Computational Design and Fabrication of Soft Pneumatic Objects with Desired Deformations

LI-KE MA*, Tsinghua University and Microsoft Research Asia
YIZHONG ZHANG*, Microsoft Research Asia
YANG LIU, Microsoft Research Asia
KUN ZHOU, Zhejiang University
XIN TONG, Microsoft Research Asia

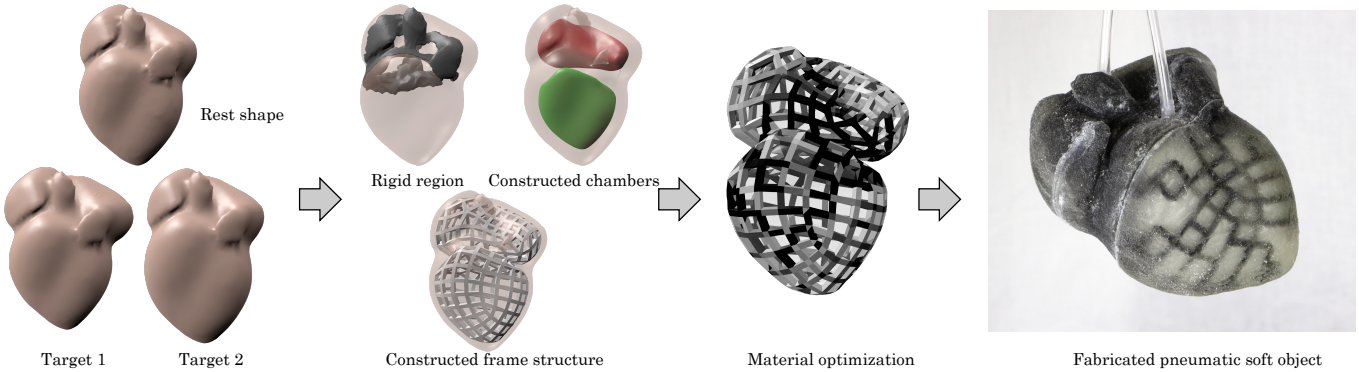


Fig. 1. Our end-to-end system for soft pneumatic objects design and fabrication. Given a rest shape and deformed shapes of a 3D heart object (a), our method automatically computes the chamber and frame structure (b) and material distribution (c) of the frames so that the fabricated object (d) designed by our method can reproduce the heart beating deformations with a controlled pneumatic system. The color in (c&d) indicates the material rigidity, the darker the more rigid the material is.

We present an end-to-end solution for design and fabrication of soft pneumatic objects with desired deformations. Given a 3D object with its rest and deformed target shapes, our method automatically optimizes the chamber structure and material distribution inside the object volume so that the fabricated object can deform to all the target deformed poses with controlled air injection. To this end, our method models the object volume with a set of chambers separated by material shells. Each chamber has individual channels connected to the object surface and thus can be separately controlled with a pneumatic system, while the shell is comprised of base material with an embedded frame structure. A two-step algorithm is developed to compute the geometric layout of the chambers and frame structure as well as the material properties of the frame structure from the input. The design results can be fabricated with 3D printing and deformed by a controlled pneumatic system. We validate and demonstrate the efficacy of our method with soft pneumatic objects that have different shapes and deformation behaviors.

CCS Concepts: • **Computing methodologies** → **Modeling methodologies**; **Physical simulation**; **Mesh models**;

*Joint first authors

0730-0301/2017/11-ART239 $15.00
https://doi.org/10.1145/3130800.3130850

Additional Key Words and Phrases: soft pneumatic objects, computational fabrication, 3D printing

ACM Reference Format:
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 (November 2017), 12 pages.
https://doi.org/10.1145/3130800.3130850

1 INTRODUCTION

Pneumatically actuated objects have been widely used for realizing soft body deformations [Majidi 2013; Rieffel et al. 2014; Rus and Tolley 2015]. Different from articulated objects constructed by rigid parts linked with discrete joints, soft pneumatic objects are composed of soft materials and embedded chambers. After injecting air into the chambers, the air pressure gradually inflates the chambers and continuously deforms the object to its equilibrium state. This process can be easily controlled by a low-cost pneumatic system. Due to these advantages, pneumatically actuated objects have been used in many applications, such as soft robotics [Deimel and Brock 2016], biological simulation [Marchese et al. 2014], and entertainment [Jamele and Ellison 2013].

However, designing a soft pneumatic object with desired deformations is a very challenging task. Different from articulated objects whose deformations are controlled by handle positions and the degrees of freedom of sparse joints [Bächer et al. 2012] and soft objects

driven by external forces [Skouras et al. 2013], the deformation of a pneumatically actuated object is affected by multiple factors: the geometry (i.e., shape and layout) of the empty chambers, the material properties inside the object volume, and the air pressure in each pneumatic chamber. All these factors are coupled together in a highly non-linear physical process to determine the deformation of the object surface, which makes the design of soft pneumatic objects with desired deformations extremely difficult. Existing design efforts for soft pneumatic objects either focus on simple-shaped pneumatic actuators with 1D elasticity or bending [Elsayed et al. 2014; Yap et al. 2016], or follow trial-and-error to manually design soft pneumatic objects with specific deformations [Marchese et al. 2014]. To the best of our knowledge, there is no end-to-end solution for designing soft pneumatic objects with desired deformations.

In this paper, we take one step forward toward an end-to-end method for designing and fabricating soft pneumatic objects with desired deformations. Given a 3D object shape and a set of deformed target shapes that the object needs to achieve at its equilibrium state (Fig. 1(a)), our method automatically computes the pneumatic chamber structure and the material properties inside the object volume structure, as well as the air pressure of each individual pneumatic chamber to generate a 3D printable object so that the simulated deformations of the designed object volume can achieve all the target shapes (Fig. 1(b&c)). After fabrication, the object with designed internal structure can deform to each target pose after the air pressure in each chamber achieves the optimized value (Fig. 1(d)).

The key observation of our solution is that in a soft pneumatic object volume, the chamber and the surrounding material play different roles in object deformation. The inflated chamber driven by air pressure produces large and isotropic deformations, while the surrounding soft material provides fine scale and anisotropic constraints for the chamber deformation. Based on this observation, we model the object volume with a set of chambers separated with material shells (Fig. 1(b)). Each chamber has an individual air channel to the model surface that is connected to the pneumatic system using an air tube, thus it can be separately controlled to approximate the large scale volumetric deformation. The surrounding shells driven by the inflated chambers are used to simulate fine-scale anisotropic deformations. Inspired by the fiber structure used for many soft objects [Connolly et al. 2017; Polygerinos et al. 2015b], we further simplify the material distribution in the shells by modeling the shell with a homogeneous soft base material with embedded 2D frames. Each frame segment is homogeneous while different frame segments are made of various materials and thus have different physical properties.

Our simplified internal structure model greatly reduces the number of unknowns and makes the design optimization tractable. To this end, we develop an efficient algorithm to optimize the geometry and physical properties of the model in two separate steps. In the first step, we assume that the object volume is filled with the homogeneous base material and compute the volumetric deformations from the target poses with a physically based simulation. We then identify the volumetric regions sharing similar large deformations and replace each of them with separate chambers. After that, we embed the frames around each chamber with their directions following the principal Green strain directions as much as possible. In the second step, we optimize the material properties of each frame segment in the shell as well as the air pressure in each chamber so that the simulated deformation of the designed object can achieve the desired deformations.

We implemented our system and fabricated a set of soft pneumatic objects that have different 3D shapes and deformation behaviors. Controlled by a pneumatic system, the 3D-printed objects successfully reproduce all target poses. We also compare our method with other alternative solutions to validate the efficacy of our method.

2 RELATED WORK

2.1 Active deformable model

Active deformable models, also called *soft robots*, can mimic the physical behavior of various objects and perform different tasks such as grasping or moving on unknown surfaces. Compared to rigid robotics, they are more suitable for medical, human interaction, and wearable applications as they reduce the potential harm to the human body and thus have gained much attention in both academia and industry [Rus and Tolley 2015].

Design of active deformable models. Many active deformable models have been proposed for different tasks, and most of them used fluid control (i.e. pneumatic or hydraulic actuation) for achieving quick response and rapid motion due to the great output power of fluid control systems. Marchese *et al.* [2014] construct a self-contained soft robotic fish with some tiny and compact pneumatic devices. Polygerinos *et al.* [2015a] make a hydraulic actuated soft robotic glove to augment hand rehabilitation for individuals with functional grasp pathologies. Deimel *et al.* [2016] build a pneumatically actuated robotic hand that can perform grasping. This robot is fabricated using silicone rubber and embedded with inelastic fabric so that when the rubber actuator is inflated, it can bend to the direction where the embedded inelastic fabric is. Moseley *et al.* [2016] design the chamber layout manually for soft pneumatic actuators and validate the result by the finite element method. However, the shape of actuators after injecting air can have many bumps, and it is not suitable for matching the desired target shapes.

Besides fluid control, there are also soft robots actuated by shape memory alloys and chemical reactions. Inspired by the motion of an octopus, Laschi *et al.* [2012] build robotic octopus arms that can be bent and twisted by actuating a shape memory alloy, which shrinks under electric heating. However, the speed of arm motion is relatively slow because the temperature of a shape memory alloy cannot be suddenly changed. Wehner *et al.* [2016] use the activation of two chemical fluids to actuate the motion of fully-soft robots, but the structure of the robot is difficult to fabricate due to its complexity.

Since most existing active deformable models are built for specific tasks, they are typically manually-designed with the aid of commercial CAD systems. However, current CAD systems cannot be specified to deal with deformable models, resulting in difficulty in soft robot design. People have started to explore alternative methods for automated design. Hiller *et al.* [2012] use an evolutionary algorithm for design optimization, but their voxel representation of shapes cannot preserve the rest shape of the model well. Rieffel *et al.* [2014] also use an evolutionary method but face manufacturing difficulties because they do not take the fabrication process into consideration.

Fabrication of active deformable models. Traditionally, active deformable models are fabricated using 3D molding of silicone rubber. Recently, 3D printing has also been used in this area due to its convenience and economy [Trimmer et al. 2015] since rigid components can be printed easily [Megaro et al. 2015] and flexible materials are also available for some types of 3D printers. Bartlett *et al.* [2015] print the shell of a jumping robot with smoothly decreasing stiffness from interior to exterior using a PolyJet 3D printer. Peele *et al.* [2015] print a pneumatic artificial muscle using an SLA 3D printer. Yap *et al.* [2016] print high force pneumatic actuators with an FDM 3D printer to take advantage of the large stretchability of 3D printing filaments. Although these active deformable models take 3D printing as the fabrication method, the models are still designed manually.

2.2 Computational fabrication

As dicussed in Sec. 2.1, the conflict between design and fabrication is the main obstacle in making active deformable models. Recently with the development of fast prototyping techniques, especially 3D printing, integrating design with fabrication has become easy, with so-called *computational fabrication*. Models with various physical properties can be designed and printed via computational fabrication: user-defined BRDF materials [Lan et al. 2013], subsurface scattering [Dong et al. 2010], haptic feedback [Torres et al. 2015], deformation behavior [Bickel et al. 2010; Pérez et al. 2015], flexible shells for molding [Malomo et al. 2016], articulated models [Bächer et al. 2012], linkage structures [Bächer et al. 2015], and mechanisms with desired motion [Zhu et al. 2012]. The data-driven method is also used to reduce the computational cost of inverse design [Chen et al. 2015]. Various elasticity can be obtained using a single material with different microstructures as proposed by Schumacher *et al.* [2015]. Anisotropic physical properties can also be achieved by designing the pattern of microstructures [Panetta et al. 2015].

For active deformable models, there are a few works that integrate design and fabrication into the computation. Bickel *et al.* [2012] build a face robot that performs the action of human speaking by using the silicone rubber material and motors as actuators. The thickness of the skin and the actuation parameters are optimized to match the input deformation sequence. However, for a pneumatic actuated model, their approach cannot be applied since the target shape is hard to achieve by pneumatically actuating homogenous soft material (see 2nd row of Fig. 7). Skouras *et al.* [2012] design and fabricate rubber balloons that approximate the user-specified shape when inflated. But their rubber balloon structure can only have a single target shape because they use the isotropic thin shell model which cannot mimic the anisotropic deformation behavior required for fitting multiple target shapes. Stanley and Okamura [2017] combine the three actuation inputs of a haptic jamming surface – node pinning, chamber pressurization, and chamber jamming to generate a sequence of actuation inputs that match the desired surface output shape. However, their method also cannot handle different target shapes and is limited to modeling functional surfaces. For deforming the shape to match targets under external forces, Skouras *et al.* [2013] optimize the distribution of two kinds of materials inside a model and the actuator positions. However, the number of unknowns could be very large, which causes the solver to be slow and easily trapped in poor local minima. Recently Musialski *et al.* [2016] propose a local subspace projection that can help to solve the shape and material optimization problem more efficiently because it can reduce the underdetermined design space to a proper design space. To enable wider motions, winding frames around soft actuators is also a good solution [Connolly et al. 2017; Polygerinos et al. 2015b], especially for bending. Our proposed frame structure could be seen as a more general and flexible frame pattern for achieving desired deformation behaviors since its layout and material distribution are associated with the desired deformations.

3 METHOD OVERVIEW

Given the rest shape of a 3D object and several target shapes represented by triangle meshes, we generate a soft pneumatic model with embedded pneumatic chambers and optimized heterogenenous frames, as well as air pressures of pneumatic chambers, so that the fabricated object could physically realize the input deformation targets with the designed air pressures. Our method consists of two main computation steps and one fabrication step:

(1) **Geometry setup**. We generate the interior geometry of the object by analyzing the deformation of the mesh. First, we assume that the object is full of soft homogenous material and deform the object to the target shapes by physical simulation. By analyzing the changes in volumes of tetrahedral elements, we compute the layout of pneumatic chambers by a greedy clustering algorithm. We then construct the frame structure based on a set of quadrilateral meshes on the boundaries of chambers whose quad edges follow the principal Green strain directions so that a frame structure with proper material properties can obtain the required anisotropic deformation behaviors.
(2) **Material optimization**. We compute the material properties of each frame segment and the air pressure in each pneumatic chamber. We formulate a nonlinear optimization problem that minimizes the shape difference between the target shapes and the simulated results with the equilibrium constraints. The optimization problem can be efficiently solved since the solution space is reduced to the material space of the frame structure and the air pressures in the chambers.
(3) **Physical realization**. The designed object is fabricated using 3D printing, then connected to our customized pneumatic control system via air channels. By controlling the inflation time of each pneumatic chamber, the fabricated object can deform to the target shapes, or perform a deformation sequence.

The technical details of these three steps are described in the following sections.

4 GEOMETRY SETUP

Given a 3D model represented by a closed triangle surface mesh $\mathcal{M}$, the user can design a set of target shapes by deforming $\mathcal{M}$ using 3D modeling software. We denote the target shapes by $\mathcal{M}_1, \ldots, \mathcal{M}_K$, which share the same mesh connectivity as $\mathcal{M}$. The computation of the interior geometry of the soft pneumatic object for these inputs includes three steps: (1) tetrahedralizing $\mathcal{M}$ and deforming the resulting tetrahedral mesh $\mathcal{S}$ to the target shapes by physical

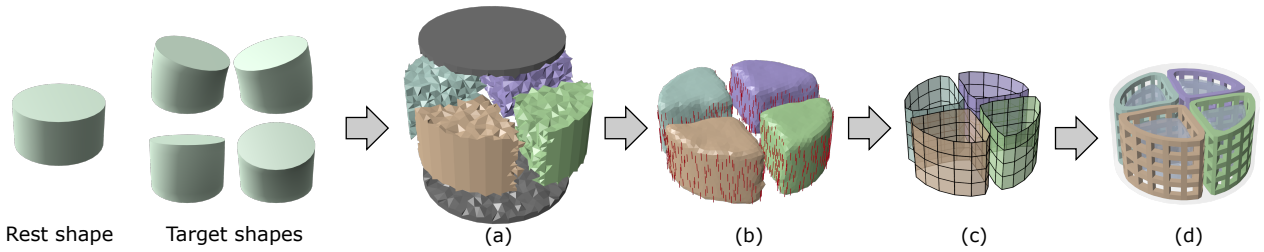


Fig. 2. Geometry setup procedure, illustrated on a bending cylinder model with four target shapes. (a) The volume of the model is discretized using a tetrahedral mesh, and segmented into regions with similar deformation behavior. (b) A pneumatic chamber is created for each region, and a directional field is calculated along the principal Green strain directions. (c) A quad mesh is generated for each pneumatic chamber along the orientation field. (d) A frame structure is generated along the edges of every quad mesh, which will serve to stiffen the object.

simulation (Sec. 4.1); (2) constructing the penumatic chambers according to the simulation results $\mathcal{S}$ (Sec. 4.2); (3) generating the frame structure around the constructed pneumatic chambers such that the frame directions follow the principal Green strain directions of deformations (Sec. 4.3).

4.1 Tetrahedral mesh generation and deformation

We first discretize the volume encompassed by $\mathcal{M}$ into a tetrahedral mesh $\mathcal{S}$ by a quality conforming Delaunay triangulation [Si 2015]. For any boundary tetrahedron whose four vertices are on the boundary surface, to avoid tangling in the target shapes, we split it into four tetrahedrons by inserting its center point, so that in the target shapes, the corresponding tetrahedrons have the freedom to be untangled. Then we deform $\mathcal{S}$ to the target shapes to obtain the target tetrahedral meshes $\mathcal{S}_1, \ldots, \mathcal{S}_K$ as follows.

Considering that the material used for soft robotics is hyperelastic and homogenous, we use the Neo-Hookean model [Sifakis and Barbič 2012] to guide the deformation. The strain energy density on a single tetrahedron $\mathbf{t}_i$ is defined as

$$\Psi_i = \frac{\mu_i}{2}\left(\mathrm{tr}(\boldsymbol{F}_i^T\boldsymbol{F}_i) - \log\left(\det(\boldsymbol{F}_i^T\boldsymbol{F}_i)\right) - 3\right) + \frac{\lambda_i}{8}\log^2\left(\det(\boldsymbol{F}_i^T\boldsymbol{F}_i)\right).$$

Here $\boldsymbol{F}_i$ is the deformation gradient, i.e., the linear transformation part of the affine transformation from the resting-state tetrahedron $\mathbf{t}_i$ to its deformed version $\hat{\mathbf{t}}_i$; $\mathrm{tr}(\cdot)$ the matrix trace operator. The parameters of the Neo-Hookean model can be derived from Young's modulus E_i and Poisson's ratio ν_i associated with the material by (1):

$$\mu_i = \frac{E_i}{2(1+\nu_i)},\ \lambda_i = \frac{E_i}{3(1-2\nu_i)}. \tag{1}$$

By summing the strain energy over the deformed tetrahedral mesh, the total elastic energy of the model is:

$$E_{\mathtt{elas}} = \sum_{\mathbf{t}_i \in \mathcal{S}} \Psi_i \,\mathrm{vol}(\mathbf{t}_i).$$

Here $\mathrm{vol}(\mathbf{t}_i)$ is the volume of $\mathbf{t}_i$.

For deforming S to the target pose $\mathcal{M}_j, j \in \{1, \ldots, K\}$, we assume there exist exterior forces on the boundary vertices and the volume is filled with the softest material available in our fabrication process. This artificial assumption gives us guidance for estimating the layout of chambers as shown in Sec. 4.2. The frame layout and material optimization in Sec. 4.3 and Sec. 5 also help the designed structure to reach the desired poses as closely as possible when real physical simulation is taken into consideration. Thus by minimizing $E_{\mathtt{elas}}$ with respect to the positions of interior tetrahedral vertices while constraining boundary vertices at their target positions in $\mathcal{M}_j$, we can obtain the deformed tetrahedral mesh $\mathcal{S}_j$. Note that when the minimum of $E_{\mathtt{elas}}$ is reached, the mesh is also at the force equilibrium state. In our implementation, we use the Newton method to solve this minimization problem. During the optimization, we add a multiple of the identity matrix to handle the non-positive Hessian [Nocedal and Wright 2006] and use linear search to avoid tangled tetrahedrons.

To accelerate the simulation, we use Laplacian smoothing to obtain the interior vertex positions while fixing the boundary vertices at the target shape and untangle the resulting mesh by a state-of-the-art tetrahedral untangling technique [Escobar et al. 2010], then provide the untangled mesh as the initialization to our optimization. By the above physical-based deformation, a series of tetrahedral meshes $\mathcal{S}_1, \ldots, \mathcal{S}_K$ can be obtained efficiently.

4.2 Chamber construction

The tetrahedral meshes $\mathcal{S}_1, \ldots, \mathcal{S}_K$ computed in Sec. 4.1 provide the deformation behaviors when the object is made of a single soft material and driven by exterior forces. By analyzing the volume changes of each tetrahedron, we propose a segmentation-based method to construct pneumatic chambers. The principles of chamber construction are simple: (1) regions with large volume changes should become chambers; otherwise, the soft material on those regions cannot bear large stretch; (2) regions with increased volumes and decreased volumes should be clustered to different chambers for separate pneumatic control.

Detecting rigid deformed regions. We first detect the tetrahedrons in $\mathcal{S}$ that undergo rigid motion in the deformations. A naïve way is to check whether the deformation of a tetrahedron is close to a rigid transformation. However, this kind of detection is too local and it is hard to find continuous regions. We propose a greedy clustering algorithm to find rigid deformed regions:

(1) for every target tetrahedral mesh $\mathcal{S}_j$, we randomly pick a seed tetrahedron $\hat{\mathbf{t}}_l$ and grow this region greedily by merging tetrahedron $\hat{\mathbf{t}}_m$ that is adjacent to this growing region if the Frobenius norm of the difference of the linear transformations between them is smaller than a threshold, i.e., $\|\boldsymbol{F}_l - \boldsymbol{F}_m\|_F^2 \leq 0.03$. We keep seeding and growing until no tetrahedron is left.

(2) for each cluster in $\mathcal{S}_j$, if the mean of its tetrahedron's linear transformations $\overline{F}$ is close to a rotation transformation, i.e. $\max(|\sigma_{\min}^{-1} - 1|, |\sigma_{\max} - 1|) \leq r_{\texttt{thres}}$, we label the tetrahedron as *rigid*. Here $\sigma_{\min}, \sigma_{\max}$ are the minimal and maximal singular values of $\overline{F}$. In our experiments, we set $r_{\texttt{thres}} = 0.01$.
(3) For any tetrahedron in $\mathcal{S}$, if all its deformed versions in $\mathcal{S}_1, \ldots, \mathcal{S}_K$ have the label *rigid*, we label it as *always rigid* and will use the hardest material to fill it in the fabrication.

Considering that in some cases parts of the boundary facets of all the target shapes have no stretch or compression, we also need to keep them unchanged. So similar to the above tetrahedron clustering, we perform the same clustering on boundary triangle facets. For any triangle in $\mathcal{S}$ which is labeled as *always rigid*, we also label the tetrahedron that contains the triangle as *always rigid* and fill it with the hardest material too. In Fig. 2(a), the detected *always rigid* region is rendered in gray color for a bending cylinder model.

Chamber initialization. After identifying the rigid regions, we initialize the chambers by clustering the tetrahedrons in the remaining regions according to their volume changes. To this end, for each tetrahedron $\mathbf{t}_i$ in $\mathcal{S}$ which is not labeled *always rigid*, we first compute its volume changes between the rest shape to each target shape: $\Delta_i^{(j)} := \text{vol}(\hat{\mathbf{t}}_i^{(j)}) - \text{vol}(\mathbf{t}_i), j = 1, \ldots, K$, where $\hat{t}_i^{(j)}$ is $\mathbf{t}_i$'s corresponding tetrahedron in $\mathcal{S}_j$. We then assign a K-dimensional descriptor vector $\mathbf{D}(\mathbf{t}_i)$ to $\mathbf{t}_i$ whose k-th entry is chosen from $\{1, 0, -1\}$, where 1 indicates that $\mathbf{t}_i$'s volume increase is the largest at the k-th target pose, -1 means that $\Delta_i^{(j)}$ is negative, and 0 for others. For tetrahedrons whose k-th entry of $\mathbf{D}$ are 1, their volume grows towards the k-th target shape and thus should be inflated and controlled separately. Therefore, we group the tetrahedrons with the same label as a cluster. Since the deformation is smooth inside the object volume, the neighboring tetrahedrons always share the same label and thus are grouped in one cluster. For a cluster that contains several isolated regions, we treat each isolated region as an individual cluster. In this way, we obtain a set of clusters, each of which is a chamber candidate.

In practice, we found that this cluster scheme generates many small chamber candidates. Since an air tube needs to be connected to each chamber, it is difficult to fabricate an object that includes many small chambers. We thus remove the small chamber candidates by merging them to adjacent clusters. Specifically, for each cluster C, we compute the ratio of its volume in the rest shape with respect to the total volume of all the non-rigid regions: $V_r := \frac{\sum_{\mathbf{t}\in C} \text{vol}(\mathbf{t})}{\sum_{\mathbf{t}\notin rigid} \text{vol}(\mathbf{t})}$. If V_r of a cluster is smaller than a threshold (0.02 in our current implementation), we merge it to a neighboring cluster that has the maximal volume among all the neighboring clusters. For all small clusters, we execute this merging process according to descending order of V_r until no cluster can be merged.

Pneumatic deformation validation. Due to the pneumatic principle, there must exist chambers whose volumes increase during the deformation. We examine all the chamber clusters for whether one of its volumes is increased from the rest state to the target state. If there is no such cluster, the target deformations cannot be realized by the soft pneumatic model and we ask the user to redesign the target shapes. For example, a pure twisted cube (Fig. 8) cannot be achieved pneumatically, so we need to extrude the cube as well when designing the target shape.

We also check the deformation on the boundary facets of $\mathcal{S}$. For a boundary facet, denote its deformation gradients with respect to all the targets by $\boldsymbol{F}_1, \ldots, \boldsymbol{F}_K$. If the principal Green strain directions derived from these deformation gradients are very different (we set the maximum angular deviation of directions to 20 degrees in our experiments), we label the facets as *violated*, since this would result in a violation in frame structure generation (Sec. 4.3). If the total area of such *violated* facets is greater than 20% of the surface area, we also require the user to redesign the target shapes.

Chamber construction. Finally, we turn the resulting clusters into chambers. Regarding each cluster as a sub tetrahedral mesh, we extract its boundary facets to form the boundary surface of the chamber. Since the extracted boundary surface is usually non-smooth and non-manifold, we sample points from it and use Poisson reconstruction [Kazhdan et al. 2006] to generate a relatively smooth manifold mesh. The region surrounded by the mesh is the pneumatic chamber.

Chamber refinement. In practice, the thickness between chambers affects the stretchability of the object. If the thickness is too small, the regions between chambers may easily break under pressure. We experimentally find that $6mm$ is a safe thickness value and is good for embedding the frame structure (which will be introduced later). We use Laplacian deformation [Sorkine-Hornung et al. 2004] to refine the boundaries of the chambers. The refinement procedure is as follows.

(1) For each boundary vertex $\mathbf{v}$ of a chamber, we find its nearest point $\mathbf{p}$ on the neighboring chamber or the boundary surface $\mathcal{S}$. If $l := \|\mathbf{v}-\mathbf{p}\| \leq 6mm$ (the safe thickness between chambers), we let $\mathbf{v}^0 := \mathbf{v} + \frac{6-l}{2} \frac{\mathbf{v}-\mathbf{p}}{\|\mathbf{v}-\mathbf{p}\|}$ be the target position of $\mathbf{v}$ in the Laplacian deformation; otherwise the vertex is fixed in the deformation.
(2) We apply Laplacian deformation to each chamber surface.
(3) We repeat steps (1)&(2) until a safe thickness is achieved.

Fig. 2(b) illustrates the four constructed chambers. Alternative refinement strategies can be applied instead, such as computing a self-intersection free offset surface.

4.3 Frame structure generation

If a soft object is fabricated using a homogeneous material, which reveals isotropic stretchability, it may become very fat locally when inflated, which does not conform to the desired shape. To control the deformation when inflating, it is necessary to use multiple materials to achieve anisotropic behaviors. One way to determine the material property is to treat the material properties of all the tetrahedrons as unknowns and solve a nonlinear optimization problem [Skouras et al. 2013]. However, the problem could contain a lot of variables and the system is under-determined if there are no regularization terms in the objective function, thus it would be costly to solve the problem and regularization terms must be included. Inspired by the fiber structure used in soft robotics and the truss structure used in architecture, we choose to embed a frame structure around the chamber boundary to achieve anisotropic deformation behaviors.

As a design criterion of the frame structure, the frame should follow the principal Green strain directions as much as possible so that it can support the major interior forces. Since the principal Green strain directions are orthogonal, they naturally form a cross field. We use a field-guided approach [Bommes et al. 2009; Ebke et al. 2013] to generate a quad mesh as the base of our frame structure. Since there are possibly several target shapes, the Green strain tensor associated with the boundary triangles of a chamber may be not unique. But note that since the chamber region is formed by tetrahedrons with similar volume change behaviors across different deformations, the deformations on the chamber boundary cannot have a large difference. Therefore, we simply pick the one with the largest eigenvalues. Fig. 2(b) shows cross fields on a bending cylinder model.

Once the quad mesh is generated, we offset it outwards to the boundary by $1.5mm$ and build a solid wireframe on the quad mesh with thickness $1mm$. The wireframe is generated as follows: for each quad edge with length l_e, a rectangular solid shape with thickness $1mm$ and length $0.8l_e$ along the edge is generated; on each quad vertex, the surrounding points of the rectangular solid shapes form a convex hull; the union of rectangular solid shapes and convex hulls form the frame structure. Fig. 2(c&d) show the quad mesh and the generated frame structure. Here, note that there is no need to generate frames on the region between chambers and *always rigid* regions.

There are two parameters in our frame structure generation: (1) the expected quad edge length l_e in quad meshing defines the density of the frames; (2) the wireframe radius r defines the thickness of frames. In practice, we find that $l_e = 6mm$ and $r = 1mm$ are good choices for minimizing shape distortion during deformation. In Sec. 8 we provide comparisons with other choices.

5 MATERIAL OPTIMIZATION

With the geometric structure of the soft pneumatic model introduced in the previous section, in the following we describe the material space and our soft pneumatic model in Sec. 5.1 and show how to assign proper material properties to the model for achieving the desired deformations by a physically-based optimization in Sec. 5.2.

5.1 Material space and pneumatic model

Tetrahedral mesh generation. Since the boundary surfaces of chambers and the frame structure are not consistent with the original tetrahedral mesh, we need to regenerate the tetrahedral mesh for the model. With the TetGen software [Si 2015], we compute a quality Delaunay triangulation conforming to the boundary surfaces of chambers, the boundary surface of the frame structure and the surface of $\mathcal{M}$. The generated tetrahedral mesh is denoted by $\hat{\mathcal{S}}$. We transfer the *always rigid* label from $\mathcal{S}$ to $\hat{\mathcal{S}}$ by labeling tetrahedrons of $\hat{\mathcal{S}}$ whose centers are inside the labeled tetrahedrons of $\mathcal{S}$. The labeling process should leave the tetrahedrons inside the frame structure untouched since the material properties of these tetrahedrons will be optimized.

Material model. To achieve the desired deformation behaviors, the material properties of all of tetrahedrons in $\mathcal{S}$ should be specified. Consider that the material used in actual fabrication is a blend of two base materials with the same Poisson ratio: the soft material (Young's modulus $E_{\mathtt{soft}}$) and the stiff material (Young's modulus $E_{\mathtt{stiff}}$), we classify the tetrahedrons into three categories and set their material model as follows.

- For *always rigid* tetrahedrons, we use the stiff material for them, i.e., the hardest material to avoid isometric distortion.
- Inside the frame structure, there are two types of tetrahedrons: tetrahedrons whose centers are in the rectangular solid region defined by a quad edge, and tetrahedrons whose centers are in the convex hull region defined by a quad vertex. We group tetrahedrons according to the quad edge index and quad vertex index. Each group is assigned to a same blended material whose Young's modulus is defined as $E_{\mathtt{blend}} = (1-\beta)E_{\mathtt{soft}} + \beta E_{\mathtt{stiff}}$. Here each group has its own β for the later optimization.
- For the remaining tetrahedrons, we use the soft material for them, i.e. make them as soft as possible.

Pneumatic model. We also use the Neo-Hookean model to study the pneumatic model. Different from Sec. 4.1, there is no exterior force on the boundary surface and the model is deformed by injected air. In our system, the air pressure can be controlled and is kept fixed for each target. For each pneumatic chamber C_i, we denote V_i as its volume along a deformation and p_i as the corresponding air pressure. Also, we denote V_0 as the volume surrounded by the outer surface and p_0 the environment air pressure, which is a standard atmosphere in our experiments. Let $\boldsymbol{X}$ denote the rest positions of the vertices of the tetrahedral mesh, $\boldsymbol{x}$ the deformed vertex positions, and $\boldsymbol{\beta}$ the set of Young's modulus interpolating parameters of the frame structure. Define $\boldsymbol{p} = [p_1, p_2, \cdots]^T$ and $\boldsymbol{V} = [V_1, V_2, \cdots]^T$. The total elastic energy of our pneumatic system is:

$$E_{\mathtt{elas}}(\boldsymbol{x}, \boldsymbol{\beta}) := \sum_{\mathbf{t}_i} \Psi(\boldsymbol{F}_i(\boldsymbol{x}); \boldsymbol{\beta}) \,\mathrm{vol}(\mathbf{t}_i).$$

The total potential energy of the system is:

$$E_{\mathtt{sys}}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{\beta}) = E_{\mathtt{elas}}(\boldsymbol{x}, \boldsymbol{\beta}) - \boldsymbol{p}^T \boldsymbol{V}(\boldsymbol{x}) + p_0 V_0(\boldsymbol{x}).$$

When the system is in quasi-equilibrium, $\boldsymbol{x}$ satisfies (2):

$$\boldsymbol{x}_{\mathtt{eq}} = \operatorname*{argmin}_{\boldsymbol{x}} E_{\mathtt{sys}}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{\beta}). \tag{2}$$

Namely $\boldsymbol{x}_{\mathtt{eq}}$, $\boldsymbol{\beta}$ and $\boldsymbol{p}$ satisfies nonlinear constraint (3):

$$\boldsymbol{f} := \left(-\frac{\partial E_{\mathtt{elas}}}{\partial \boldsymbol{x}} + \boldsymbol{p}^T \frac{\partial \boldsymbol{V}}{\partial \boldsymbol{x}} - p_0 \frac{\partial V_0}{\partial \boldsymbol{x}} \right) \Bigg|_{\boldsymbol{x}=\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta})} = \mathbf{0}. \tag{3}$$

For a given rest state $\boldsymbol{X}$ with material parameter $\boldsymbol{\beta}$, the vertex positions become $\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p}, \boldsymbol{\beta})$ when inflated with air pressure $\boldsymbol{p}$ at its equilibrium. The computation of $\boldsymbol{x}_{\mathtt{eq}}$ is similar to the physical simulation described in Sec. 4.1. In our work, we take the quasi-static assumption that the model is inflated gradually so that at any moment the system is always at its equilibrium.

5.2 Numerical optimization

The goal of our work to let the soft pneumatic model reach the desired shapes while maintaining a state of equilibrium. We formulate the problem as a constrained optimization problem: the objective should measure the difference between the deformed shape and the target, with the constraint of force equilibrium. Besides the basic

objective, we also would like to minimize the total work of the pneumatic device, to save energy and avoid large air pressure which may cause damage to the chambers. We design the objectives as follows.

Surface mismatch energy. We define the boundary mismatch energy $W^{(j)}_{\texttt{match}}$ as the integration of the point-wise squared Euclidean distance between the deformed shape and the target shape $\mathcal{M}_j$ over the surface. Denote the boundary vertices of $\mathcal{S}$ by $\boldsymbol{X}_b$, their associated areas by $\boldsymbol{s}_b$(1/3 area of their one-ring neighborhood), its deformed positions by $\boldsymbol{x}^{(j)}|_b$, and its target positions by $\boldsymbol{X}^{(j)}|_b$. $W^{(j)}_{\texttt{match}}$ can be formulated as:

$$W^{(j)}_{\texttt{match}}(\boldsymbol{x}^{(j)}) := (\boldsymbol{x}^{(j)}|_b - \boldsymbol{X}^{(j)}|_b)^T \operatorname{diag}(\boldsymbol{s}_b)(\boldsymbol{x}^{(j)}|_b - \boldsymbol{X}^{(j)}|_b).$$

Total work energy. For our pneumatically actuated model, all the work is done by air pumped into the pneumatic chambers. Assume that during the inflation procedure, the air pressure changes along path $\boldsymbol{p}(s), s \in [0,1]$. Taking the quasi-equilibrium assumption, we can get the deformation path with respect to the target $\mathcal{M}_j$:

$$\begin{aligned}\boldsymbol{x}^{(j)}(s) &= \boldsymbol{x}_{\texttt{eq}}(\boldsymbol{p}^{(j)}(s), \boldsymbol{\beta}),\\ \boldsymbol{V}^{(j)}(s) &= \boldsymbol{V}(\boldsymbol{x}^{(j)}(s)),\\ E^{(j)}_{\texttt{elas}}(s) &= E_{\texttt{elas}}(\boldsymbol{x}^{(j)}(s), \boldsymbol{\beta}).\end{aligned}$$

So the whole work accumulated is

$$\begin{aligned}\int_{s=0}^{1} (\boldsymbol{p}^{(j)} - \boldsymbol{p}_0)^T \,\mathrm{d}\boldsymbol{V}^{(j)}(s) &= \int_{s=0}^{1} (\boldsymbol{p}^{(j)} - \boldsymbol{p}_0)^T \frac{\partial \boldsymbol{V}^{(j)}}{\partial \boldsymbol{x}} \,\mathrm{d}\boldsymbol{x}^{(j)}(s)\\ &= E^{(j)}_{\texttt{elas}}(s) + p_0 \left(V^{(j)}_0(s) - \sum_i \boldsymbol{V}^{(j)}_i(s) \right)\Bigg|_0^1.\end{aligned}$$

Here $\boldsymbol{p}_0 := [p_0, p_0, \cdots]^T$ denotes the initial air pressure on all the chambers. Since $V^{(j)}_0(s) - \sum_i \boldsymbol{V}^{(j)}_i(s)$ is the volume of the material, which changes little since Poisson's ratio effect can be ignored, so the total pneumatic work is roughly equal to the total elastic energy.

Numerical optimization. We aggregate the mismatch energies and total work energies to form the objective function. The minimization is formulated as follows:

$$\begin{aligned}\min_{\boldsymbol{\beta}, \boldsymbol{p}^{(1)}, \ldots, \boldsymbol{p}^{(K)}} \quad & \sum_{j=1}^{K} W^{(j)}_{\texttt{match}} \left(\boldsymbol{x}_{\texttt{eq}}(\boldsymbol{p}^{(j)}, \boldsymbol{\beta}) \right) +\\ & \alpha \sum_{j=1}^{K} \left(E_{\texttt{elas}} \left(\boldsymbol{x}_{\texttt{eq}}(\boldsymbol{p}^{(j)}, \boldsymbol{\beta}), \boldsymbol{\beta} \right) \right)^2.\end{aligned}$$

Here, $\boldsymbol{x}_{\texttt{eq}}(\boldsymbol{p}^{(j)}, \boldsymbol{\beta})$ is plugged into the objective due to the quasi-static assumption.

In our experiments, the default α is 0.001. Notice that since all the βs should be inside the interval $[0,1]$, we express them by the sigmoid function $\beta := \frac{1}{1+e^{-t}}$ to make the above minimization problem unconstrained. The initial values of all the βs are set to 0.5 and the initial air pressure is a standard atmosphere. We minimize the objective by the Gauss-Newton method. Note that after each iteration of the Gauss-Newton method, $\boldsymbol{x}^{(j)}_{\texttt{eq}}$ should be updated by solving (2). In the appendix, we provide the details of gradient computation for numerical optimization.

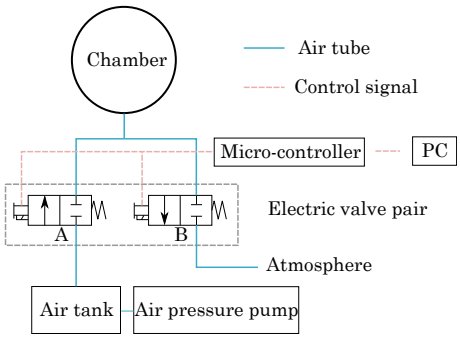


Fig. 3. The control scheme of one pneumatic chamber. A pneumatic chamber can inflate or exhaust air by changing the status of the electric valve pair, which can be controlled by PC.

6 PHYSICAL REALIZATION

Air channel setup. Once we have the interior geometry setup of the object, we need to set channels for chambers so that they can connect to the surface and are directed to air tubes for inflating or exhausting air. The thin channel (2mm radius) has little effect on the deformation behavior of the resulting object based on our observation, but the location of these channels may affect the appearance and functionality of the soft object. We manually place the channel for each chamber in our current implementation. The general placement guideline is to minimize its effects on object appearance and make the channels pass through the hardest region as much as possible. In the future we would like to develop a path finding algorithm to construct channels automatically, similar to the work of [Savage et al. 2014].

Pneumatic control system. In our implementation, we adapt the pneumatic control system designed by Hachisu and Fukumoto[2014], which contains an air pressure pump and an array of electric valves that are controlled by PC through micro-controllers. Fig. 3 shows the control scheme of one pneumatic chamber. The compressed air of the pump is directed to an air tank, which serves as an air regulator. Then the regulated compressed air is connected to the pneumatic chamber through a 2 way-2 position electric inflation valve (A in Fig. 3), controlled by PC through a micro-controller interface. When the valve is open, the pneumatic chamber will be inflated with high-pressure air quickly. The pneumatic chamber also connects to the atmosphere directly through another valve, so that when this valve is open, the pneumatic chamber will return to its rest state. To control objects with multiple pneumatic chambers, we connect each pneumatic chamber to a compressed air tank and atmosphere using an electric valve pair.

The inflated volume of a pneumatic chamber is also determined by the air pressure. Since the system has only one air pump, the air pressure should be no less than the highest pressure requirement among the pneumatic chambers. So if a pneumatic chamber needs lower air pressure, we need to switch the inflation valve off before the pressure exceeds the desired value. However, without the help of pressure sensors, it is difficult to predict the transient pressure during the inflating process, because the speed of inflation is affected by many factors, such as air pressure, the volume of the

Fig. 4. The heart model is split into three parts for gluing.

pneumatic chamber, the fractional volume of air tubes, and others. So in our current implementation, we do not control the air pressure explicitly. Instead, we record the inflation time of the chambers that can achieve the target deformation and control the inflation time to repeat the deformations. In the future, we would like to embed sensors inside each chamber, so that the pressure in each chamber can be controlled precisely and the target shape can be achieved with high accuracy.

3D printing. For fabrication, we use a Stratasys Connex350 multi-material 3D printer with two base materials: VeroBlackPlus (rigid) and TangoPlus (flexible). The 3D printer provides ten materials (including two base materials and eight interpolated materials) and their physical parameters can be found in the literature [Bartlett et al. 2015]. Among the ten materials, we simply take the one closest to the optimized material property for printing. Since the pneumatic chambers will be filled with the support material during 3D printing, we split the object into several parts, and print and glue them together. The splitting is designed manually (via boolean operations) and the basic rule is that every cut plane should pass through chambers and regions filled with the hardest material as much as possible, so that supporting material can be easily removed and the soft material region is less affected by the glue. Fig. 4 shows the splitting results of the heart model used in Fig. 1.

7 EXPERIMENTAL RESULTS

We evaluate our method in designing and fabricating six soft pneumatic objects. The number of target shapes varies from one to four. The computation is executed on a desktop PC with an Intel Core I7-6000 CPU and 16 GB memory. The statistics of computing these models are listed in Table 1. The video in our supplemental material demonstrates the dynamic behaviors of our models.

7.1 Objects with one target shape

Cylinder model. We provide three target shapes for a cylinder model, including vertical stretch, bending, and horizontal expansion (see Fig. 5). Since the principal Green strain directions of these deformations are similar, we use the same frame structure for these models and compute three soft pneumatic cylinders that can successfully achieve the desired deformations as shown in the figure. We also scan the inflated cylinders and reconstruct the mesh and compare it with the desired target shapes. The geometric difference between two surfaces are color-coded on the right of the figure. We can see that the shapes of our results are very close to the targets.

Breathing frog. We design a frog model whose target shape mimics frog breathing (Fig. 6). Our method automatically generates the buccal cavity as an empty space inside the frog head, and the material distribution is optimized so that we can physically realize the shape of a frog breathing. Since the body part is a rigid region, we can use a less expensive FDM 3D printer to print the rigid part in white color and glue all the parts together.

Gripper. A gripper-like object (Fig. 7) is made by our method. The rest shape and the target shape approximate the pick-up function of the gripper. We also demonstrate the capability of our pneumatic gripper by grasping some lightweight objects.

Twisted cube. Fig. 8 shows the fabrication result on a cube with a twisted target shape which rotates along the z-axis by 10 degrees and expands the volume a little to allow pneumatic-driven deformation. Due to the weak fatigue strength of the material, we can only produce a slightly twisted deformation. If we keep injecting air, the cube will continue to twist but the chamber will break.

7.2 Objects with multiple target shapes

Beating Heart. Pneumatic soft objects are good for simulating organs such as the heart. We design a heart model with two target shapes to mimic heart beating. Our method finds two chambers for the model, and the fabricated model under pneumatic control replicates the desired motion vividly. The design process and the fabrication result are illustrated in Fig. 1.

Bending cylinder. We add bending deformations in four directions as the target shapes for a cylinder model (see Fig. 2). Our method finds four chambers. For each target deformation, by injecting air into two chambers and releasing air from two other chambers, these target deformations can be easily achieved as shown in Fig. 9.

8 DISCUSSION ON THE FRAME STRUCTURE

In this section, we give an extended discussion on the layout of the frame structure and its material optimization. We answer the following questions with the support of our experiments.

Whether the frame structure is necessary? It is known that the air chamber presents only isotropic deformation behaviors if its surrounding material is homogenous and isotropic. When inflating air to such a chamber, the shape would grow isotropically and the target shape cannot be achieved. The 2nd row of Fig. 7 shows an example where we replace the frame structure with the homogenous material. We can see that the middle regions of the resulting gripper model would grow to a ball-like shape. In our experiments, it also fails to grasp other objects because it cannot reach the required deformation.

What is the optimal parameter setting for generating the frame structure? In Sec. 4.3 we provide our experimental optimal parameters l_e and r. Here we use an example to compare the results with other possible choices of these parameters. The model is a squared tube with a vertically-stretched target. A chamber is embedded inside the volume and the principal Green strain directions along the boundary of the chamber are the vertical and horizontal directions.

Fig. 5. Cylinder model. From left to right: target shape, the frame structure whose color encodes the material type (for frame segments with the base soft material, they are not rendered), the fabricated object at the rest pose, the inflated object, the color coding of geometric errors between the inflated object and the input target shape.

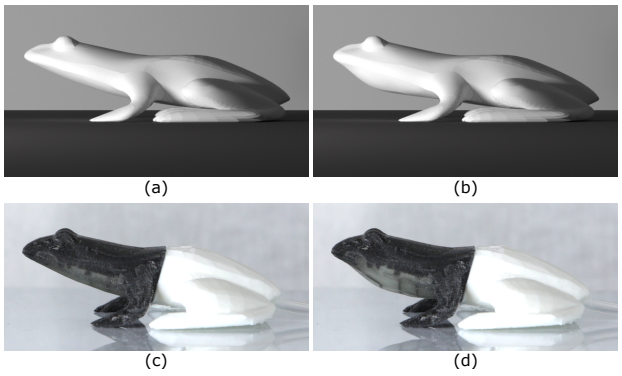


Fig. 6. A breathing frog. (a) is the rest shape of the frog and (b) is the target shape. (c) is the fabricated object at rest pose and (d) is the inflated result.

We vary the expected quad edge length and the radius of the wireframe to obtain several different frame structures. We then compute their material properties and fabricate the model. Fig. 10 shows the inflated results. We can see that frame layouts which are too coarse or too narrow cannot constrain the expanded deformation between the frames and the model with our optimal parameters has the least shape deviation compared to others.

What is the advantage of restricting the material optimization in the frame structure? As we discussed earlier, the restriction reduces the large material space, and proper material properties in the frame structure is also effective for achieving anisotropic deformation behaviors. Here we use the cylinder model as an the example to show the advantage. We compare two approaches: (1) our method (results are in Fig. 5); (2) the full-space method: treat all the material properties in the volume as unknowns and use our solver to optimize. This latter setting is similar to the work of [Skouras et al. 2013] in principle that all the material properties associated with tetrahedrons are unknowns in the optimization. We choose the same initial values of material properties for both methods for fair comparison. Considering that the number of tetrahedrons is large on this example and makes the solver extremely slow, we uniformly cluster the tetrahedrons into 1862 groups and let tetrahedrons belonging to the same group have only one material property, then optimize all these materials. We found that the optimized result of the full-space method has a larger deviation from the target than ours, and the computation time is much higher since the number of variables is still larger than ours. The numerical results can be seen from Table 1:Cylinder(stretch) II, Cylinder(bend) II, Cylinder(expand) II. Furthermore, for the expansion target deformation, the fabricated Cylinder(expand) II model cannot reach the desired shape by pneumatic deformation because the chamber is broken by the pressure due to the poor local minima found by the full-space method.

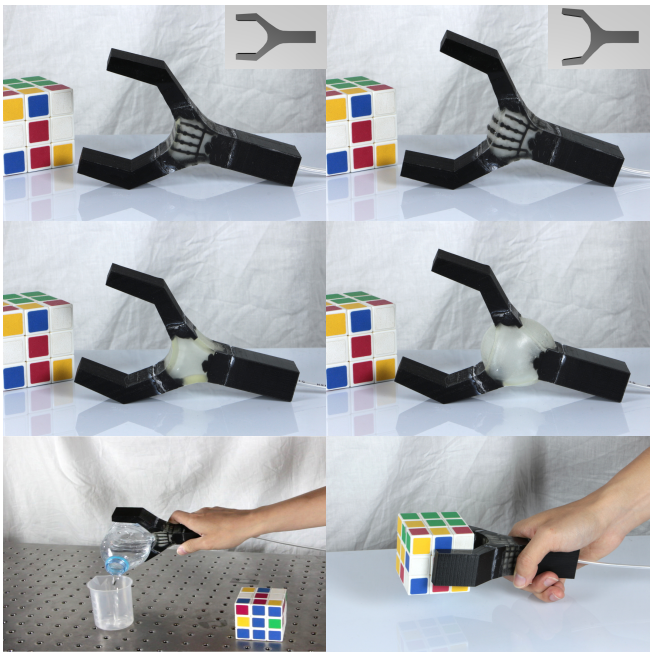
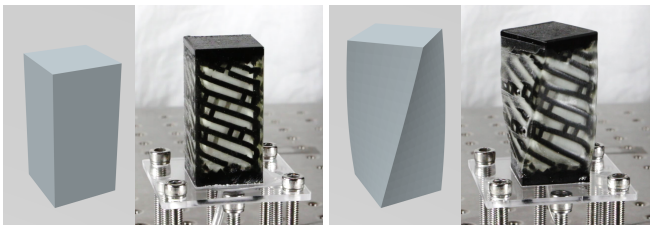


Fig. 7. Gripper model. 1st row: the rest shape and the inflated shape with the frame structure. 2nd row: the rest shape and the inflated shape without the frame structure. The joint region is inflated to a ball-like shape due to the loss of anisotropic control. 3rd row: the gripper model computed by our method can grasp lightweight objects.



(a) (b)

Fig. 8. Twisted cube model: the fabricated rest shape (a) and the inflated shape (b).

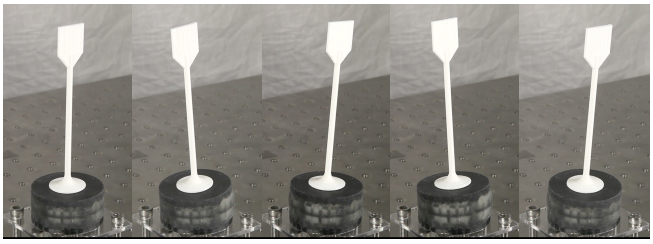


Fig. 9. Bending cylinder. From left to right: rest shape, bend-left, bend-right, bend-backward, bend-forward. The dynamic behavior of the model under pneumatic control can be observed clearly in the accompanying video.

We also examine the effects of randomized initializations. We found that the optimization based on our frame structure is very robust to different initializations and the results are always similar, but the optimization with respect to the full space is easily trapped in poor local minima. We further tried to provide our optimized result as the initialization to the full space optimization, and found that it is very close to a local minimum. This fact reveals the efficacy of using the frame structure in the design and fabrication of soft pneumatic objects.

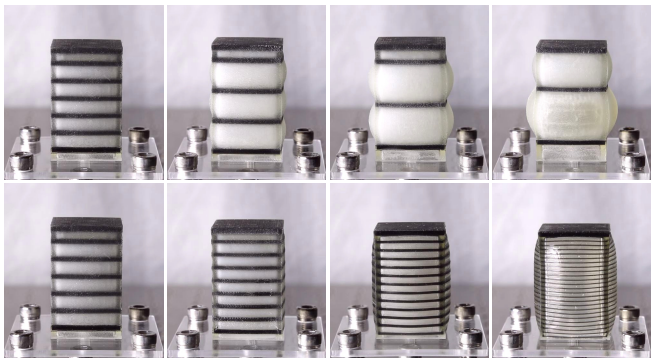


Fig. 10. Varying the parameters for generating different frame structures for a vertically stretched model. 1st row: varying l_e; 2nd row: varying r. The leftmost figures in the two rows are the same and are from our default parameter setting.

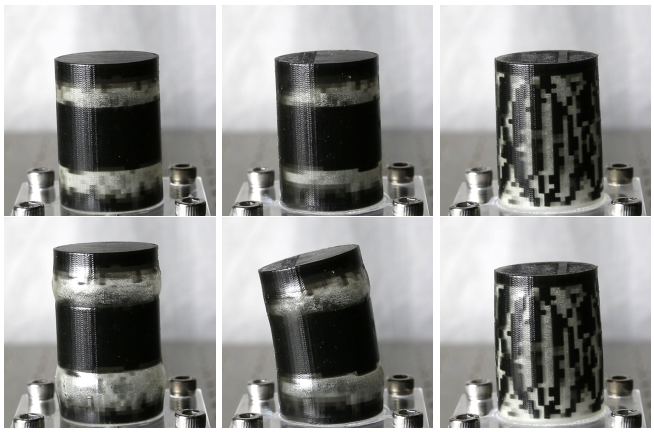


Fig. 11. Cylinder models computed by the full space method. 1st row: the rest shapes; 2nd row: the corresponding inflated shapes. The expansion cylinder breaks before reaching the target pose.

9 CONCLUSIONS

We present a novel method for design and fabrication of soft pneumatic objects with desired deformations. By analyzing the deformations of the mesh, we generate pneumatic chambers and construct the frame structure. The material properties and the air pressure are optimized by physically-based optimization for achieving desired deformation behaviors. We fabricate soft pneumatic objects by 3D printing and reproduce the input deformations with a controlled pneumatic system. The efficacy of our method is validated and demonstrated on a set of 3D objects with different target shapes.

Model	# vert	# tet	# bvert	# target	# chamber	# var-mat	# iteration	Opt. timing	Hausdorff distance(s)
Cylinder(stretch)	4173	12602	4400	1	1	661	86	8m42s	4.18e-3
Cylinder(bend)	4173	12602	4400	1	1	661	100	8m28s	4.62e-3
Cylinder(expansion)	4173	12602	4400	1	1	661	20	3m38s	2.92e-2
Cylinder(stretch) II	4173	12602	4400	1	1	1862	54	28m57s	2.22e-2
Cylinder(bend) II	4173	12602	4400	1	1	1862	53	30m39s	1.21e-2
Cylinder(expansion) II	4173	12602	4400	1	1	1862	27	23m49s	2.44e-2
Breathing Frog	5602	20270	7222	1	1	46	12	26s	1.44e-2
Gripper	8426	41449	4400	1	1	235	36	3m24s	2.24e-2
Twisted Cube	13369	67783	5074	1	1	606	45	29m04s	3.00e-2
Beating Heart	13477	57859	9950	2	2	777	36	33m31s	6.27e-3, 1.03e-2
Bending cylinder	14130	62847	6104	4	4	765	59	3h30m57s	1.95e-2, 1.99e-2, 1.99e-2, 1.98e-2

Table 1. Statistics and timings for our test models. From left to right: name of model, vertex number of the tetrahedral mesh, number of tetrahedrons, vertex number of the boundary surface, number of target shapes, number of pneumatic chambers, number of unknown material properties, iteration number, timing for optimization, Hausdorff distance between the target shape and the simulation result (with respect to bounding box diagonal). The computational time of geometry setup for each model is less than two minutes.

There are serveral research directions we would like to explore to enhance our method.

Achievable deformations. In Sec. 4.2, we provide a method to detect whether the user-provided deformation can be realized by pneumatic soft robotics. However, the method does not provide a full characterization of valid deformations and the user needs some skill and manual work to modify the target shapes. An automatic method that can guide the design of target shapes would benefit users. Another direction to improve our method is to enable active air deflation which is not considered in our current approach.

Pneumatic system. As discussed earlier, using pressure sensors will help us to control the pneumatic system automatically. It is also interesting to use small scale pneumatic devices and embed them inside the soft object, making it a self-contained soft robot.

Model fabrication. Currently we split the object into pieces to remove the supports from the 3D printed material and glue all the pieces together manually. Since this manual step is tedious for novice users, in the future, we would like to use water-dissolvable support material and wash away the support material through air channels so that the splitting step can be reduced.

Frame structure. Our frame structure is based on the field-guided quad meshing result whose edge orientations may have large deviations from the input field. As an alternative, we can use an anisotropic quad-dominant mesh [Alliez et al. 2003; Lévy and Liu 2010] for improvement. Microstructure [Schumacher et al. 2015] can also be applied to achieve more anisotropic properties.

Material blending. In our fabrication we choose from among ten available materials the one that is closest to the computed material properties. It would be possible to mix the ten materials to more closely approximate the computed material properties by halftone-like techniques.

Soft material. We find that the TangoPlus material cannot withstand large stress and may crack after repeated deformation due to its weak fatigue strength [Vu et al. 2014]. Silicone rubber has better fatigue strength than TangoPlus and our embedded frame structure can help it to achieve anisotropy property. However, in practice, we find that the frame structure may penetrate or detach from the surrounding silicone rubber after repeated deformation. These practical issues need to be addressed, and the simulation should also take the fatigue strength into the consideration.

ACKNOWLEDGMENTS

We thank Taku Hachisu and Masaaki Fukumoto for sharing the pneumatic control system for adaption, Stephen Lin for proofreading the paper and the anonymous reviewers for their constructive feedback. Kun Zhou is supported by the National Program for Special Support of Eminent Professionals of China.

REFERENCES

Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Lévy, and Mathieu Desbrun. 2003. Anisotropic polygonal remeshing. *ACM Trans. Graph.* 22, 3 (2003), 485–493.

Moritz Bächer, Bernd Bickel, Doug L. James, and Hanspeter Pfister. 2012. Fabricating articulated characters from skinned meshes. *ACM Trans. Graph.* 31, 4 (2012), 47:1–47:9.

Moritz Bächer, Stelian Coros, and Bernhard Thomaszewski. 2015. LinkEdit: interactive linkage editing using symbolic kinematics. *ACM Trans. Graph.* 34, 4 (2015), 99:1–99:8.

Nicholas W. Bartlett, Michael T. Tolley, Johannes T. B. Overvelde, James C. Weaver, Bobak Mosadegh, Katia Bertoldi, George M. Whitesides, and Robert J. Wood. 2015. A 3D-printed, functionally graded soft robot powered by combustion. *Science* 349, 6244 (2015), 161–165.

Bernd Bickel, Moritz Bächer, Miguel A. Otaduy, Hyunho Richard Lee, Hanspeter Pfister, Markus Gross, and Wojciech Matusik. 2010. Design and fabrication of materials with desired deformation behavior. *ACM Trans. Graph.* 29, 4 (2010), 63:1–63:10.

Bernd Bickel, Peter Kaufmann, Mélina Skouras, Bernhard Thomaszewski, Derek Bradley, Thabo Beeler, Phil Jackson, Steve Marschner, Wojciech Matusik, and Markus Gross. 2012. Physical face cloning. *ACM Trans. Graph.* 31, 4 (2012), 118:1–118:10.

David Bommes, Henrik Zimmer, and Leif Kobbelt. 2009. Mixed-integer quadrangulation. *ACM Trans. Graph.* 28, 3 (2009), 77:1–77:10.

Desai Chen, David I. W. Levin, Shinjiro Sueda, and Wojciech Matusik. 2015. Data-driven finite elements for geometry and material design. *ACM Trans. Graph.* 34, 4 (2015), 74:1–74:10.

Fionnuala Connolly, Conor J. Walsh, and Katia Bertoldi. 2017. Automatic design of fiber-reinforced soft actuators for trajectory matching. *PNAS* 114, 1 (2017), 51–56.

Raphael Deimel and Oliver Brock. 2016. A novel type of compliant and underactuated robotic hand for dexterous grasping. *The International Journal of Robotics Research* 35, 1-3 (2016), 161–185.

Yue Dong, Jiaping Wang, Fabio Pellacini, Xin Tong, and Baining Guo. 2010. Fabricating spatially-varying subsurface scattering. *ACM Trans. Graph.* 29, 4 (2010), 62:1–62:10.

Hans-Christian Ebke, David Bommes, Marcel Campen, and Leif Kobbelt. 2013. QEx: robust quad mesh extraction. *ACM Trans. Graph.* 32, 6 (2013), 168:1–168:10.

Yahya Elsayed, Augusto Vincensi, Constantina Lekakou, Tao Geng, C. M. Saaj, Tommaso Ranzani, Matteo Cianchetti, and Arianna Menciassi. 2014. Finite element analysis and design optimization of a pneumatically actuating silicone module for robotic surgery applications. *Soft Robotics* 2 (2014), 255–262.
José María Escobar, Eduardo Rodríguez, Rafael Montenegro, Gustavo Montero, and José María González-Yuste. 2010. SUS code: simultaneous mesh untangling and smoothing code. (2010).
Taku Hachisu and Masaaki Fukumoto. 2014. VacuumTouch: attractive force feedback interface for haptic interactive surface using air suction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 411–420.
J. Hiller and H. Lipson. 2012. Automatic design and manufacture of soft robots. *IEEE Transactions on Robotics* 28, 2 (2012), 457–466.
Daniel Robert Jamele and Norman Ellison. 2013. Motion seat systems and methods of implementing motion in seats. (2013). US Patent 8,585,142.
Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. 2006. Poisson surface reconstruction. In *Symposium on Geometry Processing*. 61–70.
Yanxiang Lan, Yue Dong, Fabio Pellacini, and Xin Tong. 2013. Bi-scale appearance fabrication. *ACM Trans. Graph.* 32, 4 (2013), 145:1–145:12.
Cecilia Laschi, Matteo Cianchetti, Barbara Mazzolai, Laura Margheri, Maurizio Follador, and Paolo Dario. 2012. Soft robot arm inspired by the octopus. *Advanced Robotics* 26, 7 (2012), 709–727.
Bruno Lévy and Yang Liu. 2010. L_p centroidal Voronoi tessellation and its applications. *ACM Trans. Graph.* 29, 4 (2010), 119:1–119:11.
Carmel Majidi. 2013. Soft robotics: a perspective–current trends and prospects for the future. *Soft Robotics* 1 (2013), 5–11.
Luigi Malomo, Nico Pietroni, Bernd Bickel, and Paolo Cignoni. 2016. FlexMolds: automatic design of flexible shells for molding. *ACM Trans. Graph.* 35, 6 (2016), 223:1–223:12.
Andrew D Marchese, Cagdas D Onal, and Daniela Rus. 2014. Autonomous soft robotic fish capable of escape maneuvers using fluidic elastomer actuators. *Soft Robotics* 1 (2014), 75–87.
Vittorio Megaro, Bernhard Thomaszewski, Maurizio Nitti, Otmar Hilliges, Markus Gross, and Stelian Coros. 2015. Interactive design of 3D-printable robotic creatures. *ACM Trans. Graph.* 34, 6 (2015), 216:1–216:9.
Philip Moseley, Juan Manuel Florez, Harshal Arun Sonar, Gunjan Agarwal, William Curtin, and Jamie Paik. 2016. Modeling, design, and development of soft pneumatic actuators with finite element method. *Advanced Engineering Materials* 18, 6 (2016), 978–988.
Przemyslaw Musialski, Christian Hafner, Florian Rist, Michael Birsak, Michael Wimmer, and Leif Kobbelt. 2016. Non-linear shape optimization using local subspace projections. *ACM Trans. Graph.* 35, 4 (2016), 87:1–87:13.
Jorge Nocedal and Stephen Wright. 2006. *Numerical Optimization* (2nd ed.). Springer.
Julian Panetta, Qingnan Zhou, Luigi Malomo, Nico Pietroni, Paolo Cignoni, and Denis Zorin. 2015. Elastic textures for additive fabrication. *ACM Trans. Graph.* 34, 4 (2015), 135:1–135:12.
Bryan N Peele, Thomas J Wallin, Huichan Zhao, and Robert F Shepherd. 2015. 3D printing antagonistic systems of artificial muscle using projection stereolithography. *Bioinspiration and Biomimetics* 10, 5 (2015).
Jesús Pérez, Bernhard Thomaszewski, Stelian Coros, Bernd Bickel, José A. Canabal, Robert Sumner, and Miguel A. Otaduy. 2015. Design and fabrication of flexible rod meshes. *ACM Trans. Graph.* 34, 4 (2015), 138:1–138:12.
Panagiotis Polygerinos, Zheng Wang, Kevin C. Galloway, Robert J. Wood, and Conor J. Walsh. 2015a. Soft robotic glove for combined assistance and at-home rehabilitation. *Robotics and Autonomous Systems* 73 (2015), 135 – 143.
Panagiotis Polygerinos, Zheng Wang, Johannes T. B. Overvelde, Kevin C. Galloway, Robert J. Wood, Katia Bertoldi, and Conor J. Walsh. 2015b. Modeling of soft fiber-reinforced bending actuators. *IEEE Transactions on Robotics* 31, 3 (2015), 778–789.
John Rieffel, Davis Knox, Schuyler Smith, and Barry Trimmer. 2014. Growing and evolving soft robots. *Artificial life* 20, 1 (2014), 143–162.
Daniela Rus and Michael T Tolley. 2015. Design, fabrication and control of soft robots. *Nature* 521, 7553 (2015), 467–475.
Valkyrie Savage, Ryan Schmidt, Tovi Grossman, George Fitzmaurice, and Björn Hartmann. 2014. A Series of Tubes: adding interactivity to 3d prints using internal pipes. In *UIST*. 3–12.
Christian Schumacher, Bernd Bickel, Jan Rys, Steve Marschner, Chiara Daraio, and Markus Gross. 2015. Microstructures to control elasticity in 3D printing. *ACM Trans. Graph.* 34, 4 (2015), 136:1–136:13.
Hang Si. 2015. TetGen, a Delaunay-based quality tetrahedral mesh generator. *ACM Trans. Math. Softw.* 41, 2 (2015), 11:1–11:36.
Eftychios Sifakis and Jernej Barbič. 2012. SIGGRAPH 2012 Course Notes: FEM simulation of 3D deformable solids: a practitioner's guide to theory, discretization and model reduction. (2012).
Melina Skouras, Bernhard Thomaszewski, Bernd Bickel, and Markus Gross. 2012. Computational design of rubber balloons. *Comput. Graph. Forum* 31 (2012), 835–844.
Mélina Skouras, Bernhard Thomaszewski, Stelian Coros, Bernd Bickel, and Markus Gross. 2013. Computational design of actuated deformable characters. *ACM Trans. Graph.* 32, 4 (2013), 82:1–82:10.
Olga Sorkine-Hornung, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Roessl, and Hans-Peter Seidel. 2004. Laplacian surface editing. In *Symposium on Geometry Processing*. 175–184.
A. A. Stanley and A. M. Okamura. 2017. Deformable model-based methods for shape control of a haptic jamming surface. *IEEE Transactions on Visualization and Computer Graphics* 23, 2 (2017), 1029–1041.
Cesar Torres, Tim Campbell, Neil Kumar, and Eric Paulos. 2015. HapticPrint: designing feel aesthetics for digital fabrication. In *UIST*. 583–591.
Barry Trimmer, Jennifer A. Lewis, Robert F. Shepherd, and Hod Lipson. 2015. 3D printing soft materials: what is possible? . *Soft Robotics* 2 (2015), 3–6.
Ivan Vu, Lindsey Bass, Nicholas Meisel, Bruce Orler, Christopher B Williams, and David A Dillard. 2014. Characterization of Mutli-Material Interfaces in PolyJet Additive Manufacturing. In *Solid Freeform Fabrication Symposium*. 959–982.
Michael Wehner, Ryan L. Truby, Daniel J. Fitzgerald, Bobak Mosadegh, George M. Whitesides, Jennifer A. Lewis, and Robert J. Wood. 2016. An integrated design and fabrication strategy for entirely soft, autonomous robots. *Nature* 536 (2016), 451–466.
Hong Kai Yap, Hui Yong Ng, and Chen-Hua Yeow. 2016. High-force soft printable pneumatics for soft robotic applications. *Soft Robotics* 3, 3 (2016), 144–158.
Lifeng Zhu, Weiwei Xu, John Snyder, Yang Liu, Guoping Wang, and Baining Guo. 2012. Motion-guided mechanical toy modeling. *ACM Trans. Graph.* 31, 6 (2012), 127:1–127:10.

A GRADIENT COMPUTATION

Since $\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p};\boldsymbol{X},\boldsymbol{\beta})$ is implicitly determined by $\boldsymbol{\beta}$ and $\boldsymbol{p}$ according to (2), namely $\boldsymbol{x}_{\mathtt{eq}}$, $\boldsymbol{\beta}$ and $\boldsymbol{p}$ satisfies nonlinear constraint (3), we have

$$\mathrm{d}\boldsymbol{f} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_{\mathtt{eq}}}\mathrm{d}\boldsymbol{x}_{\mathtt{eq}} + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}\mathrm{d}\boldsymbol{p} + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\beta}}\mathrm{d}\boldsymbol{\beta} = \boldsymbol{0}.$$

The Jacobians of the implicit function $\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta})$ are:

$$\frac{\partial \boldsymbol{x}_{\mathtt{eq}}}{\partial \boldsymbol{p}} = -\boldsymbol{K}^{-1}\boldsymbol{K}_{\boldsymbol{p}},$$
$$\frac{\partial \boldsymbol{x}_{\mathtt{eq}}}{\partial \boldsymbol{\beta}} = -\boldsymbol{K}^{-1}\boldsymbol{K}_{\boldsymbol{\beta}},$$

where $\boldsymbol{K}, \boldsymbol{K}_{\boldsymbol{p}}, \boldsymbol{K}_{\boldsymbol{\beta}}$ are the Jacobians of $\boldsymbol{f}$ w.r.t. $\boldsymbol{x}_{\mathtt{eq}}$, $\boldsymbol{\beta}$ and $\boldsymbol{p}$,

$$\boldsymbol{K} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} = -\frac{\partial^2 E_{\mathtt{elas}}}{\partial \boldsymbol{x}^2} + \sum_i p_i \frac{\partial^2 V_i}{\partial \boldsymbol{x}^2} - p_0 \frac{\partial^2 V_0}{\partial \boldsymbol{x}^2},$$
$$\boldsymbol{K}_{\boldsymbol{p}} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} = \left[\left(\frac{\partial V_1}{\partial \boldsymbol{x}}\right)^T, \left(\frac{\partial V_2}{\partial \boldsymbol{x}}\right)^T, \cdots\right],$$
$$\boldsymbol{K}_{\boldsymbol{\beta}} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\beta}} = -\frac{\partial^2 E_{\mathtt{elas}}}{\partial \boldsymbol{x}\partial \boldsymbol{\beta}}.$$

The gradient of $W_{\mathtt{match}}\left(\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta})\right)$ and $E_{\mathtt{elas}}\left(\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta}),\boldsymbol{\beta}\right)$ can be obtained by the chain rule:

$$\frac{\partial W_{\mathtt{match}}\left(\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta})\right)}{\partial \boldsymbol{p}} = \frac{\partial W_{\mathtt{match}}(\boldsymbol{x}_{\mathtt{eq}})}{\partial \boldsymbol{x}_{\mathtt{eq}}}\frac{\partial \boldsymbol{x}_{\mathtt{eq}}}{\partial \boldsymbol{p}};$$
$$\frac{\partial W_{\mathtt{match}}\left(\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta})\right)}{\partial \boldsymbol{\beta}} = \frac{\partial W_{\mathtt{match}}(\boldsymbol{x}_{\mathtt{eq}})}{\partial \boldsymbol{x}_{\mathtt{eq}}}\frac{\partial \boldsymbol{x}_{\mathtt{eq}}}{\partial \boldsymbol{\beta}};$$
$$\frac{\partial E_{\mathtt{elas}}\left(\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta}),\boldsymbol{\beta}\right)}{\partial \boldsymbol{p}} = \frac{\partial E_{\mathtt{elas}}(\boldsymbol{x}_{\mathtt{eq}},\boldsymbol{\beta})}{\partial \boldsymbol{x}_{\mathtt{eq}}}\frac{\partial \boldsymbol{x}_{\mathtt{eq}}}{\partial \boldsymbol{p}};$$
$$\frac{\partial E_{\mathtt{elas}}\left(\boldsymbol{x}_{\mathtt{eq}}(\boldsymbol{p},\boldsymbol{\beta}),\boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}} = \frac{\partial E_{\mathtt{elas}}(\boldsymbol{x}_{\mathtt{eq}},\boldsymbol{\beta})}{\partial \boldsymbol{x}_{\mathtt{eq}}}\frac{\partial \boldsymbol{x}_{\mathtt{eq}}}{\partial \boldsymbol{\beta}} + \frac{\partial E_{\mathtt{elas}}(\boldsymbol{x}_{\mathtt{eq}},\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}.$$