Observer-based Control of Inflatable Robot with Variable Stiffness

Conference Paper · July 2020

DOI: 10.1109/IROS45743.2020.9341785

5 authors, including:

Ahmad Ataka Awwalur Rizqi
Universitas Gadjah Mada
52 PUBLICATIONS **488** CITATIONS
SEE PROFILE

Taqi Abrar
Queen Mary, University of London
21 PUBLICATIONS **133** CITATIONS
SEE PROFILE

Fabrizio Putzu
Queen Mary, University of London
13 PUBLICATIONS **134** CITATIONS
SEE PROFILE

Godaba Hareesh
University of Sussex
36 PUBLICATIONS **1,126** CITATIONS
SEE PROFILE

Observer-based Control of Inflatable Robot with Variable Stiffness*

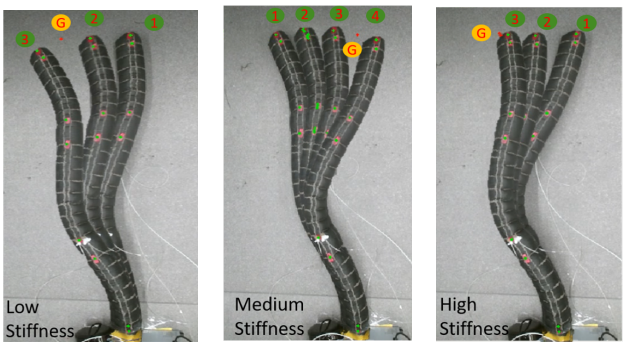
Ahmad Ataka, Taqi Abrar, Fabrizio Putzu, Hareesh Godaba, and Kaspar Althoefer

***Abstract*— In the last decade, soft robots have been at the forefront of a robotic revolution. Due to the flexibility of the soft materials employed, soft robots are equipped with a capability to execute new tasks in new application areas - beyond what can be achieved using classical rigid-link robots. Despite these promising properties, many soft robots nowadays lack the capability to exert sufficient force to perform various real-life tasks. This has led to the development of stiffness-controllable inflatable robots instilled with the ability to modify their stiffness during motion. This new capability, however, poses an even greater challenge for robot control. In this paper, we propose a model-based kinematic control strategy to guide the tip of an inflatable robot arm in its environment. The bending of the robot is modelled using an Euler-Bernoulli beam theory which takes into account the variation of the robot's structural stiffness. The parameters of the model are estimated online using an observer based on the Extended Kalman Filter (EKF). The parameters' estimates are used to approximate the Jacobian matrix online and used to control the robot's tip considering also variations in the robot's stiffness. Simulation results and experiments using a fabric-based planar 3-degree-of-freedom (DOF) inflatable manipulators demonstrate the promising performance of the proposed control algorithm.**

I. Introduction

Using soft materials as a robot structure has introduced a radical change in the robotics field in the last decade. The inherent flexibility of soft materials employed allows these robots to execute various tasks which were not possible to be achieved by rigid-link robots. These new tasks include maneuvering in a tight space [1] and performing robust grasping of unknown objects [2]. However, this inherent compliance comes at the price of the soft robot's limited capability in exerting high forces to the surrounding environment.

The drawback of purely-soft robotic systems (e.g., those made from rubber silicone) has led to the emergence of variable-stiffness robots in recent years. This type of soft robot has the ability to actively modify its structural stiffness, and hence, is able to vary the contact forces applied to the environment over a wide range as a function of the robot's stiffness [3]-[4]. As presented in a comparative study in [5], this property can be achieved via an antagonistic approach, i.e. stiffening by means of force equilibrium between multiple force components (such as [6]), or by exploiting an intrinsic property of the materials employed (such as [7]). An example of a variable-stiffness robot is an "inflatable robot" which exploits antagonistic pneumatic and tendon actuation to achieve simultaneous position and stiffness control [6], [8]. Another type of an inflatable robot is the "eversion" robot which is typified by a growing-tip movement [9], [10]. This class of inflatable robots has been used for various applications including laparoscopic surgery [6], reconfiguration of antenna [11], and exploring cluttered environments [12].



(a) (b) (c)

Fig. 1. The snapshots of the robot when the robot's stiffness is (a) low, (b) medium, and (c) high. The order of the movement is from a shape labelled 1, 2, and 3. The target is a red dot labelled G.

One of the challenges in controlling a soft robot in general is the difficulty in modelling the highly nonlinear robot structure. The fact that the inflatable robot is able to modify its structural stiffness online further complicates the modelling process. The quest to develop an accurate yet computationally fast robot model has led to the emergence of a model-less control approach which exploits sensor data to build online the robot model for control purposes. Examples of this approach include kinematic control based on a Jacobian estimation using an optimization technique [13], an adaptive Kalman filter [14], and visual servoing methods [15]. None of these works, however, take into account the variation of the robot's stiffness. Another challenge in controlling soft robots is caused by its theoretically infinite number degrees of freedom. An observer-based controller has been reported to help estimating unknown states of the robot and, in turn, facilitate the control algorithm [16]-[19]. Despite these advances, to the best knowledge of the authors, no observer-based controller has been implemented for stiffness-controllable soft robots.

Efforts to control an inflatable robot described in the literature were limited to apical extension [11] and basic

*This work was supported in part by the EPSRC in the framework of the NCNR (National Centre for Nuclear Robotics) project (EP/R02572X/1) and q-bot led Innovate UK project WormBot (2308/104059). This work was also receiving support from the European Union in the framework of FP7 project STIFF-FLOP (287728).

A. Ataka, T. Abrar, F. Putzu, H. Godaba, K. Althoefer are with the Centre for Advanced Robotics @ Queen Mary (ARQ), Faculty of Science and Engineering, Queen Mary University of London, Mile End Road, London E1 4NS, United Kingdom.

Corresponding author e-mail: a.rizqi@qmul.ac.uk

steering control [9] for the growing robot. Simple position control of an inflatable growing robot has also been reported in [12] while another research work aimed at exploiting the contact with the environment to navigate towards the target [20]. Recently, a combination of visual servoing and growth control is used to perform position control, as reported in [21]. These works, however, focus only on the growing and bending ability without considering the stiffness variation capability. A simultaneous position and stiffness controller [22] and an obstacle avoidance [23] for a stiffness-controllable robot have also been reported. However, none of these works takes into account the control of the tip's orientation which is vital in real-life applications such as object grasping and pick-and-place tasks.

In this paper, a model-based kinematic controller for the tip's pose of an inflatable robot is presented. The variation of robot's structural stiffness caused by the change of inflation pressure is taken into account when modelling the bending angle. In contrast to our previous work [24], an observer based on the Extended Kalman Filter (EKF) is designed to estimate the unknown parameters of the bending model online, making use of two sets of sensory data, i.e. chamber pressure and bending angle. By estimating the model's parameters, we then compute an estimation of the Jacobian matrix used for position and orientation control of the manipulator's tip under various robot stiffness conditions. To evaluate the performance of the proposed algorithm, we implemented the control strategy in a fabric-based planar 3-DOF inflatable manipulators as shown in Fig. 1. Hence, our contributions are as follows:

- designing an observer to develop a kinematic model of an inflatable manipulator online which considers variations of the robot's structural stiffness, and
- exploiting the model parameter estimate from the observer to perform a position and orientation control algorithm under various robot stiffness conditions.

II. Inflatable Robot

An inflatable robot is a type of soft robot that has the ability to actively modify its shape and structural stiffness. In this paper, our inflatable robot prototype is composed of unstretchable fabric which is formed into a long cylindrical shape. The robot's size is very compact in its deflated, folded state. However, upon inflation, the robot structure is able to elongate significantly and, at the same time, increase its structural stiffness considerably.

A bending mechanism for this type of robot has been designed in a recent work of ours, exploiting the use of pneumatically-actuated cylindrical pouches which are integrated into the main body of the robot [24]. These pouches initially form a flat surface in their deflated state. Upon inflation, the air exert a force in every direction, forcing the pouch to morph from its flat state into a cylindrical shape. Since the used fabric material cannot be stretched, the deformation of the pouches causes a contraction as illustrated in Fig 2a. Integrating two sets of parallel pouches on either side of the robot and inflating these sets pouches one at a time causes a length difference between one and the other side of the robot. This will create a right or left bending. Moreover, integrating multiple pairs of pouches along the robot body enables the robot to have multiple independent bending segments as illustrated in Fig. 2b.

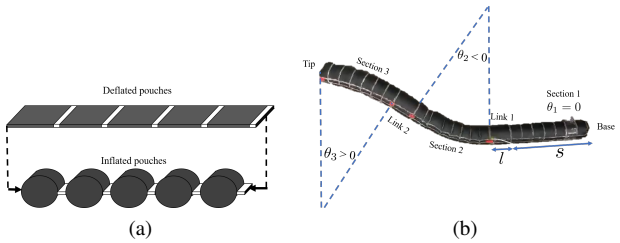


Fig. 2. (a) A set of pouches in a deflated state and inflated state. (b) An inflatable manipulator with 3 bending segments and 2 links.

III. Inflatable Robot Model

In this paper, every bending segments of the inflatable robot consists of two sets of pouches which are activated one at a time to produce left or right bending. We assume that each individual pouch produces a tension force f which is proportional to the given pressure p, i.e. $f \propto p$. The bending moment produced by this pouch with respect to the robot's central axis can be approximated by $M = rf \approx rap$, where r stands for a distance between the robot's central axis and the point where the pouch's tension is applied while a is a constant which relates pressure p to tension f. Considering a 1-DOF bending segment, we define $p > 0$ to indicate left bending and $p < 0$ to indicate right bending.

Each bending segments is modelled as a circular arc with a single curvature κ defined as $\kappa = \frac{\theta}{s}$, where θ is the bending angle and s is the length of the bending segment. Assuming that each segment behaves like an Euler-Bernoulli beam, its curvature is given by $\kappa = \frac{M}{EI}$, where E stands for the Young Modulus of the material and I stands for the cross-sectional moment of inertia. The bending angle θ can then be expressed as

$$\theta = \frac{Ms}{EI} \approx \frac{rasp}{EI}. \tag{1}$$

Inflation pressure p_0 affects the stiffness of an inflatable beam, which in turn affects the bending capability. The equivalent flexural rigidity of the inflatable beam with an axial force P caused by the internal inflation pressure p_0 is expressed as $(EI)_{eq} = (E + P/S_0)I$ where S_0 stands for the cross-sectional area [25]. The bending angle in (1) can then be rewritten as

$$\theta = \frac{rasp}{(E + P/S_0)I}. \tag{2}$$

Compared to the original bending equation in (1), we can see that the flexural stiffness of the beam is a function of the axial force P which is proportional to the main chamber's pressure p_0. Considering the fact that parameters r, a, s, E, I, S_0 are all constant and $P \propto p_0$, eq. (2) is simplified into

$$\theta = \frac{c_1 p}{1 + c_2 p_0}, \tag{3}$$

where c_1 and c_2 are positive constants.

Suppose that the robot consists of N bending sections. We define an actuator space variable $\mathbf{p}_{act}$ as $\mathbf{p}_{act} = \begin{bmatrix} p_0 & \mathbf{p} \end{bmatrix}^T$. Here, p_0 denotes the main chamber pressure while $\mathbf{p}$ denotes pouch pressure in all segments defined as $\mathbf{p} = \begin{bmatrix} p_1 & \dots & p_N \end{bmatrix}^T$ where p_i denotes the pouch pressure in segment-i. The bending angle variables for all segments are described by $\Theta = \begin{bmatrix} \theta_1 & \theta_2 & \dots & \theta_N \end{bmatrix}^T$. Using the bending model in (3), we can map $\mathbf{p}_{act}$ to the bending angle variable Θ as follows

$$\Theta = \mathbf{f}(\mathbf{p}_{act}) = \begin{bmatrix} \frac{c_{11}p_1}{1+c_{21}p_0} & \dots & \frac{c_{1N}p_N}{1+c_{2N}p_0} \end{bmatrix}^T, \tag{4}$$

where c_{1i} and c_{2i} represent the bending model parameters for segment-i.

Assuming constant curvature behavior, the tip's pose of any segment-i with respect to the segment's base can be expressed as a transformation matrix $\mathbf{T}_i^b \in SE(2)$ as follows

$$\mathbf{T}_i^b = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & \frac{(1-\cos\theta_i)}{\theta_i}s \\ \sin\theta_i & \cos\theta_i & \frac{\sin\theta_i}{\theta_i}s \\ 0 & 0 & 1 \end{bmatrix}^T. \tag{5}$$

A soft link without pouches with length l is used to connect two adjacent bending segments as depicted in Fig. 2b for $N = 3$. Suppose that $\mathbf{T}^l \in SE(2)$ denotes the tip's pose of each links with respect to its base. The tip's pose of the most distal segment with respect to the robot's base is given by

$$\mathbf{T}^t = \begin{bmatrix} \mathbf{R} & \mathbf{x} \\ \mathbf{0} & 1 \end{bmatrix} = (\prod_{i=1}^{N-1} \mathbf{T}_i^b \mathbf{T}^l)\mathbf{T}_N^b. \tag{6}$$

For simplicity, the task-space position $\mathbf{x} \in \mathbb{R}^3$ and orientation θ^t (derived from the rotational matrix $\mathbf{R} \in SO(2)$) can be written as a function of bending angle variable Θ as follows: $\mathbf{X} = \begin{bmatrix} \mathbf{x} & \theta^t \end{bmatrix}^T = \mathbf{g}(\Theta)$.

The Jacobian of the system $\mathbf{J}$ consists of the actuator-space Jacobian $\mathbf{J_f}$ and the task-space Jacobian $\mathbf{J_g}$ as follows:

$$\mathbf{J} = \mathbf{J_g}\mathbf{J_f} = \frac{\partial \mathbf{g}(\Theta)}{\partial \Theta}\frac{\partial \mathbf{f}(\mathbf{p}_{act})}{\partial \mathbf{p}}. \tag{7}$$

For simplicity, the forward kinematics in (5)-(6) can be used to derive the task-space Jacobian $\mathbf{J_g}$ numerically. The actuator-space Jacobian $\mathbf{J_f}$ can be derived analytically from (4) and expressed as a diagonal matrix:

$$\mathbf{J_f} = \text{diag}(\frac{c_{11}}{1+c_{21}p_0}, \dots, \frac{c_{1N}}{1+c_{2N}p_0}). \tag{8}$$

Note that only pouch pressure $\mathbf{p}$ is used to derive the Jacobian in (7). The main chamber pressure p_0 is excluded in the calculation because it is a user-chosen value as will be further explained in the next section.

IV. Observer-based Jacobian Estimation

Before developing a control policy for the robot, Jacobian $\mathbf{J}$ needs to be identified. Task-space Jacobian $\mathbf{J_g}$ can be retrieved by using the geometrical properties of the robot, i.e. the segment length s and the link length l. Actuator-space Jacobian $\mathbf{J_f}$, however, depends on the unknown parameters c_{1i} and c_{2i} for all bending segments $i = \{i \in \mathbb{Z} | 1 \geq i \geq N\}$. An observer is designed to perform an online estimation of these parameters to get the Jacobian $\mathbf{J_f}$ in (8).

The kinematic model presented in Section III can be reformulated as a non-linear discrete state space equation. For the purpose of parameter estimation, the state of the robot χ is chosen to be the unknown parameters from the bending model described in (4). For N bending sections, the full state space is given by $\chi \in \mathbb{R}^{2N} = \begin{bmatrix} c_{11} & c_{21} & \dots & c_{1N} & c_{2N} \end{bmatrix}^T$. Output measurement $\mathbf{y}$ is the bending angle $\Theta \in \mathbb{R}^N$ retrieved from bending sensors while input $\mathbf{u}$ is actuator space variable $\mathbf{p}_{act} \in \mathbb{R}^{N+1}$ which consists of main chamber pressure p_0 and pouch pressure $\mathbf{p} \in \mathbb{R}^N$. Therefore, the kinematic model of the robot can be reformulated as follows

$$\chi_{k+1} = f(\chi_k, \mathbf{u}_k) = \chi_k, \tag{9}$$

$$\mathbf{y}_k = g(\chi_k, \mathbf{u}_k) = \begin{bmatrix} [\theta_1]_k & \dots & [\theta_N]_k \end{bmatrix}^T, \tag{10}$$

where χ_k, $\mathbf{u}_k$, and $\mathbf{y}_k$ stand for the state of the robot, input signal, and output measurement at iteration-k. The bending angle of segment-i at iteration k is given by

$$[\theta_i]_k = \frac{[\chi_{2i-1}]_k[u_{i+1}]_k}{1+[\chi_{2i}]_k[u_1]_k}, \tag{11}$$

where $[\chi_i]_k$ and $[u_i]_k$ are components of χ_k and $\mathbf{u}_k$ in row-i.

An observer, based on the extended Kalman filter (EKF), is used to estimate the states as follows

$$\begin{aligned} \hat{\chi}_{k+1|k} &= f(\hat{\chi}_k, \mathbf{u}_k), \\ \mathbf{P}_{k+1|k} &= \mathbf{A}_k\mathbf{P}_k\mathbf{A}_k^T + \mathbf{Q}_k, \\ \mathbf{S}_k &= \mathbf{C}_k(\hat{\chi}_{k+1|k})\mathbf{P}_{k+1|k}\mathbf{C}_k(\hat{\chi}_{k+1|k})^T + \mathbf{R}_k \\ \mathbf{K}_k &= \mathbf{P}_{k+1|k}\mathbf{C}_k(\hat{\chi}_{k+1|k})^T\mathbf{S}_k^{-1}, \\ \hat{\chi}_{k+1} &= \hat{\chi}_{k+1|k} + \mathbf{K}_k(\mathbf{y}_k - g(\hat{\chi}_{k+1|k}, \mathbf{u}_k)), \\ \mathbf{P}_{k+1} &= (\mathbf{I} - \mathbf{K}_k\mathbf{C}_k(\hat{\chi}_{k+1|k}))\mathbf{P}_{k+1|k}. \end{aligned} \tag{12}$$

$\hat{\chi}_k$ stands for the state estimate while $\mathbf{P}_k \in \mathbb{R}^{2N\times 2N}$, $\mathbf{Q}_k \in \mathbb{R}^{2N\times 2N}$, $\mathbf{R}_k \in \mathbb{R}^{N\times N}$ stand for the estimation covariance, process noise variance and measurement noise variance matrix, respectively. Matrix $\mathbf{A}_k \in \mathbb{R}^{2N\times 2N}$ and $\mathbf{C}_k(\hat{\chi}_{k+1|k}) \in \mathbb{R}^{N\times 2N}$ are a local linearisation of (9) and (10) given by $\mathbf{A}_k = \frac{\partial f(\chi_k, \mathbf{u}_k)}{\partial \chi_k} = \mathbf{I} \in \mathbb{R}^{2N\times 2N}$ while $\mathbf{C}_k(\hat{\chi}_{k+1|k}) = \frac{\partial g(\hat{\chi}_{k+1|k}, \mathbf{u}_k)}{\partial \chi_k}$ and is given by

$$\mathbf{C}_k(\hat{\chi}) = \begin{bmatrix} d_1(\hat{\chi}) & h_1(\hat{\chi}) & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & d_2(\hat{\chi}) & h_2(\hat{\chi}) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & d_N(\hat{\chi}) & h_N(\hat{\chi}) \end{bmatrix}, \tag{13}$$

$$\begin{aligned} d_i(\hat{\chi}_k) &= \frac{[u_{i+1}]_k}{1+[\hat{\chi}_{2i}]_k[u_1]_k}, \\ h_i(\hat{\chi}_k) &= -\frac{[\hat{\chi}_{2i-1}]_k[u_1]_k[u_{i+1}]_k}{(1+[\hat{\chi}_{2i}]_k[u_1]_k)^2}. \end{aligned} \tag{14}$$

Due to the non-linearity of the bending angle model in (10)-(11), "persistently-exciting" input is required to allow

the state estimate in (12) to converge [26]. However, for a specific value of main chamber pressure p_0, bending angle θ_i at segment-i in (4) is linear to pouch pressure p_i, i.e.

$$\theta_i = \frac{c_{1i}}{1+c_{2i}p_0}p_i = b_i(p_0)p_i. \tag{15}$$

The linearity of this equation for a specific value of $[u_1]_k = p_0$ ensures that the estimate of variable b_i at step k, given by $[\hat{b}_i]_k = \frac{[\chi_{2i-1}]_k}{1+[\chi_{2i}]_k[u_1]_k}$ will converge without requiring a persistently-exciting input signal. The estimate of the Jacobian in (8) can then be expressed as

$$[\hat{\mathbf{J}}_\mathbf{f}]_k = \text{diag}([\hat{b}_1]_k, \ldots, [\hat{b}_N]_k). \tag{16}$$

V. Observer-based Control

The estimated Jacobian is used to control the tip's pose towards a target while taking into account variations in the structural stiffness of the robot due to the inflation pressure p_0. We assume that tip position $\mathbf{x}$ and bending angle Θ (from which the tip orientation θ^tcan be derived) are retrieved from sensor measurements. To ensure the tip to follow a desired trajectory $\mathbf{x}_d(t)$, the following control law is employed

$$\mathbf{v} = -K_P(\mathbf{x}-\mathbf{x}_d) - \dot{\mathbf{x}}_d, \tag{17}$$

where $K_P > 0$ is a constant.

For the position and orientation control task, an additional angular velocity $\dot{\theta}$ is added to task-space velocity $\mathbf{v}$. A geometric-based control function using [24] is employed:

$$\hat{\omega} = -K_\omega \log(\mathbf{R}_e). \tag{18}$$

Here, $\mathbf{R}_e = \mathbf{R}_d^T\mathbf{R}$ where $\mathbf{R} \in SO(2)$ and $\mathbf{R}_d \in SO(2)$ denote the rotation matrices which correspond to tip orientation θ^t and desired orientation θ_d, respectively. Parameter $K_\omega > 0$ is a proportional constant. Operator $\log(\mathbf{R})$ for any $\mathbf{R} \in SO(2)$ is defined as $\log(\mathbf{R}) = \frac{\beta}{2\sin\beta}(\mathbf{R}-\mathbf{R}^T)$ in which $\beta = \arccos(\frac{tr(\mathbf{R})}{2})$. From a skew-symmetric matrix $\hat{\omega}$, we can get the angular speed $\omega \in \mathbb{R}^3$ from which the angular velocity $\dot{\theta}$ can be retrieved as the z-component of ω.

Finally, for any of the tasks mentioned, the Jacobian is used to calculate the actuator-space velocity $\dot{\mathbf{p}}$ as follows:

$$\dot{\mathbf{p}} = \begin{cases} (\mathbf{J})^+\mathbf{v}, & \text{if } \mathbf{J} \text{ is a non-square matrix} \\ \mathbf{J}^{-1}\mathbf{v} & \text{if } \mathbf{J} \text{ is a square matrix} \end{cases}, \tag{19}$$

where $\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$. The generated control signal is the pressure of pouch p_i. The pressure of main chamber p_0 is designed to be independently adjustable to change the robot's stiffness during the performance of a task.

VI. Results and Analysis

A. Simulation Results

The proposed observer-based control is first implemented using a kinematic simulation model of inflatable robot. We used the Robot Operating System (ROS) framework to implement the robot model and algorithm. The algorithm is running at 50 Hz frequency. The parameters of the robot and the controller are $s = 0.42$ m, $l = 0.1$ m, $K_P = K_\omega = 5$. We used the model parameters retrieved from an offline system identification presented in an earlier paper of ours [24] as the true value of model parameters for each segments. Gaussian noise is added to the bending angle data to simulate the real sensory information. Initial estimation covariance matrix $\mathbf{P}_0$ is chosen to be 0.5 $\mathbf{I}$, the initial states χ_0 components are all chosen to be 50.0, process noise variance $\mathbf{Q}_k$ is $\mathbf{I}$, and measurement noise variance $\mathbf{R}_k$ is 50$\mathbf{I}$. To evaluate the performance of the estimation process, we use a Jacobian matrix error, $e_\mathbf{J}$, defined as $e_\mathbf{J} = \sqrt{\frac{\sum_{k=1}^{K}\sum_{m=1}^{M}(\hat{\mathbf{J}}_{km}-\mathbf{J}_{km})^2}{KM}}$ where $\hat{\mathbf{J}}_{km}$ and $\mathbf{J}_{km}$ refer to the component of Jacobian estimate and the true Jacobian in row-k and column-m, respectively. We tested two scenarios with different initial input signals given to the robot before trajectory tracking: the first one uses constant input while the second one uses sinusoidal input.

The results of the first scenario is shown in Fig. 3. At the black vertical line, an arbitrary non-zero constant input pressure is given to the robot. This includes the main chamber pressure (red line) and the pouch pressures (green, blue, and yellow lines respectively) as shown in Fig. 3a. The observer estimates the model parameters in three bending sections as shown in Fig. 3b-Fig. 3d. From the estimated bending parameters, the diagonal components of the actuator-space Jacobian $\hat{\mathbf{J}}_\mathbf{f}$ can be estimated as shown in Fig. 3e (dashed lines) in comparison to the diagonal of real Jacobian $\mathbf{J}_\mathbf{f}$ (solid lines). We can observe that the state estimates (dashed lines) are not converging towards the real model parameters (solid lines) in Fig. 3b-Fig. 3d. This is due to the nonlinearity in the bending model which requires "persistently-exciting" input to make sure the states converge towards the correct values [26]. This requirement is not fulfilled in this scenario which only uses constant input as an initial signal. However, the parameters of the Jacobian estimate, $\hat{b}_i$, in Fig. 3e (dashed lines) are able to converge towards the parameters of real Jacobian b_i (solid lines). As a result, the Jacobian matrix error shown in Fig. 3f also rapidly converges to zero. This means that the correct states estimate are not necessarily needed to achieve a correct inverse-Jacobian-based control command as long as the sensor data is always available to continuously update the Jacobian matrix estimate. At the red vertical line (t=11.5 s), the trajectory tracking starts. From this time, the pressure commands sent to the pouches (green, blue, and yellow lines in Fig. 3a) are generated by the controller. The desired trajectory is a straight horizontal line which consists of a sinusoidal function in x-axis and a straight line in y-axis while the desired orientation is set to 0. We can observe how the robot's tip position (dashed lines) converges towards the desired trajectory (solid lines) as shown in Fig. 3g. The robot's tip orientation also converges to zero, Fig. 3h. At the green vertical line (t=40.5 s), the robot's stiffness is increased by increasing the pressure in the robot's main chamber (red line in Fig. 3a). However, we can observe in Fig. 3e - Fig. 3f that the Jacobian estimate is not affected. Hence, the tracking performance is not affected as reflected by the stable tracking in Fig. 3g - Fig. 3h. Next, at the blue vertical line (t=69 s), the stiffness is increased again,

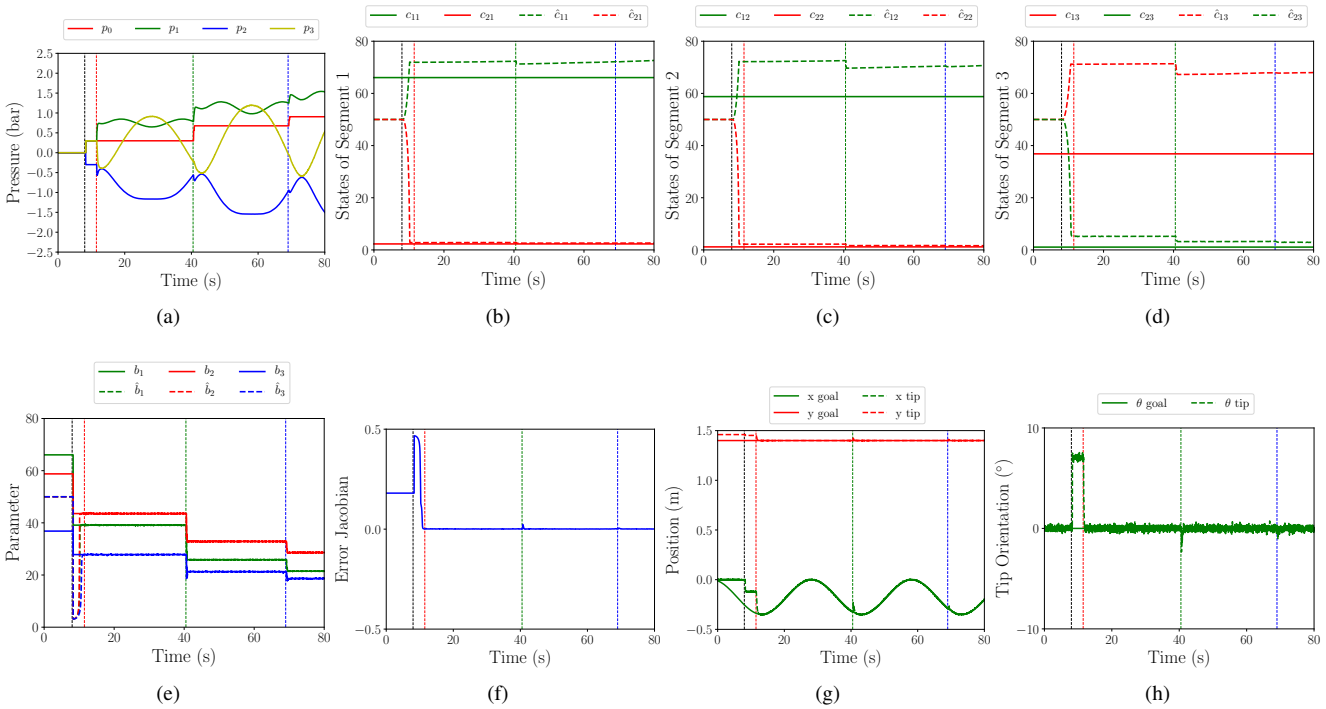


Fig. 3. The trajectory tracking results in the simulation with constant initial input. The plots show (a) the input pressure values given to the main chamber (red line) and each bending pouches (green, blue, and yellow respectively), the model parameter estimate (dashed lines) and the real parameter values (solid lines) for (b) the first, (c) second, and (d) third segment, (e) the estimated (dashed lines) and the real (solid lines) diagonal of Jacobian, (f) the Jacobian error (g) the tip's (dashed lines) and goal (solid lines) position, and (h) the tip's (dashed lines) and goal (solid lines) orientation angle.

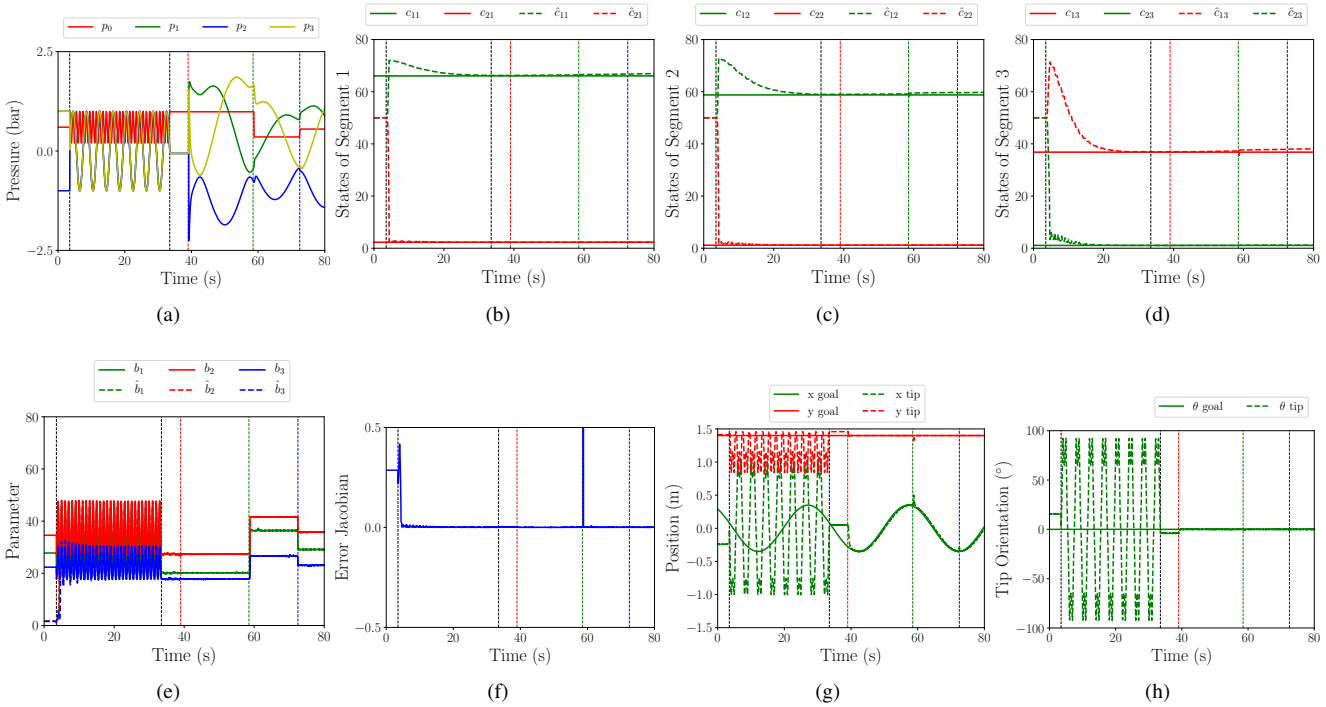
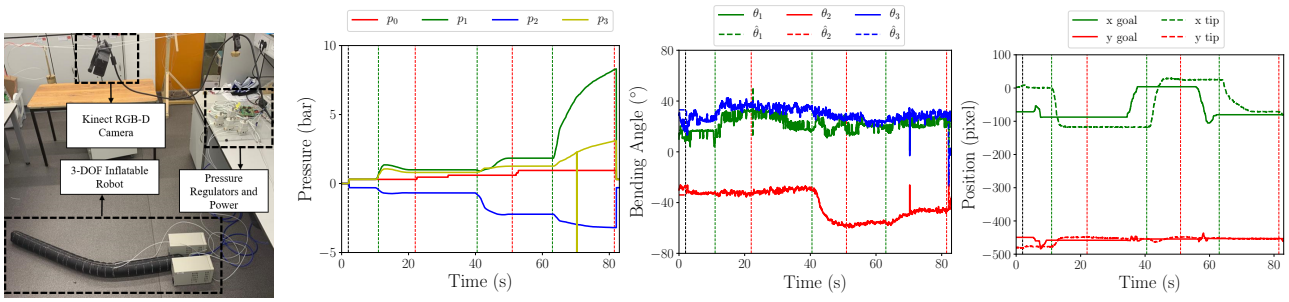


Fig. 4. The trajectory tracking results in the simulation with sinusoidal initial input. The plots show (a) the input pressure values given to the main chamber (red line) and each bending pouches (green, blue, and yellow respectively), the model parameter estimate (dashed lines) and the real parameter values (solid lines) for (b) the first, (c) second, and (d) third segment, (e) the estimated (dashed lines) and the real (solid lines) diagonal of Jacobian, (f) the Jacobian error (g) the tip's (dashed lines) and goal (solid lines) position, and (h) the tip's (dashed lines) and goal (solid lines) orientation angle.

but the control algorithm is still able to ensure trajectory tracking, confirming the ability of the proposed method to



(a) (b) (c) (d)

Fig. 5. (a) The experimental setup showing the inflatable robot and the RGB-D Camera used to detect bending angles. The plots show the trajectory tracking experimental results: (b) the input pressure values given to the main chamber (red line) and each bending pouches (green, blue, and yellow respectively), (c) the estimated (dashed lines) and measured (solid lines) bending angle, and (d) the tip's (dashed lines) and goal (solid lines) position.

perform pose control regardless of the robot's stiffness.

This scenario confirms that a simple input signal (such as a constant input given here) is sufficient to correctly estimate the Jacobian in a very short time to perform trajectory tracking as long as the sensory data is always available. Hence, no prior system identification (like the one presented in [24]) is needed. The drawback of this approach is that the algorithm will not be able to estimate the Jacobian correctly if sensor data is no longer available since the state estimates (3b-Fig. 3d) have not yet converged to the correct values.

The results of the second scenario is shown in Fig. 4. At the first black vertical line, sinusoidal pressure signals with different frequencies were sent to the robot's main chamber (red line) and the pouches (green, blue, and yellow lines respectively) as shown in Fig. 4a. This input is continuously sent to the robot until a sufficient time has passed (at the second vertical black line) so that the observer successfully estimates all the states as shown in Fig. 4b-Fig. 4d. At the red vertical line (t=40 s), we switch to the trajectory tracking stage. At the green vertical line (t=58 s), the robot's stiffness is decreased, while at the blue vertical line (t=72 s), the stiffness is increased. The rest of the results (Jacobian's parameter estimation in Fig. 4e, Jacobian matrix error in Fig. 4f, robot's position tracking in Fig. 4g, and the robot's orientation tracking in Fig. 4h) shows similar performance compared to the first scenario in Fig. 3. This confirms that, in the presence of exciting input signals, the observer is able to correctly estimate all the states. This condition enables the robot to continue the trajectory tracking task even when sensory data is no longer available. However, it requires more time (about 20 seconds) and more complex input signal compared to the first scenario to achieve this condition.

B. Experimental Results

The proposed controller is implemented in a 3-sections inflatable manipulator. An RGB-D Camera, mounted on the top of the robot, is used to detect markers on the robot body. These markers are located in every connecting links, as well as the base and the tip of the manipulator. From the location of these markers, we get the bending angle of every sections and the tip position $\mathbf{x}$. As a preliminary investigation, only position control is employed in this experiment. To control the air pressure, SMC ITV2050-212L pressure regulators are employed. The experimental setup is shown in Fig. 5a.

The results of the experimental scenario are shown in Fig. 5b - Fig. 5d. At the black vertical line, an arbitrary non-zero constant input pressure is given to the robot as shown in Fig. 5b. Since we do not have knowledge regarding the robot's real Jacobian, we compare the measured and the estimated bending angle in Fig. 5c. We can observe that the output estimates (dashed lines) quickly converge towards the real bending angle (solid lines). This implies the correctness of the Jacobian estimation since bending angle θ_i is related directly to the diagonal of Jacobian b_i in (15). At the first green vertical line (t=11 s), the position control starts for the low stiffness state (reflected by low p_0 in Fig. 5b). The goal in this case is a fixed point in the camera space. We can observe how the robot's tip position (dashed lines) converge towards the desired position (solid lines) as shown in Fig. 5d (in between the first green and red vertical lines). There is an observed steady-state error in the x-axis of the tip's final position. This happens in the low stiffness state because the robot is not stiff enough to work against the frictional effect when the tip is already close to the goal. At the first vertical red line (t=22 s), the control is stopped, the goal position (solid lines in Fig. 5d) is moved, and the robot's main chamber pressure (red in Fig. 5b) is increased. At the second vertical green line (t=40.5 s), the control is reactivated, causing the robot's tip to go towards the new target with a stiffer robot body. We can see that the steady-state error of position in x-direction in this case is less than the previous case since the robot is stiffer. Similarly, the control is stopped again at the second vertical red line (t=51 s) to allow modification of goal and stiffness and reactivated at the second vertical green line (t=63 s). In this case, the robot gets even stiffer and the steady-state error becomes minimal, mimicking the accuracy of a rigid-link robot. The movement of the robot for 3 different stiffness states can be observed in Fig. 1.

From this experiment, we can observe how the position control works properly in guiding the manipulator tip towards the target position. Moreover, position control has been suc-

cessfully achieved despite different robot inflation pressure values, i.e., different stiffness values. In the low pressure condition, the robot is safer and more flexible, however more steady-state error is observed. This is useful for applications where safety, rather than control accuracy, is prioritised, such as in a human-robot collaboration. In the high pressure condition, the steady state error becomes minimal, which is useful for other applications, such as pick-and-place tasks. Hence, the proposed algorithm is able to exploit the capability of the inflatable robot to perform position control towards the target for various stiffness conditions depending on the nature of the tasks which need to be carried out.

VII. CONCLUSIONS

In this paper, we propose a model-based kinematic control for a soft, inflatable manipulator. The bending model is based on the Euler-Bernoulli beam theory which considers the effect of the inflation pressure in the robot's main chamber onto the change of its structural stiffness. The unknown bending model parameters are estimated online using the Extended Kalman Filter, exploiting the bending angle measurement from an external sensor. The estimate of these parameters are used to construct the Jacobian matrix for pose control to achieve a desired spatial location for the manipulator tip. Simulation and preliminary experimental validation using a 3-DOF inflatable manipulator demonstrate that the proposed method is able to achieve position control in a planar environment. The simulation results showed that both position and orientation control is achievable. This tracking performance is attained regardless of the the robot's main chamber stiffness, which can be varied by changing the pressure in the main chamber. Future research will explore the consideration of the dynamics of the inflatable robot, friction, and controlled interaction with the environment.

REFERENCES

[1] D. Rus and M. T. & Tolley, "Design, fabrication and control of soft robots." *Nature*, vol. 521, pp. 467–475, 2015. [Online]. Available: http://www.nature.com/nature/journal/v521/n7553/abs/nature14543.html

[2] B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus, "Robust proprioceptive grasping with a soft robot hand," *Autonomous Robots*, vol. 43, no. 3, pp. 681–696, Mar. 2019. [Online]. Available: https://doi.org/10.1007/s10514-018-9754-1

[3] A. Shiva, A. Stilli, Y. Noh, A. Faragasso, I. D. Falco, G. Gerboni, M. Cianchetti, A. Menciassi, K. Althoefer, and H. A. Wurdemann, "Tendon-Based Stiffening for a Pneumatically Actuated Soft Manipulator," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 632–637, Jul. 2016.

[4] A. Stilli, H. A. Wurdemann, and K. Althoefer, "A Novel Concept for Safe, Stiffness-Controllable Robot Links," *Soft Robotics*, vol. 4, no. 1, pp. 16–22, 2017.

[5] M. Manti, V. Cacucciolo, and M. Cianchetti, "Stiffening in Soft Robotics: A Review of the State of the Art," *IEEE Robotics Automation Magazine*, vol. 23, no. 3, pp. 93–106, Sep. 2016.

[6] F. Maghooa, A. Stilli, Y. Noh, K. Althoefer, and H. Wurdemann, "Tendon and pressure actuation for a bio-inspired manipulator based on an antagonistic principle," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2015, pp. 2556–2561.

[7] A. Jiang, T. Ranzani, G. Gerboni, L. Lekstutyte, K. Althoefer, P. Dasgupta, and T. Nanayakkara, "Robotic Granular Jamming: Does the Membrane Matter?" *Soft Robotics*, vol. 1, no. 3, pp. 192–201, 2014.

[8] A. Stilli, E. Kolokotronis, J. Fraś, A. Ataka, K. Althoefer, and H. A. Wurdemann, "Static Kinematics for an Antagonistically Actuated Robot Based on a Beam-Mechanics-Based Model," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2018, pp. 6959–6964.

[9] E. W. Hawkes, L. H. Blumenschein, J. D. Greer, and A. M. Okamura, "A soft robot that navigates its environment through growth," *Science Robotics*, vol. 2, no. 8, 2017. [Online]. Available: http://robotics.sciencemag.org/content/2/8/eaan3028

[10] F. Putzu, T. Abrar, and K. Althoefer, "Plant-Inspired Soft Pneumatic Eversion Robot," in *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, Aug. 2018, pp. 1327–1332.

[11] L. H. Blumenschein, L. T. Gan, J. A. Fan, A. M. Okamura, and E. W. Hawkes, "A Tip-Extending Soft Robot Enables Reconfigurable and Deployable Antennas," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 949–956, Apr. 2018.

[12] M. Coad, L. Blumenschein, S. Cutler, J. R. Zepeda, N. Naclerio, H. ElHussieny, U. Mehmood, J. Ryu, E. W. Hawkes, and A. Okamura, "Vine Robots: Design, Teleoperation, and Deployment for Navigation and Exploration," *IEEE Robotics Automation Magazine*, 2019. [Online]. Available: http://dx.doi.org/10.1109/MRA.2019.2947538.

[13] M. C. Yip and D. B. Camarillo, "Model-Less Feedback Control of Continuum Manipulators in Constrained Environments," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 880–889, Aug. 2014.

[14] M. Li, R. Kang, D. T. Branson, and J. S. Dai, "Model-Free Control for Continuum Robots Based on an Adaptive Kalman Filter," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 286–297, Feb. 2018.

[15] K. Wu, G. Zhu, L. Wu, W. Gao, S. Song, C. M. Lim, and H. Ren, "Safety-Enhanced Model-Free Visual Servoing for Continuum Tubular Robots Through Singularity Avoidance in Confined Environments," *IEEE Access*, vol. 7, pp. 21 539–21 558, 2019.

[16] A. Ataka, P. Qi, A. Shiva, A. Shafti, H. Wurdemann, P. Dasgupta, and K. Althoefer, "Towards safer obstacle avoidance for continuum-style manipulator in dynamic environments," in *2016 6th IEEE Int. Conf. on Biomed. Robot. and Biomechatronics (BioRob)*, Jun. 2016, pp. 600–605.

[17] A. Ataka, P. Qi, A. Shiva, A. Shafti, H. Wurdemann, H. Liu, and K. Althoefer, "Real-time pose estimation and obstacle avoidance for multi-segment continuum manipulator in dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Oct. 2016, pp. 2827–2832.

[18] M. Thieffry, A. Kruszewski, C. Duriez, and T. Guerra, "Control Design for Soft Robots Based on Reduced-Order Model," *IEEE Robotics and Automation Letters*, vol. 4, no. 1, pp. 25–32, Jan. 2019.

[19] R. K. Katzschmann, M. Thieffry, O. Goury, A. Kruszewski, T. 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)*, Apr. 2019, pp. 717–724.

[20] J. D. Greer, L. H. Blumenschein, A. M. Okamura, and E. W. Hawkes, "Obstacle-Aided Navigation of a Soft Growing Robot," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1–8.

[21] J. D. Greer, T. K. Morimoto, A. M. Okamura, and E. W. Hawkes, "A soft, steerable continuum robot that grows via tip extension," *Soft robotics*, vol. 6, no. 1, pp. 95–108, 2019.

[22] Y. Ansari, M. Manti, E. Falotico, M. Cianchetti, and C. Laschi, "Multiobjective Optimization for Stiffness and Position Control in a Soft Robot Arm Module," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 108–115, Jan. 2018.

[23] A. Ataka, A. Stilli, J. Konstantinova, H. A. Wurdemann, and K. Althoefer, "Kinematic Control and Obstacle Avoidance for Soft Inflatable Manipulator," in *Towards Autonomous Robotic Systems*, K. Althoefer, J. Konstantinova, and K. Zhang, Eds. Cham: Springer International Publishing, 2019, pp. 52–64.

[24] A. Ataka, T. Abrar, F. Putzu, H. Godaba, and K. Althoefer, "Model-Based Pose Control of Inflatable Eversion Robot With Variable Stiffness," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3398–3405, 2020.

[25] C. Wielgosz and others, "Bending and buckling of inflatable beams: some new theoretical results," *Thin-walled structures*, vol. 43, no. 8, pp. 1166–1187, 2005.

[26] M. Evestedt, "Parameter and State Estimation with Information-rich Signals," PhD Thesis, Acta Universitatis Upsaliensis, 2007.