

Continuum Jacobian based Computational Morphogenesis for Soft Robotic Workspace Optimization

Zenan Song, Feifei Chen, *Senior Member, IEEE*, Guoying Gu, *Senior Member, IEEE*,
and Xiangyang Zhu, *Member, IEEE*

Abstract—The robot workspace, defining the complete set of reachable end-effector positions and orientations, is critical for functional capability. However, quantitative workspace design remains an open challenge in soft robotics. This is due to complicated implicit kinematics governed by nonlinear continuum mechanics, and prohibitive computational cost of evaluating deformations across the full actuation spectrum. This article presents a computational morphogenesis framework for automatic optimization of position and orientation workspaces in multi-chamber soft pneumatic actuators. Our approach is built on three key innovations: (i) a continuum Jacobian, defined as the derivative of the end-effector's degrees of freedom (DoFs) with respect to the actuation inputs, which transforms the workspace integral from the configuration space to the actuation space, making the volume computation analytically tractable, (ii) a second-order adjoint method to derive the analytical shape derivatives of both the displacement field and this Jacobian, providing the explicit gradient of the workspace volume with respect to the robot's morphology, and (iii) a differentiable, singularity-free geometric model, complemented by an adaptive surface reconstruction algorithm, to represent and evolve the robot's free-form shape. The framework is validated by designing multi-chamber pneumatic soft actuators, achieving an 8-fold increase in 3D positional workspace volume and a 1.6-fold increase in 2D orientational workspace volume compared to the baseline Pneu-Nets actuators.

Index Terms—Soft robotics, Workspace optimization, Computational Morphogenesis, Jacobian shape derivative.

I. INTRODUCTION

The workspace, encompassing all achievable end-effector positions and orientations, is a pivotal performance metric for robotic systems in both manipulation [1] and locomotion tasks [2]. While workspace synthesis and optimization have been addressed in the context of rigid-linked robots by leveraging well-established geometric kinematics [3], [4], soft robots demand fundamentally different analytical approaches. This distinction arises from their contrasting motion-generation principles: rigid robots rely on discrete joint articulations,

while soft robots produce continuous deformation under distributed actuation. This results in an implicit kinematic relationship governed by nonlinear continuum mechanics, where the robot's configuration is not an explicit function of its actuation inputs. Consequently, soft robotic workspaces exhibit an implicit and highly nonlinear dependence on actuation inputs and structural characteristics. To date, an end-to-end design methodology for maximizing the workspace of soft robots remains underdeveloped, hindering the full exploitation of soft materials' deformation potential.

For a generic soft pneumatic robot, this work focuses on the workspace of its end-effector, which is considered to have 6 degrees of freedom (DoFs), denoted by $(U_1, U_2, \dots, U_6)$. The workspace $\mathbb{W}$ is mapped from the input pressure space $\mathbb{P}$ through structural parameters $\mathbf{s}$. The analysis of workspace volume $|\mathbb{W}|$ necessitates defining the implicit relationship

$$|\mathbb{W}| = \int w(U_1, U_2, \dots, U_6) dU_1 dU_2 \dots dU_6 = F(\mathbf{s}) \quad (1)$$

where $w(\cdot)$ weighs each state's contribution. Equation (1) reveals that the workspace, derived from integration over a high-dimensional configuration space, is fundamentally governed by the structural parameters. However, this relationship is not only inherently implicit and computationally intensive, but also becomes increasingly computationally expensive when the robot features free-form structural characteristics, which is necessary for enabling a broad design space. The critical challenge, from a design perspective, is to establish an analytical relationship between the workspace volume $|\mathbb{W}|$ and free-form design parameters $\mathbf{s}$, as well as the sensitivity $\partial |\mathbb{W}| / \partial \mathbf{s}$, thereby facilitating efficient gradient-based optimization for workspace maximization of soft robots.

Initial progress has been made in addressing workspace design for soft robots. Existing studies predominantly focus on workspace analysis for pre-designed configurations, primarily employing numerical methods [5]–[7]. However, these approaches typically lack a differentiable mapping from the morphological design parameters to the workspace volume, which is required for design purposes. Conversely, emerging inverse design strategies for achieving prescribed deformations of soft robots mark early progress toward workspace optimization. Notable examples include shape and topology optimization methods developed to determine the optimal dielectric elastomer actuation fields [8], [9], pneumatic chamber morphology [10], [11], passive structural geometry [12]–

This work was supported in part by the National Key Research and Development Program of China under Grant 2023YFB4706500, in part by the National Natural Science Foundation of China under Grant 52522501, Grant T2293725, and Grant 52275026, and in part by the Shanghai Rising-Star Program under Grant 240A2704500. (*Corresponding author: Feifei Chen.*)

The authors are with State Key Laboratory of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China and Shanghai Key Laboratory of Intelligent Robotics, Meta Robotics Institute, Shanghai Jiao Tong University, Shanghai, China (email: sssnnn7119@sjtu.edu.cn; ffchen@sjtu.edu.cn; guguying@sjtu.edu.cn; mexyzhu@sjtu.edu.cn).

[14], and soft robotic surface design [15] for achieving target deformations under discrete actuation inputs. While these approaches successfully incorporate free-form structural features for localized deformation goals, they inherently fail to address the high-dimensional kinematic integration challenges central to workspace optimization, which requires performance evaluation across the entire input-output spectrum.

In this paper, we study the workspace volume maximization for soft pneumatic robots through automated morphological optimization, termed *computational morphogenesis*. We first introduce the soft robot's continuum Jacobian $\mathbf{J}$, defined as the derivative of end-effector DoFs with respect to actuation inputs ($\partial \mathbf{U} / \partial \mathbf{p}$), to transform workspace analysis from the solution-dependent configuration space to the predefined actuation space. Crucially, this formulation enables workspace-morphology sensitivity evaluation, which involves the shape derivatives of displacement fields and the Jacobian operator, as we will prove in Section IV. To derive the Jacobian shape derivative, we develop a second-order adjoint method that propagates gradients through nonlinear deformation kinematics. We then propose a differentiable, singularity-free closed-surface geometric model to dynamically track the air chamber's morphology. This framework supports both free-form design exploration and parametric implicit kinematic mapping. The parametric morphological representation and differentiable Jacobian formulation collectively establish a gradient-based, end-to-end morphogenesis pipeline for evolving free-form soft pneumatic robots. Additionally, an adaptive surface reconstruction algorithm is proposed to ensure high-quality and high-fidelity geometric representation throughout the optimization process.

This paper is organized as follows. Section II introduces related works. Section III presents the mathematical formulation of the workspace metric. In Section IV, we derive the Jacobian shape derivative using a second-order adjoint method. Section V provides a detailed description of the closed-surface geometric model for morphogenesis, and Section VI presents the optimization method. Section VII details the fabrication and experimental procedures. Section VIII demonstrates the application of our framework to multi-chamber pneumatic soft actuators for maximizing their position and orientation workspaces, with validation through both finite element analysis (FEA) and experiments. Finally, Section IX discusses the computational costs, potential extensions and current limitations of the work, and Section X concludes the work.

II. RELATED WORKS

Workspace Design. The workspace design paradigm differs fundamentally between rigid and soft robots. For rigid robots, where analytical kinematics enable computationally efficient analysis, design optimization typically focuses on geometric parameters (e.g., link lengths) to maximize reachable volume [16], [17] and minimize the Jacobian condition number to enhance dexterity [18]. In contrast, soft robots present unique challenges in workspace synthesis due to their continuum kinematics and nonlinear response across the full actuation spectrum. Current approaches for soft robot workspace characterization predominantly employ numerical methods. While

recent efforts like the optimization-based boundary detection method in [5] have advanced the field, significant limitations persist in design optimization. Existing strategies lack analytical relationships between morphology and workspace characteristics, relying instead on heuristic searches that are constrained to low-dimensional parameter spaces and simplified configurations [19]–[21]. *Our work bridges this gap by quantifying the analytical derivative of workspace volume with respect to free-form morphological parameters, enabling gradient-based computational morphogenesis for workspace optimization.*

Jacobian. In rigid robotics, the Jacobian matrix serves as a fundamental kinematic operator that relates actuator velocities to end-effector velocities, simultaneously encoding dual relationships between joint-space forces and task-space forces through its transpose properties [22], [23]. For soft robotic systems, however, the formulation of a comparable continuum Jacobian remains elusive due to its intricate dependence on structural morphology, distributed actuation, and deformation. Recently, Remy et al. formulated the Jacobian of fluid-actuated soft robots through volume expansion ratios to capture the force and motion relationships [24]. A similar definition has also been made in multi-DoF soft manipulators to develop inverse kinematics for control purposes [25]–[27]. Beyond kinematics and control, Jacobian-based analysis can also guide proprioceptive sensing in soft robots, facilitating strain-based shape and 3D force estimation via sensitivity ellipsoids [28]. *Compared with these Jacobian formulations, the present work places emphasis on the Jacobian's analytical expression and differentiability, based on which the shape derivative is established to enable the development of efficient, gradient-based computational morphogenesis.*

Geometric Model for Morphogenesis. The ability to automatically generate and optimize a robot's shape relies on a flexible and mathematically robust geometric model. For free-form shapes, B-spline surfaces are a popular choice because they use a set of control points to define a smooth, locally adjustable surface [10], [29], [30]. However, they struggle with creating seamless closed surfaces for describing the air chambers of a soft robot, as forcing the surface to close on itself often creates mathematical singularities (“the pole problem”), which can disrupt gradient-based optimization. To avoid this, other methods use a sphere as a base, mapping it to the desired closed shape. For example, spherical harmonics represent a surface as a linear combination of orthogonal basis functions defined over the entire sphere [31]. While powerful for representing complex global shapes, their basis functions exhibit global support, meaning a change in one coefficient affects the entire shape, making them unsuitable for local, iterative design changes. A more promising alternative, inspired by spherical radial basis functions [32]–[34], also uses a spherical base but represents the surface as a weighted sum of locally-supported kernel functions centered at various directions on the sphere. This allows for local control, as modifying one control point's weight only affects a nearby region of the surface. *Our work builds on this idea by developing a method that combines the local support property of B-splines with the singularity-free nature of spherical mappings. We*

TABLE I: Nomenclature for Workspace Formulation

Symbol	Description
U_i	The i -th end-effector DoF.
p_r	The r -th actuation input.
$J_{p_r}^i$	Continuum Jacobian component ($\partial U_i / \partial p_r$).
$n_1 A n_2 R n_3 T$	Notation for a workspace problem (n_1 : actuation, n_2 : rotation, n_3 : translation).
$ \mathbb{W}_X $	Volume or area of the workspace $\mathbb{W}_X$.
w_X	Weight function for workspace $\mathbb{W}_X$.
ζ	Curvilinear coordinates on the actuation space boundary.
$\mathbf{N}$	Outward unit normal on the workspace boundary.
η	The orientation angle.
$\hat{U}_i, i \in \{4, 5, 6\}$	The orientation axis.
$\bar{\mathbf{U}}, \bar{\mathbf{J}}$	Modified displacement and Jacobian for 3-DoF orientation workspace formulation.
$\mathbf{n} = (n_x, n_y, n_z)$	End-effector normal vector.

represent the robot's chambers with a set of control points that define a closed surface. To ensure a stable optimization, we also introduce an adaptive reconstruction algorithm that periodically redistributes these control points, maintaining a high-quality geometric representation throughout the morphogenesis process.

III. WORKSPACE FORMULATION

Soft robots are inherently underactuated as the degrees of freedom are theoretically infinite. In this work, we mainly investigate the workspace of the soft pneumatic robot's end-effector, with the translational DoFs $(U_1, U_2, U_3) \in \mathbb{R}^3$ and the orientation DoFs $(U_4, U_5, U_6) \in \{\mathbf{x} \in \mathbb{R}^3 \mid \|\mathbf{x}\| \leq \pi\}$ described by exponential coordinates. Collectively, these form the 6-DoF end-effector configuration vector $\mathbf{U} = (U_1, \dots, U_6)$. In the following notation, the workspace is expressed as $n_1 A n_2 R n_3 T$, where n_1 is the dimensionality of actuation space, also referred to as degrees of actuation (DoAs) [35], while n_2 and n_3 correspond to the dimensions of the orientation subspace and position subspace under investigation. All remaining unconstrained degrees of freedom are permitted to vary without restriction.

To facilitate the computation of the workspace volume, we employ Gauss's divergence theorem to transform the volume integral into a surface integral over the workspace boundary. Since the workspace is unknown, we map the workspace boundary integral to that over the boundary of the actuation space, by defining the continuum Jacobian. This procedure requires identifying the portion of the actuation space's boundary that maps to the true workspace boundary. For workspaces without self-intersections, the integral domain corresponds to the entire boundary of the actuation space. In cases involving self-intersections, the true workspace boundary can be determined through case-by-case analysis, and numerical methods have also been developed [21], [36]. To ensure clarity, the nomenclature used in this section is summarized in Table I.

A. Continuum Jacobian

To transform the workspace evaluation from the unknown configuration space to the predefined actuation space, we define the continuum Jacobian for soft robotic systems,

$$J_{p_r}^i = \frac{\partial U_i}{\partial p_r}, \quad (2)$$

where U_i represents the i -th concerned DoF of the robot, and p_r denotes the r -th actuation input. Furthermore, when evaluating integrals over the boundary of the actuation space, we use the Jacobian of the end-effector DoFs with respect to the boundary's curvilinear coordinates $\zeta = (\zeta_1, \dots, \zeta_{n-1})$. This is defined as:

$$J_{\zeta_k}^i = \frac{\partial U_i}{\partial \zeta_k} = \sum_r \frac{\partial U_i}{\partial p_r} \frac{\partial p_r}{\partial \zeta_k} = \sum_r J_{p_r}^i \frac{\partial p_r}{\partial \zeta_k}, \quad (3)$$

which is computed via the chain rule, linking the displacement to the parameterization of the actuation boundary. The actuation space is given by $\mathbb{P} \subset \mathbb{R}^{n_1}$. Three distinct operational regimes emerge from the dimensional relationship between DoA (n_1) and DoF ($n_2 + n_3$), as follows:

- DoA=DoF: The actuation space can be mapped to a continuous workspace, which is mainly investigated in this work.
- DoA>DoF: This case requires identifying the subset of the actuation space boundary [21] that, when mapped, constitutes the boundary of the workspace.
- DoA<DoF: The workspace constitutes a manifold, which can be quantified by its hypervolume, analogous to how a surface's area quantifies a 2D manifold embedded in 3D space. Such underactuated cases exceed the scope of our current framework.

B. Position Workspace

The position workspace is defined as the set of all points that the end-effector can reach when only its translational degrees of freedom are considered. The volume of the n -dimensional workspace ($n \in \{2, 3\}$) is expressed as

$$|\mathbb{W}_{nT}| = \int_{\mathbb{W}} d\nu = \int_{\mathbb{W}} dU_1 dU_2 \dots dU_n \quad (4)$$

where $\mathbb{W}$ denotes the domain that the workspace occupies. By noting the fact that $\nabla \cdot \mathbf{U} = n$ with $\mathbf{U} = (U_1, U_2, \dots, U_n)$ the configuration vector, and applying Gauss's divergence theorem, the workspace can be rewritten as

$$|\mathbb{W}_{nT}| = \frac{1}{n} \int_{\mathbb{W}} \nabla \cdot \mathbf{U} d\nu = \frac{1}{n} \int_{\partial \mathbb{W}} \mathbf{U} \cdot \mathbf{N} d\Gamma \quad (5)$$

where the workspace boundary Γ is parameterized by curvilinear coordinates, denoted as ζ , and $\mathbf{N}$ is the outward unit normal on the boundary.

For the planar 2-DoF position workspace with 2 independent actuation inputs (2A0R2T) shown in Fig. 1(a), its boundary is generally a curve, and the line element of the boundary can be expressed as

$$\mathbf{N} d\Gamma = (-J_{\zeta_1}^2, J_{\zeta_1}^1) d\zeta_1, \quad (6)$$

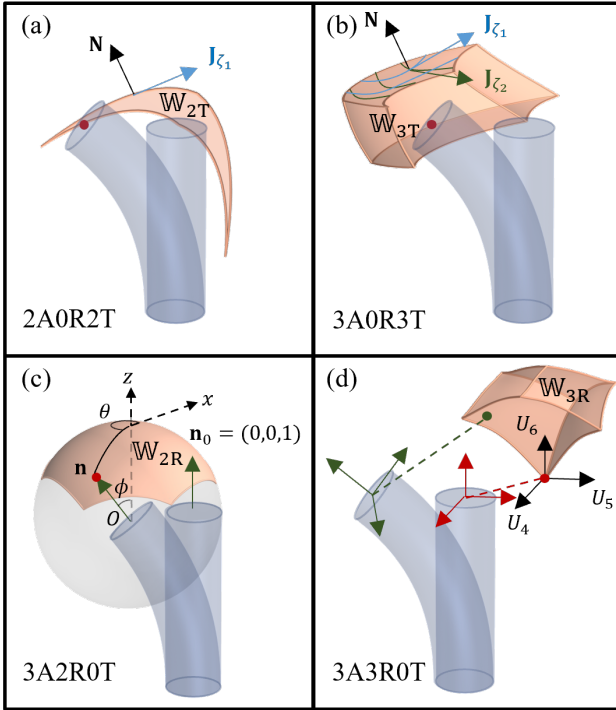


Fig. 1: Workspace concepts and their geometric representations. The orange regions ($\mathbb{W}$) illustrate different workspace types: (a) A 2-DoF planar position workspace (2A0R2T) with the boundary's tangential vector $\mathbf{J}_{\zeta_1}$ and normal vector $\mathbf{N}$. (b) A 3-DoF spatial position workspace (3A0R3T) with boundary tangential vectors $\mathbf{J}_{\zeta_1}, \mathbf{J}_{\zeta_2}$ and the normal vector $\mathbf{N}$. (c) A 2-DoF orientation workspace (3A2R0T), represented as the area swept by the end-effector's normal vector $\mathbf{n}$ on the unit sphere. (d) A 3-DoF orientation workspace (3A3R0T), visualized as a volume in exponential coordinates (U_4, U_5, U_6).

where ζ_1 represents the parameter that describes the boundary curve of the actuation space

$$p_r = p_r(\zeta_1), \quad r \in \{1, 2\}. \quad (7)$$

The 2-DoF position workspace area is then given by

$$|\mathbb{W}_{2T}| = \frac{1}{2} \int_{\partial\mathbb{P}} (U_2 J_{\zeta_1}^1 - U_1 J_{\zeta_1}^2) d\zeta_1 = \int_{\partial\mathbb{P}} w_{2T}(\mathbf{U}, \mathbf{J}_{\zeta_1}) d\zeta_1 \quad (8)$$

For the spatial 3-DoF position workspace with 3 independent actuation inputs (3A0R3T) shown in Fig. 1(b), its boundary is generally a curved surface, and the area element of the boundary is given by

$$\mathbf{N} d\Gamma = \mathbf{J}_{\zeta_1} \times \mathbf{J}_{\zeta_2} d\zeta_1 d\zeta_2, \quad (9)$$

where ζ_1 and ζ_2 are the parameters describing the boundary of the actuation surface

$$p_r = p_r(\zeta_1, \zeta_2), \quad r \in \{1, 2, 3\}, \quad (10)$$

and

$$\begin{aligned} \mathbf{J}_{\zeta_1} &= (J_{\zeta_1}^1, J_{\zeta_1}^2, J_{\zeta_1}^3), \\ \mathbf{J}_{\zeta_2} &= (J_{\zeta_2}^1, J_{\zeta_2}^2, J_{\zeta_2}^3). \end{aligned} \quad (11)$$

The volume of the 3-DoF position workspace is then expressed as

$$\begin{aligned} |\mathbb{W}_{3T}| &= \frac{1}{3} \int_{\partial\mathbb{P}} \mathbf{U} \cdot (\mathbf{J}_{\zeta_1} \times \mathbf{J}_{\zeta_2}) d\zeta_1 d\zeta_2 \\ &= \int_{\partial\mathbb{P}} w_{3T}(\mathbf{U}, \mathbf{J}_{\zeta_1}, \mathbf{J}_{\zeta_2}) d\zeta_1 d\zeta_2 \end{aligned} \quad (12)$$

C. Orientation Workspace

Unlike the position workspace, the orientation workspace is not within a Euclidean space. According to [37], it can be represented as a three-dimensional hypersphere embedded in the four-dimensional quaternion space. Thus, the orientation workspace can be quantified as the volume of the hypersphere, representing the 3 DoFs of orientation as depicted in Fig. 1(d). This concept can be reduced to represent in-plane orientation workspace volume (1 DoF) as the length of an arc on a circle. We also investigate the special case of 2-DoF orientation in 3D space, excluding rotation about its normal vector, with the workspace characterized by a portion of the unit sphere's surface, as shown in Fig. 1(c).

1) *3-DoF orientation workspace with 3 independent actuation inputs (3A3R0T)*: The orientation angle is defined as $\eta = \sqrt{U_4^2 + U_5^2 + U_6^2}$, and the volume of the orientation workspace can be formulated in exponential coordinates [37]

$$\begin{aligned} |\mathbb{W}_{3R}| &= \int_{\mathbb{W}} d\nu = \int_{\mathbb{W}} \frac{\sin^2(\eta/2)}{2\eta^2} dU_4 dU_5 dU_6 \\ &= \frac{1}{8} \int_{\mathbb{W}} \text{sinc}^2\left(\frac{1}{2}\eta\right) dU_4 dU_5 dU_6 \end{aligned} \quad (13)$$

with $\text{sinc}(x) = \sin x/x$ and $\eta \in [0, \pi]$. The weighting function $\text{sinc}^2(\eta/2)$ indicates that the contribution of each state is maximized at the origin ($\eta = 0$) and diminishes as the orientation angle η increases. Since the antiderivative of the sinc function lacks an elementary closed-form expression, the direct application of the divergence theorem becomes analytically intractable. To address this, we adopt a strategy of first approximating the integrand with its Taylor series polynomial, which then allows for the construction of a suitable vector field for the divergence theorem. We use the third-order Taylor approximation of the weight function,

$$\text{sinc}^2\left(\frac{1}{2}\eta\right) \approx 1 - \frac{1}{12}\eta^2, \quad \eta \in [0, \pi]. \quad (14)$$

Substituting this approximation into (13), the workspace volume becomes

$$|\mathbb{W}_{3R}| \approx \frac{1}{8} \int_{\mathbb{W}} \left[1 - \frac{1}{12} (U_4^2 + U_5^2 + U_6^2) \right] dU_4 dU_5 dU_6. \quad (15)$$

Then a vector field $\bar{\mathbf{U}} = (\bar{U}_1, \bar{U}_2, \bar{U}_3)$ such that its divergence satisfies $\nabla \cdot \bar{\mathbf{U}} = 3 \left[1 - \frac{1}{12} (U_4^2 + U_5^2 + U_6^2) \right]$ is constructed by integrating the divergence term with respect to each coordinate, e.g., $\bar{U}_1 = \int \left[1 - \frac{1}{12} (U_4^2 + U_5^2 + U_6^2) \right] dU_4$. It is noted that the choice for $\bar{\mathbf{U}}$ is not unique, but any valid choice will yield the same final workspace volume. A common and simple choice is

$$\bar{\mathbf{U}} = \begin{pmatrix} U_4 - \frac{1}{36}U_4^3 - \frac{1}{12}U_4U_5^2 - \frac{1}{12}U_4U_6^2 \\ U_5 - \frac{1}{36}U_5^3 - \frac{1}{12}U_5U_4^2 - \frac{1}{12}U_5U_6^2 \\ U_6 - \frac{1}{36}U_6^3 - \frac{1}{12}U_6U_4^2 - \frac{1}{12}U_6U_5^2 \end{pmatrix}. \quad (16)$$

With this definition, we can rewrite the volume integral as a surface integral over the workspace boundary $\partial\mathbb{W}$:

$$|\mathbb{W}_{3R}| \approx \frac{1}{8} \int_{\mathbb{W}} \frac{1}{3} (\nabla \cdot \bar{\mathbf{U}}) dU_4 dU_5 dU_6 = \frac{1}{24} \int_{\partial\mathbb{W}} \bar{\mathbf{U}} \cdot \bar{\mathbf{N}} d\Gamma. \quad (17)$$

The surface element $\bar{\mathbf{N}} d\Gamma$ can be expressed in terms of the actuation parameters (ζ_1, ζ_2) and the Jacobian as $\bar{\mathbf{J}}_{\zeta_1} \times \bar{\mathbf{J}}_{\zeta_2} d\zeta_1 d\zeta_2$. This leads to the final expression:

$$\begin{aligned} |\mathbb{W}_{3R}| &\approx \frac{1}{24} \int_{\partial\mathbb{P}} \bar{\mathbf{U}} \cdot \bar{\mathbf{J}}_{\zeta_1} \times \bar{\mathbf{J}}_{\zeta_2} d\zeta_1 d\zeta_2, \\ &= \int_{\partial\mathbb{P}} w_{3R}(\bar{\mathbf{U}}, \bar{\mathbf{J}}_{\zeta_1}, \bar{\mathbf{J}}_{\zeta_2}) d\zeta_1 d\zeta_2, \end{aligned} \quad (18)$$

where ζ_1 and ζ_2 are the actuation parameters defining the boundary of the orientation workspace, with

$$p_r = p_r(\zeta_1, \zeta_2), \quad r \in \{1, 2, 3\}, \quad (19)$$

and

$$\bar{\mathbf{J}}_{\zeta_1} = (J_{\zeta_1}^4, J_{\zeta_1}^5, J_{\zeta_1}^6), \bar{\mathbf{J}}_{\zeta_2} = (J_{\zeta_2}^4, J_{\zeta_2}^5, J_{\zeta_2}^6). \quad (20)$$

2) *2-DoF orientation workspace in 3D space with 3 independent actuation inputs (3A2R0T)*: This work considers that the three actuators are symmetrically distributed, as the workspace boundary can be intuitively identified. To characterize the orientation mobility while excluding the rotation along the robot's own axis, we subsequently introduce a 2-DoF orientation workspace representation. This reduced-order parameterization specifically captures the pointing capability of the end-effector while treating the axial rotation as unconstrained. Assuming the initial normal vector of the end-effector is $\mathbf{n}_0 = (0, 0, 1)$, the deformed normal vector of the end-effector $\mathbf{n} = (n_x, n_y, n_z)$ can be expressed as

$$\begin{aligned} n_x &= \hat{U}_5 \sin \eta + \hat{U}_4 \hat{U}_6 (1 - \cos \eta), \\ n_y &= -\hat{U}_4 \sin \eta + \hat{U}_5 \hat{U}_6 (1 - \cos \eta), \\ n_z &= \cos \eta + \hat{U}_6^2 (1 - \cos \eta), \end{aligned} \quad (21)$$

where $\hat{U}_i = U_i/\eta$ ($i = 4, 5, 6$) represent the normalized orientation axes. It is noted that point (n_x, n_y, n_z) lies on the unit sphere. For the 3A2R0T workspace, the surface area is described using spherical coordinates,

$$|\mathbb{W}_{2R}| = \int_{\mathbb{W}} d\nu = \int_0^{2\pi} \int_0^\phi \sin \psi d\psi d\theta = \int_0^{2\pi} (1 - \cos \phi) d\theta, \quad (22)$$

where

$$\theta = \arctan\left(\frac{n_y}{n_x}\right), \phi = \arccos(n_z), \quad (23)$$

and

$$d\theta = \frac{1}{n_x^2 + n_y^2} (n_x dn_y - n_y dn_x). \quad (24)$$

Thus, the volume of the 3A2R0T workspace, which is actually a portion of the unit sphere's surface, can be expressed as

$$|\mathbb{W}_{2R}| = \oint_{\partial\mathbb{W}} (1 - n_z) \frac{n_x dn_y - n_y dn_x}{n_x^2 + n_y^2}. \quad (25)$$

Since $\mathbf{n}$ is the function of (U_4, U_5, U_6) , (25) can be rewritten as

$$\begin{aligned} |\mathbb{W}_{2R}| &= \oint_{\partial\mathbb{P}} \frac{1}{1 + n_z} \sum_{j=4}^6 \left(n_x \frac{\partial n_y}{\partial U_j} - n_y \frac{\partial n_x}{\partial U_j} \right) J_{\zeta_1}^j d\zeta_1, \\ &= \oint_{\partial\mathbb{P}} w_{2R}(U_4, U_5, U_6, J_{\zeta_1}^4, J_{\zeta_1}^5, J_{\zeta_1}^6) d\zeta_1, \end{aligned} \quad (26)$$

with ζ_1 the actuation parameter along the boundary curve of the orientation workspace, and $n_x^2 + n_y^2 = 1 - n_z^2$ used to simplify the expression.

The preceding derivations show that the workspace volume $|\mathbb{W}_X|$, $X \in \{2T, 3T, 2R, 3R\}$, is an integral of a weight function w_X over the actuation boundary $\partial\mathbb{P}$. The variation of this volume with respect to a change in shape can therefore be expressed via the chain rule as

$$\delta|\mathbb{W}_X| = \int_{\partial\mathbb{P}} \left(\frac{\partial w_X}{\partial \mathbf{U}} \delta \mathbf{U} + \frac{\partial w_X}{\partial \mathbf{J}} \delta \mathbf{J} \right) d\zeta. \quad (27)$$

Gradient-based design optimization necessitates computing the sensitivities of the displacement field ($\delta \mathbf{U}$) and the Jacobian ($\delta \mathbf{J}$) with respect to the robot's shape, known as shape derivatives. Derivation details are provided in the following section.

IV. SENSITIVITY ANALYSIS

For the Shore 40A silicone rubber used in this study, the mechanical response is modeled within its hyperelastic regime. Under this assumption, the virtual work equation establishes an implicit relationship between the displacement field and the system inputs, from which both the Jacobian and its shape derivative can be derived. All subsequent formulas adopt the Einstein summation convention unless stated otherwise. We summarize the nomenclature used in this section in Table II.

A. Equilibrium Equation

We consider a pneumatic soft robot, where the static equilibrium is governed by the principle of virtual work. The total virtual work of the system consists of the internal work done by elastic forces and the external work done by pneumatic pressure terms. For any arbitrary kinematically admissible virtual displacement, it holds that

$$\mathcal{L}(\mathbf{u}, \mathbf{v}) = \mathcal{A}(\mathbf{u}, \mathbf{v}) - \sum_q \mathcal{B}^q(\mathbf{u}, \mathbf{v}) = 0, \quad (28)$$

where $\mathcal{L}$ is the residual of the virtual work. The term $\mathcal{A}$ represents the virtual work of the internal elastic stresses, and $\mathcal{B}^q$ is the virtual work of the external load from the q -th pneumatic actuator. The vector field $\mathbf{u}$ is the actual displacement of the robot, and $\mathbf{v}$ is an arbitrary virtual displacement field that respects the kinematic boundary conditions (e.g., it is zero at the fixed base).

In detail, the virtual work of the internal stresses is given by the integral of the stress tensor acting over the virtual strain:

$$\mathcal{A}(\mathbf{u}, \mathbf{v}) = \int_{\Omega_0} s_{ij} \frac{\partial v_j}{\partial x_i} d\Omega, \quad (29)$$

TABLE II: Nomenclature for Sensitivity Analysis

Symbol	Description
<i>Equilibrium Variables</i>	
$\mathbf{u}, \mathbf{v}$	Continuous actual and virtual displacement fields.
$\mathbf{U}, \mathbf{V}$	Discretized nodal actual and virtual displacement vectors.
$\mathcal{L}(\mathbf{u}, \mathbf{v})$	Residual of the principle of virtual work.
$\mathcal{A}(\mathbf{u}, \mathbf{v})$	Virtual work of internal stresses.
$\mathcal{B}^q(\mathbf{u}, \mathbf{v})$	Virtual work of pressure in the q -th chamber.
s_{ij}	First Piola-Kirchhoff stress tensor.
F_{ij}	Deformation gradient tensor.
$L_i(\mathbf{U})$	Discretized residual force vector.
K_{ij}	Tangential stiffness matrix ($\partial L_i / \partial U_j$).
<i>Shape Derivatives and Variations</i>	
Λ	Shape perturbation field on the boundary.
δU_l	Shape derivative of the l -th displacement DoF.
δJ_{pr}^l	Shape derivative of the Jacobian component.
Λ_i^U, Λ_i^J	Result of shape sensitivity coefficients for displacement and Jacobian analysis, respectively.
<i>Adjoint Variables</i>	
$\tilde{\mathbf{v}}_l^0$	Continuous adjoint field for the l -th degree of freedom sensitivity.
$\tilde{\mathbf{v}}_l^1, \tilde{\mathbf{v}}_{lr}^2$	Continuous first and second-order adjoint fields for the l -th degree of freedom and r -th pressure input Jacobian sensitivity.
$\tilde{\mathbf{v}}_i^0$	Discretized adjoint vector for displacement sensitivity.
$\tilde{\mathbf{v}}_i^1, \tilde{\mathbf{v}}_i^2$	Discretized first and second-order adjoint vectors for Jacobian sensitivity.
$\mathcal{Q}_i^l$	Selection vector with a one at the l -th entry.

where s_{ij} is the first Piola-Kirchhoff stress tensor, defined by the Neo-Hookean hyperelastic model as a function of the deformation gradient and the material parameters μ and κ (see Appendix A). It relates forces in the deformed configuration to areas in the undeformed configuration Ω_0 . The term $\partial v_j / \partial x_i$ is the gradient of the virtual displacement field.

For a multi-chamber pneumatic robot, the virtual work of the pressure p_q in the q -th chamber is given by:

$$\mathcal{B}^q(\mathbf{u}, \mathbf{v}) = -p_q \int_{\Gamma_0^q} |\mathbf{F}| F_{ij}^{-1} v_j m_i d\Gamma, \quad (30)$$

where the integral is taken over the undeformed chamber surface Γ_0^q . Here, Nanson's formula is used to map the pressure force, which acts on the deformed surface, back to an integral over the undeformed surface. F_{ij} is the deformation gradient, and m_i is the outward normal vector on the undeformed surface.

To solve this system, the structure is discretized using the Finite Element Method (FEM). The continuous displacement fields $\mathbf{u}$ and $\mathbf{v}$ are interpolated from nodal values using shape functions N^a :

$$u_i = N^a \hat{u}_i^a, \quad v_i = N^a \hat{v}_i^a, \quad (31)$$

where $\hat{u}_i^a$ is the displacement of the a -th node. By assembling the contributions from all elements and flattening the indices,

the virtual work equation (28) becomes a system of algebraic equations:

$$L_i(\mathbf{U}) V_i = 0, \quad (32)$$

where $\mathbf{U} = (U_1, U_2, \dots, U_n)$ is the global vector of nodal displacements, $\mathbf{V} = (V_1, V_2, \dots, V_n)$ is the global vector of virtual nodal displacements and $\mathbf{L}(\mathbf{U})$ is the residual force vector. Since this equation must hold for any arbitrary virtual displacement $\mathbf{V}$, it implies that the residual force vector must be zero: $L_i(\mathbf{U}) = 0$.

This nonlinear system is solved for $\mathbf{U}$ using the iterative Newton-Raphson method

$$U_i \leftarrow U_i - K_{ij}^{-1} L_j, \quad (33)$$

where the tangential stiffness matrix $K_{ij} = \partial L_i / \partial U_j$ is the Jacobian of the residual force vector.

B. Shape Derivative of Displacement Field

Our goal is to find how the displacement $\mathbf{U}$ changes in response to a small perturbation of the robot's shape Λ . This perturbation can be visualized as a small variation of each point on the robot's boundary surface along its normal vector. The resulting change in the displacement is denoted as $\delta \mathbf{U}$. We derive this sensitivity using the adjoint method. The starting point is the equilibrium equation (28), which must hold true for any shape. Therefore, its variation with respect to a shape perturbation must be zero

$$\delta \mathcal{L}(\mathbf{u}, \mathbf{v}) = \delta \mathcal{A}(\mathbf{u}, \mathbf{v}) - \sum_q \delta \mathcal{B}^q(\mathbf{u}, \mathbf{v}) = 0. \quad (34)$$

The variation of the virtual work terms, detailed in Appendix C, has two sources: (1) the change in the displacement field $\mathbf{u}$ itself, which leads to a change in the deformation gradient δF_{ij} , and (2) the direct change in the integration domain due to the shape perturbation Λ . After discretization, the variation can be written as

$$V_i K_{ik} \delta U_k + \int_{\Gamma_0} \sum_{i \in \{a, p_1, \dots, p_q\}} \Lambda_i^U(\mathbf{u}, \mathbf{v}) \Lambda d\Gamma = 0. \quad (35)$$

Here, the first term, $V_i K_{ik} \delta U_k$, captures the implicit dependence on shape through the change in displacement δU_k . The second term is an integral over the boundary, representing the explicit dependence on the shape perturbation Λ . The coefficients Λ_i^U are the sensitivities of the virtual work terms to a boundary movement, given by:

- For the structural stress,

$$\Lambda_a^U(\mathbf{u}, \mathbf{v}) = s_{ij} \frac{\partial v_j}{\partial x_i}, \quad (36)$$

- For the q -th actuator pressure,

$$\Lambda_{p_q}^U(\mathbf{u}, \mathbf{v}) = p_q \frac{\partial (|\mathbf{F}| F_{ij}^{-1} v_j)}{\partial x_i}. \quad (37)$$

Eq. (35) is an implicit equation for the displacement sensitivity δU_k . Solving it directly would require inverting the large stiffness matrix K_{ik} for each design parameter, which is computationally expensive. The adjoint method provides an elegant and efficient alternative. Since the virtual displacement

V_i is arbitrary, we can choose it strategically. To find the sensitivity of a specific degree of freedom, δU_l , we select a special virtual displacement field, called the adjoint field $\tilde{V}_i^0$, such that:

$$\tilde{V}_i^0 K_{ik} = -Q_k^l \quad \text{or} \quad \tilde{V}_i^0 = -Q_k^l K_{ki}^{-1}, \quad (38)$$

where Q^l is a selection vector that is one at the l -th degree of freedom and zero elsewhere. Substituting this adjoint field for V_i in (35) simplifies the first term to $Q_k^l \delta U_k = \delta U_l$. This isolates the desired sensitivity δU_l and allows us to express it directly as an integral involving the adjoint field:

$$\delta U_l = \int_{\Gamma_0} \sum_{i \in \{a, p_1, \dots, p_q\}} \Lambda_i^U(\mathbf{u}, \tilde{\mathbf{v}}_l^0) \Lambda \, d\Gamma. \quad (39)$$

Here, $\tilde{\mathbf{v}}_l^0$ is the continuous adjoint displacement field corresponding to the l -th degree of freedom.

C. Jacobian Analysis

The continuum Jacobian $J_{p_r}^j = \partial U_j / \partial p_r$ is the sensitivity of the displacement to a change in actuation pressure. To find it, we differentiate the virtual work equation (28) with respect to the r -th actuation pressure p_r :

$$\frac{\partial \mathcal{L}}{\partial p_r}(\mathbf{u}, \mathbf{v}) = \frac{\partial \mathcal{A}}{\partial p_r}(\mathbf{u}, \mathbf{v}) - \sum_q \frac{\partial \mathcal{B}^q}{\partial p_r}(\mathbf{u}, \mathbf{v}) = 0. \quad (40)$$

The detailed expressions (Appendix B) show that each term is not only directly affected by the actuation input p_r , but also indirectly influenced by the displacement field (via $\partial F_{kl} / \partial p_r$, which contains the Jacobian). After discretization, this equation becomes

$$\left(\frac{\partial L_i}{\partial p_r} + \frac{\partial L_i}{\partial U_j} \frac{\partial U_j}{\partial p_r} \right) V_i = \left(\frac{\partial L_i}{\partial p_r} + K_{ij} J_{p_r}^j \right) V_i = 0. \quad (41)$$

Since V_i is arbitrary, the term in the parenthesis must be zero, which gives the expression for the Jacobian:

$$J_{p_r}^j = -K_{ji}^{-1} \frac{\partial L_i}{\partial p_r}. \quad (42)$$

D. Shape Derivative of Jacobian

The remaining component required for the sensitivity analysis is the shape derivative of the Jacobian, $\delta J_{p_r}^l$. This is a second-order sensitivity, as it involves differentiation with respect to both shape and pressure. We construct an augmented virtual work equation by combining (28) and (40) with two independent virtual displacement fields, $\mathbf{v}^1$ and $\mathbf{v}^2$

$$\frac{\partial \mathcal{L}}{\partial p_r}(\mathbf{u}, \mathbf{v}^1) + \mathcal{L}(\mathbf{u}, \mathbf{v}^2) = 0. \quad (43)$$

Here, $\mathbf{v}^1$ and $\mathbf{v}^2$ are two independent arbitrary virtual displacement fields that satisfy the boundary conditions. Taking the variation of (43) yields

$$\delta \left[\frac{\partial \mathcal{L}}{\partial p_r}(\mathbf{u}, \mathbf{v}^1) \right] + \delta \mathcal{L}(\mathbf{u}, \mathbf{v}^2) = 0. \quad (44)$$

Detailed expressions are provided in Appendix C. After discretization and separating the terms, this equation takes the form

$$\begin{aligned} & V_i^1 K_{ij} \delta J_{p_r}^j \\ & + \left[V_i^1 \left(\frac{\partial K_{ik}}{\partial p_r} + \frac{\partial K_{ij}}{\partial U_k} J_{p_r}^j \right) + V_i^2 K_{ik} \right] \delta U_k \\ & + \int_{\Gamma_0} \sum_{i \in \{a, p_1, \dots, p_q\}} \Lambda_i^J(\mathbf{u}, \mathbf{v}^1, \mathbf{v}^2) \Lambda \, d\Gamma = 0. \end{aligned} \quad (45)$$

The coefficients Λ_i^J , which represent the explicit shape sensitivity of the augmented virtual work, are specified as follows:

- For the structural stress,

$$\Lambda_a^J(\mathbf{u}, \mathbf{v}^1, \mathbf{v}^2) = s_{ij} \frac{\partial v_j^2}{\partial x_i} + \frac{\partial s_{ij}}{\partial F_{kl}} \frac{\partial v_j^1}{\partial x_i} \frac{\partial F_{kl}}{\partial p_r}, \quad (46)$$

- For the q -th actuator pressure,

$$\begin{aligned} \Lambda_{p_q}^J(\mathbf{u}, \mathbf{v}^1, \mathbf{v}^2) &= p_q \frac{\partial (|\mathbf{F}| F_{ij}^{-1} v_j^2)}{\partial x_i} \\ &+ p_q \frac{\partial \left(\frac{\partial |\mathbf{F}| F_{ij}^{-1}}{\partial F_{kl}} v_j^1 \frac{\partial F_{kl}}{\partial p_r} \right)}{\partial x_i} \\ &+ \Delta(q, r) \frac{\partial (|\mathbf{F}| F_{ij}^{-1} v_j^1)}{\partial x_i}. \end{aligned} \quad (47)$$

Here, $\Delta(q, r)$ is the Kronecker delta.

Similar to the first-order case, we strategically choose the two arbitrary virtual fields V_i^1 and V_i^2 to isolate $\delta J_{p_r}^l$. First, we select the first adjoint field $\tilde{V}_i^1$ to isolate the desired component of the Jacobian variation in the first term

$$\tilde{V}_i^1 = -Q_j^l K_{ji}^{-1}. \quad (48)$$

This is the same type of adjoint problem as in the first-order case. Next, we select the second adjoint field $\tilde{V}_i^2$ to make the entire second term in (45) vanish

$$\tilde{V}_i^2 = -K_{ki}^{-1} \left[\tilde{V}_j^1 \left(\frac{\partial K_{jk}}{\partial p_r} + \frac{\partial K_{jl}}{\partial U_k} J_{p_r}^l \right) \right]. \quad (49)$$

This choice eliminates the term containing the first-order sensitivity δU_k . With these two adjoint fields, (45) is simplified to a direct expression for the Jacobian shape derivative

$$\delta J_{p_r}^l = \int_{\Gamma_0} \sum_{i \in \{a, p_1, \dots, p_q\}} \Lambda_i^J(\mathbf{u}, \tilde{\mathbf{v}}_l^1, \tilde{\mathbf{v}}_{lr}^2) \Lambda \, d\Gamma. \quad (50)$$

Here, $\tilde{\mathbf{v}}_l^1$ and $\tilde{\mathbf{v}}_{lr}^2$ are the continuous adjoint displacement fields corresponding to the l -th degree of freedom and r -th pressure input.

For computational efficiency, we leverage two key strategies. First, the adjoint equations (38) (48) and (49) are linear systems that share the same stiffness matrix K_{ij} as the forward problem. We only need to factorize this large matrix once. All required adjoint fields can then be found rapidly by solving systems of the form $K_{ij} x_j = b_i$ using inexpensive forward/backward substitution.

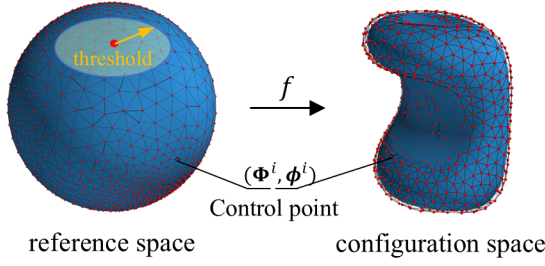


Fig. 2: Surface mapping from the reference space (unit sphere) to the configuration space. Red points indicate the control points, each with its associated influence region.

Second, we avoid explicitly forming the large and costly third-order tensor $\partial K_{ij}/\partial U_k$. Instead, we compute its product at the element level before global assembly

$$\tilde{\mathbf{V}}^1 \left(\frac{\partial \mathbf{K}}{\partial \mathbf{U}} \right) \mathbf{J} = \sum_e \left[\tilde{\mathbf{v}}_e^1 \left(\frac{\partial \mathbf{k}_e}{\partial \mathbf{u}_e} \right) \mathbf{j}_e \right], \quad (51)$$

where the product of small, local tensors $\tilde{\mathbf{v}}_e^1(\partial \mathbf{k}_e/\partial \mathbf{u}_e) \mathbf{j}_e$ is computed for each element and then assembled into a global matrix. This element-wise approach significantly reduces memory and computational costs. Subsequently, we describe the geometric representation for shape optimization, enabling the derivation of the workspace sensitivity with respect to geometric design parameters.

V. GEOMETRIC MODEL FOR MORPHOGENESIS

The morphological shape of pneumatic soft robots, especially that of air chambers where the distributed pressure loads are applied, is pivotal to their performance. To enable generative shape optimization (morphogenesis), we employ the B-spline surface to represent the robot's exterior surface. B-spline surfaces are parametric surfaces defined by a set of control points and basis functions, offering excellent local shape control. Tracking of the evolving internal air chambers requires a closed-surface geometric model with the following key characteristics:

- 1) At least thrice-differentiable to allow accurate evaluation of geometric properties such as curvature.
- 2) Free of singular points to ensure that the objective function remains continuously differentiable.
- 3) Local support property, meaning that modifying a single control point affects only a localized region of the surface, is crucial for efficient local shape modification and yields a sparse mapping function, thereby reducing computational complexity.

To meet these requirements, we introduce a spherical function based on control points to represent the closed surface, ensuring high differentiability while avoiding singularities. Additionally, since non-uniform control point distributions during optimization may hinder morphogenesis, we propose a surface reconstruction algorithm that enforces uniform re-distribution, thereby enhancing optimization robustness. Table III summarizes the nomenclature used in this section.

TABLE III: Nomenclature for Geometric Modeling and Reconstruction

Symbol	Description
$\mathbb{S}$	Reference space (unit sphere).
$f(\mathbf{S}; \hat{\Phi}, \hat{\phi})$	Mapping function from reference space (point $\mathbf{S}$) to configuration space with control points $\hat{\Phi}$ and $\hat{\phi}$.
$\mathbf{S}$	A point in the reference space $\mathbb{S}$.
$\mathbf{r}$	A point on the surface in the configuration space.
$\bar{S}_1, \bar{S}_2$	Stereographic projection coordinates.
$\hat{\Phi}, \Phi^i$	Set of control points and the i -th control point in the reference space.
$\hat{\phi}, \phi^i$	Set of control points and the i -th control point in the configuration space.
$c_i(\mathbf{S})$	Normalized weight of the i -th control point at $\mathbf{S}$.
$C_i(\mathbf{S})$	Initial weight of the i -th control point at $\mathbf{S}$.
d_0^i	Influence radius of the i -th control point.

A. Closed Surface Mapping

The closed surface is constructed by mapping points from the reference space (the unit sphere $\mathbb{S}$) to the configuration space (the surface $\mathbb{R}^3$) using a mapping function f

$$f: \mathbb{S} \rightarrow \mathbb{R}^3, \quad \mathbb{S} = \{\mathbf{S} \in \mathbb{R}^3 \mid \|\mathbf{S}\|_2 = 1\} \quad (52)$$

where $\|\cdot\|_2$ is the Euclidean norm. The mapping function f is determined by control points specified in both the reference space and the configuration space. That is, each control point consists of six coordinates, three in the reference space and three in the configuration space (see Fig. 2):

$$(\Phi^i, \phi^i), \quad \|\Phi^i\|_2 = 1, \quad i = 1, 2, \dots, n. \quad (53)$$

Once the control points are fixed, the surface is fully defined. Thus, we denote the surface representation as $\{\hat{\Phi}, \hat{\phi}\}$, with $\hat{\Phi} = \{\Phi^1, \Phi^2, \dots, \Phi^n\}$ representing the set of control points in the reference space, and $\hat{\phi} = \{\phi^1, \phi^2, \dots, \phi^n\}$ those in the configuration space. Here, we use the hat symbol ($\hat{\cdot}$) to denote a set of control points in the configuration (or reference) space.

The mapping function is then expressed as

$$\mathbf{r} = f(\mathbf{S}; \hat{\Phi}, \hat{\phi}) = \sum_{i=1}^n c_i(\mathbf{S}; \hat{\Phi}) \phi^i, \quad \|\mathbf{S}\|_2 = 1, \quad (54)$$

where $\mathbf{r}$ is a point on the closed surface corresponding to the coordinate $\mathbf{S} \in \mathbb{S}$, $c_i(\mathbf{S}; \hat{\Phi})$ is the weight of the i -th control point, satisfying the normalization condition,

$$\sum_{i=1}^n c_i(\mathbf{S}; \hat{\Phi}) = 1, \quad \forall \mathbf{S} \in \mathbb{S}. \quad (55)$$

The weight for each control point is determined based on the distance between the control point and a query point in the reference space. For any two points $\mathbf{a} = (a_1, a_2, a_3)$ and $\mathbf{b} = (b_1, b_2, b_3)$ in $\mathbb{S}$, the distance is computed as

$$d(\mathbf{a}, \mathbf{b}) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + (a_3 - b_3)^2}. \quad (56)$$

For a query point $\mathbf{S} \in \mathbb{S}$ and the i -th control point, the relative distance is defined by

$$\bar{d}^i(\mathbf{S}) = \frac{d(\mathbf{S}, \Phi^i)}{d_0^i}, \quad (57)$$

where d_0^i represents the influence radius of the i -th control point. For each control point Φ^i , the distances to all other control points are computed and sorted in ascending order. The influence radius, d_0^i , is then defined as the distance corresponding to the k -th nearest control point. The parameter k determines the smoothness and locality of the surface representation, analogous to the degree of a B-spline. Increasing k yields a smoother surface but reduces locality and raises computational cost. If k is increased without a corresponding increase in control point density, the larger influence radius leads to a loss of high-frequency surface details. This procedure ensures that, when the control points are uniformly distributed, each query point is primarily influenced by its k nearest control points (k is set to 20 in this work).

The initial weight of the i -th control point is defined by a polynomial weighting function that satisfies three essential properties: (i) it vanishes beyond the influence radius, (ii) it attains a value of one at zero distance, and (iii) it maintains high-order smoothness throughout the domain [38]. This initial weight function is expressed as

$$C_i(\mathbf{S}; \Phi^i) = \begin{cases} 20(\bar{d}^i)^7 - 70(\bar{d}^i)^6 + 84(\bar{d}^i)^5 - 35(\bar{d}^i)^4 + 1, & \text{if } \bar{d}^i < 1, \\ 0, & \text{if } \bar{d}^i \geq 1. \end{cases} \quad (58)$$

The normalized weight for the i -th control point is given by

$$c_i(\mathbf{S}; \hat{\Phi}) = \frac{C_i(\mathbf{S}; \Phi^i)}{\sum_{j=1}^n C_j(\mathbf{S}; \Phi^j)}. \quad (59)$$

The surface formulation is fully determined through the weight normalization in (59).

Since the mapping function is linear with respect to the control points, its derivatives with respect to the reference coordinates $\mathbf{S} = (S_1, S_2, S_3)$ can be computed straightforwardly. In particular, the first derivative with respect to the j -th ($j \in \{1, 2, 3\}$) reference coordinate S_j is

$$\frac{\partial \mathbf{r}}{\partial S_j} = \sum_{m=1}^n \frac{\partial c_m(\mathbf{S}; \hat{\Phi})}{\partial S_j} \phi^m, \quad (60)$$

and the second derivative is

$$\frac{\partial^2 \mathbf{r}}{\partial S_j \partial S_k} = \sum_{m=1}^n \frac{\partial^2 c_m(\mathbf{S}; \hat{\Phi})}{\partial S_j \partial S_k} \phi^m, \quad (61)$$

where $j, k \in \{1, 2, 3\}$. The derivatives of the normalized weights in (59) can be derived by standard differentiation techniques and are omitted here. The sparse nature of the mapping function significantly reduces the computational cost associated with these derivatives.

B. Surface Evolution

For the evolution of the surface, the control points in the configuration space are updated according to the gradient of the objective function. The derivatives of the surface with respect to the control points are expressed as

$$\frac{\partial r_i}{\partial \phi_l^m} = \delta_{il} c_m(\mathbf{S}; \hat{\Phi}), \quad (62)$$

$$\frac{\partial (\partial r_i / \partial S_j)}{\partial \phi_l^m} = \delta_{il} \frac{\partial c_m(\mathbf{S}; \hat{\Phi})}{\partial S_j}, \quad (63)$$

and

$$\frac{\partial [\partial^2 r_i / (\partial S_j \partial S_k)]}{\partial \phi_l^m} = \delta_{il} \frac{\partial^2 c_m(\mathbf{S}; \hat{\Phi})}{\partial S_j \partial S_k}, \quad (64)$$

where $i, j, k, l \in \{1, 2, 3\}$.

Stereographic projection is introduced as a curvilinear coordinate system to compute the surface's normal vector and curvature. In this technique, the unit sphere is mapped onto a plane, establishing a one-to-one correspondence between a point $A(S_1, S_2, S_3)$ on the sphere (excluding the North Pole, $NP(0, 0, 1)$) and a point $B(\bar{S}_1, \bar{S}_2)$ on the plane. This mapping is achieved by drawing a ray from the North Pole through A and assigning B as the intersection of this ray with the plane $S_3 = -1$. Notably, the North Pole is mapped to a point at infinity. The projection is defined by

$$\bar{S}_1 = \frac{2S_1}{1 - S_3}, \bar{S}_2 = \frac{2S_2}{1 - S_3}, \quad (65)$$

and the inverse projection is given by

$$S_1 = \frac{4\bar{S}_1}{\bar{S}_1^2 + \bar{S}_2^2 + 4}, S_2 = \frac{4\bar{S}_2}{\bar{S}_1^2 + \bar{S}_2^2 + 4}, S_3 = \frac{\bar{S}_1^2 + \bar{S}_2^2 - 4}{\bar{S}_1^2 + \bar{S}_2^2 + 4}. \quad (66)$$

Because the stereographic projection is highly differentiable, using $(\bar{S}_1, \bar{S}_2)$ as the curvilinear coordinate system enables straightforward computation of the tangential vectors. Consequently, the normal vector of the surface, $\mathbf{m}$, is given by

$$\mathbf{m} = \frac{\partial \mathbf{r}}{\partial \bar{S}_1} \times \frac{\partial \mathbf{r}}{\partial \bar{S}_2}. \quad (67)$$

Similarly, the Gaussian curvature K and mean curvature H can also be computed from $\frac{\partial \mathbf{r}}{\partial \bar{S}_1}$ and $\frac{\partial \mathbf{r}}{\partial \bar{S}_2}$, along with their derivatives. Detailed derivations can be found in [10].

The variation of the geometric boundary Λ , as expressed in (39) and (50), can be reformulated in terms of the variations of the control points in the configuration space, given by

$$\Lambda \, d\Gamma = \sum_{i,j,l} \frac{\partial r_i}{\partial \phi_l^j} \delta \phi_l^j m_i \, d\bar{S}_1 \, d\bar{S}_2 = \sum_{j,l} c_j \delta \phi_l^j m_l \, d\bar{S}_1 \, d\bar{S}_2. \quad (68)$$

By substituting this relationship into the shape derivative expressions for displacement (39) and Jacobian (50), we obtain their explicit derivatives with respect to the control points ϕ_l^m . The derivative of the displacement is:

$$\frac{\partial U_k}{\partial \phi_l^m} = \int_{\Gamma_0} \sum_{i \in \{a, p_1, \dots, p_q\}} \Lambda_i^U(\mathbf{u}, \tilde{\mathbf{v}}_k^0) c_m m_l \, d\Gamma, \quad (69)$$

and for the Jacobian, it is:

$$\frac{\partial J_{pr}^k}{\partial \phi_l^m} = \int_{\Gamma_0} \sum_{i \in \{a, p_1, \dots, p_q\}} \Lambda_i^J(\mathbf{u}, \tilde{\mathbf{v}}_k^1, \tilde{\mathbf{v}}_{kr}^2) c_m m_l \, d\Gamma. \quad (70)$$

Here, $\tilde{\mathbf{v}}_k^0$, $\tilde{\mathbf{v}}_k^1$, and $\tilde{\mathbf{v}}_{kr}^2$ are the respective adjoint fields. With the shape derivatives of the displacement and Jacobian established, the total derivative of the workspace volume with respect to the control points can be computed by applying the chain rule to the workspace integral formulations from (27).

The sensitivity of the workspace volume $|\mathbb{W}_X|$ with respect to a control point ϕ_l^m is given by:

$$\frac{\partial |\mathbb{W}_X|}{\partial \phi_l^m} = \int_{\Gamma_0} \int_{\partial \mathbb{P}} \sum_{i,r,k} \left[\frac{\partial w_X}{\partial U_k} \Lambda_i^U(\mathbf{u}, \tilde{\mathbf{v}}_k^0) + \frac{\partial w_X}{\partial J_{pr}^k} \Lambda_i^J(\mathbf{u}, \tilde{\mathbf{v}}_k^1, \tilde{\mathbf{v}}_{kr}^2) \right] c_m m_l d\zeta d\Gamma. \quad (71)$$

C. Surface Reconstruction

To ensure a robust and stable optimization process, we introduce a surface reconstruction algorithm that is periodically invoked to maintain a high-quality geometric representation. As the robot's shape evolves during morphogenesis, the control points defining the surface may become either excessively sparse or dense in distribution within certain regions [see Fig. 3(a)]. This non-uniformity degrades the accuracy of the shape representation and can lead to numerical instability (ill-conditioning), causing the optimization to stall.

The high-order differentiability of the surface allows us to employ efficient second-order optimization algorithms during reconstruction. This procedure, detailed in Algorithm 1 and illustrated in Fig. 3, is essential for enabling large-scale shape evolution. The strategy leverages established methods for mesh parameterization [39], [40], where each mesh edge is treated as a spring, causing each node to converge to the barycenter of its neighbors, to ensure a smooth and uniform mesh. The initial surface is $\{\hat{\Phi}, \hat{\phi}\}$ [Fig. 3(a)], and the goal is to find a new surface $\{\hat{\Psi}, \hat{\psi}\}$ [Fig. 3(e)] that closely matches the shape of the original surface but with uniformly distributed control points. The surface reconstruction scheme includes the following operations.

Treating each mesh edge as an elastic spring, the total energy of the surface mesh is defined as

$$E(\hat{\mathbf{r}}, \mathbf{e}) = \frac{1}{2} \sum_{m=1}^n \left\| \mathbf{r}^{e_1^m} - \mathbf{r}^{e_2^m} \right\|^2, \quad (72)$$

where $\mathbf{r}^{e_1^m}$ and $\mathbf{r}^{e_2^m}$ denote the endpoints of the m -th edge, and $\hat{\mathbf{r}}$ is the set of all point positions $\{\mathbf{r}^1, \mathbf{r}^2, \dots, \mathbf{r}^n\}$.

Uniform seeding. The first step is to uniformly seed n' control points in the configuration space [see Fig. 3(b)]. This procedure comprises two parts:

- 1) *Initial Sampling:* obtain initial samples $\hat{\mathbf{r}}_{\text{init}} = \{\mathbf{r}_{\text{init}}^1, \mathbf{r}_{\text{init}}^2, \dots, \mathbf{r}_{\text{init}}^{n'}\}$ on the surface, along with an initial connectivity $\mathbf{e} = \{e^1, e^2, \dots, e^m\}$. These samples are typically non-uniform.
- 2) *Position Optimization:* optimize the control point positions for a uniform distribution by solving

$$\begin{aligned} & \min_{\mathbf{r}^i, i=1,2,\dots,n'} E(\hat{\mathbf{r}}, \mathbf{e}) \\ & \text{subject to } \mathbf{r}^i \in \{\hat{\Phi}, \hat{\phi}\}, \quad i = 1, 2, \dots, n', \\ & \quad \frac{V - V_0}{V_0} < \epsilon, \end{aligned} \quad (73)$$

with V the volume enclosed by the surface, V_0 the initial volume, and ϵ a small tolerance (set to 0.01). Additionally, the connectivity $\mathbf{e}$ is updated using an edge flip

Algorithm 1: Surface Reconstruction Algorithm

Input: Initial surface $\{\hat{\Phi}, \hat{\phi}\}$ with non-uniform control points [Fig. 3(a)].

Output: Reconstructed surface $\{\hat{\Psi}, \hat{\psi}\}$ with uniform control points [Fig. 3(e)].

Step 1: Uniform Seeding [Fig. 3(b)]:

- 1 *Input:* $\{\hat{\Phi}, \hat{\phi}\}$, initial surface.;
- 2 *Output:* $\hat{\mathbf{r}}_{\text{opt}}$, a set of uniformly distributed points on the surface.;
- 3 Sample n' points on the surface to get initial positions $\hat{\mathbf{r}}_{\text{init}}$ and connectivity $\mathbf{e}$.;
- 4 Solve optimization problem (73) by iteratively updating point positions $\hat{\mathbf{r}}$ and connectivity $\mathbf{e}$.;

Step 2: Harmonic Refinement [Fig. 3(c)]:

- 5 *Input:* $\hat{\mathbf{r}}_{\text{opt}}$, uniformly distributed points.;
- 6 *Output:* $\hat{\Psi}$, new uniformly distributed reference control points.;
- 7 Obtain initial reference points $\mathbf{R}^i$ via inverse mapping $f^{-1}(\mathbf{r}_{\text{opt}}^i)$ (74).;
- 8 Project $\mathbf{R}^i$ onto planar disks using stereographic projection (65).;
- 9 Optimize point distribution on disks to achieve a harmonic mesh by minimizing energy (72).;
- 10 Project refined points back to the sphere (66).;

Step 3: Matching Error Minimization [Fig. 3(d)]:

- 11 *Input:* $\hat{\mathbf{r}}_{\text{opt}}$ and $\hat{\Psi}$, target points and new reference points.;
- 12 *Output:* $\hat{\psi}$, final configuration-space control points.;
- 13 Solve the linear least-squares problem (77) to find new configuration-space control points $\hat{\psi}$ that best match the original surface shape.;

algorithm to further refine the mesh. This optimization problem gives the results $\hat{\mathbf{r}}_{\text{opt}} = \{\mathbf{r}_{\text{opt}}^1, \mathbf{r}_{\text{opt}}^2, \dots, \mathbf{r}_{\text{opt}}^{n'}\}$ that are uniformly distributed in the configuration space. This procedure is illustrated in the Supplementary Video.

Harmonic refinement. Once the control points $\hat{\mathbf{r}}_{\text{opt}}$ have been uniformly distributed in the configuration space, the next step is to relocate their corresponding positions in the reference space as uniformly as possible [see Fig. 3(c)]. This is accomplished by applying a harmonic refinement method to the control points, which is formulated as a linear least squares problem and efficiently solved using the Newton method.

The initial distribution of control points in the reference unit sphere is obtained via the inverse mapping

$$\mathbf{R}^i = f^{-1}(\mathbf{r}_{\text{opt}}^i), \quad i = 1, 2, \dots, n', \quad (74)$$

which is typically non-uniform and requires further optimization. To this end, stereographic projection (65) is used to map the control points from the unit sphere onto a plane. The sphere is divided into two halves so that the upper hemisphere is projected from the South Pole, yielding the set $\hat{\mathbf{R}}_{\text{upper}} = \{\mathbf{R}_{\text{upper}}^1, \mathbf{R}_{\text{upper}}^2, \dots, \mathbf{R}_{\text{upper}}^{n'}\}$, while the lower

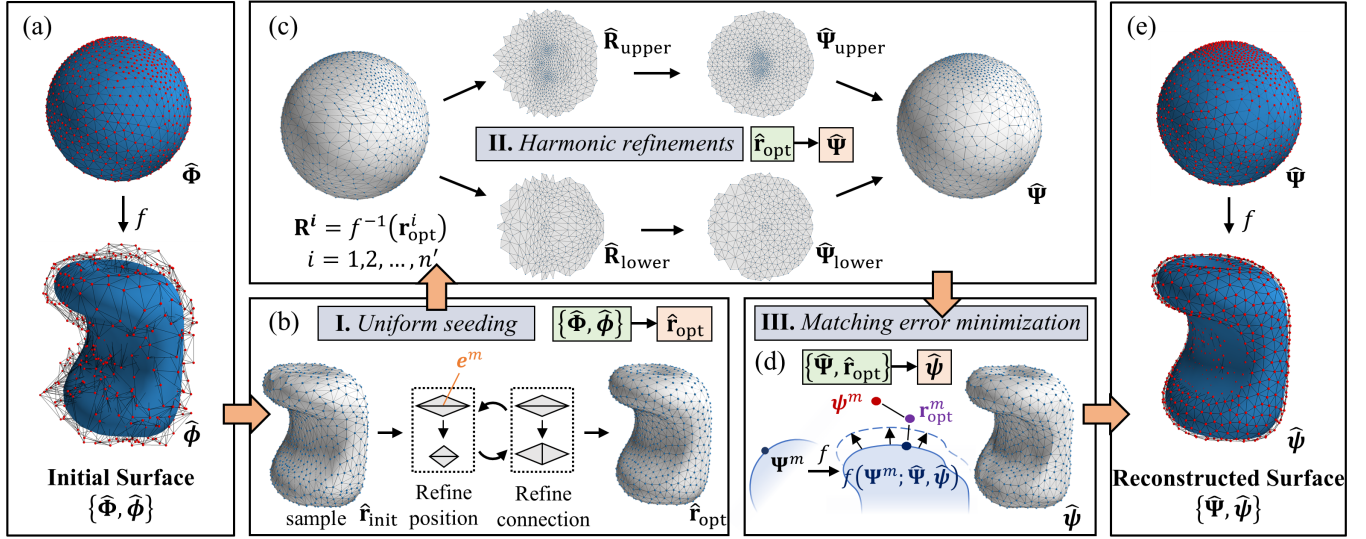


Fig. 3: The surface reconstruction workflow, which remeshes the control points to ensure geometric quality during optimization. Variables highlighted in green denote the inputs in each step, while those highlighted in orange indicate the outputs. (a) The initial surface $\{\hat{\Phi}, \hat{\phi}\}$ with a non-uniform control point distribution. (b) Uniformly seeding new control points in the configuration space to obtain the points $\hat{\mathbf{r}}_{\text{opt}}$. (c) Projecting these points back to the reference space and performing harmonic refinements to obtain a uniform distribution $\hat{\Psi}$. (d) Minimizing the matching error between the original points $\mathbf{r}_{\text{opt}}^m$ and new surfaces $f(\Psi^m; \hat{\Psi}, \hat{\psi})$ by adjusting the control points ψ^m in the configuration space, obtaining the final control points in the configuration space $\hat{\psi}$. (e) The final reconstructed surface $\{\hat{\Psi}, \hat{\psi}\}$ with uniformly distributed control points.

hemisphere is projected from the North Pole, yielding the set $\hat{\mathbf{R}}_{\text{lower}} = \{\mathbf{R}_{\text{lower}}^1, \mathbf{R}_{\text{lower}}^2, \dots, \mathbf{R}_{\text{lower}}^{n'}\}$.

This conformal mapping ensures that, even after optimization on the plane, the uniformity is preserved when the points are reprojected to the sphere. To guarantee a seamless surface, we employ a sequential optimization strategy on the planar projections. First, we optimize the point distribution for the upper hemisphere, which includes an overlapping equatorial band of points from the lower hemisphere. The resulting positions for the points in this overlapping band are then fixed. Subsequently, we optimize the point distribution for the lower hemisphere, using the fixed equatorial points as a boundary condition. This two-step process ensures a smooth stitch between the two hemispheres when they are reprojected back to the sphere. The optimization over the plane is formulated separately for the upper and lower halves as

$$\min_{\mathbf{R}_{\text{upper}}^i, i=1,2,\dots,n'} E(\hat{\mathbf{R}}_{\text{upper}}, \mathbf{e}), \quad (75)$$

$$\min_{\mathbf{R}_{\text{lower}}^i, i=1,2,\dots,n'} E(\hat{\mathbf{R}}_{\text{lower}}, \mathbf{e}). \quad (76)$$

These linear least squares problems are solved using the Newton method, yielding the optimized reference coordinates of control points on the plane, denoted by $\hat{\Psi}_{\text{upper}}$ and $\hat{\Psi}_{\text{lower}}$. Reprojection onto the unit sphere via the inverse stereographic projection (66) then provides uniformly distributed reference control points $\hat{\Psi} = \{\Psi^1, \Psi^2, \dots, \Psi^{n'}\}$.

Matching error minimization. At this stage, we have a new set of control points $\{\hat{\Psi}, \hat{\mathbf{r}}_{\text{opt}}\}$ defining a preliminary reconstructed surface. Since the new surface may not perfectly

match the original surface, a final adjustment is made by minimizing the matching error [see Fig. 3(d)]. The new control point set in the configuration space $\hat{\psi} = \{\psi^1, \psi^2, \dots, \psi^{n'}\}$ can be obtained by solving the optimization problem

$$\min_{\hat{\psi}} \sum_{m=1}^{n'} \left[\left\| \mathbf{r}_{\text{opt}}^m - f(\Psi^m; \hat{\Psi}, \hat{\psi}) \right\|^2 + \alpha \left\| \mathbf{r}_{\text{opt}}^m - \psi^m \right\|^2 \right]. \quad (77)$$

Here, the first term quantifies the discrepancy between the new and original surfaces, while the second term penalizes excessive deviation from the initial seeding, preventing overfitting and potential collapse in undersampled regions [see Fig. 3(d)]. Since f is linear in the control points, this is a linear least squares problem solved directly via its Hessian matrix, yielding the final optimal control points $\hat{\psi}$. As a result, the matching error is minimized and the surface is accurately reconstructed, as illustrated in Fig. 3(e).

D. Analysis of Surface Reconstruction

We employ a periodic surface reconstruction strategy to maintain mesh quality and numerical conditioning. During morphogenesis, control-point distributions tend to drift (e.g., local sparsity or clustering), which degrades the surface fidelity and deteriorates the optimization's numerical conditioning. A full reconstruction restores a near-uniform distribution at a modest cost of about 7.05 s for a typical model with 477 control points. To balance mesh quality and computational efficiency, we conduct a reconstruction every five optimization steps.

The reconstruction process [Fig. 3(b)] begins with the *uniform seeding* step, which redistributes control points evenly

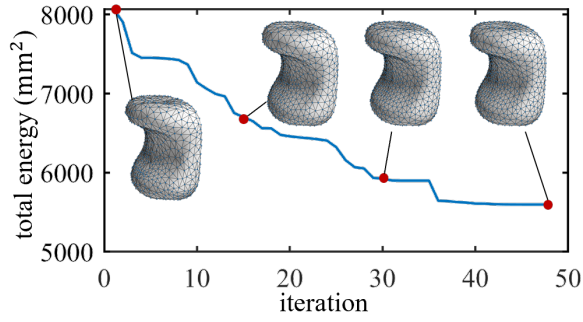


Fig. 4: Convergence of the mesh potential energy E in (73) during control-point redistribution toward a uniform distribution.

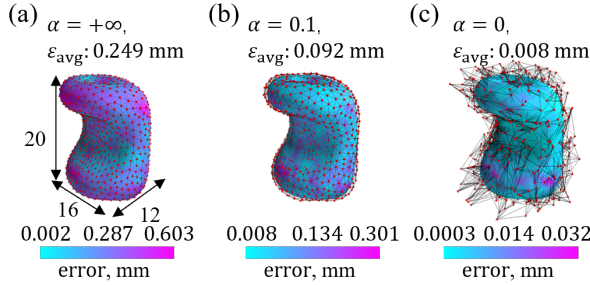


Fig. 5: Effect of the regularization parameter α in the matching error minimization step (77). ϵ_{avg} is the average matching error. (a) A large α prioritizes control point uniformity over shape fidelity, resulting in a smoother surface but a higher matching error. (b) An intermediate $\alpha = 0.1$ provides a good balance. (c) With $\alpha = 0$, the surface perfectly matches the target shape, but the control point distribution may become non-uniform again, compromising mesh quality.

across the surface in the configuration space by minimizing a potential energy function (73). As shown in Fig. 4, as the optimization converges and the objective function decreases, the control points evolve from an initial cluster to a uniform distribution, ensuring a high-quality basis for the new surface.

A key step is the final matching error minimization (77), where the regularization parameter α plays a critical role. As illustrated in Fig. 5, α balances the trade-off between matching the original surface shape and maintaining the uniformity of the newly seeded control points. A large α heavily penalizes deviation from the uniform seeding, preserving mesh quality at the cost of shape accuracy. Conversely, $\alpha = 0$ perfectly matches the shape but risks reverting to a non-uniform distribution. Since both terms in (77) are squared-length quantities, α is dimensionless. Based on our numerical studies, a sufficiently uniform control point distribution is achieved for $\alpha \geq 0.1$. We therefore adopt $\alpha = 0.1$ for ensuring adequate uniformity while minimizing the matching error.

VI. OPTIMIZATION METHOD

The morphogenesis of the soft robot's workspace is formulated as a shape optimization problem to maximize the workspace volume. The design variables are the coordinates of the control points defining both the exterior and interior

chamber surfaces. Initially, each chamber is a unit sphere and control points are seeded with an approximate spacing of 1 mm.

Since the displacement field is obtained as an implicit solution of the state equation, the corresponding integrals must be evaluated numerically. To facilitate this, Gaussian quadrature is employed to convert the continuous integrals into a discrete summation of weighted function values at the Gaussian points. In this framework, the shape derivative of the workspace is approximated at these Gaussian points, leading to an optimization problem expressed as:

$$\begin{aligned} \max_{\phi^l, l=1,2,\dots,m} \quad & \mathcal{T} = \sum_{g=1}^k \tau^g w_X (\mathbf{U}^g, \mathbf{J}^g) \\ \text{s.t.} \quad & \sum_i L_i (\mathbf{U}^g, \mathbf{p}^g) V_i = 0, \quad g = 1, 2, \dots, k \end{aligned} \quad (78)$$

where τ^g is the Gaussian weight for each case, $\mathbf{U}^g$ and $\mathbf{J}^g$ denote the DoFs and the Jacobian under the g -th load condition $\mathbf{p}^g$, and w_X represents the integrand of the workspace volume integral, evaluated specifically at the Gaussian points. $L_i (\mathbf{U}^g, \mathbf{p}^g) V_i = 0$ is the state equation as given in (32). The specific values for the pressure conditions $\mathbf{p}^g$ and their corresponding weights τ^g for each optimization case are detailed in the Supplementary Material (Table S1-Table S5). These values are determined by applying Gaussian quadrature rules to the workspace integrals defined in Section III and considering the symmetry of the robot designs.

To solve this large-scale optimization problem efficiently, we adopt a trust-region method. This iterative approach approximates the objective function at each step with its first-order Taylor expansion, forming a simpler sub-optimization problem within a trusted local region. Additionally, the optimization must also ensure the manufacturability and structural integrity of the resulting design. To this end, we introduce two key geometric constraints. First, to prevent sharp features that can cause stress concentrations, we impose a curvature constraint. The principal curvatures, c_1 and c_2 , at any point $\mathbf{r}$ on any surface $\mathcal{S}$ are limited by

$$c_1^2 + c_2^2 \leq C_{\text{max}}^2, \quad \forall \mathbf{r} \in \mathcal{S}, \quad (79)$$

where the maximum allowable curvature C_{max} is set to 1.0. Second, to ensure structural viability and prevent self-intersection or intersections between adjacent chambers, we enforce a minimum thickness constraint. The distance between any two points on distinct surfaces, $\mathcal{S}_i$ and $\mathcal{S}_j$, must exceed a minimum threshold

$$\min_{\mathbf{r}_i \in \mathcal{S}_i, \mathbf{r}_j \in \mathcal{S}_j} \text{dist}(\mathbf{r}_i, \mathbf{r}_j) \geq d_{\text{min}}, \quad i \neq j, \quad (80)$$

where d_{min} is set to 2.5 mm. These constraints are integrated into the sub-optimization problem as penalty terms [10], which guide the solver towards feasible solutions.

At each iteration of the trust-region method, the sub-optimization problem is formulated as

$$\min_{\Delta \phi^i, i=1,2,\dots,n} -\frac{\partial \mathcal{T}}{\partial \phi^i} \Delta \phi^i + \sum_j \mathcal{C}_j, \quad (81)$$

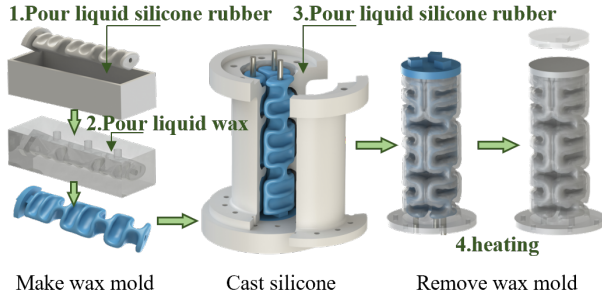


Fig. 6: Fabrication process of the soft robot by the lost-wax method.

where $\Delta\phi^i$ represents the update direction for the i -th control point, and C_j is the penalty term associated with the constraints. This sub-optimization problem is solved to update the design variables

$$\phi^i \leftarrow \phi^i + \Delta\phi^i, \quad i = 1, 2, \dots, n. \quad (82)$$

Within the trust region, (81) is solved using the quasi-Newton method. Once the sub-optimization is complete, the design variables are updated and FEA simulations and sensitivity analysis are performed for the next iteration. The convergence criterion was set to a relative change in the objective function of less than 10^{-2} over the last five iterations.

VII. FABRICATION AND EXPERIMENTAL SETUP

To validate the optimized designs, physical prototypes were fabricated and tested. The FEA simulations are performed using a custom implicit static solver developed in PyTorch. The robot is modeled using the Neo-Hookean hyperelastic material model and discretized using 10-node quadratic tetrahedral elements. For the boundary conditions, the base of the robot is fully constrained, while the nodes on the end-effector surface are coupled to a central reference point whose displacement and orientation define the robot's end-effector configuration. With the actuator pressure as the only input, the dimensionless analysis implies that the deformation is governed by the pressure to shear modulus ratio, i.e., $\tilde{p} = \frac{p}{\mu}$ (see Appendix D). The maximum value of $\tilde{p}$ is set to 0.12 to generate sufficient deformation. This dimensionless limit, combined with the experimentally measured shear modulus, determines the maximum pressure applied in the experimental validation.

Fabricating soft robots with complex internal geometries presents significant challenges. While direct 3D printing of soft materials offers a path to creating intricate structures, the selection of available materials is often limited and exhibits lower mechanical strength compared to castable silicones. In this work, we adopt the lost-wax casting method, as shown in Fig. 6, to realize the targeted internal cavity geometries. In this process, the interior molds are fabricated from wax, which can be removed by heating, while the exterior molds are produced using 3D printing. First, a hard model of the chamber is 3D printed, and then a silicone rubber mold is created from this model. Liquid wax is subsequently poured

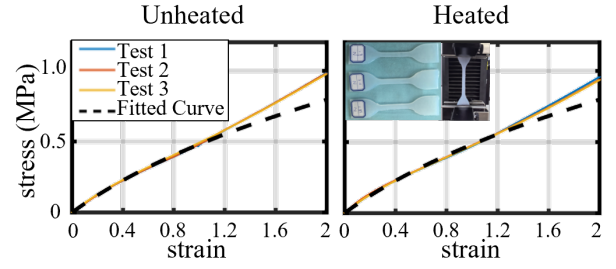


Fig. 7: Tension tests of the silicone rubber material. Curve fitting of the stress-strain data yields shear moduli $\mu = 465$ kPa for the unheated sample (left) and $\mu = 471$ kPa for the sample heated in boiling water at 100°C for 20 minutes (right).

into the silicone mold to form the wax mold. Once the wax mold is obtained, the soft robot is fabricated by casting silicone rubber (Posilicone 40, from Shenzhen Puston Silicone Materials Co., Ltd), which possesses a Shore hardness of 40A. We quantified geometric errors in the fabrication process: the 3D-printed rigid mold tolerance is about 0.006 mm, the wax shrinkage is about 0.012 mm, and the mold assembly error is about 0.1 mm. These factors contribute to a cumulative geometric deviation of about 0.12 mm. Among these factors, the dominant geometric deviation in our fabrication process mainly arises from mold positioning and assembly, whereas the choice of loss-wax method has a much smaller influence on the error. Therefore, improving mold positioning and assembly precision is expected to further reduce the manufacturing error of the actuator. While the shape derivative in Section IV can predict an upper bound for workspace sensitivity to such geometric perturbations, the actual combined effect of multi-source manufacturing errors is often smaller and requires case-by-case assessment through experimental characterizations. For instance, the workspace error for the 2A0R2T case was approximately 1.03%, whereas that for the 3A2R0T case is about 4.44%.

Uniaxial tensile tests were performed using an Instron 68SC universal testing machine to calibrate the shear modulus. The samples, with dimensions of $33 \text{ mm} \times 6 \text{ mm} \times 2 \text{ mm}$, were stretched to 200% strain at a constant rate of 20 mm/min, with tests performed under two conditions: without heat treatment and after heating in boiling water at 100°C for 20 minutes to mimic the wax removal process. Fitting the stress-strain curves yields $\mu = 465$ kPa (unheated) and $\mu = 471$ kPa (heated), corresponding to a relative change of 1.2%. This indicates that the heat treatment has a negligible effect on the material's shear modulus. The bulk modulus κ is set to 100μ to enforce near-incompressibility. The stress-strain curves and fitting results are presented in Fig. 7.

The Neo-Hookean model proves effective for capturing the strain range of the soft actuators. The FEA results show that the maximum principal strain across all cases does not exceed 100%, and the tensile tests indicate that the Neo-Hookean model provides accurate stress-strain predictions up to 120% strain. Additionally, other material models, such as the Mooney-Rivlin model and Ogden model, can be readily integrated into the proposed design framework. Only a modi-

fication to the strain energy density function is required, while the analytical expression of the shape derivative and the entire optimization algorithm remain unchanged.

To validate the optimized designs, physical prototypes were tested. The loading was applied by controlling the pressure in the pneumatic chambers. To ensure quasi-static loading, a low pressurization rate of 1.0 kPa/s was adopted, and the maximum pressure reached 56.52 kPa (i.e., corresponding to $\bar{p} = 0.12$) to align with the FEA simulations. Open-loop pressure control was achieved using a 12-channel custom regulator (Deli Group) equipped with HUBA 528 sensors, with a control accuracy of 0.1 kPa, operated via a dSPACE MicroLabBox 1202 module. The end-effector's position and orientation were measured using an OptiTrack motion capture system, with three markers attached and a calibrated measurement accuracy better than 0.2 mm.

VIII. RESULTS

This section presents the workspace optimization for five distinct design problems. The Supplementary Video shows the morphogenesis processes, FEA and experimental results of all investigated cases. We mainly focus on five representative workspace problems summarized below

- 2A0R2T: Planar position workspace in the x - z plane, under two independent actuation inputs.
- 3A0R3T: 3D position workspace under three independent actuation inputs.
- 3A2R0T: 2-DoF orientation workspace describing the set of achievable end-effector's pointing directions, characterized by an area on the unit sphere.
- 3A3R0T: 3-DoF orientation workspace, characterized by the volume of the set of all attainable end-effector rotational configurations.
- 3A1R2T: Workspace combining 2D planar position (x - z plane) with 1-DoF in-plane orientation.

A. 2A0R2T Workspace

We investigate the position workspace in the x - z plane, and assume that the structure is symmetric with respect to the y - z plane. Due to the symmetry, it suffices to evaluate only the Gaussian points on half of the boundary of the actuation space. In particular, by employing a two-point Gaussian quadrature, 4 Gaussian points are selected (see Table S1). The shape derivative of the workspace is then evaluated at these points.

The initial design and iteration history are illustrated in Fig. 8(a). The optimization results shown in Fig. 8(b) yield two symmetric bellows-like chambers that combine the characteristics of both bending and elongation actuators, thereby enhancing the workspace in both the x and z directions. FEA simulations are conducted on the boundary within the allowable pressure range at an interval of $\Delta\bar{p} = 0.006$ to evaluate the workspace. The total workspace area is 1314.05 mm², while the Gaussian quadrature evaluation yields 1359.28 mm². The error between the two results is less than 3.5%, which indicates the accuracy of the method. The average experimental workspace area is 1300.50 mm², which closely matches the FEA result with an error of less than 1.03%, as shown in Fig. 8(c).

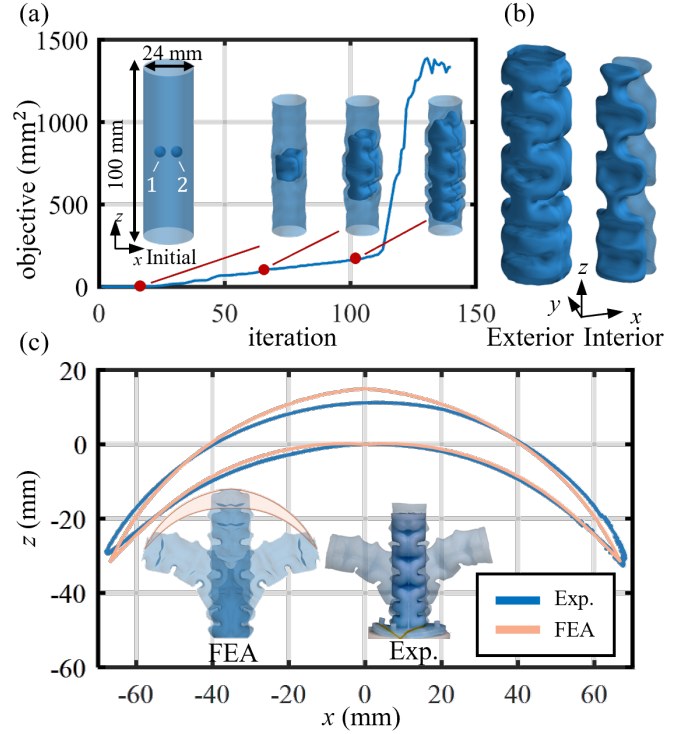


Fig. 8: 2A0R2T workspace design. (a) Initial design and optimization history, showing the evolution of morphology and workspace area. (b) The final optimized design. (c) Comparison of deformation and resulting workspace between FEA and experimental validation.

B. 3A0R3T Workspace

In the 3A0R3T case, the three air chambers are rotationally symmetric about the z -axis and each chamber is axisymmetric. Using Gauss's divergence theorem, the shape derivative of the workspace is evaluated over the boundary surface. Owing to symmetry, this derivative is computed at 6 Gaussian points (see Table S2).

Fig. 9 presents the optimization results for the 3A0R3T workspace. The optimized design exhibits bellows-like structures, analogous to the 2A0R2T workspace result, and yields a total workspace volume of 89364.4 mm³ as evaluated by FEA. The mean deviation of experimental displacements at sampling points from the FEA results is approximately 9.3 mm [see Fig. 9(c)]. This discrepancy is mainly attributed to self-contact during large deformation, which was not incorporated in the optimization, as discussed in Section IX-C.

C. 3A2R0T Workspace

For the 3A2R0T workspace design, the symmetry is analogous to that in the 3A0R3T case. The workspace boundary can be evaluated with one chamber pressurized to its maximum value, another chamber depressurized, and the third chamber's pressure varying. Consequently, the integration in evaluating (26) is performed along 6 lines on the boundary surface of the actuation space. Owing to the workspace's symmetry, two Gaussian points are required (see Table S3).

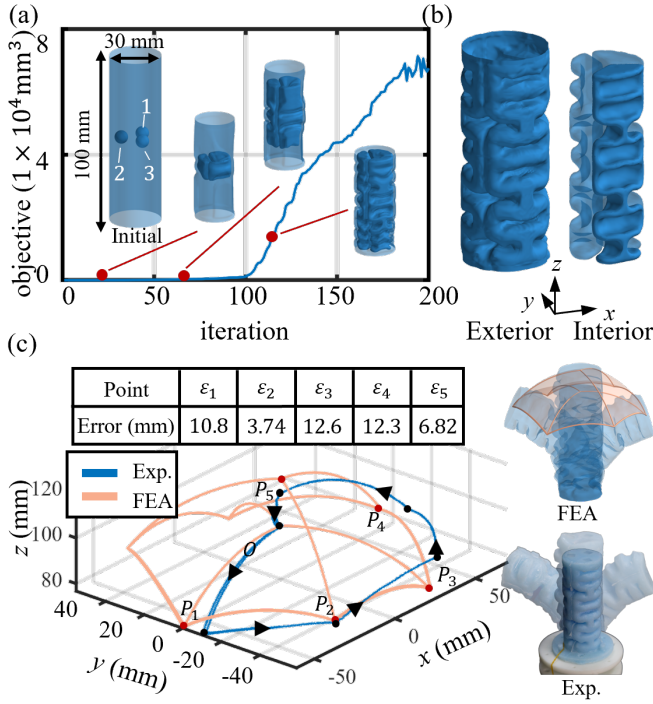


Fig. 9: 3A0R3T workspace design. (a) Initial design and optimization history, showing the evolution of morphology and workspace volume. (b) The final optimized design. (c) Comparison of deformation and resulting workspace between FEA and experimental validation.

The optimization outcomes are presented in Fig. 10. When one chamber is pressurized to $\tilde{p} = 0.12$ while the other two remain at zero, the end-effector rotates by approximately 77.7° . With two chambers at $\tilde{p} = 0.12$ and one at zero, the rotation is approximately 74.3° . The minimum bending angle is around 66.1° . The overall area of the reachable orientation on the unit sphere is measured as 4.05. Experimental validation in Fig. 10(c) shows that the average orientation deviation from the FEA results is around 5.09° and the workspace area is 3.87, which is close to the FEA result with an error of less than 4.44%.

D. Comparison of Optimization Results

Pneu-Nets actuators are among the most widely used pneumatic actuators, owing to their excellent elongation and bending deformability [41]. As such, they were selected as the benchmark for comparison. For a fair comparison, the Pneu-Nets actuators were selected such that their dimensions, materials, and wall thickness align with our optimized designs, and were actuated under the identical pressure range ($\tilde{p} \in [0, 0.12]$).

The performance gains from our free-form optimization are substantial. Compared to the baseline Pneu-Nets actuators, the optimized designs show significant improvements across all cases:

- In the 2D case, the workspace area increased by 291% (Fig. 11).
- For the 3D position-optimized design (3A0R3T), the workspace volume increased by 828% (Fig. 12).

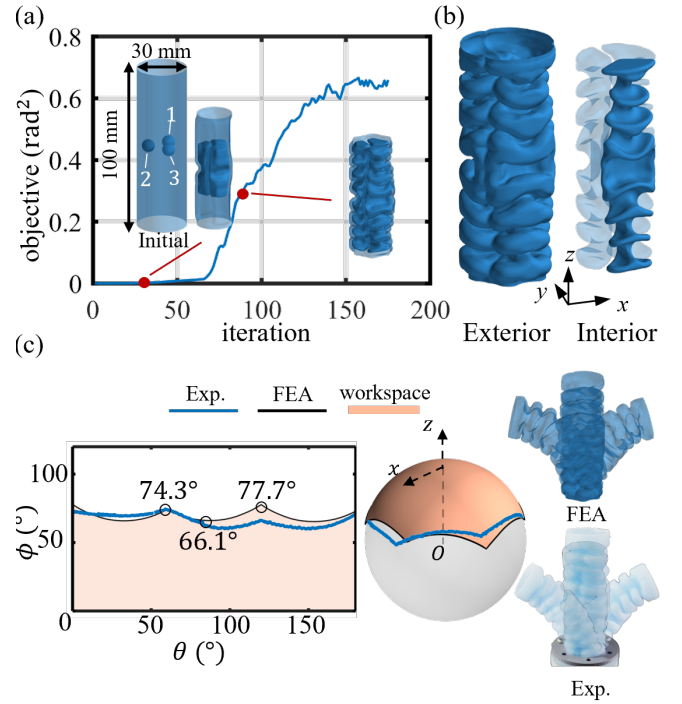


Fig. 10: 3A2R0T workspace design. (a) Initial design and optimization history, showing the evolution of morphology and workspace area. (b) The final optimized design. (c) Comparison of deformation and resulting workspace between FEA and experimental validation.

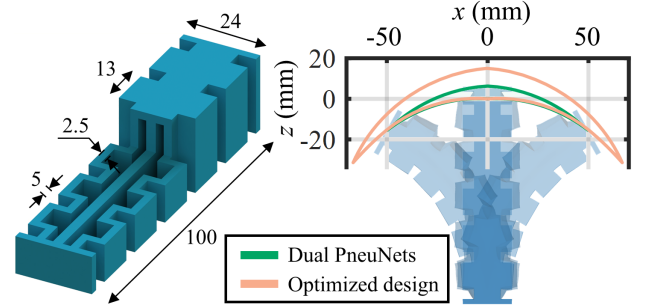


Fig. 11: Comparison between the optimized design and the empirical Pneu-Nets design.

- For the 3D orientation-optimized design (3A2R0T), the workspace area increased by 158% (Fig. 12).

E. 3A3R0T Workspace

The 3A3R0T workspace optimization addresses the 3-DoF orientation space, aiming to maximize the volume of the set of all attainable end-effector rotational configurations. The design incorporates rotational symmetry for both the exterior surface and internal chamber geometry, while deliberately excluding axial symmetry to enhance orientation capability. A total of 8 Gaussian points are employed to evaluate the shape derivative of the workspace (see Table S4).

As demonstrated in Fig. 13, each chamber possesses both bending and twisting capabilities. Deformation patterns are shown for 1-, 2-, and 3-chamber pressurization cases. The total

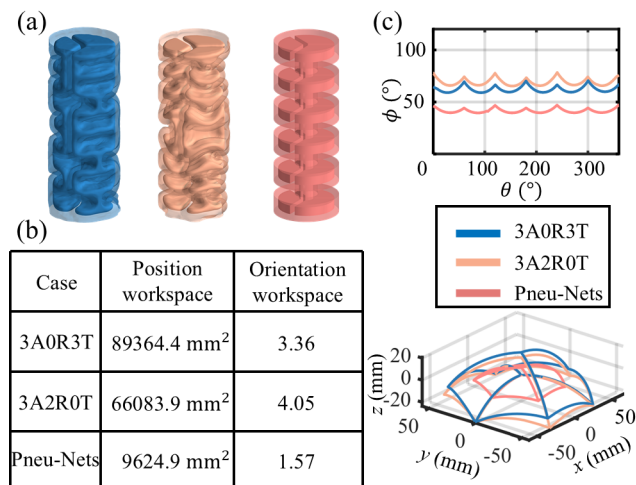


Fig. 12: Comparative analysis of workspace volume between 3A0R3T, 3A2R0T, and conventional Pneu-Nets designs. (a) Morphological comparison of optimized designs versus baseline Pneu-Nets design. (b) Quantitative comparison of the workspace volume. (c) Demonstration of the workspaces.

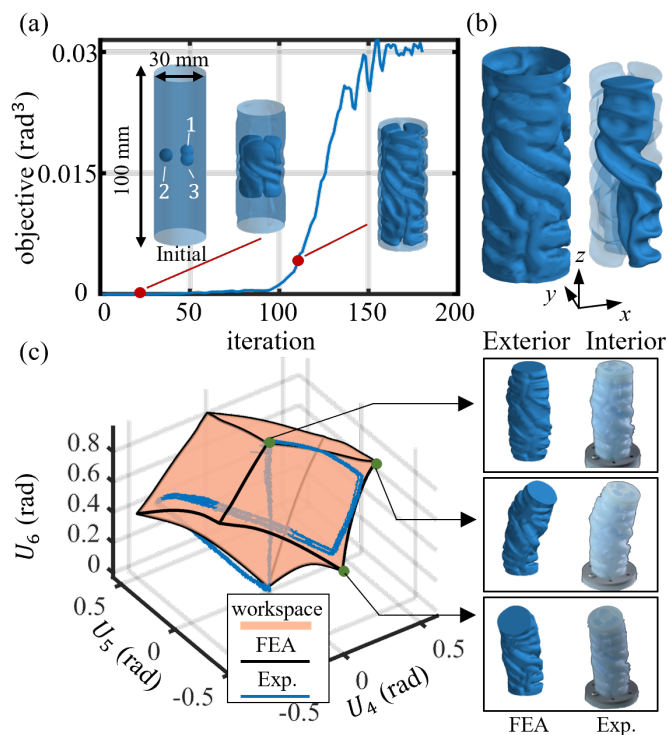


Fig. 13: 3A3R0T workspace design. (a) Initial design and optimization history, showing the evolution of morphology and workspace volume. (b) The final optimized design. (c) Comparison of deformation and resulting workspace between FEA and experimental validation.

volume of the orientation workspace is 0.0299 as measured by FEA.

Unlike the previous example 3A2R0T (which excludes axial rotation), the optimized design in this case further incorporates a helical feature, enabling twisting motion, which is consistent

with the characteristics of previously reported twisting actuators (e.g., [42], [43]). Under most actuation pressures, the robot outputs combined bending and twisting motions. When the pressures in all three chambers are equal, the bending effects cancel out, resulting in pure twisting motion.

The 3A2R0T workspace optimization yields a U_4, U_5, U_6 workspace that manifests as a shell without thickness, effectively prioritizing a wider range of bending angles at the expense of axial rotation. In contrast, the current 3A3R0T optimized design achieves omnidirectional bending capabilities with a maximum angle of 40.1° and a minimum of 28.7° . When the pressures in all three chambers are equal, the resulting pure twisting motion achieves an angle of 53.3° . This indicates a design trade-off where some bending capacity is exchanged for enhanced twisting motion, leading to a 3-DoF orientation workspace.

F. 3A1R2T Workspace

Finally, we present the in-plane 3A1R2T workspace design, which comprehensively accounts for both translational and rotational DoFs. This provides complete control over the end-effector's in-plane configuration. Specifically, for in-plane deformation within the x - z plane, the translational DoFs are characterized by U_1 and U_3 displacements, while the rotational DoF is represented by U_5 about the y -axis. By considering both the position and orientation workspaces, the overall 3A1R2T workspace is defined by

$$\mathbb{W} = \int_{\Omega_2} \int_{\Omega_1} \mathrm{d}\nu_1 \mathrm{d}\nu_2, \quad (83)$$

where Ω_1 denotes the position domain and Ω_2 denotes the orientation domain. Similarly to the 3A0R3T case, this case is evaluated using Gauss's divergence theorem. The objective function in (78) is then formulated as

$$w_{1\text{R}2\text{T}}(\mathbf{U}, \mathbf{J}) = \frac{1}{3} \tilde{\mathbf{U}} \cdot (\tilde{\mathbf{J}}_{\zeta_1} \times \tilde{\mathbf{J}}_{\zeta_2}), \quad (84)$$

with $\tilde{\mathbf{U}} = (U_1, U_3, U_5)$ and $\tilde{\mathbf{J}}_{\zeta_i} = (J_{\zeta_i}^1, J_{\zeta_i}^3, J_{\zeta_i}^5), i = 1, 2$. Since this case lacks symmetry, the shape derivative is evaluated at all 24 Gaussian points on the boundary (see Table S5).

The optimization results reveal a design with three chambers exhibiting a 180° rotational symmetry about the line $y = 0, z = 50$, as illustrated in Fig. 14(b). Notably, the orientation near the robot's base has a significant effect on the end-effector position, while its contribution to orientation is nearly uniform across different positions. As a result, chambers 2 and 3 bend in the same direction to counterbalance the bending of chamber 1, thereby effectively enlarging the 3A1R2T workspace.

Fig. 14(c) illustrates the workspace of the optimized design, with the orientation angles converted from radians to degrees. This demonstrates that the soft arm can achieve an orientation greater than 50.00° over most reachable areas. Moreover, the optimized design exhibits significant kinematic decoupling capabilities. Fig. 14(d) showcases the ability to control orientation while maintaining a fixed end-effector

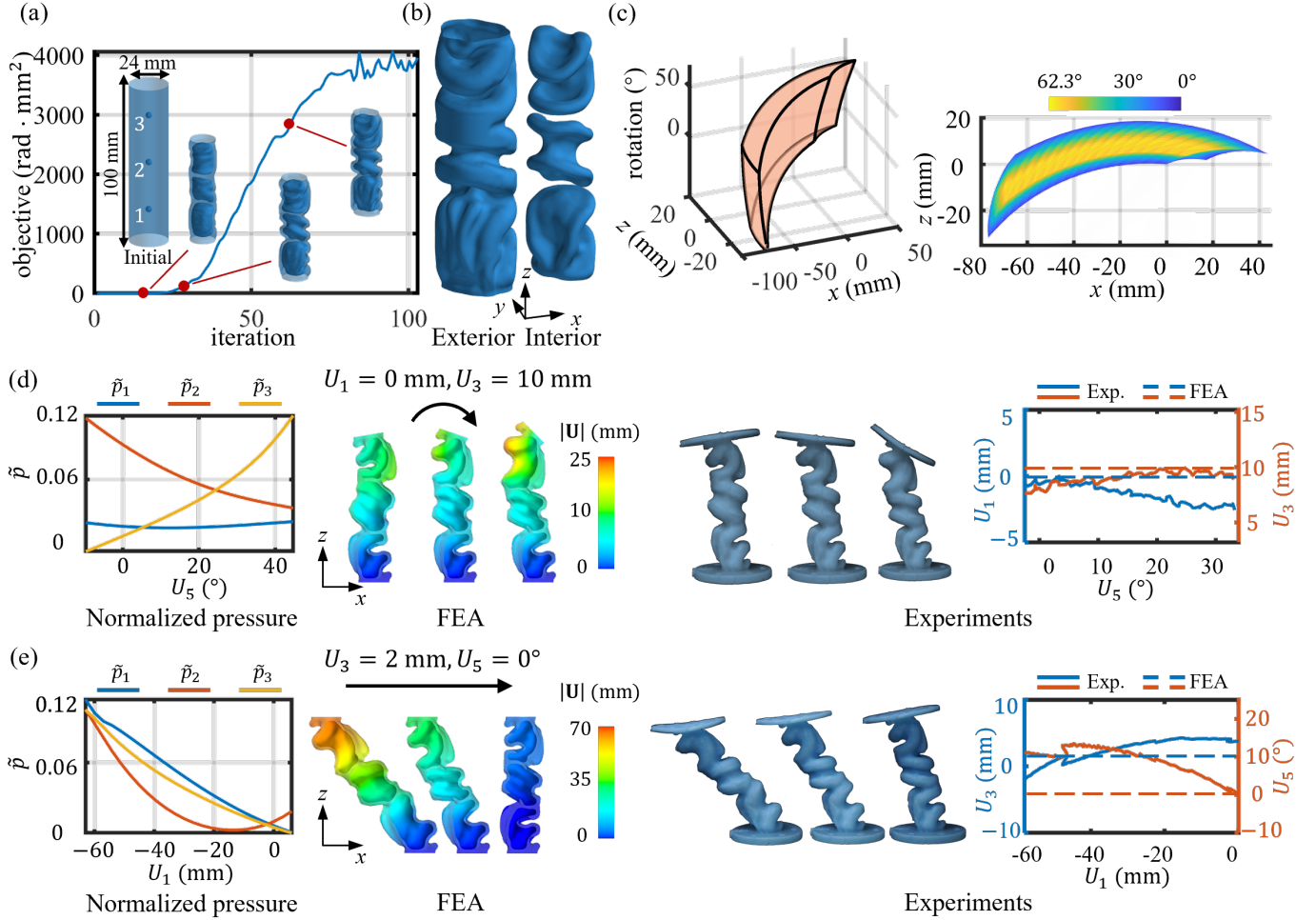


Fig. 14: 3A1R2T workspace design. (a) Initial design and optimization history, showing the evolution of morphology and workspace volume. (b) The final optimized design. (c) The workspace and the reachable range of orientation angles at different positions. (d) FEA and experiment results for fixed position deformation. (e) FEA and experiment results for fixed orientation angle deformation.

position, while Fig. 14(e) demonstrates translational motion at a constant end-effector orientation. To determine the pressure for each chamber corresponding to the above motions, inverse kinematics is formulated based on FEA. This inverse problem is set up as a nonlinear least squares problem

$$\min_{\tilde{\mathbf{p}}(\tilde{p}_1, \tilde{p}_2, \tilde{p}_3)} \frac{1}{2} \left\| \tilde{\mathbf{U}}(\tilde{p}_1, \tilde{p}_2, \tilde{p}_3) - \tilde{\mathbf{U}}^* \right\|^2 \quad (85)$$

with $\tilde{\mathbf{U}}^*$ the target displacement. This optimization problem is solved via the Gauss-Newton method using the Jacobian matrix $\mathbf{J} = \partial \tilde{\mathbf{U}} / \partial \tilde{\mathbf{p}}$ and the updating scheme

$$\tilde{\mathbf{p}} \leftarrow \tilde{\mathbf{p}} - (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T (\tilde{\mathbf{U}} - \tilde{\mathbf{U}}^*). \quad (86)$$

Based on the pressures derived from this optimization, the experimental tests are carried out as shown in Figs. 14(d) and (e). The results indicate that the mean deviation is less than 1.0 mm for the fixed position task, and 5.2°, 3.9 mm for the fixed orientation task.

IX. DISCUSSION

A. Computational Cost

The computational cost of the optimization process arises mainly from the FEA. To accurately evaluate both the workspace and its shape derivative, FEA must be performed along the boundaries of the actuation space. If the dimension of the actuation is denoted by n , then the allowable input space is a n -dimensional hypercube with $2n$ faces, each an $(n-1)$ -dimensional hypercube. Consequently, FEA must be performed on each of these faces, leading to a total of $2nm^{n-1}$ simulations, where m is the number of Gaussian points. While this computational effort is manageable for up to 3 DoAs (with 2 Gaussian points yielding 24 simulations), the cost escalates exponentially with increasing DoAs. For example, with 6 DoAs and 2 Gaussian points, a total of 384 FEA simulations would be required.

The computational cost for each optimization case is summarized in Table IV, which reports the total number of iterations, the number of FEA simulations per iteration, the average time per iteration, and the total optimization time. Notably, since the workspace evaluation entails numerous FEA

TABLE IV: The Computational Cost of Workspace Optimization

Case	Iter.	FEA per Iter.	Time per Iter. (min)	Total (min)
2A0R2T	140	4	4.5	628
3A0R3T	200	6	9.2	1841
3A2R0T	175	2	8.2	1432
3A3R0T	180	8	9.9	1785
3A1R2T	103	24	11.3	1164

* Tests on Intel Core i9-12900K, NVIDIA Titan V, 128 GB RAM.

simulations at the boundary of the actuation space, gradient-free methods such as genetic algorithms are impractical, as they incur prohibitive computational cost due to the need for workspace evaluations of randomly generated design candidates. Likewise, gradient estimation via finite difference methods is also computationally prohibitive, as it requires at least $N+1$ workspace evaluations per iteration for N design variables. This underscores the necessity of deriving the shape derivative to enable gradient-based morphogenesis with a minimal number of workspace evaluations, which constitutes a key contribution of the present work.

To mitigate this computational burden, two potential strategies can be employed. The first is to exploit the inherent symmetry of the workspace. For instance, in the 3A0R3T workspace, rotational symmetry about the z -axis and axisymmetry can be leveraged to reduce the number of required FEA simulations. The second solution involves using parallel computing to accelerate the FEA process since the simulations are independent of each other.

B. Possible Extensions

Other actuation methods. Although this work investigates only pneumatic actuation, the developed workspace optimization method can be extended to various actuation modalities such as cable actuation, dielectric elastomers, magnetic actuation, or their combinations. The required modifications primarily involve the external work terms in (29) and (30), the constitutive models of the incorporated active materials, and their corresponding geometric model and FEA simulators. Another potential extension is to apply the design framework to reconfigurable or metamorphic structures [44]. Kinematic mobility induced by reconfiguration and soft material deformation must be accounted for simultaneously to evaluate the workspace. Nevertheless, the core methodological workflow remains unchanged, including the second-order adjoint sensitivity analysis and the optimization algorithms.

6A3R3T workspace design. The inverse design can be further extended to the 6A3R3T workspace. Combining (4) and (13), a 6D integration over $(U_1, U_2, U_3, U_4, U_5, U_6)$ can be employed as the objective function for workspace optimization. Moreover, by applying Gauss's divergence theorem to evaluate the workspace at the boundary, the high-dimensional integration problem can be effectively reduced to a 5D integration.

Workspace shape design. This work primarily focuses on maximizing the overall workspace volume, assigning uniform weight to all points within the reachable space. However, the framework can be readily extended to optimize the workspace

shape by introducing a non-uniform weight function within the workspace integral defined in (1). This allows prioritizing specific regions of the workspace based on task requirements. For instance, in the design of inchworm-inspired soft robots [2], a large position workspace achieved while maintaining a specific range of end-effector orientations (e.g., bends by approximately 90°) is critical for surface transitioning (e.g., wall-climbing). This requirement translates into assigning higher weights to states with the desired orientation in the workspace optimization, thereby tailoring the workspace shape to enhance the transitioning performance.

C. Limitations

First, self-contact is not accounted for in the FEA model, which may give rise to the sim-to-exp error as observed in the 3A0R3T case. To precisely capture contact behaviors while ensuring the differentiability of the Jacobian, a differentiable contact model is indispensable for contact-aware design. Potential approaches encompass third-medium methods [45], differentiable simulators [46], and the Incremental Potential Contact model [47].

Second, our framework employs a deterministic gradient-based optimization method. As is typical of gradient-based optimization methods, the optimization process may converge to a local optimum, implying that different initial designs may lead to different local minima. Nevertheless, our numerical results indicate that, while the final morphologies may exhibit variations, they typically converge to designs with analogous key features and comparable performance.

Finally, this work formulates, optimizes, and validates the workspace under a quasi-static assumption, with a primary focus on static reachability. Unlike rigid robots, soft robots exhibit dynamic deformation induced by inertia, which can transiently extend their workspace. This represents a promising direction for future research.

X. CONCLUSION

This work presents a computational morphogenesis framework for optimizing the workspace of soft pneumatic robots. By integrating a continuum Jacobian analysis with its analytical shape derivative obtained via a second-order adjoint method, and employing a novel differentiable closed-surface geometric model with adaptive reconstruction, our method effectively addresses the challenge of optimizing robot morphology for the rational design of workspace. We demonstrate the framework's effectiveness through the optimization of various workspace design problems (2A0R2T, 3A0R3T, 3A2R0T, 3A3R0T and 3A1R2T). As validated through both simulation and experiments, the optimized designs show significant improvement over intuitive, manually-derived morphologies. This work establishes a theoretical foundation and provides an end-to-end gradient-based approach for automated soft robot workspace design. The enhanced dexterity and reachability enabled by this approach are anticipated to facilitate broader practical implementations of soft robots across various domains.

APPENDIX A

HYPERELASTIC MATERIAL MODEL AND ITS DERIVATIVE

The Neo-Hookean model is used as the hyperelastic material model. The strain energy density is defined as

$$E = \frac{\mu}{2} (\bar{I}_1 - 3) + \frac{\kappa}{2} (|\mathbf{F}| - 1)^2 \quad (87)$$

- [6] L. Cui and J. S. Dai, "Posture, workspace, and manipulability of the metamorphic multifingered hand with an articulated palm," *Journal of Mechanisms and Robotics*, vol. 3, no. 2, p. 021001, 03 2011.
- [7] X. Huang, J. Zou, and G. Gu, "Kinematic modeling and control of variable curvature soft continuum robots," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 6, pp. 3175–3185, 2021.
- [8] F. Chen and M. Y. Wang, "Design optimization of soft robots: A review of the state of the art," *IEEE Robotics and Automation Magazine*, vol. 27, no. 4, pp. 27–43, 2020.
- [9] B. Chen, N. Wang, X. Zhang, and W. Chen, "Design of dielectric elastomer actuators using topology optimization on electrodes," *Smart Materials and Structures*, vol. 29, no. 7, p. 075029, Jul. 2020.
- [10] F. Chen, Z. Song, S. Chen, G. Gu, and X. Zhu, "Morphological Design for Pneumatic Soft Actuators and Robots With Desired Deformation Behavior," *IEEE Transactions on Robotics*, pp. 1–21, 2023.
- [11] B. Caasenbrood, A. Pogromsky, and H. Nijmeijer, "A computational design framework for pressure-driven soft robots through nonlinear topology optimization," in *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*. New Haven, CT, USA: IEEE, May 2020, p. 633–638.
- [12] S. Chen, F. Chen, Z. Cao, Y. Wang, Y. Miao, G. Gu, and X. Zhu, "Topology optimization of skeleton-reinforced soft pneumatic actuators for desired motions," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 4, p. 1745–1753, Aug. 2021.
- [13] S. Koppen, M. Langelair, and F. van Keulen, "A simple and versatile topology optimization formulation for flexure synthesis," *Mechanism and Machine Theory*, vol. 172, p. 104743, 2022.
- [14] H. Kobayashi, F. Gholami, S. M. Montgomery, M. Tanaka, L. Yue, C. Yuhn, Y. Sato, A. Kawamoto, H. J. Qi, and T. Nomura, "Computational synthesis of locomotive soft robots by topology optimization," *Science Advances*, vol. 10, no. 30, p. eadn6129, Jul. 2024.
- [15] L. Chen, C. Yang, H. Wang, D. T. Branson, J. S. Dai, and R. Kang, "Design and modeling of a soft robotic surface with hyperelastic material," *Mechanism and Machine Theory*, vol. 130, pp. 109–122, 2018.
- [16] M. A. Hosseini, H.-R. M. Daniali, and H. D. Taghirad, "Dexterous workspace optimization of a tricept parallel manipulator," *Advanced Robotics*, vol. 25, no. 13–14, p. 1697–1712, Jan. 2011.
- [17] J. Zhao, C. Wu, G. Yang, C.-Y. Chen, S. Chen, C. Xiong, and C. Zhang, "Kinematics analysis and workspace optimization for a 4-dof 3tr parallel manipulator," *Mechanism and Machine Theory*, vol. 167, p. 104484, Jan. 2022.
- [18] S. Kucuk, "Dexterous workspace optimization for a new hybrid parallel robot manipulator," *Journal of Mechanisms and Robotics*, vol. 10, no. 6, p. 064503, Dec. 2018.
- [19] X. Peng, N. Zhang, L. Ge, and G. Gu, "Dimension optimization of pneumatically actuated soft continuum manipulators," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*. Seoul, Korea (South): IEEE, Apr. 2019, p. 13–18.
- [20] Y. Huang, Q. Zhao, X. Liu, H. Chen, and H. Liu, "Enhancing the workspace of magnetic continuum robots through optimized structural design and strategic magnetization," in *2024 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Bangkok, Thailand: IEEE, Dec. 2024, p. 655–660.
- [21] W. Amehri, "Workspace Estimation and Design Optimization of Soft Robots," Theses, Centrale Lille Institut, Feb. 2022.
- [22] D. Chen, Y. Zhang, and S. Li, "Tracking control of robot manipulators with unknown models: A jacobian-matrix-adaption method," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, p. 3044–3053, Jul. 2018.
- [23] C. C. Cheah, M. Hirano, S. Kawamura, and S. Arimoto, "Approximate jacobian control for robots with uncertain kinematics and dynamics," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, p. 692–702, Aug. 2003.
- [24] C. D. Remy, Z. Brei, D. Bruder, J. Remy, K. Buffinton, and R. B. Gillespie, "The "fluid jacobian": Modeling force-motion relationships in fluid-driven soft robots," *International Journal of Robotics Research*, vol. 21 20, p. 0, 2023.
- [25] Z. Chen, F. Renda, A. L. Gall, L. Mocellin, M. Bernabei, T. Dangel, G. Ciuti, M. Cianchetti, and C. Stefanini, "Data-driven methods applied to soft robot modeling and control: A review," *IEEE Transactions on Automation Science and Engineering*, p. 1–16, 2024.
- [26] G. Fang, Y. Tian, Z.-X. Yang, J. M. P. Geraedts, and C. C. L. Wang, "Efficient jacobian-based inverse kinematics with sim-to-real transfer of soft robots by learning," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, p. 5296–5306, Dec. 2022.
- [27] M. Giorelli, F. Renda, M. Calisti, A. Arienti, G. Ferri, and C. Laschi, "Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature," *IEEE Transactions on Robotics*, vol. 31, no. 4, p. 823–834, Aug. 2015.
- [28] P. Wang, D. Feliu-Talegon, Y. Sun, Z. Xie, W. Xin, M. S. Nazeer, C. D. Santina, C. Laschi, and F. Renda, "Strain-based shape and 3-d force estimation for rod-driven continuum robots with stretch sensors," *IEEE Transactions on Robotics*, vol. 42, pp. 894–911, 2026.
- [29] C. Lee, D. Koo, and D. W. Zingg, "Comparison of b-spline surface and free-form deformation geometry control for aerodynamic optimization," *AIAA Journal*, vol. 55, no. 1, pp. 228–240, 2017.
- [30] L. Mosser, L. Barbé, L. Rubbert, and P. Renaud, "Towards automatic design of soft pneumatic actuators: Inner structure design using cnn model and bézier curve-based genetic algorithm," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, p. 6603–6610, Oct. 2023.
- [31] R. A. Yotter, I. Nenadic, G. Ziegler, P. M. Thompson, and C. Gaser, "Local cortical surface complexity maps from spherical harmonic reconstructions," *NeuroImage*, vol. 56, no. 3, p. 961–973, Jun. 2011.
- [32] R. R. Currius, D. Dolonius, U. Assarsson, and E. Sintorn, "Spherical gaussian light-field textures for fast precomputed global illumination," *Computer Graphics Forum*, vol. 39, no. 2, p. 133–146, May 2020.
- [33] Y.-T. Tsai and Z.-C. Shih, "All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation," *ACM Trans. Graph.*, vol. 25, no. 3, p. 967–976, Jul. 2006.
- [34] N. Flyer and G. B. Wright, "Transport schemes on a sphere using radial basis functions," *Journal of Computational Physics*, vol. 226, no. 1, p. 1059–1084, Sep. 2007.
- [35] A. Handa, K. Van Wyk, W. Yang, J. Liang, Y.-W. Chao, Q. Wan, S. Birchfield, N. Ratliff, and D. Fox, "Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 9164–9170.
- [36] A. Walid, G. Zheng, A. Kruszewski, and F. Renda, "Discrete cosserat method for soft manipulators workspace estimation: An optimization-based approach," *Journal of Mechanisms and Robotics*, vol. 14, no. 1, p. 011012, Feb. 2022.
- [37] C. Chen and D. Jackson, "Parameterization and evaluation of robotic orientation workspace: A geometric treatment," *IEEE Transactions on Robotics*, vol. 27, no. 4, p. 656–663, Aug. 2011.
- [38] G. Maloisel, E. Knoop, C. Schumacher, and M. Bacher, "Automated routing of muscle fibers for soft robots," *IEEE Transactions on Robotics*, vol. 37, no. 3, p. 996–1008, Jun. 2021.
- [39] W. Zhang and S. Feng, "Combined parameterization of material distribution and surface mesh for stiffener layout optimization of complex surfaces," *Structural and Multidisciplinary Optimization*, vol. 65, no. 3, p. 103, Mar. 2022.
- [40] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle, "Multiresolution analysis of arbitrary meshes," in *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '95. New York, NY, USA: Association for Computing Machinery, 1995, p. 173–182.
- [41] B. Mosadegh, P. Polygerinos, C. Keplinger, S. Wennstedt, R. F. Shepherd, U. Gupta, J. Shim, K. Bertoldi, C. J. Walsh, and G. M. Whitesides, "Pneumatic networks for soft robotics that actuate rapidly," *Advanced Functional Materials*, vol. 24, no. 15, pp. 2163–2170, 2014.
- [42] F. Chen, Y. Miao, G. Gu, and X. Zhu, "Soft twisting pneumatic actuators enabled by freeform surface design," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5253–5260, 2021.
- [43] Z. Jiao, C. Zhang, W. Wang, M. Pan, H. Yang, and J. Zou, "Advanced artificial muscle for flexible material-based reconfigurable soft robots," *Advanced Science*, vol. 6, no. 21, p. 1901371, 2019.
- [44] G. Jia, H. Huang, H. Guo, B. Li, and J. S. Dai, "Design of transformable hinged ori-block dissected from cylinders and cones," *Journal of Mechanical Design*, vol. 143, no. 9, p. 094501, 2021.
- [45] A. H. Frederiksen, A. Dalkint, O. Sigmund, and K. Poulios, "Improved third medium formulation for 3d topology optimization with contact," *Computer Methods in Applied Mechanics and Engineering*, vol. 436, p. 117595, Mar. 2025.
- [46] E. Ménager, L. Montaut, Q. L. Lidec, and J. Carpentier, "Differentiable simulation of soft robots with frictional contacts," in *2025 IEEE 8th International Conference on Soft Robotics (RoboSoft)*. Lausanne, Switzerland: IEEE, Apr. 2025, p. 1–8.
- [47] Z. Huang, D. C. Tozoni, A. Gjoka, Z. Ferguson, T. Schneider, D. Panozzo, and D. Zorin, "Differentiable solver for time-dependent deformation problems with contact," *ACM Transactions on Graphics*, vol. 43, no. 3, p. 1–30, Jun. 2024.