Shape Reconstruction of Soft Manipulators Using Vision and IMU Feedback

Harish Bezawada, Cole Woods and Vishesh Vikas

***Abstract*—In recent times, soft manipulators have gardened immense interest given their dexterous abilities. A critical aspect for their feedback control involves the reconstruction of the manipulator shape. The research, for the first time, presents shape reconstruction of a soft manipulator through sensor fusion of information available from Inertial Measurement Units (IMUs) and visual tracking. The manipulator is modeled using multi-segment continuous curvature Pythagorean Hodograph (PH) curves. PH curves are a class of continuous curvature curves with an analytical expression for the hodograph (slope). The shape reconstruction is formulated as an optimization problem that minimizes bending energy of the curve with a length constraint and the information from IMUs and/or visual markers. The paper experimentally investigates the robustness of shape reconstruction for scenarios when position of all visual markers, or slope at all the knots (placement of sensors) are known. Occlusion of manipulator segments is frequent, hence, this scenario is simulated by fusing information of available slopes (IMUs) at all knots and position (vision) at some knots. The experiments are performed on a planar tensegrity manipulator with IMUs feedback and visual tracking. The robustness study indicates reliability of these models for real world applications. Additionally, the proposed sensor fusion algorithm provides promising results where, for most cases, the shape estimates benefit from additional position information. Finally, the low dimensionality of the optimization problem argues for extension of the approach for real-time applications.**

***Index Terms*—Soft Sensors and Actuators; Soft Robot Applications; Modeling, Control, and Learning for Soft Robots**

I. Introduction

SIMILAR to an elephant trunk and octopus arms, soft and continuum manipulators aim to provide dexterity to manipulation - the ability to conform to different shapes [1], [2]. Consequently, they find applications in variety of fields including surgery [3], agriculture [4] and underwater robotics [1]. The manipulator shape, often referred to as the backbone, is modeled using multiple approaches including piecewise constant curvature (PCC) [5], [6] and cosserat rod models [7]. Thereafter, the reconstruction of the shape is critical for their feedback control. The terms shape reconstruction and estimation are often interchangeably used in literature.

The shape feedback is obtained using either contact-based (embedded inside the manipulator), or non-contact (e.g., IMUs, vision) sensors and is often geometrically dependent on the robot model. Fiber Bragg Gratings sensor feedback has been combined with PCC assumptions with and without consideration of torsion [8]–[10]. Inertial Measurement Units (IMUs) have been used for reconstructing a constant curvature segment of a continuum manipulator [11]. However, the constant curvature assumption is often violated during interaction with the environment and during dynamic movements. Ma et al. [12] proposed real time shape estimation using electromagnetic sensing and with Bézier curves to model the shape. Guo et al. [13] employed permanent magnets and magnetic sensors, and used the relative bending angle to fit a quadratic Bezier curve for planar cases. They found the terminal position error to be in between $0.227mm$ and $9.339mm$ for a $195mm$ manipulator. So et al. [14] used stretchable sensor made of Multiwalled Carbon Nanotube inside silicone, and considered the stretch and the curvature to estimate the shape using the Frenet-Serret formula representation. They used tip position as an error parameter and found max error of $23mm$ for $200mm$ manipulator when the deformation is more than 40% of the total length. However, all these approaches are intrusive and are affected by external factors including electromagnetic interference. Recently, Mbakop et al. [15] used inverse dynamic model based on Pythagorean Hodograph (PH) combined with Euler-Bernouli (EB) model and visual feedback for Hermite interpolation problem. Here, the soft robot arm manipulator arm is modeled as a single segment. AlBeladi et al. explored vision-based reconstruction using geometric strain parameterization [16].

One of the scenarios for feedback sensing includes placing sensors on the continuum manipulator at predetermined locations along the backbone. Here, unlike PCC, we consider modeling the manipulator as a piecewise (set of) *continuous* curvature curves. These curves satisfy the geometric continuities of position, slope and curvature at the junction of two curve segments. This approach is used extensively in computer graphics using cubic splines, B-splines and Non-uniform rational B-splines (NURBs) among other approaches [17]. However, an analytical solution does not exist for evaluating the length of the curve. This becomes crucial if we desire to impose a length constraint, e.g., sensors placed at constant distances along the curve. Pythagorean Hodograph (PH) curves are a special family of curves with an analytical solution to the length, and are discussed in Sec. II.

Contribution. The research presents modeling of a continuum manipulator using multiple PH curves where sensors are placed at the ends of each curve segment. The shape reconstruction is formulated as an optimization problem that minimizes bending energy of the curve with a length con-

---

Manuscript received: February 24, 2022; Revised: May 27, 2022; Accepted: June 28, 2022.

This paper was recommended for publication by Yong-Lae Park upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by the National Science Foundation under Grant No. 1830432 and 1832993.

Harish Bezawada, Cole Woods and Vishesh Vikas are with the Agile Robotics Lab (ARL), University of Alabama, Tuscaloosa AL 36487, USA (e-mail: hbezawada@crimson.ua.edu; cswoods@crimson.ua.edu; vvikas@ua.edu).

Digital Object Identifier (DOI): 10.1109/LRA.2022.3191523

straint. The sensor information from visual markers and IMUs is included as constraints. The research experimentally investigates the robustness of shape reconstruction for scenarios when position of all visual markers, or slope at all the segment knots are known. Occlusion of manipulator segments is frequent, hence, this scenario is simulated by fusing information of available slopes (IMUs) at all knots and position (vision) at some knots. The experiments are performed on a planar tensegrity manipulator with IMUs feedback and visual tracking. To the best of our knowledge, this approach has not been previously investigated in literature.

The rest of the paper is organized as: Sec. II introduces PH representation of the curve with position and length as constraints. The three scenarios are formulated as an optimization problem with different sets of constraints. Next, Sec. III presents the experiments on a planar tensegrity manipulator, numerical results and robustness analysis of slope based and position based shape estimation. It also discusses the sensor fusion of IMU and vision data. Finally, the conclusion and the future work are discussed in Sec. IV.

II. PARAMETRIC MODEL AND SHAPE RECONSTRUCTION

A. Pythagorean Hodograph (PH) Curves

As a requisite background for this paper, an informal and brief introduction of required concepts for PH curves is provided. The reader may refer to [18] for additional details. Conceptually, in our context where sensors are placed at predetermined constant positions, PH-curves are beneficial to obtain analytical (closed form) solutions for lengths of curves. The curve terminology and properties are summarized as

1) *Parametric representation* of the curve is $\boldsymbol{r}(s)$ with the normalized curvilinear coordinate $s \in [0,1]$

$$\boldsymbol{r}(s) = [x(s), y(s), z(s)]^T \tag{1}$$

However, the length of the curve L does not have a generic closed-form solution

$$L = \int_0^1 |\boldsymbol{r}'(s)| ds = \int_0^1 \sqrt{x'(s)^2 + y'(s)^2 + z'(s)^2} ds$$

2) *Hodograph* is the first derivative of the curve that satisfies the Pythagorean condition, i.e.,

$$\begin{aligned} \boldsymbol{r}'(s) &= [x'(s), y'(s), z'(s)]^T \\ \text{s.t.} \quad x'(s)^2 &+ y'(s)^2 + z'(s)^2 = \sigma(s)^2 \end{aligned} \tag{2}$$

The Pythagorean constraints facilitates an analytical (closed form) solution for the curve length.

$$L = \int_0^1 |\boldsymbol{r}'(s)| ds = \int_0^1 |\sigma(s)| ds \tag{3}$$

3) *Parametric hodographs* that allow for satisfaction of the Pythagorean condition are

$$\begin{aligned} x'(s) &= \left[u^2(s) + v^2(s) - p^2(s) - q^2(s)\right] \\ y'(s) &= 2\left[u(s)q(s) + v(s)p(s)\right] \\ z'(s) &= 2\left[v(s)q(s) - u(s)p(s)\right] \\ |\boldsymbol{r}'(s)| = |\sigma(s)| &= u^2(s) + v^2(s) + p^2(s) + q^2(s) \end{aligned} \tag{4}$$

where $u(s), v(s), p(s), q(s)$ are polynomials of any degree. Here, linear polynomials do not have any degree of freedom and are not feasible. Hence, quadratic polynomials with two degrees of freedom are considered which result in a *quintic PH-curve*.

4) We will use the Hopf representation[1] of the PH-curves using third degree complex polynomials $\boldsymbol{\alpha}(s), \boldsymbol{\beta}(s)$

$$\begin{gathered} \boldsymbol{\alpha}(s) = u(s) + iv(s), \boldsymbol{\beta}(s) = q(s) + ip(s) \\ \text{s.t.} \quad x'(s) = |\boldsymbol{\alpha}(s)|^2 - |\boldsymbol{\beta}(s)|^2 \\ y'(s) = 2Re\left(\boldsymbol{\alpha}(s)\bar{\boldsymbol{\beta}}(s)\right), z'(s) = 2Im\left(\boldsymbol{\alpha}(s)\bar{\boldsymbol{\beta}}(s)\right) \\ \sigma(s)^2 = |\boldsymbol{\alpha}(s)|^2 + |\boldsymbol{\beta}(s)|^2 \end{gathered} \tag{5}$$

where $\bar{\boldsymbol{\alpha}}(s)$ and $\bar{\boldsymbol{\beta}}(s)$ represent the complex conjugates.

B. Position Feedback for Shape Reconstruction

Problem definition. A soft manipulator has N sensors located at known intervals that give information about the position (absolute or relative) $\boldsymbol{p}_i$ and the segment length l_i as illustrated in Fig. 1. The shape is defined using segments of PH-curves $\boldsymbol{r}_i(s)$. It is desired to find the parametric representation of these curve segments that preserve positional C^0, tangential C^1 and curvature C^2 continuity, and are of constant length l_i.

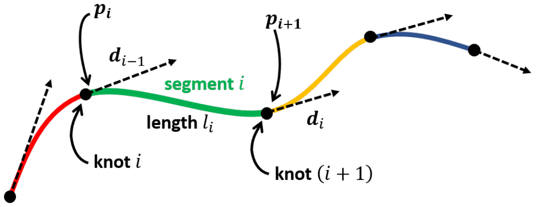

Fig. 1. Piecewise continuous modeling of a soft manipulator where the segment i between knots i and $i+1$ is of length l_i.

Parametric representation ensuring continuity. Let the Hopf form quadratic polynomials for a segment i, $(\boldsymbol{\alpha}_i(s), \boldsymbol{\beta}_i(s))$, be defined using complex coefficients $\boldsymbol{a}_i, \boldsymbol{b}_i, i = 0, \cdots, (N+1)$

$$\begin{aligned} \boldsymbol{\alpha}_i(s) &= \frac{(\boldsymbol{a}_{i-1} + \boldsymbol{a}_i)}{2}(1-s)^2 + 2\boldsymbol{a}_i(1-s)s + \frac{(\boldsymbol{a}_i + \boldsymbol{a}_{i+1})}{2}s^2 \\ \boldsymbol{\beta}_i(s) &= \frac{(\boldsymbol{b}_{i-1} + \boldsymbol{b}_i)}{2}(1-s)^2 + 2\boldsymbol{b}_i(1-s)s + \frac{(\boldsymbol{b}_i + \boldsymbol{b}_{i+1})}{2}s^2 \end{aligned} \tag{6}$$

It can be easily observed that this representation preserves C^1, C^2 continuity at the knots $i = 1, \cdots, N$, i.e.,

$$\boldsymbol{r}_i'(1) = \boldsymbol{r}_{i+1}'(0), \quad \boldsymbol{r}_i''(1) = \boldsymbol{r}_{i+1}''(0)$$

Hence, for the shape defined using segments of parameterized PH-curves $\boldsymbol{r}_i(s)$, it is desired to calculate $4(N+2) = (4N+8)$ unknowns corresponding to the complex $\boldsymbol{a}_i, \boldsymbol{b}_i, \forall i = 0, \cdots, (N+1)$.

---

[1] Alternatively, quarternion representation can also be followed.

Position constraints. Let the positions, i.e., the knots be $\boldsymbol{p}_i = (x_i, y_i, z_i)$. Each segment $\boldsymbol{r}_i(s)$ for $i = 1, ..., N$ is required to satisfy $\boldsymbol{r}_i(0) = \boldsymbol{p}_{i-1}$ and $\boldsymbol{r}_i(1) = \boldsymbol{p}_i$. Hence,

$$\int_0^1 \boldsymbol{r}_i'(s)ds = \boldsymbol{p}_i - \boldsymbol{p}_{i-1} = \Delta\boldsymbol{p}_i$$

$$\Rightarrow \Delta\boldsymbol{p}_i = \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ \Delta z_i \end{bmatrix} = \int_0^1 \begin{bmatrix} |\boldsymbol{\alpha}_i(s)|^2 - |\boldsymbol{\beta}_i(s)|^2 \\ 2Re\left(\boldsymbol{\alpha}_i(s)\bar{\boldsymbol{\beta}}_i(s)\right) \\ 2Im\left(\boldsymbol{\alpha}_i(s)\bar{\boldsymbol{\beta}}_i(s))\right) \end{bmatrix} ds$$

The above can be expressed in complex coefficients as

$$\begin{aligned} \Delta x_i = \frac{1}{120}\Big[&6\left(|\boldsymbol{a}_{i-1}|^2 - \boldsymbol{b}_{i-1}|^2\right) + 54\left(|\boldsymbol{a}_i|^2 - \boldsymbol{b}_i|^2\right) \\ &+6\left(|\boldsymbol{a}_{i+1}|^2 - \boldsymbol{b}_{i+1}|^2\right) + 13(\boldsymbol{a}_{i-1}\bar{\boldsymbol{a}}_i - \boldsymbol{b}_{i-1}\bar{\boldsymbol{b}}_i + \bar{\boldsymbol{a}}_{i-1}\boldsymbol{a}_i \\ &-\bar{\boldsymbol{b}}_{i-1}\boldsymbol{b}_i + \boldsymbol{a}_i\bar{\boldsymbol{a}}_{i+1} - \boldsymbol{b}_i\bar{\boldsymbol{b}}_{i+1} + \bar{\boldsymbol{a}}_i\boldsymbol{a}_{i+1} - \bar{\boldsymbol{b}}_i\boldsymbol{b}_{i+1}) \\ &+\boldsymbol{a}_{i-1}\bar{\boldsymbol{a}}_{i+1} - \boldsymbol{b}_{i-1}\bar{\boldsymbol{b}}_{i+1} + \bar{\boldsymbol{a}}_{i-1}\boldsymbol{a}_{i+1} + \bar{\boldsymbol{b}}_{i-1}\boldsymbol{b}_{i+1}\Big] \end{aligned} \tag{7}$$

$$\begin{aligned} (\Delta y_i + i\Delta z_i) = \frac{1}{60}\Big[&6\boldsymbol{a}_{i-1}\bar{\boldsymbol{b}}_{i-1} + 54\boldsymbol{a}_i\bar{\boldsymbol{b}}_i + 6\boldsymbol{a}_{i+1}\bar{\boldsymbol{b}}_{i+1} \\ &+13\left(\boldsymbol{a}_{i-1}\bar{\boldsymbol{b}}_i + \boldsymbol{a}_{i+1}\bar{\boldsymbol{b}}_i + \boldsymbol{a}_i\bar{\boldsymbol{b}}_{i-1} + \boldsymbol{a}_i\bar{\boldsymbol{b}}_{i+1}\right) \\ &+\boldsymbol{a}_{i-1}\bar{\boldsymbol{b}}_{i+1} + \boldsymbol{a}_{i+1}\bar{\boldsymbol{b}}_{i-1}\Big] \end{aligned} \tag{8}$$

For a segment i, the nonlinear equations (7), (8) are summarized as

$$\boldsymbol{g}_i\left(\boldsymbol{a}_{i-1}, \boldsymbol{a}_i, \boldsymbol{a}_{i+1}, \boldsymbol{b}_{i-1}, \boldsymbol{b}_i, \boldsymbol{b}_{i+1}\right) - \Delta\boldsymbol{p}_i = 0 \tag{9}$$

where $\boldsymbol{g}_i, \Delta\boldsymbol{p}_i \in \mathbb{R}^{3\times1}$. Collectively, for all the N segments,

$$\begin{aligned} \boldsymbol{g} - \Delta\boldsymbol{p} = 0 \quad \text{s.t.} \quad &\boldsymbol{g} = [\boldsymbol{g}_1^T, \boldsymbol{g}_2^T, \cdots, \boldsymbol{g}_N^T]^T, \\ \Delta\boldsymbol{p} = &\left[\Delta\boldsymbol{p}_1^T, \cdots, \Delta\boldsymbol{p}_N^T\right]^T \quad \in \mathbb{R}^{3N\times1} \end{aligned} \tag{10}$$

Length constraints. Let the length of a segment i be l_i, then analytically

$$\begin{aligned} L_i = \int_0^1 |\boldsymbol{r}_i'(s)|ds = \frac{1}{120}\Big[&6(|\boldsymbol{a}_{i-1}|^2 + |\boldsymbol{b}_{i-1}|^2) + \\ 54(|\boldsymbol{a}_i|^2 + |\boldsymbol{b}_i|^2) + 6(|\boldsymbol{a}_{i+1}|^2 &+ |\boldsymbol{b}_{i+1}|^2) + 13(\boldsymbol{a}_{i-1}\bar{\boldsymbol{a}}_i \\ +\boldsymbol{b}_{i-1}\bar{\boldsymbol{b}}_i + \bar{\boldsymbol{a}}_{i-1}\boldsymbol{a}_i + \bar{\boldsymbol{b}}_{i-1}\boldsymbol{b}_i &+ \boldsymbol{a}_i\bar{\boldsymbol{a}}_{i+1} + \boldsymbol{b}_i\bar{\boldsymbol{b}}_{i+1} + \bar{\boldsymbol{a}}_i\boldsymbol{a}_{i+1} \\ +\bar{\boldsymbol{b}}_i\boldsymbol{b}_{i+1}) + \boldsymbol{a}_{i-1}\bar{\boldsymbol{a}}_{i+1} + \boldsymbol{b}_{i-1}\bar{\boldsymbol{b}}_{i+1} &+ \bar{\boldsymbol{a}}_{i-1}\boldsymbol{a}_{i+1} + \bar{\boldsymbol{b}}_{i-1}\boldsymbol{b}_{i+1}\Big] \end{aligned}$$

$$\Rightarrow L_i\left(\boldsymbol{a}_{i-1}, \boldsymbol{a}_i, \boldsymbol{a}_{i+1}, \boldsymbol{b}_{i-1}, \boldsymbol{b}_i, \boldsymbol{b}_{i+1}\right) - l_i = 0 \tag{11}$$

For N segments, this can be assembled as N equations

$$\begin{aligned} \boldsymbol{L} - \boldsymbol{l} = 0 \quad \text{s.t.} \quad &\boldsymbol{L} = [L_1, L_2, \cdots, L_N]^T \\ &\boldsymbol{l} = [l_1, l_2, \cdots, l_N]^T \in \mathbb{R}^{N\times1} \end{aligned} \tag{12}$$

Equations (10), (12) make up to $4N$ nonlinear constraints to solve for $4N + 8$ unknowns. Hence, there are 8 degrees of freedom. A cost function, global energy, is defined that sums the bending energy of each segment

$$\begin{aligned} f(\boldsymbol{a}, \boldsymbol{b}) = \sum_{i=1}^{N}\Big(&|\boldsymbol{a}_{i-1} - 2\boldsymbol{a}_i + \boldsymbol{a}_{i+1}|^2 \\ &+ |\boldsymbol{b}_{i-1} - 2\boldsymbol{b}_i + \boldsymbol{b}_{i+1}|^2\Big) \end{aligned} \tag{13}$$

Now, the shape reconstruction problem is formulated as a constrained optimization problem that minimizes the energy function for the $(4N + 8)$ unknowns

$$\min_{\boldsymbol{a},\boldsymbol{b}} f(\boldsymbol{a}, \boldsymbol{b}) \quad \text{s.t.} \quad (\boldsymbol{g} - \Delta\boldsymbol{p}) = 0, \ (\boldsymbol{L} - \boldsymbol{l}) = 0 \tag{14}$$

Given the nonlinear nature of the problem, the initialization of the coefficients is performed as discussed in Appendix A. This is a nonlinear optimization problem that will be solved using numerical solvers like MATLAB®.

C. Slope Feedback for Shape Reconstruction

Problem definition. The problem is defined in the same manner as in the previous subsection. The curve is parameterized in the same manner, the same length constraints are valid. In this case, the derivative of the curve (direction cosines) are known at the knots $\boldsymbol{d}_i = [dx_i, dy_i, dz_i]^T \ \forall i = 1, 2, \cdots, (N + 1)$

$$\boldsymbol{d}_1 = \boldsymbol{r}_1'(0) = \underbrace{\begin{bmatrix} \frac{1}{4}(|\boldsymbol{a}_0 + \boldsymbol{a}_1|^2 - |\boldsymbol{b}_0 + \boldsymbol{b}_1|^2) \\ Re\left(\frac{1}{2}(\boldsymbol{a}_0 + \boldsymbol{a}_1)(\bar{\boldsymbol{b}}_0 + \bar{\boldsymbol{b}}_1)\right) \\ Im\left(\frac{1}{2}(\boldsymbol{a}_0 + \boldsymbol{a}_1)(\bar{\boldsymbol{b}}_0 + \bar{\boldsymbol{b}}_1)\right) \end{bmatrix}}_{\boldsymbol{h}_1(\boldsymbol{a}_0,\boldsymbol{a}_1,\boldsymbol{b}_0,\boldsymbol{b}_1)}$$

$$\boldsymbol{d}_{i+1} = \boldsymbol{r}_i'(1) = \underbrace{\begin{bmatrix} \frac{1}{4}(|\boldsymbol{a}_i + \boldsymbol{a}_{i+1}|^2 - |\boldsymbol{b}_i + \boldsymbol{b}_{i+1}|^2) \\ Re\left(\frac{1}{2}(\boldsymbol{a}_i + \boldsymbol{a}_{i+1})(\bar{\boldsymbol{b}}_i + \bar{\boldsymbol{b}}_{i+1})\right) \\ Im\left(\frac{1}{2}(\boldsymbol{a}_i + \boldsymbol{a}_{i+1})(\bar{\boldsymbol{b}}_i + \bar{\boldsymbol{b}}_{i+1})\right) \end{bmatrix}}_{\boldsymbol{h}_{i+1}(\boldsymbol{a}_i,\boldsymbol{a}_{i+1},\boldsymbol{b}_i,\boldsymbol{b}_{i+1})} \forall i \in [1, N]$$

Hence, the slope constraints can be summarized as

$$\begin{aligned} \boldsymbol{h} - \boldsymbol{d} = 0 \quad \text{s.t.} \quad &\boldsymbol{h} = [\boldsymbol{h}_1^T, \boldsymbol{h}_2^T, \cdots, \boldsymbol{h}_{N+1}^T]^T, \\ &\boldsymbol{d} = \left[\boldsymbol{d}_1^T, \boldsymbol{d}_2^T, \cdots, \boldsymbol{d}_{N+1}^T\right]^T \quad \in \mathbb{R}^{3(N+1)\times1} \end{aligned} \tag{15}$$

In contrast to the last case, here, the number of constraints is $3(N + 1) + N = (4N + 3)$. Hence, having five degrees of freedom. Consequently, the shape reconstruction problem is modified to minimize the energy function for the $(4N + 8)$ unknowns with $(4N + 3)$ nonlinear constraints

$$\min_{\boldsymbol{a},\boldsymbol{b}} f(\boldsymbol{a}, \boldsymbol{b}) \quad \text{s.t.} \quad (\boldsymbol{h} - \boldsymbol{d}) = 0, \ (\boldsymbol{L} - \boldsymbol{l}) = 0 \tag{16}$$

The initialization of coefficients $\boldsymbol{a}, \boldsymbol{b}$ is similar to the previous case and the modified approach is given in Appendix B.

D. Algorithmic Summary

We summarize the shape estimation process using position or slope feedback as:

(Input) The lengths l_i of N segments and $\{$position $\boldsymbol{p}_i$ OR slopes $\boldsymbol{d}_i\}$ at the $(N + 1)$ knots

Initializing $(4N + 8)$ unknowns

(Step 1) Find θ_i using $\sin\theta_i$, $\cos\theta_i$ from (23) OR (26)

(Step 2) Find the ϕ_i by assuming $\phi_1 = 0$ and calculating $\Delta\phi_i$ using (25) OR (27)

(Step 3) Initialize $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ where $i = 1, ..., N$. Coefficients $\boldsymbol{a}_0, \boldsymbol{a}_{n+1}, \boldsymbol{b}_0$ and $\boldsymbol{b}_{n+1}$ are initialized as

$$\boldsymbol{a}_0 = 2\boldsymbol{a}_1 - \boldsymbol{a}_2, \quad \boldsymbol{b}_0 = 2\boldsymbol{b}_1 - \boldsymbol{b}_2$$
$$\boldsymbol{a}_{n+1} = 2\boldsymbol{a}_n - \boldsymbol{a}_{n-1}, \quad \boldsymbol{b}_{n+1} = 2\boldsymbol{b}_n - \boldsymbol{b}_{n-1}$$

Optimize the constrained cost function

(Step 4) The nonlinear constrained energy cost function, (14) OR (16), minimized using off the shelf solvers, e.g., MATLAB®

(Step 5) Calculate $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ using (6)

(Step 6) Each curve segment is obtained using

$$\boldsymbol{r}_i(s) = \int_0^1 \begin{bmatrix} |\boldsymbol{\alpha}_i(s)|^2 - |\boldsymbol{\beta}_i(s)|^2 \\ 2Re\left(\boldsymbol{\alpha}_i(s)\bar{\boldsymbol{\beta}}_i(s)\right) \\ 2Im\left(\boldsymbol{\alpha}_i(s)\bar{\boldsymbol{\beta}}_i(s)\right)\end{bmatrix} ds$$

where $\boldsymbol{r}_i(1) = \boldsymbol{r}_{i+1}(0), \ \forall i \in [1, N-1]$ to ensure C^0 continuity.

E. Sensor Fusion: Slope and Position Feedback

When equipping manipulators with visual markers and IMUs, the scenarios of occlusion of the marker or possibility of failure of on-board sensor may arise. Here, we specifically simulate the scenario of visual occlusion where information from all IMUs are available while the visual marker positions are missing. For the case of availability of positions and slopes at all knots, the problem is modified to

$$\min_{\boldsymbol{a},\boldsymbol{b}} f(\boldsymbol{a}, \boldsymbol{b}) \text{ s.t. } (\boldsymbol{g} - \Delta \boldsymbol{p}) = 0, \quad (\boldsymbol{L} - \boldsymbol{l}) = 0, (\boldsymbol{h} - \boldsymbol{d}) = 0$$

where the cost function with $(7N + 3)$ nonlinear constraints is minimized for $(4N + 8)$ unknowns.

Next, we examine the case of visual occlusion when partial position information is available. For brevity, we refer to this as $\widetilde{\boldsymbol{g}}$ and $\Delta\widetilde{\boldsymbol{p}}$. For example, if the positions of markers corresponding to only segments 1 and 4 is known, then $\widetilde{\boldsymbol{g}} = [\boldsymbol{g}_1^T, \boldsymbol{g}_4^T]^T, \Delta\widetilde{\boldsymbol{p}} = [\Delta\widetilde{\boldsymbol{p}}_1^T, \Delta\widetilde{\boldsymbol{p}}_4^T]^T \in \mathbb{R}^{6\times 1}$. Concisely, when marker positions for M segments are known

$$\widetilde{\boldsymbol{g}} - \Delta\widetilde{\boldsymbol{p}} = 0, \quad \text{s.t.} \quad \widetilde{\boldsymbol{g}}, \widetilde{\boldsymbol{p}} \in \mathbb{R}^{3M\times 1}$$

Hence, the sensor fusion problem with position information available for M segments and slope for $(N + 1)$ knots, is summarized as

$$\begin{gathered}\min_{\boldsymbol{a},\boldsymbol{b}} f(\boldsymbol{a}, \boldsymbol{b}) \quad \text{s.t.} \\ (\boldsymbol{L} - \boldsymbol{l}) = 0, \quad (\boldsymbol{h} - \boldsymbol{d}) = 0, \quad (\widetilde{\boldsymbol{g}} - \Delta\widetilde{\boldsymbol{p}}) = 0\end{gathered} \tag{17}$$

where the number of nonlinear constraint equations are $N + (3N + 3) + 3M$

III. EXPERIMENTS AND RESULTS

A. Experimental Testbed: Sensorized Tensegrity Manipulator

Tensegrity mechanisms synergistically combine compression (rigid) and tension (elastic cables) to achieve structural integrity. We construct a tensegrity manipulator inspired by the human spine that is soft and agile, Fig. 2(a). The manipulator is fabricated by serially combining 10 vertebrae modules. Each vertebra is constructed using wood (rigid element), and elastic nylon cord (tensile element) that is held in place using M5 nuts and M5x12mm socket head cap screw. The tendon lengths (in green) can be varied to change the manipulator shape. Similar design concepts have been explored in literature [19], [20]. The manipulator was divided into three segments comprising of four knots with the length between consecutive knots of $175mm$ and base-to-last knot length of $525mm$. Each knot has a $6mm \times 3mm$ rectangle-shape visual marker (tracked using web-camera) and a BNO055 9-dof IMU (AHRS), Fig. 2(b). The BNO055 is a consumer-grade motion sensor that has shown to have minimal drift ($< 1\deg$) in static orientations over a period of 24 hours [21]. This makes them ideal candidates for the experiments in this research. The IMUs are connected to an Arduino Nano and the data is communicated to the computer via serial port. The IMUs measure the orientation w.r.t. earth coordinate system with y-axis aligned with the magnetic north and z-axis along the gravity. The rotation matrix $R_{ab} \in SO(3)$ (transforms vector in $\{b\}$ to $\{a\}$ coordinate system) is constructed using quarternion or Euler angle output of the sensor [22]. For an IMU, let E, C, t be the coordinate systems of earth, camera (w.r.t. which measurements will be made), and of the sensor at time t. For calibration, the manipulator is placed on the horizontal table such that the sensor z-axis aligns with the gravity vector and x-axis along the table length to obtain R_{EC}. However, at this orientation, the manipulator is in an unstable position, hence, was moved to a new stable orientation at time t_0 to obtain R_{Et_0}. At any time t, the sensor measures R_{Et} and the rotation matrix between t_0 and t represented in $\{C\}$ coordinate system is obtained by

$$R^C_{t_0 t} = R^T_{EC} R_{Et} R^T_{Et_0} R_{EC} \tag{18}$$

Derivation: We use Lie groups and notations in [22] for deriving (18). The rotation matrix $R_{t_0 t} = R^T_{Et_0} R_{Et} = \exp(\widehat{\omega}^{t_0}\theta)$ represents the rotation of angle θ along axis ω^{t_0} where superscript t_0 is the coordinate system of representation, and $\widehat{\omega} \in so(3)$. The desired rotation matrix $R^C_{t_0 t} = \exp(\widehat{\omega}^C\theta)$ where $\omega^C = R_{Ct_0}\omega^{t_0}$. Hence, $R^C_{t_0 t} = R_{Ct_0} R_{t_0 t} R^T_{Ct_0}$. ■

A combination of camera image, Autodesk® Fusion 360 and AutoCAD® is used to construct the ground truth (benchmark measurement) curve for comparison with experimental data. First, the camera image with visual markers is imported into Fusion 360. To construct the ground truth curve, quintic (5th degree) splines are fit between consecutive knots (three segments) to ensure that the resulting curve visually coincides with the backbone of the tensegrity manipulator. Here, the slope and positions at each knot are exported to MATLAB®. Finally, this true curve is imported into AutoCAD® and each segment is discretized into 100 intervals before exporting the data into MATLAB®. Traditionally the estimates at each knot are taken into consideration for experimental comparison without considering the curve profile between them [14]–[16]. In contrast, this approach constructs the ground truth for the whole parametric curve.

We consider four planar canonical poses with positive and negative curvature, change in curvature from positive to

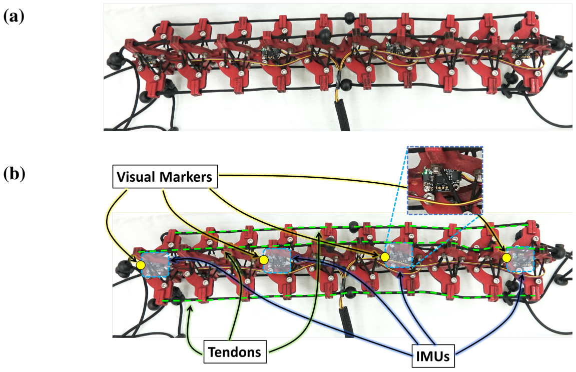

Fig. 2. (a) A highly deformable, sensorized tensegrity manipulator (b) with IMUs and visual markers placed at known intervals along the backbone.

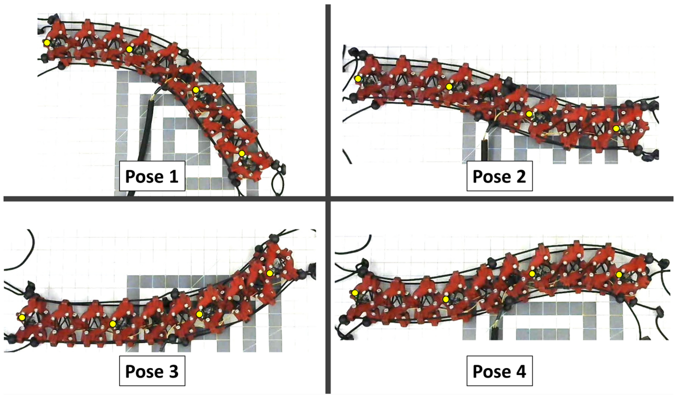

Fig. 3. Different planar canonical poses considered in the experiment.

negative and vice-versa, as shown in Fig. 3. These poses were obtained by varying the tendon lengths.

B. Error Metrics

For evaluating the accuracy of the shape reconstruction, we consider two error metrics - the root mean square (RMS) error between two curves $\boldsymbol{r}^1(s), \boldsymbol{r}^2(s)$ and the error of the final knot of the respective curves $\boldsymbol{p}^1_{N+1}, \boldsymbol{p}^2_{N+1}$ with N segments.

$$e_{rms}\left(\boldsymbol{r}^1(s), \boldsymbol{r}^2(s)\right) = \sqrt{\frac{1}{l}\int_0^l \left|\boldsymbol{r}^1(s) - \boldsymbol{r}^2(s)\right|^2 ds} \tag{19}$$

$$e_{end\ knot}\left(\boldsymbol{p}^1_{N+1}, \boldsymbol{p}^2_{N+1}\right) = \left|\boldsymbol{p}^1_{N+1} - \boldsymbol{p}^2_{N+1}\right|^2 \tag{20}$$

Here, the curve RMS error is numerically calculated by discretizing the parametric curves. For the case of robustness analysis, the errors are compared between estimated curve and the curves with noisy measurements, i.e., $e^R_{rms}\left(\boldsymbol{r}^*(s), \boldsymbol{r}^{*,noisy}(s)\right), e^R_{end\ knot}\left(\boldsymbol{p}^*_{N+1}, \boldsymbol{p}^{*,noisy}_{N+1}\right)$. However, the sensor fusion results (along with vision-only and slope-only cases) are analyzed using the error between the true and estimated curves, i.e, $e^{SF}_{rms}\left(\boldsymbol{r}^{true}(s), \boldsymbol{r}^*(s)\right), e^{SF}_{end\ knot}\left(\boldsymbol{p}^{true}_{N+1}, \boldsymbol{p}^*_{N+1}\right)$.

C. Robustness Analysis

For the cases of position or slope feedback, the robustness of the algorithm with respect to noise in the measurement of positions and slopes is examined.

Position robustness. Corresponding to the size of the visual markers, a $\pm 3mm$ random noise was added to the true knot positions to simulate the uncertainties encountered during position measurement. Thereafter, 100 samples were generated for each pose and the errors were evaluated. Fig. 4 visualizes the results and Tab. I compares the maximum RMS and end knot errors ($\max e^R_{rms}$ and $\max e^R_{end\ knot}$ respectively) of the estimated curves with noisy measurements.

Slope robustness. Since the IMUs have an accuracy of ± 2.5 degrees [23], 5 degree random noise is added into the measured visual slopes. Each pose was reconstructed with added noise for 100 samples. The results for all the poses are as visualized in Fig. 5, and tabulated in Tab. II.

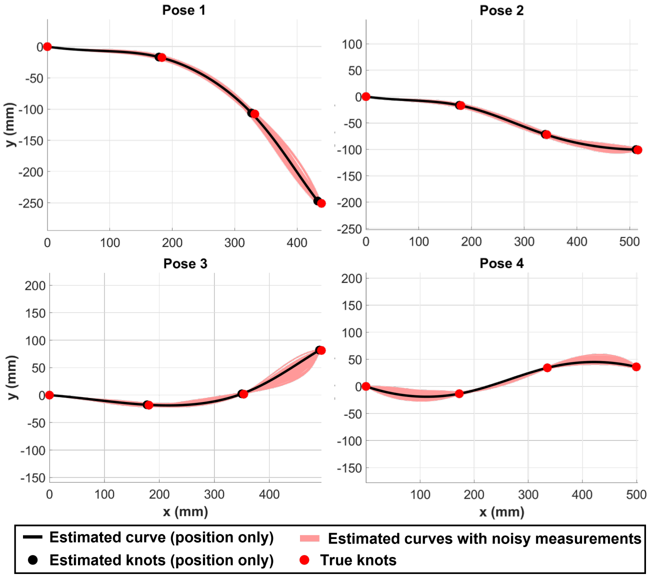

Fig. 4. Robustness of the reconstructed shape for uncertainty in position feedback: Random noise of $\pm 3mm$ was added to the position of each knot and 100 samples are plotted (light red).

TABLE I
POSITION ROBUSTNESS

Pose	max $e^R_{end\ knot}$ (mm, % length)	max e^R_{rms} (mm, % length)
Pose 1	4.63 (0.88%)	1.66 (0.32%)
Pose 2	5.08 (0.97%)	1.42 (0.27%)
Pose 3	4.95 (0.94%)	2.23 (0.43%)
Pose 4	4.33 (0.82%)	1.82 (0.35%)

TABLE II
SLOPE ROBUSTNESS

Pose	Max. $e^R_{end\ knot}$ (mm, % length)	Max. e^R_{rms} (mm, % length)
Pose 1	15.78 (3.01%)	2.05 (0.39%)
Pose 2	18.50 (3.25%)	2.22 (0.42%)
Pose 3	14.98 (2.85%)	2.22 (0.42%)
Pose 4	18.18 (3.46%)	2.20 (0.42%)

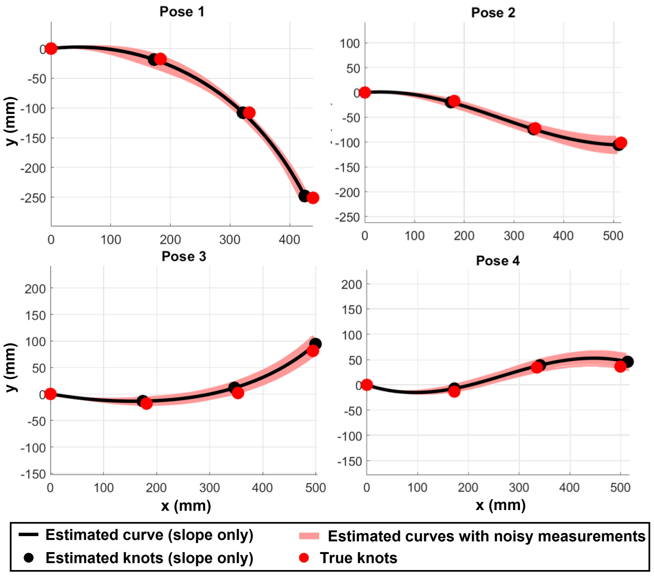

Fig. 5. Robustness of the reconstructed shape for uncertainty in slope feedback - random noise of ± 2.5 deg was added at each knot and 100 samples are plotted.

This analysis indicates the stability and robustness of this approach to sensor measurement noise - $\max e^{R}_{end\ knot}$ of 0.97% and 3.46%, and $\max e^{R}_{rms}$ of 0.43% and 0.42% for position and slope measurements respectively. The analysis also confirms the importance of visual feedback, i.e., more accurate shape estimation using position feedback.

D. Sensor Fusion

At any instant in time, IMU measurements (slopes) are always known, however due to visual occlusion, the position data may not be available. This scenario is simulated by considering multiple cases when position of the markers is known or not. For imposing position constraints on a segment, the end positions of the segments are required as observed in (9). Conceptually, given any two positions and the length between those positions, shape reconstruction can be performed. However, for this paper, we limit ourselves only to cases when consecutive positions are known. We denote the position knowledge in the following manner: $Pijk$ implies $\{\boldsymbol{p}_i, \boldsymbol{p}_j, \boldsymbol{p}_k\}$ are known. Consequently, out of the four knots for the problem considered in this study, there will be similar results for certain cases. One example of such a similar set is $P12 \iff P124$.

The experimental results for the four test cases are visualized and tabulated in Fig. 6 and Tab. III respectively. Here, the sensor fusion results are compared with those from slope-only (IMU) and position-only (vision). The sensor fusion results for different scenarios, when compared to position-only and slope-only feedback, are very interesting. First, the vision-only results are most accurate, and the sensor fusion is expected to improve upon the shape reconstructed using only the slope data at the knots. This is observed for Poses 3 and 4, however, not for all cases in Poses 1 and 2. This behavior of the shape reconstruction for the case of sensor fusion is hypothesized to be as a result of the nature of the optimization function (13). This over-constrained cost function only considers the geometric properties (bending energy) and does not include any information about the material properties. Consequently, there are instances when the estimates 'over-fit' the data, e.g., case P123 for Pose 2. Hence, a clear pattern regarding knowledge of knot positions does not emerge, e.g., is the information about knot positions toward the end of the manipulator more beneficial than at the base? Nonetheless, in majority of the cases, the knowledge of the knot position does contribute to improvement of the shape reconstruction. Apart from incorporating manipulator-specific material properties, different techniques for modifying the cost function may be explored for regularization of the estimate, e.g., weights for the data available for individual segments or inclusion of slack in the constraints as a function of measurement noise.

Computationally, the optimization of the cost functions, (14),(16),(17), on an average, takes $0.015\,\mathrm{sec}$ on an Intel i7 $2.9GHz$ processor with $16GB$ RAM. Hence, the proposed methodology is applicable for real-time, dynamic shape reconstruction.

The proposed approach involving use of multiple PH-curves for modeling soft manipulator with IMU and vision feedback is the state-of-the-art technique for shape reconstruction. Single segment cubic splines have been explored as geometric models, however, the shape reconstruction based on sensor-feedback is lacking [24]. PH-curve based models have been explored by [15] and [25] where the former considers length constraints while the latter does not. In both cases, the $330mm$ manipulator is modeled using single segment curve and only vision-based feedback is considered for reconstruction. These approaches have average errors of 8.6% and 4.9%, in contrast to maximum error of 1.39% of the manipulator length for the vision-based approach presented in this paper. Furthermore, unlike existing techniques, the proposed approach can perform reconstruction using IMUs, and a combination of IMUs and vision feedback while allowing for occlusion scenarios.

IV. CONCLUSION

In this paper, soft manipulators are modeled using multi-segment continuous curvature PH curves with position, slope and curvature continuity. This special family of curves has an analytical expression for its hodograph that facilitates the imposition of a constant length constraint, which otherwise is not possible. A soft tensegrity manipulator is sensorized with IMUs and visual markers placed at constant distances along its backbone. Manipulator shape reconstruction is formulated as an optimization problem with nonlinear constraints of length, position and slope. The robustness analysis explores the stability of the algorithm to noisy sensor measurements, and indicates reliability of these models for real world applications. Motivated by the problem of occlusion of visual markers, for the first time in literature, a sensor fusion model for simultaneously incorporating vision and IMU measurements is proposed. The sensor fusion results are promising and indicate that additional visual information positively impacts shape reconstruction. However, it also highlights cases of 'over-fitting' when the estimate is less accurate. This is possibly

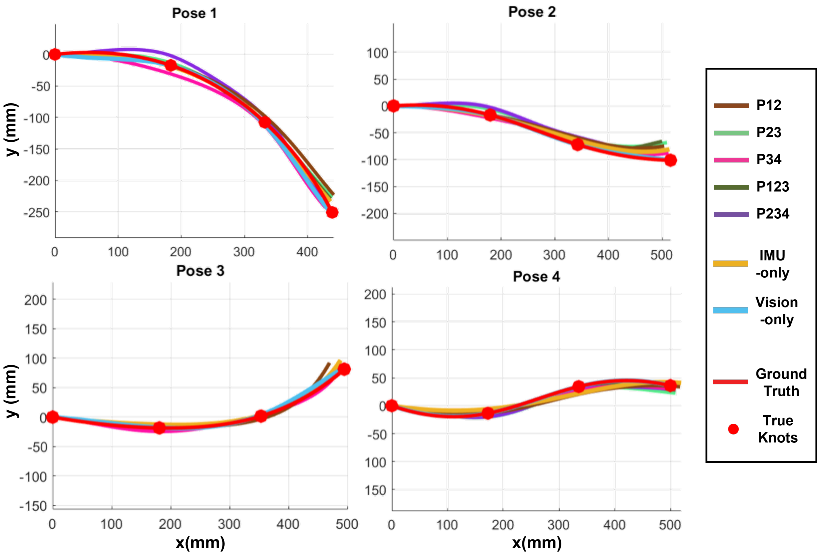

Fig. 6. Shape reconstruction for the case of sensor fusion when all the slopes at the knots are known and while the positions are partially known.

TABLE III

Sensor Fusion Errors - e^{SF}_{rms} and $e^{SF}_{end\ knot}$ (mm, % length)

Test case		**Sensor Fusion**					**Slope only (IMU)**	**Position only (vision)**
		P12	**P23**	**P34**	**P123**	**P234**		
Pose 1	RMS	12.13 (2.31%)	12.34 (2.35%)	13.30 (2.53%)	8.86 (1.69%)	20.68 (3.94%)	10.36 (1.97%)	7.32 (1.39%)
	End knot	27.94 (5.32%)	23.21 (4.42%)	13.19 (2.51%)	21.57 (4.11%)	25.20 (4.80%)	17.54 (3.34%)	7.08 (1.35%)
Pose 2	RMS	12.03 (2.29%)	15.03 (2.86%)	9.36 (1.78%)	12.07 (2.30%)	16.06 (3.06%)	8.61 (1.64%)	3.96 (0.75%)
	End knot	29.20 (5.56%)	33.72 (6.42%)	14.40 (2.74%)	38.51 (7.34%)	20.47 (3.90%)	22.29 (4.24%)	3.51 (0.67%)
Pose 3	RMS	8.53 (1.62%)	5.49 (1.05%)	6.96 (1.33%)	3.94 (0.75%)	5.69 (1.08%)	6.70 (1.28%)	4.15 (0.79%)
	End knot	27.21 (5.18%)	11.35 (2.16%)	7.93 (1.51%)	12.40 (2.36%)	7.27 (1.39%)	16.72 (3.19%)	3.57 (0.68%)
Pose 4	RMS	10.97 (2.09%)	9.45 (1.80%)	9.46 (1.80%)	6.29 (1.20%)	7.35 (1.40%)	12.17 (2.32%)	1.12 (0.21%)
	End knot	17.53 (3.34%)	16.22 (3.09%)	11.23 (2.14%)	12.35 (2.35%)	6.25 (1.19%)	20.03 (3.82%)	0.00 (0.00%)

due to lack of incorporation of material properties in the optimization cost function. In summary, the approach can be applied to a manipulator or a soft robot of any length to reconstruct its shape. Here, the low dimensional nature of the optimization problem will facilitate its application to real-time scenarios.

The future work involves reconstruction of shape in spatial (3D) and dynamic conditions. Additionally, the modification of the cost function to include material properties and weighing of different sensor measurements will be explored.

REFERENCES

[1] M. Cianchetti, A. Arienti, M. Follador, B. Mazzolai, P. Dario, and C. Laschi, "Design concept and validation of a robotic arm inspired by the octopus," *Materials Science and Engineering: C*, vol. 31, pp. 1230–1239, Aug. 2011.

[2] I. D. Walker, D. M. Dawson, T. Flash, F. W. Grasso, R. T. Hanlon, B. Hochner, W. M. Kier, C. C. Pagano, C. D. Rahn, and Q. M. Zhang, "Continuum robot arms inspired by cephalopods," vol. 5804, pp. 303–314, 2005.

[3] S. K. Sahu, C. Sozer, B. Rosa, I. Tamadon, P. Renaud, and A. Menciassi, "Shape Reconstruction Processes for Interventional Application Devices: State of the Art, Progress, and Future Directions," *Frontiers in Robotics and AI*, vol. 8, no. November, pp. 1–26, 2021.

[4] N. K. Uppalapati, B. T. Walt, A. J. Havens, A. Mahdian, G. Chowdhary, and G. Krishnan, "A Berry Picking Robot With A Hybrid Soft-Rigid Arm: Design and Task Space Control," in *Robotics: Science and Systems*, 2020.

[5] D. Trivedi, A. Lotfi, and C. Rahn, "Geometrically Exact Models for Soft Robotic Manipulators," *IEEE Transactions on Robotics*, vol. 24, pp. 773–780, Aug. 2008.

[6] R. J. Webster and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[7] F. Renda, F. Boyer, J. Dias, and L. Seneviratne, "Discrete Cosserat Approach for Multisection Soft Manipulator Dynamics," *IEEE Transactions on Robotics*, vol. 34, pp. 1518–1533, Dec. 2018. Conference Name: IEEE Transactions on Robotics.

[8] H. Wang, R. Zhang, W. Chen, X. Liang, and R. Pfeifer, "Shape Detection Algorithm for Soft Manipulator Based on Fiber Bragg Gratings," *IEEE/ASME Transactions on Mechatronics*, vol. 21, pp. 2977–2982, Dec. 2016. Publisher: Institute of Electrical and Electronics Engineers Inc.

[9] Y. He, L. Zhu, G. Sun, M. Yu, and M. Dong, "Design, measurement and shape reconstruction of soft surgical actuator based on Fiber Bragg Gratings," *Applied Sciences (Switzerland)*, vol. 8, no. 10, 2018. ISBN: 8601084176.

[10] Q. Hou, C. Lu, and X. Li, "A novel pose sensing model for soft manipulator based on helical FBG," *Sensors and Actuators, A: Physical*, vol. 321, Apr. 2021. Publisher: Elsevier B.V.

[11] J. Hughes, F. Stella, C. D. Santina, and D. Rus, "Sensing soft robot shape using imus: An experimental investigation," in *International Symposium on Experimental Robotics*, pp. 543–552, Springer, 2020.

[12] X. Ma, P. W. Y. Chiu, and Z. Li, "Real-time deformation sensing for

flexible manipulators with bending and twisting," *IEEE Sensors Journal*, vol. 18, no. 15, pp. 6412–6422, 2018. Publisher: IEEE.

[13] H. Guo, F. Ju, Y. Cao, F. Qi, D. Bai, Y. Wang, and B. Chen, "Continuum robot shape estimation using permanent magnets and magnetic sensors," *Sensors and Actuators, A: Physical*, vol. 285, pp. 519–530, 2019. Publisher: Elsevier B.V.

[14] J. So, U. Kim, Y. B. Kim, D.-Y. Seok, S. Y. Yang, K. Kim, J. H. Park, S. T. Hwang, Y. J. Gong, and H. R. Choi, "Shape Estimation of Soft Manipulator Using Stretchable Sensor," *Cyborg and Bionic Systems*, vol. 2021, pp. 1–10, Apr. 2021. Publisher: American Association for the Advancement of Science (AAAS).

[15] S. Mbakop, G. Tagne, M. H. Frouin, M. Achille, and R. Merzouki, "Inverse dynamics model-based shape control of soft continuum finger robot using parametric curve," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8053–8060, 2021. Publisher: IEEE.

[16] A. AlBeladi, G. Krishnan, M.-A. Belabbas, and S. Hutchinson, "Vision-based shape reconstruction of soft continuum arms using a geometric strain parametrization," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11753–11759, IEEE, 2021.

[17] G. Farin, J. Hoschek, and M.-S. Kim, *Handbook of computer aided geometric design*. Elsevier, 2002.

[18] R. T. Farouki, *Pythagorean-Hodograph Curves: Algebra and Geometry Inseparable*. Geometry and Computing, Berlin Heidelberg: Springer-Verlag, 2008.

[19] A. P. Sabelhaus, H. Ji, P. Hylton, Y. Madaan, C. Yang, A. M. Agogino, J. Friesen, and V. SunSpiral, "Mechanism design and simulation of the ultra spine: a tensegrity robot," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, 2015.

[20] D. Zappetti, R. Arandes, E. Ajanic, and D. Floreano, "Variable-stiffness tensegrity spine," *Smart Materials and Structures*, vol. 29, p. 075013, June 2020. Publisher: IOP Publishing.

[21] Z. Lin, Y. Xiong, H. Dai, and X. Xia, "An Experimental Performance Evaluation of the Orientation Accuracy of Four Nine-Axis MEMS Motion Sensors," in *2017 5th International Conference on Enterprise Systems (ES)*, pp. 185–189, Sept. 2017. ISSN: 2572-6609.

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

[23] B. Sensortec, "Intelligent 9-axis absolute orientation sensor," no. June 2016, pp. 1–106, 2016.

[24] M. Wiese, K. Rüstmann, and A. Raatz, "Kinematic modeling of a soft pneumatic actuator using cubic hermite splines," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7176–7182, IEEE, 2019.

[25] I. Singh, Y. Amara, A. Melingui, P. Mani Pathak, and R. Merzouki, "Modeling of continuum manipulators using pythagorean hodograph curves," *Soft Robotics*, vol. 5, no. 4, pp. 425–442, 2018.

APPENDIX A
INITIALIZATION OF COEFFICIENTS FOR POSITION FEEDBACK

The numerical optimization of (14), (16) requires initialization of unknowns $\boldsymbol{a}, \boldsymbol{b}$. The three-fold approach involves

1) *Approximating the constraints in terms of unknowns*. The relationship between consecutive a_i, b_i is assumed to be

$$\begin{aligned} \boldsymbol{a}_{i-1} &= \boldsymbol{a}_i - \delta\boldsymbol{a}_i, \quad & \boldsymbol{b}_{i-1} &= \boldsymbol{b}_i - \delta\boldsymbol{b}_i \\ \boldsymbol{a}_{i+1} &= \boldsymbol{a}_i + \delta\boldsymbol{a}_{i+1}, \quad & \boldsymbol{b}_{i+1} &= \boldsymbol{b}_i + \delta\boldsymbol{b}_{i+1} \end{aligned} \tag{21}$$

Substituting (21) in (9), (11), and ignoring the higher order terms corresponding to $\delta a_i, \delta b_i$,

$$\begin{bmatrix} |\boldsymbol{a}_i|^2 - |\boldsymbol{b}_i|^2 \\ 2Re(\boldsymbol{a}_i\bar{\boldsymbol{b}}_i) \\ 2Im(\boldsymbol{a}_i\bar{\boldsymbol{b}}_i) \end{bmatrix} = \Delta\boldsymbol{p}_i, \quad |\boldsymbol{a}_i|^2 + |\boldsymbol{b}_i|^2 = l_i \tag{22}$$

2) *Solving approximate constraint equations.* A set of solutions satisfying (22) is

$$\begin{gathered} |\boldsymbol{a}_i|^2 = \frac{1}{2}\psi_i(|\Delta\boldsymbol{p}_i| + \Delta x_i), \ |\boldsymbol{b}_i|^2 = \frac{1}{2}\psi_i(|\Delta\boldsymbol{p}_i| - \Delta x_i) \\ \boldsymbol{a}_i\bar{\boldsymbol{b}}_i = \frac{1}{2}\psi_i(\Delta y_i + i\Delta z_i), \ \text{s.t. } \psi_i = \frac{l_i}{|\Delta\boldsymbol{p}_i|} \end{gathered}$$

where $\boldsymbol{a}_i, \boldsymbol{b}_i$ are

$$\begin{gathered} \boldsymbol{a}_i = \sqrt{\frac{1}{2}\psi_i(|\Delta\boldsymbol{p}_i| + \Delta x_i)}\exp^{i(\theta_i+\phi_i)} \\ \boldsymbol{b}_i = \sqrt{\frac{1}{2}\psi_i(|\Delta\boldsymbol{p}_i| - \Delta x_i)}\exp^{i(\phi_i)}, \ \text{s.t. } \theta_i = \arctan\left(\frac{\Delta z_i}{\Delta y_i}\right) \\ \sin\theta_i = \frac{\Delta z_i}{\sqrt{\Delta y_i^2 + \Delta z_i^2}}, \cos\theta_i = \frac{\Delta y_i}{\sqrt{\Delta y_i^2 + \Delta z_i^2}} \end{gathered} \tag{23}$$

and ϕ_i is a free parameter.

3) *Calculating the free parameters.* We assume $\phi_1 = 0$ and the remaining are obtained by minimizing $\left(|\Delta\boldsymbol{a}_i|^2 + |\Delta\boldsymbol{b}_i|^2\right)$ to minimize the errors of approximation assumption. For ease of calculation, $\Delta\phi_i = \phi_i - \phi_{i-1}$ and $\Delta\theta_i = \theta_i - \theta_{i-1}$

$$\begin{gathered} |\Delta\boldsymbol{a}_i|^2 + |\Delta\boldsymbol{b}_i|^2 = \psi_{i-1}|\Delta\boldsymbol{p}_{i-1}| + \psi_i|\Delta\boldsymbol{p}_i| - \sqrt{\psi_{i-1}\psi_i} \\ \Big(\sqrt{(\Delta\boldsymbol{p}_{i-1} - \Delta x_{i-1})(\Delta\boldsymbol{p}_i - \Delta x_i)}\cos(\Delta\phi_i) \\ + \sqrt{(\Delta\boldsymbol{p}_{i-1} + \Delta x_{i-1})(\Delta\boldsymbol{p}_i + \Delta x_i)}\cos(\Delta\theta_i + \Delta\phi_i)\Big) \end{gathered} \tag{24}$$

Analytically, the minima is found equating the derivative of the above expression with respect to $\Delta\phi_i$ with zero.

$$\tan(\Delta\phi_i^*) = \frac{-\sin(\Delta\theta_i)}{\cos(\Delta\theta_i) + \sqrt{\frac{(\Delta\boldsymbol{p}_{i-1} - \Delta x_{i-1})(\Delta\boldsymbol{p}_i - \Delta x_i)}{(\Delta\boldsymbol{p}_{i-1} + \Delta x_{i-1})(\Delta\boldsymbol{p}_i + \Delta x_i)}}} \tag{25}$$

Solving for ϕ_i from the above expression will yield two values due to the square root present in the denominator. The value yielding the minimum when substituted into (24) is taken.

APPENDIX B
INITIALIZATION OF COEFFICIENTS FOR SLOPE FEEDBACK

$$\begin{gathered} \boldsymbol{a}_i = \sqrt{\frac{1}{2}\psi_i(|\boldsymbol{d}_i| + dx_i)}\exp^{i(\theta_i+\phi_i)} \\ \boldsymbol{b}_i = \sqrt{\frac{1}{2}\psi_i(|\boldsymbol{d}_i| - dx_i)}\exp^{i(\phi_i)} \\ \text{where } \theta_i = \arctan\left(\frac{dz_i}{dy_i}\right), \quad \psi_i = \frac{l_i}{|\boldsymbol{d}_i|} \\ \sin\theta_i = \frac{dz_i}{\sqrt{dy_i^2 + dz_i^2}}, \quad \cos\theta_i = \frac{dy_i}{\sqrt{dy_i^2 + dz_i^2}} \end{gathered} \tag{26}$$

$$\tan(\Delta\phi_i^*) = \frac{-\sin(\Delta\theta_i)}{\cos(\Delta\theta_i) + \sqrt{\frac{(\boldsymbol{d}_{i-1} - dx_{i-1})(\boldsymbol{d}_i - dx_i)}{(\boldsymbol{d}_{i-1} + dx_{i-1})(\boldsymbol{d}_i + dx_i)}}} \tag{27}$$