Design and Fabrication of Flexible Rod Meshes

Jesús Pérez[1] Bernhard Thomaszewski[2] Stelian Coros[2,3] Bernd Bickel[2,4]
José A. Canabal[1] Robert Sumner[2] Miguel A. Otaduy[1]

[1]URJC Madrid [2]Disney Research Zurich [3]Carnegie Mellon University [4]IST Austria

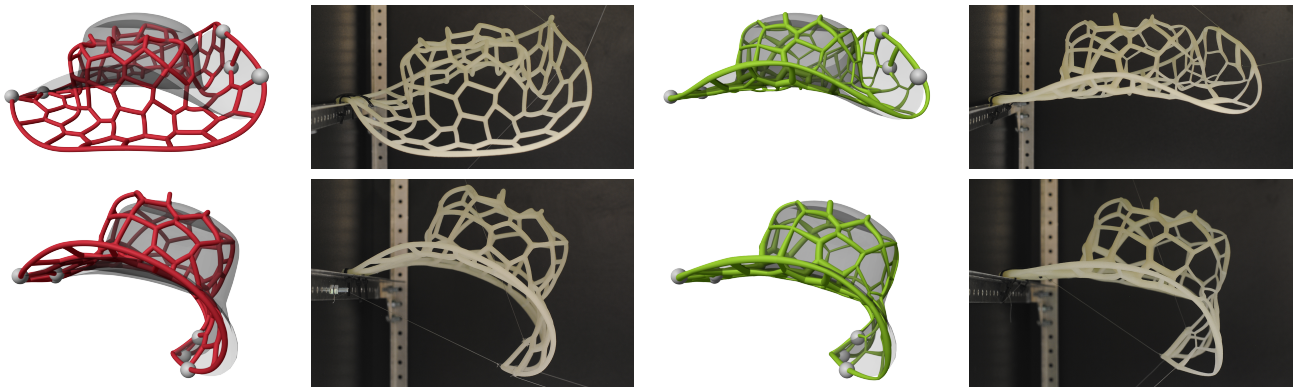

Figure 1: *Our computational design method allows us to fabricate a deformable hat with a desired deformation behavior. The two left columns show a hat with the default rod mesh, which does not deform as desired (shown in transparent gray). The two right columns show that, by optimizing the radii and rest-shape of the rod mesh, we can fabricate in one piece a deformable hat that deforms as desired.*

Abstract

We present a computational tool for fabrication-oriented design of flexible rod meshes. Given a deformable surface and a set of deformed poses as input, our method automatically computes a printable rod mesh that, once manufactured, closely matches the input poses under the same boundary conditions. The core of our method is formed by an optimization scheme that adjusts the cross-sectional profiles of the rods and their rest centerline in order to best approximate the target deformations. This approach allows us to locally control the bending and stretching resistance of the surface with a single material, yielding high design flexibility and low fabrication cost.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling

Keywords: fabrication, material design, rods

1 Introduction

Digital fabrication enables an extremely fast transition from virtual prototypes to their physical realization. Originally limited to rigid materials, 3D printers have now opened the door for fast prototyping of deformable objects. But while rigid designs merely required the specification of geometry, deformable printing technology poses a grand challenge on computer-aided design: how to specify the desired deformation properties of the target object. The simplest approach could be to specify local material parameters at each point in the target object, and then map these parameters to the closest available material. Unfortunately, it is highly non-trivial to understand what combination of local material parameters will produce a desired global deformation behavior.

We explore the use of flexible rod meshes as an implementation of deformable surfaces. Rod meshes are ideal for building lightweight and cost-efficient physical shapes. The mesh provides a truss structure for a deformable surface and can even be filled with foam or dressed with fabric that is allowed to slide. But most importantly, the global deformation properties of a rod mesh can be adjusted simply by locally varying the cross-sectional parameters of the rods. As a result, a heterogeneous deformable object can be fabricated in one piece and from a single base material using a variety of rapid prototyping technologies. We can find instances of flexible rod meshes in furniture, apparel, architecture or accessories, for which even a standalone rod mesh might be found aesthetically pleasing. There are many other applications of rod meshes as supporting structures, which include but are not limited to toys, puppeteering, costumes, animatronics and robotics. Those use cases would potentially need volume to accommodate internal components and/or actuators, what encourages even further the use of deformable surfaces.

In this paper, we propose a computational tool for the design and fabrication of flexible rod meshes. As shown in Fig. 1, our method takes as input several poses of a deformable surface with known position and/or force constraints and automatically computes a printable representation of a rod mesh that best approximates the shapes. One of the main features of our method is the choice of design space. We use rod meshes dominated by hexagonal faces, because hexagons can stretch, shear and bend by deforming their edges. Given such a mesh, its design space consists of the rest-shape and two orthogonal radii describing the ellipsoidal cross section at each point of the rods. By adjusting two orthogonal radii conveniently oriented, we manage to control in-plane deformations (stretch and shear) and out-of-plane deformations (bending) of the surface independently. In order to estimate the design parameters, we propose a simulation-based optimization approach. To this end, we extend

the discrete elastic rod model [Bergou et al. 2008; Bergou et al. 2010] to handle connections between multiple rods. We follow a co-rotational approach to estimate the orientations of connections kinematically from incident rods and show how to transmit bending and twist forces between connected rods. In addition, we propose an optimization framework that determines design parameters by minimizing the approximation error with respect to the input poses while satisfying static equilibrium constraints. In order to handle these constraints efficiently, we compute gradients of the objective function that satisfy the constraints implicitly.

We have applied our design and fabrication method to a variety of examples. In particular, we have explored the potential of the method for toy and apparel design. In order to validate the behavior of our designs in practice, we additionally fabricated three physical prototypes.

2 Related Work

Computational Design and Fabrication Designing physical objects whose appearance, motions or deformation behaviors can be intuitively specified and controlled is an important research challenge that is quickly gaining interest in the computer graphics community. Consequently, a variety of methods that investigate computational aspects of fabrication have been proposed. For example, design methods for creating objects that can stand on their own [Prévost et al. 2013] or spin stably [Bacher et al. 2014] have been presented. Thanks to recent work, it is also now possible to create articulated, 3D printable representations of virtual characters [Bächer et al. 2012; Calì et al. 2012], and to design mechanical characters capable of interesting motions [Coros et al. 2013; Ceylan et al. 2013; Thomaszewski et al. 2014].

Our work is most related to methods that control the deformation behavior of elastic objects—a problem that has received increased attention from the research community over the last few years. For instance, Chen et al. [2013] proposed a unified framework to fabricate objects with different controllable properties, including deformation, while Vidimče et al. [2013] proposed a framework for the fabrication of objects composed of multiple materials. Bickel et al. [2010] showed how to fabricate objects whose force-deformation response matches measurements of real objects. A small set of template materials with different deformation behaviors was used for this purpose. Skouras and her colleagues [2013] described an optimization method for computing an inhomogeneous distribution of material parameters to control the way in which fabricated elastic objects deform under the influence of external forces. The rest configuration of elastic objects can also be computed using automated methods. For instance, Skouras et al. [2012] showed that the shape of inflatable balloons can be controlled to match input targets by optimizing the rest state of membrane-based models. Similarly, the rest state of volumetric objects can be modified to control the deformation behavior of skin for robotic faces [Bickel et al. 2012], or to specify the way in which elastic objects deform under gravity [Chen et al. 2014]. Our work shares a similar goal to this class of methods. However, the types of objects we consider are represented using rod meshes, whose rest configurations and material properties are optimized.

Our rod meshes share some similarity with truss structures, which have also received attention from the computer graphics community. Wang et al. [2013], for instance, have recently proposed a computational design method that optimizes a truss structure to minimize the amount of printing material used in the fabrication of rigid objects, while Song and coauthors [2013] introduced an interactive design tool for creating stable networks of interleaved rods (known as reciprocal frames). Also related is the recent work of Garg et al. [2014], who develop techniques for designing wire-mesh structures with prescribed shapes. However, while these methods aim to create structures that are statically stable and self-supporting, the method we describe allows us to explicitly control the way in which our rod meshes deform under the influence of external forces.

Material and rest-shape Optimization The problem of optimizing material parameters and rest-shapes has also been explored in the context of animation. For instance, Kondo et al. [2005] control the deformation of elastic objects by keyframing rest-shapes. Twigg and Kacic-Alesic [2011] compute rest lengths for mass springs systems in order to achieve desired garment drapes under gravity. Coros and his colleagues [2012] show that it is possible to create autonomous characters modeled as elastic objects by optimizing their rest-shapes as a function of high-level motion goals. By adding a potential energy term specified relative to a set of input example shapes, the methods described by Martin et al. [2011] and Schumacher et al. [2012] are also related to the problem of modulating rest-shapes in order to effect the behavior of dynamic simulations. More recently, Li et al.[2014] have demonstrated the use of material parameter optimization within a space-time method, and Xu et al. [2015] have proposed an interactive method based on optimization to assist users in the design of deformable materials. The method we propose is inspired by this body of work, but our aim is to design physical objects with interesting deformation behaviors. As a further departure from previous work, we represent our objects using rod meshes.

Rod Simulation A variety of rod simulation methods have been presented in the literature, based, for example, on mass-spring systems [Rosenblum et al. 1991; Selle et al. 2008; Iben et al. 2013], Cosserat models [Pai 2002; Bertails et al. 2006; Spillmann and Teschner 2007], or articulated rigid body systems [Hadap 2006]. The discrete elastic rod model described by Bergou et al. [2008; 2010], however, constitutes a better starting point for our work, as it is experimentally validated, and it lends itself well to numerical optimization, as we demonstrate in this paper. Other approaches, such as the model introduced by Spillmann and Teschner [2009], which can handle rod meshes, albeit at the expense of using constraints, or the implicit centerline representation used by Casati and Bertails-Descoubes [2013], could in principle be used as well, but at the expense of increasing the complexity of the optimization method.

In addition to forward simulation, the problem of finding a rest-shape in order to obtain desired deformations has been investigated for a number of different curve and rod models, including articulated rigid body chains [Hadap 2006], 2D elastic curves [Derouet-Jourdan et al. 2010], and the super helices rod model [Derouet-Jourdan et al. 2013]. These methods are designed for systems of individual strands, possibly with intermittent contacts, but they do not directly extend to the globally-coupled rod networks that we consider in this work.

3 Rod Networks

Our design tool builds on physics-based simulation in order to predict the deformation of rod networks in response to applied forces. Among the many existing computational approaches, we choose the discrete elastic rod model by Bergou et al. [2008; 2010], due to its compact curve-angle representation with explicit centerline. We note, however, that other approaches based on reduced coordinates [Bertails et al. 2006] or full coordinates with constraints [Spillmann and Teschner 2007] would, in principle, be possible as well.

In order to identify the design parameters that we expose to our optimization routine, we briefly summarize the relevant part of the

theory here. Since the model by Bergou et al. does not account for coupling among rods, we propose a model for rod connections, formulate elastic energy terms, and describe how to correctly transmit bending and twist forces across connections.

3.1 Discrete Elastic Rods

Thin Kirchhoff rods are conveniently modeled as adapted framed curves with an arc-length parameterized centerline $\gamma(s) \in \mathbb{R}^3$ and an orthonormal material frame $\mathbf{m}(s) = [\mathbf{t}(s)\, \mathbf{b}(s)\, \mathbf{n}(s)] \in SO(3)$, where $\mathbf{t}$ is the tangent and $\mathbf{b}$, $\mathbf{n}$ span the cross-sectional plane of the rod. In the curve-angle representation, the material frame is expressed through a twist angle $\theta(s)$ from a reference frame $\mathbf{r}(s) = [\mathbf{t}(s)\, \mathbf{u}(s)\, \mathbf{v}(s)] \in SO(3)$. In the method of Bergou et al. [2010], which we adopt, this reference frame is constructed by parallel transport over time of a frame initialized to the material frame in rest state.

In the discrete setting, the centerline of the rod is represented as a piece-wise linear curve defined by a set of n nodes $\{\mathbf{x}_i\}$. The material frames are represented by a set of twist angles $\{\theta_i\}$ that pertain to the edges of the centerline.

We can generally expect bending to be the dominant deformation mode in rod networks, and it is the one that we control most through our optimization tool. Although we also account for stretching and twisting, we therefore focus on the bending energy here, which is defined as

$$E_b = \frac{1}{2\, l_0} (\boldsymbol{\kappa} - \boldsymbol{\kappa}_0)^T \mathbf{B} (\boldsymbol{\kappa} - \boldsymbol{\kappa}_0)\,, \tag{1}$$

where $\boldsymbol{\kappa}$ and $\boldsymbol{\kappa}_0$ are discrete curvatures in the deformed and undeformed configurations, respectively, and $\mathbf{B} \in \mathbb{R}^{2\times 2}$ models the material's anisotropic bending stiffness. It should be noted that $\boldsymbol{\kappa}$ is defined per vertex as a function of the incident edges and their material frames. Hence, κ depends on both, centerline positions and twist angles. Please see [Bergou et al. 2008] for details.

Material Parameterization To simplify the fabrication of our rod meshes, we assume that the entire network is built from a single material. In order to still control the deformation properties of the resulting surface locally, we adjust the cross-sectional geometry of the rods. Assuming elliptical cross sections, we can adjust the two radii in order to independently control bending of rods in the tangent plane of the resulting surface or bending out-of-plane (Fig. 3). To this end, the anisotropic bending model of Bergou provides the appropriate flexibility.

We define the elliptical cross-section at rod nodes with major and minor radii a and b. Knowing the Young modulus E, and with a cross-section area $A = \pi\, a\, b$, bending stiffness can be computed as [Bergou et al. 2010]:

$$\mathbf{B} = \frac{E\, A}{4} \begin{pmatrix} a^2 & 0 \\ 0 & b^2 \end{pmatrix}. \tag{2}$$

Simulation In our optimizations we solve static equilibria of rod meshes using dynamic relaxation with kinetic damping [Volino and Magnenat-Thalmann 2007], with adaptive time stepping for higher efficiency. To advance the configuration of rods over time, we follow the dynamic simulation approach described in [Bergou et al. 2010].

3.2 Rod Networks

In order to simulate the deformations of rod networks, we must model the coupling at connections such as to correctly transmit bending and twist forces between different rod segments.

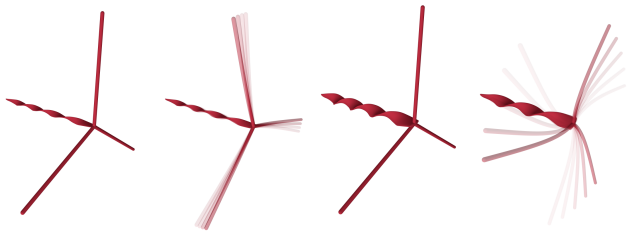

Figure 2: *This example depicts force transfer at rod connections with our connection model. The axis of a windmill is twisted, and upon release its twist transfers into bending of the blades. A thinner axis (left) produces less momentum than a thicker one (right), as one would expect. The centerline of the axis was kept fixed.*

Coupling Energy For each rod incident in a given connection, we seek to define an energy that captures its deformation relative to all other incident rods. We follow a corotational approach, measuring deformation potentials w.r.t. rigidly transformed connections. Specifically, for each rod incident in a connection, we condense the effect of the remaining rods into a single *connection edge* whose orientation is determined kinematically using best-fit rigid transformations. Another possibility would be to define pairwise energies for the segments incident in a connection, as done in the Cosserat Nets framework [Spillmann and Teschner 2009], but we found it more difficult to weight pair energy contributions to ensure a correct integration volume.

We consider the set of rod edges incident in a connection, with rest-shape material frames $\{\mathbf{m}_{i,0}\}$. In the deformed setting, we estimate a rotation $\mathbf{R}$ of the rest-shape connection that best matches the deformed connection. Then, for each incident edge, we define a connection edge with material frame $\bar{\mathbf{m}}_i = \mathbf{R}\, \mathbf{m}_{i,0}$, i.e., the rigidly rotated rest-shape material frame. Reference frames of connection edges, $\bar{\mathbf{r}}$, are initialized as their corresponding material frames, and then parallel transported over time, just as for regular rod edges. Based on the material normal $\bar{\mathbf{n}}$ and the reference normal $\bar{\mathbf{v}}$, the twist angle at a connection edge can be defined as $\bar{\theta}_i = \arccos(\bar{\mathbf{n}}^T\, \bar{\mathbf{v}})$.

Given material and reference frames of incident edges and their connection edges, we can formulate bending and twist energies at the connections based on the deviations between incident edges and their corresponding connection edges. By summing up energies for all edges incident in a connection, we obtain the total coupling energy at a connection:

$$\mathbf{V} = \sum_i V_i(\mathbf{t}_i, \theta_i, \bar{\mathbf{t}}_i, \bar{\theta}_i). \tag{3}$$

Here, V_i represents the sum of bending and twist energies of the discrete elastic rods model, between the i-th incident edge and its corresponding connection edge.

In practice, we estimate the rotation $\mathbf{R}$ of a connection by minimizing the deformation of material frames of all incident edges. We measure the deviation of the material tangent $\mathbf{t}$ and the material normal $\mathbf{n}$. We first estimate the linear transformation that minimizes the least-squares geometric error:

$$\mathbf{A} = \arg\min \sum_i \frac{\mathrm{tr}(\mathbf{B}_i) \|\mathbf{t}_i - \mathbf{A}\, \mathbf{t}_{i,0}\|^2}{\sum_j \mathrm{tr}(\mathbf{B}_j)} + \frac{\beta_i \|\mathbf{n}_i - \mathbf{A}\, \mathbf{n}_{i,0}\|^2}{\sum_j \beta_j}, \tag{4}$$

and then extract the rotation through polar decomposition $\mathbf{A} = \mathbf{R}\, \mathbf{S}$. By weighting the deformations of tangents and normals with each edge's bending and twist stiffness respectively, we favor the

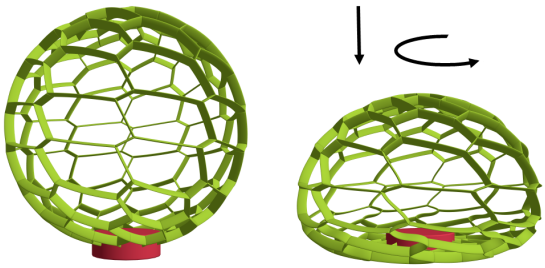

Figure 3: *Dynamics of a sphere mesh that is fixed at the bottom and deforms under gravity. The highly anisotropic bending behavior favors tangent-plane over out-of-plane deformation, causing twist around the vertical axis to accommodate compression.*

alignment of thicker, and hence, stiffer edges. Here, β refers to the twist stiffness. Note that weights are normalized to avoid any bias towards the set of tangents or normals.

Coupling Forces To compute the bending and twist forces at connections, the respective energy terms need to be differentiated w.r.t. the degrees of freedom of incident rod edges. The difficulty stems from the fact that the material frame at connection edges is kinematically defined by the best-fit rigid transformation. Thus, it depends through rotation on all material frames of incident edges. Therefore, the coupling energy of an edge pair formed by an incident edge and its connection edge produces non-zero bending and twist forces in all other incident edges. This is actually the expected behavior, because bending of one incident edge may produce twist on other incident edges and vice versa (Fig. 2).

Let us denote as $\mathbf{f}_{\mathbf{t}_i}$ and $\mathbf{f}_{\theta_i}$ the coupling forces on the i-th incident edge and twist angle respectively. By differentiating (3), we obtain the coupling force on a generic scalar degree of freedom w_i of the i-th incident rod:

$$\mathbf{f}_{w_i} = -\frac{\partial V_i}{\partial w_i} - \sum_j \frac{\partial V_j}{\partial \bar{\mathbf{t}}_j}\frac{\partial \bar{\mathbf{t}}_j}{\partial w_i} + \frac{\partial V_j}{\partial \bar{\mathbf{n}}_j}\frac{\partial \bar{\mathbf{n}}_j}{\partial w_i}. \tag{5}$$

The partial derivatives of individual bending and twist energies, $\partial V/\partial w_i$, $\partial V/\partial \bar{\mathbf{t}}$, and $\partial V/\partial \bar{\mathbf{n}}$ are obtained easily from the discrete elastic rods energy expressions. To this end, recall that the twist at the connection edge can be linked to the material normal through $\bar{\theta}_i = \arccos(\bar{\mathbf{n}}^T \bar{\mathbf{v}})$.

To complete the definition of coupling forces in (5), we must compute the differentials of material tangents and normals at connection edges from $\bar{\mathbf{m}} = \mathbf{R}\,\mathbf{m}_0$. By the chain rule, the differential of the j-th rotated rest-shape tangent or normal $\bar{\mathbf{q}}_j$ is:

$$\frac{\partial \bar{\mathbf{q}}_j}{\partial w_i} = \sum_k \frac{\partial \mathbf{R}}{\partial a_k}\left(\frac{\partial a_k}{\partial \mathbf{t}_i}\frac{\partial \mathbf{t}_i}{\partial w_i} + \frac{\partial a_k}{\partial \mathbf{n}_i}\frac{\partial \mathbf{n}_i}{\partial w_i}\right)\mathbf{q}_{j,0}, \tag{6}$$

where $\{a_k\}$ represent the elements of the linear transformation of the connection, $\mathbf{A}$. In the expression above, the terms $\partial \mathbf{R}/\partial a_k$ correspond to derivatives of polar decomposition [Barbic and Zhao 2011], while $\partial a_k/\partial \mathbf{t}_i$ and $\partial a_k/\partial \mathbf{n}_i$ are constant and easily computed from the linear expression that defines $\mathbf{A}$ from (4) [Müller et al. 2005]. Derivatives of the material tangent and normal w.r.t. the edge and the twist angle can be deduced from the definition of the material frame in discrete elastic rods. Note that $\partial \mathbf{n}/\partial \mathbf{e}$ is not zero, as we must account for the implicit dependence of the normal on the tangent due to parallel transport.

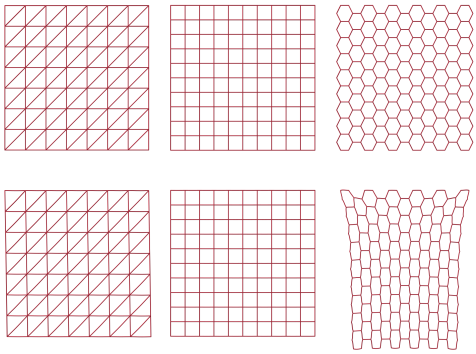

Figure 4: *With the same material and mesh density, a hexagonal mesh is more compliant to stretch and shear deformations than a triangle or quad mesh, thereby increasing the flexibility of our optimization method. In this example we show meshes with approximately the same number of cells, in their rest configuration (top), and after hanging from them the same weight (bottom).*

Stable implicit integration requires the computation of energy Hessians, i.e., force Jacobians. Other than straightforward derivatives, or already present in the original formulation, our rod networks require the computation of second derivatives of polar decompositions. Barbic and Zhao [2011] derived the scalar second derivative of polar decomposition, but we require a matrix of second derivatives with mixed terms, shown in Appendix A.

Fig. 3 shows the application of our connection model to the dynamic simulation of a sphere-shaped rod mesh. When the sphere is compressed under gravity, it also twists around its vertical axis, as the highly anisotropic bending behavior of its rods favors tangent-plane over out-of-plane deformations.

4 Optimization

Our optimization framework takes as input a few deformed poses of a rod mesh under known boundary conditions. Then, it automatically computes the rest-shape and cross-section of the rods such that the mesh best matches the input poses under the same boundary conditions. The resulting rod mesh geometry is finally used as input for an automatic fabrication process.

In this section, we first discuss our choice of design space. Next, we describe the general optimization framework. We use a Newton optimization method subject to boundary conditions, design constraints, and static equilibrium constraints, which are enforced implicitly. To conclude, we describe an optimization scheme to optimize both the material (i.e., radii) and rest-shape of the rod mesh.

4.1 Design Space

One of the main features of our approach is the choice of design space, which aims at minimizing fabrication complexity while maximizing design flexibility. There are three major design decisions in our approach that make this possible.

Rod Mesh Using a rod mesh to model a deformable object allows us to obtain different mechanical properties with a single material, simply by adjusting the radii of rod cross-sections. This decision simplifies the fabrication process.

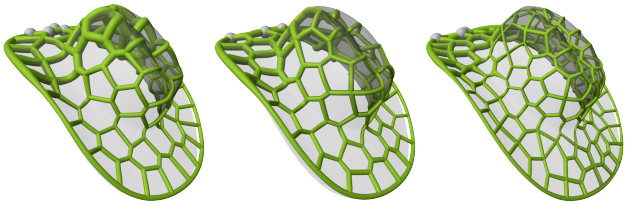

Figure 5: *A hat mesh is optimized to reach the target deformation (shown in light gray) with three different meshings. All three meshes were obtained through centroidal Voronoi tessellation; the first two with* 64 *cells and different initializations, and the third one with* 128 *cells.*

Hexagonal Mesh Topology We have opted for a mesh topology that minimizes structural stiffness, thus enabling a larger range of feasible designs under the same magnitude of external forces. Rods are radically more compliant in bending than stretch, hence the topology of the mesh should be one that enables deformations in all directions simply by bending rods. We have opted for a mesh topology dominated by hexagonal faces, as hexagons can deform in all directions by bending their edges. Fig. 4 compares the structural stiffness of a triangle-mesh, a quad-mesh, and a hexagon-mesh of rods. The hexagon-mesh can stretch and shear in all directions, the quad-mesh resists stretch in directions aligned with quad edges, and the triangle-mesh is almost inextensible.

To construct a mostly hexagonal mesh, we have explored two options in our examples. One is building the dual mesh of an input triangle mesh. The other one is to initialize a number of samples on a surface and then compute a centroidal Voronoi tessellation [Valette and Chassery 2004]. With hexagon meshes, we have observed that our optimization method is largely insensitive to the specific mesh topology. Fig. 5 shows an optimization result that is matched using three different meshes.

Parameterization Our design space consists of the rest-shape of the rod mesh and two orthogonal radii at each point of rod cross sections. By adjusting two orthogonal radii independently, we manage to control in-plane deformations of the surface (stretch and shear) and out-of-plane deformations (bending) independently.

Both for radii and rest positions, we set as design parameters control points along the rods. We place four control points per rod, two at the end connections and two evenly spaced, and smooth their values using cubic Hermite interpolation.

To further improve the visual appearance of the resulting rod meshes, we also enforce continuity and alignment of cross-section at connections. First, we share control points among all rods incident in a connection, such that radii magnitude smoothly varies throughout the rod mesh. Plus, we keep material and reference frames at connections aligned in the rest configuration. Whenever the rest-shape changes, we fit a plane to the incident edges of each connection. We use the normal of this plane as the normal axis of rest-shape frames of incident edges. This typically defines the first and the last rest frames of each rod in the mesh; the alignment of the other frames of the rod is linearly interpolated from the extremal values. We span the orthogonal cross-section radii at each point in the rod mesh using material frames. Thus, as an additional effect of frame alignment, we gain more flexibility during optimization; adjusting the two orthogonal radii is guaranteed to control tangent plane and out-of-plane deformations.

4.2 Numerical Optimization

Let us denote by $\mathbf{q}$ the vector of positions of all nodes in a rod mesh. Our optimization framework receives as input a set of N target poses $\{\bar{\mathbf{q}}_k\}$ in static equilibrium, and for each pose a subset of nodes $\hat{\mathbf{q}}_k = \mathbf{S}_k\,\bar{\mathbf{q}}_k$ specified as boundary conditions. Here $\mathbf{S}_k$ is a per-pose selection matrix. The method also supports using forces as boundary conditions.

We denote as $\mathbf{p}$ a generic set of P design parameters, which could be shape parameters (i.e., rest-shape coordinates of rod nodes) and/or material parameters (i.e., radii of rod cross-sections). We formulate the objective as the squared error between rod node positions and the input poses:

$$f(\mathbf{p}, \{\bar{\mathbf{q}}_k\}, \{\hat{\mathbf{q}}_k\}) = \frac{1}{2}\sum_{k=1}^{N} w_k \|\mathbf{q}_k(\mathbf{p}, \hat{\mathbf{q}}_k) - \bar{\mathbf{q}}_k\|^2. \qquad (7)$$

And we pose a constrained optimization that minimizes this objective function, subject to static equilibrium, box-constraints on design parameters and the input boundary conditions:

$$\mathbf{p} = \arg\min f(\mathbf{p}, \{\bar{\mathbf{q}}_k\}, \{\hat{\mathbf{q}}_k\}), \qquad (8)$$
$$\text{s.t. } \mathbf{F}_k(\mathbf{q}_k, \mathbf{p}, \hat{\mathbf{q}}_k) = \mathbf{0}, \forall k.$$
$$\mathbf{p}_m \leq \mathbf{p} \leq \mathbf{p}_M \qquad (9)$$

$\mathbf{F}_k$ denotes a vector that concatenates the forces on all rod nodes for the k^{th} pose, w_k is a scalar used to weight target poses and $\mathbf{p}_m$, $\mathbf{p}_M$ are minimum and maximum constraints on design parameters.

To solve the constrained optimization problem, we iterate updates of the design parameters using a quasi-newton method with constraint projection. In each update of the parameter vector we need to handle two types of constraints: (i) design constraints and (ii) static equilibrium constraints.

Our design constraints are maximum and minimum radii imposed by 3D printing limitations. These are easy to handle, as they constitute box constraints on the parameter space. We simply project the gradient for those radii that have reached a limit.

We enforce static equilibrium constraints implicitly. Similarly to others [Bickel et al. 2009; Umetani et al. 2011; Miguel et al. 2012], we apply the implicit function theorem and differentiate the equilibrium constraint for each pose. In this way we obtain the Jacobian of rod node positions w.r.t. design parameters:

$$\frac{\partial \mathbf{F}_k}{\partial \mathbf{p}} + \frac{\partial \mathbf{F}_k}{\partial \mathbf{q}_k}\frac{\partial \mathbf{q}_k}{\partial \mathbf{p}} = \mathbf{0} \;\Rightarrow\; \frac{\partial \mathbf{q}_k}{\partial \mathbf{p}} = -\frac{\partial \mathbf{F}_k}{\partial \mathbf{q}_k}^{-1}\frac{\partial \mathbf{F}_k}{\partial \mathbf{p}}. \qquad (10)$$

And as a result we reach a gradient of the objective function that locally satisfies static equilibrium constraints:

$$\frac{\partial f}{\partial \mathbf{p}} = \sum_{k=1}^{N} w_k\,(\mathbf{q}_k - \bar{\mathbf{q}}_k)\,\frac{\partial \mathbf{q}_k}{\partial \mathbf{p}}. \qquad (11)$$

In practice, we compute the force Jacobian $\partial \mathbf{F}_k / \partial \mathbf{q}_k$ analytically, but evaluate $\partial \mathbf{F}_k / \partial \mathbf{p}$ using finite differences to allow for arbitrary design parameters. After every update of the parameter vector $\mathbf{p}$, a static equilibrium problem must be solved on all poses to ensure that the constraints are enforced.

By handling equilibrium constraints implicitly, the minimization in (8) turns into a nonlinear least-squares problem. Computing the full Hessian of the problem is prohibitive, as it would require too many Jacobian estimations using finite differences. In addition, we have observed that the full Hessian occasionally becomes indefinite. In such cases, typical approximations (e.g., Gauss-Newton,

Levenberg-Marquardt, or BFGS) become ill-conditioned and bring the optimization to a halt. In practice, we have observed best performance using BFGS with line-search. When the new search direction fails to reduce the objective function, we switch to steepest descent. Thus, controlling the step-length is key to the performance of our method and so we enforce a maximum step-length which is adaptively refined depending on success.

Finally, to apply position boundary conditions $\{\hat{\mathbf{q}}_k\}$, we found that fixing simulation nodes increases excessively the error on free nodes. Instead, we enforce boundary conditions using a penalty force, and we progressively adjust the penalty stiffness such that the error at constrained nodes is similar to elsewhere in the mesh.

4.3 Optimization Scheme

As already mentioned, the vector of design parameters $\mathbf{p}$ may be formed by material parameters (i.e., rod cross-section radii) and/or shape parameters (i.e., rest-shape coordinates of rod nodes). We have observed that adjusting cross-section radii has the largest effect on the bulk fitting error, while adjusting rest-shape coordinates increases fitting quality for designs that are close to the optimum and have to deal with opposing objectives. For this reason, we propose the following optimization scheme.

We start with a multiresolution optimization of the material of the rod mesh. We first optimize only the radii control points at connections, linearly interpolated along rods; then we add radii control points in the middle of rods, with quadratic interpolation; and we finally optimize for all radii control points with cubic interpolation. In our examples, we did not find necessary to use higher order interpolations as obtained solutions were not any better. Once the material optimization alone has converged, we start iterating steps of material and shape optimization directly on all control points. After each rest-shape optimization step, we reinitialize the rest-shape material and reference frames. Please see Fig. 10 for examples of convergence with our optimization scheme.

5 Results

Design Pipeline In all our examples we follow a similar design pipeline, with small variations. We start with a surface description of the deformable object to be fabricated, typically a high-res triangle mesh $\mathcal{H}$. At this point, we define the connectivity of the rod mesh $\mathcal{R}$ to be optimized. If the user does not enforce a specific topology, we automatically compute one that is close to a hexagonal mesh using a centroidal Voronoi tessellation as described in Section 4.1. We project the resulting connections onto $\mathcal{H}$, and construct rods connecting them by following shortest geodesic paths. For each vertex of $\mathcal{R}$, we store a mapping to its projection triangle in $\mathcal{H}$.

To create the target poses, we apply some deformation to $\mathcal{H}$. In our examples, we have explored different deformations, mainly direct artist manipulation and embedding in a another physical model. Given a target deformation defined on $\mathcal{H}$, we can determine the target configuration of the rod mesh by its stored mapping.

Table 1 summarizes the rod mesh size, complexity, continuum material parameters and radii constraints for all our benchmarks. We have printed some of the obtained results to serve as a validation of both the simulation model and the optimization procedure. In all the cases, we have used laser sintering technology with the material TPU 92A.

Sheet Our first example consists of a simple sheet (Fig. 6), showing the capability of hexagonal rod meshes of exhibiting varied behaviors. The rod mesh is a square with 0.2m side, and consists of 71 rods and 54 connections forming a regular hexagonal pattern, with a total of 693 nodes.

Model	Sheet	Hat	Dino	Smiley
Nodes	693	1727	3219	2663
Edges	710	1790	3330	2892
Rods	71	179	333	723
Connections	54	116	222	494
Material Params.	392	948	1776	2664
rest-shape Params.	588	1422	2664	3663
Young mod. (MPa)	32	32	32	10.0
Poisson ratio	0.48	0.48	0.48	0.48
Density (Kg/m^3)	1200.0	1200.0	1200.0	1000.0
Size (m)	0.2	0.23	0.20	0.25
Max. Radius (mm)	5	3	3	4
Min. Radius (mm)	1	1	1	1
RMS Error (mm)	1.03	1.13	0.56	0.41
Computation Time	35min	1h45min	2h20min	3h10min

Table 1: *Statistics of benchmarks: model discretization, optimization complexity, mechanical parameters, design constraints, RMS error of the results and computation time.*

Four target poses are defined by computing the static equilibrium of the rod mesh for different radii configurations and boundary conditions (fixed points and weights). These targets are designed to be opposing: the first pair (first and second row) impose bending anisotropy; the second pair (third and forth row), impose stretching capabilities. The optimized rod mesh effectively captures all required behaviors.

The images in Fig. 6 show in semi-transparent gray the target configurations of the rod mesh. The default behavior at the beginning of the optimization is shown in red against the deformations obtained with our optimization framework in green. Cross-sectional radii optimization is capable of reducing the RMS error to 1.4756mm per node. We achieve an overall fitting of 1.0322mm upon convergence of rest-shape optimization.

Simulated results has been validated by testing physical printed realizations, which closely match the behavior predicted by the simulation. Although low frequency deformations are perfectly replicated, small differences appear which may be caused by inaccuracies in the validation process.

Hat Our second example is a hat of approximately 0.23m in length, shown in Fig. 7 and Fig. 1. The connectivity of the rod mesh is automatically computed using a centroidal Voronoi tesselation on a high-res triangle mesh. This results in an almost hexagonal mesh consisting of 179 rods, 116 connections, and a total of 1727 nodes.

Five target poses (four of them as rows in Fig. 7) are defined using a thin-shell model on the triangle mesh. We have designed several material distributions and used them to create target deformations. Target poses 1, 2 and 3, use a material for which the front side of the hat is much softer than the back side. Target pose 4 uses a material for which the left side of the hat is clearly softer than the right side. And additional pose was considered using a homogeneous material intended to make the hat maintain its rest-shape under gravity. We fix some points and pull from handles using forces to achieve interesting deformations. The deformed position of each handle is considered a boundary condition for our simulated rod mesh.

As shown in Fig. 7, our optimization framework is capable of combining all behaviors into a single hat. The first column shows in blue the target deformation of the rod mesh, according to the thin-

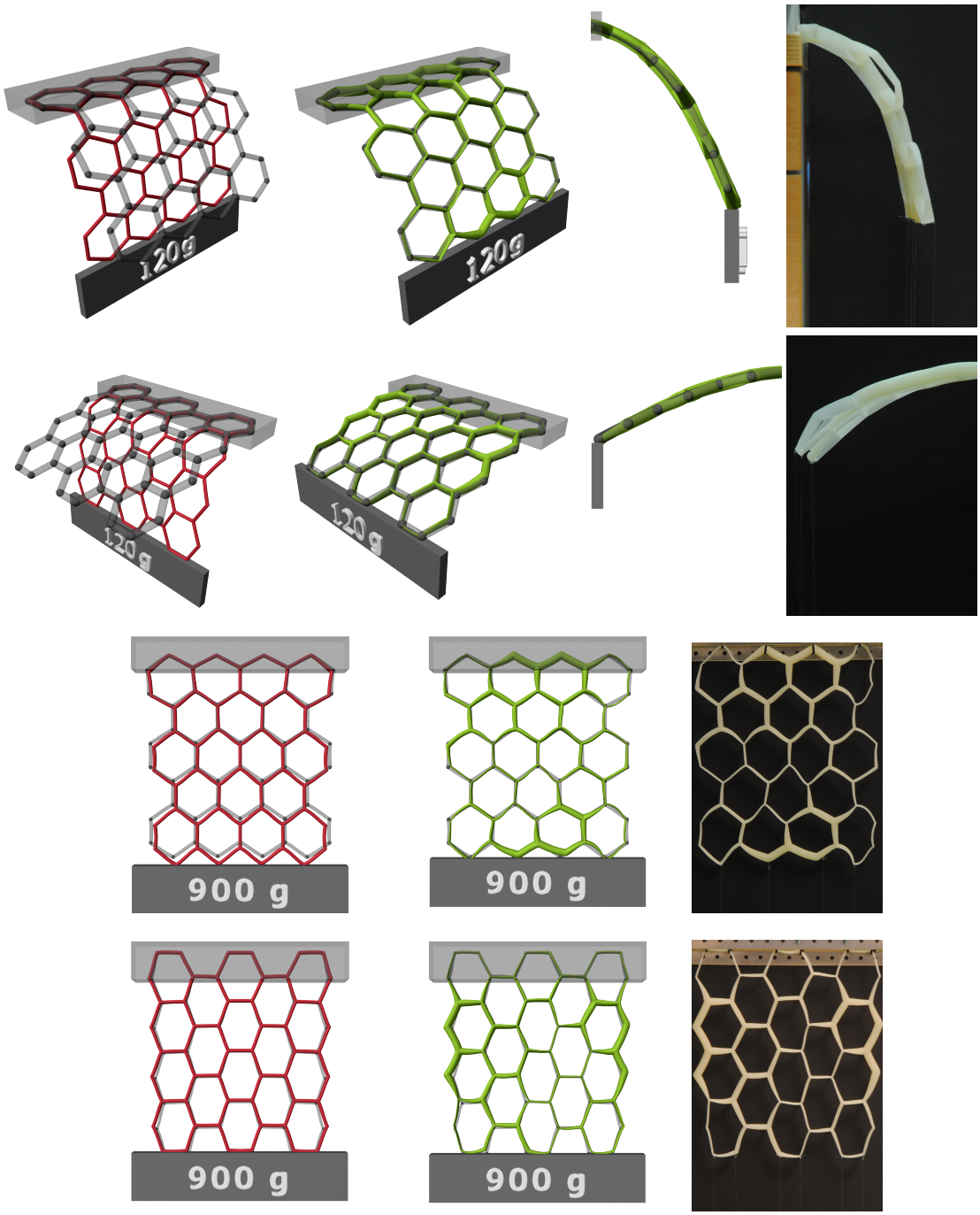

Figure 6: *Resulting deformations for each of the four target poses of the sheet demo. Default deformations (in red) are compared against the optimized results obtained using our framework (in green). Last column shows the physical printed sheet matching the behavior predicted by the simulation model. Target configurations of the rod mesh are shown in light transparent gray.*

shell model. The second column shows in red the deformation of the rod mesh with default parameters. The third column shows in green the final optimized mesh given by our method. We achieve an overall fitting RMS error of 1.3482mm per node, solely by optimizing cross-section radii. In this benchmark, optimizing the rest-shape does not improve the solution dramatically, reducing it down to 1.1128mm.

The images in Fig. 1 provide a visual validation of obtained results, for two of the target poses considered. Both default and optimized configurations of the rod mesh have been printed and fixed to a supporting structure. We tie wires to the constrained points in our simulated rod mesh and pull from them until approximately reaching their deformed position. The resulting deformation is visually appealing and resembles our model prediction for both default and optimal configurations.

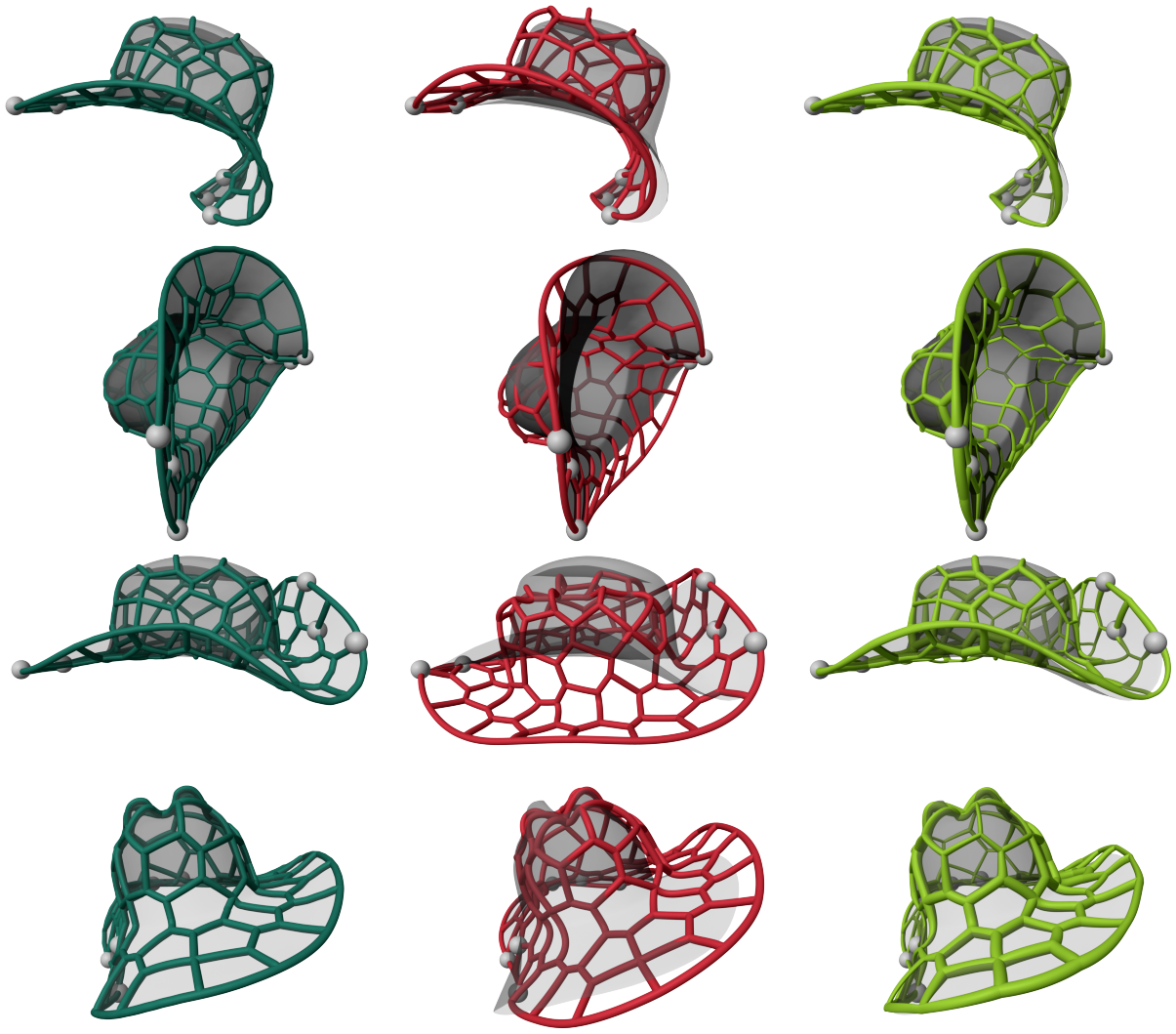

Figure 7: *Resulting deformations for four of the target poses of the hat demo (shown in blue). Default deformations (in red) are compared against the results obtained using optimization (in green). An overlay in transparent gray is added to help understand how far the solution is from the target deformation. Note that the rod mesh is capable of combining all behaviors into a single hat, even though target poses have been generated using different heterogeneous material distributions.*

Dinosaur This example consists of a dinosaur toy of approximately 0.21m in length, shown in Fig. 8. The connectivity of the rod mesh is semi-automatically computed from a high-res triangle mesh. We apply mesh decimation to obtain a low-res version $\mathcal{L}$, in the order of hundreds of triangles, and construct its dual mesh. That is, we place a rod connection at the centroid of each triangle in $\mathcal{L}$ and project it into $\mathcal{H}$. Then, we set a rod between two connections if their corresponding triangles in $\mathcal{L}$ share an edge. This results in a mesh consisting of 333 rods with 222 connections and 3219 nodes.

Five target poses are defined using a FEM deformation on $\mathcal{H}$. We have designed a heterogeneous FEM model, where the tail and the joints between the legs and the body are notably softer. The head is heavier and it should tip the dinosaur under gravity. To set boundary conditions during posing, we select a handle on the mesh and pull using forces. Using forces instead of translating the handle keeps the deformations plausible and easier to reproduce with real-world materials.

The images in Fig. 8 show both simulated and real results. Semi-transparent gray represents the target pose, as defined by the high-res mesh $\mathcal{H}$ deformation using FEM. The first column shows in red the behavior of the rod mesh for the default parameters. The second column shows in green the performance of our method. The third column in Fig. 8 shows the resulting deformation of the real printed dinosaur. For this demo, we follow the same validation methodology described above for the hat, obtaining similar results. Cross-section radii optimization achieves an already low fitting RMS error of 0.8622mm. Alternating rest-shape and full resolution material optimization is capable of further improving the solution to 0.5564mm.

Smiley Our last result (Fig. 9) is intended to demonstrate the performance of our method for an arbitrary user-defined target pose. For this demo, we apply a centroidal Voronoi tessellation to a high-res triangle mesh of a sheet of 0.25m. We use a serious smiley texture as a biasing weight for face areas. This creates a hexagonal mesh where cell distribution is determined by the color of the tex-

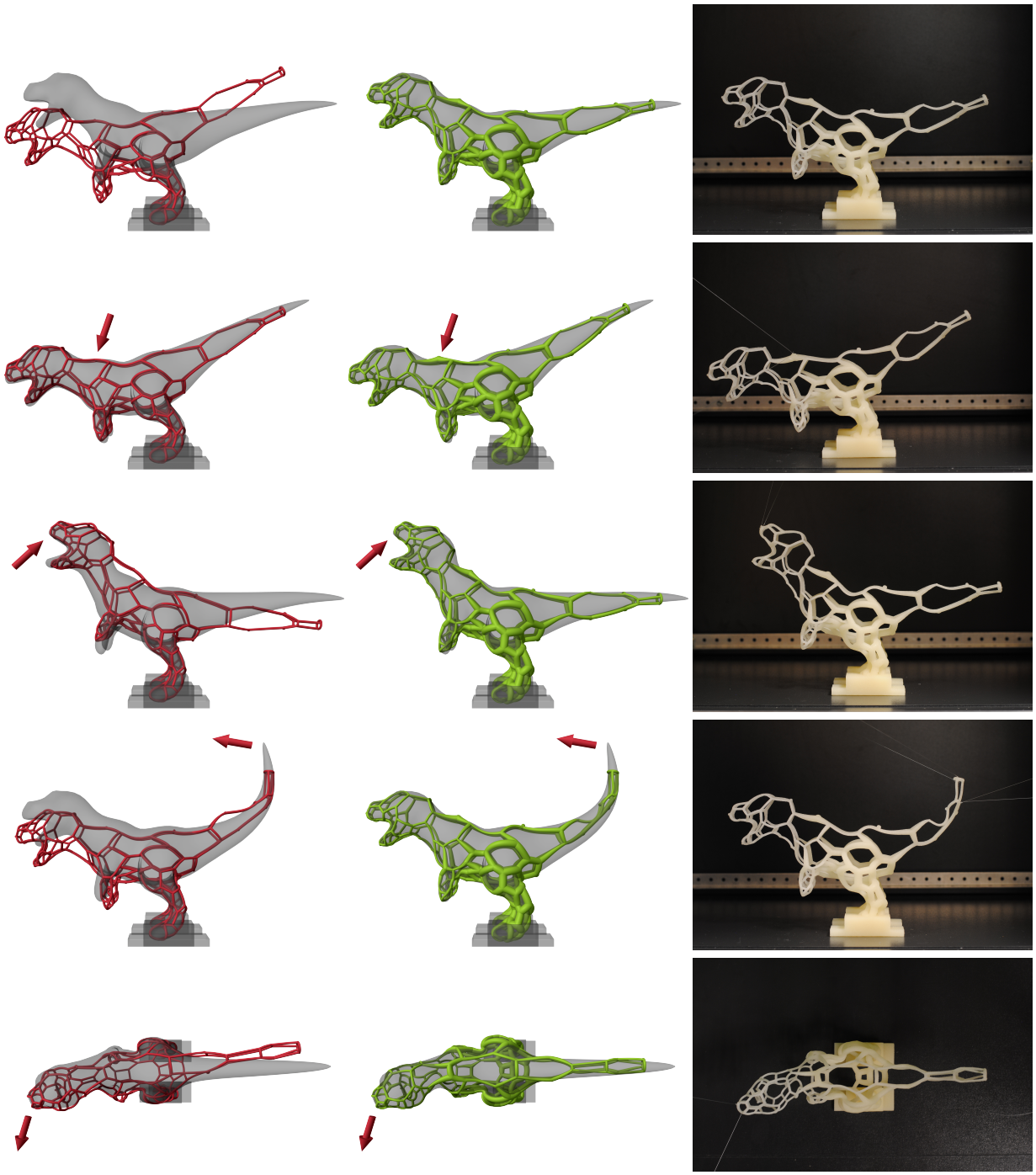

Figure 8: *Resulting deformations for each of the five target poses of the dinosaur demo. Default deformations (in red) are compared against optimized results (in green). An overlay in transparent gray represents the target surface generated by the FEM model. The right column shows our visual validation of obtained results, being pulled with wires from constrained points.*

ture. In particular, the density is incrementally higher at the face, mouth and eyes of the figure. The resulting rod mesh is formed by 723 rods, 494 connections, and a total of 2663 nodes.

The single target is manually defined by an artist, geometrically deforming the original texture. We map each point in the rod mesh with a texture coordinate and apply the same displacement to create the target mesh pose. Our intention is to make the character smile when stretched in one direction; this simple setup entails two main difficulties. First, for homogeneous radii throughout the mesh, variable rod densities tend to create heterogeneous deformations that might not match the target texture. Second, artist deformations are not based on any physical model and so the target pose might not be reachable at all by a mesh deformed using simple stretch forces.

The images in Fig. 9 compare default and optimized deformations to the warped texture. An overlay with the typical smiley colors is added to help understanding how the texture has been deformed

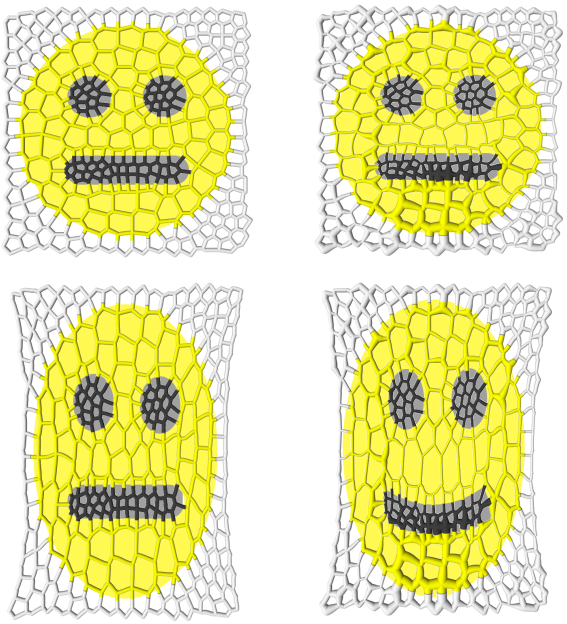

Figure 9: *This example shows a planar rod mesh that, when stretched, produces a smiley. In this example we combine the use of a mesh with an aesthetic design (i.e., higher rod density in the mouth and eyes) and an artist-painted target (i.e., the smiley drawn behind the meshes). On the left, we show the default mesh. When stretched, its deformation is nearly uniform. On the right, we show the optimized rod mesh. When stretched, it produces a smiley.*

in each case. Our optimization framework is capable of closely matching the smile with almost no rest-shape deformation close to the mouth, by placing thick rods below and thin rods above it. Plus, it approximates the overall shape of the deformed texture much better compared to the default configuration. We achieve an overall fitting RMS error of 0.4854mm.

Simulation Performance We integrate rod dynamics using a backward Euler integrator with Newton iteration and adaptive time-stepping. Our rod structure model introduces additional computational complexity to the discrete elastic rods approach. Computing the energy gradient and Hessian at connections is costly and involves a reduction in performance. However, we are aiming for sparse rod structures and so we can expect the overall complexity to grow linearly on the number of rods. The dynamic simulation is robust and remains stable using reasonable time steps, despite the existence of buckling effects.

Table 2 summarizes our performance evaluation of dynamic step solving for three of our examples. All tests have been performed using a desktop machine with an Intel i7 4470K processor and 16 GB DDR3-2133 RAM. Here, N denotes the total number of degrees of freedom, including both node positions and frame orientations; dt is the average time step used during dynamic relaxation, which is subject to adaptivity; t and T indicate the time required, on average, for a single dynamic step and a static solve respectively.

Optimization Performance Our optimization scheme considers implicit constraints for each target pose, which are independent from each other. The step is broken into a set of individual problems which can be solved separately, and the complexity grows linearly in the number of target poses. As a counterpart, two operations

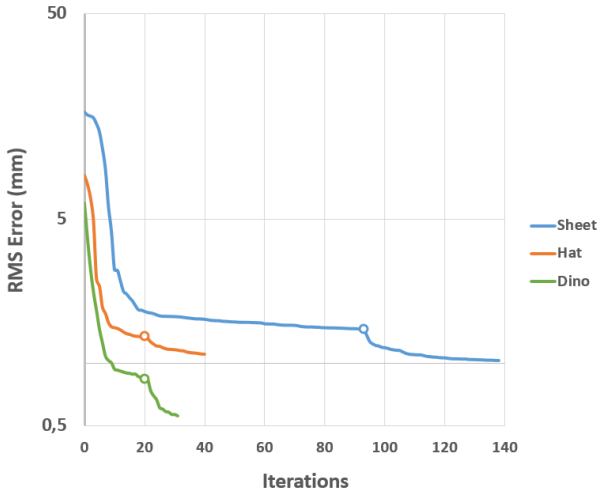

Figure 10: *Convergence of the optimization for the three printed results, showing the evolution of RMS error (in log scale). White dots indicate the iteration from which the optimization alternates updating radii and rest-shape parameters.*

are costly: performing the sensitivity analysis and computing static equilibriums. This has to be done, at least, once in a step per pose.

The cost of the sensitivity analysis is partially alleviated by incrementally adding resolution to the parametrization. As shown in Fig. 10, error is greatly reduced in the first steps of the optimization, when only one control point per connection is used. However, the improvement gained by subsequently refining the parametrization and iterating radii and rest-shape optimization steps is still significant. At this stage of the optimization, steps are more time-consuming so a trade-off must be considered between computation time and the quality of the solution.

The cost of computing the static equilibrium is highly dependent on the size of the parameter update, as each computation is initialized to the previous equilibrium configuration. If the parameter change is large, it is convenient to start the static solve from the rest configuration instead. Moreover, convergence problems may activate line-search bisections, leading to additional static solves. Adaptively controlling the maximum allowed update length, along with using the quasi-Newton update, has an important positive impact on performance. Table 1 shows an approximation of the time needed to converge to the optimal solution for each presented result.

6 Discussion and Future Work

We have described a computational design method that allows fabricating flexible objects with a desired deformation behavior. We opt to represent such flexible objects using meshes of deformable rods, and we show that a key component of our method is a dynamic simulation algorithm for rod meshes. Using this simulation algorithm, we have designed an optimization scheme that allows us to estimate the material and rest-shape of rods such that the result-

Model	N	dt (s)	t(ms)	T(s)
Hat	6971	0.0078	62.956	21.06
Dinosaur	12987	0.0138	108.78	19.321
Sphere	10076	0.0082	118.40	45.321

Table 2: *Summary of dynamic simulation performance.*

ing object approximates a set of input poses.

While there is no guarantee that the global optimum will always be found, we have observed that our alternating multiresolution scheme increases the chances of finding good solutions. Indeed, we did not encounter a case of objectionable solutions corresponding to local minima in the examples that we considered.

Our technique is currently limited on the range of deformation behaviors that it can replicate. Target poses should be physically plausible, and they should not differ much under similar boundary conditions. These limitations could be alleviated by providing the final user with tools to explore the feasible design space.

We have also seen the importance of selecting an adequate topology for the structure; one which does not impose a strong constraint on in-plane stretch. Both the quality of the results and the applicability of the method could be improved by incorporating topology optimization. This may require a new definition of the objective function to appropriately evaluate surface deviation under arbitrary rod sampling.

To conclude, flexible rod meshes might provide a supporting structure to be filled or dressed with other materials. Although we expect the structure to generally dominate the mechanical behavior, in some cases the filling or dressing material might affect this behavior. Capturing the coupling between different materials during the inverse design process constitutes also an interesting direction for future investigation.

Acknowledgements

The authors wish to thank Maurizio Nitti, Igor Trébol, Peter Kaufmann, Jorge López, Alvaro Pérez, and other team members at URJC and DRZ. This work was supported in part by grants from the Spanish Ministry of Economy (TIN2012-35840), and the European Research Council (ERC Starting Grant no. 280135 Animetrics).

References

BÄCHER, M., BICKEL, B., JAMES, D. L., AND PFISTER, H. 2012. Fabricating articulated characters from skinned meshes. *ACM Trans. Graph. 31*, 4, 47:1–47:9.

BACHER, M., WHITING, E., BICKEL, B., AND SORKINE-HORNUNG, O. 2014. Spin-it: Optimizing moment of inertia for spinnable objects. *ACM Trans. Graph. 33*, 4.

BARBIC, J., AND ZHAO, Y. 2011. Real-time large-deformation substructuring. *Proc. of ACM SIGGRAPH*.

BERGOU, M., WARDETZKY, M., ROBINSON, S., AUDOLY, B., AND GRINSPUN, E. 2008. Discrete elastic rods. *Proc. of ACM SIGGRAPH*.

BERGOU, M., AUDOLY, B., VOUGA, E., WARDETZKY, M., AND GRINSPUN, E. 2010. Discrete viscous threads. *Proc. of ACM SIGGRAPH*.

BERTAILS, F., AUDOLY, B., CANI, M.-P., QUERLEUX, B., LEROY, F., AND LÉVÊQUE, J.-L. 2006. Super-helices for predicting the dynamics of natural hair. In *ACM SIGGRAPH 2006 Papers*, ACM, New York, NY, USA, SIGGRAPH '06, 1180–1187.

BICKEL, B., BÄCHER, M., OTADUY, M. A., MATUSIK, W., PFISTER, H., AND GROSS, M. 2009. Capture and modeling of non-linear heterogeneous soft tissue. *ACM Trans. Graph. 28*, 3 (July), 89:1–89:9.

BICKEL, B., BÄCHER, M., OTADUY, M. A., LEE, H. R., PFISTER, H., GROSS, M., AND MATUSIK, W. 2010. Design and fabrication of materials with desired deformation behavior. *ACM Trans. Graph. 29*, 4, 63:1–63:10.

BICKEL, B., KAUFMANN, P., SKOURAS, M., THOMASZEWSKI, B., BRADLEY, D., BEELER, T., JACKSON, P., MARSCHNER, S., MATUSIK, W., AND GROSS, M. 2012. Physical face cloning. *ACM Trans. Graph. 31*, 4, 118:1–118:10.

CALÌ, J., CALIAN, D. A., AMATI, C., KLEINBERGER, R., STEED, A., KAUTZ, J., AND WEYRICH, T. 2012. 3d-printing of non-assembly, articulated models. *ACM Trans. Graph. 31*, 6 (Nov.), 130:1–130:8.

CASATI, R., AND BERTAILS-DESCOUBES, F. 2013. Super space clothoids. *ACM Trans. Graph. 32*, 4, 48:1–48:12.

CEYLAN, D., LI, W., MITRA, N. J., AGRAWALA, M., AND PAULY, M. 2013. Designing and fabricating mechanical automata from mocap sequences. *ACM Trans. Graph. 32*, 6, 186.

CHEN, D., LEVIN, D. I. W., DIDYK, P., SITTHI-AMORN, P., AND MATUSIK, W. 2013. Spec2fab: A reducer-tuner model for translating specifications to 3d prints. *ACM Trans. Graph. 32*, 4, 135:1–135:10.

CHEN, X., ZHENG, C., XU, W., AND ZHOU, K. 2014. An asymptotic numerical method for inverse elastic shape design. *ACM Trans. Graph. 33*, 3.

COROS, S., MARTIN, S., THOMASZEWSKI, B., SCHUMACHER, C., SUMNER, R., AND GROSS, M. 2012. Deformable objects alive! *ACM Trans. Graph. 31*, 4 (July), 69:1–69:9.

COROS, S., THOMASZEWSKI, B., NORIS, G., SUEDA, S., FORBERG, M., SUMNER, R. W., MATUSIK, W., AND BICKEL, B. 2013. Computational design of mechanical characters. *ACM Trans. Graph. 32*, 4, 83:1–83:12.

DEROUET-JOURDAN, A., BERTAILS-DESCOUBES, F., AND THOLLOT, J. 2010. Stable inverse dynamic curves. *ACM Trans. Graph. 29*, 6, 137:1–137:10.

DEROUET-JOURDAN, A., BERTAILS-DESCOUBES, F., DAVIET, G., AND THOLLOT, J. 2013. Inverse dynamic hair modeling with frictional contact. *ACM Trans. Graph. 32*, 6, 159:1–159:10.

GARG, A., SAGEMAN-FURMAS, A., DENG, B., YUE, Y., GRINSPUN, E., PAULY, M., AND WARDETZKY, M. 2014. Wire mesh design. *ACM Trans. Graph. 33*, 3.

HADAP, S. 2006. Oriented strands: Dynamics of stiff multi-body system. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SCA '06, 91–100.

IBEN, H., MEYER, M., PETROVIC, L., SOARES, O., ANDERSON, J., AND WITKIN, A. 2013. Artistic simulation of curly hair. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ACM, New York, NY, USA, SCA '13, 63–71.

KONDO, R., KANAI, T., AND ANJYO, K.-I. 2005. Directable animation of elastic objects. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '05)*, 127–134.

LI, S., HUANG, J., DE GOES, F., JIN, X., BAO, H., AND DESBRUN, M. 2014. Space-time editing of elastic motion through material optimization and reduction. *ACM Trans. Graph. 33*, 3.

MARTIN, S., THOMASZEWSKI, B., GRINSPUN, E., AND GROSS, M. 2011. Example-based elastic materials. *ACM Trans. Graph. 30*, 4 (July), 72:1–72:8.

MIGUEL, E., BRADLEY, D., THOMASZEWSKI, B., BICKEL, B., MATUSIK, W., OTADUY, M. A., AND MARSCHNER, S. 2012. Data-driven estimation of cloth simulation models. *Computer Graphics Forum (Proc. of Eurographics) 31*, 2 (may).

MÜLLER, M., HEIDELBERGER, B., TESCHNER, M., AND GROSS, M. 2005. Meshless deformations based on shape matching. *Proc. of ACM SIGGRAPH*, 471–478.

PAI, D. K. 2002. Strands: Interactive simulation of thin solids using cosserat models. *Proc. of Eurographics*.

PRÉVOST, R., WHITING, E., LEFEBVRE, S., AND SORKINE-HORNUNG, O. 2013. Make it stand: Balancing shapes for 3d fabrication. *ACM Trans. Graph. 32*, 4, 81:1–81:10.

ROSENBLUM, R. E., CARLSON, W. E., AND E., T. 1991. Simulating the structure and dynamics of human hair: modelling, rendering and animation. *J. Vis. and Comput. Anim. 2*.

SCHUMACHER, C., THOMASZEWSKI, B., COROS, S., MARTIN, S., SUMNER, R., AND GROSS, M. 2012. Efficient simulation of example-based materials. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SCA '12, 1–8.

SELLE, A., LENTINE, M., AND FEDKIW, R. 2008. A mass spring model for hair simulation. *ACM Trans. Graph. 27*, 3 (Aug.), 64:1–64:11.

SKOURAS, M., THOMASZEWSKI, B., BICKEL, B., AND GROSS, M. 2012. Computational design of rubber balloons. *Comp. Graph. Forum 31*, 835–844.

SKOURAS, M., THOMASZEWSKI, B., COROS, S., BICKEL, B., AND GROSS, M. 2013. Computational design of actuated deformable characters. *ACM Trans. Graph. 32*, 4, 82:1–82:10.

SONG, P., FU, C.-W., GOSWAMI, P., ZHENG, J., MITRA, N. J., AND COHEN-OR, D. 2013. Reciprocal frame structures made easy. *ACM Transactions on Graphics 32*, 4.

SPILLMANN, J., AND TESCHNER, M. 2007. Corde: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SCA '07, 63–72.

SPILLMANN, J., AND TESCHNER, M. 2009. Cosserat nets. *IEEE Transactions on Visualization and Computer Graphics 15*, 2, 325–338.

THOMASZEWSKI, B., COROS, S., GAUGE, D., MEGARO, V., GRINSPUN, E., AND GROSS, M. 2014. Computational design of linkage-based characters. *ACM Trans. Graph. 33*, 4.

TWIGG, C. D., AND KAČIĆ-ALESIĆ, Z. 2011. Optimization for sag-free simulations. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '11*, 225–236.

UMETANI, N., KAUFMAN, D. M., IGARASHI, T., AND GRINSPUN, E. 2011. Sensitive couture for interactive garment modeling and editing. *ACM Trans. Graph. 30*, 4, 90:1–90:12.

VALETTE, S., AND CHASSERY, J.-M. 2004. Approximated centroidal voronoi diagrams for uniform polygonal mesh coarsening. *Computer Graphics Forum 23*, 3, 381–389.

VIDIMČE, K., WANG, S.-P., RAGAN-KELLEY, J., AND MATUSIK, W. 2013. Openfab: A programmable pipeline for multi-material fabrication. *ACM Trans. Graph. 32*, 4, 136:1–136:12.

VOLINO, P., AND MAGNENAT-THALMANN, N. 2007. Stop-and-go cloth draping. *Vis. Comput. 23*, 9, 669–677.

WANG, W., WANG, T. Y., YANG, Z., LIU, L., TONG, X., TONG, W., DENG, J., CHEN, F., AND LIU, X. 2013. Cost-effective printing of 3d objects with skin-frame structures. *ACM Trans. Graph. 32*, 6, 177:1–177:10.

XU, H., LI, Y., CHEN, Y., AND BARBIC, J. 2015. Interactive material design using model reduction. *ACM Transactions of Graphics 34*, 18.

A Second Derivatives of Polar Decomposition

We follow a very similar derivation to the one by Barbic and Zhao [2011]. We denote with $\mathrm{sk}(\boldsymbol{\omega})$ the 3×3 skew-symmetric matrix of a vector $\boldsymbol{\omega} \in \mathcal{R}^3$, i.e., $\mathrm{sk}(\boldsymbol{\omega})\,\mathbf{x} = \boldsymbol{\omega} \times \mathbf{x}, \forall \mathbf{x} \in \mathcal{R}^3$. Similarly, we denote with $\mathrm{sk}^{-1}(\mathbf{A})$ the unique skew-vector $\boldsymbol{\omega} \in \mathcal{R}^3$, such that $\mathrm{sk}(\boldsymbol{\omega}) = \frac{1}{2}(\mathbf{A} - \mathbf{A}^T)$.

Given a polar decomposition $\mathbf{A} = \mathbf{R}\,\mathbf{S}$, its first derivative is:

$$\frac{\partial \mathbf{R}}{\partial u} = \mathrm{sk}(\boldsymbol{\omega}(u))\,\mathbf{R}, \quad \frac{\partial \mathbf{S}}{\partial u} = \mathbf{R}^T \left(\frac{\partial \mathbf{A}}{\partial u} - \frac{\partial \mathbf{R}}{\partial u}\mathbf{S} \right), \tag{12}$$

$$\text{with } \boldsymbol{\omega}(u) = 2\,\mathbf{G}^{-1}\,\mathrm{sk}^{-1}\left(\mathbf{R}^T \frac{\partial \mathbf{A}}{\partial u} \right),$$

$$\text{and } \mathbf{G} = (\mathrm{tr}(\mathbf{S})\mathbf{I} - \mathbf{S})\mathbf{R}^T.$$

The mixed second derivative of the rotation matrix is:

$$\frac{\partial^2 \mathbf{R}}{\partial u \partial v} = \mathrm{sk}\left(\frac{\partial \boldsymbol{\omega}(u)}{\partial v} \right) \mathbf{R} + \mathrm{sk}(\boldsymbol{\omega}(u))\,\mathrm{sk}(\boldsymbol{\omega}(v))\,\mathbf{R}, \tag{13}$$

$$\text{with } \frac{\partial \boldsymbol{\omega}(u)}{\partial v} = 2\,\mathbf{G}^{-1}\,\mathrm{sk}^{-1}\left(\mathbf{R}^T \left(\frac{\partial^2 \mathbf{A}}{\partial u \partial v} - \mathrm{sk}(\boldsymbol{\omega}(v)) \frac{\partial \mathbf{A}}{\partial u} \right) \right)$$

$$- \mathbf{G}^{-1} \left(\mathrm{tr}\left(\frac{\partial \mathbf{S}}{\partial v} \right) \mathbf{I} - \frac{\partial \mathbf{S}}{\partial v} \right) \mathbf{R}^T \boldsymbol{\omega}(u) + \mathrm{sk}(\boldsymbol{\omega}(v))\,\boldsymbol{\omega}(u).$$