensure that no triangle bending constraint is created within the same half-space of $\mathbf{v}, \mathbf{v}_i$. The check in line 14 guarantees that all created constraints are distinct. The variable *cons* is an index map to record the constraint connections.

It should be noted that depending on how the triangle bending constraints are assigned to the mesh topology, we can obtain a variety of different material properties. This is different from the dihedral bending constraints which by design can only be created on shared triangle edges.

5. Experiments and Results

As reference for our experiments, we use an implementation of the dihedral-based bending model as the state-of-the-art method [MHHR06]. PhysX employs the same model where certain stability measures have been taken into account, such as clamping parameter values or excessive velocities, forces, etc. Similar position-based dynamics is employed in other software such as Bullet.

The experiments and measurements in this section were performed on an Intel Core2 Duo CPU at 2GHz with 3GB memory running 32bit Windows Vista. As shown in Table 1, our triangle bending model matches the interactive real-time performance demands on a level equal to that of the dihedral bending model. On average our method is more than a factor 3 faster than the dihedral model.

The convergence rate of our triangle bending model was compared against the dihedral bending model and the convergence plots are shown in Figure 7. We have plotted the

Rigid Fabric Performance Times

Con. type	# Cons	Computation time (ms) Avg.	Total
Triangle	160	0.000158	0.405987
Dihedral	144	0.000503	1.159640
Stick	160	0.000113	0.288305

Table 1: *Comparison of performance measurements in milliseconds for a single frame between the three different types of bending constraints. The rigid fabric model consists of* 100 *vertices.* 16 *solver iterations are used in all three cases. The test case is depicted in Figure 3. Observe, on average our method is a factor* 3 *faster than the dihedral bending method.*

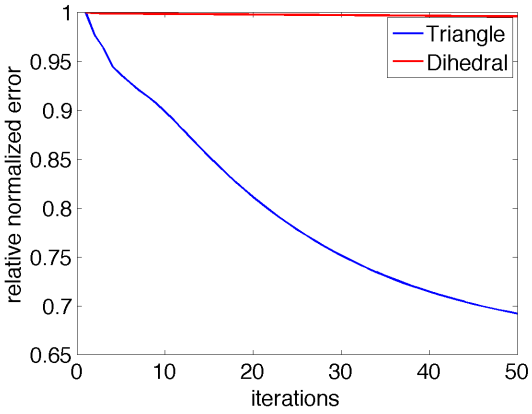
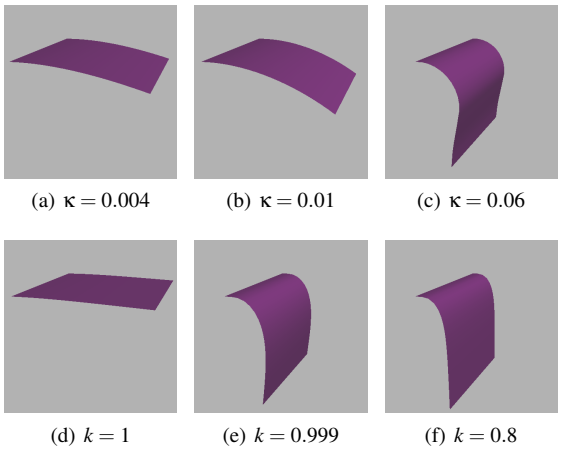


Figure 7: *Convergence plots showing how fast the triangle bending model converges, compared to the dihedral-based bending model. Observe, we have Q-linear convergence in both cases but the triangle method has a lower rate.*

normalized relative constraint error as a function of the number of iterations used in the constraint solver. For this study, a random frame from the first part of the simulation in Figure 3 was used. The convergence plots indicate similarity with Q-linear convergence. This is not surprising, as the constraint projection approach in nature is a steepest descent method. However, as shown our method has a lower convergence rate which yields a faster response. This can also be observed in the supplementary video that displays live frame grabs from our real-time interactive system.

We have performed a sensitivity study where we have subjected a rigid fabric to a wide range of parameter values and compared the different bending behaviors. Frame grabs from our results are depicted in Figure 8. The results show that our introduced parameter κ gives more sensitive control over the bending property, compared to the stiffness constant k that is usually employed in position-based dynamics.

All complex models that we have demonstrated make use



(a) $\kappa = 0.004$ (b) $\kappa = 0.01$ (c) $\kappa = 0.06$

(d) $k = 1$ (e) $k = 0.999$ (f) $k = 0.8$

Figure 8: *Visual comparison of varied parameter values. (a)- (c) The stiffness parameter is* $k = 1$ *(d)- (f) The curvature parameter is* $\kappa = 0$*. Observe that k gives a limited control over the bending property where* κ *is able to fine tune the bending property to a much higher level.*

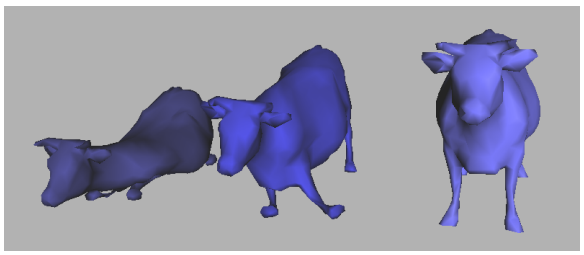


Figure 9: *The deformable cows employ the same mesh and triangle bending constraints, all with parameters* $k = 1$ *and* $\kappa = 0$*. The constraints are set to equality but the solver iterations vary. Left cow uses only 2 iterations, middle cow uses 16 iterations, and the right cow uses 64 iterations.*

of our constraint creation algorithm to place the triangle bending constraints. Examples of such models are depicted in Figure 1 where a quasi-plant illustrates nice curves, in Figure 2 where we collapse and inflate the Stanford Bunny, and in Figure 9 that illustrates different properties when varying the solver iterations. The first two figures make use of the inequality triangle bending constraint, where the last figure illustrates that the equality version can render a deformable shell surface nearly stiff.

6. Discussion and Conclusion

With this paper, we have presented a novel geometric bending model as an alternative to the current state-of-the-art dihedral angle-based bending model for interactive real-time simulations of deformable models. The new method was de-

veloped from geometric principles and guided by intuition. Experiments show that the new triangle bending model have at least the same simulation and animation qualities, but with faster convergence properties than the dihedral bending model. This is in our opinion an incremental, although substantial, improvement for position-based dynamics and its applications.

There are still issues that remain to be solved to further adapt deformable objects in interactive design applications. For starters, the inability to discover symmetrical reflections which cause inversions of the bending model are left open. Still, from an artistic point of view one must still know a certain amount of technical details to apply the methods. For instance, the connectivity of the mesh elements is necessary, and although our method appears to be less affected by tessellation more work on this is needed. We speculate that a preprocessing of geometries with the purpose of creating constraints in the physical model could be interesting. The current strategy for handling collisions in physics-based dynamics often result in what resembles ghost forces, and this is also an issue with our bending constraint model. A more robust way to handle collisions by projections is worth investigating.

We speculate that one possible avenue for improvement would be to use a "signed" height to deal with symmetrical reflection of the surface. A possible idea is to compute the signed area of the triangle defined by the triangle bending constraint upon creation of the constraint. During simulation we may then recompute the sign of the area and if it differs from the initial rest sign we could apply a special case for the constraint projection. We leave this for future work.

References

[BFA02] BRIDSON R., FEDKIW R., ANDERSON J.: Robust treatment of collisions, contact and friction for cloth animation. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2002), ACM, pp. 594–603. 2

[BMF03] BRIDSON R., MARINO S., FEDKIW R.: Simulation of clothing with folds and wrinkles. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 28–36. 2

[BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1998), ACM, pp. 43–54. 2

[BWR*08] BERGOU M., WARDETZKY M., ROBINSON S., AUDOLY B., GRINSPUN E.: Discrete elastic rods. *ACM Trans. Graph. 27*, 3 (2008), 1–12. 2

[CK02] CHOI K.-J., KO H.-S.: Stable but responsive cloth. *ACM Trans. Graph. 21*, 3 (2002), 604–611. 2

[Cor10] CORPORATION N.: Nvidia physx. http://developer.nvidia.com/object/physx.html, 2010. 1

[Cou05] COUMANS E.: The bullet physics library. http://www.continuousphysics.com, 2005. 1

[EKS03] ETZMUβ O., KECKEISEN M., STRAβER W.: A fast finite element solution for cloth modelling. In *PG '03: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications* (Washington, DC, USA, 2003), IEEE Computer Society, p. 244. 2

[GHDS03] GRINSPUN E., HIRANI A. N., DESBRUN M., SCHRÖDER P.: Discrete shells. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 62–67. 2

[Gri06] GRINSPUN E.: A discrete model of thin shells. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Courses* (New York, NY, USA, 2006), ACM, pp. 14–19. 2

[Jak01] JAKOBSEN T.: Advanced character physics. Presentation, Game Developer's Conference, 2001. 1, 2

[MÖ8] MÜLLER M.: Hierarchical position based dynamics. In *Proceedings of Virtual Reality Interactions and Physical Simulations (VRIPhys2008)* (Grenoble, November 2008). 2

[MHHR06] MÜLLER M., HEIDELBERGER B., HENNIX M., RATCLIFF J.: Position based dynamics. In *Proceedings of Virtual Reality Interactions and Physical Simulations (VRIPhys)* (Madrid, November 2006), pp. 71–80. 1, 2, 3, 4, 5

[MHHR07] MÜLLER M., HEIDELBERGER B., HENNIX M., RATCLIFF J.: Position based dynamics. *J. Vis. Comun. Image Represent. 18*, 2 (2007), 109–118. 2

[PKST08] PABST S., KRZYWINSKI S., SCHENK A., THOMASZEWSKI B.: Seams and bending in cloth simulation. In *Eurographics Workshop on Virtual Reality Interaction and Physical Simulation (VRIPHYS)* (2008). 2

[Pro95] PROVOT X.: Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *In Graphics Interface* (1995), pp. 147–154. 2

[SLF08] SELLE A., LENTINE M., FEDKIW R.: A mass spring model for hair simulation. *ACM Trans. Graph. 27*, 3 (2008), 1–11. 2

[ST07] SPILLMANN J., TESCHNER M.: Corde: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects. In *SCA '07: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2007), Eurographics Association, pp. 63–72. 2

[ST09] SPILLMANN J., TESCHNER M.: Cosserat nets. *IEEE Transactions on Visualization and Computer Graphics 15*, 2 (2009), 325–338. 2

[THMG04] TESCHNER M., HEIDELBERGER B., MULLER M., GROSS M.: A versatile and robust model for geometrically complex deformable solids. In *CGI '04: Proceedings of the Computer Graphics International* (Washington, DC, USA, 2004), IEEE Computer Society, pp. 312–319. 2

[VCMT95] VOLINO P., COURCHESNE M., MAGNENAT THALMANN N.: Versatile and efficient techniques for simulating cloth and other deformable objects. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1995), ACM, pp. 137–144. 2

[VMT06] VOLINO P., MAGNENAT-THALMANN N.: Simple linear bending stiffness in particle systems. In *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2006), Eurographics Association, pp. 101–105. 2

[WBH*07] WARDETZKY M., BERGOU M., HARMON D., ZORIN D., GRINSPUN E.: Discrete quadratic curvature energies. *Comput. Aided Geom. Des. 24*, 8-9 (2007), 499–518. 2