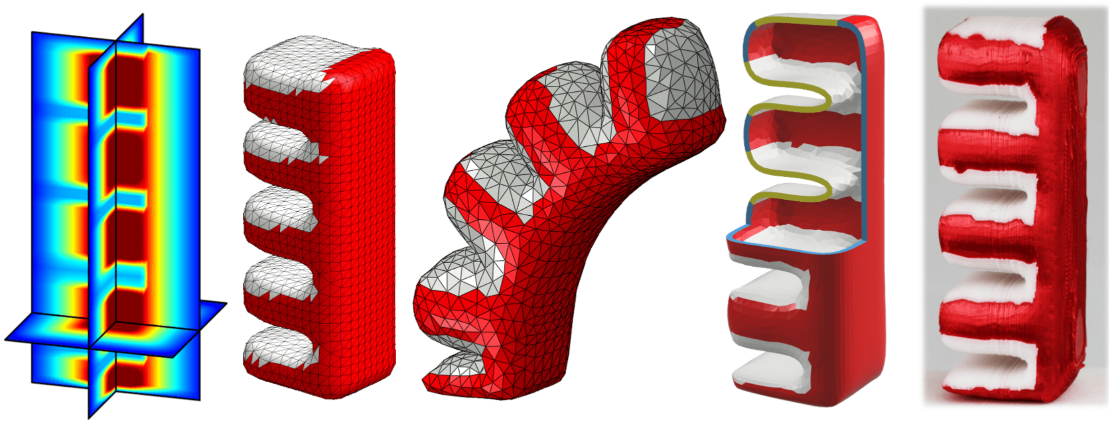
A Seamless Workflow for Design and Fabrication of Multimaterial Pneumatic Soft Actuators

Lawrence Smith[1], Travis Hainsworth[1], Zachary Jordan[1], Xavier Bell, Robert MacCurdy[1*]



Multimaterial soft actuators at progressive stages of development in the workflow presented here. From left: signed distance function geometry description; surface triangulation and material assignment; simulated deformation during actuation; preparation for additive manufacturing; 3D-printed actuator.

***Abstract*—Soft robotic actuators offer a range of attractive features relative to traditional rigid robots including inherently safer human-robot interaction and robustness to unexpected or extreme loading conditions. Soft robots are challenging to design and fabricate, and most actuators are designed by trial and error and fabricated using labor-intensive multi-step casting processes. We present an integrated collection of software tools that address several limitations in the existing design and fabrication workflow for pneumatic soft actuators. We use implicit geometry functions to specify geometry and material distribution, a GUI-based software tool for interactive exploration of computational network representations of these implicit functions, and an automated tool for generating rapid simulation results of candidate designs. We prioritize seamless connectivity between all stages of the design and fabrication process, and elimination of steps that require human intervention. The software tools presented here integrate with existing capabilities for multimaterial additive manufacturing, and are also forward-compatible with emerging automated design techniques. The workflow presented here is intended as a community resource, and aimed at lowering barriers for the discovery of novel soft actuators by experts and novice users. The data gathered from human-interaction with this tool will be used by future automation tools to enable fully-automated soft actuator design based on high-level specifications.**

***Index Terms*—Additive Manufacturing, Hydraulic/Pneumatic Actuators, Compliant Joints and Mechanisms**

I. Introduction

SOFT actuators grasp delicate objects, respond robustly to uncertain and dynamic loading scenarios, and resist damage more effectively than traditional rigid end effectors [1]. Soft robot design is challenging and inherently interdisciplinary, and most soft actuators are designed by teams of human engineers by trial-and-error and fabricated using labor-intensive casting processes [2]. The pace of development of novel actuator designs is limited by human intuition in the soft robotics research field [3], and fabrication constraints of casting approaches constrict the design space [4]. Previous efforts to standardize the design and fabrication of soft robots such as the *Soft Robotics Toolkit* [2] have made enormous impact on the field; in this work we offer a new design and fabrication paradigm to the soft robotics community. Our work endeavors to increase the pace of pneumatic soft actuator development by bringing automated design and fabrication tools to bear on the design space.

The application of semi- or fully-automated fabrication methods to soft robotic actuators, namely additive manufacturing, is widespread [5]. Fused filament fabrication [6], direct silicone extrusion [7][8], embedded 3D printing [9], inkjet printing [10][4][11], and stereolithography [12] have proven effective means of fabricating geometrically complex soft actuators. Advantages of additive manufacturing include the release of fabrication constraints inherent in traditional casting techniques [13], more rapid and consistent reproduction of existing actuator designs [4], and increased capacity for actuator customization.

Soft actuator literature emphasizes the benefits of soft actuators fabricated with multiple materials of varying stiffness. The notable *PneuNet* [14] design, for example, features a strain limiting layer made of a thin, inextensible material to preferentially induce bending about one axis. Recent work has shown the advantages of fabricating multimaterial soft actuators using fused filament fabrication [13] and silicone extrusion [8], although this technology is still emerging.

In this paper we present a collection of integrated software tools aimed at making pneumatic soft actuator design

We would like to thank Festo for the donation of pneumatic components that allowed us to collect the test data presented in this report

[1]Authors are with the Department of Mechanical Engineering, University of Colorado Boulder, Boulder, CO, USA

* Corresponding author: *maccurdy@colorado.edu*

and fabrication simpler, faster, and more accessible to non-experts. With further development effort, we believe that this software workflow can be incorporated into automated design algorithms, enabling fully automated design and fabrication of multimaterial pneumatic soft actuators. In Section 2, we detail components of our integrated workflow, highlighting advantages and limitations of our choices. In Section 3, we present a variety of pneumatic soft actuators designed and fabricated with our workflow, and compare simulated results to empirical measurements. In Section 4, we comment on conclusions of this work and opportunities for improvements.

II. Overview of Design Workflow

The existing design and fabrication workflow for soft robotics is characterized by manual processes and bottlenecks between design, characterization, and fabrication phases [4]. Interactive computer aided design (CAD) is typically used to specify geometry, which requires expert attention and is typically not robust to parametric revisions that produce significant morphological changes [1]. Nonlinear finite element analysis (FEA) is typically used to assess candidate designs [2], which often requires modifications to CAD geometry, manual assignment of boundary conditions, and manual mesh generation. Finally, multi-step casting processes are used to fabricate designs, which require manual mold design and significant effort to cast and de-mold soft robot designs.

In contrast, the workflow presented in this paper prioritizes seamless connectivity between stages of the design and fabrication process, and seeks to minimize steps that require manual intervention. Our method provides easy, interactive generation of implicit geometry representations, an online static geometry preview, one-click creation of the corresponding 3D printable design files, and automatic creation of complete FEA simulation files for dynamic visualization.

A. Implicit Geometry Modeling

The geometry and material distribution in many pneumatic soft actuators may be efficiently described by analytical functions in cylindrical coordinates, with the z axis parallel to the long dimension of the actuator. Because geometric features often repeat along the length of the actuator, and descriptions of geometry/material distribution in the transverse plane are natural in polar coordinates, representations of actuator geometry can be surprisingly compact. For example, the expression

$$(rsin(\theta)-\tfrac{1}{2})^4+2(rsin(\theta+\tfrac{\pi}{2}))^4+\Upsilon(\theta-\tfrac{1}{2})\Upsilon(z)=\psi \quad (1)$$

produces the scalar field shown in Figure 1, where the symbol $\Upsilon(\cdot)$ indicates the trigonometric operator $sin(tan(20sin(\cdot))^{-1})+1$. Triangulating an isosurface through this scalar field where $\psi=0$ produces a surface representation of a *PneuNet*-like actuator with distinctive pleating and neutral bending axis.

Describing geometry with implicit functions (or *signed distance functions* [15]) may appear unwieldy to a designer accustomed to building geometry using interactive CAD software. However implicit geometry modeling offers several

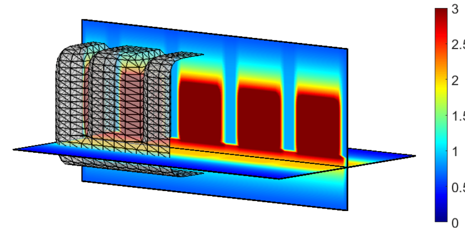


Fig. 1: Section planes through scalar field representation of a *PneuNet* [14]-like soft actuator designed in our workflow.

advantages over traditional tools, which represent geometry by mathematically describing the surfaces and edges that bound volumes [15]. Implicit geometry representations contain no edges or surfaces - instead, functions operate on a design domain, producing a continuous scalar field. Since no references to edges or faces exist, traditionally "fragile" CAD operations such as shelling, Boolean operations, and complex filleting are reliable and near-instantaneous. Additionally, the scalar field output of a geometry function contains valuable information beyond the position of the object's boundary - the sign of the field indicates whether a point is interior to the surface of the part, and the magnitude of the field gives a rough measure of the distance to the surface.

For extremely complex designs, which may contain tens of thousands of faces when represented in traditional CAD software, implicit geometry modeling shines. The memory footprint of implicit geometry designs does not scale with geometric complexity, and evaluations of geometry are embarrassingly parallel and well-suited for GPU implementation. Commercial implicit geometry modeling platforms demonstrate 60x speed increases over traditional CAD packages when evaluating complex geometries [16]. Finally, implicit geometry functions are naturally represented by computational networks, which are compact and offer forward-looking advantages for soft actuator design. Previous results have noted the benefits of network-based representations of implicit geometry for automated design [17], but our work leverages the particular elegance of descriptions of soft actuator geometry in cylindrical coordinates, and focuses directly on the design of pneumatic soft actuators.

B. Computational Network Explorer

Any implicit function may be expressed as a computational network, which readily exposes the function's structure - relationships between inputs and outputs, symmetries, repeating motifs, etc. - to an analyst. Computational networks are recognized as particularly compact and powerful generative encodings [18], and various graph and network structures have been summarized in [19]. Nodes in theses networks represent elementary functions, and links represent the flow of data between nodes. Figure 2 shows the computations inside each node of the networks we use in this work.

All input signals to a node are scaled by weights associated with the links connecting them to the node, then combined into a single value using an *aggregation function*, which in our implementation can be either summation or multiplication. The aggregated value is fed into an *activation function*,

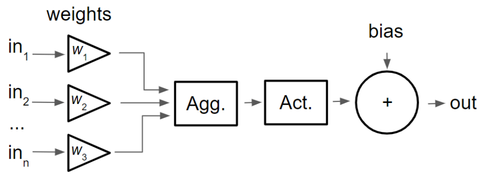


Fig. 2: Implementation of aggregation, weighting, activation, and bias in the functional nodes of these computational networks.

which transforms the input signal to an output signal using one of a handful of elementary functions like $sin(\cdot)$ or $(\cdot)^2$. Finally, a *bias* is applied to the signal, yielding the output value. By arranging nodes with this simple internal structure, very complex networks can be assembled, which transform (θ, r, z) coordinates into scalar field values.

Computational network representations of implicit geometry functions can make interactive design more natural for a human designer than the direct manipulation of implicit geometry functions. In this work we present a GUI-based software tool (Figure 3) for the interactive design of network representations of soft actuator geometry. Our tool is built in *Matlab* (©Mathworks 2021) and aimed to be intuitive and accessible to non-expert users. The Computational Network Explorer gives users control of basic editing functionality such as adding and deleting nodes and links, modifying weight and bias terms, and node insertion.

The Explorer also allows users to visualize resulting geometries in 3D, remesh and smooth isosurface triangulations, save and load networks, and write designs to STL files, which are naturally portable to existing 3D printers. Users may also build or load a secondary network to define material distribution if they choose to design a multimaterial soft actuator, and write separate STL files for each distinct material. We have used this tool to discover a variety of interesting soft actuator designs (see Figure 5), and hope that distribution of the Explorer to the soft robotics community spurs collaboration and the discovery of novel actuator designs. We have hosted the designs presented in this work, along with dozens of others, in a public repository so that users can access them and load them into the Computational Network Explorer to make their own alterations.

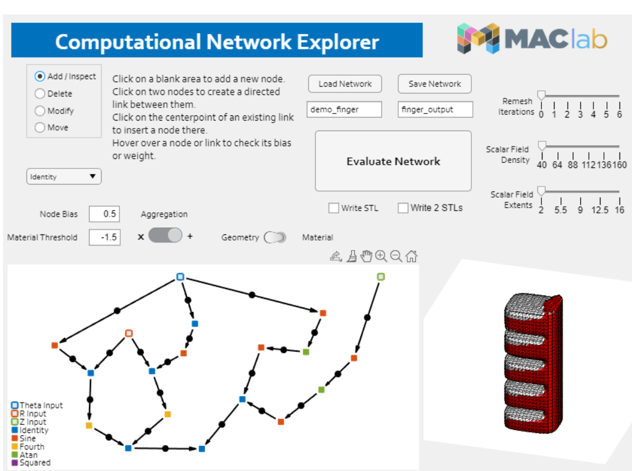


Fig. 3: Computational Network Explorer graphical user interface for exploring multimaterial soft actuator design space.

C. Simulation Pipeline

Numerical simulation accelerates design when it replaces or augments time-intensive design/fabricate/test/redesign iterations. However, simulation is underutilized in the design of soft robotic actuators - popular simulation methods require impractical amounts of computation time and setup [3]. We note several recurring properties in soft robots that make them challenging to simulate from a numerical standpoint:

1) *Finite Strain*: common fabrication materials for soft robots support large deformations, precluding linearized small strain theory
2) *Hyperelastic, Nearly Incompressible Materials*: commonly used materials are nearly incompressible and cause stability problems and nonphysical locking effects in simulation
3) *Actuated*: pneumatic loads which are strongly coupled to deformation variables are common, creating additional boundary condition nonlinearities
4) *Multi-Material Constitution*: wave speeds of constituent materials span orders of magnitude, leading to stiff dynamics that challenge integration in time
5) *Intermittent, Dense Contact*: contact interactions, expensive to resolve in FEA, are expected and exploited for performance
6) *Instability Phenomena*: structural instabilities are leveraged rather than avoided

The most widely used simulation tool for evaluating soft robot designs by far is nonlinear FEA [2], which is both general and powerful [20], and well-developed in commercial packages. The preparation of a candidate actuator design for FEA requires discretizing the solid geometry into a *computational mesh*, which typically consists of hexahedral or tetrahedral finite elements. Despite enormous effort, general algorithms for automated generation of high-quality hexahedral meshes on arbitrary geometry is an open problem [20]. Tetrahedtral meshes, which can be automatically generated, suffer stability issues and nonphysical locking effects when subjected to the large deformations typical of soft robots. Beyond the difficulties of FE mesh creation, simulation of soft robots is widely recognized as nontrivial [2] and computationally expensive. Thus the cost of iteratively generating actuator designs and simulating them using traditional approaches is both likely to require significant human intervention and long simulation times.

Shell based finite elements offer attractive solutions to both of these limitations. Shell finite elements are highly suited to represent geometries where one dimension of an object (the thickness dimension) is significantly smaller than two others. Solution variables are considered constant in the thickness direction and integrations are performed over an assumed thickness, resulting in a zero-thickness finite element with stiffness in tension, compression, shear, and bending.

Because shell finite elements have no volume, they can endure extreme deformations without loss of stability associated with element inversion and mesh entanglement, common in volumetric elements. The accuracy of simulation results

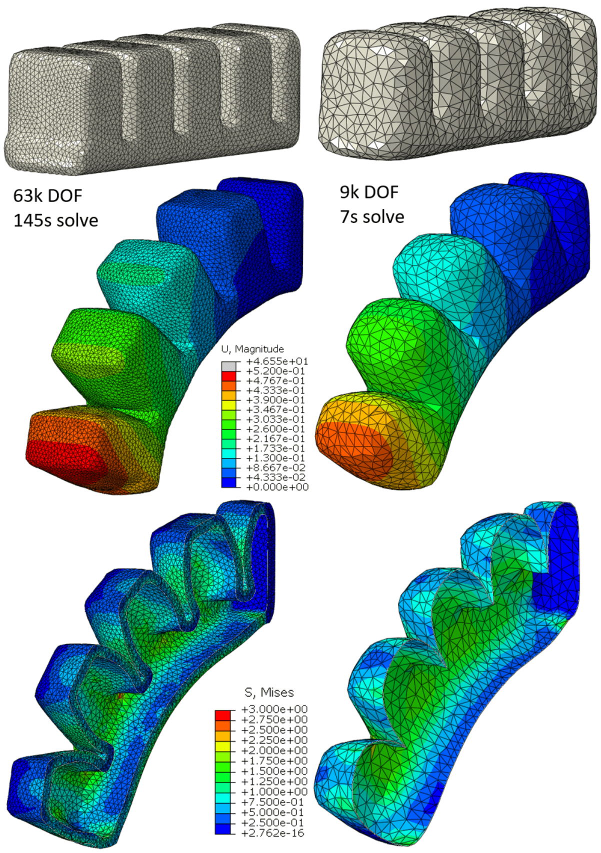


Fig. 4: FEA simulation of a soft actuator defined by scalar field of Figure 1, using tetrahedral volume elements (left) and triangular shell elements (right). From top, undeformed computational meshes; deformed geometry after application of 40kPa internal pressure, with contours showing displacement magnitude *mm*; deformed geometry in cross section with contours showing Von Mises stress *MPa*. Simulations performed in *Abaqus*.

using shell elements suffers in bending-dominated loading scenarios, and when the shell thickness to span ratio falls below 1:10. These assumptions are clearly violated in many soft actuator designs (such as in Figure 4). However, for the purposes of rapid evaluation of candidate soft actuator designs, we propose that the absolute accuracy of a simulation result should be considered *alongside* other factors such as labor cost of simulation setup, stability, and runtime. The application of shell elements to soft actuators is atypical but not without precedent: Zhang [10] uses shell finite element to accurately model the inflation of pneumatic soft actuators for wearable assitive devices.

Figure 4 shows the geometry of a typical *PneuNet*-like soft actuator designed in our workflow (the same geometry captured in the scalar field of Figure 1), simulated in two ways. At left we simulate the actuator using the traditional volumetric element workflow, consistent with many published results [2][4][8][6][14]: the thickness of the actuator walls is meshed using volumetric finite elements. Because the actuator walls are thin, small tetrahedral elements must be used in order to ensure adequate element quality. At right, we simulate the same actuator using triangular shell elements, with the shell thickness set equal to the wall thickness of the actuator. In regions of low curvature, shell element size can be relatively large while preserving element quality. Both simulations are executed using identical loading conditions, material properties, incrementation settings, and hardware: a fast but affordable desktop PC with a AMD Ryzen 3900x CPU (24 threads, 4.6GHz) and 64GB of RAM.

The volumetric mesh contains $87k$ tetrahedral elements, with a total of $63k$ degrees of freedom, while the shell mesh contains $3k$ triangular elements and a total of $9k$ degrees of freedom. The shell-based finite element model is significantly ($7x$) smaller than the volumetric-based model in terms of global DOF, and contains $30x$ fewer finite elements. These factors both contribute to the difference in solve time, but the difference in element count is particularly significant. For a problem with structural nonlinearity, the global tangent stiffness matrix must be assembled many times in the full Newton-Raphson solution scheme implemented here. For the soft actuator shown in Figure 4, using a shell mesh approach decreases solve time by a factor of 20.

Despite the difference in solve time, the simulation results are remarkably similar: the shell-based simulation predicts the deformed shape of the actuator well, and the peak displacement magnitude is within 10% of the result generated on the volumetric mesh. The stress field is resolved in more detail in the volumetric mesh, but the mean Von Mises stress over the bulk of the actuator is very similar.

As part of the workflow presented in this paper, we implement *fully automated* simulation execution; actuator designs discovered in the Explorer are simulated with zero human intervention required for simulation setup or mesh generation. Simulations are executed in the high-performance nonlinear FEA platform *Abaqus* (©Dassault Systemes), which is highly scriptable. *Abaqus* offers optimized handling of contact interactions, natural implementation of advanced nonlinear material models, CPU and GPU parallelization, and advanced nonlinear solution techniques.

Once a suitable actuator design is found in the Explorer, an isosurface triangulation through the implicit geometry function is automatically smoothed and remeshed [21], and element quality is monitored. Triangulation data is written to an input file as the location matrix for a finite element mesh, and additional data is automatically added to this input file (called an *Abaqus .inp* file), including boundary conditions, material models, solution settings, and output variables to monitor. The completed input file can then be passed to the solver directly through the command line, bypassing the GUI interface of *Abaqus* completely.

This functionality lowers barriers associated with building complex nonlinear FEA models, which are prohibitive to non-expert soft robotics enthusiasts. The ability to automatically generate simulation results completes the seamless connection between tools for design exploration and design evaluation, a key feature of the workflow presented here.

III. Results: Actuator Designs

Using the workflow described in the previous section, we have begun exploration of the multimaterial soft actuator design space, with exciting preliminary results. We find that

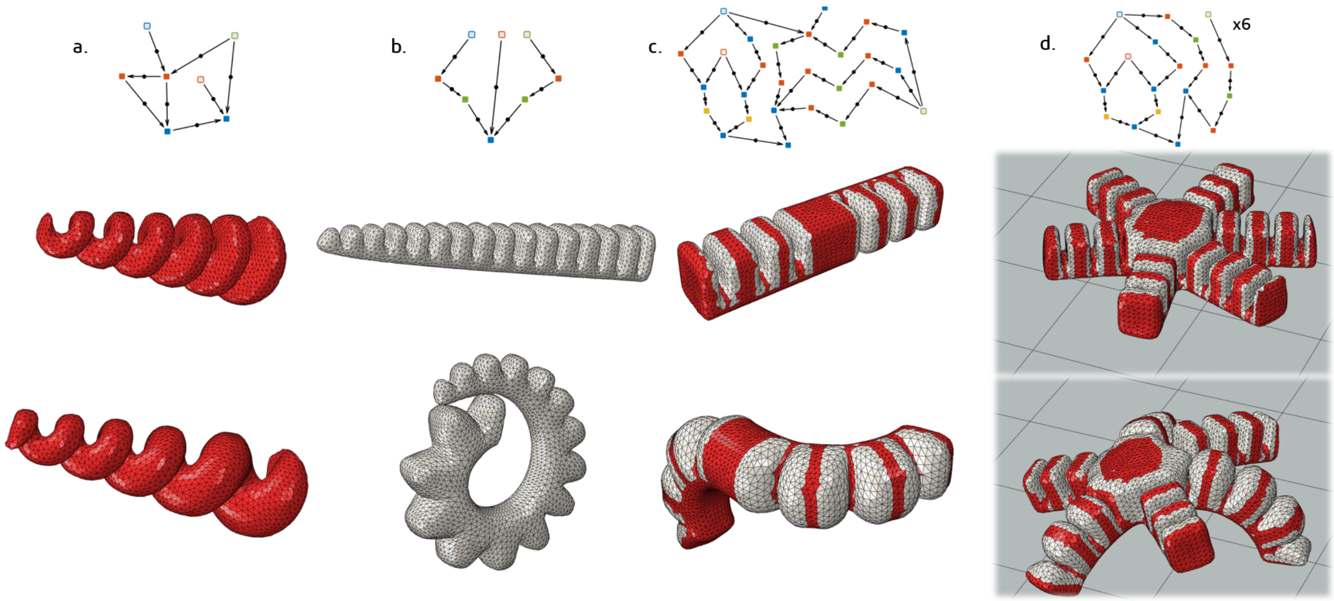


Fig. 5: Four morphologies designed using the Computational Network Explorer presented in this work. By row, from top to bottom, the computational network representation of the implicit geometry function; a triangulation of an isosurface taken through the resulting scalar field, with colors indicating stiff (red) and soft (white) material distribution; deformed geometry after application of 40-160kPa of internal pressure.

using the Computational Network Explorer makes designing with implicit geometry functions intuitive, and the ability to change network structure or weights and receive immediate feedback accelerates the design iteration process.

Figure 5 shows four such designs in the mathematically exact network representation (upper row), the undeformed isosurface triangulation (middle row), and the deformed computational mesh (lower row). Small changes to network structure may produce large changes to actuator morphology, and subnetworks which govern global geometric features (e.g. tapering in the z-direction) may be combined with other networks to modify them. We present a tapering helical spiral which extends and unwinds when pressurized, a tentacle-like actuator which curls onto itself, a jointed actuator which exhibits a complex deformation mode with two discrete bending directions, and a hexapod which lifts itself.

The hexapod in the rightmost column, reminiscent of the multigait soft walker presented in [22], highlights the particular advantages of designing with implicit geometry functions. To create the hexapod, we simply sum the scalar field representation of a single limb with itself five times, each time applying a translation and rotation to the coordinate system of the scalar field. No detailed design or "cleaning up" of the geometry was required - boolean operations yield a smooth scalar field, producing an airtight structure.

Empirical testing confirms that the shell-based finite element simulation approach adopted in this work is useful in predicting the behavior of fabricated actuators. We fabricated *PneuNet*-like actuators designed in our workflow by specifying a $2mm$ wall thickness, extending inward from the surface defined by Eq. 1. Actuators were fabricated from the thermoplastic polyurethane (TPU) material *NinjaFlex* (©NinjaTek), which is tough, compliant (Shore 85A hardness), and can be readily used by many commercial 3D printers.

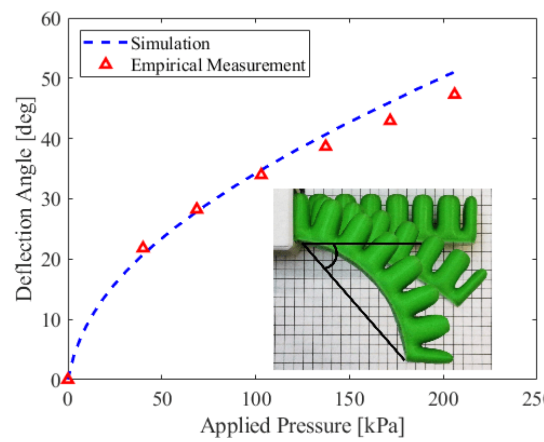


Fig. 6: Simulated and empirical results for a single actuator's displacement under pressure loading. Results agree to within 5%.

TABLE I: FEA Model Parameters

Mesh		Ogden Parameters			
Type	Linear Tri S3	μ_1	-30.921	α_1	0.508
Elements	3676	μ_2	10.342	α_2	1.375
		μ_3	26.791	α_3	-0.482

We subjected actuators to a quasistatic 200 kPa internal pressure load, both in simulation and experiment [13]. The simulation was conducted on the shell-based finite element mesh described in Table I, and a material model for NinjaFlex presented in [6].

We track displacement at the tip of the actuator and plot the angle of a line connecting each end of the actuator against the applied pneumatic load (see Figure 6). The simulated and empirical results agree to within 5% on average. By orienting a pair of these actuators in a 3D printed mount, we create a gripper capable of grasping delicate objects using modest pneumatic pressures (Figure 7).

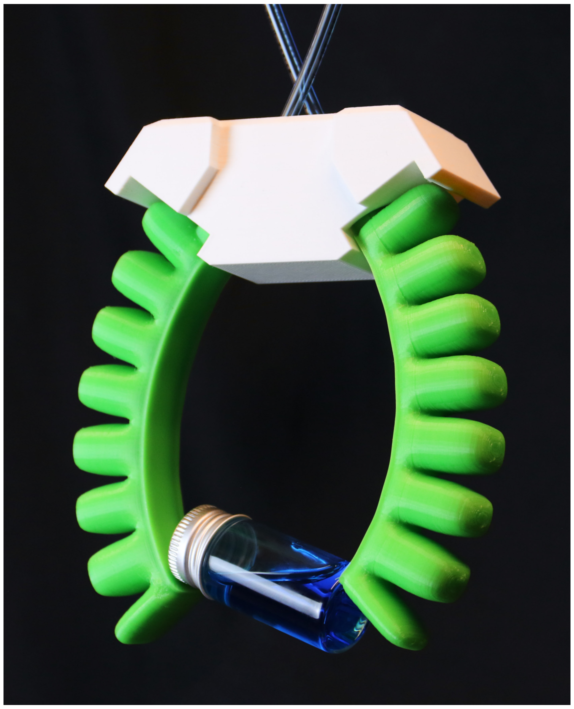


Fig. 7: Soft gripper capable of grasping delicate objects, designed, characterized, and fabricated using the workflow described in this paper.

IV. Conclusions and Further Research

We present a streamlined process for facile design, simulation, and fabrication of multimaterial soft robotic actuators. By compromising slightly on the accuracy of simulation results and the comforting habits of interactive CAD, our process makes gains in both speed and inter-connectivity between different stages of the design process.

We hope to see extensions made to the workflow presented here which improve the generality of the tool while maintaining the "seamless" nature of the workflow. Our shell-based finite element simulation approach could be augmented to include anisotropic material models appropriate for simulating fiber/elastomer composites common in soft actuators [10][23] (though not usually 3D printed). The implicit geometry implementation presented here creates manifold structures with constant wall thickness only; this could be modified in order to increase the geometric complexity on the interior of a design, or to improve printability. Print slicing and design checking is performed offline by human designers - our tool could be improved to automatically assess designs for printability and return feedback to the designer.

The tools presented here enable non-experts to rapidly design and improve soft robotic actuators, and integrate with existing additive manufacturing and emerging automated design tools. By creating a central repository where users can upload as well as download and "remix" others' soft robot designs, we hope to accelerate progress in the soft robotics field. We intend to analyze this rich dataset assembled by many human designers, and extract insights about implicit geometry modeling of high-performing soft actuators. The computation network representations of actuator geometry that we have chosen for our workflow are ideally suited for this analysis, because we can borrow heavily from extensive theoretical and practical work in graph theory. The insights gained from analysis of community-sourced soft robot designs will be critical to realizing workflows for fully automated design and fabrication of soft actuators.

The work presented here is publicly accessible at *https://github.com/MacCurdyLab/Soft-Actuator-Synthesis*.

References

[1] D. Rus and M. Tolley. "Design, fabrication and control of soft robots". *Nature* (2015).

[2] D. Holland et al. "The Soft Robotics Toolkit: Shared Resources for Research and Design". *Soft Robotics* (2014).

[3] M. Pozzi et al. "Efficient FEM-Based simulation of soft robots modeled as kinematic chains". *Proceedings - IEEE Intl. Conference on Robotics and Automation* (2018).

[4] C. Pasquier et al. "Design and Computational Modeling of a 3D Printed Pneumatic Toolkit for Soft Robotics". *Soft Robotics* (2019).

[5] A. Zolfagharian et al. "Evolution of 3D printed soft actuators". *Sensors and Actuators A: Physical* (2016).

[6] H. K. Yap et al. "High-Force Soft Printable Pneumatics for Soft Robotic Applications". *Soft Robotics* (2016).

[7] O. Yirmibesoglu et al. "Direct 3D printing of silicone elastomer soft robots and their performance comparison with molded counterparts". *2018 IEEE International Conference on Soft Robotics, RoboSoft* (2018).

[8] O. Byrne et al. "Additive manufacture of composite soft pneumatic actuators". *Soft Robotics* (2018).

[9] R. Truby et al. "Soft Somatosensitive Actuators via Embedded 3D Printing". *Advanced Materials* (2018).

[10] H. Zhang et al. "Design and development of a soft gripper with topology optimization". *IEEE International Conference on Intelligent Robots and Systems* (2017).

[11] R. MacCurdy et al. "Printable hydraulics: A method for fabricating robots by 3D co-printing solids and liquids". *IEEE Intl. Conference on Robotics and Automation* (2016).

[12] B.. Peele et al. "3D printing antagonistic systems of artificial muscle using projection stereolithography". *Bioinspiration and Biomimetics* (2015).

[13] T. Hainsworth et al. "A Fabrication Free, 3D Printed, Multi-Material, Self-Sensing Soft Actuator". *IEEE Robotics and Automation Letters* (2020).

[14] B. Mosadegh et al. "Pneumatic networks for soft robotics that actuate rapidly". *Advanced Functional Materials* (2014).

[15] Q. Li et al. "Towards additive manufacturing oriented geometric modeling using implicit functions". *Visual Computing for Industry, Biomedicine, and Art* (2018).

[16] G. Allen. "nTopology Modeling Technology" (2019).

[17] J. Hiller and H. Lipson. "Automatic design and manufacture of soft robots". *IEEE Transactions on Robotics* (2012).

[18] N. Cheney et al. "Unshackling evolution". *ACM SIGEVOlution* (2014).

[19] R. Kicinger. "Evolutionary computation and structural design : a survey of the state of the art". *Computers Structures* (2005).

[20] G. Joldes et al. "Suite of Meshless Algorithms for Accurate Computation of Soft Tissue Deformation for Surgical Simulation" (2019).

[21] Q. Fang and D. Boas. "Tetrahedral mesh generation from volumetric binary and grayscale images". *IEEE International Symposium on Biomedical Imaging* (2009).

[22] R. Shepherd et al. "Multigait soft robot". *Proceedings, Natl. Academy of Sciences, United States of America* (2011).

[23] F. Connolly et al. "Sew-free anisotropic textile composites for rapid design and manufacturing of soft wearable robots". *Extreme Mechanics Letters* (2019).