The Tall, the Squat, & the Bendy: Parametric Modeling and Simulation Towards Multi-functional Biohybrid Robots

Saul Schaffer and Victoria A. Webster-Wood(✉)

Carnegie Mellon University, Pittsburgh, PA 15213, USA
vwebster@andrew.cmu.edu
http://engineering.cmu.edu/borg

Abstract. Efficient computational mechanical models are needed to develop complex, multifunctional soft robotics systems. The field of biohybrid robotics is no exception and is currently limited to low degree-of-freedom lab-chained research devices due in part to limitations in existing design tools and limitations in the morphological sophistication of existing biohybrid structures. Here, we present an expanded use of an existing soft body modeling tool PyElastica, along with a parametric design pipeline for generating and simulating a lattice-based distributed actuation biohybrid robot. Our key contribution in this work is the parameterization of the geometry defining the lattice robot architecture both in terms of the bulk structure and the patterning of the muscles on that structure. By encoding multifunctionality in the robot's architecture, we demonstrate extension, compression, and bending motion primitives exhibited by the same base lattice structure. We achieve structure-wide strains of 49.73% for extension, 30.59% for compression, and a 48.60° bend angle for the bent configuration, with all simulations completing in less than 45 s. From this pilot study, the computational model will be expanded to capture more complex and functional behaviors, such as esophageal-inspired peristalsis for internal transport, as well as earthworm-inspired locomotion. The computational modeling of these behaviors is a critical step toward the eventual design, fabrication, and deployment of complex biohybrid robots.

Keywords: Biohybrid · Distributed actuation · Soft-body modeling · Lattice · PyElastica

1 Introduction

Biohybrid robots show substantial promise for applications ranging from medicine to agriculture and environmental monitoring. Living muscle-based

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-39504-8_15.

F. Meder et al. (Eds.): Living Machines 2023, LNAI 14158, pp. 217–226, 2023.
https://doi.org/10.1007/978-3-031-39504-8_15

actuators can endow mechanical systems with desirable capabilities such as high compliance [9,21], self-healing [13], and biocompatibility [14]. To date, most biohybrid robots consist of low-degree-of-freedom systems that can achieve basic motions such as crawling [2,7,11,19], swimming [1,6,8,12], and flexion [9]. While promising, these modes of motion are limited. To achieve higher levels of functionality, biohybrid robots with more complex behavioral capabilities are needed.

A major challenge in designing and realizing complex biohybrid robots capable of multifunctional behavior is the limited number of modeling tools well-suited for biohybrid simulation [21]. Due to the time and expense required to fabricate and assess biohybrid robots, experimentally iterating to refine designs for complex, multifunctional robots is impractical. Unfortunately, many existing robot modeling tools focus on rigid body systems and traditional modes of actuation [10]. Few tools can capture the mechanics of multi-actuator biohybrid systems [15,18]. One could use approaches from solid mechanics modeling. However, traditional finite element tools are not well-suited for handling complex morphologies with large deformations in a reasonable (sub-hour) amount of time, resulting in long design-iteration cycle times. To enable the design of complex, multifunctional biohybrid robots, modeling tools need to be computationally efficient, amenable to large deformations, modular enough to allow the implementation of biologically relevant muscle contraction profiles, and accessible enough that they do not pose major barriers to entry to new researchers in the field [20].

To address the need for an accessible modeling pipeline for biohybrid robotics researchers, this work presents a novel framework for parametrically defining model morphology in the open-source soft modeling tool PyElastica [4,22]. Our pipeline is developed specifically for biohybrid robots with distributed muscle-actuators and lattice-based architectures, inspired by the "Meshworms" from MIT [16] and Case Western Reserve University [3]. These lattice-based architectures provide the physical complexity needed to achieve multifunctional behaviors while remaining amenable to both parameterization for design exploration and future physical fabrication via 3D bioprinting methods [5]. Our lattice robot features tens of muscle actuators that are patterned across their bulk structure (Fig. 3), which would be challenging to implement in PyElastica manually. As a test case for our parametric model definition approach, we use our tool to automatically create the biohybrid lattice robot model and position muscles based on user input. For this pilot study, we then demonstrate the simulated lattice robot achieving three basic motion primitives: extension (*i.e.* making the robot tall), compression (squat), and bending. By strategically combining these motion primitives, future research will build off the work presented here to demonstrate functional behaviors using this lattice architecture, such as esophageal-inspired peristalsis and locomotion. Furthermore, this work has the potential to broaden participation in biohybrid robotics by making biohybrid robot modeling and design more accessible through parametric model definition.

2 Methods

Simulation of the lattice worm (Fig. 1) was conducted using PyElastica (version 0.3.0) [4,22]. PyElastica is a software package for simulating assemblies of soft slender bodies using Cosserat Rod theory. A Cosserat rod is a mathematical description that captures the 3D dynamics of 1D slender bodies while accounting for all modes of deformation (*i.e.* bending, twisting, stretching, and shearing as well as mechanical instabilities such as buckling) [22]. For a detailed explanation of Cosserat Rod Theory, readers are encouraged to visit www.cosseratrods.org, a site managed by the creators of PyElastica.

For curved rods, PyElastica requires that every node be defined by an (x, y, z) position. Thus, generating architectures within PyElastica comprised of multiple curved rods can be challenging. To generate the helices that comprise our lattice structure, we created a pipeline for parametrically generating the discrete elements that comprise each helix, as well as all necessary connections and boundary conditions. This pipeline is a suite of custom Python functions that enable the facile instantiation and simulation of lattice-based biohybrid structures.

2.1 Parametric Model Generation for a 3D Biohybrid Lattice Robot

The pipeline for generating the structural rods and connections of the lattice as well as the muscle rods and their connections, is detailed in Fig. 2. In the proof-of-concept work described here, we have focused on lattice structures composed of 12 rods. However, this framework can be extended to other rod quantities by implementing the appropriate intersection geometry calculations. At a high level, the geometric parameters that define the rods (*i.e.* height h, width w, phase angle ϕ, and chirality χ) are input by the user into a Parametric Structure Rod Generator. From these parameters, all 12 rods that compose the bulk structure of the lattice (Fig. 1, purple rods) are generated. The intersections R between structure rods are automatically calculated based on the geometry and discretization of the structure. These intersections are used to define connections between structure rods, as well as locations for muscle rods

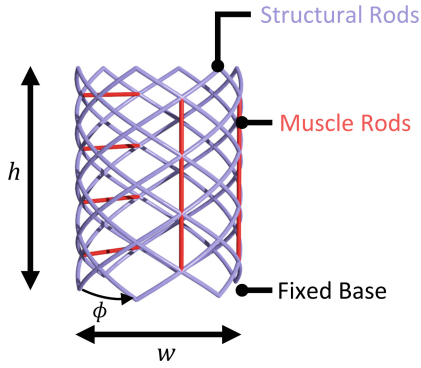

Fig. 1. The geometry of the lattice worm is defined and generated parametrically based on user-specified height, width, phase, chirality, and muscle locations. The structural rods (purple) are generated first, and all intersections are connected. The user then specifies which muscle rods (red) to include, and they are automatically connected at the appropriate nodes. For this work, the structure's base was grounded with a fixed boundary constraint on the first node of each structural rod. (Color figure online)

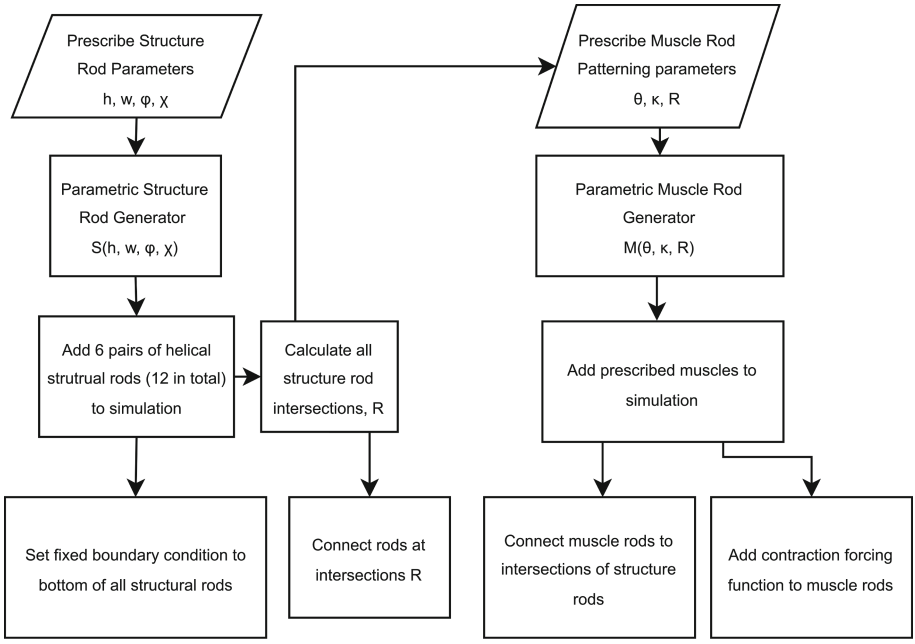

Fig. 2. High-level overview of muscle-actuated lattice generation pipeline. The user defines the height h and width w of the bulk lattice structure and the phase offset ϕ of a given rod, as well as its chirality χ (*e.g.* clockwise or counterclockwise). These parameters are fed into the Parametric Structure Rod Generator, which outputs 6 pairs of helical rods. The bottoms of these rods have a fixed boundary condition applied to them. Additionally, the location of intersections between interlaced structure rods R is calculated. These intersections act as both the location for connections between structure rods and locations where muscle rods can connect to the bulk structure. Muscle parameters are defined by the muscles orientation θ (*e.g.* vertical or horizontal), the on/off state of the muscle κ, and the locations R of the muscles on the lattice, which is the same R used to define structure rod connections.

to be patterned on the structure. Muscle rod (Fig. 1, red rods) orientation θ and on/off state κ are also prescribed by the user. These three muscle patterning parameters are then input into the Parametric Muscle Rod Generator, which outputs all prescribed muscles onto the correct location and with correct orientation on the lattice. Then, the muscles are programmatically connected to the structural rod elements at the appropriate intersections.

The geometry of the structural rods is defined by the parametric functions,

$$x(t) = \frac{d}{2}\cos(\frac{2\pi t}{T} + \phi)$$

$$y(t) = \frac{d}{2}\sin(\frac{2\pi t}{T} + \phi)$$

$$z(t) = t$$

for $0 < t < h$, where $x(t)$, $y(t)$, and $z(t)$ correspond to the x, y, and z positions for each node of a structural rod. h refers to the total height of the structure, and t is the parameterized variable. Variables d and T correspond to the diameter of the lattice and the period of the sinusoid defining individual structure rods, respectively. To achieve the desired closed lattice morphology, an additional geometrical constraint $T = \frac{3}{2}h$ was enforced. The numerical values of all the parameters that govern the simulation are detailed in Table 1.

Table 1. Parameters that define the lattice worm morphology, material model, and simulation variables.

Simulation Parameters	Value & Units
Structural rod elements	40 elements
Muscle rod elements	2 elements
Lattice height (h)	100 mm
Lattice diameter (d)	75 mm
Damping coefficient	30 mNs/m
Contraction magnitude	500 mN
Muscle Young's modulus	25 kPa
Structure Young's modulus	70 kPa
Muscle Poission's ratio	0.5
Structure Poission's ratio	0.5
Structure density	1070 kg/m^3
Muscle density	1060 kg/m^3
Structure rod radius	5 mm
Muscle rod radius	5 mm
Connection stiffness (k)	100 mN/mm

The parameters that define the lattice worm morphology were selected to create a tubular, closed lattice worm with element nodes located at structure rod intersections. The density and Young's modulus for the structure rods are an approximation taken from the Smooth-On website for their Ecoflex 00-30 product [17]. The muscle material model is the same as that in work detailed by Pagan-Diaz and colleagues [11]. Simulation variables (e.g., damping coefficient, connection stiffness) were hand-tuned to ensure simulation stability.

2.2 Quantifying Lattice Deformations

To assess our pipeline, we used it to generate three lattice models with *i)* circumferential, *ii)* symmetric longitudinal, and *iii)* asymmetric longitudinal and circumferential muscles. These muscle patterns give rise to the three motion

primitives (*i.e.* extension, compression, and bending) that, when properly coordinated, can facilitate multifunctional behaviors such as locomotion and peristalsis. We evaluated the ability of the pipeline to successfully generate these lattice-based biohybrid robots in which all structural elements and muscles successfully attach and interact. During simulation, the muscles applied force to the lattice, and lattice extension and compression deformations were quantified by calculating how much the bulk structure of the lattice had deformed along its longitudinal axis. This deformation was calculated by finding the average change in the z position of the top of the lattice using the formula:

$$\frac{1}{n}\sum_{r=1}^{n}(z_{r,f} - z_{r,i})$$

where n is 12 for the number of structural rods, $z_{r,f}$ and $z_{r,i}$ are the initial and final z positions, respectively, of the top-most node of a given rod r. From this calculation, we get the average displacement of the top of the structure, which serves as a metric for the deformation of the structure as a whole. This displacement is divided by the height h of the structure to calculate the strain.

To quantify bending due to asymmetric longitudinal and circumferential actuation, we first created a plane that spanned the top of the lattice defined by three structure rod end nodes. The normal vector $\mathbf{n}$ of this plane, which initially pointed vertically, bends away from vertical due to the lattice's bulk deformation due to asymmetric actuation. To calculate the final angle change of that normal vector (*i.e.* the angle that the lattice bent), we used the following formula:

$$\theta = \cos^{-1}\left[\frac{\mathbf{n_i} \cdot \mathbf{n_f}}{|\mathbf{n_i}||\mathbf{n_f}|}\right]$$

where θ is the angle between the planes, $\mathbf{n_i}$ and $\mathbf{n_f}$ are the normal vectors that define where the lattice is pointing before and after deformation, respectively, and $|\mathbf{n_i}|$ and $|\mathbf{n_f}|$ are the magnitudes of those normal vectors. The nature of the lattice worm's deflection leads to the surface formed by the distal nodes of the lattice being non-planar due to its asymmetric deformation. The authors have incorporated this non-planar deformation into our calculations, and have selected coplanar nodes that are representative of the distatal end of the robot.

For all actuation modalities, if any of the structural rod intersections or muscle rod connections have not been successfully defined by the parametric pipeline, the lattice would become unstable and disintegrate. In addition to assessing the deformation of each lattice quantitatively, each lattice simulation result was visually inspected to ensure that no rods were buckled or detached after muscle actuation.

2.3 Simulation Hardware

All simulations in this study were executed on a local computer workstation running Ubuntu 22.04, equipped with an Nvidia GeForce RTX 2080 SUPER

graphics card and an AMD Ryzen 7 3700 8-core Processor, offering 2 threads per core. It is important to note that the version of PyElastica used in this work is not easily parallelizable, as it runs on a single CPU thread. Consequently, CPU clock speed becomes the most relevant hardware metric for enhancing simulation performance.

Despite this limitation, it is still possible to run multiple PyElastica simulations simultaneously on the same machine. In our case, we were able to successfully run up to 15 concurrent simulations, reserving one thread for maintaining operating system stability. This approach effectively increases the overall throughput of simulation runs, allowing for faster evaluation of different design parameters and system configurations. "These simulation's wall time for a given actuation condition was very consistent, always under one minute on the hardware used in this work.

3 Results and Discussion

Using our parametric lattice-robot design pipeline, we were successfully able to generate biohybrid structure models with user-specified geometry and muscle placements. All structure-structure intersection connections and muscle-structure connections translated successfully into PyElastica. Additionally, we were able to successfully demonstrate all three desired motion primitives of extension, contraction, and bending (Fig. 3). A quantitative assessment of these deformational modes is detailed in Table 2. Each mode deforms as expected, with no unexpected actuation asymmetries indicating muscle rods have not been properly connected to the structure.

Table 2. Quantified performance of three deformational modes demonstrated. Videos of each deformation can be found in supplementary materials.

Motion	Value & Unit	Wall Time (s)	Final Sim. Time (s)	Timestep (ms)
Extension	49.73 % strain	25.90	0.5	0.2
Compression	−30.59 % strain	29.27	0.3	0.1
Bending	48.60 °	44.31	0.5	0.1

These preliminary results support the prospect of a lattice-based biohybrid architecture being capable of multifunctional behavior and successfully demonstrate our parametric model definition pipeline. While these deformational modes (*i.e.* extension, compression, bending) do not accomplish any functional task when executed alone, we believe that being able to extend, compress, and bend are the motion primitives necessary for more complex behaviors in lattice-based biohybrid robots, such as locomotion and esophageal-inspired peristalsis. The speed of these simulations, all under 45 s, provides the computational efficiency needed for a biohybrid design robot tool. The strains the bulk structure

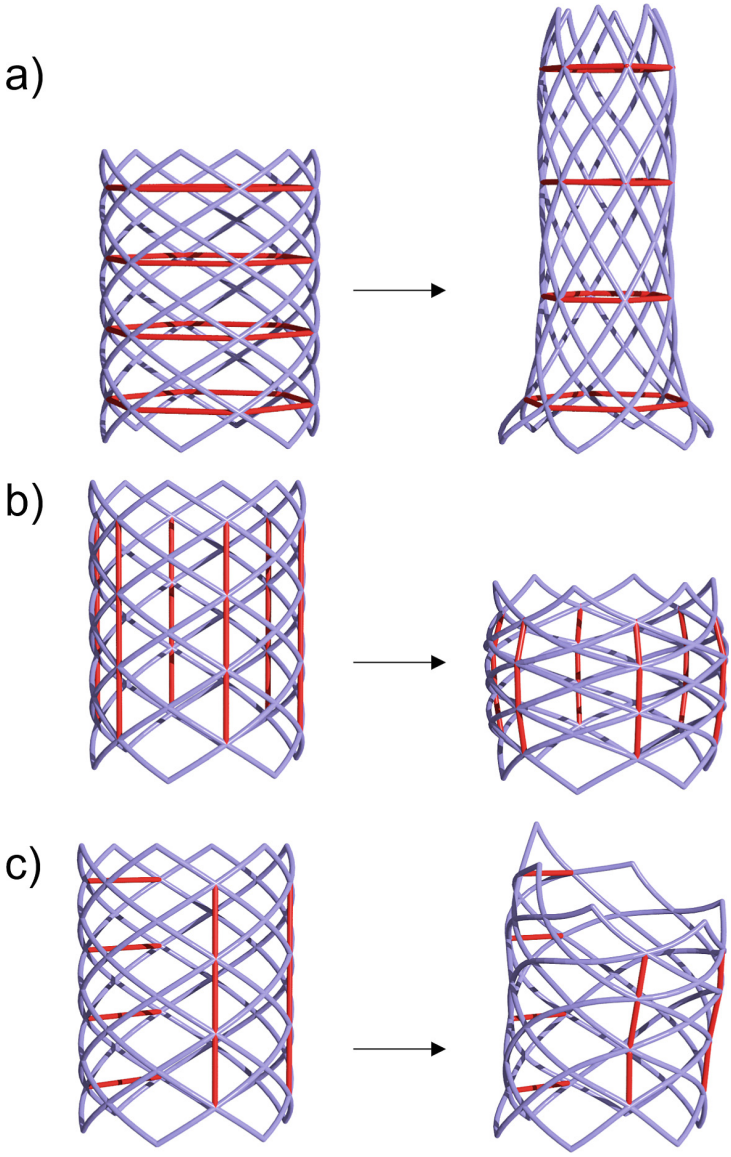

Fig. 3. a) Extension of the lattice is achieved by actuating circumferentially patterned muscles. **b)** Compression is achieved by actuating longitudinally patterned muscles. **c)** Bending is achieved by actuating circumferentially patterned muscles on the posterior of the lattice and actuating longitudinally on the anterior of the lattice. Videos of extension, compression, and bending can be found in the supplementary materials.

was able to undergo demonstrate that the simulation could generate large deformations required for future functional behavior.

4 Conclusions

This work presents a pilot study towards the design and simulation of multi-degree-of-freedom, functional lattice-based biohybrid robots. We present a parametric tool for computer-aided modeling of lattice-based structures using PyElastica and demonstrate the success of this pipeline by simulating three motion primitives *i.e.* extension, compression, and bending. These preliminary results indicate that careful coordination of actuators on this lattice architecture may enable more functional behaviors such as locomotion and peristalsis. Simulations for biohybrid robots, perhaps even more so than other mechanical systems, are crucial to the continued advancement of the field, as they allow researchers to perform countless iterations *in silico* before committing to the costly, time-intensive experiments needed to fabricate a functional living machine. Future work on this project will include verification of our computational predictions for biohybrid lattice robot behavior through physical, *in vitro* experimentation. Furthermore, the parametric model generation approach presented here makes simulating these lattice-based biohybrid robots more accessible, thereby reducing barriers to entry to biohybrid robot modeling.

Acknowledgements. This material is based on work supported by a Carnegie Mellon University (CMU) Dean's Fellowship and the National Science Foundation Graduate Research Fellowship Program under grant No. DGE1745016 and by the National Science Foundation CAREER award program (grant No. ECCS-2044785). The authors would also like to acknowledge OpenAI's ChatGPT for answering innumerable inane coding questions.

References

1. Aydin, O., et al.: Neuromuscular actuation of biohybrid motile bots. Proc. Natl. Acad. Sci. **116**(40), 19841–19847 (2019). https://doi.org/10.1073/pnas.1907051116
2. Cvetkovic, C., et al.: Three-dimensionally printed biological machines powered by skeletal muscle. Proc. Natl. Acad. Sci. U.S.A. **111**(28), 10125–10130 (2014). https://doi.org/10.1073/pnas.1401577111
3. Daltorio, K.A., Boxerbaum, A.S., Horchler, A.D., Shaw, K.M., Chiel, H.J., Quinn, R.D.: Efficient worm-like locomotion: slip and control of soft-bodied peristaltic robots. Bioinspiration Biomimetics **8**(3), 035003 (2013). https://doi.org/10.1088/1748-3182/8/3/035003
4. Gazzola, M., Dudte, L.H., McCormick, A.G., Mahadevan, L.: Forward and inverse problems in the mechanics of soft filaments. R. Soc. Open Sci. **5**(6), 171628 (2018). https://doi.org/10.1098/rsos.171628
5. Guix, M., et al.: Biohybrid soft robots with self-stimulating skeletons. Sci. Robot. **6**(53), eabe7577 (2021). https://doi.org/10.1126/scirobotics.abe7577
6. Herr, H., Dennis, R.G.: A swimming robot actuated by living muscle tissue. J. NeuroEng. Rehabil. **1**, 1–9 (2004). https://doi.org/10.1186/1743-0003-1-6
7. Kim, Y., et al.: Remote control of muscle-driven miniature robots with battery-free wireless optoelectronics. Sci. Robot. **8**, eadd1053 (2023). https://doi.org/10.1126/scirobotics.add1053

8. Lee, K.Y., et al.: An autonomously swimming biohybrid fish designed with human cardiac biophysics. Science **375**(6581), 639–647 (2022). https://doi.org/10.1126/science.abh0474
9. Morimoto, Y., Onoe, H., Takeuchi, S.: Biohybrid robot powered by an antagonistic pair of skeletal muscle tissues. Sci. Robot. **3**, 4440 (2018). https://doi.org/10.1126/scirobotics.aat4440
10. Naughton, N., Sun, J., Tekinalp, A., Parthasarathy, T., Chowdhary, G., Gazzola, M.: Elastica: a compliant mechanics environment for soft robotic control. IEEE Robot. Autom. Lett. **6**(2), 3389–3396 (2021). https://doi.org/10.1109/LRA.2021.3063698
11. Pagan-Diaz, G.J., et al.: Simulation and fabrication of stronger, larger, and faster walking biohybrid machines. Adv. Funct. Mater. **28**(23), 1801145 (2018). https://doi.org/10.1002/adfm.201801145
12. Park, S.J., et al.: Phototactic guidance of a tissue-engineered soft-robotic ray. Science **353**(6295), 158–162 (2016). https://doi.org/10.1126/science.aaf4292
13. Raman, R., et al.: Optogenetic skeletal muscle-powered adaptive biological machines. Proc. Natl. Acad. Sci. **113**(13), 3497–3502 (2016). https://doi.org/10.1073/pnas.1516139113
14. Ricotti, L., et al.: Biohybrid actuators for robotics: a review of devices actuated by living cells. Sci. Robot. **2**(12), eaaq0495 (2017). https://doi.org/10.1126/scirobotics.aaq0495
15. Schaffer, S., Lee, J.S., Beni, L., Webster-Wood, V.A.: A computational approach for contactless muscle force and strain estimations in distributed actuation biohybrid mesh constructs. Biomimetic Biohybrid Syst. **13548**, 140–151 (2022). https://doi.org/10.1007/978-3-031-20470-8_15
16. Seok, S., Onal, C.D., Cho, K.J., Wood, R.J., Rus, D., Kim, S.: Meshworm: a peristaltic soft robot with antagonistic nickel titanium coil actuators. IEEE/ASME Trans. Mechatron. **18**(5), 1485–1497 (2013). https://doi.org/10.1109/TMECH.2012.2204070
17. Smooth-On: Ecoflex 00–30 (2023). https://www.smooth-on.com/products/ecoflex-00-30/, Accessed 15 Jan 2023
18. Wang, J., et al.: Computationally assisted design and selection of maneuverable biological walking machines. Adv. Intell. Syst. **3**, 2000237 (2021). https://doi.org/10.1002/aisy.202000237
19. Webster, V.A., et al.: Aplysia Californica as a novel source of material for biohybrid robots and organic machines. Biomimetic Biohybrid Syst. (2016). https://doi.org/10.1007/978-3-319-42417-0
20. Webster, V.A., Nieto, S.G., Grosberg, A., Akkus, O., Chiel, H.J., Quinn, R.D.: Simulating muscular thin films using thermal contraction capabilities in finite element analysis tools. J. Mech. Behav. Biomed. Mater. **63**, 326–336 (2016). https://doi.org/10.1016/j.jmbbm.2016.06.027
21. Webster-Wood, V.A., Akkus, O., Gurkan, U.A., Chiel, H.J., Quinn, R.D.: Organismal engineering: toward a robotic taxonomic key for devices using organic materials. Sci. Robot. **2**(12), 1–19 (2017). https://doi.org/10.1126/scirobotics.aap9281
22. Zhang, X., Chan, F.K., Parthasarathy, T., Gazzola, M.: Modeling and simulation of complex dynamic musculoskeletal architectures. Nat. Commun. **10**(1), 4825 (2019). https://doi.org/10.1038/s41467-019-12759-5