

Computational Inverse Design of Surface-based Inflatables

JULIAN PANETTA, UC Davis, USA

FLORIN ISVORANU, EPFL, Switzerland

TIAN CHEN, EPFL, Switzerland

EMMANUEL SIÉFERT, Hebrew University of Jerusalem, Israel

BENOÎT ROMAN, PMMH (CNRS, ESPCI Paris, Université PSL, Sorbonne Université, Université de Paris), France

MARK PAULY, EPFL, Switzerland

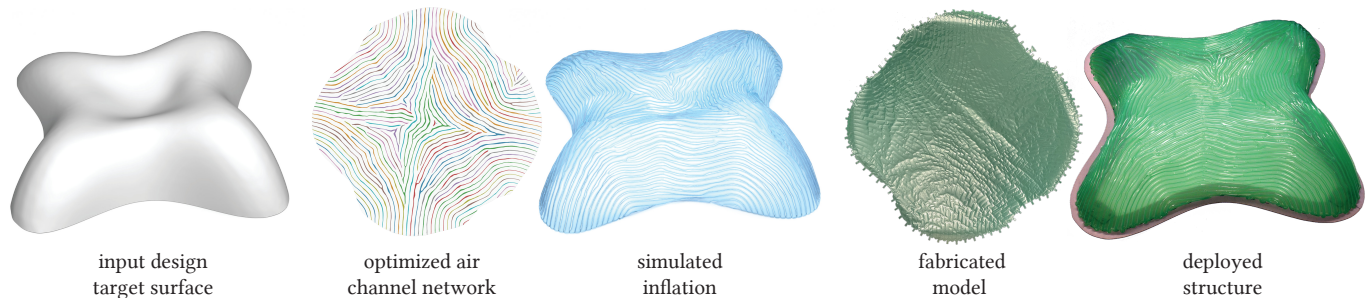


Fig. 1. Inflatables are composed of two thin sheets of inextensible material, fused together along carefully selected curves to form a network of channels that can be inflated with air. Our method optimizes the layout of these curves so that the inflated structure best approximates the desired input shape.

We present a computational inverse design method for a new class of surface-based inflatable structure. Our deployable structures are fabricated by fusing together two layers of inextensible sheet material along carefully selected curves. The fusing curves form a network of tubular channels that can be inflated with air or other fluids. When fully inflated, the initially flat surface assumes a programmed double-curved shape and becomes stiff and load-bearing. We present a method that solves for the layout of air channels that, when inflated, best approximate a given input design. For this purpose, we integrate a forward simulation method for inflation with a gradient-based optimization algorithm that continuously adapts the geometry of the air channels to improve the design objectives. To initialize this non-linear optimization, we propose a novel surface flattening algorithm. When a channel is inflated, it approximately maintains its length, but contracts transversally to its main direction. Our algorithm approximates this deformation behavior by computing a mapping from the 3D design surface to the plane that allows for anisotropic metric scaling within the bounds realizable by the physical system. We show a wide variety of inflatable designs and fabricate several prototypes to validate our approach and highlight potential applications.

CCS Concepts: • **Computing methodologies** → **Shape Modeling; Simulation**.

Authors' addresses: Julian Panetta, UC Davis, julian.panetta@gmail.com; Florin Isvoranu, EPFL, florin.isvoranu@epfl.ch; Tian Chen, EPFL, tian.chen@epfl.ch; Emmanuel Siéfert, Hebrew University of Jerusalem, emmanuel.siefert@gmail.com; Benoît Roman, ESPCI, benoit.roman@espci.fr; Mark Pauly, EPFL, mark.pauly@epfl.ch.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM. 0730-0301/2021/8-ART40 \$15.00

<https://doi.org/10.1145/3450626.3459789>

Additional Key Words and Phrases: deployable structures, simulation, numerical optimization, computational design, fabrication

ACM Reference Format:

Julian Panetta, Florin Isvoranu, Tian Chen, Emmanuel Siéfert, Benoît Roman, and Mark Pauly. 2021. Computational Inverse Design of Surface-based Inflatables. *ACM Trans. Graph.* 40, 4, Article 40 (August 2021), 14 pages. <https://doi.org/10.1145/3450626.3459789>

1 INTRODUCTION

Shape-morphing structures are physical systems that can transition (morph) from one geometric state to another. They find applications in diverse fields, for example as deployable solar panels for satellites, medical implants, morphable air foils, smart textiles, soft robotics, building instrumentation, or tissue engineering; see [Bertoldi et al. 2017; Zhang et al. 2017] for recent surveys. The shape-morphing deformation can be initiated by external forces, for example, a sheet of paper folded into an elaborate origami shape [Demaine and O'Rourke 2008], or triggered by an internal actuation mechanism like inflation [Siéfert et al. 2019a] or swelling [Gladman et al. 2016].

In this paper, we present computational design methods for a recently proposed shape-morphing surface structure [Siéfert et al. 2020], subsequently called *inflatables*. The physical system is composed of a double-layer membrane that deforms from a flat fabrication state to a double-curved target shape when actuated via inflation (see Figure 1). For this purpose, the surface is covered with a dense network of tubular channels that induce an anisotropic contraction of the surface sheet when inflated. By prescribing the orientation and magnitude of this contraction, we can program the local deformation of the material and thus control its global shape.

Key benefits of inflatables are: (i) Ease of fabrication: fusing two sheets of inextensible material is simple, efficient, and can be applied across a large range of scales; (ii) Ease of deployment: inflatables can be deployed easily from a compact flat state without requiring temporary supports, and the structure can be inflated and deflated many times; (iii) Structural performance: inflatables are lightweight and become stiff when fully actuated, without requiring the volume bounded by the surface to be pressurized; (iv) Rich shape space: as our results demonstrate, a wide variety of shapes can be deployed automatically from the flat fabrication state.

Contributions. The key challenge we address in this paper is the *inverse* design problem for inflatables: How can we find the optimal network of air channels such that the deployed surface best approximates the target design surface? This is a challenging problem, because the deployed state needs to satisfy a static equilibrium condition, which requires running a full physical deployment simulation in the inner loop of the optimization. To solve this problem, we make the following core technical contributions:

- We present a numerical optimization algorithm that continuously modifies air channel boundary curves to minimize the distance to the target design. This optimization integrates a forward simulation method to predict the deployed shape of an inflatable.
- We introduce a surface flattening algorithm and curve tracing method to find an initial layout of air channels that is suitable as a starting point for the optimization. This approach abstracts the deformation of an air channel to a metric distortion of a planar surface sheet and integrates additional objective terms to account for fabrication and deployment constraints.

In combination, these two methods provide a complete solution for inverse design that directly produces the fabrication instructions as a set of air channel boundary curves. We simulate and fabricate a number of prototypes using a custom-built device that essentially works like 2D plotter, fusing surface sheets along the drawn curves by laser fusing (Figure 2).

Overview. The rest of the paper is organized as follows. In Section 2 we discuss related previous work. Section 3 explains the key physical principle of the inflatable material system, i.e., how the inflation of tubular channels can be exploited to deploy flat sheets into double-curved surfaces. Section 4 provides a high-level overview of our inverse design algorithm that is composed of two parts corresponding to our two main contributions: a method to compute an initial layout of air channels (Section 5) and an optimization algorithm to improve the layout (Section 6). In these sections, we focus on the conceptual and mathematical formulation of our algorithms and delay the explanation of specific features of our numerical methods as well as implementation details to Section 7 and the supplemental material. We show results in Section 8, including several scanned prototypes to validate our method, before concluding with a discussion of limitations and ideas for future work.

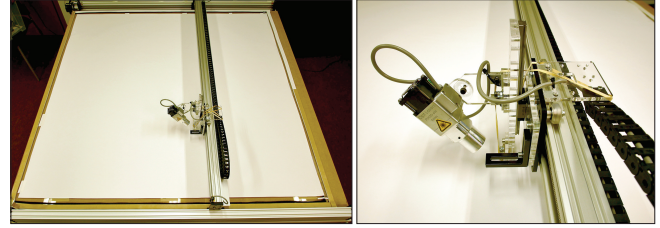


Fig. 2. Fabrication of an inflatable with a computer-controlled 2D plotter. A laser source mounted on gantry carriage (right) fuses two airtight inextensible fabrics along the computed air channel curves.

2 RELATED WORK

A wide variety of fabrication-aware computational design methods have recently been presented in the graphics community. We refer to recent surveys [Attene et al. 2018; Bermano et al. 2017; Bickel et al. 2018] for a general overview. In our discussion, we focus on previous work most related to the key aspects of our work. In particular, we discuss representative work on material systems that allow shaping curved surfaces from planar material, actuated materials that deploy automatically to a desired target surface, and methods specifically using inflation for deployment. Throughout our discussion, we highlight methods that provide inverse design solutions.

From 2D material to 3D surfaces. A classical method to obtain 3D surfaces from planar material is origami, the art of folding a piece of paper into a desired shape without cutting or gluing [Demaine and O’Rourke 2008]. Tachi [2009] presented an inverse design method for general origami surfaces that leverages isometry and folding abstractions to derive suitable constraints for valid origami crease patterns.

Malomo and co-workers [2018] introduced *FlexMaps*, an inverse design method for approximating a given 3D shape by an assembly of initially flat panels tiled with parametric spiral patterns. This method decomposes the target surface into a number of patches that can be flattened with low distortion and then tunes the resulting panels’ stiffnesses by optimizing spiral parameters so that the assembly achieves the desired equilibrium shape. Konaković and colleagues [2016] present an optimization method for planar auxetic materials that employs conformal geometry to solve the inverse design problem. The method of Garg and co-workers [2014] combines optimization with user interaction to lay out a regular patch of wiremesh material on a 3D surface based on results from Chebyshev net theory. Sageman-Furnas and co-workers [2019] extended this work to also handle singularities. In these examples, the 3D surface state has to be manually assembled or deformed using a guide surface.

Actuated shape-morphing surfaces. Numerous papers investigate automated shape deformation using various actuation systems. We discuss a few representative examples below.

Actuated origami for soft robotics has been investigated by Tolley and colleagues [2014], who drive the folding process by activating shape memory composites with heat. They focus on the physical

mechanism of folding and present several basic shapes fabricated in this way, but do not investigate the general inverse design problem. Tachi and co-workers [2012] showed how rigid origami structures can be stiffened by applying a vacuum once deployed to the target state.

Kilian and co-workers extended earlier work on curved foldings [2008] to incorporate force actuation by a sparse set of strings [Kilian et al. 2017]. Curved foldings are developable, so they can be flattened into the plane without distortion. As a consequence, the resulting shape space is rather limited, and few applications beyond artistic designs currently exist.

A shape-morphing surface structure based on kirigami has been proposed in [Celli et al. 2018]. They show how pulling on certain boundary points induces out-of-plane buckling of an elastic sheet with a carefully designed pattern of cuts. A variety of 3D shapes has been experimentally validated, but no solution for inverse design is proposed. Panetta and co-workers [2019] introduced *X-Shells*, a non-uniform linkage of elastic beams that can be deployed from a flat assembly state to a double-curved target state via torque actuation at the linkage joints. A gradient-based method supports local optimization of a given design, but finding a suitable initialization for a given input surface currently remains an open problem. Pillwein and colleagues [2020] solve the inverse problem for a subclass of deployable geodesic gridshells. They leverage insights from differential geometry to implement a purely geometric approach for inverse design that does not rely on numerical shape optimization.

Gladman and colleagues [2016] drew inspiration from botanical systems to investigate composite hydrogel architectures. These materials can be programmed by aligning and embedding cellulose fibers to exhibit locally anisotropic swelling. They studied how to combine simple shapes into more complex structures based on curvature analysis as a first step towards an inverse design solution.

Shi and colleagues [2017] study the assembly of origami-inspired structures by attaching a patterned 2D sheet material onto a pre-stretched membrane. When the membrane is released, the contractive forces cause the material to buckle out of plane. The obtained 3D shape can then be controlled by introducing folding joints to create an origami structure. While many different actuated shapes are demonstrated, a comprehensive inverse design method is still lacking for this method. Actuation by contraction of an elastic sheet has also been applied in [Guseinov et al. 2017] and [Pérez et al. 2017], who embed 3D-printed elements in a pre-stretched membrane that contracts to a prescribed 3D form when released. Inverse design is facilitated by optimization.

Siéfert and colleagues [2019a] introduced *baromorphs*, thin elastomer sheets that are equipped with air channels so that applied pressure changes deform the sheet into double-curved 3D shapes. They show how varying the density and arrangement of the air channels controls the deformation of the material and present a strategy for inverse design suitable for a limited set of heightfield geometries. They observe how actuation effectively leads to an anisotropic change in the surface metric, which in turn controls the Gaussian curvature of the surface. This connection is also studied in [Aharoni et al. 2014] and has inspired an inverse design algorithm presented in [Aharoni et al. 2018] that tries to find a suitable anisotropic metric embedding for a given 3D surface by solving a least-squares

optimization problem. We compare our parametrization algorithm to this method in Section 5.

Metric analysis has also been used in [van Rees et al. 2017], where an inverse process is described for elastic bilayer structures that expand anisotropically. Multimaterial 4D printing is employed in [Boley et al. 2019], who design structured lattices that morph to a desired target shape under thermal expansion. [Konaković-Luković et al. 2018a,b] work with varying scale auxetic materials and rely on inflation and gravitational loading to deform a flat sheet towards the desired 3D configuration. This approach approximates material deformation by a conformal map, thus capturing locally isotropic material expansion only.

Inflatable Structures. Research on inflatable structures spans numerous disciplines, including HCI [Yao et al. 2013], soft robotics [Mosadegh et al. 2014], materials science, and mechanical engineering.

Ou and co-workers [2016] introduced an inflatable surface structure called *areoMorphs*. Similar to our work, they use heat sealing to connect two sheets of material that are then inflated to morph to a 3D surface. Their method is based on a general parameterized hinge mechanism that can be used to induce bending. By combining multiple such hinge elements, more complex shape transformations can be created. They demonstrate a variety of shapes based on a simulation-assisted forward design process but do not address inverse design.

Ma and co-workers [2017] propose a method for designing and fabricating soft pneumatic objects that deform to a given set of target poses. Their approach splits the object into a set of chambers with embedded frame structures, each individually controlled with pneumatic actuators. Skouras and colleagues [2012] proposed a shape optimization approach for rubber balloons. Their method simulates the inflation of an elastic material and solves the inverse problem of finding the rest shape of a balloon that approximates a given target shape when inflated. [Skouras et al. 2014] presented an interactive system for designing volumetric inflatables composed of patches of approximately inextensible material. They develop an optimization approach to find the 2D layout of panels that, when joined together by seams, deploy to the desired target shape. The crucial difference to our work is that we deploy a surface structure: air does not fill the volume enclosed by our designs, but only a thin surface boundary layer defined by the air channels.

Our work builds upon recent studies by Siéfert and colleagues [2019b; 2020]. They introduce the concept of inflatables as networks of air channels, present a study of the mechanical behavior of basic shapes, and propose some elementary forward design tools. We focus on geometric and computational aspects for inverse design of inflatables and present physics-based optimization methods suitable for general free-form surfaces.

3 MATERIAL SYSTEM

An inflatable is fabricated by fusing two airtight sheets of material along carefully selected paths to form a network of channels that can be inflated with air (see Figure 2). We assume the sheet material to be approximately inextensible and exhibit little resistance to bending.

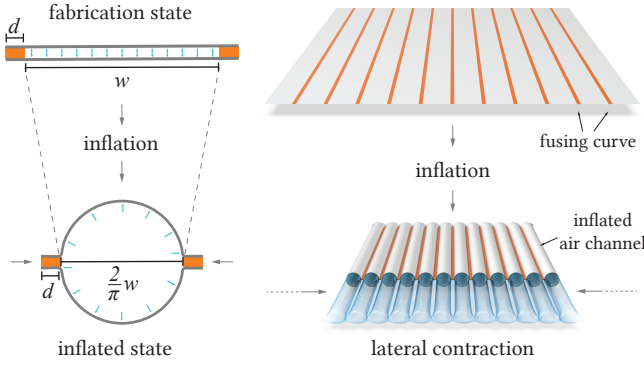


Fig. 3. Air channels maximize volume when inflated by assuming a circular cross section. As a consequence, the surface contracts laterally. For inextensible sheet material, the width of a channel contracts by a factor of approximately $2/\pi$.

3.1 Inflating a tubular channel

To understand how inflatables are deployed, let us first consider a single air channel as sketched in Figure 3, left. The two planar sheets have been fused together along two straight parallel paths of width d , leaving a channel of unfused material of width w . We assume that the channel is significantly longer than it is wide and that its ends have been sealed (apart from a small inlet for inflation).

As the channel is inflated, pressure forces push the two sheets apart to maximize the internal volume. When fully inflated, the channel assumes an approximately cylindrical form. This means that the channel width contracts from w to approximately $2w/\pi$. If we incorporate the width of the seam, which does not contract, we get a relative contraction of $\frac{2w/\pi+d}{w+d}$ across the width of the channel, while the length stays approximately constant. We refer to the work of [Siéfert et al. 2019b, 2020] for a detailed study of the mechanical properties of inflatable tubular structures and an experimental validation of their contractive deformation behavior.

3.2 A network of channels

If we now distribute a network of curved air channels across a surface as shown in Figure 1, more complex deformations can be observed when inflating the structure. Here we assume that the curve network is designed so that all channels are connected with a single inlet at the boundary where air is pumped into the network of channels to deploy the structure.

Each air channel tries to assume a cylindrical configuration locally, i.e. contract orthogonally to its main direction. This behavior can be abstracted as an anisotropic in-plane contraction. Effectively, inflation modifies the surface metric, which in turn determines Gaussian curvature. If the contraction varies across the surface in direction and/or magnitude, the surface assumes non-zero Gaussian curvature, i.e. is forced into a double-curved configuration. We can therefore control the deployed shape by manipulating the local direction of air channels and varying the amount of contraction locally for each air channel.

Varying contraction. Since the parts of the surface that have been fused together do not contract, we can effectively control the maximal contraction by adapting the width of the fusing curves. In theory, the local contraction factor could be between 1 (no contraction, sheets fully fused) and $\frac{2w/\pi+d_{\min}}{w+d_{\min}}$, where d_{\min} is the minimal width of a fusing curve imposed by the particular fabrication method. Specifically, our fabrication method employs a laser that generates a fusing curve of width $d_{\min} = 1$ mm. To increase the width of the fusing curve to some value $d > d_{\min}$, we simply trace two paths with a spacing of $d - d_{\min}$. We therefore represent the 2D layout of air channels by a network of curves that define the boundaries of the fused regions of the surface, allowing continuously varying channel widths to program local anisotropic contraction factors. For a more regular inflated surface appearance, our implementation currently targets designs with roughly evenly spaced fused regions and uniform tube sizes. Accordingly, we choose $w + d$ to be constant throughout the sheet and additionally bound d from above to avoid narrow tubes; this translates into upper and lower bounds on the contraction factors that are enforced by our flattening algorithm.

Discretization. For inflation simulation and design optimization, we employ a discrete model based on piecewise linear triangle elements. The boundaries of the varying-width fusing curves are sampled equidistantly, and the fused and unfused regions enclosed by these polylines are discretized using the constrained Delaunay meshing algorithm of Triangle [Shewchuk 1996] (see also Figure 4). Two copies of this planar mesh are super-imposed, and the corresponding vertices of the fusing curves are identified; we refer to the non-manifold triangle mesh obtained by gluing together these two copies (with opposite orientations) as S_0 . In practice, our implementation does not perform this duplication: it stores only the top sheet mesh, duplicates deformation variables for the tube vertices, and visits all triangles twice to evaluate the integral in (1).

3.3 Inflation Simulation

In order to compare the deployed surface with the target, we need to accurately simulate the inflation process to predict its shape. We perform a quasi-static simulation of the sheet's inflation by minimizing the total potential energy of the system, solving for the equilibrium deformation $\Phi_{S_0}^* : S_0 \rightarrow \mathbb{R}^3$ that minimizes the sum of the elastic energy stored in the sheet and the internal energy stored in the pressurized channels [Feynman et al. 1963],

$$\Phi_{S_0}^* := \operatorname{argmin}_{\Phi} \int_{S_0} \psi(\nabla\Phi) \, dA - p \operatorname{Vol}(S_{\text{tube}}). \quad (1)$$

Here S_{tube} is the union of the *deformed* triangles forming the boundary of the tubular air channels, Φ is the deformation mapping from S_0 to the inflated configuration, p is the pressure, and $\operatorname{Vol}(\cdot)$ denotes the volume enclosed by a surface. Since we discretize our sheet with linear triangle elements, the pressure term can be computed with the following particularly simple formula:

$$p \operatorname{Vol}(S_{\text{tube}}) = \frac{p}{6} \sum_{T \in S_{\text{tube}}} \det([\mathbf{x}_1 \quad \mathbf{x}_2 \quad \mathbf{x}_3]),$$

where $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ are triangle T 's deformed corner positions.

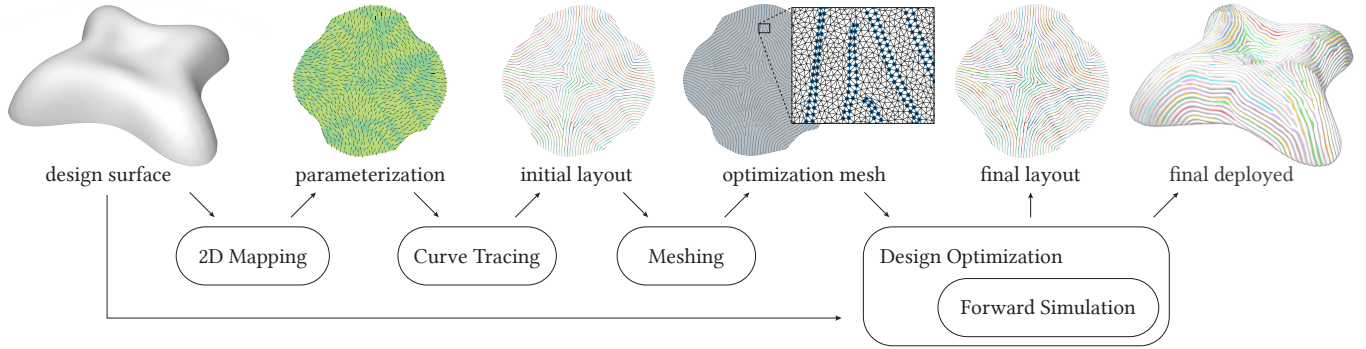


Fig. 4. Overview of the computational pipeline for inverse design of inflatables. Colors in the layouts indicate connected components of the air channel boundary curves. Blue dots in the zoom highlight the air channel wall vertices that are the unknowns in the design optimization.

For the elastic energy density function ψ , we use an incompressible neo-Hookean sheet material with a tension field theory approximation as proposed in [Skouras et al. 2014]:

$$\psi(F) = \frac{Yh}{6} \begin{cases} 0 & \sigma_0 \leq 1, \sigma_1 \leq 1 \\ \sigma_0^2 + \frac{2}{\sigma_0} - 3 & \sigma_0 \geq 1, \sigma_1 \leq \frac{1}{\sqrt{\sigma_0}} \\ F : F + \frac{1}{\det(F^T F)} - 3 & \text{otherwise,} \end{cases}$$

where Y is the Young's modulus of the material, h is the sheet thickness, and $\sigma_0 \geq \sigma_1$ are the singular values of $F \in \mathbb{R}^{3 \times 2}$. This tension field theory approximation neglects elastic forces arising from compression (capturing the tendency of the sheet to wrinkle rather than resist compression) and results in a convex elastic energy, which eases our equilibrium solve. However, the full potential energy function of the system is still non-convex due to the pressure term, and our structure frequently encounters unstable equilibria during deployment (e.g., as it breaks symmetry to pop into 3D). We use a Newton optimizer with a Hessian modification to efficiently and robustly minimize the potential energy as detailed in Section 7.

4 ALGORITHM OVERVIEW

Our inverse design method expects as input the desired deployed target surface \mathcal{M} . We assume \mathcal{M} is represented as a manifold triangle mesh, possibly with cuts so it can be flattened into the plane. The goal of our algorithm is to compute a network of air channels such that the inflated surface is in static equilibrium and conforms as closely as possible to \mathcal{M} .

Figure 4 provides an overview of our computational pipeline. Given the design surface, we first optimize for a mapping into the plane that respects the intrinsic anisotropic distortion induced when inflating air channels. We then compute an initial layout by tracing fusing paths in the 2D domain so that the resulting network of air channels respects the direction and magnitude of contraction of the mapping as closely as possible. Given this initialization, we further optimize the channel layout to improve the design objectives. In particular, we adapt the 2D locations of the vertices defining the fusing curves to best fit the inflated equilibrium surface to the target. For this purpose, the optimization runs a forward simulation of the inflation process (Section 3.3) in its inner loop.

5 FLATTENING OPTIMIZATION

To initialize our design optimization, we seek a parametrization $f : \mathcal{M} \rightarrow \mathbb{R}^2$ mapping the target surface $\mathcal{M} \subset \mathbb{R}^3$ to the plane that is compatible with the air channel inflation mechanism. Specifically, we analyze the anisotropic stretching of f in terms of the singular value decomposition of its Jacobian at each point on the surface:

$$\nabla f = U \begin{pmatrix} \sigma_0 & 0 & 0 \\ 0 & \sigma_1 & 0 \end{pmatrix} V^T, \quad (2)$$

where $U \in \mathbb{R}^{2 \times 2}$ is an orthogonal matrix whose two columns give the directions in the plane along which lengths are stretched or preserved, respectively. The singular values $\sigma_0 \geq \sigma_1$ quantify the amount of stretching along these directions, while $V \in \mathbb{R}^{3 \times 3}$ indicates the directions in 3D space that are stretched by the mapping.

The discussion in Section 3 revealed that air channels essentially do not contract along their main direction, i.e., f must have $\sigma_1 = 1$. Furthermore, the transversal contraction is constrained, meaning its *reciprocal* σ_0 should be in some admissible stretching range $[\sigma_{\min}, \sigma_{\max}]$ that depends on the minimal fusing width d_{\min} and the desired tube diameters as explained in Section 3.2. Mappings of this form were already constructed by Aharoni et al. [2018], who used an iterative local-global optimization to fit ∇f to the desired form by minimizing the energy:

$$E_{\text{fit}}[f] := \min_{U, V, \sigma} \frac{1}{2} \int_{\mathcal{M}} \left\| \nabla f - U \begin{pmatrix} \sigma & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} V^T \right\|_F^2 dA. \quad (3)$$

Their approach alternates between (i) computing a closed-form solution to the minimization over U, V and σ for fixed f by computing the SVD of the current Jacobian ∇f for each triangle and setting σ to σ_0 clamped to the admissible range; and (ii) solving for the optimal mapping f while holding the newly computed U, V, σ fixed. Running this method from a reasonable initialization quickly reaches a parametrization f with singular values close to the admissible range (provided such a mapping exists).

Unfortunately, this approach results in noisy channel directions and stretch factors, as shown in Figure 5 top. To successfully trace air channels on the sheet material, we require smoother directions and scale factor distributions without discontinuities. We address

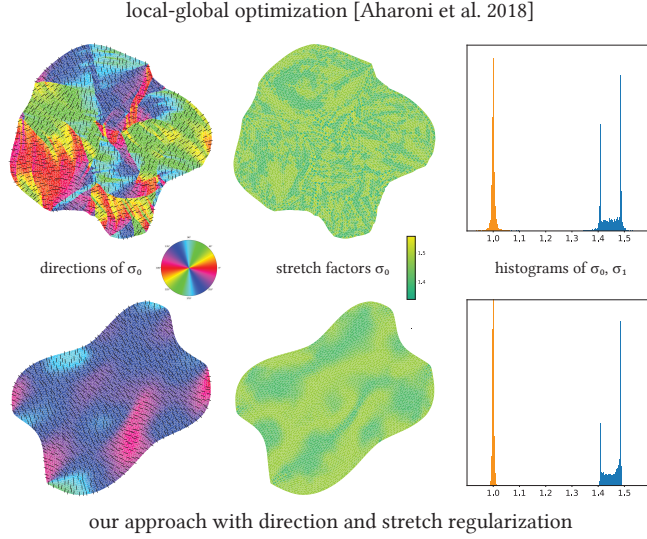


Fig. 5. Flattening with a local-global approach leads to very noisy tube directions and stretch factors (top row). Our new approach achieves a smooth gradation while better fitting the singular values to the admissible range, here chosen to be the interval $[1.4, 1.5]$ for σ_0 .

this issue by adding regularization terms to (3) to encourage more structure in the stretching factor and direction fields:

$$E_{\text{flat}}[f] := \underbrace{\frac{1}{2|\mathcal{M}|} \int_{\mathcal{M}} (\sigma_1 - 1)^2 + (\sigma_0 - [\sigma_0]_{\sigma_{\min}^{\max}}^{\sigma_{\max}})^2 dA}_{\text{singular value fitting}} \quad (4)$$

$$+ \underbrace{\frac{w_\phi}{4} \int_{\mathcal{M}} \left\| \nabla \begin{bmatrix} \cos(2\phi) \\ \sin(2\phi) \end{bmatrix} \right\|_F^2 dA}_{\text{direction regularization}} + \underbrace{\frac{w_\sigma}{2} \int_{\mathcal{M}} \|\nabla \sigma_0\|^2 dA}_{\text{stretch regularization}}$$

where $[\cdot]_{\sigma_{\min}^{\max}}$ clamps to the interval $[\sigma_{\min}, \sigma_{\max}]$, ϕ is the angle between singular vector \mathbf{u}_0 (i.e., the first column of U in (2)) and the x -axis of the plane. The direction regularization term is analogous to the 2-direction field smoothness energies used in several other works [Vaxman et al. 2017], except it measures the smoothness of a direction field in the plane rather than a tangent direction field on the surface \mathcal{M} . The angle ϕ is doubled in this expression to account for the direction field's symmetry with respect to rotations by π .

We discretize this energy using piecewise linear finite elements, so that ∇f , along with its singular values and vectors, is constant on each triangle. The singular-value-fitting term of (4) can be directly evaluated by summing each triangle's area-weighted integrand. To discretize the direction and stretch regularization terms, we compute the Dirichlet energy on the barycentric dual mesh: we triangulate the polygons of the dual mesh arbitrarily without inserting new vertices, and then construct the intrinsic Delaunay triangulation (IDT) using libigl [Jacobson et al. 2018] to avoid issues caused by poorly shaped triangles. Our regularization terms then measure the Dirichlet energy of the piecewise linearly interpolated fields

on this IDT. We use the following identity to simplify the direction regularization term:

$$\frac{1}{2} \left\| \begin{bmatrix} \cos(2\phi_i) \\ \sin(2\phi_i) \end{bmatrix} - \begin{bmatrix} \cos(2\phi_j) \\ \sin(2\phi_j) \end{bmatrix} \right\|^2 = \sin(\phi_i - \phi_j)^2 = \|\mathbf{u}_0^i \times \mathbf{u}_0^j\|^2,$$

which leads to the discrete regularization terms,

$$\frac{w_\phi}{2} \sum_{ij} l_{ij} \|\mathbf{u}_0^i \times \mathbf{u}_0^j\|^2 + \frac{w_\sigma}{2} \sum_{ij} l_{ij} (\sigma_0^i - \sigma_0^j)^2.$$

Here, the sums range over edges ij of the dual mesh's IDT and the scalars l_{ij} are the corresponding off-diagonal entries of the IDT's negative semidefinite piecewise linear FEM Laplacian matrix (i.e., the standard cotan weights [Botsch et al. 2010]). The weights w_ϕ and w_σ control the trade-off between preferring smoothness in the corresponding fields and fitting to the admissible singular values.

Figure 5 bottom shows how the additional regularization terms lead to significantly smoother tube directions and scaling factors. Somewhat surprisingly, despite these added terms, the histograms reveal that our parametrizations better fit the specified target singular values than the unregularized ARAP approach of [Aharoni et al. 2018].

5.1 Curve Tracing

Given a mapping function f , we can now trace out the boundary curves of the air channel network it implies. We use the stripe patterns method of [Knöppel et al. 2015; Ray et al. 2006]. We first upsample f by repeatedly subdividing triangles and applying linear interpolation. Next, we average the stretch directions \mathbf{u}_1 onto the vertices using a circular mean and pass these directions to the reference implementation [Crane 2015], which we modified to sample the following scalar field on a further subdivided triangle mesh:

$$|\pi - \underbrace{(\alpha \bmod 2\pi)}_{\hat{\alpha}}|,$$

where α is the texture coordinate function from [Knöppel et al. 2015] that we reduce to its principal value $\hat{\alpha} \in [0, 2\pi)$. This scalar field is an approximate distance to the wall centerlines (placed where $\hat{\alpha} = \pi$) measured in normalized units where the spacing between wall centers is $w+d = 2\pi$. The desired wall width is determined from the spatially varying stretch factor σ_0 by inverting the relationship from Section 3.1, and we subtract half this width to obtain a signed distance to the wall regions:

$$|\pi - \hat{\alpha}| - \frac{\frac{\pi}{\sigma_0} - 2}{1 - \frac{2}{\pi}}.$$

We extract the zero contour of this signed distance field and uniformly resample the resulting curves to obtain our wall boundaries. We also use this distance field to classify the regions of our sheet mesh bounded by these curves as either wall or tube triangles (see also Figure 4).

5.2 Homogenized bending regularization

Given a layout of air channels computed with the above flattening optimization and curve tracing algorithm, we can now run the inflation simulation discussed in Section 3.3 to compare the deployed

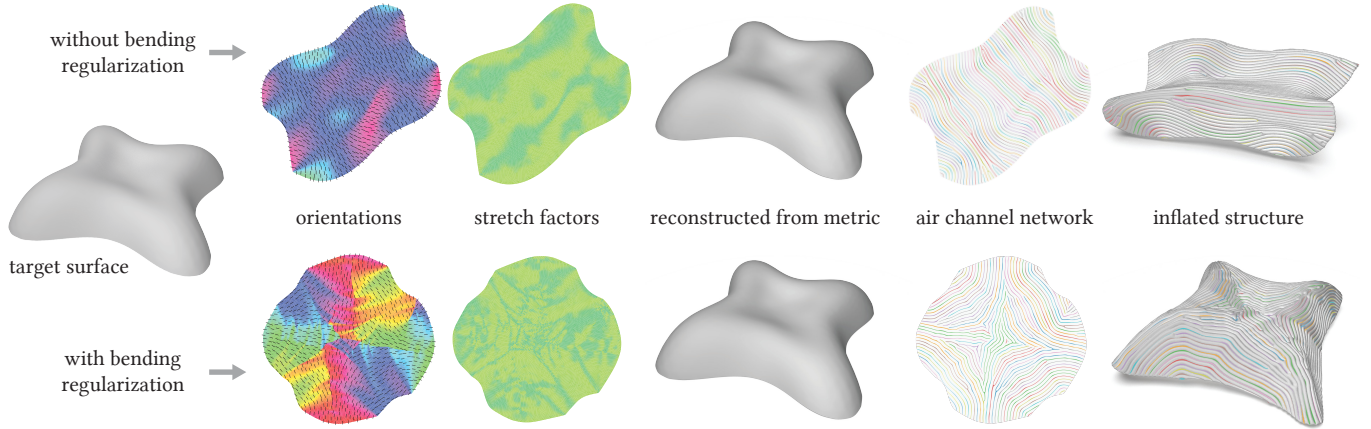


Fig. 6. Flattening the input design surface without bending regularization (top row) yields a metric that closely encodes the target shape. However, the deployed surface does not match the input, indicating that metric properties alone are not sufficient to capture the deformation behavior of inflation. With our bending regularization (bottom row), a completely different layout is obtained that more accurately deploys to the target surface, providing a suitable initialization for subsequent gradient-based optimization.

surface to the input target geometry. Unfortunately, as Figure 6 (top) illustrates, the inflated surface is far from the desired target state. Even when initializing the inflation simulation from the target state, the surface deforms to a completely different equilibrium state. This raises the question whether the computed metric used to design the air channels encodes the target surface correctly. To answer this question, we reconstruct the surface from the anisotropic metric clamped to the admissible range,

$$g := \mathbf{u}_1 \otimes \mathbf{u}_1 + 1/([\sigma_0]_{\sigma_{\min}}^{\sigma_{\max}})^2 \mathbf{u}_0 \otimes \mathbf{u}_0,$$

by solving for an immersion $\hat{f}: f(\mathcal{M}) \rightarrow \mathbb{R}^3$ of the flattened surface that minimizes deviation from this metric:

$$\min_{\hat{f}} \frac{1}{2} \int_{\mathcal{M}} \left\| \nabla \hat{f}^T \nabla \hat{f} - g \right\|_F^2 dA + E_{\text{shell}}(\hat{f}).$$

Here, E_{shell} is the Discrete Shells bending energy [Grinspun et al. 2003], added with a low weight as a regularization to avoid crumpled surfaces. As shown in Figure 6 (top left), the reconstruction is much closer to the target than the surface obtained by inflation simulation even when we initialize the reconstruction with a nearly-planar configuration that is biased only slightly towards the target surface to encourage each bump to pop in the correct direction. This discrepancy suggests that our flattening approach fails to capture important properties of the physical behavior of inflated air channels. In other words, the metric contraction alone does not adequately model the deformation of the surface under inflation.

Indeed, we observe that air channels resist bending out of plane, which is not captured in the metric approach. This means that air channels that traverse the surface along curves of high normal curvature will induce a high bending force on the surface during inflation, which pushes the equilibrium state away from the desired target surface.

To address this issue, we define an additional regularizer that penalizes alignment of the air channels with directions of high normal curvature. We define our regularization energy density at a

point on the target surface \mathcal{M} by drawing an analogy to the energy stored in parallel beams that we can imagine bending to follow geodesics of \mathcal{M} along the tube direction \mathbf{v}_1 . The bending energy stored in a segment of one of these beams is proportional to $\int \kappa^2 ds$, where κ is the beam's centerline curvature; considering our final designs' roughly uniform tube diameters, we imagine all beams to have identically sized circular cross-sections. Summing the bending energies of all the parallel beam segments within an area element dA gives us an energy proportional to $\kappa^2 dA$. Finally, we note that κ is given by the normal curvature of \mathcal{M} in direction \mathbf{v}_1 , which can be evaluated using \mathcal{M} 's shape operator S as $\mathbf{v}_1^T S \mathbf{v}_1$. Integrating this bending energy density over the surface, we obtain our bending regularization term,

$$\frac{w_\kappa}{4} \int_{\mathcal{M}} (\mathbf{v}_1^T S \mathbf{v}_1)^2 dA, \quad (5)$$

that we add to (4) to define the full continuous flattening energy.

We discretize this regularization term by computing a shape operator for each vertex using the implementation of [Panozzo et al. 2010] in libigl; this approach computes the shape operator of a quadric surface fit to the vertex's neighborhood. This defines a piecewise linear field S over \mathcal{M} , and we evaluate (5) with a one-point quadrature rule (i.e., we average S onto the elements to obtain a piecewise constant integrand).

After incorporating all regularization terms above, we are no longer able to use the local-global optimization scheme to solve the parameterization problem. In particular, it is unclear how to solve for optimal U , σ , and V fields in closed form (for fixed f). Instead we directly solve the nonlinear optimization problem using Newton's method with a Hessian modification. We give more details and provide the necessary gradient and Hessian formulas in the supplementary material.

As can be seen in Figure 6 (bottom), incorporating the homogenized bending term into the flattening algorithm yields significant improvements in terms of the deployed surface's geometric fidelity.

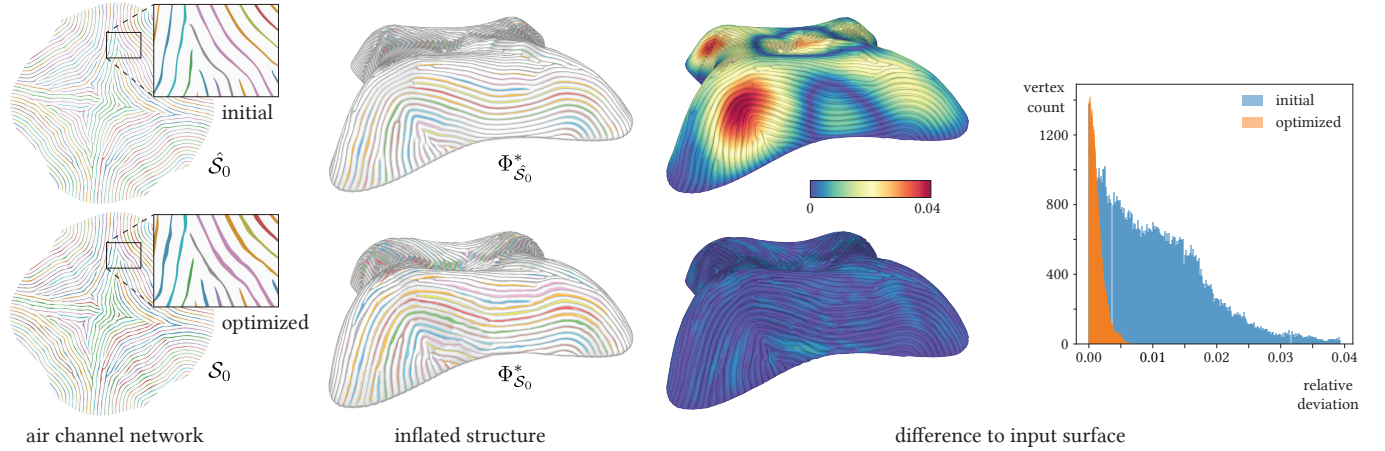


Fig. 7. Air channel network optimization. The initial network computed with the 2D mapping approach yields a deployed shape that does not optimally approximate the desired design surface (top). The optimization modifies the air channel geometry so that the deployed shape more closely matches the target (bottom). The deviation from the input surface is measured relative to the bounding box diagonal.

6 DESIGN OPTIMIZATION

The parameterization-based approach described above computes a 2D channel layout by optimizing the anisotropic metric distortion and minimizing tube bending. While this geometric abstraction captures the essence of the contraction behavior of inflated air channels, it does not directly measure closeness to the design surface, nor does it fully model the physical deformation of an inflatable, which can lead to noticeable deviation from the desired target state (see Figure 7). Better target geometry reproduction requires further optimization.

Starting from the initial undeformed sheet mesh \hat{S}_0 , we perform this optimization directly on the mesh vertices, seeking new positions that result in an inflated equilibrium minimizing the distance to target surface M . However, care is needed to keep curves from degenerating or becoming too jagged.

We formulate this design optimization in a reduced form, using only the positions X_{wall} of vertices on the union of air channel boundary curves γ as variables. We include the sheet’s boundary as part of γ , so the shape of the entire sheet will be optimized as well.

To promote well-shaped, smooth triangles, we solve for the full corresponding undeformed sheet $S_0(X_{\text{wall}})$ by harmonically interpolating the wall vertex positions—and their displacements—into the interior vertices of the fused and unfused regions (see inset). Decomposing the linear FEM Laplacian constructed on \hat{S}_0 into blocks \hat{L}_{ii} , \hat{L}_{iw} , and \hat{L}_{ww} denoting the interior-interior, interior-wall, and wall-wall blocks respectively, we solve for the interior vertex positions:

$$X_{\text{interior}} = -\hat{L}_{ii}^{-1} \hat{L}_{iw} X_{\text{wall}}. \quad (6)$$

These positions define a piecewise linear mapping $\varphi : \hat{S}_0 \rightarrow S_0(X_{\text{wall}})$ representing the design alteration indicated by X_{wall} . We measure the distance from the wall vertices’ equilibrium positions to the

target surface using the target-fitting term:

$$T(\Phi) := \sum_{\mathbf{x} \in \Phi(X_{\text{wall}})} \frac{A_{\mathbf{x}}}{2} \|\mathbf{x} - P_{\mathcal{M}}(\mathbf{x})\|^2,$$

where $P_{\mathcal{M}}(\mathbf{x})$ projects deformed vertex \mathbf{x} to its closest point on target surface \mathcal{M} , and $A_{\mathbf{x}}$ is the area associated with wall vertex \mathbf{x} in \hat{S}_0 (one-third the sum of incident triangle areas). We note that, even for piecewise planar surfaces \mathcal{M} , this distance measure is at least C^1 away from \mathcal{M} ’s medial axis. On the medial axis, it is only C^0 , but this nonsmoothness did not cause difficulties in our experiments.

We can now formulate the design optimization in terms of the modified design’s inflated equilibrium, $\Phi_{S_0}^*(X_{\text{wall}})$:

$$\begin{aligned} \min_{X_{\text{wall}}} J(X_{\text{wall}}) \quad (7) \\ J(X_{\text{wall}}) := T(\Phi_{S_0}^*(X_{\text{wall}})) + w_b \int_{\hat{S}_0} b(\nabla\varphi) \, dA \\ + \frac{w_{\text{sm}}}{2} \int_{\hat{\gamma}} (|\kappa| - |\hat{\kappa}|)_+^2 + \left(\frac{\partial\lambda}{\partial s}\right)^2 \, ds, \quad (8) \end{aligned}$$

where κ and $\hat{\kappa}$ are curvatures of the current and initial channel boundaries γ and $\hat{\gamma}$, respectively, $\lambda = \|\nabla\varphi\hat{\gamma}'\|/\|\hat{\gamma}'\|$ is the tangential length scaling factor induced by the design alteration, and $(\cdot)_+$ clamps negative values to zero. Weights w_b and w_{sm} control the trade-off between fitting the surface and maintaining well-shaped fusing curves and triangles.

The smoothing terms involving κ and λ prevent fusing curves from becoming significantly more jagged than they started and preserve the roughly uniform spacing of X_{wall} . Crucially, they still permit large design modifications; the simpler Dirichlet smoothness energy we first tried posing on displacements $\mathbf{u}(\hat{X}) = \varphi(\hat{X}) - \hat{X}$ resists these large modifications, preferring to shrink $\|\mathbf{u}\|$. We discretize the curvature term using a piecewise constant κ : we assign each vertex Voronoi region the average curvature over that region (the vertex’s turning angle divided by region length). The length scaling term is discretized using a piecewise linear λ defined by

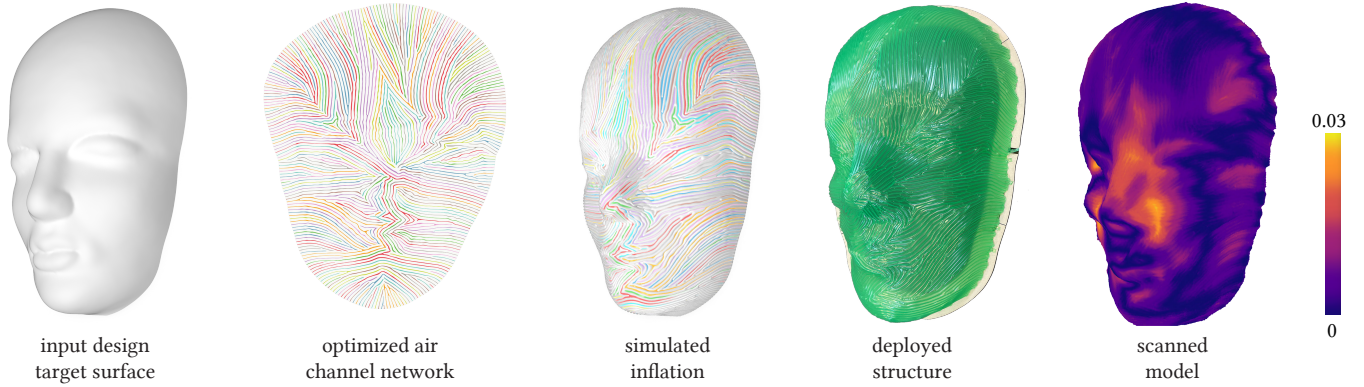


Fig. 8. An inflatable face mask accurately reproduces fine surface detail. The error bar is scaled with respect to the bounding box diagonal.

interpolating the edges’ scaling factors, which are imagined to live at the edge midpoints, along the curve.

The function b is a barrier aiming to preserve admissible fusing curve widths and reasonable tube widths by rapidly growing to infinity when an undeformed triangle is excessively squashed by the design alteration. We specifically use:

$$b(\nabla\varphi) = \sum_{i=0}^1 \left(\log \left(\frac{a}{\sigma_i(\nabla\varphi)} \right) \right)_+^2, \quad (9)$$

which ramps up smoothly from 0 to ∞ as either singular value of the Jacobian of the design alteration map shrinks below the activation threshold a (i.e., if the triangle has been squashed in either direction by the design modification). Since this collapse barrier is integrated over both the fused and unfused regions of the original sheet design $\hat{\mathcal{S}}_0$, it prevents both the walls and the channels from becoming too narrow, as well as keeping any triangle from degenerating. We set $a = 0.9$ for fused triangles and $a = 0.5$ for tube triangles. For fused triangles, we include in (9) an additional stretching term,

$$\left(\log \left(\frac{s_m - s_a}{s_m - \sigma_i(\nabla\varphi)} \right) \right)_+^2,$$

that prohibits the fused regions from thickening by more than a maximum factor s_m (kicking in at activation stretch s_a). We used $s_a = 1.75$, $s_m = 2.25$ in our results.

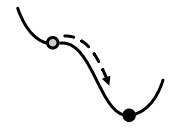
We note that every time the objective J in (7) needs to be evaluated for a new set of wall vertex positions X_{wall} , we must first solve (6) to construct the full corresponding sheet mesh $\mathcal{S}_0(X_{\text{wall}})$ and then solve the nonlinear equilibrium problem (1). We discuss how this entire process can be differentiated to compute gradients of J with respect to X_{wall} in the supplemental material.

Uniqueness of Deployment. One of the primary challenges in solving this design optimization is that the sheet can inflate into multiple stable states. Since the initial state is planar, at the very least we know the structure could inflate into a surface resembling \mathcal{M} or \mathcal{M} ’s reflection across the ground plane. But any individual bump of a structure might independently pop into an inverted configuration. We resolve these ambiguities and stabilize the equilibrium deformations where necessary by adding a small fictitious force to the

simulation (1) that pulls the wall vertices toward their closest point on the target surface (see also [Ly et al. 2018]). This force is applied by adding the term $\epsilon T(\Phi)$ to the total potential energy (1). We start the design optimization with a moderate ϵ and gradually reduce ϵ so that at the end the inflated surface is in equilibrium close to the target without any additional forces. We note that this fictitious force is itself proportional to the distance to the target surface, so the applied forces will automatically diminish with each improvement to the design even without adjusting ϵ . We comment on how to deal with multiple stable states during physical deployment in Section 8.

Navigating local minima. As the design evolves during the course of the optimization, the inflated equilibrium Φ^* being tracked can become unstable, causing the inflated sheet to abruptly pop into a lower-energy configuration. The objective at this new configuration was not accurately predicted by sensitivity analysis, which leads to problems for the optimizer. If this jump in the equilibrium deformation happens to decrease our design objective J , the optimizer will accept the step and carry on tracking the new lower-energy equilibrium.

However, often the jump actually increases the objective. Usually this means that the design change was too large and the step should be rejected by the backtracking line search, but if this happens for small design perturbations, it generally means the equilibrium we were tracking was only barely stable (having only a small energy barrier separating it from a lower energy configuration); the fabricated model is unlikely to inflate into it without falling into this second configuration. Effectively, the objective measured before the jump underestimated the inflated shape’s probable deviation from the target in these cases. Eventually, the design optimizer becomes stuck in its line search, unable to find a design change that obtains a lower objective value than the old, overly-optimistic value. To prevent the optimization from simply terminating, we handle this issue by detecting if the post-jump equilibrium for the new design is actually a lower energy state for the old design candidate. If so, we know the old inflated equilibrium was not a robust local minimum, and we force the optimization to restart with the new lower-energy



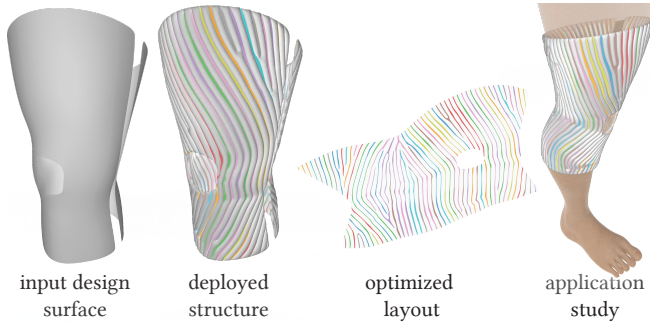


Fig. 9. An inflatable cast provides support when inflated and can be adapted to the specific geometry of the patient’s knee.

equilibrium. While this strategy means the objective is not guaranteed to monotonically decrease throughout our optimization, in practice the optimizer quickly recovers to low objective values after the jump and progress continues.

7 NUMERICAL METHODS AND IMPLEMENTATION

We implement our Newton solver for the flattening and forward simulation problems, as well as all energy, gradient, and Hessian expressions, in C++ using Intel’s TBB for parallelization. We used CHOLMOD [Chen et al. 2008] to solve the Newton systems. The outer design optimization is implemented in Python using wrappers for these C++ routines.

We provide an extensive discussion of specific aspects of the implementation in our supplemental material. In particular, we show we improve the robustness and efficiency of the inflation equilibrium and design optimization solvers through careful caching of matrix factorizations, Hessian modification, weight scheduling, and remeshing. We also show how the objective terms can be reformulated to be scale-independent, simplifying parameter tuning.

8 RESULTS AND DISCUSSION

We evaluate our computational design process with a series of examples listed in Table 1. These models highlight a wide range of possible double-curved surfaces that can be inflated from a flat fabrication state.

Figure 8 shows a high-resolution model that illustrates how fine-scale surface detail can be reproduced accurately. The boundary of this model is fixed during the optimization. Figure 9 illustrates the potential of our method for medical applications. For this example, the cylindrical input geometry is cut along a user-selected curve to obtain a punctured disk topology that can be processed with our algorithm. The model is optimized with a fixed top and bottom boundary, while the hole around the knee cap is not constrained during deployment. In Figure 10, we show a product design study of an inflatable bike helmet. In this example, we also show the effect of the stripe pattern wavelength $w + d$ that controls the density and size of air channels. In general, we observe a trade-off between closer approximation of the target surface for small wavelength, and reduced fabrication time and higher stiffness for large wavelength.

Figure 11 shows potential applications of inflatables in architecture. Key benefits are the fast and simple fabrication, compact storage and efficient transportation, rapid deployment without complex scaffolding, and re-usability. These advantages make inflatables suitable for a variety of architectural applications, such as temporary event spaces or emergency shelters. In these examples, the parts of the boundary that are in contact with the ground are anchored prior to inflation.

8.1 Fabrication and Deployment

The physical prototypes of Figure 11 have been fabricated with the custom-built CNC machine shown in Figure 2. We use $75 \mu\text{m}$ thick Polyethylene vacuum bagging film (80 g/m^2) as the sheet material, but many other thin and approximately inextensible materials such as metallic foils could be used as well. In order to inflate the sheet from a single inlet, we connect all tube regions in a post process. We automatically add small semi-circular bypass tubes around the fused walls at the mesh boundary, which are sufficient to obtain a single connected tube component for most designs. For the infrequent cases where internal walls form closed loops, we manually break the loops, inserting small tubes tunneling through them. These alterations are small enough not to significantly affect the deformation. Each prototype took between two and three hours to fabricate.

The examples shown have been inflated to approximately 0.5 bar. Once fully deployed, the inflatable retains the same surface geometry over a fairly wide range of pressure values. This insensitivity to pressure avoids the need for careful pressure control, which simplifies deployment and maintenance of the inflated structure.

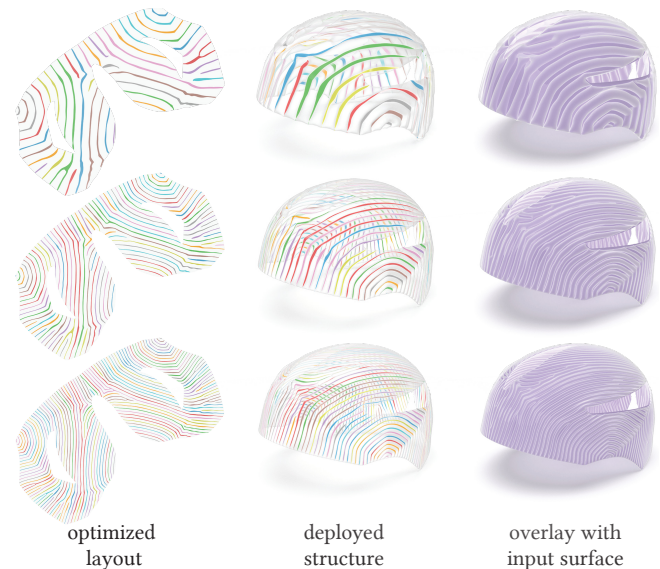


Fig. 10. An inflatable bike helmet. Varying the pattern wavelength used to trace the initial air channels leads to inflatables of different resolution. The boundaries of the inner holes are freed in the simulation, while the outer boundary is fixed.

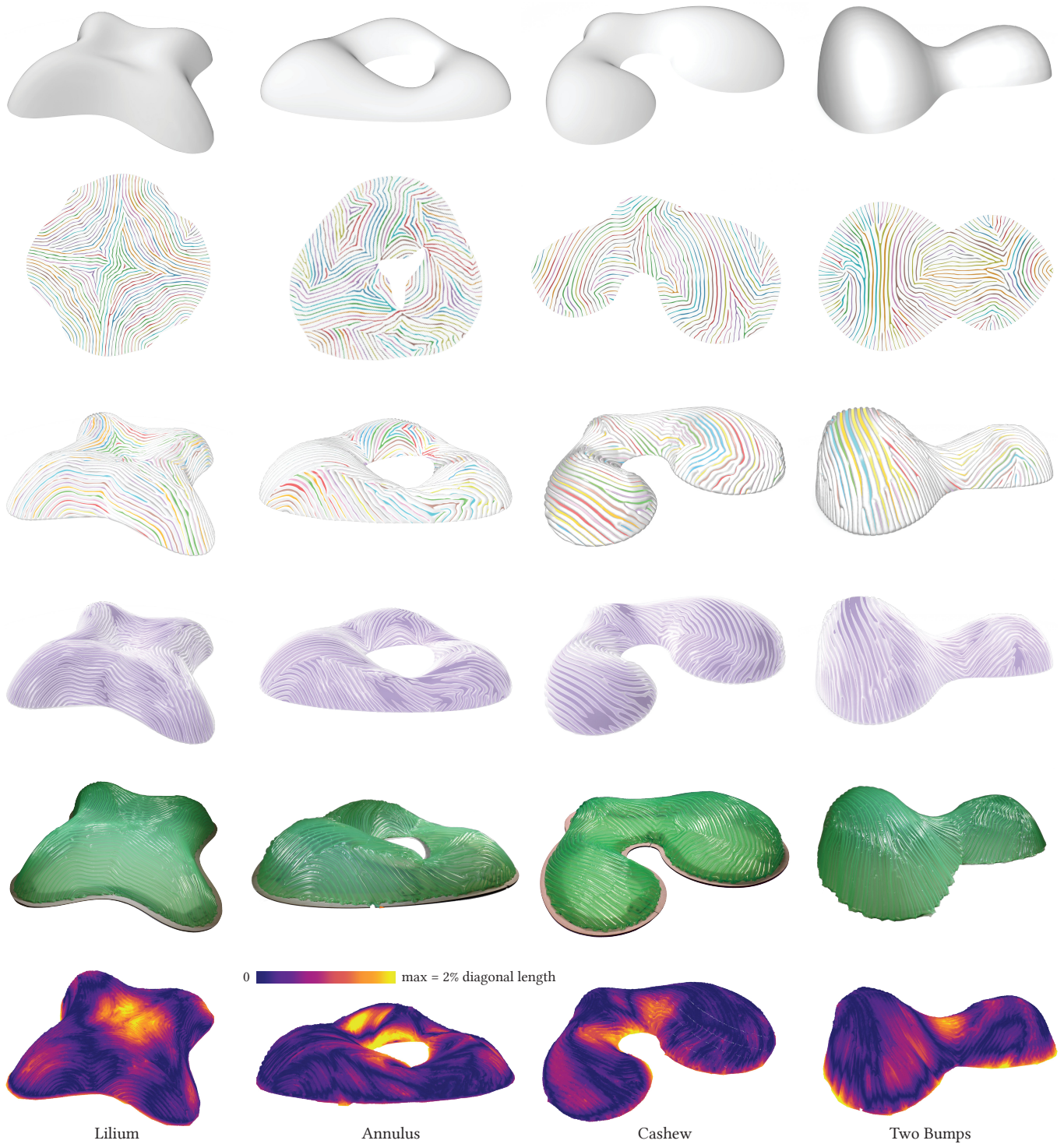


Fig. 11. Inflatable structures for potential applications in architecture offer a wide variety of freeform geometries. From top to bottom: Input surface, planar air channel layout, simulated deployment, overlay with reference surface, deployed physical model, deviation between scan and input surface. The examples shown here were simulated and optimized with a fixed boundary. The deployed prototypes have been scanned using photometric stereo.

Table 1. Statistics for our examples. *Sim. DoF* refers to the degrees of freedom for the inflation simulation, *Opt. DoF* counts the number of variables in the design optimization, i.e. the 2D coordinates of the fusing wall vertices. Timings are shown in (hour):min:sec for flattening optimization (*Flat*), stripe generation and meshing (*Stripe*), initial equilibrium solve (*Equi*), and design optimization (*Opt*). All computations are performed on an Apple M1 processor with 16GB of RAM. *Sim. Fit* and *Scan Fit* report the Avg/Max deviations from the reference surface of the simulated and fabricated inflatable, respectively, as percentages of the bounding box diagonal. The Max error is indicated as the 99th percentile to remove scanning outliers.

Model (Figures)	Sim. DoF	Opt. DoF	Flat	Stripe	Equi	Opt	Sim. Fit (%)	Scan Fit (%)
Cast (9)	398,043	35,550	1:03	0:32	0:45	1:02:21	0.134/0.499	-/-
Cashew (11)	412,404	30,968	0:46	0:18	0:40	1:40:18	0.185/0.591	0.410/1.57
Med. Helmet (10)	416,409	36,466	2:02	0:37	0:49	1:08:56	0.086/0.386	-/-
Two Bumps (11)	439,245	33,878	0:42	0:22	0:44	2:08:55	0.124/0.443	0.511/1.83
Annulus (11)	484,125	47,978	2:11	0:40	0:55	2:38:48	0.183/0.643	0.515/2.06
Lilium (11)	503,421	48,846	1:26	0:26	0:41	2:32:55	0.127/0.487	0.846/4.39
Face (8)	1,309,785	136,834	21:41	1:42	5:02	11:30:01	0.070/0.350	0.604/2.09

8.2 Validation

To verify that our simulation adequately predicts the deformation behavior of physical inflatables, we compare the simulated and fabricated deployed surfaces. The latter have been scanned using the Agisoft Metashape software that reconstructs a 3D model based on photometric stereo. As these examples demonstrate, the automatically inflated models achieve a good agreement with the predicted deployed shape.

Figure 12 analyzes the structural properties of an inflatable (see also accompanying video). We observed that our physical models are stiff and can support point loads significantly exceeding their total weights. In general, we found that models with free boundaries that exhibit near-isometric deformations require fixing (parts of) the boundary to maintain a stable shape.

8.3 Limitations and Future Work

The bounded in-plane contraction of air channels imposes limits on the target shapes that can be achieved with our method. Figure 13 illustrates how our method fails when the flattening of the design surface induces distortion outside of the material bounds. One possible remedy is to add cuts to the surface to enable a flattening with less distortion, but this might reduce the structural stability of the deployed surface and complicate deployment.

A related aspect is that our current implementation can only handle input surfaces with disk topology. More complex topologies could be achieved by gluing the boundary (see Figure 9) or by connecting multiple sheets. Currently, this requires manually specifying appropriate cuts on the input surface. Automatically finding optimal cuts, both for reducing distortion and handling more complex topologies, is an interesting topic for future work.

Our forward simulation computes an accurate prediction of the inflated state, but does not simulate the temporal evolution of the inflation process directly. We thus cannot guarantee that deployment is always successful, because self-collisions, for example, could lead to locked states, which might require additional control during deployment. In practice, we did not observe this behavior on any of our models. As discussed in Section 6, ambiguities of deployment can arise due to symmetries that lead to multiple stable equilibria states. We currently do not augment the physical structure to break such

symmetries, relying instead on manual support during deployment to guide the inflated surface towards the desired target state.

Tension field theory assumes that no compressive elastic forces act on the deployed configuration; compressed elements are assumed to have simply wrinkled into configurations storing insignificant energy. This is a good approximation for thin sheets with negligible bending stiffness—the film used in our prototypes readily wrinkles even in fused regions—but an accurate simulation for thicker materials would require a high-resolution shell simulation with a finite bending energy term and an un-relaxed energy density.

Our experiments indicate favorable structural properties of our optimized inflatables. We have not yet conducted a comprehensive analysis, however. Interesting future work could incorporate structural aspects into the optimization, for example to optimize an architectural design to withstand snow or wind loads.

Other extensions include more user control, for example by drawing feature lines onto the surface that act as (soft) constraints for the direction field optimization. Another interesting avenue for future work is to investigate multiple actuation states. The surface could be equipped with simple switches that open or close certain air channels to dynamically alter the target state, opening up new paths for applications in soft robotics, for example.

9 CONCLUSION

Inflatable structures are a new type of deployable freeform surface structure that can be fabricated easily while offering a rich shape space for a wide variety of applications. We found the two-stage algorithm essential to make the inverse design process computationally tractable. The optimization of air channels is highly nonlinear and non-convex and will not converge to an acceptable solution if initialized far from an appropriate local optimum. On the other hand, the parameterization approach alone is not sufficient, since it mainly optimizes for metric distortion, but does not capture important aspects of the physical behavior of inflated tube networks. The integration of both methods yields an effective solution that combines geometric abstraction with predictive physical modeling. As such, our approach exemplifies a general strategy for deployable structures that can potentially be transferred to other material systems with a similar in-plane actuation, such as new engineered actuated materials or biologically growing structures.

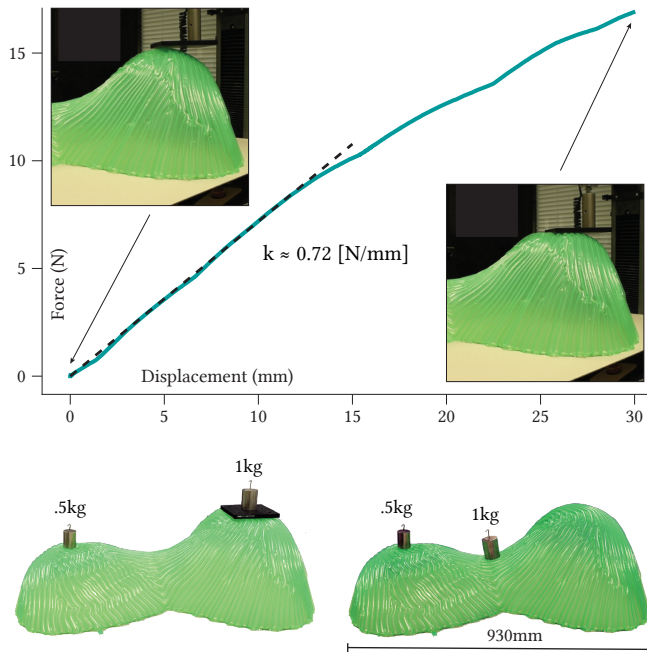


Fig. 12. Structural testing: We apply progressively larger vertical displacements and measure the reaction force. From the force-displacement diagram, we estimate the stiffness modulus k as the slope of the linear portion of the curve (top). At $75\ \mu\text{m}$ material thickness and a total weight of only 0.161 kg, this inflatable can support sizeable loads (bottom).

We believe our method can inspire numerous new applications. Industrial processes already exist to fabricate inflatables at many different scales and with many different materials. With our computational inverse design solution, new freeform structures will now become possible, with potential applications in diverse fields such as architecture, consumer products, medicine, or soft robotics.

ACKNOWLEDGMENTS

This research was supported by the NCCR Digital Fabrication, funded by the Swiss National Science Foundation (NCCR Digital Fabrication Agreement #51NF40-141853) and the SNF Award (FNS 514543 / CF 1156).

REFERENCES

- Hillel Aharoni, Eran Sharon, and Raz Kupferman. 2014. Geometry of Thin Nematic Elastomer Sheets. *Phys. Rev. Lett.* 113 (Dec 2014), 257801. Issue 25. <https://doi.org/10.1103/PhysRevLett.113.257801>
- Hillel Aharoni, Yu Xia, Xinyue Zhang, and Shu Kamien, Randall D. and Yang. 2018. Universal inverse design of surfaces with thin nematic elastomer sheets. *Proceedings of the National Academy of Sciences of the United States of America* 115, 28 (2018).
- Marco Attene, Marco Livesu, Sylvain Lefebvre, Thomas Funkhouser, Stefano Ellero, Szymon Rusinkiewicz, Jonàs Martínez, and Amit Haim Bermano. 2018. *Design, Representations, and Processing for Additive Manufacturing*. Vol. 10. Morgan & Claypool Publishers. 146 pages. <https://hal.inria.fr/hal-01836525>
- Amit H. Bermano, Thomas Funkhouser, and Szymon Rusinkiewicz. 2017. State of the Art in Methods and Representations for Fabrication-Aware Design. *Comput. Graph. Forum* 36, 2 (May 2017), 509–535. <https://doi.org/10.1111/cgf.13146>
- Katia Bertoldi, Vincenzo Vitelli, Johan Christensen, and Martin van Hecke. 2017. Flexible mechanical metamaterials. *Nature Reviews Materials* 2 (17 10 2017), 17066 EP –. <https://doi.org/10.1038/natrevmats.2017.66>



Fig. 13. Not all shapes can be inflated from a single patch. For increasingly larger parts of the sphere, metric distortion eventually exceeds the bounds realizable by the limited air channel contraction, leading to increasingly larger deviations from the target shape.

- Bernd Bickel, Paolo Cignoni, Luigi Malomo, and Nico Pietroni. 2018. State of the Art on Stylized Fabrication. *Computer Graphics Forum* (2018). <https://doi.org/10.1111/cgf.13327>
- J. William Boley, Wim M. van Rees, Charles Lissandrello, Mark N. Horenstein, Ryan L. Truby, Arda Kotikian, Jennifer A. Lewis, and L. Mahadevan. 2019. Shape-shifting structured lattices via multimaterial 4D printing. *Proceedings of the National Academy of Sciences* 116, 42 (2019), 20856–20862. <https://doi.org/10.1073/pnas.1908806116> arXiv:<https://www.pnas.org/content/116/42/20856.full.pdf>
- Mario Botsch, Leif Kobbelt, Mark Pauly, Pierre Alliez, and Bruno Levy. 2010. *Polygon Mesh Processing*. AK Peters.
- Paolo Celli, Connor McMahan, Brian Ramirez, Anton Bauhofer, Christina Naify, Douglas Hofmann, Basile Audoly, and Chiara Daraio. 2018. Shape-morphing architected sheets with non-periodic cut patterns. *Soft matter* (2018).
- Yanqing Chen, Timothy A Davis, William W Hager, and Sivasankaran Rajamanickam. 2008. Algorithm 887: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate. *ACM Transactions on Mathematical Software (TOMS)* 35, 3 (2008), 22.
- Keenan Crane. 2015. stripe: A vector field editor and stripe pattern generator. <http://www.cs.cmu.edu/~kmc Crane/Projects/StripePatterns/code.zip>
- Erik D. Demaine and Joseph O'Rourke. 2008. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra* (reprint ed.). Cambridge University Press, USA.
- R.P. Feynman, R.B. Leighton, and M.L. Sands. 1963. *The Feynman Lectures on Physics*. Number Bd. 1 in The Feynman Lectures on Physics. Addison-Wesley. <https://books.google.ch/books?id=UjEAAAIAAJ>
- Akash Garg, Andrew O. Sageman-Furnas, Bailin Deng, Yonghao Yue, Eitan Grinspun, Mark Pauly, and Max Wardetzky. 2014. Wire Mesh Design. *ACM Trans. Graph.* 33, 4, Article 66 (July 2014), 12 pages. <https://doi.org/10.1145/2601097.2601106>
- S. A. Gladman, Elisabetta A. Matsumoto, Ralph G. Nuzzo, L. Mahadevan, and Jennifer A. Lewis. 2016. Biomimetic 4D printing. *Nature Materials* 15 (25 01 2016), 413 EP –. <https://doi.org/10.1038/nmat4544>
- Eitan Grinspun, Anil N. Hirani, Mathieu Desbrun, and Peter Schröder. 2003. Discrete Shells. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Diego, California).
- Ruslan Guseinov, Eder Miguel, and Bernd Bickel. 2017. CurveUps: Shaping Objects from Flat Plates with Tension-actuated Curvature. *ACM Trans. Graph.* 36, 4, Article 64 (July 2017), 12 pages.
- Alec Jacobson, Daniele Panozzo, et al. 2018. libigl: A simple C++ geometry processing library. <https://libigl.github.io/>.

- Martin Kilian, Simon Flöry, Zhonggui Chen, Niloy J. Mitra, Alla Sheffer, and Helmut Pottmann. 2008. Curved Folding. *ACM Trans. Graph.* 27, 3, Article 75 (Aug. 2008), 9 pages. <https://doi.org/10.1145/1360612.1360674>
- Martin Kilian, Aron Monszpart, and Niloy J. Mitra. 2017. String Actuated Curved Folded Surfaces. *ACM Trans. Graph.* 36, 4, Article 64a (May 2017). <https://doi.org/10.1145/3072959.3015460>
- Felix Knöppel, Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2015. Stripe Patterns on Surfaces. *ACM Trans. Graph.* 34, 4, Article 39 (July 2015), 11 pages. <https://doi.org/10.1145/2767000>
- Mina Konaković, Keenan Crane, Bailin Deng, Sofien Bouaziz, Daniel Piker, and Mark Pauly. 2016. Beyond Developable: Computational Design and Fabrication with Auxetic Materials. *ACM Trans. Graph.* 35, 4, Article 89 (July 2016), 11 pages.
- Mina Konaković-Luković, Pavle Konaković, and Mark Pauly. 2018a. Computational Design of Deployable Auxetic Shells. In *Advances in Architectural Geometry 2018*. 94–111.
- Mina Konaković-Luković, Julian Panetta, Keenan Crane, and Mark Pauly. 2018b. Rapid Deployment of Curved Surfaces via Programmable Auxetics. *ACM Trans. Graph.* (Aug. 2018).
- Mickaël Ly, Romain Casati, Florence Bertails-Descoubes, Mélina Skouras, and Laurence Boissieux. 2018. Inverse Elastic Shell Design with Contact and Friction. *ACM Trans. Graph.* 37, 6, Article 201 (Dec. 2018), 16 pages. <https://doi.org/10.1145/3272127.3275036>
- Li-Ke Ma, Yizhong Zhang, Yang Liu, Kun Zhou, and Xin Tong. 2017. Computational Design and Fabrication of Soft Pneumatic Objects with Desired Deformations. *ACM Trans. Graph.* 36, 6, Article 239 (Nov. 2017), 12 pages. <https://doi.org/10.1145/3130800.3130850>
- Luigi Malomo, Jesús Pérez, Emmanuel Iarussi, Nico Pietroni, Eder Miguel, Paolo Cignoni, and Bernd Bickel. 2018. FlexMaps: Computational Design of Flat Flexible Shells for Shaping 3D Objects. *ACM Trans. Graph.* 37, 6, Article 241 (Dec. 2018), 14 pages. <https://doi.org/10.1145/3272127.3275076>
- Bobak Mosadegh, Panagiotis Polygerinos, Christoph Keplinger, Sophia Wennstedt, Robert F. Shepherd, Unmukt Gupta, Jongmin Shim, Katia Bertoldi, Conor J. Walsh, and George M. Whitesides. 2014. Pneumatic Networks for Soft Robotics that Actuate Rapidly. *Advanced Functional Materials* 24, 15 (2014), 2163–2170. <https://doi.org/10.1002/adfm.201303288>
- Jifei Ou, Mélina Skouras, Nikolaos Vlavianos, Felix Heibeck, Chin-Yi Cheng, Jannik Peters, and Hiroshi Ishii. 2016. aeroMorph - Heat-sealing Inflatable Shape-change Materials for Interaction Design. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (Tokyo, Japan) (UIST '16)*. ACM, New York, NY, USA, 121–132. <https://doi.org/10.1145/2984511.2984520>
- J. Panetta, M. Konaković-Luković, F. Isvoranu, E. Bouleau, and M. Pauly. 2019. X-Shells: A New Class of Deployable Beam Structures. *ACM Trans. Graph.* 38, 4, Article 83 (July 2019), 15 pages. <https://doi.org/10.1145/3306346.3323040>
- Daniele Panozzo, E Puppo, and L Rocca. 2010. Efficient multi-scale curvature and crease estimation. *Proceedings of Computer Graphics, Computer Vision and Mathematics (Brno, Czech Republic)* 1, 6 (2010).
- Jesús Pérez, Miguel A. Otaduy, and Bernhard Thomaszewski. 2017. Computational Design and Automated Fabrication of Kirchhoff-plateau Surfaces. *ACM Trans. Graph.* 36, 4, Article 62 (July 2017), 12 pages.
- Stefan Pillwein, Kurt Leimer, Michael Birsak, and Przemyslaw Musialski. 2020. On Elastic Geodesic Grids and Their Planar to Spatial Deployment. *ACM Trans. Graph.* 39, 4, Article 125 (July 2020), 12 pages. <https://doi.org/10.1145/3386569.3392490>
- Nicolas Ray, Wan Chiu Li, Bruno Lévy, Alla Sheffer, and Pierre Alliez. 2006. Periodic Global Parameterization. *ACM Trans. Graph.* 25, 4 (Oct. 2006), 1460–1485. <https://doi.org/10.1145/1183287.1183297>
- Andrew O. Sageman-Furnas, Albert Chern, Mirela Ben-Chen, and Amir Vaxman. 2019. Chebyshev Nets from Commuting PolyVector Fields. *ACM Trans. Graph.* 38, 6, Article 172 (Nov. 2019), 16 pages. <https://doi.org/10.1145/3355089.3356564>
- Jonathan Richard Shewchuk. 1996. Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. In *Applied Computational Geometry: Towards Geometric Engineering*, Ming C. Lin and Dinesh Manocha (Eds.). Lecture Notes in Computer Science, Vol. 1148. Springer-Verlag, 203–222.
- Yan Shi, Fan Zhang, Kewang Nan, Xueju Wang, Juntong Wang, Yijie Zhang, Yutong Zhang, Haiwen Luan, Keh-Chih Hwang, Yonggang Huang, John A. Rogers, and Yihui Zhang. 2017. Plasticity-induced origami for assembly of three dimensional metallic structures guided by compressive buckling. *Extreme Mechanics Letters* 11 (2017), 105 – 110. <https://doi.org/10.1016/j.eml.2016.11.008>
- Emmanuel Siéfert, Etienne Reyssat, José Bico, and Benoit Roman. 2019a. Bio-inspired pneumatic shape-morphing elastomers. *Nature Materials* 18, 1 (2019), 24–28. <https://doi.org/10.1038/s41563-018-0219-x>
- Emmanuel Siéfert, Etienne Reyssat, José Bico, and Benoit Roman. 2019b. Programming curvilinear paths of flat inflatables. *Proceedings of the National Academy of Sciences* 116, 34 (2019), 16692–16696. <https://doi.org/10.1073/pnas.1904544116> arXiv:<https://www.pnas.org/content/116/34/16692.full.pdf>
- Emmanuel Siéfert, Etienne Reyssat, José Bico, and Benoit Roman. 2020. Programming stiff inflatable shells from planar patterned fabrics. *Soft Matter* 16 (2020), 7898–7903. Issue 34. <https://doi.org/10.1039/D0SM01041C>
- Mélina Skouras, Bernhard Thomaszewski, Bernd Bickel, and Markus Gross. 2012. Computational Design of Rubber Balloons. *Comput. Graph. Forum* 31, 2pt4 (May 2012), 835–844. <https://doi.org/10.1111/j.1467-8659.2012.03064.x>
- Mélina Skouras, Bernhard Thomaszewski, Peter Kaufmann, Akash Garg, Bernd Bickel, Eitan Grinspun, and Markus Gross. 2014. Designing Inflatable Structures. *ACM Trans. Graph.* 33, 4, Article 63 (July 2014), 10 pages. <https://doi.org/10.1145/2601097.2601166>
- T. Tachi. 2009. Origamizing Polyhedral Surfaces. *IEEE Transactions on Visualization and Computer Graphics* 16, 2 (2009), 298–311. <https://doi.org/10.1109/TVCG.2009.67>
- Tomohiro Tachi, Motoi Masubuchi, and M. Iwamoto. 2012. Rigid Origami Structures with Vacuumatics: Geometric Considerations. In *IASS Proceedings*.
- Michael T Tolley, Samuel M Felton, Shuhei Miyashita, Daniel Aukes, Daniela Rus, and Robert J Wood. 2014. Self-folding origami: shape memory composites activated by uniform heating. *Smart Materials and Structures* 23, 9 (aug 2014), 094006. <https://doi.org/10.1088/0964-1726/23/9/094006>
- Wim M. van Rees, Etienne Vouga, and L. Mahadevan. 2017. Growth patterns for shape-shifting elastic bilayers. *Proceedings of the National Academy of Sciences* 114, 44 (2017), 11597–11602. <https://doi.org/10.1073/pnas.1709025114>
- Amir Vaxman, Marcel Campen, Olga Diamanti, David Bommes, Klaus Hildebrandt, Mirela Ben-Chen Technion, and Daniele Panozzo. 2017. Directional Field Synthesis, Design, and Processing. In *ACM SIGGRAPH 2017 Courses*. ACM, Article 12, 30 pages. <https://doi.org/10.1145/3084873.3084921>
- Lining Yao, Ryuma Niiyama, Jifei Ou, Sean Weston Follmer, Clark Della Silva, and Hiroshi Ishii. 2013. PneuUI: Pneumatically Actuated Soft Composite Materials for Shape Changing Interfaces. *UIST 2013 - Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, 13–22. <https://doi.org/10.1145/2501988.2502037>
- Yihui Zhang, Fan Zhang, Zheng Yan, Qiang Ma, Xiuling Li, Yonggang Huang, and John A. Rogers. 2017. Printing, folding and assembly methods for forming 3D mesostructures in advanced materials. *Nature Reviews Materials* 2 (29 03 2017), 17019 EP -. <https://doi.org/10.1038/natrevmats.2017.19>