Kinematic Analysis of Soft Continuum Manipulators Based on Sparse Workspace Mapping

Jing Li[1], Xiaojiao Chen[2], Yinyin Su[1,3], Wenping Wang[2], James Lam[1], Zheng Wang*[3]

Abstract—**Soft robots, with advantages of high adaptability to the environment, relatively easy and simple fabrication process as well as promising performances, have been thoroughly investigated and widely applied lately, the superiority of which has been proved in areas such as medicine, industry, daily life service and so on. However, it is still challenging to realize stable, efficient, and accurate control of soft robots due to their high compliance and hyper-redundancy. One of the main causes is the difficulty in building an accurate model for analyzing the relation between control input and output (force and/or deformation). In this paper, we proposed an intuitive approach for solving inverse kinematics of soft manipulators, where the relation between actuation pressures and end-effector motions was established by analyzing the sampled workspace from a real platform. A quantitative measurement between control accuracy and computational efficiency was performed and applied in achieving a reasonable balance in between. Based on the proposed approach, real-time motion tracking of a self-developed soft manipulator was implemented on Raspberry Pi taking less than 10 ms and the tracking error was 3.35% of the full workspace in average, comparable to the system capability. Our approach has validated the feasibility of fast searching in finding inverse kinematics solutions with satisfactory accuracy and simple implementation process and demonstrated its potential in working as the basis in advanced control.**

Index Terms—**Modeling, Control, and Learning for Soft Robots**

I. INTRODUCTION

IN the past few decades, conventional manipulators composed of rigid links, have achieved great success especially in manufacturing industry, where precision, repeatability as well as efficiency are highly demanding [1]. Most industrial manipulators can be modelled and controlled under a set of control algorithms [2], [3]. However, they become less appropriate when interacting with humans and environment, where soft robots become advantageous due to their high adaptability [4]–[8]. Nevertheless, the nonlinearity introduced from the inherent compliance of soft manipulators brings difficulties in modelling and control and makes conventional modelling methods suited for rigid ones inapplicable. Existing methods for modelling and control of soft manipulators can be divided into two categories, model-based [9]–[12] and model-free approaches [13]–[15].

Model-based approaches can be further distinguished according to whether the model is derived based on the steady state assumption or coupled with dynamic formulations [16], [17]. Constant curvature (CC) assumption has been widely applied in kinematics analysis of soft manipulators [18]–[21]. It works well when external force effects can be ignored and the manipulator itself is designed and fabricated symmetrically and uniformly but appears inaccurate under high load and high-speed dynamic process [16], [22]. To improve the accuracy, models based on CC assumption introducing varied curvatures have been developed [23]–[25], however, considering the increase in their computational cost, the improvements of accuracy are not considerably enough. To take the effect of external forces into consideration, dynamic equations can be derived based on Lagrangian formulation or Euler-Newton equation, where the corresponding items can be obtained by analyzing the specific manipulators and simplifications were usually required considering the complexity, which might cause inaccuracies [17], [26], [27]. Meanwhile, beam theories were also used to derive dynamic equations [28]–[31], where solving the partial differential equations in real-time has become the key step [32], [33]. Similarly, computation of the updated node positions at each time step is needed in Finite element model (FEM)-based methods [34]–[36]. Meanwhile, the computational power is highly demanding in both methods.

Model-free approaches, on the other hand, are usually data-driven combined with machine learning strategies and have played an important role in soft robots control, especially for the ones with highly nonlinear properties, whose models of are hard to build [37]. However, they usually require a large set of data and might suffer from uncertainty of stability and convergence [16], [17], which has somewhat affected their performances.

The difficulty of establishing an accurate model of soft manipulators and the dilemma of choosing accuracy over computational efficiency or otherwise have motivated us to investigate on analyzing soft manipulators with less reliance on models. In this article, we focus on static kinematics analysis of soft continuum manipulators, where an intuitive approach based on workspace analysis was proposed including three steps as workspace sampling, gridding and searching (WSGS).
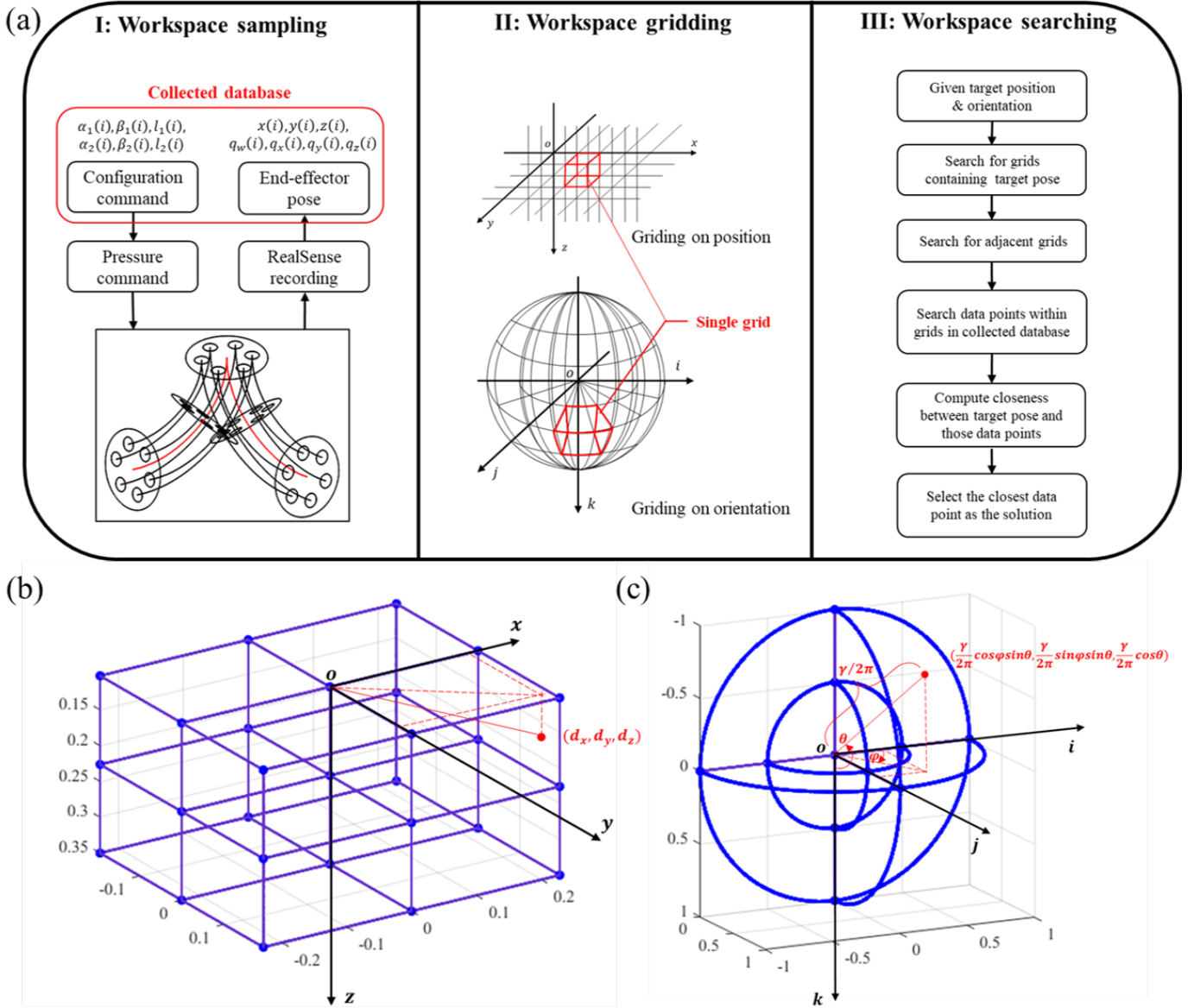
Fig. 1. The proposed WSGS approach. (a) I: Workspace sampling step. II: Workspace gridding step. III: Workspace searching step, with sub-steps presented sequentially. (b) Position gridding in Cartesian Coordinate System. (c) Orientation gridding in Spherical Coordinate System.

The proposed WSGS approach has built a bi-directional correspondence between control input (pressure) and output (end-effector position and orientation) by combining a simple model and a straightforward data-driven strategy. It has been proved that, for our soft manipulator, sparse sampling and gridding is sufficient and thus rapid searching can be realized. Additionally, the variance in control results with different factors applied in WSGS steps, provides us with control adaptability when facing different requirements or under varied conditions. Meanwhile, different from learning-based methods, the data collected for analysis provides us a concrete idea of the workspace, therefore the results can still be completed and achieve certain accuracy even with sparse sampled data. With its simple implementation process, promising performance as well as configuration-independent feature, the proposed WSGS approach could be applied in a wide range of soft manipulators

and become the basis of several advanced control algorithms. Conditions with external load, although have not been included in this work, can be further considered by collecting data points under different payloads or applying sensory feedback.

The main contributions of this paper includes:

1) An intuitive approach for modelling and control of soft continuum manipulators was proposed, where we utilized the inherent compliance of soft manipulators to simplify the kinematic analysis based on workspace mapping.

2) The proposed approach proved the feasibility of simplifying the inverse kinematics problem to an intuitive space searching problem.

3) A quantitative relation between control accuracy and efficiency was established based on our workspace mapping strategy and it works either in favor of accuracy or
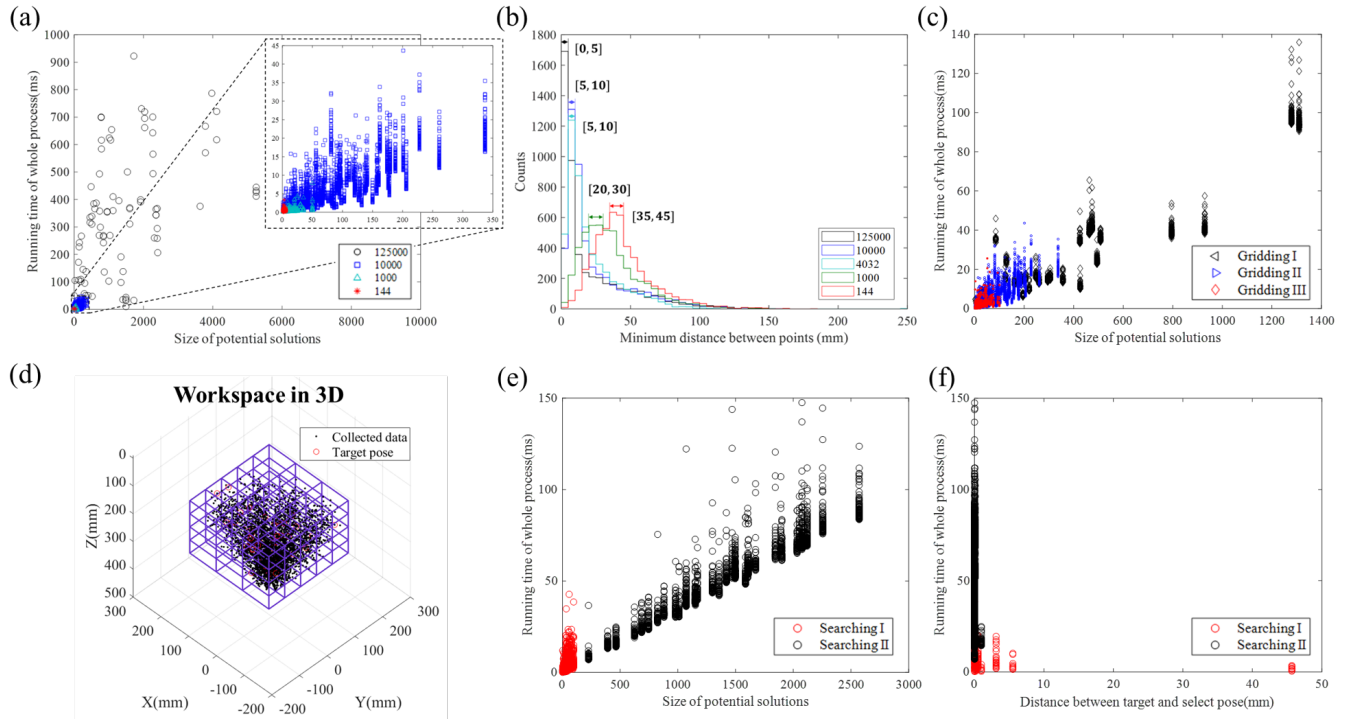
Fig. 2. Workspace analysis and mapping strategies. (a) Plot of running time versus size of potential solutions with different workspace sampling strategies. (b) Histograms of positioning errors under different workspace sampling strategies. (c) Plot of running time versus size of potential solutions with different workspace gridding strategies. (d) Workspace in 3D with data points applied for experiments comparing two searching strategies. (e) Plot of running time versus size of potential solutions with different workspace searching strategies. (f) Plot of running time versus distance between target and selected poses computed by the algorithm for different searching strategies.
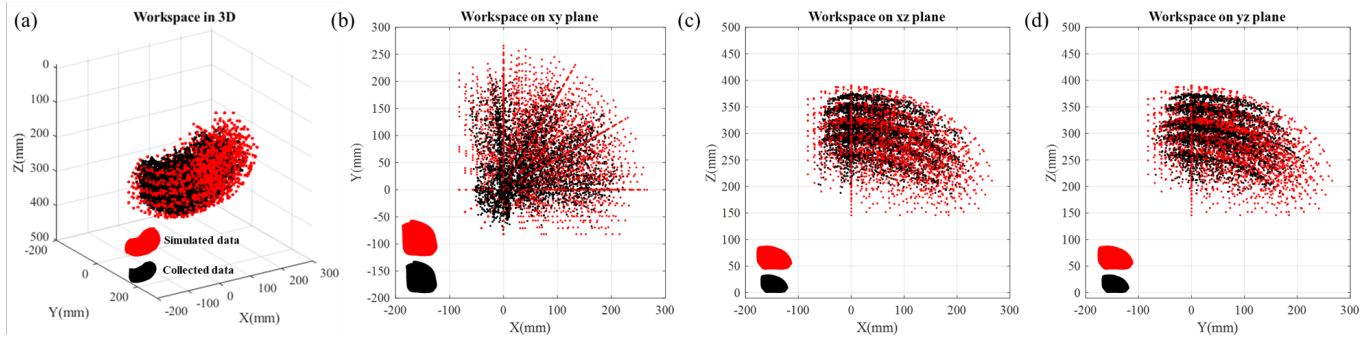

Fig. 3. Workspace plot from different views. (a) Workspace in 3D. (b) Workspace plot on xy plane. (c) Workspace plot on xz plane. (d) Workspace plot on yz plane. Black dots are the data collected from real platform, and red dots are the data simulated based on PCC assumption. The graphics at the bottom left side of each figure are the envelope shapes of corresponding data points distributions.

efficiency depending on the real situations. The proposed method has been validated in real-time motion tracking test, where sparse sampling was verified to be sufficient to achieve satisfactory accuracy in a fast searching manner.

The proposed approach was validated on our self-developed soft manipulator, which was based on our previous work [38], [39] and has been redesigned both mechanical and control wise. This paper includes five sections. Apart from the first introduction and the last conclusion section, the implementation of the proposed approach is described in section II, while the workspace mapping strategies and analysis are presented in section III. The results of the system capability and a real-

time motion tracking test will be presented in section IV.

II. KINEMATIC ANALYSIS OF SOFT MANIPULATOR BASED ON WORKSPACE MAPPING

The detailed implementation of the proposed method includes three steps as shown in Fig.1 combined with a simple model based on PCC assumption as described in below.

Based on the Piece-wise Constant Curvature (PCC) assumption [19], only three configuration parameters are necessary to define the kinematics of unit piece, including the bending angle α, describing the radian angle of the corresponding arc, rotational angle β, determining the plane where the segment bends, and the central arc length l.
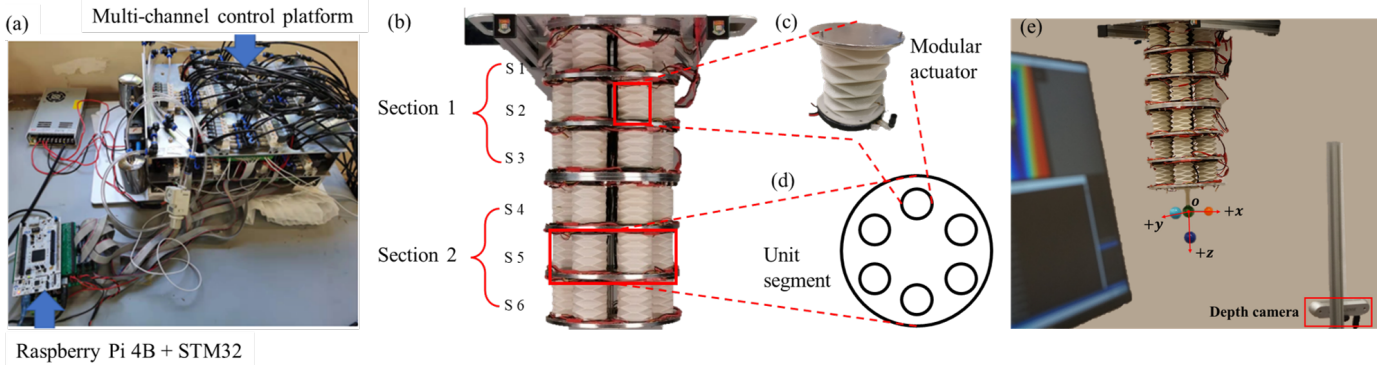
Fig. 4. Soft manipulator system. (a)~(d) Self-developed soft manipulator, with unit segment and modular actuator presented separately. (e) Imaging system.

A. The Proposed WSGS Method

- Workspace sampling

  Workspace sampling was performed in terms of joint level, by selecting values of configuration parameters as α, β, l within certain range. The sampled data points for each parameter were spaced by the same value. Considering the symmetrical feature of our manipulator and to shorten the process of data collection, only the first quadrant of workspace was taken into consideration. There are two individually controlled segments of our manipulator, therefore, values for six parameters as $(\alpha_1, \beta_1, l_1, \alpha_2, \beta_2, l_2)$ need to be assigned.

- Workspace gridding

  Workspace gridding can be separated as gridding for end-effector position and for orientation as shown in Fig.1.(b) and (c). Positions of end-effector can be represented by points in three-dimensional space, expressed by their Cartesian Coordinates. Gridding on positions is therefore straightforward, by directly dividing the three-dimensional workspace based on its Cartesian Coordinates. The closeness of two points can therefore be measured by their Euclidean distance.

  Gridding on orientations, on the other hand, requires two steps of coordinate conversions in advance. Firstly, a coordinate conversion of orientation from axis-angle to Spherical Coordinate expression needs to be performed. For each orientation expressed as a point in Spherical Coordinate as (r, θ, φ), the unit vector pointing from the origin to the point determined by polar angle θ and azimuth angle φ represents the rotational axis and the distance between the origin and the point determined by radial distance r corresponds to the rotational angle in radians and is normalized by dividing it with 2π, the magnitude of which is therefore constrained between 0 and 1. The second conversion from Spherical Coordinates to Cartesian Coordinates will be needed when computing the Euclidean distance in between for measuring the closeness between two orientations.

  Considering the equivalence of two orientations with rotational axes in opposite directions and rotational angles with opposite signs but the same absolute values, the full orientation workspace can be simplified to half of the

sphere, where the polar angle θ is in the range $[0, \pi)$, azimuth angle φ is within $[0, \pi)$, and radial distance r is limited to $[0, 1)$, corresponding to the orientation, whose rotational axis lies within half of the sphere, and the rotational angle varies within $[0, 2\pi)$. Gridding on orientations is therefore based on the orientation workspace expressed in Spherical Coordinates, by dividing the polar angle, the azimuth angle as well as the radial distance accordingly. The closeness between two poses, when position and orientation are both considered, is measured by the weighted combination of normalized Euclidean distance between positions and orientations respectively.

- Workspace searching

  For any given target pose, first step is to search the grids which it lies in. the next step is to search all the points inside those grids, and the closeness between the points and the target pose will be evaluated and compared. If no point is found, searching adjacent grids will be performed until either at least one point is found or the whole workspace has been searched. The adjacent grids mean the grids next to the given grid including the grids on its left, right, up, down, forward and backward. Once the closest point is found, its corresponding command will become the solution. Otherwise, the soft arm remains at its previous pose.

B. Workspace Mapping Strategies

Different strategies can be applied in each step for workspace mapping and they will affect the control results by changing the corresponding computational efficiency and positioning accuracy.

It is intuitive that the denser the sampling is, the better the accuracy should be, however, larger the data size is, slower the searching process would be. The results presenting computational efficiency of different sampling strategies are shown in Fig.2.(a), where the data sizes vary from 144 to 125000. It was found that the running time reduced to within 10 ms with no more than 1000 data points while achieving almost 1000 ms with data size larger than tens of thousand.

The positioning accuracy was measured by the statistic results of tracking errors from 2000 randomly picked data points as shown in Fig.2.(b), where the histogram shows
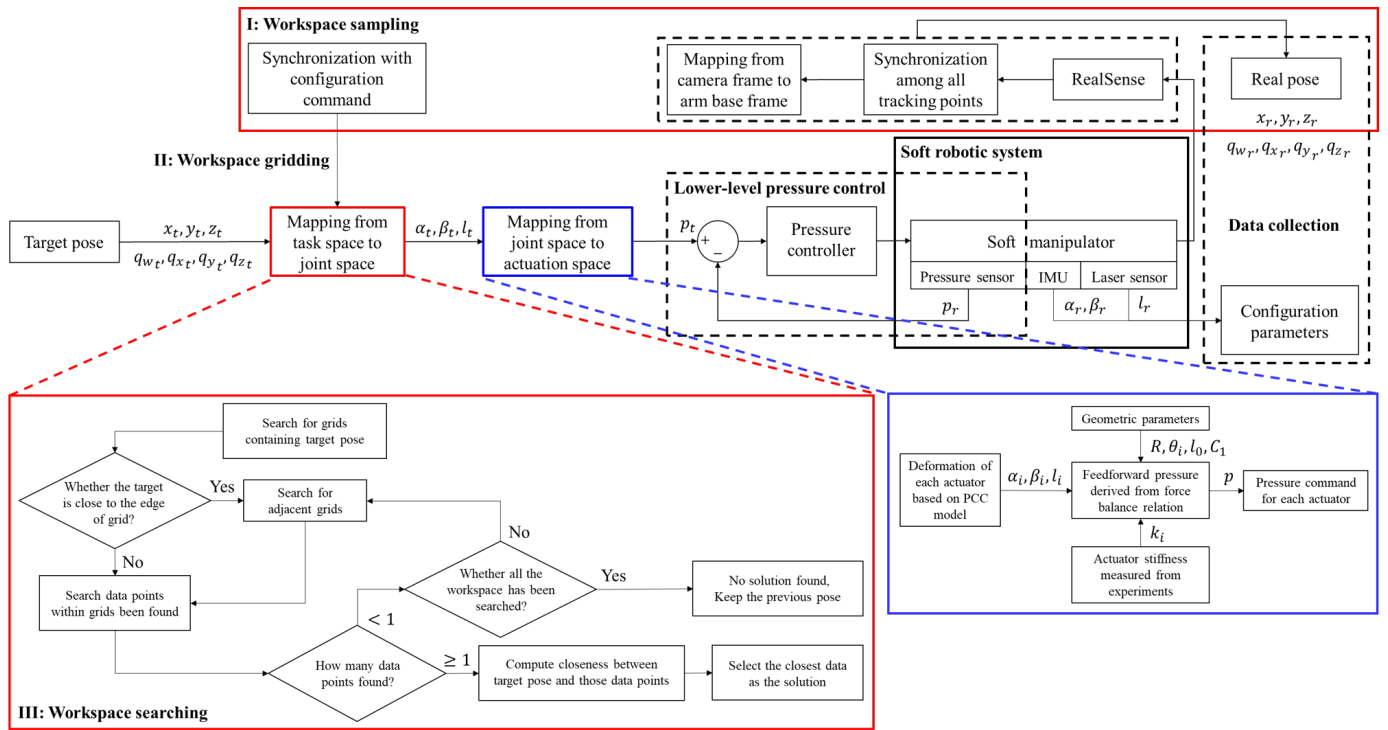
Fig. 5. Control scheme of the proposed system, where the given target pose is $(x_t, y_t, z_t, q_{wt}, q_{xt}, q_{yt}, q_{zt})$, and configuration parameters are (α_t, β_t, l_t), manipulator parameters are represented as (R, θ_i, l_0, C_1) as described in our previous work [38], [39].

the distribution of tracking errors of each sampling strategy. We evaluated the positioning accuracy of each case based on the tracking error that has the largest counts. The larger the data size is, the higher its positioning accuracy becomes. To determine workspace sampling strategy, the efficiency, positioning accuracy as well as the system capability need to be considered. To achieve a comparable repeatability accuracy of the system, we chose a data size of 4932 for experimental validation, the positioning accuracy of which lies between 5 to 10 mm and the running time should be less than 45 ms.

Gridding strategy affects the efficiency by changing data size in certain grids. In dense gridding, data points in each grid will be less than those with sparse gridding. Therefore, it reduces the time of searching for the closest data point in one grid, while increasing the time of searching for the right grid. Specific relation in between can be inferred from Fig.2.(c), where multiple tests have been performed with different gridding strategies as splitting the workspace in x, y, z directions by 3,3,2 grids in gridding strategy I, and by 4, 6, 5 grids in strategy II, and by 10, 10, 10 grids in strategy III, respectively. Based on the results of running time, we chose gridding strategy II for implementation with running time no more than 50 ms.

As mentioned above, the searching strategy we applied is quite straightforward by sequentially comparing closeness between each data point and the target pose in the grid been found. However, it may lead to inaccuracies when the target point is close to or on the edge of a grid, because the nearest data point may not be in the grid where the target is located, but in the grid next to it. One solution is to search not only the grid where the target is located but also the grid next to

it, which will increase the number of data points evaluated and thus the time spent on the search. We simulated this situation by intentionally setting the target pose close to the edge, as shown in Fig.2. (d). The results are consistent with our intuition, but how long the run time is and how large the error is depends on the specific data point, as can be seen from Fig.2. (e) and (f). Therefore, we suggest that a prediction can be made in advance about which part of the working area the target is located in. If it is located near the edge of the grid, then it is necessary to search all adjacent grids beyond the one it is on, otherwise only data points within that grid can be searched.

III. WORKSPACE ANALYSIS

Workspace plot under different cases is shown in Fig.3.(a)~(d), where envelopes shapes can be found at the bottom left side. The similarities of these envelopes have validated our workspace sampling results as well as the feasibility of PCC assumption on our arm. The distributions of data points appear to be radial symmetric, which is in consistent with the uniform design of our manipulator. Meanwhile, the differences between distributions have revealed the self-weight effect, and it becomes larger when the arm deviates more from the center, which is in correspondence with the increase of load relative to the base.

Our proposed approach is capable of turning the kinematics analysis and motion planning into an intuitive space division problem, based on the workspace plotted via data from the real platform. To reach or to avoid some points in space, it becomes a matter of figuring out in which section of the workspace
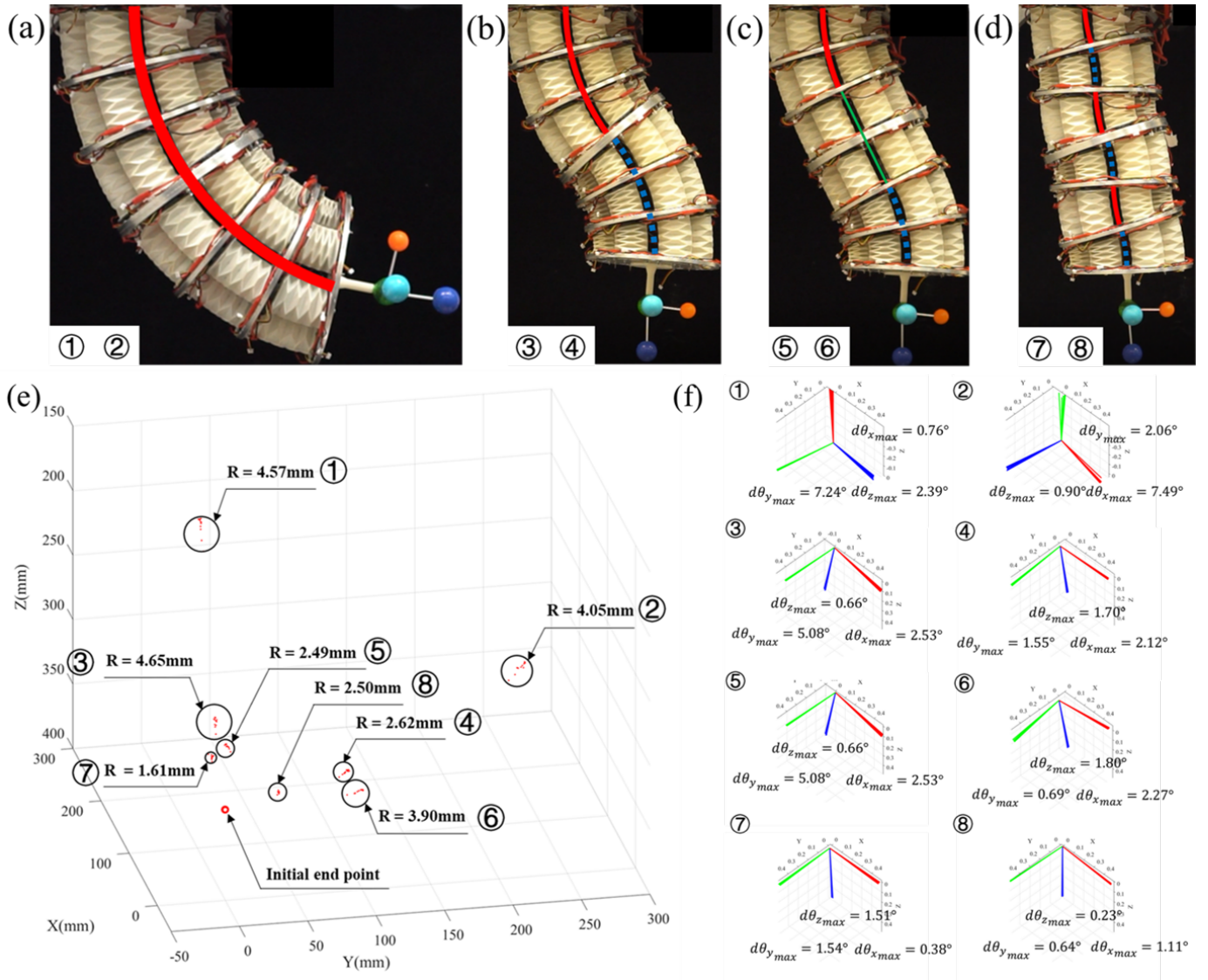
Fig. 6. Repeatability performance test and results. (a) Schematic showing configurations of trial 1 and 2 bending on different planes. (b)~(e) Pictures of different configurations for repeatability test. (f) Repeatability test results of positions. (g) Repeatability test results of orientations.

should or should not the arm be. And in control algorithm, it will be simplified to a space searching problem for both position and orientation. It becomes extremely useful when it comes to problems such as avoiding singularity or obstacles. Instead of adding constrains or tuning parameters in algorithms based on optimization techniques, it would be one simple step in our algorithm to just remove certain grids during searching process.

IV. EXPERIMENTAL VALIDATION

A. Experimental Apparatus

The soft manipulator used for experimental validation was based on our previous work [38], [39], and we have updated both the mechanical design and control of this version. Different from the previous design with one segment composed of six bellows, here a modular actuator design was applied with origami structure [40] as shown in Fig.4.(c). The manipulator we developed weighs around 2.5 kg with a cross section

radius around 100 mm and a variable length ranging from 100 mm to 500 mm. It consists of six segments labeled by $s_1 \sim s_6$ as shown in Fig.4.(a) and (b). Six identical actuators are evenly distributed in each segment as shown in Fig.4.(d), each with a diameter as 58 mm. A pneumatic platform with 54 independent channels was developed, and here only 36 of the channels were used (6 actuators for each segment and 6 segments in total). The embedded control platform weighs around 10 kg with a size of 500 mm x 400 mm x 200 mm as shown in Fig.4.(a). For our algorithm validation, to cover degrees of freedom of both positions and orientations without losing generality, only two sections were individually controlled, each consists of three segments, labeled as Section 1 and 2 in Fig.4.(b).

The control scheme of our soft manipulator is shown in Fig.5, where pressure sensors are embedded in each actuator for feedback control, and the inner pressure control loop is similar in our previous work [38], [39]. Once given
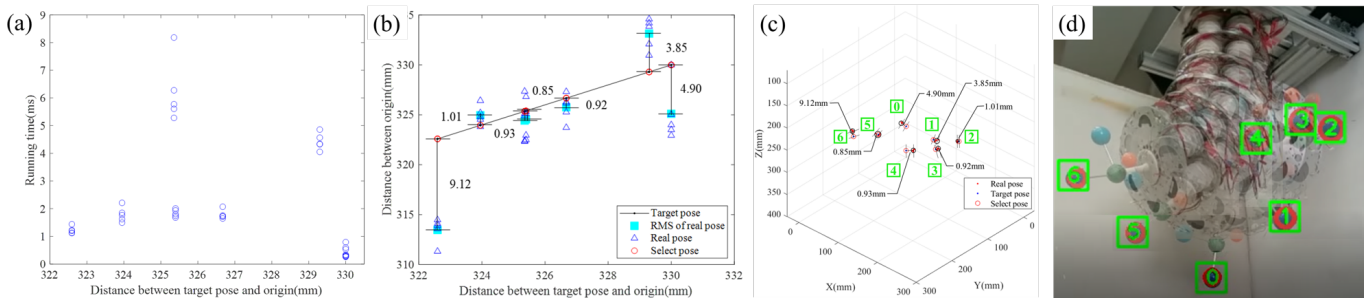
Fig. 7. Motion tracking test results. (a) Running time versus distance between target position and origin for different tracking points. (b) Distance between origin and different data points versus distance between target and origin point. (c) Data points plotted in 3D. (d) Overlapped pictures of motion tracking test, where each target point and positions of the corresponding ball are labelled with numbers.

a target pose, including position (x_t, y_t, z_t) and orientation $(q_{wt}, q_{xt}, q_{yt}, q_{zt})$, the configuration parameters for each segment (α_t, β_t, l_t) will be determined by the proposed WSGS approach. The pressure command p_t of each actuator will then be computed based on least squares method as described in [38], [39].

The imaging setup is shown in Fig.4.(e), where the position and orientation of the end-effector was computed via positions of four balls attached to it by a depth camera (Intel RealSense), and the computational steps include the synchronization of four balls' positions, mapping from camera base frame to arm base frame and synchronization with configuration command. Considering the identification error during tracking, the results were computed by averaging combinations of three balls among four.

1) System Capability Test by Measuring its Repeatability: The system capability that we care about mostly is its repeatability as the ability to sequentially position to the same target. To measure the system repeatability, we actuated the manipulator to reach eight different positions as shown in Fig.6.(b) $\sim$ (e), and four of them are bending with the same angle but on different planes as shown in Fig.6.(a). Each trial was repeated ten times. The results are presented in Fig.6.(f) and (g), where the position deviation is between 3 to 9.14 mm (0.75% $\sim$ 2.285% of full workspace), and the orientation deviation varies from 7.49° to less than 0.1°. It also appears a tendency that the closer the end-effector is to the initial posture, the smaller the deviation and therefore the higher the repetition accuracy. It is consistent with an intuition that to reach a position further from the initial posture requires more pressure and more force to balance the effect of gravity, and therefore the system becomes less stable than if it were held in a position that requires less effort.

B. Motion Tracking Test

A real-time motion tracking test was implemented on our self-developed manipulator, where seven points were tracked in space and repeated three times for each test as shown in Fig.7.(e). The position tracking error measured by distance between real and target point was 13.40 mm (3.35% of full workspace) in average as shown in Fig.7.(b)$\sim$(c), which is comparable to the repeatability accuracy, and therefore validated our control method.

The whole algorithm was implemented on Raspberry Pi (Raspberry Pi 4B with 4G RAM, Quad core Cortex-A72 64-bit SoC at 1.5GHz and Ubuntu 20.04 installed.) and the running time of finding the solution for target pose tracking was within 10 ms for each test as shown in Fig.7.(a), the fast speed of which proves its potential of working as a fundamental for advanced control algorithm.

V. Conclusions

In this work, an approach for solving inverse kinematics of a soft manipulator was proposed, based on which, a real-time motion tracking test was implemented and validated on a real platform. The approach is performed in three steps including workspace sampling, gridding and searching. By visualizing the workspace of the soft manipulator, complicated motion planning problem can be turned into an intuitive and straightforward space searching problem. The control accuracy and computational efficiency are known as two contradictory issues that favoring one means compromise to the other and in most cases, it cannot be changed, and choices need to be made once certain method has been selected. Based on our proposed method, the quantitative measurement of the relation between control accuracy and efficiency has enabled us the room for regulation and it can also work as a guidance for different situations.

Compared with other searching-based approaches, where efforts are needed in exhausting all the possible states of the system, it has been shown that a sparse sampling with order of thousands of data points should be sufficient since the inherent compliance of soft manipulators naturally constrains its accuracy. Therefore, what really matters is to make the sampling density match with the system capability as well as taking the real application scenario into considerations. As a result, fast searching can be realized and based on our motion tracking tests, it takes less than 10 ms to find a solution on Raspberry Pi and can therefore become the fundamental of further advanced control. It was validated that the average tracking error of our motion tracking test is around 13.40 mm (3.35% of full workspace), which is comparable with the system capability of holding still at one position as maximum ±4.57mm (1.14% of full workspace).

For further development, learning techniques can be applied based on the proposed method to refine the results or dealing

with dynamic tasks considering external loads or for manipulators with hyper-redundancy.

REFERENCES

[1] S. Cubero, *Industrial robotics: theory, modelling and control*. Pro Literatur Verlag, 2006.

[2] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[3] A. Mueller, "Modern robotics: Mechanics, planning, and control [bookshelf]," *IEEE Control Systems Magazine*, vol. 39, no. 6, pp. 100–102, 2019.

[4] P. Polygerinos, Z. Wang, K. C. Galloway, R. J. Wood, and C. J. Walsh, "Soft robotic glove for combined assistance and at-home rehabilitation," *Robotics and Autonomous Systems*, vol. 73, pp. 135–143, 2015.

[5] J. Hughes, U. Culha, F. Giardina, F. Guenther, A. Rosendo, and F. Iida, "Soft manipulators and grippers: a review," *Frontiers in Robotics and AI*, vol. 3, p. 69, 2016.

[6] C. Lee, M. Kim, Y. J. Kim, N. Hong, S. Ryu, H. J. Kim, and S. Kim, "Soft robot review," *International Journal of Control, Automation and Systems*, vol. 15, no. 1, pp. 3–15, 2017.

[7] J. Zhou, X. Chen, J. Li, Y. Tian, and Z. Wang, "A soft robotic approach to robust and dexterous grasping," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2018, pp. 412–417.

[8] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.

[9] D. B. Camarillo, C. R. Carlson, and J. K. Salisbury, "Task-space control of continuum manipulators with coupled tendon drive," in *Experimental Robotics*. Springer, 2009, pp. 271–280.

[10] ——, "Configuration tracking for continuum manipulators with coupled tendon drive," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 798–808, 2009.

[11] 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, 2008.

[12] V. Falkenhahn, A. Hildebrandt, R. Neumann, and O. Sawodny, "Model-based feedforward position control of constant curvature continuum robots using feedback linearization," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 762–767.

[13] J. M. Bern, Y. Schnider, P. Banzet, N. Kumar, and S. Coros, "Soft robot control with a learned differentiable model," in *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2020, pp. 417–423.

[14] T. G. Thuruthel, B. Shih, C. Laschi, and M. T. Tolley, "Soft robot perception using embedded soft sensors and recurrent neural networks," *Science Robotics*, vol. 4, no. 26, 2019.

[15] T. Thuruthel, E. Falotico, M. Cianchetti, F. Renda, and C. Laschi, "Learning global inverse statics solution for a redundant soft robot," in *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics*, vol. 2, 2016, pp. 303–310.

[16] T. George Thuruthel, Y. Ansari, E. Falotico, and C. Laschi, "Control strategies for soft robotic manipulators: A survey," *Soft robotics*, vol. 5, no. 2, pp. 149–163, 2018.

[17] F. Xu and H. Wang, "Soft robotics: Morphology and morphology-inspired motion strategy," *IEEE-CAA JOURNAL OF AUTOMATICA SINICA*, vol. 8, no. 9, pp. 1500–1522, 2021.

[18] 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, 2003.

[19] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 43–55, 2006.

[20] 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*. IEEE, 2012, pp. 3440–3446.

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

[22] 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, 2003.

[23] 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, 2014.

[24] T. Mahl, A. E. Mayer, A. Hildebrandt, and O. Sawodny, "A variable curvature modeling approach for kinematic control of continuum manipulators," in *2013 American Control Conference*. IEEE, 2013, pp. 4945–4950.

[25] X. Huang, J. Zou, and G. Gu, "Kinematic modeling and control of variable curvature soft continuum robots," *IEEE/ASME Transactions on Mechatronics*, pp. 1–1, 2021.

[26] V. Falkenhahn, T. Mahl, A. Hildebrandt, R. Neumann, and O. Sawodny, "Dynamic modeling of bellows-actuated continuum robots using the euler–lagrange formalism," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1483–1496, 2015.

[27] ——, "Dynamic modeling of constant curvature continuum robots using the euler-lagrange formalism," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 2428–2433.

[28] A. D. Marchese, R. Tedrake, and D. Rus, "Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator," *The International Journal of Robotics Research*, vol. 35, no. 8, pp. 1000–1019, 2016.

[29] G. Gao, H. Wang, J. Liu, and Y. Zheng, "Statics analysis of an extensible continuum manipulator with large deflection," *Mechanism and Machine Theory*, vol. 141, pp. 245–266, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0094114X19301806

[30] M. Gazzola, L. Dudte, A. McCormick, and L. Mahadevan, "Forward and inverse problems in the mechanics of soft filaments," *Royal Society open science*, vol. 5, no. 6, p. 171628, 2018.

[31] A. Doroudchi and S. Berman, "Configuration tracking for soft continuum robotic arms using inverse dynamic control of a cosserat rod model," in *2021 IEEE International Conference on Soft Robotics, RoboSoft*, 2021.

[32] 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.

[33] C. Della Santina, C. Duriez, and D. Rus, "Model based control of soft robots: A survey of the state of the art and open challenges," *arXiv preprint arXiv:2110.01358*, 2021.

[34] K. Wu and G. Zheng, "Fem-based gain-scheduling control of a soft trunk robot," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3081–3088, 2021.

[35] R. K. Katzschmann, M. Thieffry, O. Goury, A. Kruszewski, T.-M. Guerra, C. Duriez, and D. Rus, "Dynamically closed-loop controlled soft robotic arm using a reduced order finite element model with state observer," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2019, pp. 717–724.

[36] T. M. Bieze, F. Largilliere, A. Kruszewski, Z. Zhang, R. Merzouki, and C. Duriez, "Finite element method-based kinematics and closed-loop control of soft, continuum manipulators," *Soft robotics*, vol. 5, no. 3, pp. 348–364, 2018.

[37] G. Fang, M. C. Chow, J. D. Ho, Z. He, K. Wang, T. Ng, J. K. Tsoi, P.-L. Chan, H.-C. Chang, D. T.-M. Chan *et al.*, "Soft robotic manipulator for intraoperative mri-guided transoral laser microsurgery," *Science Robotics*, vol. 6, no. 57, p. eabg5575, 2021.

[38] X. Chen, Y. Guo, D. Duanmu, J. Zhou, W. Zhang, and Z. Wang, "Design and modeling of an extensible soft robotic arm," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4208–4215, 2019.

[39] X. Chen, D. Duanmu, and Z. Wang, "Model-based control and external load estimation of an extensible soft robotic arm," *Frontiers in Robotics and AI*, vol. 7, p. 215, 2021.

[40] S. Liu, Y. Zhu, Z. Zhang, Z. Fang, J. Tan, J. Peng, C. Song, H. Asada, and Z. Wang, "Otariidae-inspired soft-robotic supernumerary flippers by fabric kirigami and origami," *IEEE/ASME Transactions on Mechatronics*, 2020.