Motion and Morphology Planning for Autonomous Grasping in Clutter with a Shape-Shifting, Variable-Stiffness Robot

[1]*Luiza Labazanova*, [1]*Liuming Qiu*, [2]*Jose Guadalupe Romero* [1]*Li Cheng*, [3]*Thrishantha Nanayakkara*, [1,*]*David Navarro-Alarcon*

[1]The Hong Kong Polytechnic University, Hong Kong
[2]ITAM, Mexico City, Mexico
[3]Imperial College London, London, UK
*Corresponding author.

Keywords: *Adaptive morphology, variable stiffness, mobile manipulation, full-body grasping, motion planning, robot design*

Autonomous operation in confined spaces demands robots that can simultaneously navigate tight passages and manipulate objects. However, conventional mobile manipulators are often too large for such environments, while compliant soft manipulators, though compact, are typically limited by tethers and a small operational workspace. This paper introduces a novel hybrid robot designed to address this trade-off. The robot achieves mobile manipulation without dedicated end-effectors by leveraging its entire body. It comprises two autonomous locomotion units linked by a Variable-Stiffness Bridge (VSB). This design enables transitions between a rigid state for efficient locomotion and a flexible state for enveloping and securing objects via full-body grasping. We present the second iteration of this platform, featuring a modular VSB design that achieves a 2.5-fold increase in state transition speed. Furthermore, we introduce a comprehensive planning framework, the Voronoi-based Motion and Morphology Planner (VMMP) to enable autonomous navigation in highly cluttered environments. VMMP addresses the high-dimensional planning challenge through a hierarchical decomposition: it first uses Voronoi diagrams to generate traversable paths for three key control points on the robot's body, and then reconstructs the kinematically-feasible robot configuration sequence via constrained optimization. The source code and simulations are available at https://github.com/Louashka/2sr-robot.

1 Introduction

Navigating and operating within unstructured environments presents a formidable challenge for robotic systems. Many real-world scenarios, such as industrial inspection or search and rescue, demand systems that can retrieve objects in cluttered environments with limited access. To be effective, robots must combine robust locomotion with dextrous manipulation capabilities. Early attempts to integrate these functions led to solutions like mobile manipulators, which mount robotic arms on mobile bases [1, 2], or warehouse robots designed to lift and transport cargo [3]. However, these platforms are typically bulky and rely on rigid components, requiring carefully supervised workspaces specifically prepared for their operation.

Robots designed specifically for confined spaces typically follow two paths: compact sensor platforms [4, 5] or multi-legged systems for rough terrain [6, 7]. In either case, integrating manipulation capabilities is a significant challenge, as adding grippers or tools often compromises the robot's primary navigation and mobility functions. This challenge stems from conventional robotics' reliance on rigid components and fixed structures, which constrains functional versatility.

To overcome the constraints of rigid robotics, a paradigm shift towards adaptive morphology has gained significant traction, enabling the creation of multimodal and multifunctional systems [8–10]. This paradigm has manifested in several distinct research directions. One strategy involves reconfigurable truss architectures, where active control of elastic or jointed elements allows a robot to alter its own shape and passive mechanical properties [11, 12]. Such systems have demonstrated the ability to switch between locomotion modes like driving, rolling, and even swimming [13]. While promising, their range of shape change is often constrained, and their form factor can be too large to navigate narrow passages or perform fine-grained object interaction.

Another strategy involves modular robotics, which leverages collective behaviors from individual components. These systems achieve complex tasks through coupled mechanisms [14,15], multi-agent coordination [16–18], or deformable chains [19–21]. The aggregate mechanics of these modules allow them to squeeze through tight spaces or envelop objects for transport. However, their locomotion is often slow or dependent on specialized environmental conditions, such as vibrating surfaces, which limits their real-world applicability.

The biggest departure from conventional design is found in soft robotics, whose inherent compliance and continuous deformation offer unparalleled adaptability to the environment [22,23]. Soft robots excel at exploring constrained spaces like pipelines [24, 25], but often at the cost of payload capacity and force output. To address this, nature-inspired designs have emerged, such as inchworm-like [26] and ant-like [27] soft robots capable of both locomotion and grasping. Soft components can also augment other architectures; for instance, the truss robot Tetraflex [28] uses inflatable bellows to roll, crawl, and transport objects. Nevertheless, a major drawback of many soft systems is their reliance on pneumatic actuation, which typically requires a tether, results in limited mobility, and offers low payload capacity. While untethered alternatives exist, such as the isoperimetric robots developed by Usevitch et al. [29], their large scale (0.85 m^2 to 2.9 m^2) and unproven scalability highlight the ongoing challenges.

The preceding survey reveals the inherent trade-off between soft and conventional rigid robots that present designers with a fundamental dilemma: a choice between the efficiency, autonomy, and force capacity of rigid systems and the adaptability and versatility of soft, morphing systems. Hybrid robotics has emerged as a powerful synthesis, seeking to

combine the "best of both worlds" by integrating soft, flexible, and rigid components into a single platform [30–32]. These hybrid platforms have demonstrated remarkable versatility, including the ability to seamlessly switch between distinct locomotion gaits and operational modes [33–35].

An important innovation within this hybrid paradigm is the implementation of variable-stiffness materials, which grant robots the ability to be rigid or compliant when needed, significantly expanding their functional envelope. Several technologies have been explored to this end. Shape-memory materials (SMMs), such as shape memory alloys (SMAs), can be programmed to hold temporary shapes and revert to their original form upon thermal stimulation [36]. This principle has been used to create crawling microrobots [37, 38]. Another effective approach is particle jamming, where vacuum pressure is used to transition a granular material between a fluid-like and a solid-like state [39]. Jamming has been successfully integrated into crawling robots [40] and loop-shaped robots for combined locomotion and manipulation [41].

While effective, these methods have drawbacks. SMAs can have slow actuation cycles, and jamming systems require a constant vacuum source, complicating untethered designs. In this context, low-melting-point alloys (LMPAs) have emerged as a highly promising alternative [42]. LMPAs offer several key advantages: a large stiffness variation between their solid and liquid phases, superior thermal conductivity for rapid phase transitions (e.g., Field's metal at 18 W/(mK) [43] versus 0.15-0.3 W/(mK) for SMPs [44]), and greater suitability for robust, untethered systems. These compelling properties led us to select LMPA-based variable stiffness as the core technology for our work.

Building upon this foundation, we present a novel hybrid robot capable of efficient motion, robust object manipulation, and traversal through highly obstructed environments, uniting mobility, autonomy, and flexibility within a single platform. This paper substantially advances our previous work on the Self-Reconfigurable Soft-Rigid (2SR) robot [45] by introducing a refined mechanical design and a comprehensive control and planning framework. In its rigid state, the robot operates as an efficient omnidirectional platform for planar navigation. In its flexible state, it can fundamentally alter its morphology to perform full-body grasps or conform to environmental constraints.

The implementation of this system required us to address several interconnected technical challenges, and our primary contributions are:

1. An improved robot design centered on a modular variable stiffness bridge composed of compact, independent LMPA based units. This architecture achieves a 2.5-fold improvement in phase transition speed and superior power efficiency over the previous monolithic design, while simultaneously enhancing system robustness.

2. A comprehensive framework that integrates a novel hybrid kinematic model with a hierarchical "Motion & Morphology" (M&M) controller. The kinematic model, based on cardioid motion analysis, describes the robot's complex dynamics across both rigid and flexible states. This model enables the M&M controller to coordinate discrete stiffness transitions with continuous motion control, utilizing mode-specific Model Predictive Controllers (MPC) for precise trajectory tracking and a supervisory finite-state machine for reliable stiffness management.

3. The Voronoi-assisted Motion and Morphology Planning (VMMP) algorithm, a high-level planner that accounts for the robot's complex, high-dimensional (5D) configuration space, which encompasses its planar pose, variable morphology and multiple motion modes. VMMP enables autonomous navigation and shape adaptation of a 2SR robot in densely cluttered environments. The planner intelligently analyzes the environment to compute a sequence of critical configurations, balancing path feasibility with the robot's unique kinematic and morphological constraints.

2 System Architecture and Design

A self-reconfigurable soft-rigid robot consists of two locomotion units (LU) and a variable-stiffness bridge (VSB). The units, equipped with omni-wheels, house all electronics and power supply for autonomous operation. The variable stiffness (VS) bridge can change its compliance, enabling the robot to switch between various locomotion and deformation modes.

When developing a bridge, we followed Tonazzini et al.'s approach [46] that demonstrated a VS fiber composed of a silicone tube filled with low melting point alloy (LMPA) and wrapped by an enamelled wire heater. LMPAs transition from solid to liquid state when heated above their melting temperature T_m, allowing the assembled structure to become soft. The previous design of a 2SR robot [45] featured a bridge with two independently controlled segments. Each segment measured 8mm in diameter and 40mm in length, with a layered structure consisting of a silicone rod (5mm diameter), coiled enamelled wire (0.1mm diameter), and Field's metal (32.5% bismuth, 51% indium, 16.5% tin by weight; $T_m \approx 62°C$, Young's modulus > 3 GPa), all encapsulated in a silicone tube. The fabricated robot demonstrated shape-shifting capabilities and promising potential for grasping and manipulating objects.

However, this design has limitations including slow phase transitions (> 100 seconds), high power consumption (~ 9V for each segment), and bending constraints. VS segments were joined through a rigid 30mm plastic connector that restricted deformation in the middle of the bridge. To address these limitations, we propose a new bridge design incorporating a cable chain with a modular VS backbone. Compact VS modules reduce heating area and alloy volume, enabling faster transitions and lower power consumption. A single cable chain ensures uniform bending throughout the bridge while maintaining the ability to control segments independently by arranging modules in separate circuits. The modular approach enhances flexibility and robustness, facilitating length adjustments and easy replacement of failed or damaged modules.

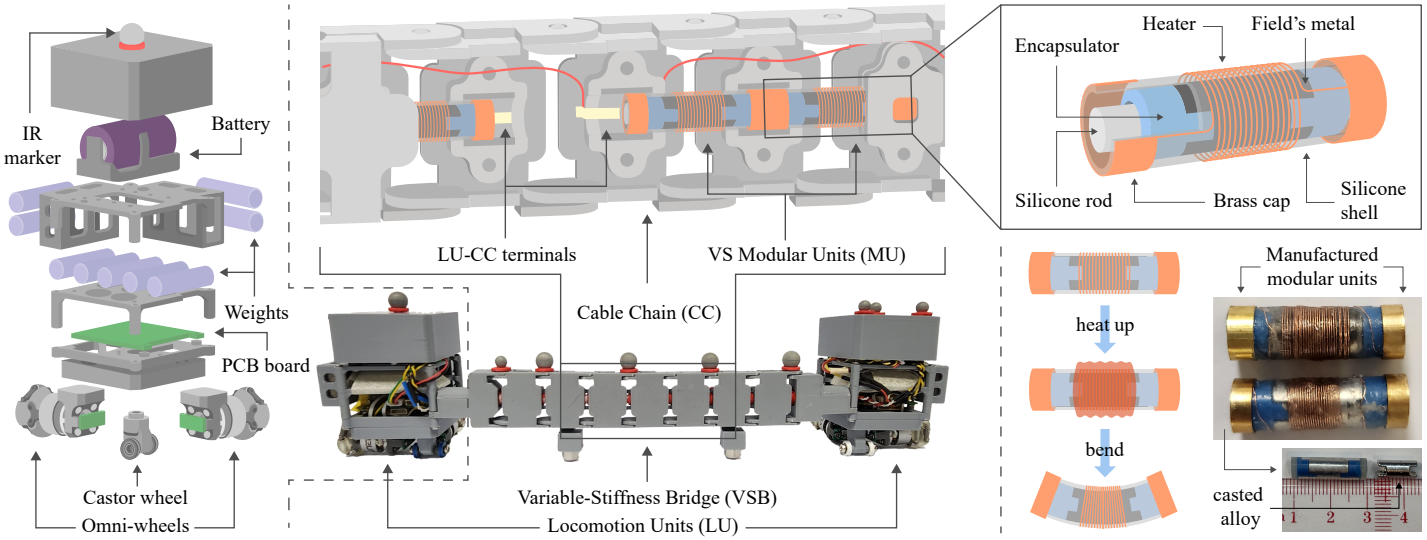

Figure 1: Structural overview of the 2SR robot: exploded view of the locomotion unit showing its internal architecture; composition of the assembled variable-stiffness (VS) bridge with modular units forming a cable chain backbone; detailed view of a single VS modular unit showing its components and operating principle; photographs of manufactured components including the cast alloy layer and fully assembled module.

Modular Variable-Stiffness Bridge

The variable-stiffness bridge consists of a cable chain structure that connects locomotion units, with chain nodes featuring internal channels to house VS modular units (MU). Each MU follows the core principle of unitary VS segments but introduces several key improvements in design and fabrication.

The MU comprises a silicone rod (3mm diameter) running through a Field's metal layer (10mm length, 3mm inner diameter, 5mm outer diameter), encapsulated by plastic caps at both ends, all assembled within a silicone shell (5mm inner, 6mm outer diameter). Unlike the previous design where the heater was positioned between the silicone rod and alloy, a copper wire (0.1mm diameter) is now coiled around the silicone shell. This modification prevents wire damage during fabrication and enables easier heater replacement without destroying the alloy-containing section. Instead of external cables, brass caps are soldered to the heater and mounted at the unit's ends.

Each manufactured modular unit measures 20mm in length and 7mm in diameter. The fabrication process uses precision molding for the alloy layer and automated coiling to produce a 1.7Ω heater. This resistance was selected as a trade-off to enable faster phase transition at 1A while minimizing power consumption by requiring only 1.7V. The assembly is finished with a protective silicone coating (for more details, see Section S1, Supporting Information).

The bridge comprises two independently controlled segments, each containing four modules ($\leq 4 \cdot 1.7 = 6.8$V). Modules are inserted between chain nodes, with brass caps pressing against copper contacts within node channels for electrical connectivity. At the middle chain node where segments meet, modules have no outward-facing brass caps to prevent electrical interference between segments. Terminal modules at locomotion unit junctions incorporate NTC thermistors for temperature monitoring. Each VS segment connects to an adjacent LU through four cables: two for the sensor and two for the heater (one from the terminal module, another from the middle node contact) to establish the control circuit. A calibration model translates resistance readings from the NTC thermistor into an internal temperature estimate for the alloy. On a higher level, a Finite-State Machine manages stiffness by controlling the heating and cooling cycles based on this temperature feedback. Additionally, the controller accounts for the alloy's thermal hysteresis by using distinct temperature thresholds for melting and solidification to ensure reliable transitions between the flexible and rigid states (for more details, see Section S2, Supporting Information).

The cable chain limits the total bending of the bridge to π radians ($\pi/2$ per segment), a constraint that prevents collision between the LUs while preserving sufficient deformation capability. This modular architecture, which pairs the cable chain with a VS backbone, requires significantly less alloy than monolithic approaches. This reduction directly translates to faster phase transitions and lower power consumption. Experimental validation confirms these advantages: the average rigid-to-flexible transition occurred in 35.9 s, while the flexible-to-rigid transition took 47.1 s. This represents a 2.5-fold improvement in overall transition speed compared to the previous monolithic design (for more details, see Section S3, Supporting Information).

Locomotion Units

Locomotion units (LU's) are designed as self-contained, non-holonomic robots ($47 \times 47 \times 75.5$mm). Each unit has two custom-made omni-wheels with a radius of 10mm, positioned on adjacent sides of the unit frame facing outwards. A castor wheel is used to create a three-point contact with the substrate and stabilize the unit's motion. Steel bars within the frame serve as weights to ensure solid wheel-ground contact.

Each locomotion unit is equipped with AS5600 encoders,

a Li-ion battery (3.7V, 1000mAh) boosted to 9.6V, and an STM32-based controller. This controller operates brushless DC motors to drive the wheels and actuates a modular VS backbone inside the bridge. The signal captured from the segment's NTC thermistor is converted into temperature values that are used to generate appropriate PWM output to control the MU heaters. Communication with the central computer is established through a wireless serial board.

This configuration allows precise VSB manipulation in its soft state while maintaining three degrees of freedom when rigid, regardless of bridge shape. The complete 2SR robot structure is shown in Fig. 1.

3 Motion and Morphology Control

The dual capability of the 2SR robot to function as both an omnidirectional mobile platform and a deformable manipulator expands its potential applications. However, this versatility presents a unique challenge in developing a unified control framework that accommodates the distinct characteristics of both operational modes. This section introduces a hybrid kinematic model that seamlessly integrates rigid locomotion and soft manipulation, along with a control strategy enabling smooth transitions between modes. To facilitate the mathematical formulation, we establish the following key assumptions:

- VS segments are inextensible i.e. their length is constant.
- Flexible segments deform into constant-curvature arcs.
- LUs move independently when segments are flexible.
- The robot maintains omnidirectional mobility in a rigid state through strategically positioned omni-wheels, independent of its configuration.

3.1 Hybrid Kinematics

Given the shape of the robot is not constant, defining its configuration is not trivial. Based on our constant-curvature assumption, each VSS configuration can be fully determined by three parameters: the position of one endpoint, the tangent direction at this endpoint, and VSS curvature. Since both segments share a common endpoint at the bridge's approximate midpoint, we assign the robot's body frame $\{B\}$ at this location, oriented along the local tangent direction. This allows us to describe the robot's complete configuration using five generalized coordinates: two for position x_b, y_b, one for orientation θ_b, and two curvature values κ_1 and κ_2 (one for each segment s_1 and s_2), so that it can be expressed as $\mathbf{q} = [x_b, y_b, \theta_b, \kappa_1, \kappa_2]^\intercal$. With this configuration defined, the positions of the wheels can be computed from LU geometry.

3.1.1 Rigid State

To analyze the robot's motion, it is essential to identify its velocities and the mechanisms generating them in each stiffness state. When rigid ($s_1 = s_2 = 0$), the 2SR agent behaves akin to a conventional omnidirectional robot.

Then, its kinematics can be defined as:

$$\dot{\mathbf{q}} = \underbrace{\overline{(s_1 \vee s_2)} \begin{bmatrix} \mathbf{R}_z(\theta_b) \\ 0 \end{bmatrix}}_{\mathbf{J}_r} \mathbf{u}_r, \tag{1}$$

where $\mathbf{R}_z(\theta_b) \in \mathbb{R}^3$ is a rotation matrix around the vertical axis of the global frame and $\mathbf{u}_r = [v_{bx}, v_{by}, \omega_b]^\intercal$ is a vector of robot's "rigid" control velocities associated with $\{B\}$.

3.1.2 Flexible States

During flexible operation, LUs function as independent actuators. To prevent collisions between locomotion units, the cable chain bending is restricted to a half-circle, constraining segment curvatures to $-\frac{\pi}{2l} \leq \kappa_i \leq \frac{\pi}{2l}$. It is observed that during the bending of the bridge, LUs trace a distinctive curve. The previous work [45] employed logarithmic spirals to model this motion, but it required separate functions for positive and negative curvatures, creating discontinuities at boundary conditions. Here, we use a cardioid model, which provides a single mathematical shape to approximate LU paths across all permissible curvature values:

$$x(\phi) = \rho \cos\phi, \quad y(\phi) = \rho \sin\phi, \tag{2}$$

where $0 \leq \phi \leq 2\pi$ is the rolling angle, $\rho = 2r(1 - \cos\phi)$ is the cardioid's radius at ϕ, and r is the common radius of the two generating circles of the cardioid.

To fully comprehend the robot kinematics in a soft state, we categorize the operational modes into three distinct scenarios where different combinations of bending and motion occur. Each scenario is associated with a unique cardioid, see Fig. 2:

1. One segment is flexible while the other remains rigid. The LU adjacent to the flexible segment is in motion.
2. One segment is flexible while the other remains rigid. The LU adjacent to the rigid segment is in motion.
3. Both segments are flexible, with either LU moving.

Through path analysis and curve fitting, we determine the cardioid radius r and the corresponding ϕ range for each scenario, as listed in Table 1. The VSS curvature exhibits an inverse linear relationship with the rolling angle ϕ, enabling reliable tracking of the robot's frame displacement using the cardioid equations. The robot's motion in flexible states is controlled through "soft" velocities $\mathbf{u}_s = [v_1, v_2]^T$, where v_j represents the velocity of the j-th locomotion unit traversing a specific cardioid path.

Table 1: Cardioids Parameters

Cardioid	r	ϕ^{min}	ϕ^{max}
1	0.021	2.42	3.87
2	0.049	2.19	4.09
3	0.043	1.73	4.56

This control approach differs fundamentally from the rigid state. By analyzing flexible modes independently and then integrating their contributions, we develop comprehensive kinematic equations governing the robot's movement.

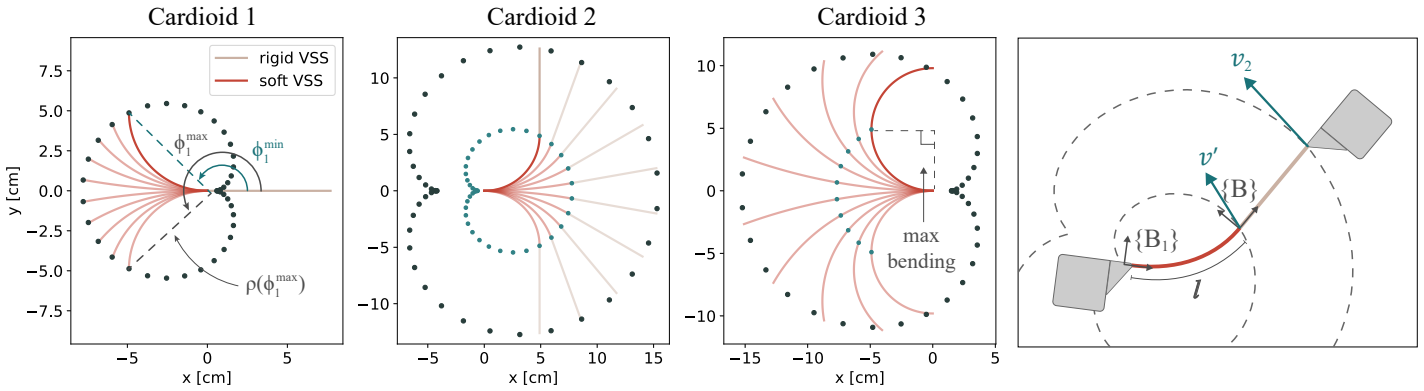

Figure 2: Kinematic representation of the soft robot deformation and locomotion unit motion approximated by three distinct cardioid trajectories.

Cardioid 1: $s_i = 1$, $s_{3-i} = 0$, $v_j \neq 0$, and $v_{3-j} = 0$; where $i, j \in [1, 2]$, $i = j$. Consequently, $\dot{x}_b = \dot{y}_b = \dot{\theta}_b = \dot{\kappa}_{3-i} = 0$.

The first flexible scenario is characterized by the motion of the j-th LU along a cardioid with velocity v_j, causing an adjacent flexible segment to bend while $\{B\}$ remains stationary. Curvature is derived through the rolling angle by mapping their corresponding ranges, $\Delta\kappa = \frac{\pi}{l}$ and $\Delta\phi_1 = \phi_1^{max} - \phi_1^{min}$:

$$\begin{cases} \kappa_i \approx -\frac{\pi}{2l} + \frac{\pi}{l\Delta\phi_1}\left(\phi_1^{max} - \phi_1\right) \\ \phi_1(\kappa_i) \approx \phi_1^{min} + \frac{l\Delta\phi_1}{\pi}\left(\frac{\pi}{2l} - \kappa_i\right). \end{cases} \tag{3}$$

The rolling angle rate is proportional to the speed of the object tracing the curve. Note that LU2 is located on the positive side of the x-axis of the frame, while LU1 is on the negative side. Therefore, a positive velocity of LU2 increases the curvature (and decreases ϕ), while a positive velocity of LU1 decreases the curvature (and increases ϕ).

Differentiating curvature in (3) with respect to ϕ_1, we get:

$$\dot{\phi}_1 = (-1)^{j-1}\frac{v_j}{\rho_1} \;\Rightarrow\; \dot{\kappa}_i = (-1)^j \underbrace{\frac{\pi}{l\Delta\phi_1\rho_1}}_{K_1(\kappa_i)} v_j. \tag{4}$$

Cardioid 2: $s_i = 1$, $s_{3-i} = 0$, $v_j \neq 0$, and $v_{3-j} = 0$; where $i, j \in [1, 2]$ and $i \neq j$; $\dot{\kappa}_{3-i} = 0$.

In this scenario, one segment remains flexible while the other is rigid, but unlike Cardioid 1, the moving locomotion unit is adjacent to the rigid segment. This configuration causes frame $\{B\}$ to follow the bending motion of the flexible segment. The locomotion unit in motion traces Cardioid 2, while the body frame traverses Cardioid 1 with its origin at the stationary LU. The frame body frame position can be determined through its location on Cardioid 1 using (2):

$$\begin{bmatrix} \mathbf{p}_b \\ 1 \end{bmatrix} = \mathbf{T}_{bi} \begin{bmatrix} (-1)^{j-1}\rho_1 \cos\phi_1 \\ \rho_1 \sin\phi_1 \\ 1 \end{bmatrix}, \tag{5}$$

where $\mathbf{p}_b = [x_b, y_b]^T$ is the body frame's position vector and $\mathbf{T}_{bi}$ represents the coordinate transformation from the stationary LU frame $\{B_i\}$ to global coordinates. The orientation of the frame $\{B_i\}$ is calculated as $\theta_{bi} = \theta_b + (-1)^i\kappa_i l$.

Figure 2 illustrates that the body frame traverses Cardioid 1 with velocity v', which differs from the LU velocity v_j along Cardioid 2. We can establish the relationship between these velocities by leveraging the fact that the curvature rate remains constant regardless of which cardioid we use for its derivation: $\dot{\kappa}_i = -\frac{\pi}{l\Delta\phi_1}\dot{\phi}_1 = -\frac{\pi}{l\Delta\phi_2}\dot{\phi}_2$. Then, using the definition of the rolling angle velocity from (4) we obtain:

$$v' = \underbrace{\frac{\Delta\phi_1}{\Delta\phi_2\rho_2}}_{\lambda_2(\kappa_i)} \rho_1 v_j. \tag{6}$$

Now, differentiating (5) with respect to ϕ_1 and substituting rolling angle's derivative with (6), we find the rate of body frame's displacement:

$$\begin{bmatrix} \dot{x}_b \\ \dot{y}_b \end{bmatrix} = \underbrace{(-1)^{j-1}\frac{\partial \mathbf{p}_b}{\partial \phi_1}\lambda_2(\kappa_i)}_{\mathbf{J}_{j2}(\kappa_i,\theta_b)} v_j. \tag{7}$$

The orientation of frame $\{B\}$ changes proportionally to the variation in segment curvature: $\Delta\theta_b = (-1)^i\Delta\kappa_i l$. Consequently, using the previously derived relationships for curvature rate in (4), we obtain:

$$\dot{\theta}_b = lK_2(\kappa_i)v_j. \tag{8}$$

Cardioid 3: $s_i = 1$, $s_{3-i} = 1$, $v_j \neq 0$, $v_{3-j} = 0$, $\dot{\kappa}_1 = \dot{\kappa}_2$.

In this scenario, both segments maintain flexibility while undergoing simultaneous deformation, resulting in equal curvature rates. The generalized coordinates and their derivatives follow the same principles established in the previous cases, with the key distinction being the synchronized bending of both segments. The frame position and orientation can be determined using the previously derived equations, accounting for the combined effect of dual segment flexibility.

3.1.3 Unified Model

Assembling equations (4)-(8), we can express the kinematics of all soft states through the following unified Jacobian

matrix:

$$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q}, \mathbf{s})\mathbf{u}$$
$$\mathbf{J}(\mathbf{q}, \mathbf{s}) = [\mathbf{J}_r, \mathbf{J}_s]^\intercal, \quad \mathbf{u} = [\mathbf{u}_r, \mathbf{u}_s]^\intercal$$
$$\mathbf{J}_s = \begin{bmatrix} s_2 & s_1 \\ s_2 & s_1 \\ s_2 & s_1 \\ s_1 & s_1 \\ s_2 & s_2 \end{bmatrix} \circ \begin{bmatrix} \mathbf{J}_{1n}(\kappa_2, \theta_b) & \mathbf{J}_{2n}(\kappa_1, \theta_b) \\ lK_n(\kappa_2) & lK_n(\kappa_1) \\ -K_m(\kappa_1) & K_n(\kappa_1) \\ -K_n(\kappa_2) & K_m(\kappa_2) \end{bmatrix} \tag{9}$$
$$(m, n) = \begin{cases} (3,3) & \text{if } s_1 \wedge s_2, \\ (1,2) & \text{otherwise.} \end{cases}$$

This kinematic model features nonholonomic constraints and coupling between the robot's pose and shape, particularly in the flexible states. A detailed analysis was performed to characterize the reachable configuration manifolds for each flexible mode. The analysis demonstrates that the fully flexible mode offers superior controllability, enabling decoupled rotation and translation and providing a connected, hole-free workspace. In contrast, the semi-flexible modes exhibit strong coupling between rotation and translation, which makes pure in-place rotation impossible and results in a reachable workspace with a significant excluded region near the origin (for more details, see Section S4, Supporting Information). These inherent kinematic constraints highlight the distinct capabilities and limitations of each operational mode, underscoring the need for a control strategy that can prioritize the appropriate mode for a given task.

3.2 Multi-Modal Motion and Shape Control

Having established the robot's operational modes through hybrid kinematics, we implement a comprehensive control strategy comprising four distinct model predictive controllers (MPC) corresponding to each possible stiffness configuration $\mathbf{s} \in \mathbf{S}$, where $\mathbf{S} = \{[0,0], [1,0], [0,1], [1,1]\}$. Each controller follows the same framework while incorporating mode-specific kinematics. For a prediction horizon N, the MPC optimization problem is formulated as:

$$\begin{aligned} \text{minimize} \quad & \sum_{k=0}^{N} \tilde{\mathbf{q}}_k^\intercal \mathbf{Q} \tilde{\mathbf{q}}_k + \sum_{k=0}^{N} \mathbf{u}_k^\intercal \mathbf{R} \mathbf{u}_k \\ \text{subject to} \quad & \tilde{\mathbf{q}}_k = \mathbf{q}_k - \mathbf{q}_{ref} \\ & \mathbf{q}_{k+1} = \mathbf{q}_k + \mathbf{J}(\mathbf{q}_k, \mathbf{s}_k)\mathbf{u}_k \\ & |w_i| \leq w_{max}, \ i = 1...4, \end{aligned} \tag{10}$$

where $\mathbf{q}_k$ is the robot's state vector, $\mathbf{q}_{ref}$ is the robot's reference state, $\mathbf{u}_k$ is the control velocity vector, $\mathbf{Q}$ and $\mathbf{R}$ are positive definite weighting matrices, $\mathbf{s}_k$ is the stiffness configuration, $\mathbf{J}(\mathbf{q}_k, \mathbf{s}_k)$ is the Jacobian defined in (9), and w_i represents individual wheel velocities bounded by w_{max}.

To seamlessly integrate continuous motion control with discrete stiffness transitions, we implement a hierarchical Motion & Morphology (M&M) Controller. This supervisory framework operates through two key mechanisms: (1) stiffness state management, which triggers transitions when shape error crosses a threshold $\Delta\kappa_{min}$, and (2) motion coordination, which activates the appropriate mode-specific MPC controller. This approach optimizes system performance by maintaining segment rigidity by default and initiating transitions only when necessary for shape control.

4 Voronoi-assisted Motion and Morphology Planning

The M&M Controller provides the low-level capability to execute specific configurations and motions by managing the robot's hybrid kinematics. However, to operate autonomously in complex environments, the robot requires a high-level planner that can reason about its goals and the surrounding space. The integration of shape adaptation significantly expands the configuration space to include five generalized coordinates, two stiffness states, and four possible motion modes. This expansion introduces the challenge of planning not only motion trajectories but also coordinated shape deformations and stiffness transitions, resulting in a high-dimensional planning problem that must be solved to navigate cluttered environments and interact with objects.

The ultimate planning goal is to enable the robot to traverse through cluttered environments and perform complex interaction tasks, such as:

- **Full-Body Grasping**: Unlike traditional mobile robots that rely on specialized end-effectors, the 2SR robot can use its entire deformable body to envelop and secure objects. This approach is particularly effective for irregularly shaped items. A secure grasp is achieved by executing a sequence of robot configurations that effectively envelop the object. Experimental validation was conducted on various objects with a detailed analysis of each grasping phase (for more information, see Section S5, Supporting Information).
- **Mobile Manipulation**: Once an object is securely grasped, the robot can solidify in its new shape and perform mobile manipulation which includes pushing the object along the path (for more information, see Section S6, Supporting Information).

To address the complexity of motion planning, we adopted a systematic decomposition approach that breaks down the problem into manageable components:

1. **Environment Analysis**: Voronoi diagrams are employed to identify potentially traversable narrow passages.
2. **Critical Point Path Planning**: Using the Voronoi structure, we determine feasible trajectories for three key points on the robot (front, middle, and rear) that define its fundamental shape during navigation.
3. **Configuration Fitting**: The method maps robot configurations to the identified point sequences, ensuring planned shapes respect the robot's kinematic constraints.
4. **Traverse Configuration Selection**: Finally, we identify a minimal configuration set that fully defines the traversal sequence while minimizing potential stiffness transitions.

4.0.1 Voronoi Analysis

The first step involves analyzing the environment to identify potentially traversable narrow passages. We employ Voronoi diagrams, a fundamental geometric structure that partitions space based on proximity to obstacle boundaries. For our application, we define the Voronoi diagram $\mathcal{V}(\mathcal{O})$ for a set of obstacles $\mathcal{O}$.

Each obstacle in the workspace is defined as a polygon with a set of vertices $\mathcal{O}_i = \{\mathbf{p}_{i,1}, \mathbf{p}_{i,2}, ..., \mathbf{p}_{i,n_i}\}$, where $\mathbf{p}_{i,j} \in \mathbb{R}^2$ denotes the j-th vertex of the i-th obstacle, and n_i is the number of vertices in obstacle i. The complete workspace obstacle set is denoted as $\mathcal{O} = \{\mathcal{O}_1, \mathcal{O}_2, ..., \mathcal{O}_m\}$, where m is the total number of obstacles.

For an obstacle $\mathcal{O}_i \in \mathcal{O}$, its Voronoi cell $\mathcal{V}(\mathcal{O}_i)$ is the set of all points in the workspace that are closer to $\mathcal{O}_i$ than to any other obstacle:

$$\mathcal{V}(\mathcal{O}_i) = \{\mathbf{v} \in \mathbb{R}^2 \mid d(\mathbf{v}, \mathcal{O}_i) \leq d(\mathbf{v}, \mathcal{O}_j), \forall j \neq i\}, \tag{11}$$

where $d(\mathbf{v}, \mathcal{O}_i)$ denotes the minimum Euclidean distance between point $\mathbf{v}$ and obstacle $\mathcal{O}_i$. A Voronoi diagram $\mathcal{V}(\mathcal{O})$ is the union of all cells $(\mathcal{V}(\mathcal{O}_i))_{\mathcal{O}_i \in \mathcal{O}}$.

The Voronoi diagram is used to build a weighted graph $\mathcal{G} = (\mathcal{V}, \mathbf{E})$ where edges $\mathbf{E}$ connect adjacent vertices in $\mathcal{V}$. Each vertex $\mathbf{v} \in \mathcal{V}$ has a clearance $c(\mathbf{v}) = \min_{\mathcal{O}_i \in \mathcal{O}} d(\mathbf{v}, \mathcal{O}_i)$ and an edge $\mathbf{e} \in \mathbf{E}$ with endpoints $(\mathbf{v}_1, \mathbf{v}_2)$ has length $l(\mathbf{e}) = ||\mathbf{v}_2 - \mathbf{v}_1||_2$ and clearance $c(\mathbf{e}) = \min(c(\mathbf{v}_1), c(\mathbf{v}_2))$.

To ensure practical traversability, multiple checks are performed along each edge by sampling a passage $\mathcal{S}_\mathbf{e}$ with k points: $\mathbf{p}_i = \mathbf{v}_1 + \frac{i}{k}(\mathbf{v}_2 - \mathbf{v}_1), \quad i \in \{0, 1, ..., k-1\}$ and verifying clearance at each point $\mathbf{p}_i$. We want to make sure that locomotion units can go through a passage. Therefore, a minimum clearance threshold is calculated as $\tau = a + \delta$, where a is the length of the LU side and δ is the safety margin.

Lemma 1 *A passage $\mathcal{S}_\mathbf{e}$ between vertices $\mathbf{v}_1$ and $\mathbf{v}_2$ is traversable if $\forall \mathbf{p}_i \in \mathcal{S}_\mathbf{e}$: $\mathbf{p}_i \cap \mathcal{O} = \varnothing$, $\min_{\mathbf{p}_i \in \mathcal{S}} c(\mathbf{p}_i) \geq \tau$, and there exists at least one continuous connection between $\mathcal{S}_\mathbf{e}$ and another traversable passage.*

Based on the set of traversable passages identified according to Lemma 1, we construct a subgraph $\hat{\mathcal{G}} = (\hat{\mathcal{V}}, \hat{\mathbf{E}}) \subseteq \mathcal{G}$ where $\hat{\mathbf{E}} = \{\mathbf{e} \in \mathbf{E} \mid \mathcal{S}_\mathbf{e} \text{ is traversable}\}$ contains only edges corresponding to traversable passages and $\hat{\mathcal{V}}$ consists of their associated vertices. This reduced graph $\hat{\mathcal{G}}$ forms the basis for subsequent motion planning and analysis.

4.0.2 Control Points Path Planning

Given the complex geometry of a flexible 2SR robot, we can simplify the path planning problem by focusing on three key characteristic points along the robot's length, see Fig. 3: the *front* point $\boldsymbol{\xi}_f$ serving as the leading point during passage traversal (head or tail LU), the *middle* point $\boldsymbol{\xi}_m$ acting as the central reference point of the robot, and the *rear* point $\boldsymbol{\xi}_r$ functioning as the trailing point during traversal. These points are positioned along the robot's centerline (in a default configuration) such that $||\boldsymbol{\xi}_f - \boldsymbol{\xi}_m|| = ||\boldsymbol{\xi}_r - \boldsymbol{\xi}_m|| = \frac{L}{2}$, where L is the robot's length. This minimal set of points is sufficient to verify feasibility of the robot's placement within narrow passages and serve as control points for subsequent configuration reconstruction.

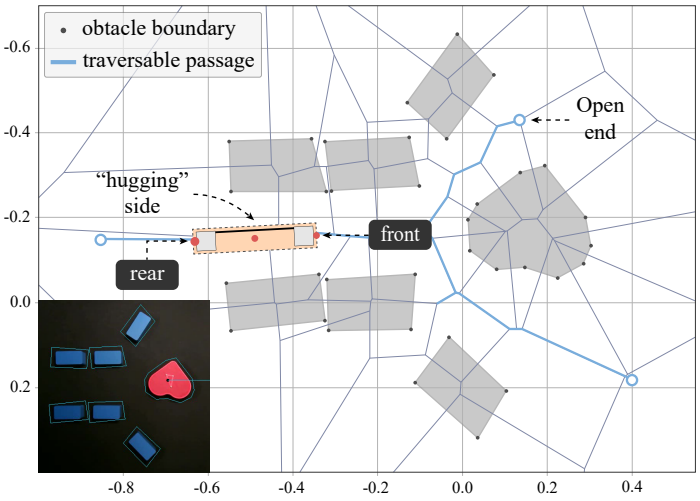

Figure 3: Voronoi diagram of the highly cluttered environment with detected obstacles. The location of critical points on the robot is highlighted in red.

The selection of the front point depends on the environment. In most cases, either locomotion unit can be designated as the front. However, when the manipulandum is asymmetrically obstructed, the choice depends on the accessibility relative to the body frame in the target (grasp) configuration that is determined as: $\mathbf{q}' = [\mathbf{x}'_r, 0, 0]^\intercal$. The head locomotion unit is designated as front when the region corresponding to the negative x-axis of the body frame offers greater accessibility, while the tail LU is selected when the positive x-axis region is more accessible. This selection ensures that the robot's "hugging" front side will be properly oriented towards the manipulandum in the grasp configuration.

For a given target configuration $\mathbf{q}'$, we first identify the global coordinates of the target control points:

$$\boldsymbol{\Xi}' = \{\boldsymbol{\xi}_f(\mathbf{q}'), \boldsymbol{\xi}_m(\mathbf{q}'), \boldsymbol{\xi}_r(\mathbf{q}')\}. \tag{12}$$

Each control point must be mapped to its nearest edge in the traversable graph $\hat{\mathcal{G}}$. Let $d(\boldsymbol{\xi}, \mathbf{e})$ denote the distance from point $\boldsymbol{\xi}$ to edge $\mathbf{e}$. Then, the set of nearest edges for all target control points is defined as:

$$\mathbf{E}' = \{\mathbf{e}_i \mid \mathbf{e}_i = \arg\min_{\mathbf{e} \in \hat{\mathbf{E}}} d(\boldsymbol{\xi}_i, \mathbf{e}), \boldsymbol{\xi}_i \in \boldsymbol{\Xi}'\}. \tag{13}$$

The robot's entry to the passage network is determined by finding the open-ended vertex closest to its initial position:

$$\mathbf{v}_0 = \arg\min_{\mathbf{v} \in \mathcal{V}'_{open}} ||\mathbf{v} - \mathbf{q}_0||, \tag{14}$$

where $\mathcal{V}'_{open} \subseteq \mathcal{V}'$ is the set of vertices with one incident edge.

Let $\mathbf{V}(\mathbf{E}')$ denote the set of vertices incident to edges in $\mathbf{E}'$. For each vertex $\mathbf{v} \in \mathbf{V}(\mathbf{E}')$, we define $\mathcal{P}_\mathbf{v}$ as the set of all possible paths from $\mathbf{v}_0$ to $\mathbf{v}$, where each path $\mathcal{P} \in \mathcal{P}_\mathbf{v}$ is a sequence of edges in $\hat{\mathbf{E}}$. The optimal paths to all target vertices are determined using Dijkstra's algorithm:

$$\mathcal{P}^* = \{\mathcal{P}^*_i \mid \mathcal{P}^*_i = \arg\min_{\mathcal{P} \in \mathcal{P}_{\mathbf{v}_i}} \sum_{\mathbf{e} \in \mathbf{P}} l(\mathbf{e}), \mathbf{v}_i \in \mathbf{V}(\mathbf{E}')\}. \tag{15}$$

For each edge $\mathbf{e} \in \mathbf{E}'$ we include only one endpoint ($\mathbf{v}_1$ or $\mathbf{v}_2$) to $\mathbf{V}(\mathbf{E}')$ based on which vertex yields the shorter path from $\mathbf{v}_0$. After determining the optimal path $\mathcal{P}_i^*$, we compute the projection point $\bar{\boldsymbol{\xi}}_i$ on the corresponding edge $\mathbf{e}_i$ that is closest to the control point $\boldsymbol{\xi}_i$:

$$\bar{\boldsymbol{\xi}}_i = \arg\min_{\mathbf{p} \in \mathcal{S}_{\mathbf{e}_i}} ||\mathbf{p} - \boldsymbol{\xi}_i||_2, \ \boldsymbol{\xi}_i \in \boldsymbol{\Xi}', \mathbf{e}_i \in \mathbf{E}'. \tag{16}$$

The final path is then augmented by adding an edge connecting vertex $\mathbf{v}$ to the projection point $\bar{\boldsymbol{\xi}}_i$.

4.0.3 Path Coordination under Physical Constraints

While individual paths for control points provide general guidance through passages, maintaining the physical constraints of the robot's structure is crucial. The robot consists of three critical points connected by VS segments of fixed length. Throughout the motion, these distances must remain constant to prevent unrealistic stretching or compression of the segments.

We use the rear point's path $\mathcal{P}_r^*$ as the primary reference, as it represents the "pushing" end of the robot through the passage sequence. The front and middle points' paths are then adjusted to satisfy physical constraints. Let $\mathcal{S}_r = (\mathcal{S}_\mathbf{e})_{\mathbf{e} \in \mathcal{P}_r^*}$ be a sequence of equally spaced sample points along the edges in $\mathcal{P}_r^*$. For each point $\boldsymbol{\xi}_r(k) \in \mathcal{S}_r$, we determine the corresponding locations of the middle and front points $\boldsymbol{\xi}_m(k)$ and $\boldsymbol{\xi}_f(k)$ subject to the following constraints:

$$\begin{cases} d\hat{\mathcal{G}}(\boldsymbol{\xi}_f(k), \boldsymbol{\xi}_r(k)) = l, \\ d\hat{\mathcal{G}}(\boldsymbol{\xi}_m(k), \boldsymbol{\xi}_f(k)) = d\hat{\mathcal{G}}(\boldsymbol{\xi}_m(k), \boldsymbol{\xi}_r(k)) = \frac{l}{2}, \end{cases} \tag{17}$$

where $d\hat{\mathcal{G}}(\cdot, \cdot)$ denotes the geodesic distance along edges in $\hat{\mathcal{G}}$.

The adjustment of front and middle points' paths is dynamically governed by the rear point's progression through the passages. At branching points where multiple edges are available, we first prioritize edges present in the guidance paths $\mathcal{P}_m^*$ and $\mathcal{P}_f^*$. When points are pushed beyond their guidance paths, the next edge connected to their last visited edge is selected. In cases where multiple options exist, we select edges incident to vertices with higher degree in $\hat{\mathcal{G}}$.

4.0.4 Paths Smoothing and Optimization

The final step in determining paths for the robot control points is refinement. Path planning through Voronoi analysis, while ensuring feasibility, often results in trajectories containing unnecessary fluctuations and sharp turns. These artifacts, inherent to Voronoi diagrams, can lead to jerky motions and excessive shape deformations of a 2SR robot. A smoothing step is therefore crucial to refine the raw paths while preserving their essential navigational properties and maintaining clearance constraints.

The smoothing process begins with geometric analysis of the path structure, treating it as a collection of straight sections. Each j-th section in $\mathcal{S}_i$, $i \in \mathbb{S}$, $\mathbb{S} = \{r, m, f\}$ is characterized by its direction vector $\vec{\mathbf{d}}_{i,j}$ and length $l_{i,j}$. The smoothing strategy employs a two-phase approach.

The first phase focuses on section merging, targeting nearly parallel adjacent path sections. For two consecutive

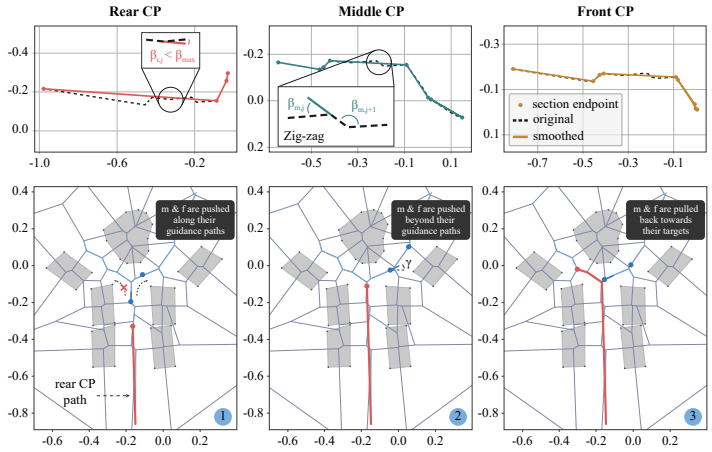

Figure 4: Optimized paths for 2SR robot's control points (CP). **Top**: Comparison of paths before and after smoothing optimization. **Bottom**: Evolution of the robot's motion through the workspace, showing the Voronoi diagram and the coordinated movement of the three control points.

sections, we compute their directional similarity:

$$\cos \beta_{i,j} = \frac{\vec{\mathbf{d}}_{i,j} \cdot \vec{\mathbf{d}}_{i,j+1}}{||\vec{\mathbf{d}}_{i,j}|| \cdot ||\vec{\mathbf{d}}_{i,j+1}||}. \tag{18}$$

When $\beta_{i,j} < \beta_{max}$, these sections are merged through cubic spline interpolation, effectively eliminating small oscillations while maintaining the path's overall structure.

The second phase addresses zigzag patterns that typically emerge when the path navigates between multiple obstacles. For any three consecutive sections, we analyze the total direction change:

$$\Delta\beta_{i,j} = |\beta_{i,j} - \beta_{i,j+1}|. \tag{19}$$

If $\Delta\beta_{i,j} < \Delta\beta_{max}$, we replace the zigzag pattern with a single straight section while preserving endpoint positions and ensuring clearance constraints are maintained.

Finally, we perform uniform resampling to generate equally spaced points along the smoothed paths $\mathcal{S}_r^*$, $\mathcal{S}_m^*$, and $\mathcal{S}_f^*$, ensuring they contain the same number of points. Implementation of this path optimization is illustrated in Fig. 4.

4.0.5 Fitting Configurations

After obtaining smoothed paths for the critical points, we need to determine the complete robot configurations that satisfy these paths while maintaining the kinematic constraints of a 2SR robot.

For each n-th point along the path, we seek to find the optimal robot configuration $\mathbf{q}_n$ that best matches the desired positions of control points while ensuring smooth transitions between consecutive states.

The optimization problem is formulated as:

$$\begin{aligned} \underset{\mathbf{q}_n \in \mathbb{R}^5}{\text{minimize}} \quad & \sum_{i \in \mathbb{S}} w_i ||\boldsymbol{\xi}_i(\mathbf{q}_n) - \mathcal{S}_i^*(n)||_2 + \tilde{\mathbf{q}}_n^\intercal \mathbf{W} \tilde{\mathbf{q}}_n + \\ & + \lambda_\theta |\theta_{b,n} - \gamma_n| + \lambda_\kappa (|\kappa_{1,n}| + |\kappa_{2,n}|) \\ \text{subject to} \quad & \tilde{\mathbf{q}}_n = \mathbf{q}_n - \mathbf{q}_{n-1} \\ & |\kappa_{j,n}| \leq \frac{\pi}{2l}, \ j \in \{1, 2\}, \end{aligned} \tag{20}$$

where $\mathbb{S} = \{f, m, r\}$ is the set of critical points and $\boldsymbol{\xi}_i(\mathbf{q}_n)$ represents the mapping from configuration space to the global position of the i-th control point.

The objective function balances multiple requirements through its components. The first term ensures accurate positioning of critical points, with weights w_i prioritizing the middle point's placement. The smoothness term penalizes large changes between consecutive configurations through a weighted quadratic form with $\mathbf{W}$ being a positive definite weighting matrix. The third term aligns the robot's orientation with the desired direction γ_n, computed as the tangent at $\mathcal{S}_m^*(n)$ pointing towards the tail LU. The final term minimizes absolute curvatures to reduce unnecessary bending

This constrained optimization problem can be solved sequentially along the path using Sequential Least Squares Programming (SLSQP), with each solution serving as the initial guess for the next point.

4.0.6 Discretizing Crucial Configurations

After obtaining a configurations path, we must address a fundamental limitation. A physical 2SR robot cannot execute continuous shape changes with arbitrary precision in a flexible state, especially when target configurations are significantly distant from each other. Substantial planar displacements require omnidirectional motion of the robot in a rigid state.

Instead of attempting to find a continuous motion with shape deformation trajectory, we identify a discrete set of key configurations that capture the essential characteristics of the planned motion. This set can then be sequentially supplied to the M&M controller for robot execution. This discretization process is critical for practical implementation while maintaining the path's navigational intent.

The selection of key configurations follows a mathematical framework based on three criteria. First, we perform curvature extrema analysis. Let $\kappa_j(n)$ be a curvature function for the j-th VS segment along the robot configuration path with a path length parameter n. We identify local extrema where:

$$\frac{d\kappa_j}{dn} = 0 \text{ and } \left|\frac{d^2\kappa_j}{dn^2}\right| > 0, \quad j \in 1, 2. \tag{21}$$

Specifically, we detect peaks and valleys where $|\kappa_j| > \eta$.

Second, we perform flat region detection by identifying regions of approximately constant curvature where:

$$\frac{1}{\Delta n}\sum_{n=n_1}^{n_2}\left|\frac{d\kappa_j}{dn}\right| < \epsilon. \tag{22}$$

These regions represent constant shape configurations that the robot should maintain during motion. Third, we include configurations at spatial extrema: $\arg\min_n \mathbf{p}(n)$ and $\arg\max_n \mathbf{p}(n)$, where $\mathbf{p} = [x_b, y_b]^\intercal$ is a robot position vector.

After the initial set of crucial configurations $\mathcal{Q}$ is found, it is refined through a proximity filter to prevent redundancy:

$$\mathcal{Q}^* = \{\mathbf{q}_i \in \mathcal{Q} : ||\mathbf{p}_i - \mathbf{p}_{i+1}||_2 > d_{\min},\ \mathbf{p}_i \in \mathbf{q}_i, \tag{23}$$

where $d_{\min}$ is a minimum distance threshold in the configuration space, ensuring that consecutive configurations are sufficiently distinct to warrant separate execution.

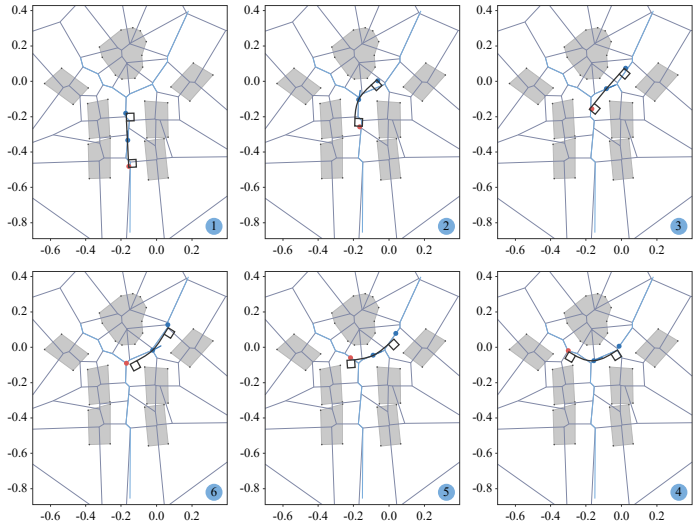

Figure 5: Evolution of the robot's crucial configurations through the workspace. Configurations are reconstructed from the robot control points.

The resulting discrete sequence of robot configurations $\mathcal{Q}^* = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ provides a set of waypoints that captures essential shape changes along the passage traversal path while minimizing unnecessary deformations, see Fig. 5. We denote this comprehensive framework for integrated motion and morphological planning as *VMMP (Voronoi-assisted Motion and Morphology Planning)*.

5 Results

To validate the proposed design and control methodology, we fabricated and tested a 2SR robot prototype. The experimental platform consisted of a $2.78 \times 1.4 \times 1.4$m metal frame workspace equipped with six OptiTrack cameras for high-precision motion capture and a ceiling-mounted RGB

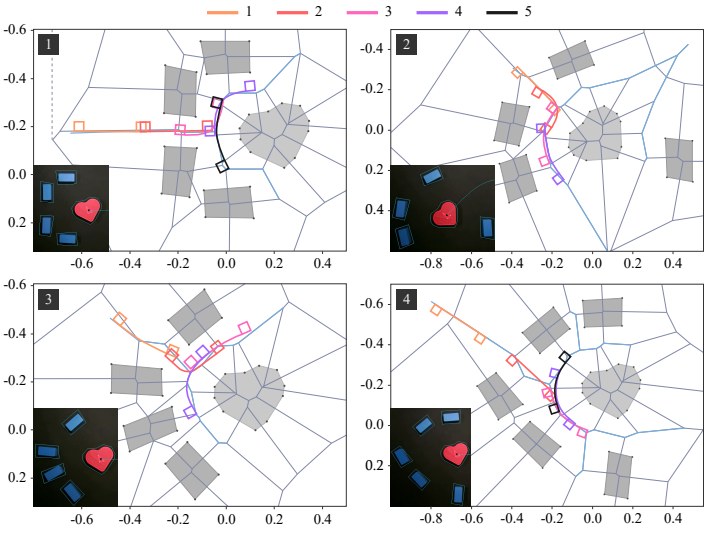

Figure 6: Configuration sequences constructed by the Voronoi-assisted Motion and Morphology Planner (VMMP) for four different environment scenarios.

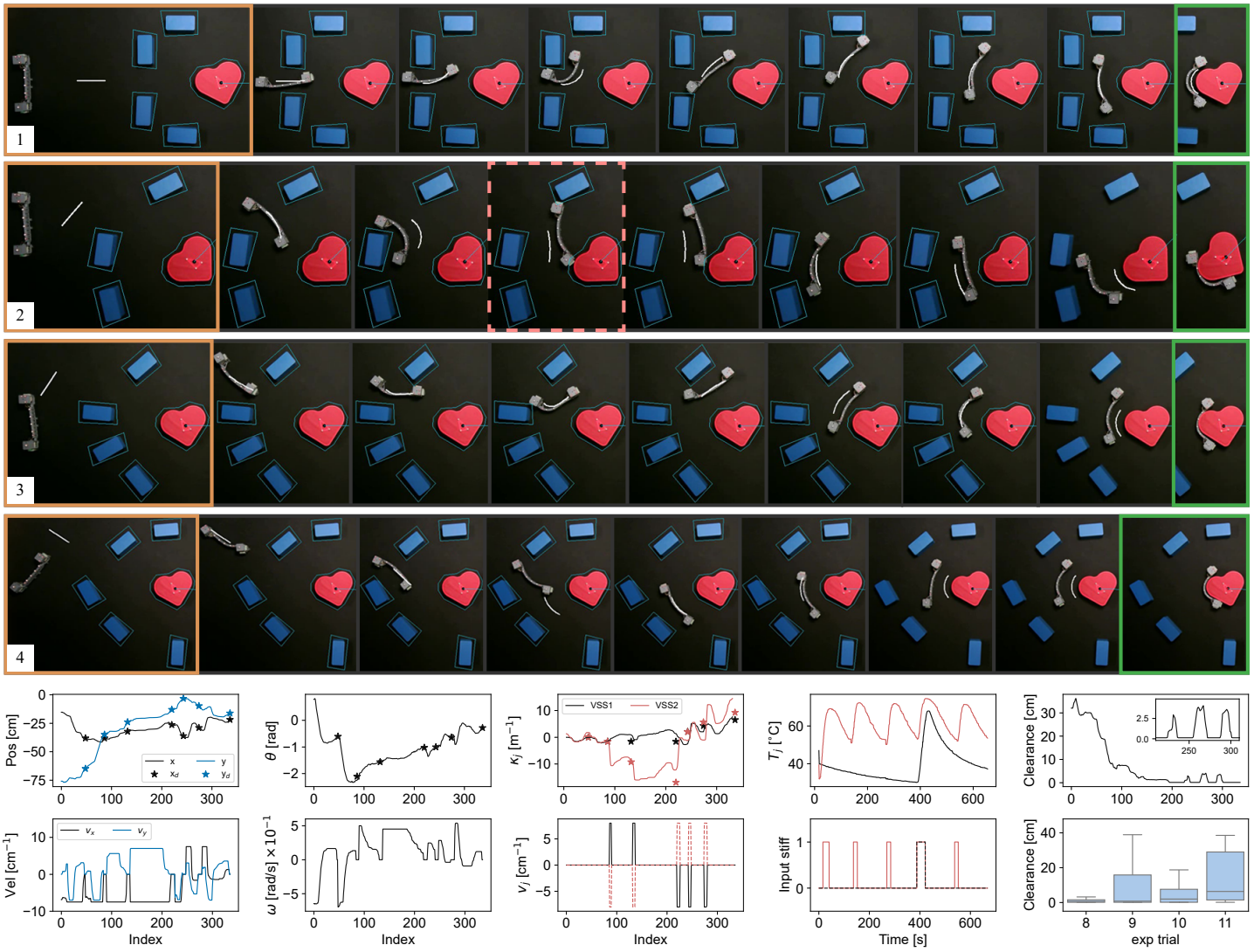

Figure 7: **Top**: Time-lapse sequence showing the 2SR robot navigating through a cluttered environment using the VMMP motion and morphology planning method in four different scenarios. **Bottom**: System response data during experiments, including: robot configuration evolution, VSS temperature feedback, velocity and stiffness control inputs and clearance measurements over time for trial "Heart 3" as well as clearance variations across all experiment trials.

camera for visual monitoring. To ensure stable and repeatable locomotion, the arena's floor was covered with a non-slip silicone mat, which guarantees reliable traction. A heart-shaped object with three low-friction support wheels was used as a manipulandum. The conducted experiment, consisting of four trials, assessed the robot's ability to execute a complete operational sequence, which comprised three consecutive tasks:

(A) Traversing through the obstacles

(B) Full-body grasping

(C) Transporting the grasped object along the path.

5.1 Traversing through the obstacles

Fig. 6 illustrates the robot configuration sequences generated by the VMMP method across four distinct scenarios. The process begins with converting the workspace into a Voronoi diagram, which enables the identification of traversable passages (highlighted in blue). For each scenario, VMMP generates optimal paths $\mathcal{Q}^*$ comprising four to five key configurations, starting from the robot's default shape. As the robot encounters constricted areas, the method determines the minimum deformation needed for the robot to fit through the narrow spaces while maintaining adequate clearance. The resulting configuration sequences demonstrate smooth morphological transitions while ensuring collision-free navigation through the constrained environment.

Fig. 7 illustrates the robot's navigation through obstacles, demonstrating its dynamic stiffness switching, motion mode changes, and adaptive deformation. The execution process involves interpolating between key configurations in the VMMP-generated optimal path $\mathcal{Q}^*$ to ensure smooth motion, with each waypoint processed by the Motion and Morphology Controller. The quantitative plots in Fig. 7 show robot state responses and velocity control inputs for trial "Heart 3".

In this scenario, the second segment underwent the most significant deformation, requiring ten phase transitions (five rigid-to-soft and five soft-to-rigid), while the first segment

switched stiffness only twice with a single deformation. Trial "Heart 4" presented the simplest case, requiring only one deformation in each segment. Other trials required 3-4 deformations (6-8 phase transitions) per segment. Completion times varied significantly: trial "Heart 3" took the longest at nearly 11 minutes, while trial "Heart 4" was the fastest at 2.5 minutes. Trials "Heart 1" and "Heart 2" completed in 9.45 and 8.87 minutes respectively.

When the robot navigates and deforms in the confined space between obstacles and the manipulandum, its locomotion units frequently approach the edges of obstacle buffer zones. This behavior is evident in the clearance plot in Fig. 7, where values fluctuate considerably, often coming close to zero during deformation phases near obstacles. The 2SR robot generally managed these challenging zones effectively, with only a few buffer zone violations like during trial "Heart 2" (red frame in the execution sequence in Fig. 7). Importantly, even in this instance, no physical collisions occurred, and neither the obstacles nor the manipulandum were disturbed despite the safety boundary violation.

The box plot in Fig. 7 provides a summary of clearance measurements across all trials, reflecting the environmental complexity of each scenario. Trial "Heart 1," featuring the most congested workspace, exhibited the lowest average clearance, while trial "Heart 4," with its simpler configuration, maintained the highest clearance values.

5.2 Full-body grasping

The grasping process begins with the robot at the VMMP target, which defines the approach configuration for the full-body grasping method. Figure 8 illustrates this process with a time-lapse sequence of the 2SR robot executing the final two steps: deformation into a pre-grasp configuration, followed by final positioning for contact. The quantitative plots for the "Heart 3" trial provide detailed insights into the robot's states and control inputs throughout this sequence.

First, the robot initiates a 13.5-second transition to a flexible state in its second segment. Once flexible, the robot uses "flexible" control velocities to adjust its segment curvature, conforming to the target shape required for grasping. After achieving the desired shape, the robot undergoes a 47.7-second transition back to a rigid state. This rigidity allows for final adjustments to its overall pose to make contact with the object. During these rigid-flexible phase transitions, the robot remains stationary — resulting in constant state values and zero control inputs — while the temperature of the second variable stiffness segment (VSS2) increases or decreases. Notably, in the rigid phase, minor fluctuations in the bridge curvatures are observed, which are attributed to the inherent, residual flexibility of the robot's mechanical structure.

5.3 Object Transportation

Fig. 9 illustrates the performance of a robotic transportation task. Part (a) shows time-lapsed sequences of the robot moving an object along a pre-defined reference path generated with a Bézier curve. Part (b) presents the system's velocity response during the "Heart 2" trial, plotting the heading (tangential) and lateral (normal) velocities. The

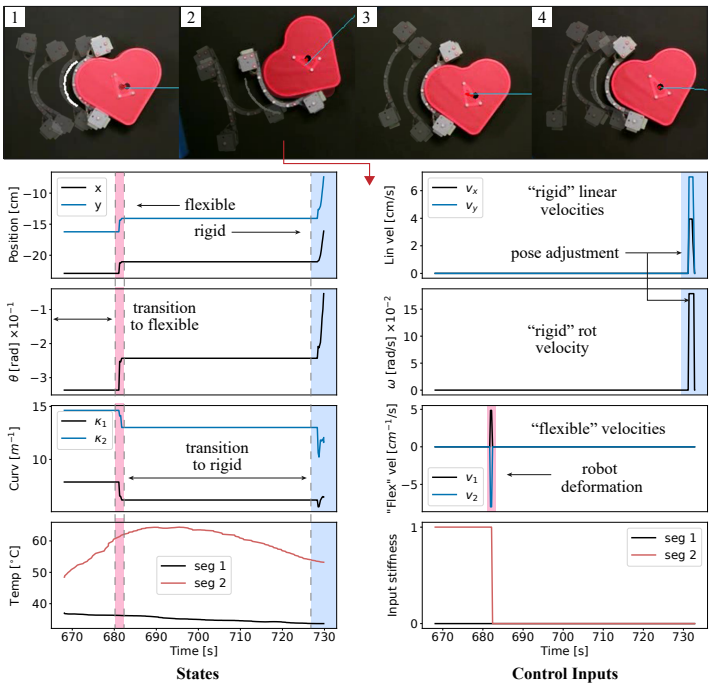

Figure 8: **Top**: Time-lapsed sequences of the 2SR robot executing a three-phase grasping method on a heart-shaped object during the fixed-morphology experiment. **Bottom**: System responses during trial "Heart 3" showing robot states (pose, curvatures and temperature feedback) and control inputs (target velocities and stiffness transition commands) throughout the grasping process.

manipulandum maintains a smooth heading velocity of $6\frac{cm}{s}$. The robot's heading velocity closely tracks this target, exhibiting corrective oscillations around the reference profile. The lateral velocity profiles demonstrate a symmetrical pattern with a minor offset between the robot and the object.

Fig. 9(c) compares the spatial trajectories of the robot and the object against the reference path. The plot shows that the object adheres closely to the intended trajectory, confirming the success of the transportation task. Fig. 9(d) plots the orientation of both the robot and the object over time. Their orientations remain closely aligned throughout the task, which indicates that the robot maintains a stable grasp during transport.

The quality of the grasp has a significant influence on subsequent manipulation performance. In some instances, robot locomotion units may inadvertently disturb the object during grasping, causing misalignment with the reference path and inducing oscillatory motion. These issues often stem from imperfect orientation or position overshooting at the grasp configuration. The effect of these initial disturbances is evident in Figure 9(e), which compares path tracking errors across different trials. The second trial exhibits the most significant oscillations at the start of the path, likely due to such a disturbance. Nevertheless, the data confirms that in all cases, the robot effectively stabilizes, damping the oscillations and converging onto the intended trajectory.

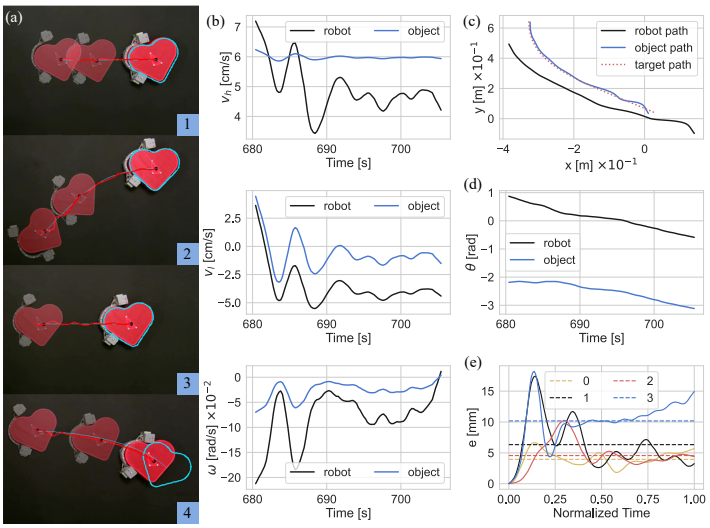

Figure 9: (a) Time-lapsed sequences of the 2SR robot transporting the object along the path. Second trial: (b) Control velocity inputs (heading, lateral, and rotational) for both the robot and the object; (c) Spatial trajectories of robot and object centroids, demonstrating path-following performance; (d) Comparison of robot-object orientation dynamics. (e) Time evolution of object tracking error across all trials.

6 Conclusion

This paper introduced a comprehensive framework for autonomous mobile manipulation in cluttered environments, centered on a novel shape-shifting, variable-stiffness robot. Our approach integrates an improved 2SR robot design featuring a modular, LMPA-based variable-stiffness bridge—which achieves a 2.5-fold increase in state transition speed and enhanced robustness—with a hierarchical Motion and Morphology (M&M) control architecture. This controller leverages a hybrid kinematic model to seamlessly manage discrete stiffness transitions and continuous, mode-specific motion control, enabling the execution of complex configuration sequences. These sequences are generated by our high-level planner, the Voronoi-assisted Motion and Morphology Planner (VMMP), which effectively addresses the high-dimensional planning problem by decomposing it into environment analysis, multi-point pathfinding, and kinematically constrained configuration reconstruction.

Experimental validation demonstrated the system's efficacy in executing a complete operational sequence: autonomous navigation through densely packed obstacles, full-body grasping of an irregularly shaped object, and subsequent object transportation. The results confirm that the integration of our hardware design, control strategy, and planning algorithm enables the robot to dynamically adapt its morphology and stiffness to successfully navigate and interact within challenging, constrained spaces. By unifying efficient locomotion, adaptive morphology, and autonomous planning within a single untethered platform, this work presents a significant step towards a new paradigm for mobile manipulation. It demonstrates the viability of hybrid soft-rigid systems as a practical solution to the long-standing trade-off between the adaptability of soft robots and the performance of conventional rigid systems.

In future work, we aim to explore several promising directions. First, reinforcement learning approaches will be investigated to better capture the complex non-linear behavior of the robot that is not fully represented in analytical models. Second, the robot's unique combination of conforming ability and mobility will be leveraged for active environment exploration applications. Finally, we plan to extend the system to multi-agent scenarios, studying how multiple 2SR robots can cooperatively accomplish more complex tasks. These developments will further expand the potential applications of this versatile robotic platform in challenging real-world environments.

Acknowledgements

This work was supported in part by the Research Grants Council (RGC) of Hong Kong under grant 15231023, and in part by the PolyU FENG Intra-Faculty Interdisciplinary Project under grant WZ8B.

References

[1] N. Ghodsian, K. Benfriha, A. Olabi, V. Gopinath, A. Arnou, *Sensors* **2023**, *23*, 19 8026–8026.

[2] P. Ben-Tzvi, A. A. Goldenberg, J. W. Zu, *Journal of Mechanical Design* **2008**, *130*, 7.

[3] J. Laber, R. Thamma, E. Kirby, *IJISET - International Journal of Innovative Science, Engineering & Technology* **2020**, *7*, 8 63–70.

[4] A. Gunatilake, L. Piyathilaka, A. Tran, V. K. Vishwanathan, K. Thiyagarajan, S. Kodagoda, *IEEE Sensors Journal* **2021**, *21*, 10 11926–11934.

[5] Y.-S. Chen, J.-G. Juang, In *2009 ICCAS-SICE*. **2009** 3199–3204.

[6] M. Bjelonic, N. Kottege, T. Homberger, P. Borges, P. Beckerle, M. Chli, *Journal of Field Robotics* **2018**, *35*, 7 1063–1079.

[7] A. Elfes, R. Steindl, F. Talbot, F. Kendoul, P. Sikka, T. Lowe, N. Kottege, M. Bjelonic, R. Dungavell, T. Bandyopadhyay, M. Hoerger, B. Tam, D. Rytz, In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. **2017** 1050–1057.

[8] S. Mintchev, D. Floreano, *IEEE Robotics & Automation Magazine* **2016**, *23*, 3 42–54.

[9] H. Hauser, *Nature Machine Intelligence* **2019**, *1*, 8 338–339.

[10] B. T. Wilcox, J. Joyce, M. D. Bartlett, *Advanced Intelligent Systems* **2024**, *7*, 7.

[11] A. P. Sabelhaus, J. Bruce, K. Caluwaerts, P. Manovi, R. F. Firoozi, S. Dobi, A. M. Agogino, V. SunSpiral, In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. **2015** 2867–2873.

[12] J. Petrš, R. Kobayashi, F. van Diggelen, H. Nabae, K. Suzumori, D. Floreano, *Advanced Intelligent Systems* **2025**, *7*, 9.

[13] M. Polzin, Q. Guan, J. Hughes, *Science Robotics* **2025**, *10*, 99.

[14] S. Li, R. Batra, D. Brown, H.-D. Chang, N. Ranganathan, C. Hoberman, D. Rus, H. Lipson, *Nature* **2019**, *567*, 7748 361.

[15] Y. Xi, T. J. Jones, R. Huang, T. Marzin, P.-T. Brun, *arXiv preprint arXiv:2404.10614* **2024**.

[16] Z. Wang, V. Kumar, In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, volume 1. **2002** 394–399 vol.1.

[17] F. Bertoncelli, M. Selvaggio, F. Ruggiero, L. Sabattini, In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. **2022** 1639–1646.

[18] F. Bertoncelli, L. Sabattini, In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. **2024** 5203–5210.

[19] M. Kulkarni, H. Nguyen, K. Alexis, *IFAC-PapersOnLine* **2020**, *53*, 2 9295.

[20] H. Zhi, B. Zhang, J. Qi, J. G. Romero, X. Shao, C. Yang, D. Navarro-Alarcon, *IEEE Robotics and Automation Letters* **2024**.

[21] K. Son, K. Bowal, K. Kim, L. Mahadevan, H.-Y. Kim, *Science Advances* **2025**, *11*, 19 eadu8326.

[22] A. Sarker, T. U. Islam, M. R. Islam, *Advanced Intelligent Systems* **2024**, *7*, 3.

[23] O. Rabaux, C. Jérôme, *Advanced Intelligent Systems* **2024**, *7*, 2.

[24] M. Sun, C. Tian, L. Mao, X. Meng, X. Shen, B. Hao, X. Wang, H. Xie, L. Zhang, *Advanced Functional Materials* **2022**, *32*, 26 2112508.

[25] L. V. Nguyen, H. Kim, K. T. Nguyen, F. Alambeigi, V. A. Ho, *Science Advances* **2025**, *11*, 18 eads3006.

[26] T. Wang, J. Zhang, G. Zhao, Y. Li, J. Hong, M. Y. Wang, In *2018 IEEE/ASME international conference on advanced intelligent mechatronics (AIM)*. IEEE, **2018** 1087–1091.

[27] A. Yin, H.-C. Lin, J. Thelen, B. Mahner, T. Ranzani, *Advanced Intelligent Systems* **2019**, *1*, 8 1900089.

[28] P. Wharton, T. L. You, G. P. Jenkinson, R. S. Diteesawat, N. H. Le, E.-C. Hall, M. Garrad, A. T. Conn, J. Rossiter, *IEEE Robotics and Automation Letters* **2023**, *8*, 8 5007–5014.

[29] N. S. Usevitch, Z. M. Hammond, M. Schwager, A. M. Okamura, E. W. Hawkes, S. Follmer, *Science Robotics* **2020**, *5*, 40 eaaz0492.

[30] H. C. Lee, N. Elder, M. Leal, S. Stantial, E. Vergara Martinez, S. Jos, H. Cho, S. Russo, *Nature Communications* **2024**, *15*, 1.

[31] L. Paternò, G. Tortora, A. Menciassi, *Soft Robotics* **2018**, *5*, 6 783–799.

[32] H.-T. Lin, G. G. Leisk, B. Trimmer, *Bioinspiration & Biomimetics* **2011**, *6*, 2 026007.

[33] C. Liu, O. Edwards, K. Althoefer, K. Zhang, H. Godaba, In *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*. **2022** 715–721.

[34] R. J. Lock, S. C. Burgess, R. Vaidyanathan, *Bioinspiration & Biomimetics* **2013**, *9*, 1 011001.

[35] M. Kovač, Wassim-Hraiz, O. Fauria, J.-C. Zufferey, D. Floreano, In *2011 IEEE International Conference on Robotics and Biomimetics*. **2011** 1503–1508.

[36] Z. Li, X. Chu, X. Hu, Z. Zhang, N. Li, J. Li, *Smart Materials and Structures* **2023**, *33*, 6 063002–063002.

[37] J.-S. Koh, S.-M. An, K.-J. Cho, In *2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*. **2010** 610–615.

[38] B. Kim, M. G. Lee, Y. P. Lee, Y. Kim, G. Lee, *Sensors and Actuators A: Physical* **2006**, *125*, 2 429–437.

[39] H. M. Jaeger, *Soft Matter* **2015**, *11*, 1 12–27.

[40] E. Steltz, A. M. Mozeika, J. Rembisz, N. Corson, H. M. Jaeger, *Proceedings of SPIE* **2010**.

[41] M. A. Karimi, V. Alizadehyazdi, H. M. Jaeger, M. Spenko, *IEEE Transactions on Robotics* **2021**, *38*, 2 1–12.

[42] B. E. Schubert, D. Floreano, *RSC Advances* **2013**, *3*, 46 24671.

[43] A. Lipchitz, G. Harvel, T. Sunagawa, In *Proceedings of ICONE-23/23rd International Conference on Nuclear Engineering*. **2015** .

[44] C. Liu, H. Qin, P. T. Mather, *Journal of Materials Chemistry* **2007**, *17*, 16 1543–1558.

[45] L. Labazanova, S. Peng, L. Qiu, H.-Y. Lee, T. Nanayakkara, D. Navarro-Alarcon, *IEEE Robotics and Automation Letters* **2023**, *8*, 3 1643–1650.

[46] A. Tonazzini, S. Mintchev, B. Schubert, B. Mazzolai, J. Shintake, D. Floreano, *Advanced Materials* **2016**, *28*, 46 10142–10148.