Contact-Rich Soft-Rigid Robots Inspired by Push Puppets

James M. Bern, Leonardo Zamora Yañez, Emily Sologuren and Daniela Rus

***Abstract*— Human-centric problem domains will require robots with an unprecedented mix of traits–combining the inherent safety and robustness of soft materials with the strength and precision of rigid systems. Foundational work is needed to understand how to design and simulate a new generation of hybrid, soft-rigid robots. We draw inspiration from the classic push puppet toy to design soft-rigid robot modules capable of transitioning between soft and rigid states. We show how multiple modules can be combined into exciting exemplar robots, such as a shape-shifting soft-rigid arm. We also show how to build differentiable finite element-based simulations of these robots, including state-of-the-art soft body contact and motorized revolute joints.**

I. Introduction

Robots have yet to achieve their potential to help in human-centric problem domains, such as homes, hospitals, and nursing homes. We posit that these areas require robots with an unprecedented blend of traits. E.g. the ideal nursing home robot will have high strength to help residents out of bed; high precision to help them put in their hearing aids; and high robustness to help them eat. Above all else, it must be safe. We seek to develop robots that fuse the strength and precision of rigid systems [1], [2] with the safety and robustness of soft materials [3]–[5].

We are inspired by classic push puppet toys [6], which collapse when their internal cables are loosened. In this work, we design and simulate robots that dramatically stiffen when their cables are contracted. This enables them to transition between soft and rigid states. Related work in continuum soft robotics includes a robot that can telescope and bend using magnetic disks [7], a robot using origami patterns to similar ends [8], [9], a robot with an elastic backbone that controls its stiffness using rigid curvature-constraining rods [10], a crustacean-inspired soft-rigid actuator [11], cable-driven vertabrae held together by friction [12], and chain-like jamming [13]. Our approach has the benefit of mechanical simplicity and generality–extending to morphologies beyond continuum robot arms.

In addition to design and fabrication, we explore how to model push puppet-inspired soft-rigid robots. Related work includes various modeling approaches for telescoping and bending continuum robots [9], [14], involving piecewise constant curvature (PCC) and physically-based rod models. We employ a soft robot simulator based on the finite element method FEM) [15], the same general approach used in simulators like SOFA [16]. Our use of a volumetric FEM-based simulator enables us to go beyond continuum robots, and model a rich family of push puppet-inspired designs.

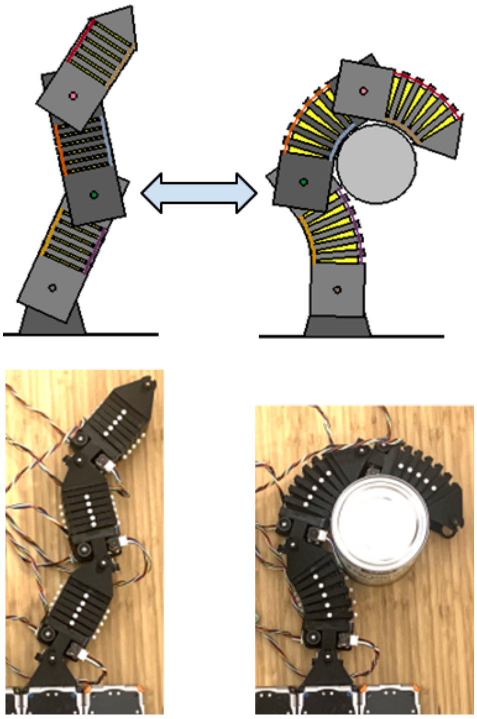

Fig. 1. We draw inspiration from push puppets to design and simulate soft-rigid robots like this arm, which can transition between rigid (left) and soft (right) actuation modes at will. Our vision is to create robots that achieve the best of both worlds, and fuse the strength and precision of rigid system with the compliance and inherent safety of soft materials.

We adapt our simulator to model push puppet-inspired soft rigid robots by adding contact (which occurs when a module stiffens) and motorized revolute joints (to join multiple modules into larger robots). Soft body contact has a rich body of literature in computer graphics, with extremely exciting methods emerging in the past couple years [17]–[19]. In this work, we apply a 2D version of Li et al.'s Incremental Potential Contact (IPC) [18]. Other examples of contact modeling in soft robotics include for manipulation [20] and growing robots [21]. Motorized revolute joints are a standard feature of rigid body engines like Box2D [22] or Bullet [23]. However, soft body simulators like ours [24] or Vega [25] typically have no such functionality. One approach would be to develop a new simulator coupling soft bodies and rigid bodies, like Geilinger et al.'s ADD [26] or the future work of Ferguson et al.'s Intersection-free Rigid Body Dynamics [27]. Instead, we model motorized revolute joints within the framework of a soft body simulator. This approach is simple to implement and compatible out-of-the-box with model-based control methods like Soft IK [24], [28].

Computer Science and Artificial Intelligence Laboratory, MIT. jamesmbern@gmail.com, rus@csail.mit.edu

We contribute:

- Design and characterization of a push puppet-inspired robot module, which uses cables to transform between soft and rigid states.
- An approach to simulating push puppet-inspired soft-rigid robots with contact and revolute joints, compatible with existing model-based control algorithms.
- Multiple simulated and fabricated exemplar robots, including a shape-shifting soft-rigid arm.

II. Key Idea: Cable-Driven Shape-Shifting Using Softness and Rigidity

The classic push puppet toy can dramatically transition between stiffer and softer modes (see Figure 2). It is made from multiple disconnected rigid pieces, which are held together by taught internal cables. When the user depresses the base of the toy–compressing the spring hidden within–the cables slacken and the toy "falls to pieces."

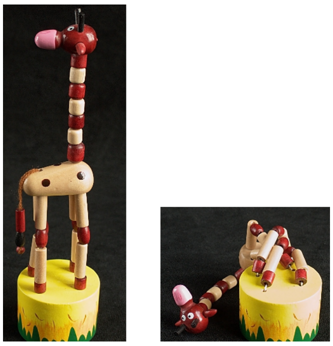

Fig. 2. Depressing the base of a thumb puppet slackens internal cables, which transitions the puppet from a stiffer state (left) to a softer, collapsed state (right). We propose a similar approach to dramatically modulate the stiffness of soft-rigid robots. *Photo: Wikipedia Commons*

We adapt the push puppet into the robot module in Figure 3 by adding soft material between the rigid segments. When its cables are loose the module acts soft. When we co-contract the cables, the soft material is compressed inside the rigid segments, and the module acts rigid. In summary: a classic push puppet toy is naturally rigid and soften when pressed by a human; a proposed push puppet-inspired robot is naturally soft and can rigidify when actuated by motors.

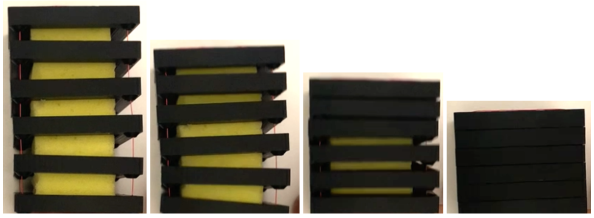

Fig. 3. Snapshots of a push puppet-inspired soft-rigid module transitioning between from its soft mode to its rigid modes. The module is made from rigid plastic (black) and soft foam (yellow). When the cables running along the length of the module are relaxed, the foam can bend freely and so the module acts soft. When the cables are contracted, the foam becomes hidden within the plastic, creating the effect of a single rigid block.

Furthermore, we can compose modules to produce larger robots like the soft-rigid arm in Figure 1. Note that the cables play an important role in both the arm's soft mode and its rigid mode. For the soft mode, cable actuation bends the modules and moves the robot. For the rigid mode, cable tension maintains the modules' rigidity. Finally, hybrid modes are possible, either with some modules soft and other modules rigid, or with any given module in a partially-rigidified state.

III. Modeling and Simulation

We build upon the FEM-based soft robot simulator presented in [15], which can handle soft bodies, cables, basic contact with a ground plane, and dynamics. Given control inputs $\boldsymbol{u}_k$, and previous mesh configurations $\boldsymbol{x}_{k-1}, \boldsymbol{x}_{k-2}$, the simulator solves for a mesh configuration $\boldsymbol{x}_k$ satisfying Newton's second law

$$\boldsymbol{F}_k = \boldsymbol{M}\boldsymbol{a}_k, \tag{1}$$

with nodal accelerations $\boldsymbol{a}_k$ discretized according to implicit Euler. This simulator solves Equation (1) via minimization

$$\boldsymbol{x}_k(\boldsymbol{u}_k) = \arg\min_{\boldsymbol{x}_k} \left(U(\boldsymbol{u}_k, \boldsymbol{x}_k) + \frac{h^2}{2}\boldsymbol{a}_k^T \mathbf{M}\boldsymbol{a}_k \right), \tag{2}$$

where U is the potential energy stored in the system, h is the timestep and $\boldsymbol{M}$ is the mass matrix.

To make this simulator capable of handling push puppet-inspired hybrid soft-rigid robots, we need to add the ability to model 1) soft and rigid regions, 2) arbitrary contact, and 3) motorized joints.

A. Soft and Rigid Regions

To model soft and rigid regions we specify the Young's modulus per element. Note that we use automated routines to mesh our designs, specifically Triangle for 2D meshes [29] and TetGen for 3D meshes [30]. These libraries can take region markers as part of their input, enabling automatic assignment of material properties (see Figure 4).

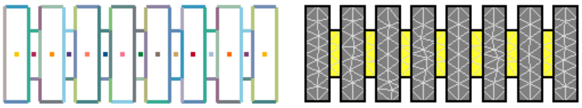

Fig. 4. Left: sample input to Triangle, consisting of boundary segments and region markers (dots). Right: the output finite element mesh, with stiffer elements in gray, and softer elements in yellow.

We employ a Neo-Hookean material model. Young's modulus helps determine each element's energy density

$$\Psi(\boldsymbol{x}, \boldsymbol{X}) = \frac{\mu}{2}\mathrm{tr}(\boldsymbol{\mathcal{F}}^T\boldsymbol{\mathcal{F}} - \boldsymbol{I}) - \mu \ln J + \frac{\lambda}{2}(\ln J)^2, \tag{3}$$

where $\boldsymbol{X}$ is the rest shape, $\boldsymbol{\mathcal{F}}$ is the deformation gradient, $J = \det \boldsymbol{\mathcal{F}}$, and λ and μ are material parameters that can be expressed in terms of Young's modulus and Poisson's ratio. For linear elements we multiply each element's energy density by its rest volume to find its contribution to U.

B. Contact

We implement a 2D version of Li et al.'s state of the art Incremental Potential Contact (IPC) [18], [19]. We neglect friction in our implementation for simplicity, and rely on the cables to resist tangential motion in the robot's rigid mode.

IPC is based on the unsigned distance d between pairs of primitives (in 2D, just points and edges). The two crucial ingredients of IPC are 1) a $\mathcal{C}^2$ barrier function that goes to infinity as $d \to 0$, and 2) a contact-aware line search that uses continuous collision detection (CCD) to ensure all descent steps are penetration-free. The specific barrier employed by Li and colleagues is $b(d^2, \hat{d}^2)$, where

$$b(d, \hat{d}) = \begin{cases} -(d - \hat{d})^2 ln\left(\frac{d}{\hat{d}}\right), & 0 < d < \hat{d} \\ 0 & d \geq \hat{d}, \end{cases} \tag{4}$$

and $\hat{d}$ is the *target distance*, i.e. the distance at which the minimizer starts "seeing" the collision [18].

To incorporate (a stripped down, simplified version of) IPC into our simulator, we 1) add the contributions of the barrier terms to our energy and derivative calculations, and 2) use a CCD line search when solving Equation (2).

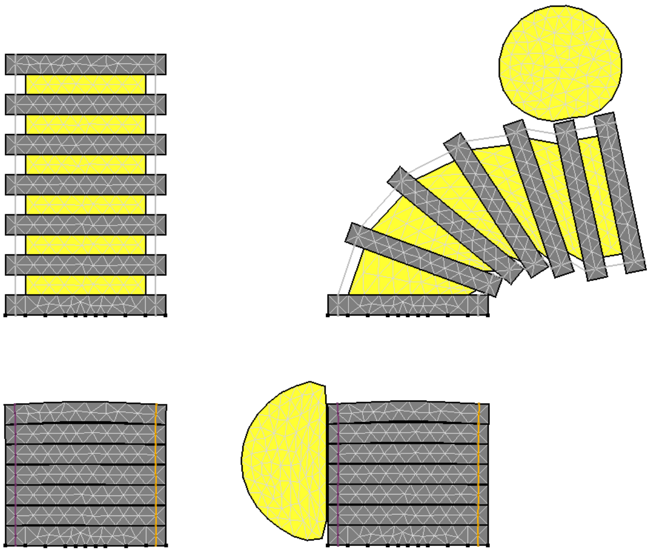

Fig. 5. Tests of a soft ball impacting a push puppet-inspired module in its soft mode (top) and rigid mode (bottom). Notice that there are rich, intermittent contact events occurring between the ball and the module, and also between the module's various rigid segments.

C. Motorized Joints

We use zero-length springs to model motorized revolute joints, as outlined in Figure 6. To start, consider connecting bodies A and B of a multi-body mesh with a joint. Given the mesh in its rest configuration $\boldsymbol{X}$, we specify the position of the joint in world coordinates. From this, we determine the joint's position in the barycentric coordinates on each of the two bodies. Now, for a deformed shape of the mesh $\boldsymbol{x}$, we can calculate the position of the joint on each body in world coordinates, which we denote $\boldsymbol{s}_A(\boldsymbol{x})$ and $\boldsymbol{s}_B(\boldsymbol{x})$.

To model a basic (unmotorized) pin joint, we add energy

$$\frac{1}{2}||\boldsymbol{s}_A - \boldsymbol{s}_B||^2, \tag{5}$$

which is a zero-length spring that "pins" $\boldsymbol{s}_A$ to $\boldsymbol{s}_B$.

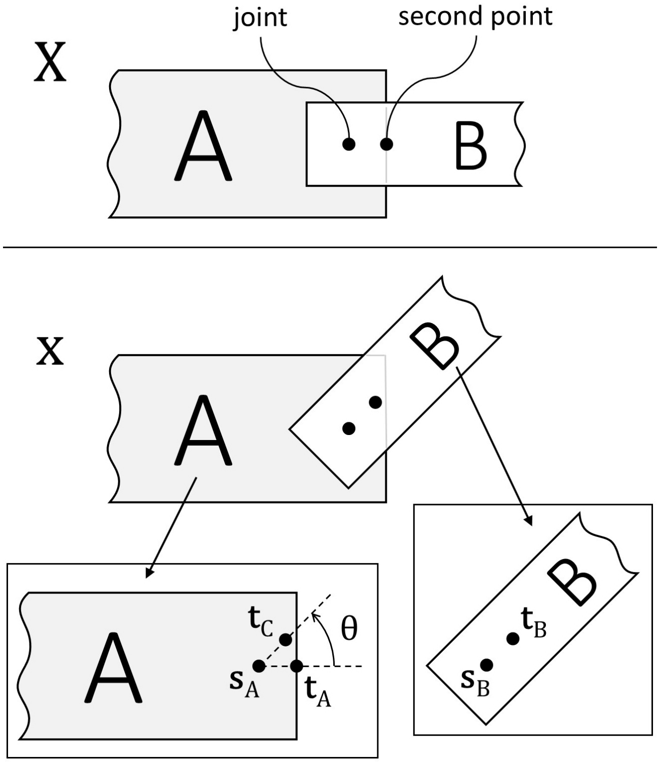

Fig. 6. Our scheme for modeling a motorized revolute joint connecting bodies A and B. To define the position of the joint, we pin $\boldsymbol{s}_A(\boldsymbol{x})$ to $\boldsymbol{s}_B(\boldsymbol{x})$. To define the motor angle, we pin $\boldsymbol{t}_C(\boldsymbol{x}, \theta)$ to $\boldsymbol{t}_B(\boldsymbol{x})$.

To motorize the joint, we first pick out the position of a second point in world coordinates. We require this second point to lie on both bodies when the mesh is in its rest configuration $\boldsymbol{X}$. Again we find this second point's position in the barycentric coordinates of each body, and for a deformed mesh shape $\boldsymbol{x}$ we can calculate the position of the second point on each body, which we denote $\boldsymbol{t}_A(\boldsymbol{x})$ and $\boldsymbol{t}_B(\boldsymbol{x})$. To model the motor, we pin $\boldsymbol{t}_B$ to the result of rotating $\boldsymbol{t}_A$ about $\boldsymbol{s}_A$ by motor angle θ, which we call

$$\boldsymbol{t}_C(\boldsymbol{x}, \theta) = \boldsymbol{R}_\theta(\boldsymbol{t}_A - \boldsymbol{s}_A) + \boldsymbol{t}_A, \tag{6}$$

where $\boldsymbol{R}_\theta$ is a rotation matrix. This amounts to adding energy

$$\frac{1}{2}||\boldsymbol{t}_B - \boldsymbol{t}_C||^2. \tag{7}$$

We finish with the special case of connecting a single body B to a fixed position in the world, as is done for the proximal joint of the soft-rigid arm in Section V-A. This is done by pinning to the rest configuration. Define $\boldsymbol{S}_B(\boldsymbol{X})$ and $\boldsymbol{T}_B(\boldsymbol{X})$ to be the positions of the joint and the second point in the mesh's rest configuration. Pin joint position $\boldsymbol{s}_B$ to $\boldsymbol{S}_B$, and pin second point position $\boldsymbol{t}_B$ to the result of rotating $\boldsymbol{T}_B$ about $\boldsymbol{S}_B$ by θ.

IV. Fabrication and Characterization

We present an accessible method for fabricating push puppet-inspired soft rigid modules, for example the soft-rigid arm module in Figure 7. Each module consists of a 30mm × 30mm × 55mm polyurethane foam core surrounded by ABS plastic segments. We can use pins to define the position of each plastic segment along the foam core. One side of each plastic segment has four hollow studs, and the other side has four corresponding holes. at its corners. This geometry helps locate and rigidify the module as it is compressed, as shown in Figure 8. We route cables through the hollow studs. We can use motorized brackets to compose modules into larger robots.

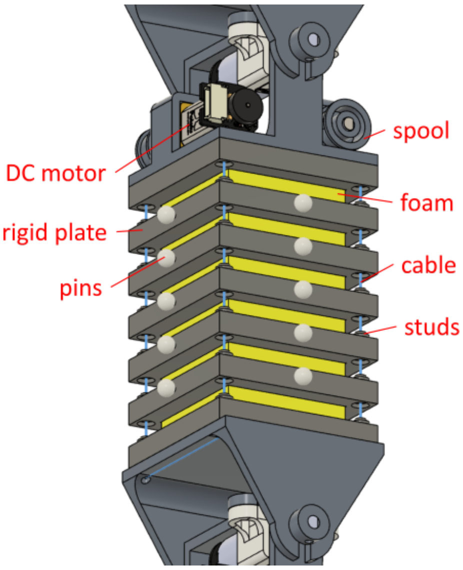

Fig. 7. CAD model of a soft-rigid module used in our soft-rigid arm.

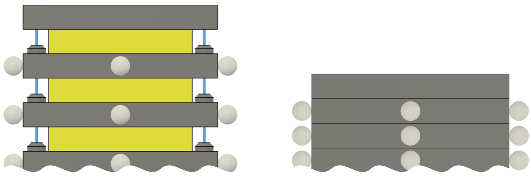

Fig. 8. Each rigid segment has a matching set of studs and holes. This geometry helps locate and rigidify the module as the cables running through the studs are contracted.

We characterize the effect of cable retraction on the effective stiffness of a push puppet-inspired soft-rigid module (see Figure 9). To conduct each trial, the cable was first set to some known length and then a known mass was placed at the end of the beam to induce a load. This was repeated for four discrete masses with three trials for each load. The displacement of the end of the module was measured and used to calculate the effective stiffness under each condition. The cable length was then decreased by three discrete amounts and the same trials were repeated. The module was given three minutes between trials to decompress and return to its natural state to reduce the memory effect on the foam.

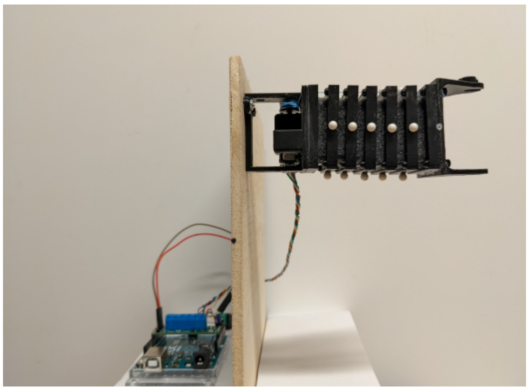

Fig. 9. Experimental setup used to characterize the stiffness of the push puppet-inspired soft-rigid module.

Figure 10 illustrates the effective stiffness' across all three trials for each discrete load of force applied. To gain a better understanding of the effects of cable retraction on rigidity, we applied a polynomial fit on the points of data. As can be seen, we notice how increasingly retracting the supporting length generally increases the effective stiffness of the beam composite, with an especially noticeable increase in stiffness for the trial that retracts 17.68 mm from the supporting length. Another interesting observation is how the trends across all trials are nonlinear in behavior; as the load force increases, the effective stiffness starts to plateau regardless of the length retraction. This highlights the complex interaction of foam, contact between the rigid segments, and taught cables. These results confirm the dramatic change in stiffness we are able to achieve with the push puppet-inspired robots as they transition between soft and rigid states. Additionally, they illustrate the potential to use push puppet-inspired robots to *continuously* control stiffness, potentially leveraging model-based stiffness control strategies like those in [28].

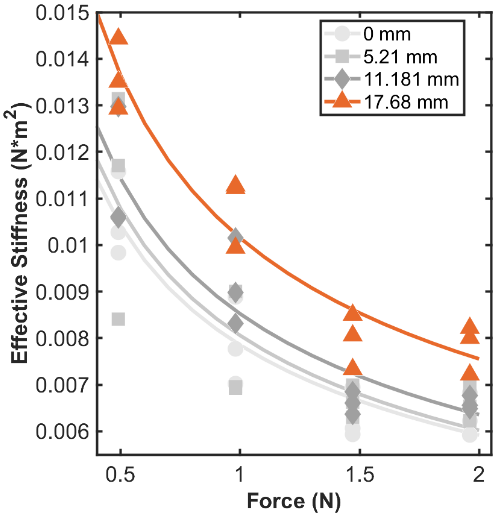

Fig. 10. Force against Effective Stiffness for all three trials testing the four cable displacements. Force was calculated from mass-load and effective stiffness was calculated for each displacement.

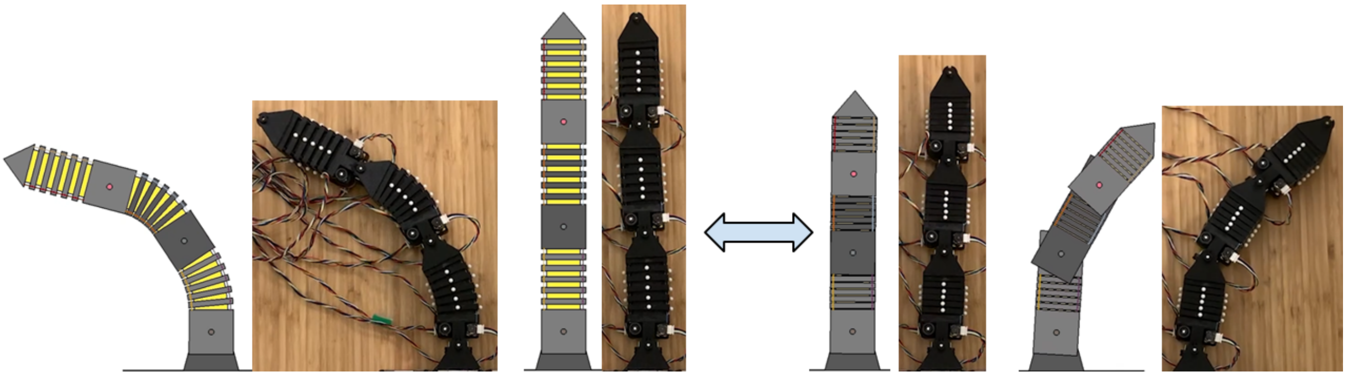

Fig. 11. Simulation and fabrication of a soft-rigid arm composed of push puppet-inspired modules joined at motorized joints. We can either use in the arm in a compliant *soft mode* (left), where the arm moves by using the cables to bend the modules, or in a articulated *rigid mode* (right), where the cables keep the modules rigid, and the arm moves using the motors at its revolute joints.

V. Examples

We present several simulated and fabricated examples that compose and extend the capabilities of push puppet-inspired soft rigid modules. Please see our Supplementary Video.

A. Arm

We compose multiple modules together with revolute joints to create a three-link hybrid soft-rigid arm, capable of both soft and rigid actuation modes, shown in Figure 11. This robot has three motors per link, two of which contract the cables running along either side, and one of which rotates the link about the joint at the link's base. In the robot's *soft mode*, the cables are used to bend the links. In the robot's *rigid mode*, the cables hold the modules stiffness, and the revolute joints move the robot as a traditional rigid arm.

1) Soft-Rigid Control: We explore the robot's workspace as shown in our Supplementary Video. For the soft mode we keep all joint angles at zero, and use a sinusoidal control sequence to contract the cables. To transition to the rigid mode, we contract all cables by a nominal length $\bar{u}$. For the rigid mode, we sweep out a sinusoidal sequence of joint angles. We can then relax the cables back to zero to come back to the soft mode. In the soft mode, we can execute *hybrid motions*, which we illustrate by sending sinusoidal targets to all motors. We note that hybrid motions dramatically increase the size of the workspace.

B. 2D Spiraling Modules

We can create modules that can either bend or straighten as they stiffen, shown in Figures 12 and 13. These modules could serve, for example, as the fingers of a soft-gripper.

C. Sheet

We extend our push puppet-inspired module into a shape-shifting sheet in Figure 14. Note that this designs contains the same alternating pattern as the original push puppet module.

D. 3D Twisting Module

We route cables helically through the rigid segments to create a 3D modules that twists as is compresses in Figure 15.

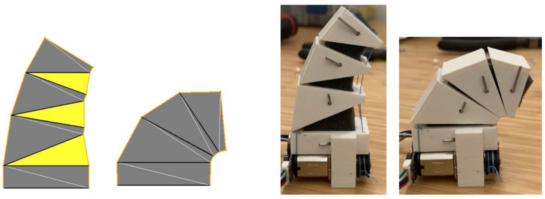

Fig. 12. Simulated and fabricated concepts for a 2D finger-like module that compresses triangular regions of foam to bend as it becomes rigid.

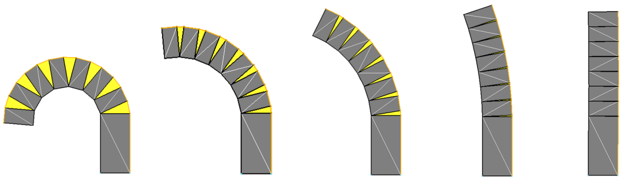

Fig. 13. Simulated concept of a 2D module that compresses triangular regions of foam to straighten as it becomes rigid.

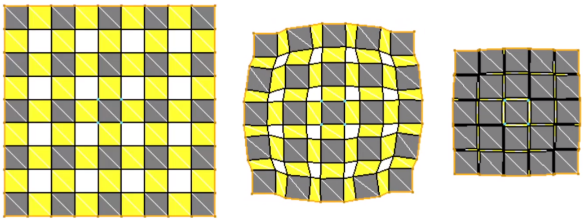

Fig. 14. Simulated concept of a soft-rigid sheet inspired by [31], [32].

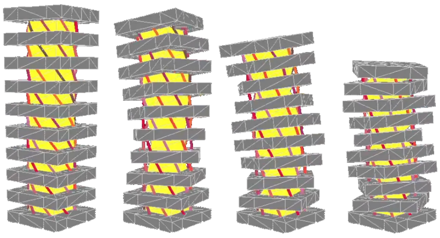

Fig. 15. Simulated concept of a 3D module that uses helically-routed cables to twist as it becomes rigid, reminiscent of [33].

VI. Discussion and Future Work

A. Hybrid Control Approaches

The way we simulate the robots in this paper is compatible out of the box with model-based control methods like Soft IK [28]. However, particularly as we explore increasingly complex robot designs, these methods may become too slow for real-time use. It would be interesting to exploit the structure of the push puppet-inspired robots to increase computational efficiency. One way to do this would be to explore hybrid approaches to control, for example bootstrapping the Soft IK problem off of the solution to a typical rigid inverse kinematics problem, in a similar vein as [34].

B. Computational Design of Stiffness and Shape Change

This work is just an initial exploration of what is possible when we design robots using a pattern or soft and rigid materials. Already we can achieve stiffness modulation for various morphologies, in concert with programmable motions like twisting. Other applications are certainly possible. To open the door to more sophisticated mechanisms, we can draw on the field of computational design to enable authorship of geometrically complex parts. An exciting goal for future work would be to develop a framework to produce robots with fine-grained programmability of stiffness and shape change, leveraging geometric complexity in a similar vein to work on metamaterial mechanisms [35].

C. Decoupling Stiffness Modulation from Length Change

Our proposed approach to stiffness modulation is currently coupled to length change. Because the approach works by hiding soft material within rigid material, parts naturally become smaller as they are stiffened. Future work on the mechanical design side should be done to remove this coupling. This would give designers greater freedom, and also likely simplify the control problem.

VII. Conclusion

Human-centric problem domains call for a new breed of robots, that marry the safety and robustness of softness with the strength and precision of rigidity. In this work we present an exploration inspired by classic push puppet toys, which leverage cable-actuation to dramatically alter their stiffness. We adapted the push puppet design for use in robotics, and show how the robot's cables can function both as soft bending actuators and to maintain rigidity. This observation enables the creation of new soft-rigid robots like a shape-shifting soft-rigid arm, capable of acting both like a traditional rigid robot, and a continuum soft robot. We also showed how to apply differentiable, physically-based simulation to model these robots, drawing on state of the art soft body contact methods, as well as a technique for incorporating motorized revolute joints into a soft robot simulator. In the future it would be exciting to apply this simulator to problems like computational design, and intelligently leverage softness and rigidity to enable fine-grained programmability of stiffness and shape change.

Acknowledgment

We thank the NSF for support under Grant EFRI-1830901. We would also like to thank John Romanishin.

Appendix

A. Contact Derivatives

In 2D it is necessary only to prevent nodes from passing through edges. The unsigned distance between a node and an edge resolves to either the *point-point distance*

$$d_{PP} = |\boldsymbol{s}_a - \boldsymbol{s}_b|, \tag{8}$$

or the *point-edge distance*

$$d_{PE} = \frac{|(\boldsymbol{s}_a - \boldsymbol{s}_c) \times (\boldsymbol{s}_b - \boldsymbol{s}_c)|}{|\boldsymbol{s}_a - \boldsymbol{s}_b|}. \tag{9}$$

To help differentiate d_{PE} we rewrite Equation (9) as

$$d_{PE} = \frac{|\boldsymbol{u}^T \boldsymbol{M} \boldsymbol{v}|}{|\boldsymbol{w}|}, \tag{10}$$

where

$$\boldsymbol{M} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \text{ and } \begin{cases} \boldsymbol{u} = \boldsymbol{s}_a - \boldsymbol{s}_c \\ \boldsymbol{v} = \boldsymbol{s}_b - \boldsymbol{s}_c \\ \boldsymbol{w} = \boldsymbol{s}_a - \boldsymbol{s}_b. \end{cases}$$

Equation (10) is of the form

$$f(\boldsymbol{u}, \boldsymbol{v}) g(\boldsymbol{w}), \tag{11}$$

where $f(\boldsymbol{u}, \boldsymbol{v}) = |\boldsymbol{u}^T \boldsymbol{M} \boldsymbol{v}|$ and $g(\boldsymbol{w}) = 1/|\boldsymbol{w}|$. Analytic first and second derivatives of f and g can found via e.g. automatic differentiation on MatrixCalculus.org [36], [37].

References

[1] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, "Bigdog, the rough-terrain quadruped robot," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 10 822–10 825, 2008.

[2] L. Ghanbari, M. L. Rynes, J. Hu, D. S. Schulman, G. W. Johnson, M. Laroque, G. M. Shull, and S. B. Kodandaramaiah, "Craniobot: A computer numerical controlled robot for cranial microsurgeries," *Scientific reports*, vol. 9, no. 1, pp. 1–12, 2019.

[3] 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.

[4] B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus, "Robust proprioceptive grasping with a soft robot hand," *Autonomous Robots*, vol. 43, no. 3, pp. 681–696, 2019.

[5] U. Culha, J. Hughes, A. Rosendo, F. Giardina, and F. Iida, "Design principles for soft-rigid hybrid manipulators," in *Soft robotics: Trends, applications and challenges*. Springer, 2017, pp. 87–94.

[6] M. Meinrad, "Body with movable parts," U.S. Patent 2421279A, May 1947.

[7] T.-D. Nguyen and J. Burgner-Kahrs, "A tendon-driven continuum robot with extensible sections," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 2130–2135.

[8] Y. Xu, Q. Peyron, J. Kim, and J. Burgner-Kahrs, "Design of lightweight and extensible tendon-driven continuum robots using origami patterns," in *2021 IEEE 4th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2021, pp. 308–314.

[9] J. Santoso and C. D. Onal, "An origami continuum robot capable of precise motion through torsionally stiff body and smooth inverse kinematics," *Soft Robotics*, vol. 8, no. 4, pp. 371–386, 2021.

[10] B. Zhao, W. Zhang, Z. Zhang, X. Zhu, and K. Xu, "Continuum manipulator with redundant backbones and constrained bending curvature for continuously variable stiffness," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7492–7499.

[11] Y. Chen, S. Le, Q. C. Tan, O. Lau, F. Wan, and C. Song, "A reconfigurable hybrid actuator with rigid and soft components," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 58–63.

[12] P. M. Loschak, S. F. Burke, E. Zumbro, A. R. Forelli, and R. D. Howe, "A robotic system for actively stiffening flexible manipulators," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 216–221.

[13] Y. Jiang, D. Chen, C. Liu, and J. Li, "Chain-like granular jamming: a novel stiffness-programmable mechanism for soft robotics," *Soft robotics*, vol. 6, no. 1, pp. 118–132, 2019.

[14] M. T. Chikhaoui, S. Lilge, S. Kleinschmidt, and J. Burgner-Kahrs, "Comparison of modeling approaches for a tendon actuated continuum robot with three extensible segments," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 989–996, 2019.

[15] J. M. Bern, P. Banzet, R. Poranne, and S. Coros, "Trajectory optimization for cable-driven soft robot locomotion." in *Robotics: Science and Systems*, vol. 1, no. 3, 2019.

[16] C. Duriez and T. Bieze, "Soft robot modeling, simulation and control in real-time," in *Soft Robotics: Trends, Applications and Challenges*. Springer, 2017, pp. 103–109.

[17] M. Verschoor and A. C. Jalba, "Efficient and accurate collision response for elastically deformable models," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 2, pp. 1–20, 2019.

[18] M. Li, Z. Ferguson, T. Schneider, T. Langlois, D. Zorin, D. Panozzo, C. Jiang, and D. M. Kaufman, "Incremental potential contact: Intersection-and inversion-free, large-deformation dynamics," *ACM transactions on graphics*, 2020.

[19] M. Li, D. M. Kaufman, and C. Jiang, "Codimensional incremental potential contact," *arXiv preprint arXiv:2012.04457*, 2020.

[20] E. Coevoet, A. Escande, and C. Duriez, "Optimization-based inverse model of soft robots with contact handling," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1413–1419, 2017.

[21] J. D. Greer, L. H. Blumenschein, R. Alterovitz, E. W. Hawkes, and A. M. Okamura, "Robust navigation of a soft growing robot by exploiting contact with the environment," *The International Journal of Robotics Research*, vol. 39, no. 14, pp. 1724–1738, 2020.

[22] E. Catto, "Box2d: A 2d physics engine for games," *URL: http://www. box2d. org*, 2011.

[23] E. Coumans *et al.*, "Bullet physics library," *Open source: bulletphysics. org*, vol. 15, no. 49, p. 5, 2013.

[24] J. M. Bern, G. Kumagai, and S. Coros, "Fabrication, modeling, and control of plush robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3739–3746.

[25] F. S. Sin, D. Schroeder, and J. Barbič, "Vega: non-linear fem deformable object simulator," in *Computer Graphics Forum*, vol. 32, no. 1. Wiley Online Library, 2013, pp. 36–48.

[26] M. Geilinger, D. Hahn, J. Zehnder, M. Bächer, B. Thomaszewski, and S. Coros, "Add: analytically differentiable dynamics for multi-body systems with frictional contact," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, pp. 1–15, 2020.

[27] Z. Ferguson, M. Li, T. Schneider, F. Gil-Ureta, T. Langlois, C. Jiang, D. Zorin, D. M. Kaufman, and D. Panozzo, "Intersection-free rigid body dynamics," *ACM Transactions on Graphics*, vol. 40, no. 4, p. 183, 2021.

[28] J. M. Bern and D. Rus, "Soft ik with stiffness control," in *2021 IEEE 4th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2021, pp. 465–471.

[29] J. R. Shewchuk, "Triangle: Engineering a 2d quality mesh generator and delaunay triangulator," in *Workshop on Applied Computational Geometry*. Springer, 1996, pp. 203–222.

[30] H. Si, "Tetgen, a delaunay-based quality tetrahedral mesh generator," *ACM Transactions on Mathematical Software (TOMS)*, vol. 41, no. 2, pp. 1–36, 2015.

[31] M. Konaković, K. Crane, B. Deng, S. Bouaziz, D. Piker, and M. Pauly, "Beyond developable: computational design and fabrication with auxetic materials," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–11, 2016.

[32] C. Schumacher, S. Marschner, M. Gross, and B. Thomaszewski, "Mechanical characterization of structured sheet materials," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–15, 2018.

[33] L. Chin, J. Lipton, R. MacCurdy, J. Romanishin, C. Sharma, and D. Rus, "Compliant electric actuators based on handed shearing auxetics," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2018, pp. 100–107.

[34] M. Trumić, C. Della Santina, K. Jovanović, and A. Fagiolini, "Adaptive control of soft robots based on an enhanced 3d augmented rigid robot matching," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 4991–4996.

[35] A. Ion, D. Lindlbauer, P. Herholz, M. Alexa, and P. Baudisch, "Understanding metamaterial mechanisms," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–14.

[36] S. Laue, M. Mitterreiter, and J. Giesen, "Computing higher order derivatives of matrix and tensor expressions," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[37] ——, "A simple and efficient tensor calculus," in *AAAI Conference on Artificial Intelligence, (AAAI)*, 2020.