2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)
Yale University, USA

Discretized Modeling of Planar Pneumatic Continuum Manipulators

Erik H. Skorina[1] and Cagdas D. Onal[1]

Abstract— **Soft Robots, robots that are constructed out of soft materials or using compliant actuation methods, can operate safely in complex environments without fear of damaging their surroundings or themselves. However, the soft materials and structures can be imprecise and difficult to control. We then developed a discretized constant-curvature model to predict the behavior of planar bending actuators, both under tip load and while pressurized internally. We experimentally verified this model under a range of configurations, using the model to perform open-loop inverse kinematics. These techniques represent a meaningful advancement in understanding and improving soft actuators, allowing them to move with speed and precision while resisting external forces.**

I. Introduction

Due to their weight and rigidity, articulated robots operated by traditional motors can be dangerous to their environment, limiting their ability to operate efficiently in unstructured settings and alongside humans. Soft actuators can absorb energy to enable safe and compliant physical interaction with the environment in a way that is similar to biological muscles, allowing for a bio-inspired approach to robotics and actuation. One particular bio-inspired manipulator is the continuum manipulator. Continuum manipulators are manipulators that bend continuously along their length instead of at discrete axes like traditional manipulators [1]. Constant-curvature continuum manipulators have been widely studied, including forward kinematics [2], inverse kinematics [3], and dynamics [4], [5]. However, a problem with these manipulators is the difficulty in accurately modeling their behavior in complex situations under external loads, where the constant curvature assumptions break down [6].

Continuum manipulators tend to fall into two categories: pneumatically driven and cable driven. Cable driven continuum manipulators, such as in [7], can combine compliant structures with electric motors. Forward and inverse kinematics can be performed in relation to cable lengths [8], assuming limited external forces. One group who did incorporate external forces into a dynamic model was [9], where the authors used numerical solutions to Cosserat rod theory to perform sliding mode control, though they did not perform physical experiments. A similar bio-inspired continuum manipulator made of interlocking fibers was discussed in [10], where the authors performed rudimentary external force-modeling with an eye toward understanding segment stiffness.

There are difficulties with this approach, such as the physical challenges of dealing with cables, and many groups use pneumatically driven continuum actuators, such as the OctArm [11]. Cosserat rod theory was adapted to this platform as well in [12]. The authors highlighted the advantages in accuracy over using constant curvature segments, though they did not discuss the speed of calculations. Neural networks were used in [13] to develop an inverse kinematic (IK) approximation to be used as a feedforward term in low-level control. A separate group [14] incorporated fixed universal and prismatic joints in their manipulator, which aided kinematics but reduced the system flexibility.

Another approach that is less common is the discretized constant curvature method, which uses simple constant curvature beam bending. This approach was described by [15], though the authors did not do any verification or inverse kinematics. The authors of [16] used a discretized model with inverse kinematics for path following, but ignored the effects of external forces.

In this paper, we expand the discretized approach to include non-constant curvatures and external forces in modeling continuum manipulator behavior. We divide the manipulator into a series of subsegments, propagate external normal and tangential forces through the length of the manipulator, and perform linear beam bending at each subsegment. The simplified nature of this model allows for rapid state calculation without needing to solve the complicated equations found in methods utilizing Cosserat Rod Theory [12]. We verify this model in a planar setting using experiments with a soft pneumatic bending actuator like those used in [17], including performing experiments to calculate the changing cross-sectional area of the pneumatic pressure chambers.

The physical experiments in this paper revolve around a soft linear actuation concept inspired by biological anatomy we call the *reverse Pneumatic Artificial Muscle* (rPAM). The rPAM consists of tubes of elastomer wrapped thread. When pressurized, the thread prevents the actuator from extending and instead causes the elastomer tube to extend; this actuator is called the rPAM because it operates on similar principles to the traditional PAM (also known as the McKibben actuator [18]), only with a reversed direction of actuation (similar to the work of [19]).

The work in this paper uses two rPAMs mounted together in a single soft structure with an inextensible constraint layer in between them. When a single rPAM chamber is pressurized it extends relative to the other, causing the entire

*This material is based upon work supported by the National Science Foundation (NSF) under Grant Nos. IIS-1551219 and CMMI-1728412. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

[1]WPI Soft Robotics Lab, Worcester Polytechnic Institute 100 Institute Rd Worcester, MA 01609

Please direct all correspondence to Cagdas D. Onal at cdonal@wpi.edu

segment to bend. Our model can predict planar bending behavior under internal pressure and external tip forces. The discretized nature of our approach allows for continuum kinematics to be calculated faster than any existing method that incorporates external forces. The contributions of this paper include:

- The creation of a high-speed discretized continuum bending model incorporating external tip loads
- The experimental determination of actuator pressure-chamber deformation
- The verification of the inverse kinematic model performance in a planar setting

II. Discretized Soft Actuator Bending Model

Existing pneumatic continuum models can easily predict actuator behavior when subjected to uniform moments, such as those applied by internal pressures. However, they struggle with incorporating external forces, which result in non-constant moments along the length of each segment. We divide external forces into two categories: normal forces and tangential forces. To deal with these forces, we divide each segment into a number of subsegments, each with a constant curvature different from those around it. This can be used to approximate a complex continuous shape using simple equations. Forces that are fixed in the global coordinate frame, such as gravity, can be converted into the local frame of normal and tangential forces.

A. Normal Force Calculation

The simplest external force to incorporate is the normal force. Normal forces are forces acting perpendicular to the body of the bending actuator. We treat the bending actuators as beams in a cantilever configuration, with the moments caused by external normal forces effecting the bending. In order to calculate this, we start from the tip and use the following equations:

$$N_i = N_{i+1} + F_i, \tag{1}$$

where N_i is the total normal force effecting a subsegment and F_i is the external normal force applied to the subsegment. This creates a running total of the normal force effecting a given subsegment, allowing us to easily calculate moment caused by normal forces, using the following equation:

$$M_{i,n} = M_{i+1,n} + N_i L_i, \tag{2}$$

where $M_{i,n}$ is the external moment caused by the normal force and L_i is the length of the subsegment. We then combine this moment at every subsegment from normal forces, with the moment enacted by the pressure in the pneumatic actuators. We simplified the fabrication of our actuators, keeping the constraint threading surrounding the actuator chambers as shallow as possible. This allowed us to ignore some of the complexities taking into account threading constraint in our previous work [21], and simplified the actuator force to be proportional to force of the air pressure on the end of the pressure chamber. Thus the combined moment can be calculated with the following equation:

$$M_i = P_{a,i} A_i(P_{a,i}) w - P_{b,i} A_i(P_{b,i}) w + M_{i,n}, \tag{3}$$

where M_i is the total moment, $P_{a,i}$ and $P_{b,i}$ are the pressures in the two bending actuator pressure chambers, A_i is the area of the corresponding pressure chamber cross-section as a function of pressure, and w is the distance between the centroid of the pressure chamber and the neutral axis (representing the moment arm of the pressure torque). We use this moment to perform simple beam bending using the following equation:

$$\theta_{i,n} = \frac{M_{i,n} L_i}{(E_i I_i)}, \tag{4}$$

where E_i is the young's modulus of the bending actuator, I_i is the second area moment of inertia, and $\theta_{i,n}$ is the bending angle of the end of subsegment i.

We then use these angles to calculate the state of the manipulator using existing constant-curvature forward-kinematic equations [22], combining the bending of each subsegment to calculate the entire manipulator kinematics. These equations, included in Algorithm 1, treat each subsegment as an arc of a circle and calculate the tip position and orientation accordingly.

B. Tangential Force Calculation

Tangential forces are forces directly along the length of the manipulator. They can be added to Equation 3 based on the moment arm between the direction of subsegment base and the tangential force vector. However, calculating the tangential moment arm is more complicated, as the moment arm effects the moment caused by the tangential force, which in turn effects the moment arm. To account for this without an iterative process at this stage, we made the assumption that, as each subsegment is small, the length of the moment arm for that subsegment is not substantially changed as the subsegment bending angle changes.

Thus we:

1) Calculate the subsegment angle without tangential forces (using Equation 4)
2) Calculate the segment base position with respect to the tip frame using this angle and all previous. This is done using existing continuum kinematic equations, applied in the reverse direction, and allows us to calculate the moment arm of the tangential force and thus the moment it applies on the subsegment
3) Recalculate the subsegment angle incorporating the tangential moment

A pseudocode version of this method can be found in Algorithm 1. This method could be generalized to take into account multiple tangential forces, which could require keeping track of a new moment arm for every additional tangential force. In this paper, we only deal with forces at the tip.

```
Mn = 0, Fnormal = 0, [xo yo] = [0.0];
tSum = 0; // A rolling total of the angle
    prior to the bending of the current
    segment
for i = Segments, from the tip do
    Fnormal = Fnormal + Fexternal(i);
    for j = SubSegments do
        // Normal-Force angle calculation
        Mn = Mn+Fnormal*Length(i);
        Mtotal = Pa(i)*A(i)*w(i)-Pb(i)*A(i)*w(i)+Mn;
        theta(i,j) = = Mtotal*L(i)/(E(i)*I(i));
        // reversed kinematics from the
            perspective of the tip to
            calculate tangential force
            moment arm.
        r = L(i)/theta(i,j);
        xn = -r*(1-cos(theta(i,j)));
        yn = r*sin(theta(i,j));
        [xo_t yo_t] = [xo yo] + Rotate([xn,yn], tSum);
        // incorporate temporary tangential
            moment into angle calculation
        Mtotal = Mtotal-Ft*xo;
        theta(i,j) = Mtotal*L(i)/(E(i)*I(i));
        // update kinematics, from the
            perspective of the tip
        r = L(i)/theta(i,j);
        xn = -r*(1-cos(theta(i,j)));
        yn = r*sin(theta(i,j));
        [xo yo] = [xo yo] + Rotate([xn,yn], tSum);
        tSum = tSum + theta;
    end
end
```

Algorithm 1: A calculation of the subsegment angles of the discritized bending model when subjected to normal and tangential forces at the tip.

C. Global Force Conversion

In order to calculate the effect of forces not fixed to the tip frame of the manipulator, such as gravity, it is necessary to convert them into the tip frame. Direct conversion to the tip frame is impossible, since the forces change the state of the manipulator and thus change the rotation matrix between global and tip. Thus, we need to find the rotation between global and tip coordinate frames that causes the final tip angle to be equal to that rotation. A visualization of this can be seen in Figure 1. Figure 1a shows the conversion between the global and local forces, while 1b shows the final simulated state of the manipulator, with the local force matching the final tip orientation.

We solve this by iteratively using the following procedure:

1) Rotate the global forces into the tip frame using the guess of the steady state tip angle
2) Run the simulation (Algorithm 1) to calculate the steady state actuator position using those normal and tangential forces
3) Check that the final tip angle is close to the that of the guess
    a) If not, repeat using a new guess based on the error

This can be seen in more detail in Algorithm 2.

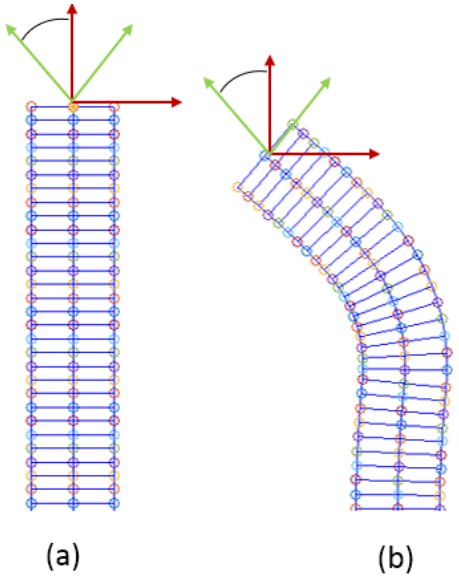

Fig. 1. An arbitrary actuator to illustrate the global force algorithm. (a) Any forces in the global coordinate frame (red) are converted into the local tip coordinate frame (green) using a guess of the steady-state tip angle. (b) The forward kinematic simulation is run using normal/tangential forces, and we check to make sure that the guess of the tip angle is correct.

```
ForceAngle = 0;
LocalForces = Rotate(GlobalForces,ForceAngle);
TipAngle = LocalForceKinematics(LocalForces, Pressures);
Error = TipAngle-ForceAngle;
while | Error |>0.01 do
    // adjust global/local conversion
        angle
    ForceAngle = ForceAngle+Error*Alpha;
    // calculate local forces
    LocalForces = Rotate(GlobalForces,ForceAngle);
    // Perform simulation
    TipAngle = LocalForceKinematics(LocalForces,
     Pressures);
    // Does tip angle match local force
        angle?
    Error = TipAngle-ForceAngle;
end
```

Algorithm 2: Iterative calculation of the actuator state when subjected to external forces tied to the global reference frame.

D. Inverse Kinematics

One purpose of having a model of segment behavior is that it allows us to predict the pressures required to reach a certain point, known as inverse kinematics (IK). We minimized the error between the desired tip position and orientation and the actual tip position and orientation returned by the forward kinematic function from the previous section.

We used Matlab's constrained minimization toolbox with the Active-set algorithm to perform this minimization, taking into account the constraints of the pressure that the actuators can withstand. In addition, to simplify the minimization we reduced the pressure inputs to the minimum necessary to traverse the entire workspace. For a 1-DoF planar bending actuator with 2 inputs, we only considered a single pressure

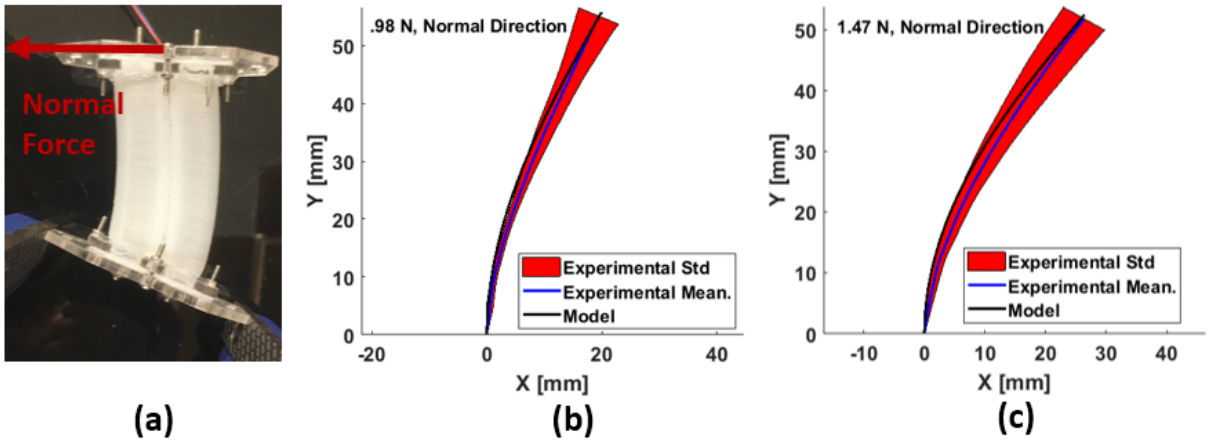

Fig. 2. (a) 1-Segment planar model experimental setup. (b-c) 1-Segment model force verification.

input ranging from $-p_{max}$ to p_{max}. When applying these pressures to a physical system, negative pressures were treated as positive pressures for the other pressure chamber.

III. Discretized Model Verification

We verified the behavior of the discretized bending model in a planar setting through experimentation. We used soft, 1 Degree-of-Freedom pneumatic bending actuators for our verification. These segments are fabricated out of EcoFlex 0030 silicone, with two chambers wrapped in thread on either side of a thin, plastic constraint layer. When pressurized, a single chamber will try to extend, causing the entire segment to bend. More detail about our bending actuators can be found in [17], [23], [24].

A. Model Force Verification

The first step in model verification is to verify a single segment under external load. This was first done by mounting a single bending segment with a fixed base and applying a normal load to its tip. An example of this can be seen in Figure 2. We first performed experiments using 0.98 N of applied load. Using the data from these experiments, we calculated the Young's Modulus to be 75 kPa.The experimental results for this experiment, along with the calibrated model results, can be seen in Figure 3(a).

To verify the model, we performed the same experiment multiple times. One example can be seen in Figure 3(b), where a normal force of 1.47 N was applied. The model predicts end-effector positions within 1 mm.

B. Pressure Verification

The first step was to determine the relationship between the actuator cross-sectional area and input pressure. We analyzed the changing cross section by pouring resin (Smooth-Cast 305) into the pressure chamber, pressurizing it to a desired level, and letting the resin cure. A series of resin slices from this experiment can be found in Figure 3(a), along with their cross-sectional areas. The cross-section increases from 65 mm^2 at 0 psi to 125 mm^2 at 9 psi, nearly doubling the force output from the given pressure. We created a lookup table to capture this change in area, and used it as in the calculation of the moment (Equation 3).

Using this, we performed similar experiments to verify the ability of the model to respond to applied pressures. This experiment was set up similarly to that in Figure 2. However, in order to test a wider range of actuator states we used a motion tracking system to gather data, which only recorded the endpoint of the actuator under pressure. Casters were added to the bottom of the actuator to reduce the effect of friction when moving between states. An example of the results of this can be seen in Figure 3(c), verifying the accuracy of model to incorporate pressure into the bending.

C. IK Verification

We performed experiments verifying the inverse kinematic algorithm. In order to create a more complicated workspace, we attached two actuators in series. An example state of this can be seen in Figure 4(a). To minimize the valving requirements for these experiments, we limited the inputs to a single pressure chamber in each actuator. The interface between the two actuator sections was treated as a subsection with a Young's Modulus much higher than the bending segments.

For load, a 30 g weight was attached to the end effector and hung off of a pulley. The pulley was situated so that this would cause a constant 0.3 N force vertically with respect to the actuator, and was incorporated into the algorithm as such. The IK was performed with and without factoring the weight into (though the weight was included in both experiments), and the results of these experiments can be seen in Figure 4(b-c). This data shows the averages over nine trials. From Figure 4(b), we can see that both paths show an approximation of the desired points, though the path without gravity compensation is smaller. Gravity compensation provides a consistently good approximation of the desired point, except for point 4. As travel between point 3 and point 4 requires moving backward and thus letting off pressure, the motion is not actively assisted by the actuator.

We also performed a similar experiment using a smoother series of points, the results of which can be seen in Fig-

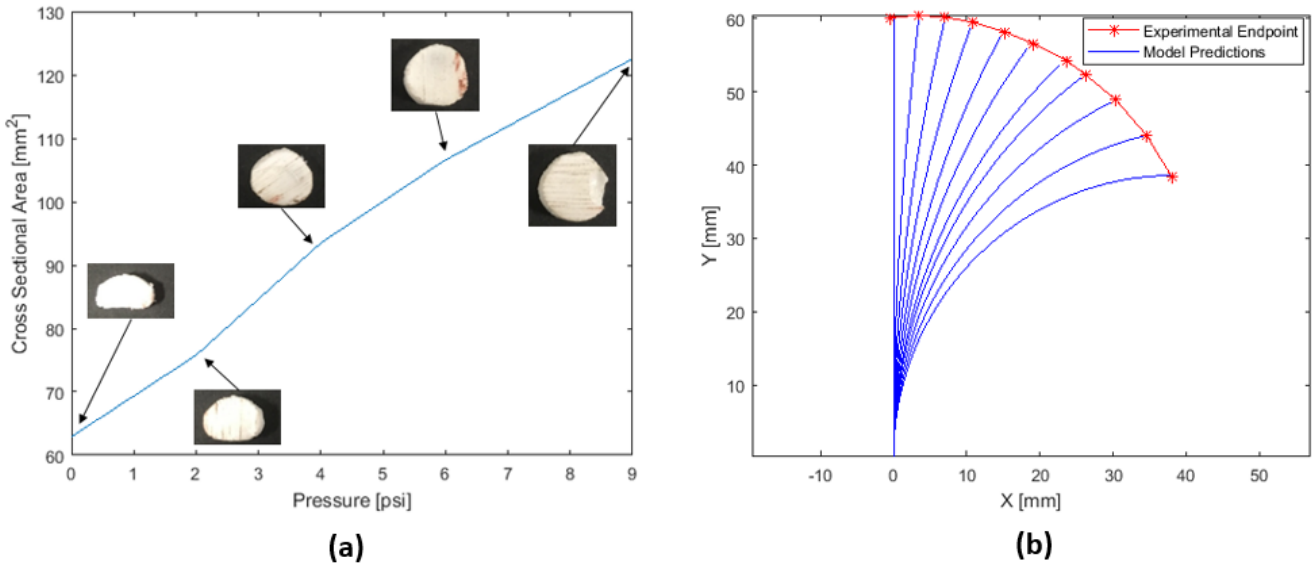

Fig. 3. (a) Actuator cross-sectional areas were found by filling the pressure chamber with pressurized resin and extracting the resulting cores after the resin had cured. (b) Pressure Verification.

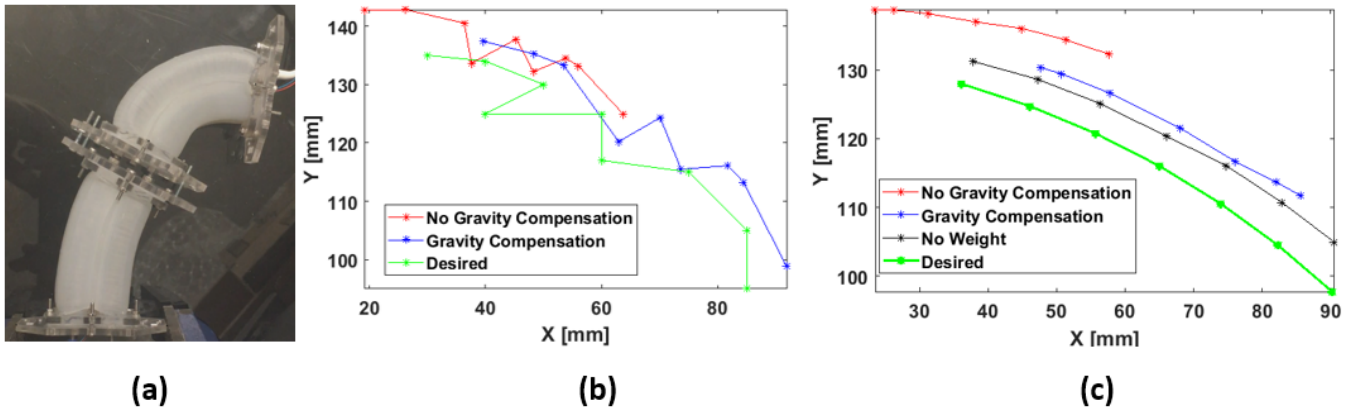

Fig. 4. (a) The two DoF Inverse Kinematic Setup. Each actuator is 60 mm long with a 22 mm interface section in between. (b-c) A verification of the inverse kinematic model with a 0.294 N vertical external force. Experiments both with and without the force factored into the model were performed. (b) shows the tip trajectories for a jagged trajectory, while (c) shows the tip positions for a smooth, circular trajectory.

ure 4(c). In this case, we also included the no-weight experiment. Again, without gravity compensation the path is smaller and situated closer to the y-axis. There was broad agreement between the experiments with the weight factored in and the experiments with no weight, highlighting the accuracy of our model in calculating the effect of external forces.

The 'No Gravity Compensation' error increases throughout the 7 desired points, as the desired x-position increase, while the 'Gravity Compensation' error remains consistent throughout. This constant overshoot was likely the result of a combination of factors, including imprecision in motion tracking beacon placement, errors in actuator fabrication, and the fact that the actuators were not perfectly inextensible. The latter could be incorporated as an additional DoF in future versions of this model, with the length of the actuator being a function of total pressure instead of being constant.

D. Speed Verification

To justify the simplicity of our model for rapidly performing kinematic calculation, we compared the run-times against the real-time continuum manipulator model discussed in [25]. We ran each method in Matlab over 100 iterations with a 40 subsegment beam. When testing the speed of our discretized method, we used a global force that changed each iteration to ensure a variety of states to solve, while when running the code provided by [25] we let the dynamic nature of the simulation ensure variety .We found our method proved to be an average of 3.5 times faster (0.025 sec versus 0.085 sec average run time). Though this comparison is imperfect because the work of [25] was a dynamic simulation, it represented the most up-to-date work on fast actuator simulation that also provided the code to allow us to run a direct time comparison.

IV. Conclusion

This paper used a simple constant curvature beam bending to model the behavior of infinite passive-DoF structures. We divided each segment into a series of sub-segments, each with a constant curvature but a different externally-applied moment, which allowed us to accurately model bending segment behavior under both normal and tangential external loads. We used this model to perform inverse kinematics in a planar setting, demonstrating its usability in both unloaded and loaded situations. Finally, we compared the run times of our loaded kinematic model with the state of the art, and found ours to be approximately 3 times faster.

This model represents a meaningful step towards being able to rapidly calculate the behavior of soft continuum manipulators under external load. This ease of calculation improves soft robotic control and motion planning, and can be used to help soft robotic systems operate with more confidence and precision, which would allow them to effectively perform tasks in environments with humans or other delicate obstacles. It helps pave the way for soft robots collaborating with humans, bringing humans and robots closer like never before.

The next step is to expand this high-speed, discretized approach to predict continuum manipulator behavior in three dimensions or with extensible actuators. The former could be achieved by running two separate models of planar bending in the X-Z and Y-Z planes, calculating the resulting subsegment-angles, and combining them to calculate the full manipulator state.

ACKNOWLEDGMENT

The authors would also like to acknowledge Sarah E. Schultz and Ann-Marie Votta for help with the manuscript.

References

[1] G. Chirikjian, "Conformational modeling of continuum structures in robotics and structural biology: A review," *Advanced Robotics*, vol. 29, pp. 1–13, 08 2015.

[2] I. A. Gravagne, C. D. Rahn, and I. D. Walker, "Large deflection dynamics and control for planar continuum robots," *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 2, pp. 299–307, June 2003.

[3] S. Neppalli, M. A. Csencsits, B. A. Jones, and I. D. Walker, "Closed-form inverse kinematics for continuum manipulators," *Advanced Robotics*, vol. 23, no. 15, pp. 2077–2091, 2009. [Online]. Available: https://doi.org/10.1163/016918609X12529299964101

[4] E. Tatlicioglu, I. D. Walker, and D. M. Dawson, "Dynamic modelling for planar extensible continuum robot manipulators," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, April 2007, pp. 1357–1362.

[5] I. A. Gravagne and I. D. Walker, "Uniform regulation of a multi-section continuum manipulator," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 2, May 2002, pp. 1519–1524 vol.2.

[6] M. B. Pritts and C. D. Rahn, "Design of an artificial muscle continuum robot," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 5, April 2004, pp. 4742–4746 Vol.5.

[7] A. K. Mishra, A. Mondini, E. Del Dottore, A. Sadeghi, F. Tramacere, and B. Mazzolai, "Modular continuum manipulator: Analysis and characterization of its basic module," *Biomimetics*, vol. 3, no. 1, 2018. [Online]. Available: http://www.mdpi.com/2313-7673/3/1/3

[8] M. Rolf and J. J. Steil, "Constant curvature continuum kinematics as fast approximate model for the bionic handling assistant," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2012, pp. 3440–3446.

[9] A. A. Alqumsan, S. Khoo, and M. Norton, "Robust control of continuum robots using cosserat rod theory," *Mechanism and Machine Theory*, vol. 131, pp. 48 – 61, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0094114X18311777

[10] M. S. Moses, M. D. M. Kutzer, H. Ma, and M. Armand, "A continuum manipulator made of interlocking fibers," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 4008–4015.

[11] W. McMahan, V. Chitrakaran, M. Csencsits, D. Dawson, I. D. Walker, B. A. Jones, M. Pritts, D. Dienno, M. Grissom, and C. D. Rahn, "Field trials and testing of the octarm continuum manipulator," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, May 2006, pp. 2336–2341.

[12] D. Trivedi, A. Lotfi, and C. D. Rahn, "Geometrically exact models for soft robotic manipulators," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 773–780, Aug 2008.

[13] D. Braganza, D. M. Dawson, I. D. Walker, and N. Nath, "A neural network controller for continuum robots," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1270–1277, Dec 2007.

[14] R. Kang, D. T. Branson, T. Zheng, E. Guglielmino, and D. G. Caldwell, "Design, modeling and control of a pneumatically actuated manipulator inspired by biological continuum structures," *Bioinspiration & Biomimetics*, vol. 8, no. 3, pp. 1270–1277, 2013.

[15] J. Fraś, J. Czarnowski, M. Maciaś, and J. Główka, "Static modeling of multisection soft continuum manipulator for stiff-flop project," in *Recent Advances in Automation, Robotics and Measuring Techniques*, R. Szewczyk, C. Zieliński, and M. Kaliczyńska, Eds. Cham: Springer International Publishing, 2014, pp. 365–375.

[16] T. Mahl, A. Hildebrandt, and O. Sawodny, "A variable curvature continuum kinematics for kinematic control of the bionic handling assistant," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 935–949, Aug 2014.

[17] M. Luo, E. H. Skorina, W. Tao, F. Chen, S. Ozel, Y. Sun, and C. D. Onal, "Towards modular soft robotics: Proprioceptive curvature sensing and sliding-mode control of soft bidirectional bending modules," *Soft Robotics*, 2016.

[18] G. K. Klute, J. M. Czerniecki, and B. Hannaford, "Mckibben artificial muscles: pneumatic actuators with biomechanical intelligence," in *Advanced Intelligent Mechatronics, 1999. Proceedings. 1999 IEEE/ASME International Conference on*. IEEE, 1999, pp. 221–226.

[19] R. H. Gaylord, "Fluid actuated motor system and stroking device," Jul. 22 1958, uS Patent 2,844,126.

[20] H. Schulte Jr, "The characteristics of the mckibben artificial muscle. application of external power in prosthetics and orthotics, washington dc," *Nat. Acad. Sci.-Nat. Res. Council*, 1961.

[21] E. H. Skorina, M. Luo, W. Y. Oo, W. Tao, F. Chen, S. Youssefian, N. Rahbar, and C. D. Onal, "Reverse pneumatic artificial muscles (rpams): Modeling, integration, and control," *PLOS ONE*, vol. 13, no. 10, pp. 1–24, 10 2018. [Online]. Available: https://doi.org/10.1371/journal.pone.0204637

[22] M. W. Hannan and I. D. Walker, "Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots," *Journal of Robotic Systems*, vol. 20, no. 2, pp. 45–63. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.10070

[23] W. Tao, E. H. Skorina, F. Chen, J. McInnis, M. Luo, and C. D. Onal, "Bioinspired design and fabrication principles of reliable fluidic soft actuation modules," in *Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2169–2174.

[24] E. H. Skorina, M. Luo, W. Tao, F. Chen, J. Fu, and C. D. Onal, "Adapting to flexibility: Model reference adaptive control of soft bending actuators," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 964–970, April 2017.

[25] J. Till, V. Aloi, and C. Rucker, "Real-time dynamics of soft and continuum robots based on cosserat rod models," *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 723–746, 2019. [Online]. Available: https://doi.org/10.1177/0278364919842269