# Geometry-based Direct Simulation for Multi-Material Soft Robots

Guoxin Fang[1], Christopher-Denny Matte[2], Tsz-Ho Kwok[2*], Charlie C.L. Wang[1]

*Abstract*— **Robots fabricated by soft materials can provide higher flexibility and thus better safety while interacting with natural objects with low stiffness such as food and human beings. However, as many more degrees of freedom are introduced, the motion simulation of a soft robot becomes cumbersome, especially when large deformations are presented. Moreover, when the actuation is defined by geometry variation, it is not easy to obtain the exact loads and material properties to be used in the conventional methods of deformation simulation. In this paper, we present a direct approach to take the geometric actuation as input and compute the deformed shape of soft robots by numerical optimization using a geometry-based algorithm. By a simple calibration, the properties of multiple materials can be modeled geometrically in the framework. Numerical and experimental tests have been conducted to demonstrate the performance of our approach on both cable-driven and pneumatic actuators in soft robotics.**

## I. Introduction

In recent years, soft robotics has become a popular multidisciplinary research area due to its better robustness and safety. Most common designs for soft robots are realized by distributed actuation on soft materials [1]. With more *Degrees-Of-Freedom* (DOFs) than rigid robots can provide, it can better complete highly dexterous tasks like grasping [2] and detection of confined area [3]. While molding techniques are used to fabricate soft robots in the past, the advancement in 3D printing allows the fabrication of soft robots with multi-materials [4], [5], which provides a new method to control the deformation of soft manipulator to handle more complicated tasks. For example, a cable-driven soft hand shown in Fig.1 is 3D-printed with two materials having different elasticity. The fingers are in the same shape but with different material compositions. When applying the same actuation – i.e., the same length of string stretching, different deformed shapes are presented on the four fingers. In short, designing soft robots by different material compositions can achieve a variety of behaviors without changing the shape.

### A. Motivation

Soft matter and multi-material printing open up many opportunities in designing new robots. However, the high DOFs have also brought many challenges to numerical simulations. Unlike the rigid robots for which the forward and inverse kinematics can be used to compute the position of end-effector or the joint parameters, soft robots are deformable objects which can be actuated by various mechanisms.

[1]G. Fang and C.C.L. Wang are with the Department of Design Engineering and Robotics Institute, Delft University of Technology, Netherlands. G. Fang is financially supported by Chinese Scholarship Council.

[2]C.-D. Matte and T.-H. Kwok are with the Department of Mechanical, Industrial and Aerospace Engineering, Concordia University, Canada.

*Corresponding Authors – T.-H. Kwok (tszho.kwok@concordia.ca).

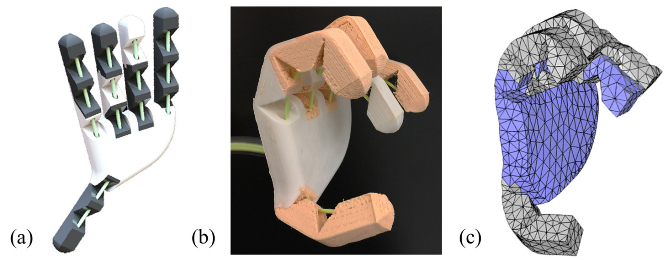


Fig. 1. A cable-driven soft hand with multiple materials. (a) The digital model designed with different material compositions on different fingers. (b) The actuated physical model fabricated by 3D printing, and the fingers have different shapes under the same actuation. (c) The simulation result generated by the proposed method.

The problem to be solved in this paper is how to predict the deformed shape of a soft manipulator fabricated by multiple materials effectively. A common technique for estimating the deformation of elastic materials is the *Finite Element Method* (FEM). However, the framework of FEM relies on the accurate input of structural loads and material properties, which is not easy to be obtained in many scenarios of soft manipulator. Actually, actuations in soft robotics are often defined by geometric variations. For instance, a cable-driven gripper actuated by motors [6] is controlled by the length change of cables. Pneumatic actuators are usually driven by the volume change in a chamber [7]. Converting these actuations into structural loads will cause unnecessary errors of approximation. Differently, we develop a novel algorithmic approach in this paper to compute the deformed shape of a soft manipulator directly.

### B. Related Work

With a good understanding of material properties and the mechanism of actuation, precise FEA can be conducted with given forces / torques. Commercial software like Abaqus and ComSol have been used in the research of soft robotics [7]. On one hand, small time-steps are needed for systems with large stiffness for simulating large deformation; on the other, it requires modeling complicated multi-material properties as well as their interaction. In order to get a fast simulation for interactive and iterative design of soft robots, Hiller and Lipson [8] developed a platform called Voxelyze that is able to generate results of dynamic simulation for multi-material soft objects. Voxel representation is used for simulating large deformation and evolutionary computation is employed to obtain optimized material distributions [9]. Nevertheless, large quantity of voxels are needed to represent models with complex shape, which will tremendously slow down the computation.

SOFA [10] is a widely used framework in the field of surgical and biomedical simulation. Based on SOFA, Duriez *et al.* [11] developed a plug-in for real-time simulation of soft robots that supports interactive deformation. Inverse design can also be conducted by an optimization based algorithm. Their algorithm uses the iterative method to solve ordinary differential equations meanwhile transferring the boundary conditions using Lagrangian multipliers. This method is fast but suffers from the problem of numerical accuracy, particularly if there is large deformation (rotations). However, one benefit of soft actuator is its capability of adapting to highly curved contact by large deformation [12], which needs to be precisely simulated for many applications.

With the help of mass-spring system, Allison et al. [13] presented a close-loop control for haptic jamming deformable surface. However, they are application-specific and may not be generalized to other soft actuators.

There is research [14]–[16] that applies geometry-based algorithm in optimization. Different from using constrained nonlinear optimization [17], the geometry-based numerical computation can converge in a few iterations. This paper extends this idea to the simulation for soft robots with multiple materials.

### C. Contribution

Based on the observation that many actuations in soft robotics are directly related to geometry, we hypothesize that *a geometry-based simulation gives better convergence and accuracy than the mechanics methods*. Here higher accuracy means the results of simulation are closer to the physical tests. In addition, the simulation should be able to handle multiple materials. To test this hypothesis, we apply the technique from geometric computing to formulate a framework that can directly model and simulate soft robots based on geometry.

To realize our framework, three research questions need to be investigated: 1) how to convert the mechanical analysis to a geometric problem, 2) how to apply different actuations in the simulation, and 3) how to model the material properties geometrically. Answering these questions brings the contributions of our paper as follows:

- A geometric optimization to minimize the elastic energy with reference to shape variations is formulated to mimic the physical phenomenon during deformation.
- The geometric constraints of actuations are modeled by a type of element, which can be directly integrated in the optimization.
- A simple calibration method is developed to learn the relationship between material properties and shape parameters, which are used in our framework to simulate the deformation of objects with multiple materials.

Our framework is direct and efficient, and its functionality will be demonstrated and verified on cable-driven and pneumatic soft robots with multiple materials.

The rest of this paper is organized as follows. Section II introduces our framework of the geometry-based simulation. After that, Section III discusses how to formulate the actuations as geometric constraints, which is followed by presenting a calibration method for multi-material simulation in Section IV. The experimental tests and validation are given in Section V, and our paper ends with the conclusion and discussion in Section VI.

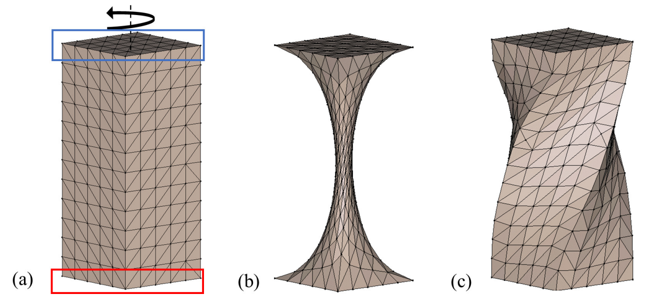


Fig. 2. (a) A bar is being twisted by 90°. (b) The result without preserving its shape looks unreal. (c) By preserving the original shape of each element, the numerical simulation can mimic the physical phenomenon.

## II. Geometry-based Simulation

When different boundary conditions or external loads are applied to deform an object $\mathcal{M}$, the elastic energy is transferred by the corresponding work and distributed internally in the materials of $\mathcal{M}$. Here the elastic energy is caused by the shape deformation, which can be evaluated from the strains (i.e., local deformations throughout $\mathcal{M}$). In this sense, the total elastic energy can be minimized when the original shape of $\mathcal{M}$ is preserved as much as possible. To mimic this physical phenomenon, this section formulates a geometry-based simulation as an optimization problem to preserve a target shape while satisfying the imposed boundary conditions and actuation constraints.

Assume a soft robot is digitally represented by a volumetric mesh $\mathcal{M}_s = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ stand for the sets of vertices and elements on the mesh. The shape of the $i$-th element is defined as $\mathbf{V}_i = [\mathbf{v}_1 \ \mathbf{v}_2 \ \dots \ \mathbf{v}_n]$, where $n$ is the number of vertices of the element, e.g., $n = 4$ for a tetrahedron in this paper. Let $\mathbf{V}_i^t = [\mathbf{v}_1^t \ \mathbf{v}_2^t \ \dots \ \mathbf{v}_n^t]$ be the target shape that the element would preserve, then the optimization can be formulated as minimizing the difference between $\mathbf{V}_i$ and $\mathbf{V}_i^t$ for all $m$ elements. That is defined by an energy as

$$E = \sum_{i=1}^{m} d(\mathbf{V}_i, \mathbf{V}_i^t). \tag{1}$$

To measure the difference $d(\cdot, \cdot)$ of two shapes, they have to be properly aligned in terms of both position and orientation. Therefore, both the shapes are centered at the origin and a rotation is applied to match $\mathbf{V}_i^t$ with $\mathbf{V}_i$, such that the above energy can be further defined as

$$E = \sum_{i=1}^{m} \omega_i ||\mathbf{N}\mathbf{V_i} - \mathbf{R_i}(\mathbf{N}\mathbf{V_i^t})||_F^2. \tag{2}$$

$\omega_i$ is a weight for each element, which is normally set as the element's volume. $|| \cdot ||_F$ is the Frobenius norm, and $\mathbf{N}$ is a

$4 \times 4$ matrix to transfer an element's center to origin:

$$\mathbf{N}(i,j) = \begin{cases} 3/4 & \text{if} \quad i = j \\ -1/4 & \text{if} \quad i \neq j \end{cases} \quad \forall i,j \in (1,2,3,4).$$

There are two sets of unknowns in Eq.(2): one is the positions of vertices $\mathbf{V_i}$, and the other one is the rotation matrices $\mathbf{R_i}$. They are dependent on each other, which leads to a nonlinear system. A two-step iterative method [16] is conducted to solve this problem. Specifically, one set of unknowns is fixed while solving the other set, and the fixed set is switched alternatively between two neighboring steps. This two-step method has been proven to be very efficient. When the target shape $\mathbf{V}_i^t$ for each element has been defined, this framework of optimization can deform the element shape $\mathbf{V}_i$ to approach its target shape as much as possible. A demonstration of its functionality is given in Fig.2, where a bar in (a) is twisted by 90° (b) without and (c) with preserving the target shapes $\mathbf{V}_i^t$. The target shape of each element in this example is set as its original shape shown in (a). When applying this optimization based deformation framework for the simulation of soft robots, we need to tackle the problem of defining an appropriate target shape so that different actuations and materials can be incorporated. These will be discussed in the following sections.

## III. Actuation as Geometry Constraints

Soft robots are deformed by external actuations such as shortening the length of a cable or expanding the interior volume of a chamber. These actuations actually are the geometric hard constraints $\mathcal{C}$ for the simulation framework, which leads to a constrained optimization problem as

$$\min_{\mathcal{V},\mathcal{R}} E \qquad \text{subject to } \mathcal{C}. \tag{3}$$

When solving such a problem by a penalty-based method such as Lagrange multiplier, the convergence problem may occur especially when the initial value is not feasible (i.e., the constraints $\mathcal{C}$ are not satisfied at the initial value of numerical computation). To solve this problem efficiently, we formulate an actuation as a special type of element and use the target shape $\mathbf{V}_i^t$ to model these hard constraints, which can be seamlessly integrated to our geometry-based simulation framework. Details about how to convert physical actuations into target shapes for both the cable-driven and the pneumatic actuations are presented below.

### A. Cable-driven actuation

Figure 3(a) demonstrates a common model of cable-driven soft gripper, which is a rectangular bar with a few gaps located in one side. The gripper is fixed at one end, and a cable shown as a dotted line is passed through the holes along the gripper. The gripper is actuated by pulling the cable (i.e., by changing its length). As a result, it bends towards the side with gaps. To integrate the cable in the simulation, the V-shaped gaps are modeled as a set of tetrahedral elements $\tilde{\mathbf{V}}_\mathbf{1}, \tilde{\mathbf{V}}_\mathbf{2}, ..., \tilde{\mathbf{V}}_\mathbf{n}$. There is a triangular face on each of these elements that aligns with the cable. These faces called *cable-component* will be used to drive the simulation.

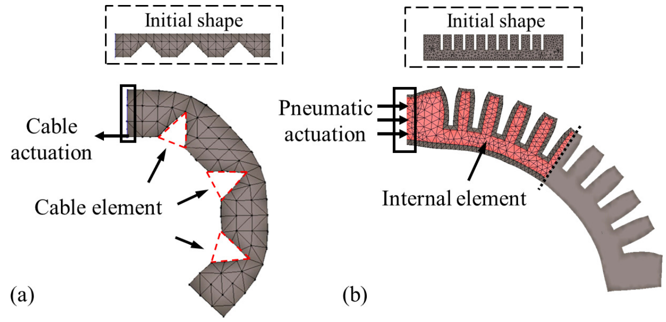


Fig. 3. The geometric constraints of actuation. (a) In cable-driven actuation, the gaps are modeled as tetrahedral elements and the edges aligned with the cable will be shortened as the cable is stretched. (b) In pneumatic actuation, the chamber are modeled by tetrahedral elements, which will expand when air is pumped in.

The total length of the cable $L$ equals to the length of the gripper that includes the rigid portions $R_d$ and the gaps $l_i$, i.e., $L = R_d + \sum_{i=1}^{k} l_i$, where $k$ is the number of gaps. Given the cable-driven actuation with a shrinking ratio $S$, the geometric constraint is defined as:

$$C_c(\tilde{\mathbf{V}}_\mathbf{1}, \tilde{\mathbf{V}}_\mathbf{2}, ..., \tilde{\mathbf{V}}_\mathbf{n}) : SL = R_d + \sum_{i=1}^{k} s l_i, \tag{4}$$

where $s$ is a local shrinking ratio for the gaps. If the local shrinking ratio $s$ of a gap is given, the target shape for its corresponding tetrahedral elements can be computed. By rotating the tetrahedral element $\tilde{\mathbf{V}}_\mathbf{i}$ to its local coordinate system (resulting in $\tilde{\mathbf{V}}_\mathbf{i}^\mathbf{L}$) where its cable-component is aligned to the $xy$-plane, and the main axis of the gap is aligned to the $y$-axis, the target shape $\tilde{\mathbf{V}}_\mathbf{i}^\mathbf{t}$ can be computed by scaling in the $x$-axis, i.e., $\tilde{\mathbf{V}}_\mathbf{i}^\mathbf{t} = [s \ \ 1 \ \ 1]\tilde{\mathbf{V}}_\mathbf{i}^\mathbf{L}$.

Note that, the input shrinking ratio $S$ is different from the local shrinking ratio $s$. The problem here is to compute the local shrinking ratio $s$ to satisfy the hard constraint $C_c$. Computing the ratio and the shapes at the same time is nonlinearly coupled and therefore hard to solve. Fortunately, the deformation is a dynamic process with a number of time steps. We can then determine the ratio during optimization. Due to the material distribution, the gaps will be optimized to different shapes. Specifically, starting from $s = 0$, a small shrinking ratio, e.g., 0.01, is added to $s$ for each incremental step in the time domain. This process is iterated until the constraint $\mathcal{C}_c$ is satisfied. Figure 3(a) shows the simulation result for a shrinking ratio $S = 0.7$.

### B. Pneumatic actuation

A pneumatic actuator usually drives soft robots by pumping pressurized air into a bellow formed by soft materials. As shown in Fig.3(b), our method is demonstrated by a commonly used tooth-shape soft gripper. The left part is fixed while pumping air along the direction of the arrows into the bellows. The internal tetrahedral elements $\tilde{\mathbf{V}}_\mathbf{1}, \tilde{\mathbf{V}}_\mathbf{2}, ..., \tilde{\mathbf{V}}_\mathbf{n}$ highlighted in Fig.3(b) are used to model the expanding behavior of air inside the bellows.

Let the volume of an element be $u_i$ for the $i$-th element,

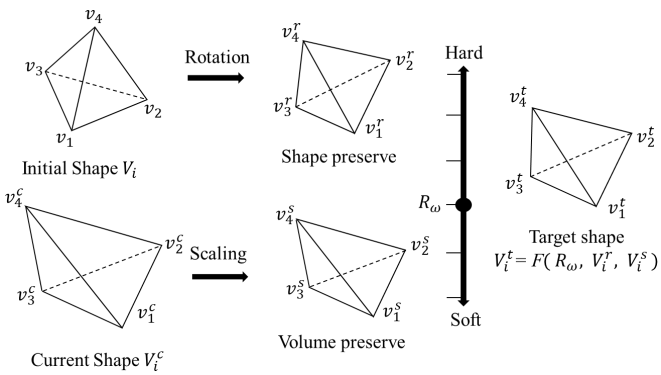


Fig. 4. (Top) The target shape for rigid material is computed by rotating the initial shape to align with the current shape. (Bottom) The target shape for extremely soft material is computed by scaling the current shape to preserve the volume of the initial shape. (Right) The shape blending method is applied to align the rigid and the soft materials, and merge their shapes using material property to define the target shape for an intermediate material.

the total volume of a bellows is then $U = \sum_{i=1}^{n} u_i$. Given the volume change, the geometric constraint for a pneumatic actuator can be described by an expansion ratio $E$:

$$C_p(\tilde{\mathbf{V}}_\mathbf{1}, \tilde{\mathbf{V}}_\mathbf{2}, ..., \tilde{\mathbf{V}}_\mathbf{n}) : EU = \sum_{i=1}^{n} eu_i \tag{5}$$

where $e$ is a local expansion ratio of the internal elements. When $e$ is defined, the target shape $\tilde{\mathbf{V}}_\mathbf{i}^\mathbf{t}$ for the tetrahedral element can be computed by $\mathbf{V}_\mathbf{i}^\mathbf{t} = e\tilde{\mathbf{V}}_\mathbf{i}^\mathbf{L}$ after centering the tetrahedron $\tilde{\mathbf{V}}_\mathbf{i}$ to origin. The process of calculating the final shape of a pneumatic actuator is similar to the cable-driven actuation discussed above, which is to add a small expansion ratio to the elements in each time step and iterate until the constraint $C_p$ is satisfied. Fig.3(b) shows the simulation result with given expansion ratio $E = 1.25$.

## IV. Shape Parameters for Multiple Materials

Under an external load, the material deforms and stores potential elastic energy. The total energy can be minimized by preserving the original shape of an object. However, if an object contains multiple materials, regions with different materials will deform in different ways. In this section, we propose a method to simulate soft objects with multiple materials by using shape parameters. The relationship between material properties and shape parameters needs to be found. One can calibrate the relationship by applying the same force to different materials and measuring how much they deform – e.g., a conventional tensile test. However, when the actuation is geometry-based, this calibration is indirect and requires an additional conversion between force and deformation. Rather than calibrating each material separately with force, we develop a simple method to calibrate the relative properties between two materials. Before that, we present how to model different material properties geometrically.

### A. Deformation with different materials

To model the different properties of materials, a simple way is to assign different weights $\omega_i$ for each element in

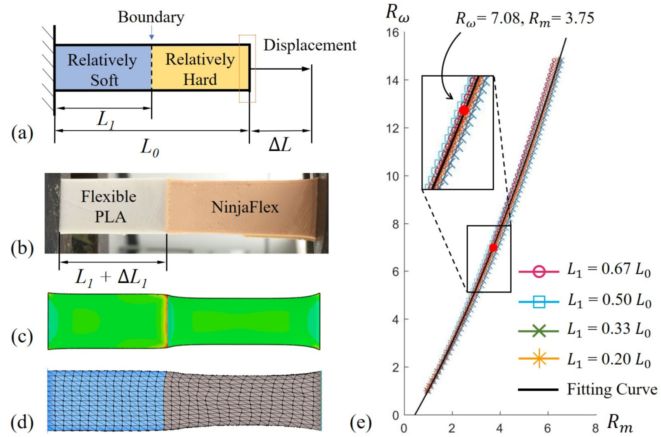


Fig. 5. Calibration of the shape parameter for simulating objects with multiple materials: (a) a multi-material bar with displacement on the right, (b) a physical elongation test on 3D printed specimen using NinjaFlex and Flexible PLA materials, (c) a simulation result by using the Abaqus FEA software, (d) the result generated by our simulation framework, and (e) the calibrated relationship between $R_m$ and $R_\omega$.

Eq.(2), and the shapes of elements with different weights will be preserved differently through the optimization. This mimics the deformation of multiple materials. However, this way of handling the material difference at the global blending step by least-square solution will lead to large approximation error. In order to gain a better control and reinforce the physical property in large deformations, we control the deformation behavior of elements at the local step by altering their target shapes, $\mathbf{V}_\mathbf{i}^\mathbf{t}$, according to different material properties. Basically, if a material is extremely hard, it will be rigid during the deformation. Respectively, an extremely soft material will deform and conform to its neighbors and external loads while preserving its volume.

A shape blending method is developed in our framework to compute the target shapes $\mathbf{V}_\mathbf{i}^\mathbf{t}$ for different materials based on their relative properties. As shown in Fig.4, the target shape of a rigid element comes from the rigid transformation of its original shape. For a soft element, its target shape comes from the current shape by scaling back to its original volume (see Fig.4). For a material in-between (e.g., with a ratio $R_\omega$), the rigid and soft target shapes are aligned and blended together using the concept of isometric morphing [18] to get the target shape as shown in the right of Fig.4.

In this way, the target shapes of elements according to different materials are properly controlled during the deformation, and thus the result of optimization will not be prone to large approximation error caused by least-square solution. The next sub-section will discuss how to determine the ratio $R_\omega$ – the shape parameter – for the relative material properties.

### B. Calibration of shape parameter

To calibrate the shape parameter $R_\omega$ for the deformation of multiple materials, we impose a displacement on a rectangular specimen at one end while fixing another end (as shown in Fig.5(a)). Without loss of generality, the specimen is fabricated with two materials A and B joined with a sharp interface. Let the length of the whole specimen be $L$ and the distance between the interface and the fixed end be $L_1$, where different values of $L_1 \in (0, L)$ are used for different specimens. When imposing a displacement $\Delta L$ at the free end of the bar, the displacement of the interface will be located at $\Delta L_1 \in (0, \Delta L)$ depending on the relative material properties between A and B. The relationship of two materials can be presented by a *material ratio* $R_m$, which is mathematically defined as

$$R_m = \frac{E_A}{E_B} = \frac{L_1(\Delta L - \Delta L_1)}{(L - L_1)\Delta L_1}, \tag{6}$$

where $E_A$ and $E_B$ are the Young's modulus of two materials with A being linked to the fixed end and B locating at the free end. To verify the quality of 3D printed specimens, we perform the elongation test on them. The results of physical tests match well with the simulation results generated in Abaqus (see Fig.5(b) and (c)). With this insight, the rest of the problem is to find the relationship between the material ratio $R_m$ and the shape parameter $R_\omega$. The basic idea is to apply different values of $R_\omega$ to run the elongation tests in our geometry-based simulation by the same setup (see Fig.5(d)). By matching our simulation results with the results of Abaqus, we can determine the value of shape parameter $R_\omega$ for two particular materials.

To calibrate the relationship for different material ratios $R_m$, we apply different values of $R_m$ in Abaqus and find the matching value of $R_\omega$ in our simulator, which is plotted in Fig.5(e). Note that samples are also generated by using different values of $L_1$ (i.e., the locations of the interface) to validate the correctness of calibration. Figure 5(e) shows that the data have a very good alignment, and a second order polynomial curve is fitted to define the relationship as

$$R_\omega = 0.114R_m^2 + 1.665R_m - 0.766 \tag{7}$$

When new materials are used, the ratio $R_m$ in Eq.(6) can be computed directly if their Young's modulus are known, or obtained through a tensile test as in Fig.5(b). Then, the shape parameter $R_\omega$ can be determined by Eq.(7).

## V. Results

The proposed method of geometry-based direct simulation has been implemented in C++ and tested on a standard PC with an Intel i7 2.4GHz CPU and 8GB RAM. All the simulations can be run in an interactive speed (i.e., 4-5 fps) with a mesh size of up to 7.5k tetrahedra, which can be seen in the supporting video of this paper. With the same configuration, the commercial FEM software Abaqus needs $1.5$ minute to compute the deformation for a single frame, i.e, our result is $45$ times faster.

The models of soft robot are digitally represented by tetrahedral meshes, and their corresponding physical objects are fabricated by Ultimaker 3 which can print two materials in a build. The two materials used in our experiments are NinjaFlex and Flexible PLA with Young's modulus 12MPa and 45MPa respectively. Therefore, $R_\omega = 7.08$ is used for our simulation. Our results are compared with the SoftRobots plug-in for SOFA [11] and also verified with physical experiments.

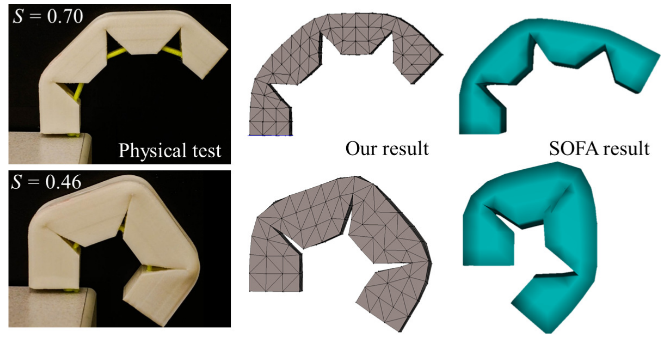

Fig. 6. Comparisons on a cable-driven gripper among physical test (left), our simulation(middle), and the simulation by the SoftRobots plug-in for SOFA [11](right).

### A. Comparison with the SoftRobots plug-in for SOFA

To compare the performances between SOFA and our framework, a cable-driven gripper with single material is used as shown in Fig.6, which is fabricated with the Flexible PLA. The top and bottom rows show two sequences of deformations at different time instants, where from left to right show the results of physical test, our simulation and SOFA. Due to the reason that the deformation accuracy is traded off for computational speed in SOFA, its results do not match with the physical tests in large deformation. Specifically, simulation starts to variate from reality when cable length change is larger than $45\%$ or chamber's volume change is greater than $30\%$. In contrast, our simulation can produce very realistic results while having a similar computational speed as SOFA.

### B. Verification

To verify the result of our simulation for multiple materials, we tested two cable-driven grippers with different material compositions. The simulation and physical results are compared visually with its dynamics in the top and the middle rows of Fig.7. The deformations are also compared quantitatively by the trajectory of three corresponding markers located on the boundary of the grippers (i.e., $P_1$, $P_2$ and $P_3$). It can be seen that both results match with the physical experiments very well. Another example has been shown in Fig.1 as a hand model compounded by a few manipulators with different material compositions. The results of simulation and physical test show a great match.

As presented in Section III, our framework can work not only for the cable-driven but also the pneumatic soft robots. One example is shown in Fig.8, and it is compared with the physical experiment by increasing the pressure of air pumped into the chamber to control bending of the gripper. From all these tests, it is easy to conclude the high accuracy of our geometry-based direct simulation framework.

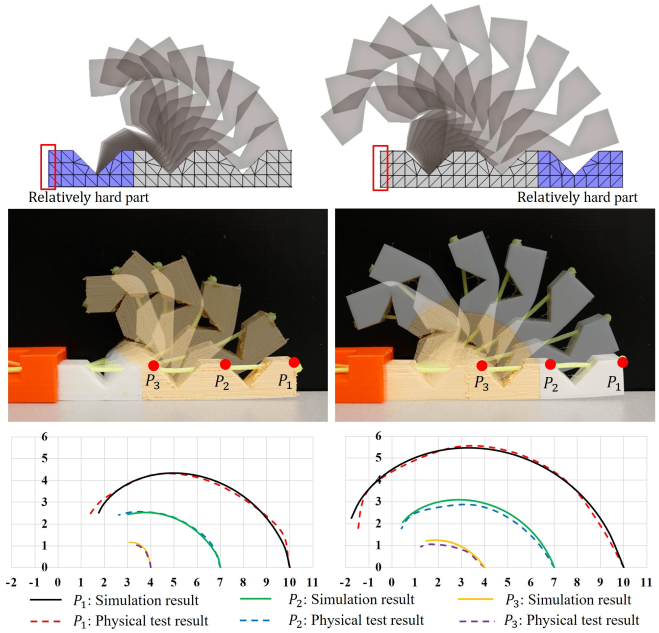


Fig. 7. Two cable-driven soft grippers (left and right) with different material distributions have different behaviors under actuation. Locations of markers determined by our simulation are well-matched with theirs in physical test.

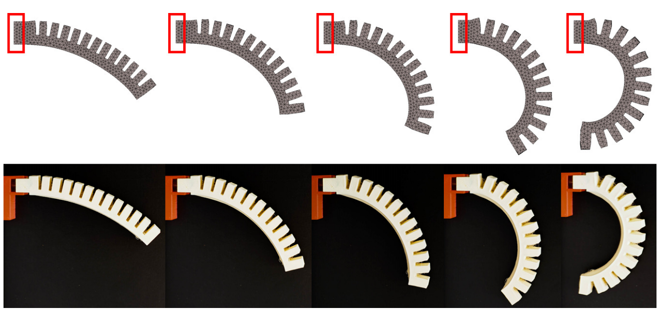

Fig. 8. Pneumatic-driven soft gripper: (top) the results of our simulation and (bottom) physical test by increasing the pressure of air pumped into the chamber. The gripper is fabricated by the NinjaFlex material.

## VI. Conclusion and Future Work

In this paper, we develop a new geometry-based simulation for soft robots. The motivation of this work comes from the observation that the current actuations of soft robots such as length shortening of cable and volume changing of chamber are based on geometry variation. In summary, we develop a geometric optimization for preserving shape during deformation with the function of representing actuations as different type of geometric constraints to be imposed on specially designed elements. Moreover, multi-material simulation is also supported by our framework with a well-designed calibration process for finding relative material properties. The experimental results support our hypothesis and verify that the proposed simulation framework is valid, direct, and promising.

We have demonstrated the framework by cable-driven and pneumatic-driven soft robots, but it will be extended to other actuations driven by geometry transformation and to model multiple-actuations. Another future work is to extend this simulation for the soft robots that are made up of more than two types of materials. One way is to introduce more degree of freedom in the calibration of shape parameters.

## References


[1] J. Hiller and H. Lipson, "Automatic design and manufacture of soft robots," *IEEE Transactions on Robotics*, vol. 28, pp. 457–466, 2012.

[2] R. Katzschmann, A. D. Marchese, and D. Rus, "Autonomous object manipulation using a soft planar grasping manipulator," *Soft Robotics*, vol. 2, pp. 155–164, 2015.

[3] A. D. Marchese, R. Katzschmann, and D. Rus, "Whole arm planning for a soft and highly compliant 2d robotic manipulator," *in IEEE Int. Conf. on Intelligent Robots and Systems*, pp. 554–560, 2014.

[4] M. Skouras, B. Thomaszewski, S. Coros, B. Bickel, and M. Gross, "Computational design of actuated deformable characters," *ACM Transactions on Graphics*, vol. 32, 2013.

[5] N. Bartlett, M. Tolley, J. Overvelde, J. Weaver, B. Mosadegh, K. Bertoldi, G. M Whitesides, and R. Wood, "A 3d-printed, functionally graded soft robot powered by combustion," *Science*, vol. 349, pp. 161–5, 2015.

[6] M. Manti, T. Hassan, G. Passetti, D. Nicolo, C. Laschi, and M. Cianchetti, "A bioinspired soft robotic gripper for adaptable and effective grasping," *Soft Robotics*, vol. 2, 08 2015.

[7] H. K. Yap, H. Yong Ng, and R. C.-H. Yeow, "High-force soft printable pneumatics for soft robotic applications," *Soft Robotics*, vol. 3, pp. 144–158, 2016.

[8] J. Hiller and H. Lipson, "Dynamic simulation of soft multimaterial 3d-printed objects," *Soft Robotics*, vol. 1, pp. 88–101, 2014.

[9] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson, "Evolving soft robots with multiple materials and a powerful generative encoding," *in Genetic and Evolutionary Computation Conference*, 2013.

[10] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and S. Cotin, "SOFA: A Multi-Model Framework for Interactive Physical Simulation." Springer, Jun. 2012, vol. 11, pp. 283–321.

[11] C. Duriez, E. Coevoet, F. Largilliere, T. Morales-Bieze, Z. Zhang, M. Sanz-Lopez, B. Carrez, D. Marchal, O. Goury, and J. Dequidt, "Framework for online simulation of soft robots with optimization-based inverse model," *in IEEE Int. Conf. on Simulation, Modeling, and Programming for Autonomous Robots*, pp. 111–118, 12 2016.

[12] R. V. Martinez, J. L. Branch, C. R. Fish, L. Jin, R. F. Shepherd, R. M. D. Nunes, Z. Suo, and G. M. Whitesides, "Robotic tentacles with three-dimensional mobility based on flexible elastomers," *Advanced Materials*, vol. 25, no. 2, pp. 205–212, 2013.

[13] A. A. Stanley and A. M. Okamura, "Deformable model-based methods for shape control of a haptic jamming surface," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, pp. 1–1, 02 2016.

[14] S. Bouaziz, M. Deuss, Y. Schwartzburg, T. Weise, and M. Pauly, "Shape-up: Shaping discrete geometry with projections," *Comp. Graph. Forum*, vol. 31, pp. 1657–1667, 08 2012.

[15] S. Bouaziz, S. Martin, T. Liu, L. Kavan, and M. Pauly, "Projective dynamics: Fusing constraint projections for fast simulation," *ACM Transactions on Graphics*, 01 2014.

[16] T.-H. Kwok and Y. Chen, "Geometry-driven finite element for four-dimensional printing," *Journal of Manufacturing Science and Engineering*, 07 2017.

[17] S. Jin, Y. Zhang, and C. C. L. Wang, "Deformation with enforced metrics on length, area and volume," *Computer Graphics Forum*, vol. 33, 05 2014.

[18] B. Prosenjit, O. Joseph, S. Chang, and W. Stefanie, "Isometric morphing of triangular meshes," *The Canadian Conference on Computational Geometry*, 08 2008.