

Tumbling Locomotion of Tetrahedral Soft-limbed Robots

Dimuthu D. K. Arachchige¹, Dulanjana M. Perera², Umer Huzaifa¹, Iyad Kanj¹, and Isuru S. Godage³

Abstract—Soft robots, known for their compliance and deformable nature, have emerged as a transformative field, giving rise to various prototypes and locomotion capabilities. Despite continued research efforts that have shown significant promise, the quest for energy-efficient mobility in soft-limbed robots remains relatively elusive. We introduce a discrete locomotion gait called "tumbling," designed to conserve energy and implemented in a topologically symmetric soft-limbed robot. The incorporation of tumbling enhances the overall locomotive abilities of soft-limbed robots, offering advantages such as increased agility, adaptability, and the ability to correct orientation, which are essential for navigating non-engineered environments that include natural-like irregular terrains with obstacles. The principle behind tumbling locomotion involves a deliberate shift in the robot's center of gravity in the direction of motion, guided by the kinematics of its soft limbs. To validate this locomotion strategy, we developed a robot simulation model operating within a virtual environment that incorporates physics and contact interactions. After optimizing the tumbling locomotion strategy through simulations, we conducted experimental tests on a physical robot prototype. The experiments validate the effectiveness of the proposed tumbling gait. We conducted an energy cost analysis to compare the tumbling locomotion with the previously reported crawling gait of the robot. The results of this analysis demonstrate that tumbling represents an energy-efficient mode of locomotion for soft robots, saving up to 60% and 65% energy than crawling locomotion on flat and natural-like irregular terrains, respectively.

Index Terms—kinematics, locomotion, soft-limbed robots, tumbling.

I. INTRODUCTION

SOFT mobile robots present a significant departure from traditional rigid-bodied robots in the field of robotics. Unlike their rigid counterparts, soft robots are characterized by their flexible, pliable, and compliant structures, enabling them to navigate through complex environments with greater

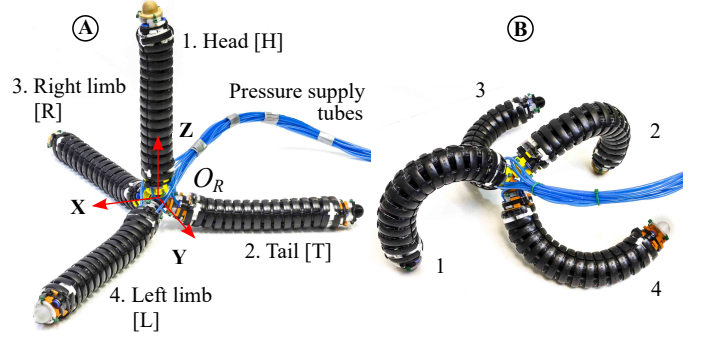


Fig. 1. Tetrahedral robot at (A) unactuated and (B) tumbling poses.

dexterity and adaptability. Consequently, they hold the promise of transforming various robotic applications, ranging from search and rescue missions in disaster-stricken areas to farming practices in agriculture.

Numerous soft mobile robot prototypes and locomotion modes have been proposed to date [1]. However, the pursuit of energy-efficient locomotion in soft-limbed robots remains relatively uncharted territory. Efforts to achieve efficient movement in soft-limbed robots necessitate the exploration of novel topological designs and locomotion modes that can withstand various environmental challenges. Furthermore, drawing inspiration from nature, where some organisms exhibit energy-efficient locomotion, provides valuable insights for designing more efficient soft robotic locomotive systems [2]. For instance, small invertebrates such as tumblebugs (dung beetles) utilize tumbling movement (i.e., rapid deliberate rolling) to transport dung balls efficiently. Another example is the "tumbleweed", a plant structure that can be detached from its roots and blown by the wind, rolling and tumbling across arid landscapes. Additionally, some caterpillars employ tumbling locomotion by curling into a ball and rolling downhill as a defensive mechanism against predators.

Similar to biological creatures, soft mobile robots inherently have the potential to adopt unconventional locomotion modes such as tumbling, flipping, etc., in unique ways due to their deformable structures. Such locomotion methods require minimal energy expenditure as the robot utilizes its compliant body to initiate and sustain the rolling motion. Additionally, they offer several distinct advantages such as high maneuverability, orientation correction, adaptability to rough terrain, impact resilience, safe interactions, etc. For example, tumbling allows for enhanced maneuverability, enabling the robot to make tight turns and agile movements. In the event of the robot falling and resulting in a change in limb orientation, the

Manuscript received: October, 24, 2023; Revised January, 27, 2024; Accepted February, 21, 2024.

This paper was recommended for publication by Editor Yong-Lae Park upon evaluation of the Associate Editor and Reviewers' comments. This work is supported in part by the National Science Foundation (NSF) Grants IIS-2008797, CMMI-2048142, and CMMI-2132994.

¹Dimuthu D. K. Arachchige, Umer Huzaifa, and Iyad Kanj are with the School of Computing, DePaul University, Chicago, IL 60604, USA. {darachch, mhuzaifa, ikanj}@depaul.edu

²Dulanjana M. Perera is with the Department of Multidisciplinary Engineering, Texas A&M University, College Station, TX 77843, USA. dperera@tamu.edu

³Isuru S. Godage is with the Department of Engineering Technology & Industrial Distribution, J. Mike Walker '66 Department of Mechanical Engineering (affiliated), and Department of Multidisciplinary Engineering (affiliated), Texas A&M University, College Station, TX 77843, USA. igodage@tamu.edu

Digital Object Identifier (DOI): see top of this page.

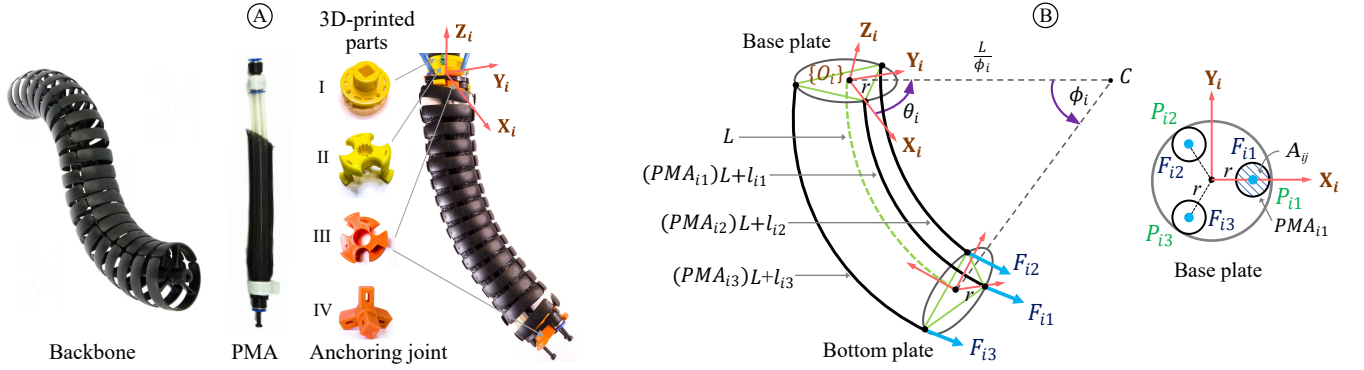


Fig. 2. (A) Deformed soft limb with design elements, showcasing a cross-section of a pneumatic muscle actuator (PMA) that reveals the interior design. (B) Schematics of the limb, illustrating curve parameters, actuator arrangement, and the pneumatic pressure forces exerted on the base plate by each actuator.

tumbling facilitates effective orientation correction. Moreover, their inherent flexibility provides a level of impact resistance. When encountering obstacles or collisions while tumbling, the robot's deformable body can absorb and distribute forces, reducing the risk of damage.

There are soft-limbed (or soft-bodied) robots that utilize rolling as an effective energy-conserving mode of locomotion. For example, soft snake robots reported in [3]–[5] utilize planar and spatial rolling to move across surfaces. The work reported in [6], [7] shows bioinspired caterpillar rolling. Additionally, soft robot prototypes proposed as soft-wheeler robots [8], [9], isoperimetric soft robot [10], magnetic grasping robot [11] effectively mimic rolling. Moreover, some soft robots such as those appeared in [12], [13] employ flipping as a mode of rolling to move the entire body in numerous directions. Soft-bodied robots have also utilized airborne locomotion (i.e., jumping) as an energy-efficient way of moving as demonstrated by the work reported in [14], [15]. Nevertheless, soft-bodied robots discussed herein suffer from limitations such as inadequate payload capacities and lack of versatility in locomotion modes due to their limbless designs.

Research shows a limited number of soft-limbed robots that utilize energy-efficient locomotion gaits such as rolling [16]. The soft robot prototypes reported in [17] and [18] have four limbs anchored in the open and closed tetrahedral topologies, respectively. They demonstrate rolling as a mode of locomotion, albeit at considerably low speeds. The soft tripod robot appeared in [19] utilizes jumping as its primary locomotion mode. However, it is not capable of producing successive jumps, hence the locomotion is limited to one cycle. In our previous work, [20], [21], we proposed a topologically symmetric soft-limbed robot assembled at the open tetrahedral topology. Therein, we studied pinniped locomotion and gait control with teleoperation. Their locomotion was based on crawling which is not an energy-efficient way of moving as it wastes a substantial amount of energy for friction due to distributed contacts of crawling limbs. However, the topologically symmetric limb arrangement of the robot can be leveraged to derive spatially symmetric energy-efficient locomotion patterns. In this work, we extend our previous work and systematically look into a novel discrete locomotion gait named tumbling. Our specific technical contributions include:

- i. proposing a locomotion strategy for tumbling utilizing soft

TABLE I
PHYSICAL DETAILS OF THE UNACTUATED TETRAHEDRAL ROBOT.

Item	Detail
Soft limb (Fig. 2A)	Initial length, $L = 24 \text{ cm}$
	Radius, $r = 2 \text{ cm}$
	Bending limit, $\phi_i = [0, 180^\circ]$
	Damping coefficient at 3 bar, $K_e = 700 \text{ Nm}^{-1}$
	Bending stiffness at 3 bar, $K_b = 0.94 \text{ Nmrad}^{-1}$
	Weight = 0.15 kg
Tetrahedral robot (Fig. 1A)	Body length = 45 cm
	Body width = 45 cm
	Body height = 30 cm
	Weight (without pressure supply tubes) = 0.85 kg

- limb kinematics,
- ii. developing a robot simulation model in a virtual environment that supports Physics interactions,
- iii. optimizing and validating the proposed tumbling locomotion strategy on the Physics-based robot simulation model,
- iv. validating the locomotion trajectories on the tetrahedral robot prototype based on optimized tumbling parameters of the Physics-based simulation model,
- v. evaluating and comparing the energy efficiency of tumbling against the robot's conventional locomotion gaits.

The results show that tumbling is an energy-efficient mode of locomotion than previously demonstrated crawling. To date, no study has been conducted on energy estimation in soft-limbed robot locomotion.

II. SYSTEM MODEL

A. Robot Design and Fabrication

The robot shown in Fig. 1A is made of four identical soft limbs named head (H), tail (T), right (R), and left (L) limbs. A soft limb is primarily fabricated using pneumatic muscle actuators (PMAs) and an inextensible rigid backbone (Fig. 2A). PMAs are fabricated in extension mode using silicone tubes, braided sleeves, and quick-connect air hose fittings. A commercial cable carrier (Triflex TRL40, Igus Inc USA) serves as the backbone, providing the required structural support for PMA integration. The backbone, comprising both an inner skeleton and an outer shell, supports omnidirectional bending. Three PMAs are placed within radial grooves of the backbone skeletal and locked in place using 3D-printed

units (Fig. 2A-I, II, III) and Nylon ties. This specific PMA and backbone arrangement creates an antagonistic actuator setup, where the backbone allows the PMAs to undergo length changes during operation while maintaining unrestricted omnidirectional bending.

Four soft limbs are connected using the tetrahedral-shaped limb anchoring joint (Fig. 2A-IV) to obtain the open tetrahedral topology shown in Fig. 1A. The robot has a symmetrical mass distribution supporting uniform maneuverability in all directions. Additionally, the spatial symmetry allows for generating discrete locomotion cycles after reorienting to an original pose while tumbling. Physical details of the robot are given in Table I. More technical details of the soft limb and the robot fabrication can be found in [22] and [20], respectively.

B. Soft Limb Kinematic Relationships

The schematics of Fig. 2B shows the PMA arrangement of any i -th soft limb, where $i \in \mathbb{Z}^+ \wedge [1 \dots 4]$ is the soft limb index of the robot. By assuming constant curvature bending, the configuration space parameters of the i -th soft limb are defined by the orientation angle, $-\pi \leq \theta_i \leq \pi$ and the bending angle, $0 < \phi_i \leq \pi$. The length changes of each PMA (i.e., joint variables), $l_{ij} \in \mathbb{R}$ with $j \in \mathbb{Z}^+ \wedge [1 \dots 3]$ being the PMA index, can be geometrically related to configuration space parameters as derived in [23] as

$$l_{ij} = -r\phi_i \cos\left(\frac{2\pi}{3}(j-1) - \theta_i\right). \quad (1)$$

Due to the inextensible nature, a soft limb holds the length constraint, $l_{i1} + l_{i2} + l_{i3} = 0$ [22]. This reduces the kinematic degrees of freedom of a soft limb into two. Applying this length constraint to (1), configuration space parameters (or curve parameters) can be derived in terms of joint variables in the simplified form in (2) where r is the radius of any soft limb (Fig. 2B).

$$\phi_i = \frac{2}{\sqrt{3}r} \sqrt{l_{i2}^2 + l_{i3}^2 + l_{i2}l_{i3}}, \quad (2a)$$

$$\theta_i = \arctan\left\{(l_{i3} - l_{i2}), \sqrt{3}(l_{i2} + l_{i3})\right\}. \quad (2b)$$

The Cartesian space to jointspace relationship (i.e., inverse kinematics) can be obtained as presented in [24] as

$$\theta_i = \arctan(y_i, x_i), \quad (3a)$$

$$\frac{1}{\phi_i} [1 - \cos(\phi_i)] = \frac{1}{L} \sqrt{x_i^2 + y_i^2}. \quad (3b)$$

C. Soft Limb Kinetostatic Relationships

Refer to the force, F_{ij} , acting on the base plate due to the distributed pneumatic pressure, P_{ij} , of PMAs, as illustrated in Fig. 2B. A pressure differential across PMAs creates a force imbalance, resulting in a net torque at the tip relative to O_i , causing the limb to bend. Following standard sign conventions, the X , Y , and Z components of the torque M_{ij} generated by each F_{ij} can be written as

$$M_{ij}|_X = 0F_{i1} + rF_{i2} \sin\left(\frac{\pi}{3}\right) - rF_{i3} \sin\left(\frac{\pi}{3}\right), \quad (4a)$$

$$M_{ij}|_Y = -rF_{i1} + rF_{i2} \cos\left(\frac{\pi}{3}\right) + rF_{i3} \cos\left(\frac{\pi}{3}\right), \quad (4b)$$

$$M_{ij}|_Z = 0. \quad (4c)$$

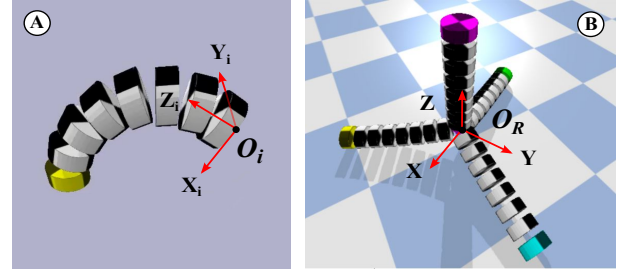


Fig. 3. A) Soft limb PyBullet model, B) tetrahedral PyBullet model.

Assuming a uniform cross-sectional area, A_{ij} exists in each PMA, then $F_{ij} = P_{ij}A_{ij}$ and, (4) can be further deduced as

$$M_{ij}|_X = \frac{\sqrt{3}A_{ij}r}{2} (P_{i2} - P_{i3}), \quad (5a)$$

$$M_{ij}|_Y = \frac{A_{ij}r}{2} (-2P_{i1} + P_{i2} + P_{i3}). \quad (5b)$$

Utilizing results in (5), the net bending torque which incorporates pneumatic actuation pressures, is expressed by (6), where K_b represents the bending stiffness.

$$M_{ij}|_{Net} = \sqrt{(M_{ij}|_X)^2 + (M_{ij}|_Y)^2} = K_b\phi_i. \quad (6)$$

III. PHYSICS-BASED ROBOT SIMULATION MODEL

Herein, we develop a Physics-based real-time simulation model of the tetrahedral robot to streamline the process of obtaining tumbling trajectories. It is used to test, fine-tune and validate the proposed locomotion method in advance before moving into the experimental testing.

A. Bullet Physics Engine

We utilized an open-source Physics engine, Bullet Physics to develop the tetrahedral simulation model [25]. It offers a collection of resources and software components for emulating the behavior of solid objects in motion, detecting collisions, and enabling physics-driven interactions within a virtual environment that replicates real-world physics characteristics. It includes functionalities like kinematic and dynamic simulations, friction models, gravitational effects, and contact forces. The Physics engine incorporates PyBullet, a Python interface designed for robotic simulations with a primary emphasis on rigid components and their joints [26]. However, it only accommodates simulations of inert soft materials such as fabric, rubber, foam, and other substances that display properties of softness and flexibility. The soft limb under consideration in this work is an active soft body that generates motions. In Bullet-Physics, the mass-spring-damper approach is effectively applied for modeling soft bodies. Herein, we utilized the same approach to model soft limbs for effectiveness [27].

B. Soft-limb and Tetrahedral Robot Simulation Models

A soft limb is formed by serially joining discrete rigid disks. In order to create a soft limb within the Bullet Physics environment, a URDF (Unified Robot Description Format) file that encompasses both the physical and visual characteristics of the limb is created. Therein, masses of rigid disks, their positions and orientations, moments of inertia, joint types, joint positions and orientations, joint physics (stiffness, friction),

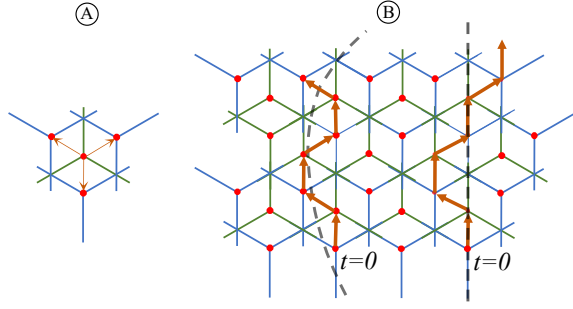


Fig. 4. Discrete Tumbling Locomotion: (A) shows the three directions in which the robot can tumble, with green lines indicating the robot's initial pose. (B) illustrates the robot's workspace, with red dots representing the points where the robot can move (i.e., the movement of the robot's center of gravity). The black dotted lines denote desired straight and curvilinear paths.

etc., are defined. Additionally, the appearance of rigid disks is defined by setting a mesh target using 3D-modeling software, Blender. Using a URDF plugin installed in Blender, an XML-based URDF file is generated. It is then loaded into Python for simulations with the help of PyBullet.

By incorporating the physical details of the limb prototype given in Table I, the soft limb PyBullet model is constructed using eight discrete rigid disks as shown in Fig. 3A. The total weight is evenly distributed among eight disks. Two revolute joints, controlled in position control mode through PyBullet, are placed along the X and Y axes between two rigid disks. The two revolute joints are sequentially actuated to manipulate the limb according to the given curve parameters (θ_i, ϕ_i in Sec. II-B). The bending torque, as defined in (6), is uniformly distributed across all revolute joints. In position control mode, the physics engine automatically calculates and applies the corresponding directional torques to the revolute joints. The damping and bending stiffness coefficients of the soft limb, as provided in Table I, are set following an experimental identification process outlined in our previous work [28]. A tetrahedral-shaped fixed joint (Fig. 2A-IV) is utilized to anchor four limbs and assemble the tetrahedral PyBullet model shown in Fig. 3B.

We characterized the soft limb PyButllet model to match the behavior of the soft limb prototype. Therein, first, we tested a circular trajectory on the limb prototype as demonstrated in our previous work [24]. Following this, an identical limb trajectory was applied to the PyBullet soft limb model. We obtained a comparable behavior on the PyBullet model by fine-tuning the bending stiffness and damping coefficients.

IV. TUMBLING LOCOMOTION

Tumbling locomotion refers to an unconventional method of movement where the robot intentionally induces rolling or flipping motions to navigate through its environment. Due to the spatially symmetric limb structure, the tumbling of the tetrahedral robot generates a discrete locomotion pattern. This motion creates discrete points in Cartesian space following a honeycomb structure as illustrated in Fig. 4. Therein, any locomotion trajectory (such as the highlighted straight and curvilinear paths) can be realized by moving the robot across discrete points marked in red color.

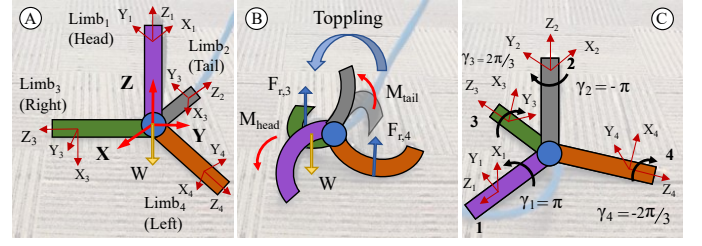


Fig. 5. A tumbling cycle: (A) initial pose, (B) an intermediate tumbling pose, and (C) a tumbled pose. The black arrow and γ_i indicate the required direction and magnitude of rotation to maintain the standard frame orientation.

A. Tumbling Strategy

The tumbling is an extension of the previously proposed crawling locomotion [20]. Here, the robot's center of gravity (CoG) is moved beyond the balance triangle to topple the robot [29]. This is achieved by bending the tail against the floor (i.e., downward) and the head towards the moving direction (i.e., forward), simultaneously in a linear trajectory as illustrated in Fig. 5. Consequently, the generated angular moments in each limb (M_{head}, M_{tail}) tip over the robot around ground contact points. In order to guarantee the completion of a rolling cycle, the robot must be given an adequate amount of angular momentum towards the tumbling direction. This is attained by reversing the bending direction of the tail towards the moving direction (i.e., forward) at the time the robot completes its rolling cycle. Additionally, the left and right limbs crawl and move the robot forward while shifting the ground contact closer to the CoG facilitating the robot's translation [21]. Herein, head, tail, and left-right limbs undergo 3 distinct movements that can be parameterized by the curve parameters (θ_i, ϕ_i in Sec. II-B) of each limb.

Let the $\vec{p}_i = [x_i, y_i]$ be the Cartesian coordinate vector of the heel of any i -th limb relative to its own coordinate frame, $\{O_i\}$. Then, the taskspace, (x_i, y_i) of a circular limb heel trajectory is given as presented in [20] as

$$\begin{aligned} x_i &= \rho_i \cos(\alpha_i), \\ y_i &= \rho_i \sin(\alpha_i), \end{aligned} \quad (7)$$

where ρ_i is the trajectory stride radius and $\alpha_i = \frac{2\pi t}{T}$ is the angular displacement of the i -th limb at a time, t within the trajectory period, T .

The proposed limb motions can be repeated to obtain multiple tumbling cycles. Therein, based on the new tumbling direction, limbs' linear and circular trajectory taskspace is transformed via corresponding Z axis rotations, R_z as

$$\vec{\hat{p}}_i = R_z(\gamma_i) \cdot \vec{p}_i \quad (8)$$

where $\hat{p}_i$ is the remapped taskspace of the i -th limb and γ_i is the corresponding rotation angle needed to reorient the limb frame in the next tumbling cycle. γ_i is determined by taking into account the tetrahedron geometry, current robot orientation, and next tumbling direction. Fig. 5C illustrates the aforementioned remapping according to the corresponding angle offsets of each limb. For example, during the previous tumbling cycle, limb-4 rotated $\frac{2\pi}{3}$ around the $Z+$ axis. Therefore, to maintain the standard frame orientation, we must rotate the frame $-\frac{2\pi}{3}$ around the $Z+$ axis.

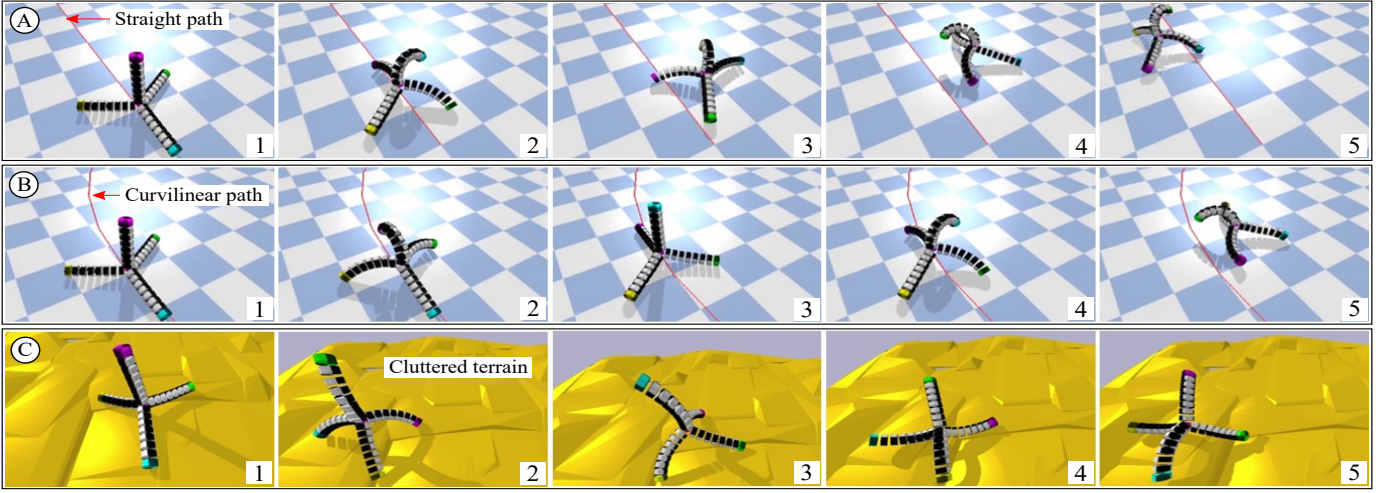


Fig. 6. Progression of consecutive tumbling cycles on tetrahedral PyBullet model under, (A) straight, (B) curvilinear, and, (C) cluttered paths.

TABLE II
OPTIMIZED CURVE PARAMETERS OF A TUMBLING CYCLE IN PYBULLET.

Limb	Curve parameter	Optimized ceiling [rad]	
		$0 < t \leq T/2$	$T/2 < t \leq T$
Head	θ_1	$-\pi$	0
	ϕ_1	$[0, 5\pi/12]$	0
Tail	θ_2	0	$-\pi$
	ϕ_2	$[0, \pi/3]$	$[0, \pi/4]$
Right	θ_3	$[0, -3\pi/4]$	0
	ϕ_3	$7\pi/18$	0
Left	θ_4	$[0, 3\pi/4]$	0
	ϕ_4	$7\pi/18$	0

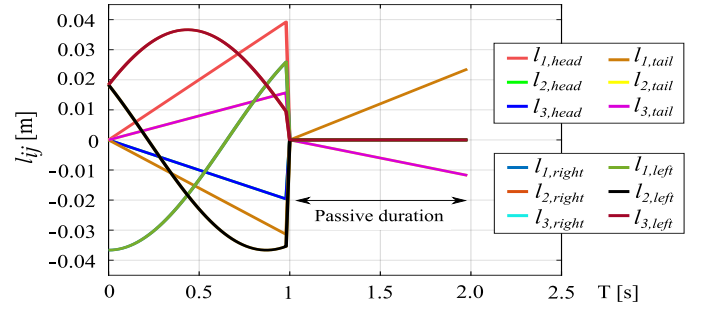


Fig. 7. Joint variables of soft limbs during a single tumbling cycle.

B. Validate Tumbling on Tetrahedral Simulation Model

The proposed limb actuation philosophy for tumbling in Sec. IV-A are tested in the Tetrahedral PyBullet model to validate and fine-tune.

The friction coefficients of the simulation floor are set as, $\mu_x = \mu_y = 0.6$, closely matching those of the actual floor (experimentally found in our previous work [4]), where the tetrahedral prototype will be tested. The gravitational acceleration is set to 9.81 ms^{-2} . Following the proposed tumbling strategy, the testing range of each limb parameter (θ_i, ϕ_i) is computed using (3), based on empirically identified taskspaces, (x_i, y_i) of each limb as presented in our previous work [20]. For example, the testing range of stride radius, ρ_i given in (7) (i.e., radii of circular crawling trajectories of left and right limbs) is determined as $\rho_i \in [4, 10] \text{ cm}$ based on the operating workspace of soft limbs. In the end, the joint position control method available in the PyBullet is applied to feed curve parameters into the PyBullet simulation model.

The testing range of limb actuating frequencies is set as, $f \in [0.25, 1.00] \text{ Hz}$ based on the operating bandwidth of soft limb prototypes [24]. The tumbling ability of the PyBullet simulation model is iteratively tested for different combinations of limb parameters and frequency ceilings. Through repeated testing, we identified the optimized regions of each limb parameter approximated to the ceilings presented in Table II. We observed that low limb actuation frequencies ($0.25 \text{ Hz} < f < 0.50 \text{ Hz}$) led to unsuccessful tumbling attempts. However, the tumbling was sustained at a critical threshold of $f = 0.55 \text{ Hz}$ and beyond. The reason behind this behavior is the inadequacy

of low limb actuation rates in generating the essential angular momentum to topple the robot. Next, we repeat the tumbling motion within the optimized trajectory margins to obtain straight and curvilinear locomotion simulations. Additionally, we simulated tumbling on a cluttered terrain characterized by obstacles, sharp edges, slopes, and varied frictional surfaces.

Here, for a given trajectory path (e.g., straight or curvilinear), the tumbling sequence is determined by the robot's operator. Since the robot rolls on the $X-Z$ plane (Fig. 5A), the operator can tumble the robot into a desired direction by inputting the limb parameters of the head and tail that operate on the $X-Z$ plane. We implemented the limb remapping approach proposed in Sec. IV-A to transform the trajectories from the current orientation into a new orientation. Figures 6A, 6B, and 6C illustrate successive simulation frames depicting both straight and curvilinear tumbling, as well as tumbling on a cluttered terrain, in the PyBullet simulation model. The complete simulations are included in our supplementary file (Refer to Movie 01). The results show that the PyBullet simulation model successfully tracks straight, curvilinear, and cluttered paths via discrete tumbling locomotion.

V. EXPERIMENTAL RESULTS

A. Testing Tumbling

Initially, the robot prototype is tested on a carpeted floor (Fig. 8 - flat terrain) that has nearly consistent friction. To actuate the prototype, the joint variables (l_{ij}) shown in Fig. 7 are generated by applying optimized limb parameters in Table

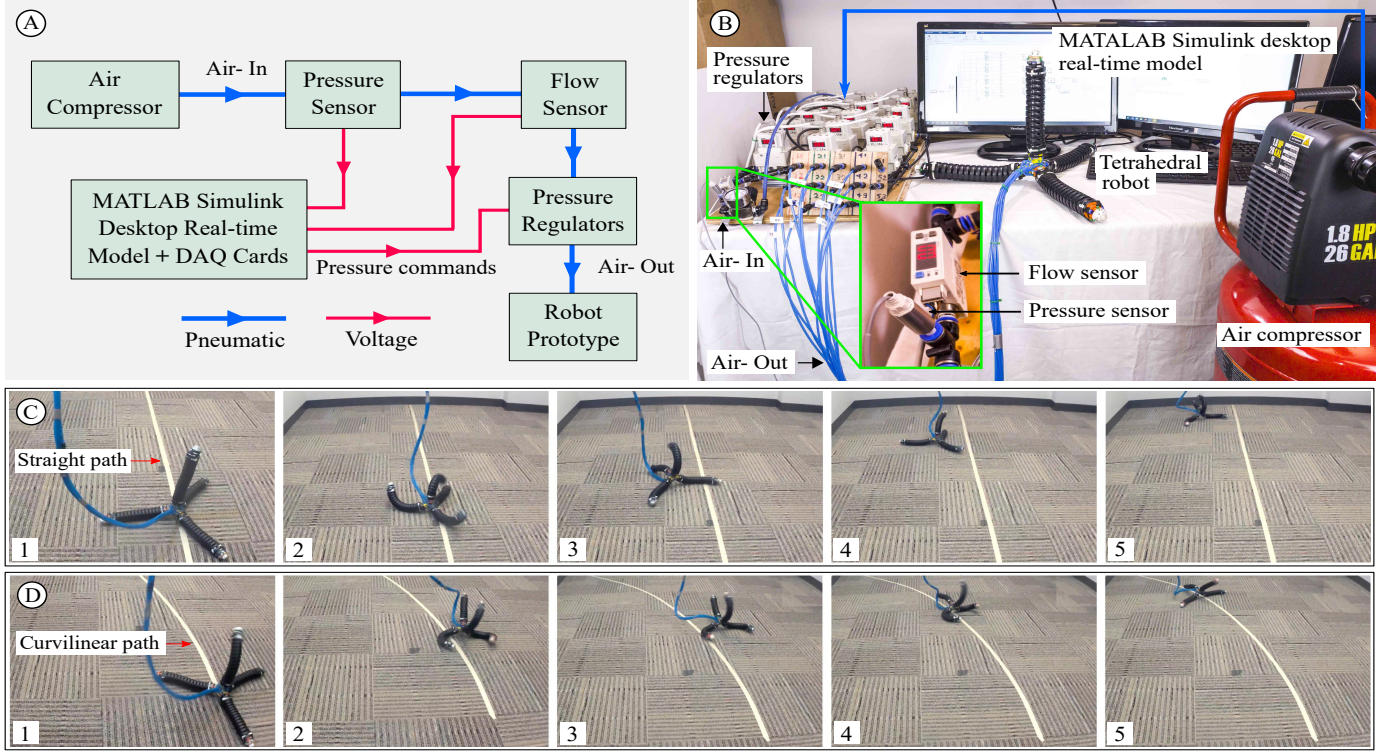


Fig. 8. (A) Block diagram and (B) hardware components of the robot operating setup. Experimental progression of (C) straight and (D) curvilinear tumbling on flat terrain at $3 \text{ bar} - 0.75 \text{ Hz}$, actuation pressure - frequency combination.

II into (1). Next, l_{ij} should be mapped to actuation pressure trajectories. The joint variable – pressure mapping approach proposed in [24] is applied to obtain pressure trajectories.

Figures 8A and 8B show the block diagram and hardware components of the robot actuation setup, respectively. The air pressure from the compressor is fed to 12 pressure regulators (ITV3050-31F3N3, SMC USA) via a pressure sensor and a flow sensor, corresponding to 12 PMAs of 04 limbs. The pressure regulators release air pressure based on voltage signals issued by an analog output data acquisition (DAQ) card (PCI-6703, NI USA). The DAQ card is connected to a MATLAB Simulink desktop real-time model where corresponding actuation pressure trajectories are set. Therein, the pressure values are mapped to the voltage signals applied to pressure regulators (i.e., pressure commands).

We tested the robot tumbling by applying the optimized curve parameters in Table II within the optimized actuation frequency region, $f \in [0.55, 1.00] \text{ Hz}$. We initiated the testing at the pressure ceiling, $p = 3 \text{ bar}$, and the actuation frequency, $f = 0.55 \text{ Hz}$. We found that a high-pressure ceiling ($p > 3.0 \text{ bar}$) generated unexpected jerks due to over-stiffed limbs. Consequently, the pressure ceiling for subsequent testing was set at 3 bar . We further noted that low frequencies ($f < 0.55 \text{ Hz}$) onto PMAs cannot generate adequate forward momentum to tumble the robot, which is consistent with the testing results of the PyBullet model. Hence, the robot testing was repeated by increasing the frequency ceiling by 0.05 Hz steps. The robot started tumbling effectively at about 0.65 Hz and sustained it till about 0.90 Hz . We observed that high frequencies ($f > 0.90 \text{ Hz}$) led to unsuccessful tumbling attempts. This is owing to the fact that high limb actuation

rates result in incomplete limb deformation because those exceed the operational bandwidth of PMAs.

Figures 8C and 8D depict the trajectory tracking of straight and curvilinear tumbling on flat terrain, respectively, at a frequency of 0.75 Hz — the optimal frequency that produces the best tumbling motion. Figures 9A, 9B, 9C, and 9D demonstrate various testing scenarios for tumbling, including orientation correction, negotiating inclined surfaces (slope angle = 30°), cluttered terrains (fabric surface formed with underneath obstacles), and natural-like irregular terrains (sand, pebbles, and grass), respectively. Refer to Movie 02 of the supplementary file to see the complete tumbling videos.

B. Discussion

Discrepancies can be observed between the simulated behavior of the PyBullet model and the actual performance of the robot prototype in experiments. The reasons for that can be identified as follows. The robot prototype is bound by its operational limitations. For example, the inconsistency in pressure reaching PMAs can be attributed to the use of long pressure supply tubes, which act as a low-pass filter, limiting their ability to transmit rapid pressure changes in real-time. Additionally, the PyBullet soft limb model did not account for limb characteristics such as deadzone, hysteresis of PMAs, etc. that are present in the soft limb prototype.

C. Estimate Energy Consumption in Tumbling

We investigated the energy usage of tumbling locomotion and compared it against the robot's conventional locomotion gait, i.e., crawling [20] (see Movie 02 of the supplementary file). If the robot is actuated by an input pneumatic pressure, $P_{in} [\text{Nm}^{-2}]$ with a volumetric air flow rate, $\dot{V} [\text{m}^3\text{s}^{-1}]$, the

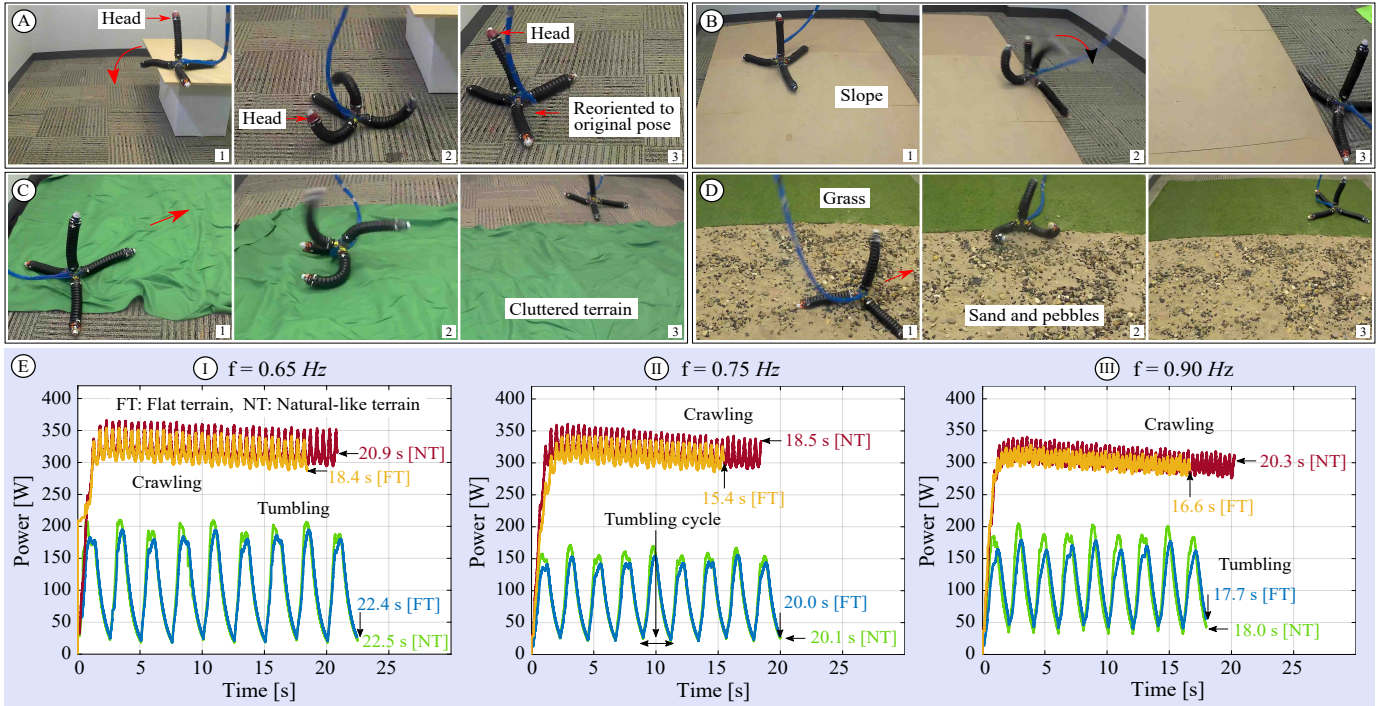


Fig. 9. Robot testing for (A) orientation correction, (B) inclined surface, (C) cluttered terrain, & (D) natural-like irregular terrain. (E) Assessing power output of the pneumatic system involves tumbling & crawling at actuation frequencies of (I) 0.65 Hz, (II) 0.75 Hz, & (III) 0.90 Hz on flat & natural-like terrains.

TABLE III
ENERGY ESTIMATION OF LOCOMOTION

Locomotion gait	Energy consumption [KJ]					
	Flat terrain			Natural-like terrain		
	Frequency [Hz]			Frequency [Hz]		
	0.65	0.75	0.90	0.65	0.75	0.90
Crawling	5.71	4.60	4.87	6.52	5.71	5.58
Tumbling	2.33	1.70	2.01	2.40	1.82	2.10
Energy saving, E_{saving} [%]	59.2	63.0	58.7	63.2	68.1	62.4

power output of the pneumatic system can be computed as, $P_{in}\dot{V}$. Accordingly, the energy spent by the pneumatic system (or energy input to the robot), E is given by (9) where t denotes the robot actuation duration.

$$E = \int_0^t P_{in}\dot{V} dt. \quad (9)$$

To measure the input air pressure and flow rate, a pressure sensor (PSE530-R07-L, SMC USA) and a flow sensor (PFM711-C6-E-M, SMC USA) are serially coupled at the air inlet of pressure regulators (Refer Figs. 8A and 8B). An analog input DAQ card (PCI-6255, NI USA) is interfaced with the MATLAB Simulink desktop real-time model to acquire sensor data that are generated as voltage signals.

We tumbled the robot a fixed straight distance (2.5 m) at low, mid, and high frequencies ($f = \{0.65, 0.75, 0.90\}$ Hz), independently on flat terrain (Fig. 8C) and recorded pressure input, flow rate, and traversal time during each locomotion cycle. Next, we repeated the same step for crawling under identical conditions. Subsequently, both tumbling and crawling actuations were repeated on natural-like irregular terrain (Fig. 9D). Figure 9E shows the computed power outputs of the pneumatic system (or power input to the robot) using (9). The crawling exhibits a slight downward slope in power output,

which is a result of the drop in the flow rate from the air tank due to the loss of tank pressure because crawling draws air at a faster rate. Table III summarizes the respective energy outputs. The decrease in flow rate has no impact on the energy estimation because the flow rate is accurately measured.

D. Analyse Energy Consumption

According to Fig. 9E, the robot takes longer traversing the desired fixed distance via tumbling than crawling under each actuation frequency at all times. This is due to tumbling being a discrete locomotion gait that requires time to recover or stabilize between tumbles, whereas crawling can continue cyclically. The average moving speed of crawling and tumbling under 3 actuation frequencies on all terrains can be found as 13.8 cms^{-1} and 12.5 cms^{-1} , respectively. Here, the relative speed decrease in tumbling is 10.4%.

Based on the data in Table III tumbling uses lower energy than crawling at all times. Accordingly, we computed the energy efficiency of tumbling relative to crawling, given in the last row of Table III, as

$$E_{saving}[\%] = \frac{E_{crawling} - E_{tumbling}}{E_{crawling}} \times 100. \quad (10)$$

According to Table III, tumbling saves approximately 60% of energy on flat terrain and 65% on natural-like irregular terrain compared to crawling, irrespective of actuation frequencies and similar locomotion speeds. Tumbling significantly reduces floor friction by reducing contact surface area. In contrast, crawling, where continuous contact with the floor consumes significant energy due to distributed limb contacts. Additionally, not all limbs are continuously actuated during the entire tumbling cycle, as seen in Table II and Fig. 7. Limbs are active during the first half, contributing to the tumbling action, and passive during the second half. Consequently, the

pneumatic system does not consume power throughout the entire tumbling cycle, unlike continuous crawling.

The lowest energy saving occurs at the highest limb actuation frequency, $f = 0.90 \text{ Hz}$, due to the increased work needed to overcome floor friction. At high frequencies, limbs cannot reach their full deformation potential, given fast pressure changes. Conversely, moderate actuation frequencies enable full limb operation within their workspace. Therefore, Table III highlights the highest energy saving at the moderate frequency, $f = 0.75 \text{ Hz}$, identified as the optimal actuation frequency for tumbling. It is important to note that the energy saving at the lowest limb actuation frequency, $f = 0.65 \text{ Hz}$, falls between the highest and moderate actuation frequencies. At reduced limb actuation frequencies, the robot takes an extended time to reach its destination, as seen in Fig. 9E. Thus, the energy input to the robot increases due to the prolonged operating time of the pneumatic power system.

According to Table III, the energy saving in tumbling locomotion on natural-like irregular terrains is higher than on flat terrains. Fig. 9E illustrates that on both flat and natural-like irregular terrains, tumbling covers the same distance within comparable time intervals. This advantage of discrete locomotion modes like tumbling arises from dealing less with floor friction. Conversely, during crawling, the robot must negotiate surface variations in natural-like irregular terrain through distributed contact forces, leading to higher frictional resistance, which causes increased energy consumption compared to traversing flat terrain.

VI. CONCLUSIONS

The exploration of energy-efficient locomotion in soft robotics represents a significant step forward in the field. The inherent compliance and deformability of soft robots have paved the way for unconventional, innovative approaches. In this work, we demonstrated an unconventional discrete locomotion gait named tumbling using a topologically symmetric soft-limbed robot. Tumbling offers numerous benefits, including enhanced maneuverability, in situ orientation correction, energy conservation, adaptability to challenging terrain, resilience to impacts, and safe interactions, among others. Through a combination of rigorous simulation and prototype testing, we demonstrated the viability and energy efficiency of this novel locomotion mode. Our future efforts will focus on advancing tumbling with multimodal locomotion to navigate unstructured terrains.

REFERENCES

- [1] M. Calisti, G. Picardi, and C. Laschi, "Fundamentals of soft robot locomotion," *Jnl. of Royal Society Interface*, vol. 14, no. 130, 2017.
- [2] F. Ahmed, M. Waqas, B. Jawed, A. M. Soomro, S. Kumar, A. Hina, U. Khan, K. H. Kim, and K. H. Choi, "Decade of bio-inspired soft robots: A review," *Smart Materials & Structures*, vol. 31, no. 7, 2022.
- [3] D. D. Arachchige, Y. Chen, and I. S. Godage, "Soft robotic snake locomotion: Modeling and experimental assessment," in *IEEE Intl. Conf. Automation Science and Engineering (CASE)*, 2021, pp. 805–810.
- [4] D. D. Arachchige, S. Mallikarachchi, I. Kanj, D. M. Perera, Y. Chen, H. B. Gilbert, and I. S. Godage, "Dynamic modeling and validation of soft robotic snake locomotion," in *IEEE Intl. Conf. on Control, Automation, and Robotics (ICCAR)*, 2023.
- [5] D. D. Arachchige, D. M. Perera, S. Mallikarachchi, I. Kanj, Y. Chen, and I. S. Godage, "Wheelless soft robotic snake locomotion: Study on sidewinding and helical rolling gaits," in *IEEE Intl. Conf. on Soft Robotics (RoboSoft)*, 2023.
- [6] H.-T. Lin, G. G. Leisk, and B. Trimmer, "Gogobot: a caterpillar-inspired soft-bodied rolling robot," *Bioinspiration & biomimetics*, 2011.
- [7] D. K. Patel, X. Huang, Y. Luo, M. Mungekar, M. K. Jawed, L. Yao, and C. Majidi, "Highly dynamic bistable soft actuator for reconfigurable multimodal soft robots," *Advanced Material Technologies*, 2023.
- [8] W.-B. Li, W.-M. Zhang, H.-X. Zou, Z.-K. Peng, and G. Meng, "A fast rolling soft robot driven by dielectric elastomer," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 4, pp. 1630–1640, 2018.
- [9] W.-B. Li, W.-M. Zhang, Q.-H. Gao, Q. Guo, S. Wu, H.-X. Zou, Z.-K. Peng, and G. Meng, "Electrically activated soft robots: Speed up by rolling," *Soft Robotics*, vol. 8, no. 5, pp. 611–624, 2021.
- [10] N. S. Usevitch, Z. M. Hammond, M. Schwager, A. M. Okamura, E. W. Hawkes, and S. Follmer, "An untethered isoperimetric soft robot," *Science Robotics*, vol. 5, no. 40, 2020.
- [11] X. Wang, Z. Bo, C. Guo, J. Wu, Y. Liu, and W. Sun, "A magnetic soft robot with rolling and grasping capabilities," *Journal of Mechanical Engineering Science*, 2022.
- [12] M. Malley, M. Rubenstein, and R. Nagpal, "Flippy: A soft, autonomous climber with simple sensing and control," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017, pp. 6533–6540.
- [13] J. Wang, Y. Fei, and Z. Liu, "Fifobots: foldable soft robots for flipping locomotion," *Soft Robotics*, vol. 6, no. 4, pp. 532–559, 2019.
- [14] N. W. Bartlett, M. T. Tolley, J. T. Overvelde, J. C. Weaver, B. Mosadegh, K. Bertoldi, G. M. Whitesides, and R. J. Wood, "A 3d-printed, functionally graded soft robot powered by combustion," *Science*, 2015.
- [15] A. Fernandes Minori, S. Jadhav, H. Chen, S. Fong, and M. T. Tolley, "Power amplification for jumping soft robots actuated by artificial muscles," *Frontiers in Robotics and AI*, vol. 9, p. 844282, 2022.
- [16] Y. Sun, A. Abudula, H. Yang, S.-S. Chiang, Z. Wan, S. Ozel, R. Hall, E. Skorina, M. Luo, and C. D. Onal, "Soft mobile robots: A review of soft robotic locomotion modes," *Current Robotics Reports*, 2021.
- [17] Y. Wang, J. Wang, and Y. Fei, "Design and modeling of tetrahedral soft-legged robot for multigait locomotion," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 3, pp. 1288–1298, 2021.
- [18] P. Wharton, T. L. You, G. P. Jenkinson, R. S. Diteesawat, N. H. Le, E.-C. Hall, M. Garrad, A. T. Conn, and J. Rossiter, "Tetraflex: A multigait soft robot for object transportation in confined environments," *IEEE Robotics and Automation Letters*, 2023.
- [19] M. T. Tolley, R. F. Shepherd, M. Karpelson, N. W. Bartlett, K. C. Galloway, M. Wehner, R. Nunes, G. M. Whitesides, and R. J. Wood, "An untethered jumping soft robot," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2014, pp. 561–566.
- [20] D. D. Arachchige, T. Varshney, U. Huzaifa, I. Kanj, T. Nanayakkara, Y. Chen, H. B. Gilbert, and I. S. Godage, "Study on soft robotic pinniped locomotion," in *IEEE/ASME Intl. Conf. on Advanced Intelligent Mechatronics (AIM)*, 2023, pp. 65–71.
- [21] D. M. Perera, D. D. Arachchige, S. Mallikarachchi, T. Ghafoor, I. Kanj, Y. Chen, and I. S. Godage, "Teleoperation of soft modular robots: Study on real-time stability and gait control," in *IEEE Intl. Conf. on Soft Robotics (RoboSoft)*, 2023.
- [22] D. D. Arachchige and I. S. Godage, "Hybrid soft robots incorporating soft and stiff elements," in *IEEE Intl. Conf. on Soft Robotics (RoboSoft)*, 2022, pp. 267–272.
- [23] I. S. Godage, G. A. Medrano-Cerda, D. T. Branson, E. Guglielmino, and D. G. Caldwell, "Modal kinematics for multisection continuum arms," *Bioinspiration & biomimetics*, vol. 10, no. 3, p. 035002, 2015.
- [24] D. D. Arachchige, D. M. Perera, S. Mallikarachchi, U. Huzaifa, I. Kanj, and I. S. Godage, "Soft steps: Exploring quadrupedal locomotion with modular soft robots," *IEEE Access*, vol. 11, pp. 63 136–63 148, 2023.
- [25] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," <http://pybullet.org>.
- [26] —, "Pybullet quickstart guide," 2021.
- [27] A. Amaya, D. D. Arachchige, J. Grey, and I. S. Godage, "Evaluation of human-robot teleoperation interfaces for soft robotic manipulators," in *Intl. Conf. on Robots & Human Interactive Communication (RO-MAN)*, 2021, pp. 412–417.
- [28] D. D. Arachchige, Y. Chen, I. D. Walker, and I. S. Godage, "A novel variable stiffness soft robotic gripper," in *IEEE Intl. Conf. on Automation Science and Engineering (CASE)*, 2021, pp. 2222–2227.
- [29] A. Iscen, A. Agogino, V. SunSpiral, A. Tumer, Kagan Tuleu, M. Vespignani, M. Ajallooeian, E. Badri, and A. J. Ijspeert, "Flop and roll: Learning robust goal-directed locomotion for a tensegrity robot," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2014.