

A Geometric Deformation Model for Stable Cloth Simulation

Thomas Stumpp¹ Jonas Spillmann^{1,2} Markus Becker¹ Matthias Teschner¹

¹ Computer Graphics, University of Freiburg, Germany

² Computer Vision Laboratory, ETH Zurich, Switzerland

Abstract

We propose an adapted shape-matching approach for the efficient and robust simulation of clothing. A combination of two different cluster types is employed to account for high stretching and shearing, and low bending resistance. Due to the inherent handling of overshooting issues, the proposed shape-matching deformation model is robust. The proposed cluster types allow for a computationally efficient handling of bending. The geometric deformation model is combined with a novel collision handling approach. The technique employs spatial subdivision to detect collisions and self-collisions. The response scheme is derived from an existing approach for elastic rods. To illustrate the physically plausible dynamics of our approach, it is compared to a traditional physically-based deformation model. Experiments indicate that similar cloth properties can be reproduced with both models. The computational efficiency of the proposed scheme enables the interactive animation of clothing and shells.

1. Introduction

The simulation of clothing, i. e. two-dimensional deformable objects, is a challenging research field in both mechanics and computer graphics. Clothing is characterized in having a small thickness, compared to its planar extends. Therefore, clothes are usually modeled as surfaces in space, and as a consequence, they are usually discretized into triangle surfaces.

In the context of the deformation modeling of clothing, we observe that clothing is nearly inextensible, which results in large *in-plane* forces (i. e., the shear and stretch forces). This is unfortunate, since stiff equations require a small time step, which aggravates interactive animations. In turn, small *in-plane* forces allow for large time steps, but make the clothing looking unpleasantly ‘rubber-like’. In contrast, the *out-of-plane* forces (i. e., the bending forces) are significantly smaller. Some approaches [GHF^{*}07, MHHR07] have addressed the large *in-plane* forces by imposing constraints and iteratively solving the governing system of equations. In turn, this degrades the performance of the system. Moreover, the required codes are often complex from an algorithmic point of view.

The *geometric deformation model* of Müller *et al.* [MHTG05] enforces inextensibility-constraints in a non-iterative fashion based on the knowledge of the reference shape of the object. By registering the reference

shape of the object with the simulated deformed shape, they can compute goal positions that allow for an explicit and unconditionally stable dynamic simulation. To account for deformations, they blend the rotated reference shape with the linearly or quadratically deformed reference shape.

Using the original linear and quadratic transformations proposed in Müller *et al.*, we cannot model the rich non-linear deformations of clothing. Therefore, we adopt the cluster concept of [MHTG05] to enrich the possible deformation modes while still exploiting the stability of the approach to model significantly higher stretching resistance compared to previous approaches.

Our contribution We propose a novel deformation model for the robust and efficient simulation of clothing. We base our deformation model on the geometric deformation model of Müller *et al.* [MHTG05]. Using a suitable mesh partitioning, both *in-plane* and *out-of-plane* deformations are handled simultaneously. This contrasts previous works that usually computed the *in-plane* deformation per triangle, and the *out-of-plane* deformations per edge.

We start by partitioning the nodes of the simulation mesh into clusters that contain three points each. Using triangular clusters, we can efficiently extract the optimal rotation for the shape matching without the iterative computation of the polar decomposition. We then compute goal positions based on the rigidly transformed, initial cluster points and pull

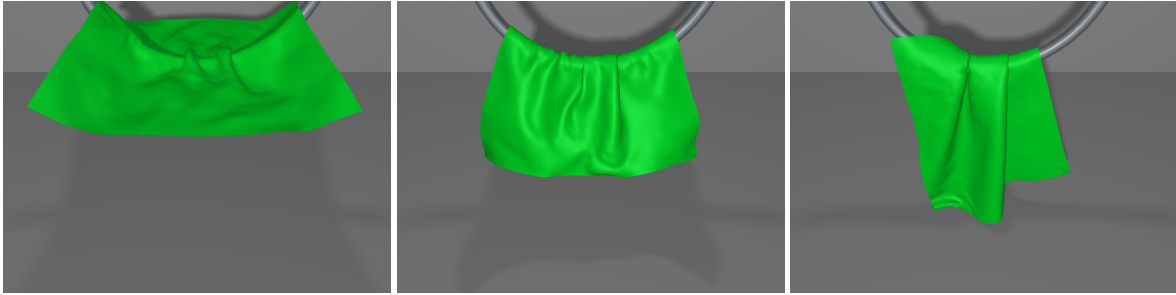


Figure 1: By employing our geometric deformation model for the simulation of cloth, we can reproduce the fascinating wrinkling patterns of real cloth, although our method is not physically-based. The mesh consists of 5k vertices and 10k faces, the time step is 2ms, and the computation of the elastic response takes 13.53 ms per simulation step.

the mass points towards these goal positions. We can model clothing and shells by choosing appropriate clusters. By additionally incorporating so-called inextensible fiber clusters the method allows to compute very stiff material. Due to the shape matching, stiff materials can be handled without the need of too restrictive time steps. For stable handling of collisions and self-collisions we adopt an existing approach for elastic rods [ST08].

Although our method is not physically-based, the simplicity of the implementation and the ability to simulate both, clothing and shells at interactive rates, are key benefits of our approach. Experiments and comparisons with a previous approach illustrate that the visual behavior of clothing can be plausibly reproduced (see Fig. 1).

Organization The remainder of this paper is organized as follows: In Sec. 2, the related work is discussed. Then, in Sec. 3, we give an overview on our cloth simulation loop. We then explain the clustering and the corresponding elastic response in Sec. 4. The contact handling is addressed in Sec. 5. In Sec. 6, we evaluate our approach with respect to performance and stability and illustrate the wide applicability.

2. Related work

In the field of computer graphics, the modeling of thin shells has always been an active research area. A remarkable effort has especially been made in deriving bending forces that are identified as the main challenge in thin shell modeling. This comes from the observation that the bending potential depends intuitively on the angle between adjacent triangles, whose computation cannot be done with linear functions. For a comprehensive review, the reader is referred to the book of House and Breen [HB00], and to the survey of Choi and Ko [CK05a].

One of the first investigations are the geometric deformation model of Weil [Wei86], and Terzopoulos *et al.*'s classic paper [TPBF87]. Later, Provot [Pro95] identified that

the stretch forces must be significantly larger than the bending forces. Of note are further the works of Volino *et al.*, e. g. [VCT95] that present simulated garments on moving characters. A significant advance in the simulation of cloth has been made by Baraff and Witkin [BW98]. They derive the elastic forces from conditions that are related to the deformation modes. To integrate the mass-points, they employ a backward Euler scheme which requires the solution of a sparse linear system.

In [CK05b], Choi and Ko addressed the instability that comes with the buckling behavior of cloth. Bending is as well focused in the work of Bridson *et al.* [BMF03]. They have recognized that the forces arising from bending deformation must exclusively affect the bending modes, and have given the corresponding formulas. Moreover, they have proposed an adaptive model to address the high curvature when cloth is draped over sharp edges. Grinspun *et al.* [GHDS03] have derived a similar bending model from discrete geometry. Our model shares similarities with the fiber-based deformation model of Wicke *et al.* [WSG05]. They sample a meshless surface into fibers, and compute the elastic forces based on the geometric configuration of the fibers. In contrast, we sample the meshless surface into clusters. The elastic forces are then approximated by blending the clusters accordingly.

All these models have in common that the solid is basically simulated as a particle system, i. e. as a discrete material. An alternative way is opened by Etmuss *et al.* [EKS03] who have proposed to employ the FE method to compute the stretch-, shear- and bending forces. To compute the in-plane deformation forces, a co-rotational approach in the spirit of [MG04] is employed. Later, Thomaszewski *et al.* [TWS06] have extended their approach by modeling in-plane deformation modes and bending modes in a unified way. However, the high computational costs forbid interactive simulations. In contrast to these works, we address the animation of thin shells, having in mind that in interactive applications, the goal is to have a robust and efficient deformation model.

One of the main challenges of cloth simulation comes from the non-linearity of the bending forces. This issue is addressed in the works of Bergou *et al.* [BMWG07], Volino and Magnenat-Thalmann [VMT06] and Garg *et al.* [GGWZ07]. The latter have shown that for inextensible thin shells, the bending energy is cubic, which results in quadratic bending forces. Consequently, an implicit solver can rely on a relatively simple force Jacobian. Since in our deformation model, the elastic response is governed by the geometric configuration of the clusters, we do not rely on the order of the force terms. Instead, an explicit integration is employed to evolve the mass points towards their goal positions. Moreover, in our model, the bending resistance is realized through overlapping clusters. Thus, expensive trigonometric functions are avoided.

The inextensibility of cloth has been addressed by Goldenthal *et al.* [GHF*07] that have proposed an approach based on Lagrangian dynamics to solve for the constraints. In this context to be mentioned is also the cloth simulation framework of Müller *et al.* [MHHR07] who have been focused on the stable simulation of cloth at interactive rates. These approaches differ significantly in terms of the formulation of the constraints. However, in the end, both perform an iterative computation for maintaining the constraints. In contrast, our scheme does not perform any iterative computations while still obtaining a large stretching resistance.

Our work is inspired by the geometric deformation model of Müller *et al.* [MHTG05]. There have been a couple of other works drawing upon their work. Rivers and James have proposed to subdivide the volume of an object into overlapping box-clusters in order to allow for more realistic deformations [RJ07]. And recently, Steinemann *et al.* have extended the approach of Rivers and James by considering adaptive clustering in order to simulate cutting [SOG08]. Our work is similar in spirit, but since we employ clusters with only three points, a more efficient extraction of the optimum rotation is enabled.

3. Overview

Our approach bases on the work of Müller *et al.* [MHTG05] where they have addressed a deformation model for point clouds. To compute an elastic deformation, the deformed point cloud is registered with the undeformed point cloud to calculate *goal positions*. Particles are then pulled towards these goal positions. As the computation of the goal positions is limited to rigid, linear or quadratic deformations, the overshooting problems are avoided even for objects with low elasticities. Thereby, the introduction of stiffness to the system that could lead to instabilities when using an explicit integration scheme is avoided. To enhance the range of possible deformation, the points can be partitioned into clusters, where the deformation is computed for each cluster separately and the results are blended. We adopt the idea of

Müller *et al.* in the context of cloth deformation modeling, where our algorithm summarizes as follows:

- As input, we expect a connected triangle mesh where at most two triangles share a common edge. Although the deformation model considers only the mass points, the edges are mandatory to enable the contact handling. The nodes of the mesh constitute the initial (undeformed) mass points $\mathbf{x}(0)$.
- As a preprocessing step, we partition the mesh into a set of overlapping *clusters* that contain three points each (see Fig. 2). We additionally employ *inextensible fibers*. They are used to link the clusters and thereby improve the stretching stiffness. The fibers are discussed in more detail in Sec. 4.2.
- In the simulation loop, three integration steps are performed. First, the external forces such as gravity or user-interactions are computed that act at the time t on the object. Then the mass points $\mathbf{x}(t)$ are evolved in time to obtain the *unconstrained positions* $\mathbf{x}^*(t+h)$.
- Next, each cluster of the object is matched with its rigidly transformed, undeformed cluster. This results in the goal positions $\mathbf{g}(t+h)$ and corresponding displacements $\Delta\mathbf{g}(t+h)$. Then the mass points $\mathbf{x}^*(t+h)$ are pulled towards their goal positions, yielding the positions $\mathbf{x}^{**}(t+h)$.
- Based on the elastic positions $\mathbf{x}^{**}(t+h)$, we detect collisions. Then, an iterative process computes feasible positions and corresponding contact forces $\mathbf{f}^{**}(t+h)$. By again numerically integrating the mass points, the *constrained positions* $\mathbf{x}(t+h)$ are obtained.

4. Deformation approach

The deformation modeling of clothing and thin shells is a challenging problem. To model visually pleasing clothing, a cloth should be inextensible and unshearable, but at the same only have a low bending resistance.

In this section, we discuss the clustering and the computation of the goal positions and corresponding displacements. Pulling the points to their goal positions afterwards is termed *elastic response* in this section. This approach enables the efficient and physically plausible simulation of clothing.

4.1. Cluster-based deformations

Our approach is motivated by the observation that the in-plane deformations can be consistently captured by considering a deformed surface patch and the corresponding patch in the resting state. Having in mind that we want the cloth to be inextensible and unshearable, we observe that this is the case if the deformed surface patch conforms to the undeformed patch that is only transformed by its rigid modes that are three translation and three rotation modes. However, applying the rigid shape matching of [MHTG05] to the whole

simulation mesh would eliminate in- and out-of-plane deformations. The cloth would then behave like a rigid plate, without the characteristic wrinkling behavior.

To enable out-of-plane deformations while still allow for low shearing and extension we employ the so-called cluster-concept [MHTG05, RJ07]. Thereby the mesh is decomposed into M overlapping clusters. For each cluster i the optimal rotation \mathbf{R}_i is calculated for each cluster separately. This concept has also been used by Rivers and James [RJ07] to enrich the possible deformation modes of volumetric objects.

Cluster geometry To enable an efficient extraction of the rotation, we use a triangular cluster geometry. Thereby we can avoid Jacobi rotations [SD92]. Details are found in Appendix A.

The first idea might be to define the triangles of the mesh as clusters. While this approach cancels out shearing and stretching, it does not allow for an elastic response to bending, since a triangle considered in isolation cannot be bent. Instead, we form a cluster i for a triangle T_i using the corners of the triangles that are adjacent to T_i . Then the outer corners of this triangle constitute the cluster i . Fig. 2 (left) illustrates the cluster geometry.

This cluster geometry has several advantages. First, the elastic response to bending is now enabled since the clusters do overlap, as indicated in Fig. 2 (right). Second, since each cluster consists of three points, an efficient extraction of the optimal rotation is enabled.

Goal positions Having extracted the optimal rotation \mathbf{R}_i as proposed in the appendix, we are ready to compute the goal positions \mathbf{g} , which is accomplished by rotating the undeformed cluster geometry about \mathbf{R}_i . Following Müller *et al.*, we account for the conservation of momentum by rotating the relative positions of the cluster with respect to its center of mass [MHTG05]. However, for points that belong to several clusters multiple goal positions \mathbf{g}_i^c are computed and conservation of momentum is not guaranteed anymore.

To fight this problem, Müller *et al.* have proposed to add the terms $\frac{\mathbf{g}_i^c - \mathbf{x}_i}{h}$ for all clusters c in the velocity calculation. This procedure conserves the momentum, but it is subject to overshooting. The reason is that if a point is in M clusters, then the differences $\mathbf{g}_i^c - \mathbf{x}_i$ might all point in the same direction; by adding the differences, an obviously wrong goal position \mathbf{g}_{sum} is obtained (see Fig. 3).

Instead we follow [RJ07] and scale the sum of the goal positions for each point with the total number of clusters M_i it belongs to. Thereby overshooting for points in overlap regions is avoided.

The elastic response is then performed by pulling the mass points towards the summed and scaled differences. To accomplish this, we employ the integration scheme that has been proposed by Müller *et al.* [MHTG05]. In doing so, the blending parameter α can be used to model the stiffness of

the elastic response. Details on the integration and the blending parameters are found in [MHTG05]. Currently we do not model damping, but we believe that the model in [RJ07] could easily be adopted for our approach.

At this point, we want to note that our method is not scale-invariant. The reason is that the flexibility of the cloth depends on the overlap of the clusters. Since these clusters have always the same topology (see Fig. 2), the number of triangles that are employed to represent a given piece of cloth significantly influences the material behavior. Meshes with fewer vertices always look stiffer than large meshes. While this feature can be regarded as a limitation, it can in turn be exploited to simulate shells at interactive rates, as illustrated in Sec. 6.

Complexity Using the proposed cluster generation scheme the number of clusters corresponds to the number of number of triangles for a regular simulation mesh. The extraction of the optimal rotation is less complex than computing the stretching, shearing and bending forces. All three deformation modi can now be handled simultaneously. Moreover, the direct computation of the bending moments is related to the number of edges while our scheme depends on the number of triangles. For regular triangle meshes, the number of edges is about 1.5 times the number of triangles. Details are given in Sec. 6.

4.2. Inextensible fiber clusters

With the concept proposed in the preceding section, a consistent treatment of shearing, stretching and bending moments is enabled. To further reduce stretching we introduce so-called inextensible fiber clusters. That is, we partition the mesh into one-dimensional clusters. To accomplish this, we walk from each edge to the adjacent edge sharing an angle closest to 180° , thereby ensuring that the clusters are

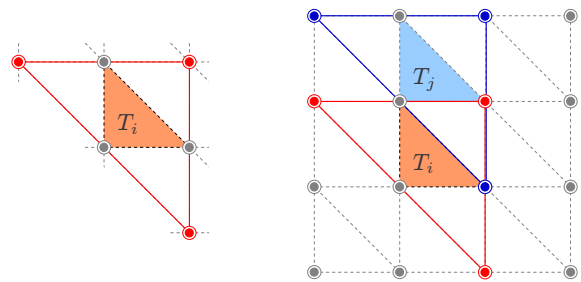


Figure 2: The geometry for the triangle clusters. The three outer corners of the three adjacent triangles of triangle T_i constitute the cluster i (left). Since there is a cluster for each triangle in the mesh, the clusters do overlap. To give an example, the clusters for the triangles T_i and T_j are shown in red and blue (right).

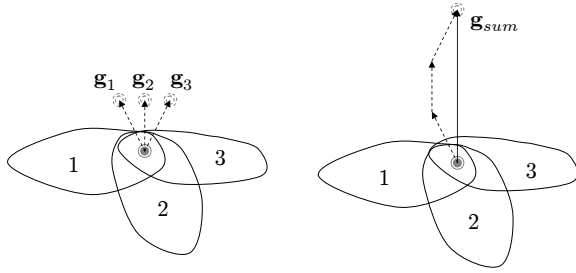


Figure 3: If a point belongs to multiple clusters, then each cluster contributes a goal point difference $\mathbf{g}_i^c - \mathbf{x}_i$. Since these differences are added in the original shape matching, the resulting goal position \mathbf{g}_{sum} is not plausible anymore. We denote this problem as ‘overshooting’.

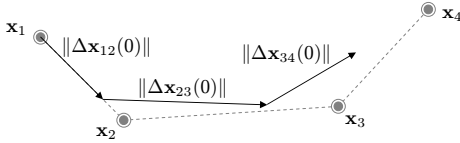


Figure 4: To compute the goal positions for the inextensible fiber cluster, we iteratively walk from point to point, thereby considering the initial distance $\|\Delta\mathbf{x}_{i,i+1}(0)\| = \|\mathbf{x}_{i+1}(0) - \mathbf{x}_i(0)\|_2$.

as straight as possible. For regular meshes, the fiber clusters are chosen along the dominant directions.

Instead of extracting a rotation from these clusters, we then compute the goal positions from the current mass point positions under the constraint that the distances between the points conform to the distances in the undeformed configuration. To accomplish, we perform two sweeps, once from the first to the last point, and once from the last back to the first point. The resulting goal positions are averaged and blended with the goal positions of the triangular clusters.

In the sweep, we go from the point in direction of the subsequent point, thereby considering the initial distance (see Fig. 4). In order to preserve the momentum, the resulting displacements are translated such that they sum to 0. The resulting goal positions and corresponding displacements are treated in an identical way as the goal positions of the triangle clusters. This procedure is fast, but it is not physically correct since it does not correspond to the minimization of the elastic energy, and since it does not necessarily preserve the angular momentum. For typical cloth simulation scenarios, the resulting artifacts are hardly visible though.

5. Contact handling

The fascinating behavior of highly flexible material such as clothing cannot be plausibly reproduced without considering collisions and self-collisions. In this section, we briefly

address the employed contact handling. For a concise discussion, we refer to the cited documents.

Similar to many cloth contact handling schemes [BFA02], we assume the cloth to have a user-defined thickness. Repulsion forces ensure that the separation distance between any two primitives is at least the given thickness. Since the thickness is usually small compared to the area of the cloth, interpenetrations must be strictly avoided. Therefore, we handle both, edge- and point-collisions.

To accomplish this, we employ a predictor-corrector approach. That is, we first evolve the mass points in an unconstrained way. Then, collisions are detected on the future positions $\mathbf{x}^{**}(t+h)$. The collision detection step is accelerated by employing a spatial subdivision scheme [TKH*05, THM*03].

After grouping collisions into collision groups, an iterative scheme is employed to compute the constrained, feasible positions $\mathbf{x}(t+h)$. The approach is an adoption of the edge-based scheme of Spillmann *et al.* [ST08]. That is, we resolve collisions between edges locally such that the momentum is conserved. By iteratively repeating the process, the resulting displacements quickly propagate through stacked clothing. Then, contact forces $\mathbf{f}(t)$ can be computed such that the mass points meet the constrained positions $\mathbf{x}(t+h)$ after numerical re-integration. Currently we do not handle friction in our models. Fig. 5 illustrate the stable handling of collisions and self-collisions with our scheme. For details about the collision handling scheme, we refer to the works of Spillmann *et al.* [ST08] and Bridson *et al.* [BFA02].

6. Results

In this section, we evaluate our geometric deformation model for the simulation of cloth. We start with a comparison to a traditional physically-based deformation model in terms of performance and stability. Then, we show that by employing our deformation model, clothing can be simulated plausibly, and the characteristic bending patterns can be reproduced. Moreover, we show applications in the field of thin shell simulation.

All experiments have been carried out on a 2.13 GHz Intel Core 2 Duo using a single thread. The real-time renderings have been done with Ogre3D (<http://www.ogre3d.org>) the off-line renderings with Povray (<http://www.povray.org>). To enhance the visual quality of the renderings, we have employed a pure geometric surface subdivision procedure.

6.1. Evaluation

We are aware that it is always difficult to compare different approaches as each one has its own (dis-)advantages. However, to evaluate our deformation model, we have implemented the mass-spring deformation model that has been proposed by Bridson *et al.* [BMF03]. Their deformation

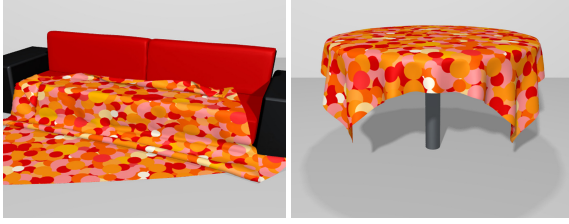


Figure 5: Using our collision handling scheme, a large number of collisions and self-collisions can be stably handled. The cloth in the left scenario comprises of 19k triangles and 10k vertices, the cloth on the right comprises of 33k triangles and 17k vertices.

model is force-based and the bending forces are guaranteed not to affect the in-plane deformations. To evolve the mass points, we employ an explicit second order leapfrog scheme.

Stability In a first experiment, we evaluate the stability for low extensibility. Thus, we perform an experiment similar to what Goldenthal *et al.* [GHF*07] did: A rectangular piece of cloth is attached in space at two corners, and hangs vertically. In this setting, the stretch of the cloth is particularly well visible. In our setting, the mesh consists of 1K vertices and 2K triangles. We parameterize Bridson *et al.*'s model such that the stretch is comparable to the stretch of our model. Evaluating the maximum possible time step for a stable dynamic simulation, we found that a timestep of at most 1 ms for the approach of Bridson *et al.* is possible. Using our model, a time step of 5 ms can be chosen. However, we have to underline that we did not examine the general case. This issue will be addressed in future work.

Performance In another experiment we evaluate the performance of both approaches. As already discussed, by employing the triangular clusters, an efficient extraction of the rotation is enabled. Shear, stretch and bending forces are all handled using the clusters. In contrast, the traditional approach of Bridson *et al.* computes the stretch and shear forces per triangle, and the bending forces per edge.

To measure the performance, we use a rectangular piece of cloth of varying size that is vertically hanging down. The lower corners are then elongated such that the cloth is swinging about its vertical resting state. As we solely want to measure the cost of the deformation model, we neglect collisions and self-collisions. For both methods, the computation time grows linearly in the size of the mesh. In this experiment, the approach of Bridson *et al.* is about 50% slower than our approach. (see Fig. 6).

6.2. Application

To show that our approach can plausibly reproduce the fascinating wrinkling pattern of knitted cloth, we have simulated a rectangular piece of cloth that is falling onto a sphere.

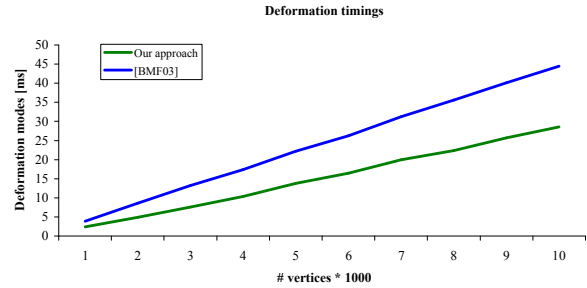


Figure 6: Timings for computing the elastic response of meshes with varying sizes by employing our approach, and by employing Bridson *et al.* [BMF03] approach. As the timings indicate, the complexity of our approach grows linearly in the size of the mesh. Further, the approach as proposed in [BMF03] is about 50% slower than our approach.

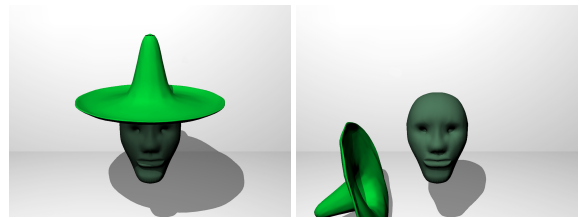


Figure 8: User interaction with an elastic model built from 109 vertices and 204 triangles.

The mesh consists of 6k vertices and 12k triangles. The time step of the simulation is 2ms. Though our approach is not physically-based, it nevertheless provides similar results than the approach of Bridson *et al.*, as illustrated in Fig. 7.

By exploiting the dependency of our approach on the number of clusters, we can simulate thin shells. By linking the clusters cyclically, objects with an intrinsic curvatures can be animated.

Fig. 8 shows two screenshots where the user interacts with an elastic object. The object is modeled from 109 vertices and 204 triangles. The time step of the simulation is 2 ms, and the simulation runs at 38 frames per second.

Fig. 5 illustrates two examples with a larger number of collisions and self-collisions.

7. Conclusion

We have presented an approach for the robust simulation of clothing based on shape matching. We have started by partitioning the mesh into triangular clusters. We then have extracted the optimal rotation with an efficient scheme that avoids an iterative computation of the polar decomposition. Based on the optimal rotation, we have computed the goal positions and corresponding displacements. By scaling these

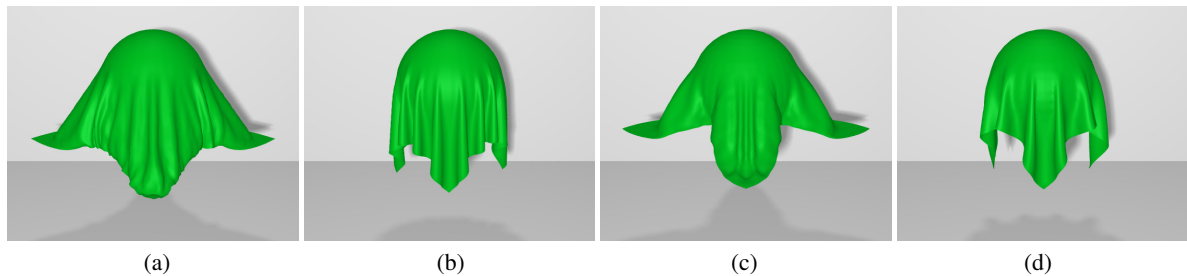


Figure 7: Simulation of a piece of cloth falling onto a sphere. The mesh consists of 6k vertices and 12k triangles. The goal of this experiment is to show that the wrinkling patterns of the physically-based, traditional approach (a, b) can be plausibly reproduced with our approach (c, d).

displacements in the same way as Rivers and James [RJ07], we have addressed the overshooting problem of the original shape matching approach [MHTG05]. The elastic response is then established by pulling the mass points towards these goal positions. In doing so, both in-plane and out-of-plane deformations can be handled simultaneously.

The performed experiments indicate that we can take up to 5 times larger time steps for clothing with comparable stretching resistance than the mass-spring approach of Bridson *et al.* [BMF03]. At the same time the proposed approach is about 50% faster.

We have shown various applications of our approach, ranging from complex off-line computation of visually appealing clothing up to interactive animations of clothing and shells. Our approach is rounded off by our robust contact handling scheme.

Limitations and future work Since the size of the clusters are related to the topology of the underlying mesh, our method is not scale-invariant. Consequently, larger meshes give less resistance to bending while small meshes result in a stiff behavior, similar to thin shells. This is an obvious drawback of our approach, although we believe that it will not limit the application of our approach in interactive animations such as games. Moreover, we are working on an alternative clustering method in order to improve the resistance to in-plane deformations while preserving the flexibility in the out-of-plane deformations. An appropriate friction model is also part of the future work.

References

- [BFA02] BRIDSON R., FEDKIW R., ANDERSON J.: Robust treatment of collisions, contact and friction for cloth animation. *ACM Transactions on Graphics* (2002), 594–603.
- [BMF03] BRIDSON R., MARINO S., FEDKIW R.: Simulation of clothing with folds and wrinkles. In *Proc. ACM SIGGRAPH/Eurographics symposium on Computer animation* (2003), pp. 28–36.
- [BMWG07] BERGOU M., MATHUR S., WARDETZKY M., GRINSPUN E.: Tracks: toward directable thin shells. *ACM Transaction on Graphics (Proc. SIGGRAPH)* 26, 3 (2007), 50.
- [BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *Proc. SIGGRAPH* (1998), pp. 43–54.
- [CK05a] CHOI K., KO H.: Research problems in clothing simulation. *Computer-Aided Design* 37, 6 (2005), 585–592.
- [CK05b] CHOI K.-J., KO H.-S.: Stable but responsive cloth. *ACM Transaction on Graphics* 21, 3 (2005), 604–611.
- [EKS03] ETZMUß O., KECKEISEN M., STRAßER W.: A fast finite element solution for cloth modelling. In *Proc. 11th Pacific Conference on Computer Graphics and Applications* (2003), p. 244.
- [GGWZ07] GARG A., GRINSPUN E., WARDETZKY M., ZORIN D.: Cubic shells. In *Proc. ACM SIGGRAPH/Eurographics symposium on Computer animation* (2007), pp. 91–98.
- [GHDS03] GRINSPUN E., HIRANI A. N., DESBRUN M., SCHRÖDER P.: Discrete shells. In *Proc. ACM SIGGRAPH/Eurographics symposium on Computer animation* (2003), pp. 62–67.
- [GHF*07] GOLDENTHAL R., HARMON D., FATTAL R., BERCOVIER M., GRINSPUN E.: Efficient simulation of inextensible cloth. *ACM Transaction on Graphics (Proc. SIGGRAPH)* 26, 3 (2007), 49.
- [HB00] HOUSE D., BREEN D.: *Cloth modeling and animation*. A. K. Peters Ltd, 2000.
- [MG04] MÜLLER M., GROSS M.: Interactive virtual materials. In *Proc. Graphics Interface* (2004), pp. 239–246.
- [MHHR07] MÜLLER M., HEIDELBERGER B., HENNIX M., RATCLIFF J.: Position based dynamics. *Journal of Visual Communication and Image Representation* 18, 2 (2007), 109–118.
- [MHTG05] MÜLLER M., HEIDELBERGER B., TESCHNER M., GROSS M.: Meshless deformations based on shape matching. *ACM Transaction on Graphics (Proc. SIGGRAPH)* 24, 3 (2005), 471–478.
- [Pro95] PROVOT X.: Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *Graphics Interface '95* (1995), pp. 147–154.
- [RJ07] RIVERS A. R., JAMES D. L.: FastLSM: fast lattice shape matching for robust real-time deformation. *ACM Transaction on Graphics (Proc. SIGGRAPH)* 26, 3 (2007), 82.

- [SD92] SHOEMAKE K., DUFF T.: Matrix animation and polar decomposition. In *Proc. Graphics interface '92* (1992), pp. 258–264.
- [SOG08] STEINEMANN D., OTADUY M., GROSS M.: Fast adaptive shape matching deformations. In *Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008). to appear.
- [ST08] SPILLMANN J., TESCHNER M.: An Adaptive Contact Model for the Robust Simulation of Knots. *Computer Graphics Forum (Proc. Eurographics)* 27, 2 (2008), 497–506.
- [THM*03] TESCHNER M., HEIDELBERGER B., MÜLLER M., POMERANTES D., GROSS M. H.: Optimized spatial hashing for collision detection of deformable objects. In *Proc. Vision, Modeling, Visualization* (2003), pp. 47–54.
- [TKH*05] TESCHNER M., KIMMERLE S., HEIDELBERGER B., ZACHMANN G., RAGHUPATHI L., FUHRMANN A., CANI M., FAURE F., MAGNENAT-THALMANN N., STRASSER W.: Collision Detection for Deformable Objects. *Computer Graphics Forum* 24, 1 (2005), 61–81.
- [TPBF87] TERZOPOULOS D., PLATT J., BARR A., FLEISCHER K.: Elastically deformable models. *Computer Graphics (Proc. SIGGRAPH)* 21, 4 (1987), 205–214.
- [TWS06] THOMASZEWSKI B., WACKER M., STRASSER W.: A consistent bending model for cloth simulation with corotational subdivision finite elements. In *Proc. ACM SIGGRAPH/Eurographics symposium on Computer animation* (2006), pp. 107–116.
- [VCT95] VOLINO P., COURCHESNE M., THALMANN N. M.: Versatile and efficient techniques for simulating cloth and other deformable objects. In *SIGGRAPH '95: Proc. of the 22nd annual conference on Computer graphics and interactive techniques* (1995), pp. 137–144.
- [VMT06] VOLINO P., MAGNENAT-THALMANN N.: Simple linear bending stiffness in particle systems. In *Proc. ACM SIGGRAPH/Eurographics symposium on Computer animation* (2006), pp. 101–105.
- [Wei86] WEIL J.: The synthesis of cloth objects. *SIGGRAPH Comput. Graph.* 20, 4 (1986), 49–54.
- [WSG05] WICKE M., STEINEMANN D., GROSS M.: Efficient animation of point-based thin shells. In *Proc. Eurographics* (2005), pp. 667–676.

Appendix A: 2D shape matching

The extraction of the optimal rotation for a cluster with exactly three points corresponds to match a triangle with its undeformed counterpart. To accomplish this, we apply the following procedure.

Let $\mathbf{x}_1, \mathbf{x}_2$ and \mathbf{x}_3 be the points in \mathbb{R}^3 . Then we define points $\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2 \in \mathbb{R}^2$ with

$$\bar{\mathbf{x}}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \bar{\mathbf{x}}_2 = \begin{pmatrix} \|\mathbf{x}_2 - \mathbf{x}_1\| \\ 0 \end{pmatrix} \quad (1)$$

If we further define the relative position vectors $\mathbf{b}_1 = \mathbf{x}_2 - \mathbf{x}_1$ and $\mathbf{b}_2 = \mathbf{x}_3 - \mathbf{x}_1$, then we can compute a third point $\bar{\mathbf{x}}_3$ from the cosine and sine of the angle θ ,

$$\cos \theta = \frac{\mathbf{b}_1 \cdot \mathbf{b}_2}{\|\mathbf{b}_1\| \|\mathbf{b}_2\|} \quad \sin \theta = \frac{\|\mathbf{b}_1 \times \mathbf{b}_2\|}{\|\mathbf{b}_1\| \|\mathbf{b}_2\|} \quad (2)$$

$$\Rightarrow \bar{\mathbf{x}}_3 = \|\mathbf{b}_2\| \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix} \quad (3)$$

Now we proceed similarly to the 3D case, that is, we look for the matrix \mathbf{A} that minimizes $\sum_{i=1}^3 m_i (\mathbf{A}\mathbf{q}_i - \mathbf{p}_i)^2$. Thus $\mathbf{A} = \mathbf{A}_{pq}\mathbf{A}_{qq}$ and $\mathbf{A}_{pq} = \sum_{i=1}^3 m_i \mathbf{p}_i \mathbf{q}_i^T$. To establish the 2D case, we follow Shoemake and Duff [SD92], i. e., if the entries of \mathbf{A}_{pq} are

$$\mathbf{A}_{pq} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

then

$$\mathbf{R} = \mathbf{A}_{pq} + \text{sign}(\det \mathbf{A}_{pq}) \cdot \begin{pmatrix} A_{22} & -A_{21} \\ -A_{12} & A_{11} \end{pmatrix} \quad (4)$$

We then compute the goal positions $\bar{\mathbf{g}}_i \in \mathbb{R}^2$ as $\bar{\mathbf{g}}_i = \mathbf{R}(\bar{\mathbf{x}}_i(0) - \mathbf{c}(0)) + \mathbf{c}$, where $\mathbf{c}(0)$ and \mathbf{c} is the center of mass in the undeformed and current configuration. Now let

$$\alpha_i = \frac{\bar{g}_{i,x} \cdot \bar{x}_{3,y} - \bar{g}_{i,y} \cdot \bar{x}_{3,x}}{\bar{x}_{2,x} \cdot \bar{x}_{3,y}} \quad (5)$$

$$\beta_i = \frac{\bar{g}_{i,y}}{\bar{x}_{3,y}} \quad (6)$$

where $\bar{g}_{i,x}, \bar{g}_{i,y}, \bar{x}_{i,x}$ and $\bar{x}_{i,y}$ are the components of $\bar{\mathbf{g}}_i$ and $\bar{\mathbf{x}}_i$. The goal positions in \mathbb{R}^3 are obtained as

$$\mathbf{g}_i = \mathbf{x}_1 + \alpha_i \mathbf{b}_1 + \beta_i \mathbf{b}_2 \quad (7)$$

Of note is that the goal positions can be extracted directly, i. e. without employing an iterative scheme.