A VERSATILE DATA-DRIVEN FRAMEWORK FOR MODEL-INDEPENDENT CONTROL OF CONTINUUM MANIPULATORS INTERACTING WITH OBSTRUCTED ENVIRONMENTS WITH UNKNOWN GEOMETRY AND STIFFNESS

A PREPRINT

Farshid Alambeigi
Department of Mechanical Engineering
University of Texas at Austin
Austin, TX 78712
farshid.alambeigi@austin.utexas.edu

Zerui Wang
Cornerstone Robotics Limited
Shatin, Hong Kong SAR
zerui.j.wang@gmail.com

Yun-Hui Liu
Department of Mechanical and Automation Engineering
The Chinese University of Hong Kong
Shatin, Hong Kong SAR
yhliu@mae.cuhk.edu.hk

Russell H. Taylor
Laboratory for Computational Sensing and Robotics
Johns Hopkins University
Baltimore, MD 21218.
rht@jhu.edu

Mehran Armand
Applied physics Laboratory
Johns Hopkins university
Laurel, MD 20723.
mehran.armand@jhuapl.edu

May 6, 2020

ABSTRACT

This paper addresses the problem of controlling a continuum manipulator (CM) in free or obstructed environments with no prior knowledge about the deformation behavior of the CM and the stiffness and geometry of the interacting obstructed environment. We propose a versatile data-driven priori-model-independent (PMI) control framework, in which various control paradigms (e.g. CM’s position or shape control) can be defined based on the provided feedback. This optimal iterative algorithm learns the deformation behavior of the CM in interaction with an unknown environment, in real time, and then accomplishes the defined control objective. To evaluate the scalability of the proposed framework, we integrated two different CMs, designed for medical applications, with the da Vinci Research Kit (dVRK). The performance and learning capability of the framework was investigated in 11 sets of experiments including PMI position and shape control in free and unknown obstructed environments as well as during manipulation of an unknown deformable object. We also evaluated the performance of our algorithm in an ex-vivo experiment with a lamb heart. The theoretical and experimental results demonstrate the adaptivity, versatility, and accuracy of the proposed framework and, therefore, its suitability for a variety of applications involving continuum manipulators.

Keywords Model-independent control · continuum manipulator · obstructed environment · shape control

†Farshid Alambeigi and Zerui Wang are co-first authors in this work.
1 Introduction

Continuum Manipulators (CM) can improve robot manipulation due to their added dexterity and safety as compared to conventional rigid link robots. These features make CMs a viable candidate for a variety of applications especially in medical interventions where a CM, due to its compliance, may interact safely with a deformable or rigid tissue with unknown deformation behavior. The conformable structure of CMs, however, introduce additional challenges for accurate modeling and control of these robots. These challenges will become more significant when the CM is working in a congested and obstructed environment. In such environments, kinematics and dynamics of the CM differ from its free motion due to its undetermined interactions with obstacles. Adaptive and intelligent control approaches are, therefore, necessary to maneuver the CM into a congested environment with unmodeled obstacles.

1.1 Prior Work

To overcome the difficulties associated with the control of the end-effector position and shape of the CM, researchers have implemented two types of approaches: priori-model-dependent (PMD) and priori-model-independent (PMI) control strategies. In the PMD approaches, a control strategy is implemented based on an obtained fixed and off-line kinematics- or dynamics-based model of the CM (e.g., [7][8][9][10][11]). However, robustness and performance of these analytical approaches are highly dependent on the accuracy of the developed model, which in turn, is a function of simplifying assumptions made during the modeling step. Literature documents various PMD modeling approaches and the parameters affecting the CM behavior including: deformation behavior (i.e., constant curvature or non-constant curvature behavior); mechanical structure (e.g., precurved concentric tubes and concentric tubes with lateral notches); actuation mechanism (e.g., tendons, pneumatic actuators, and shape memory alloys); physical phenomenons such as hysteresis-caused by the elastic properties of the CM’s constitutive material; friction (e.g., caused by actuation tendons); slack in the actuation mechanism; etc. Of note, neglecting any of these parameters may reduce the performance of a PMD control algorithm.

Although most of the mentioned PMD control studies have been focused on free bending motion of the CM, some researchers have taken into account the effect of external disturbances in a constrained environment. For instance, [23] studied a kinematics-based model identifying the contact point between a CM and a fixed obstacle in an unknown environment. [9] used a static CM deflection model and implemented an Extended Kalman filter to estimate the contact force.

PMI control approaches, however, do not rely on a fixed CM model and use adaptive methods to compensate undesired effects, caused by inaccurate modeling or environment disturbances, and estimate the CM behavior in real-time. [4] documented one of the first works in the priori-model-independent control of continuum manipulators interacting with an obstructed environment. They used an optimal control strategy, called "model-less" control method, on a tendon-driven CM and guided their robot along a planar predefined trajectory in a constrained environment. [4] have mentioned some limitations for their approach. First, they used backward differencing of measurements to estimate Jacobian of their robot, resulting in a lag relative to the true Jacobian of the CM. This means that their algorithm uses the estimated Jacobian of the previous step for the current step, which may differ from the true Jacobian. Further, their method was sensitive to the measurement signal noise and using filters added to the lag of the estimation. Second, to estimate the Jacobian, their model-less method solves an under-constrained optimization problem minimizing the change in the Frobenius norm of the estimated Jacobian. However, there is more than one Jacobian solution that satisfies the defined constraints, which may not return the optimal solution and cause some misalignment in estimation of the Jacobian in each iteration. Third, using the proposed controller, the CM may get stuck in a singularity. This happens when the CM interacts with an obstructed environment via its end-effector (tip), where the estimation of the Jacobian with the proposed method becomes ill-conditioned. To solve the third issue, [5] implemented a model-less hybrid position/force control and navigated their CM under body and tip constraints. However, the aforementioned first and second problems were not addressed in this work. They evaluated the performance of their approach in a constraint-free environment.

Machine learning-based PMI controllers are another alternative solution for controlling the CMs. The use of neural networks for compensating unknown kinematics and dynamics of a CM, without presence of external disturbances, was studied by many researchers (e.g., [24][25][26]). As another example, [22] implemented an adaptive support vector regressor method to address position control of a CM in a free environment. In a recent work, [6] trained their neural network to move their continuum soft manipulator in an unstructured environment with external disturbances. A machine learning-based approach was also implemented to develop a dynamic model and a trajectory optimization method for predictive control of a particular soft robotic manipulator [27]. In these studies, however, the networks were trained for a particular type of CM and environment and cannot be generalized for different types of CMs and
environments. Also, similar to other machine learning approaches, the performance of this PMI off-line approach is highly dependent on the training data set obtained by cumbersome identification experiments, which may result in a poor local minimum solution. A reinforcement learning technique was implemented in [28] to address the problem of reaching a point by a simulated multi-segmented dynamical planar model of an octopus arm. In this study, a Hidden Markov Model was implemented and solved using a nonparametric Gaussian temporal difference learning algorithm. [29] demonstrated that similar problem can be solved in the context of continuous action spaces using an actor-critic reinforcement learning approach. However, both of these studies performed in simulations and suffer from demanding real-time computational costs for generating solutions in practice [30].

Visual servoing is another approach, which can be utilized for PMD and PMI control of continuum manipulators. In a PMD approach (i.e. calibrated visual servoing), similar to other model-dependent approaches, a kinematics or dynamics model of the CM together with intrinsic and extrinsic camera parameters are required. Recently, [31] utilized a PMD visual servoing approach to control a concentric tube robot in a free environment. Similar to other PMD approaches, those authors focused on control of the CM in a free environment and did not study the effect of unknown external disturbances on the performance of their algorithm. On the other hand, uncalibrated visual servoing is a PMI control approach, which does not rely on robot kinematics and camera parameters as a priori information. Uncalibrated visual servoing was utilized for controlling conventional rigid-link robotic manipulators in a free environment without the presence of disturbances (e.g. [32]).

1.2 Our Contribution

We propose a versatile framework for Data-Driven PMI (DD-PMI) control of continuum manipulators. The algorithm extends the potential and application of uncalibrated visual servoing approach [32] to the control of CMs. The presented method not only can be used with various sensing modalities (e.g. camera and embedded sensors) but it also can be utilized to simultaneously learn the deformation behavior of the CM and the unknown environment. The performance and scalability of the DD-PMI method are investigated using two different types of CMs with 3 degrees of freedom (DoF): a 3mm continuum manipulator especially designed and fabricated from nitinol tube for head and neck surgeries [33], and a 5mm Endowrist Instrument (Intuitive surgical, Inc., California, USA). We integrated the two CMs with the da Vinci Research Kit (dVRK) [34] and performed several experiments in environments including rigid and deformable obstacles with different geometries. Specific contributions of this study considering the mentioned setup include:

1. Design of a versatile DD-PMI framework addressing the limitations of prior PMI control studies.
2. Theoretical proof for the optimality of the estimated deformation behavior using the proposed learning method.
3. Implementing both DD-PMI CM shape and end-effector position control.
4. Experimental evaluation of the DD-PMI framework in the case of both CM body and end-effector interactions with 6 different unknown obstructed environments without using a force sensor.
5. Experimental demonstration of the DD-PMI algorithm robustness in the presence of the CM hysteresis, backlash, and unknown dynamic disturbances during manipulation of a deformable object.

The remainder of this paper is organized as follows. In Section 2, we present the mathematical models needed for model independent manipulation of CMs and discuss the deformation control framework. Experimental setup and evaluation procedures are presented in Section 3. In Section 4 results of three different sets of experiments are reported including DD-PMI control of both CMs in free and obstructed environments as well as the DD-PMI control during manipulation of an unknown deformable object. We also evaluated the DD-PMI algorithm in an ex-vivo experiment on a lamb heart and investigated the effect of parameters initialization on its convergence rate. In Section 5 we discuss the results and Section 6 concludes the paper.

2 Method

The goal of the DD-PMI algorithm is to autonomously manipulate an unmodeled continuum manipulator to satisfy a control objective in an unknown environment using an uncalibrated sensor. To achieve this objective, unknown deformation behavior of the CM interacting with an environment needs to be estimated and perceived in real time from the provided feedback by an uncalibrated sensor considering the following assumptions:

Assumption 1. We use the visual feedbacks computed from the dVRK endoscope video to implement the DD-PMI method in the image plane. We assume that this visual feedback is always available from a camera, and we can measure the CM’s deformation visually in real-time. The camera parameters (both extrinsic and intrinsic), however, are not known and/or calibrated with respect to the robot(s). Also, this sensing can be noisy.
Figure 1: Conceptual illustration of the DD-PMI control of continuum manipulators in an obstructed environment using the visual feedback or an embedded sensing unit (e.g. FBG or magnetic sensors). In this method, \( M \) sensing points on the surface of the CM are used to either control their positions or control the robot’s shape, without having prior knowledge about the deformation behavior of the CM in interaction with the obstructed environment. In this figure, \( r_i \in \mathbb{R}^3 \) represents the position of the \( i \)th feature point in the Cartesian frame \( \{W\} \) or the embedded sensing unit frame \( \{EM\} \), and \( v_i \in \mathbb{R}^2 \) represents the position of the corresponding sensing point in the image plane of the visual sensing unit. Also, \( O_c \) denotes the camera optical center.

**Remark 1.** For the proposed approach, other imaging modalities such as ultrasound or optical tracker can substitute the endoscope video used for this study. Additionally, if line-of-sight is an issue, we can obtain the required feedback from embedded sensors such as Fiber Bragg Grating (FBG) optical [35] or magnetic sensors [16].

**Assumption 2.** The proposed algorithm enables defining various control objectives (e.g. the control of the position of any point(s) on the body of the CM and its S-shape control) provided that 1) there exist(s) feedback(s) from all the desired point(s) of interest on the CM; 2) there is at least one possible physical solution for the desired control problem considering the CM actuation mechanism.

**Remark 2.** There is no need to identify the CM’s behavior prior to its manipulation by the proposed algorithm. Moreover, the CM’s structure or actuation system can have unknown friction, hysteresis, slack, or backlash.

**Remark 3.** We used two types of CM developed for medical applications. However, the proposed PMI control method can be implemented to various types of continuum manipulators and soft robots.

### 2.1 Continuum Manipulator Jacobian

Consider \( M \) points on the surface of the CM demonstrating locations we are either interested in controlling their position or using them to control the robot’s shape. As shown in Fig. 1, \( r_i(t) \in \mathbb{R}^3 \) represents the position of the \( i \)th point in the Cartesian space:

\[
r_i(t) = \begin{bmatrix} x_i(t) \\ y_i(t) \\ z_i(t) \end{bmatrix}^T
\]

where \( i \in \{1, \ldots, M\} \).

Now, define vector of actuation inputs \( \theta(t) \in \mathbb{R}^n \) and for the ease of presentation stack the coordinates of all of the points in a long vector \( r(t) \in \mathbb{R}^{3M} \) as the following:

\[
r(t) = [r_1^\top(t) \ r_2^\top(t) \ \ldots \ r_M^\top(t)]^\top
\]

Considering Assumption 2, if the position of the \( i \)th point \( r_i(t) \) can be (locally) expressed as a smooth function of the vector of CM actuation inputs \( \theta(t) \), (i.e. \( r_i(t) = r_i(\theta(t)) : \mathbb{R}^n \rightarrow \mathbb{R}^3 \)), we can then locally model the interactive CM...
deformation behavior between displacements of the robot(s) actuation system and displacements of the points on the CM body as the following:

\[
d\frac{dt}{dt}r(t) = \frac{\partial r}{\partial \theta}(\theta(t)) \frac{d\theta(t)}{d\theta}J_{cm}(\theta(t))
\]  

(1)

where \(J_{cm}(r(t)) \in \mathbb{R}^{3M \times n}\) is the CM Jacobian matrix mapping variations of the actuation inputs to the related task space displacements of the points on the CM.

Of note, considering (1), for an infinitesimal time period \(\Delta t\), the changes in the position(s) of the points \(\Delta r(t)\) can be related to the actuation input changes \(\Delta \theta(t)\) as:

\[
\Delta r(t) \approx J_{cm}(\theta(t)) \Delta \theta(t)
\]  

(2)

2.2 Feedback Feature Vector Function

In this study, we use the visual feedback provided by an uncalibrated camera (as our sensor) to implement the DD-PMI control algorithm in the image plane. However, as mentioned in Assumption 1, other sensors can also be used to provide the required feedback in their local coordinate frames. Considering the provided feedback, we can mathematically define \(N\) desired control objectives (e.g. overlaying CM’s end-effector position to a predefined position or giving a particular curvature to the CM) using the Feedback Feature Vector (FFV) function \(\Gamma(t) \in \mathbb{R}^{N}\).

To this end, considering the visual feedback, we first need to relate the coordinates of the feature points on the CM in the image plane \(v_i(t) \in \mathbb{R}^2\) to their corresponding position in the Cartesian space \(r_i(t) \in \mathbb{R}^3\) using the pin-hole perspective projection model without distortion:

\[
\frac{v_i(t)}{1} = \begin{bmatrix} p_i(t) \\ q_i(t) \\ \text{cam} z_i(t) \end{bmatrix}
\]

\[
\begin{bmatrix} v_i(t) \\ 1 \end{bmatrix} = \frac{1}{J_{pi}(t)} \begin{bmatrix} r_i(t) \\ 1 \end{bmatrix}
\]

where \(p_i(t)\) and \(q_i(t)\) are the coordinates of the feature points in the image frame, \(\text{cam} z_i(t)\) is the third element of the feature point position \(r_i(t)\) expressed in the camera frame and \(P\) is the perspective projection matrix, which contains the information of intrinsic and extrinsic camera parameters. Of note, in this paper, besides unknown kinematics behavior of the CM, we assume the projection Jacobian matrix \(J_{pi}(t) \in \mathbb{R}^{3 \times 4}\) is unknown. Using (4), we can write the FFV as a smooth function of the feature points coordinates in the image plane \(\Gamma(v(t))\), where \(v(t) \in \mathbb{R}^{2M}\) is the stack of all the feature points positions in the image plane:

\[
v(t) = [v_1^T(t) \ v_2^T(t) \ ... \ v_M^T(t)]^T
\]

As two examples of the FFV, in this paper, we consider the following functions for evaluation of the DD-PMI method.

2.2.1 Overlaying FFV Function:

Considering Assumption 2 and stacked feature vector, we use the following FFV to overlay position of one or multiple feature points, on the CM, on to the desired predefined location(s) in the image plane \(v_d(t) \in \mathbb{R}^{2M}\):

\[
\Gamma(t) = v(t)
\]

(6)

2.2.2 Curvature Control FFV Function:

We use this FFV to control the shape (i.e. curvature) of the CM in a free environment without the presence of external disturbances. For this function, we need three feature points \((v_1, v_2, \text{ and } v_3)\) on the CM to define a curve that passes through these points with the curvature \(\kappa \in \mathbb{R}\) obtained by the following relation:

\[
\begin{align*}
\frac{v_2 - v_1}{2} &= \frac{v_3 - v_1}{2} \\
\frac{v_3 - v_2}{2} &= \frac{v_3 - v_1}{2}
\end{align*}
\]
\[
\Gamma(v(t)) = \kappa(t) = \frac{2\|v_1 - v_2\| \times (v_2 - v_3)\|}{\|v_1 - v_2\|\|v_2 - v_3\|\|v_3 - v_1\|}
\] (7)

2.3 The Feedback Feature Jacobian Matrix

Considering the definition of the FFV, the CM Jacobian, and the perspective projection model, we can relate the time derivative of the FFV to the CM actuation inputs as the following:

\[
\frac{d}{dt} \Gamma(v(t)) = \frac{\partial \Gamma}{\partial v}(v(t)) \frac{d}{dt} v(t)
= \frac{\partial \Gamma}{\partial r}(r(t)) \frac{d}{dt} r(t)\]
\[
= J_{if}(r(t)) \frac{\partial \theta}{\partial \theta(t)} \frac{d}{dt} \theta(t)
= J_{if} J_{cm} \frac{d}{dt} \theta(t)
\]

where we call \( J_{if}(r(t)) \in \mathbb{R}^{N \times 3M} \) (\( n \geq N \)) as the feedback feature Jacobian matrix. This matrix relates the change of the feature point(s) positions to the change of the FFV. With a similar analogy used in (2), we can approximate (8) as the following:

\[
\Delta \Gamma(t) \approx J_{if} J_{cm} \Delta \theta(t)
\]

(9)

where \( J_{comb} \in \mathbb{R}^{N \times n} \), called combined Jacobian matrix, is the multiplication of the CM and feedback feature Jacobian matrices.

**Remark 4.** In this paper, both \( J_{if} \) and \( J_{cm} \), and subsequently \( J_{comb} \), are considered as unknown Jacobian matrices, which need to be estimated in real-time. However, as mentioned, depending on the sensor used to provide feedback during DD-PMI control, we may need to only estimate \( J_{cm} \).

2.4 Data-Driven PMI Control Approach

As mentioned, the control objectives can be mathematically defined using the feedback feature vector function (e.g. (6) and (7)). Hence, if the FFV converges to a predefined desired feature vector \( \Gamma_d \) then these objectives have been satisfied:

\[
\tilde{\Gamma}(t) = \Gamma(t) - \Gamma_d
\]

(10)

where \( \tilde{\Gamma}(t) \in \mathbb{R}^N \) defines the error vector of control objectives.

So, considering (9) and (10), we define the following optimization problem- subject to various constraints- to successfully achieve the control objectives:

\[
\arg \min_{\Delta \theta(t)} \left\| J_{comb} \Delta \theta(t) - \tilde{\Gamma}(t) \right\|_2^2
\]
\[
s.t. \quad A(t) \Delta \theta(t) \leq b(t),
\]
\[
\Delta \theta(t) \leq \Delta \theta_{\text{max}},
\]
\[
\Delta \theta(t) \geq \Delta \theta_{\text{min}}
\]

(11)

where matrix \( A(t) \in \mathbb{R}^{h \times n} \) and vector \( b(t) \in \mathbb{R}^h \) define \( h \) linear inequality constraints. Further, the last two constraints impose bounds on the CM actuation inputs. Of note, these constraints can easily be defined in the task space and (or) the sensor space (e.g. image plane) using the appropriate Jacobian matrix.
2.4.1 Real-Time Estimation of the Deformation Jacobian:

As mentioned in Remark (4), in each time instant using the provided feedback, we need to estimate \( \mathbf{J}_{comb} \) before solving the defined optimization problem in (11). In fact, equation (9) is a system of \( N \) linear equations with \( N \times n \) unknowns, where the unknowns are the elements of \( \mathbf{J}_{comb} \). Therefore, this equation is an under-determined equation, which does not uniquely determine all the components of \( \mathbf{J}_{comb} \). To solve this problem, [31] used a constrained optimization method, which tries to minimize the change in the Frobenius norm of the estimated Jacobian. However, there is more than one Jacobian solution that satisfies the defined constraints, which may not return the optimal solution and cause some misalignment in estimation of Jacobian in each iteration. To address this issue, we instead implement the Broyden’s method [33] to iteratively approximate \( \mathbf{J}_{comb} \). This method is a recursive first-rank update rule used for solving under-determined nonlinear system of equations [39]. For instance, this update rule is used for visual servo control of uncalibrated robotic systems [32].

If we assume \( \mathbf{J}_{comb} \) is a matrix approximating \( \mathbf{J}_{comb} \), then, it should satisfy the following equation—called "Secant condition"—to ensure both \( \mathbf{J}_{comb}(t + \Delta t) \) and \( \mathbf{J}_{comb}(t + \Delta t) \) demonstrate similar behavior along the direction \( \mathbf{s}_k \) [39]. In other words, given the same input \( \mathbf{s}_k \), both estimated and real Jacobians return identical output \( \mathbf{y}_k \).

\[
\mathbf{y}_k(t) = \mathbf{J}_{comb}(t + \Delta t)\mathbf{s}_k(t)
\]

where \( \mathbf{s}_k(t) = \mathbf{\theta}(t + \Delta t) - \mathbf{\theta}(t) \) and \( \mathbf{y}_k(t) = \mathbf{\Gamma}(t + \Delta t) - \mathbf{\Gamma}(t) \).

**Theorem 1.** Among all matrices \( \hat{\mathbf{J}}_{comb} \) satisfying the Secant condition (12), the matrix \( \hat{\mathbf{J}}_{comb}(t + \Delta t) \) defined by the following equation results in the smallest possible change in the Frobenius matrix norm \( \| \mathbf{J}_{comb}(t + \Delta t) - \hat{\mathbf{J}}_{comb}(t) \| \):

\[
\hat{\mathbf{J}}_{comb}(t + \Delta t) = \mathbf{J}_{comb}(t) + \beta \frac{\Delta \mathbf{\Gamma}(t) - \hat{\mathbf{J}}_{comb}(t) \Delta \mathbf{\theta}(t)}{\Delta \mathbf{\theta}(t)\mathbf{\Gamma}(t)\Delta \mathbf{\theta}(t)} \mathbf{\Gamma}(t)\Delta \mathbf{\theta}(t)
\]

where \( 0 \leq \beta \leq 1 \) is a constant parameter, controlling the rate of change of \( \hat{\mathbf{J}}_{comb}(t) \). In other words (13) is the solution of the following optimization problem:

\[
\arg \min_{\hat{\mathbf{J}}_{comb}(t + \Delta t)} \| \mathbf{J}_{comb}(t + \Delta t) - \hat{\mathbf{J}}_{comb}(t) \|^2_2
\]

s.t. \( \mathbf{y}_k(t) = \mathbf{J}_{comb}(t + \Delta t)\mathbf{s}_k(t) \)

**Proof.** Let \( \hat{\mathbf{J}}_{comb} \) be any estimated matrix satisfying (12), using the properties of Euclidean norm and (13) for any vector \( \Delta \mathbf{\theta} \), we have:

\[
\| \mathbf{J}_{comb}(t + \Delta t) - \hat{\mathbf{J}}_{comb}(t) \| = \| \Delta \mathbf{\Gamma}(t) - \hat{\mathbf{J}}_{comb}(t) \Delta \mathbf{\theta}(t) \| = \| \hat{\mathbf{J}}_{comb}(t) - \mathbf{J}_{comb}(t) \| \Delta \mathbf{\theta}(t) \| \leq \| \hat{\mathbf{J}}_{comb}(t) - \mathbf{J}_{comb}(t) \| \| \Delta \mathbf{\theta}(t) \| \| \Delta \mathbf{\theta}(t) \| = \| \mathbf{J}_{comb} - \hat{\mathbf{J}}_{comb}(t) \| \| \Delta \mathbf{\theta}(t) \| \| \Delta \mathbf{\theta}(t) \|
\]

In this proof, we used the fact that

\[
\| \Delta \mathbf{\theta}(t) \| \| \Delta \mathbf{\theta}(t) \| = 1
\]

7
since $\Delta \theta(t)(\Delta \theta(t))^T$ is symmetric: $\forall \nu \in \mathbb{R}^n$

$$\left\| \Delta \theta(t)(\Delta \theta(t))^T \right\| = \max_{\|\nu\|=1} \nu^T \Delta \theta(t)(\Delta \theta(t))^T \nu$$

$$= \frac{\|\Delta \theta(t)\|^2}{\|\Delta \theta(t)\|^2} = 1$$

Therefore, considering Theorem 1, using (13) guarantees that the estimated Jacobian in each step approximately has identical behavior with the true Jacobian and does not show lag relative to its true value due to imposing the Secant condition. Further, it has the minimum possible change relative to the last estimated matrix, which results in a uniform behavior for the continuum manipulator and non-jerky motions.

2.4.2 DD-PMI Control Algorithm:

We summarize the DD-PMI control method as the following: first, we define the control objective(s) of the problem using appropriate FFV functions (e.g. (6) and (7)) as well as their desired values $\Gamma_d$. We also need to define the constraints of the optimization problem in (11) and the error threshold for the algorithm $\epsilon$. Further, the estimated combined Jacobian should be initialized with a non-singular matrix $\hat{J}_{comb}(0)$ (e.g. Identity matrix) in (13). Of note, better initialization of this matrix makes its estimation faster.

After defining the problem and initializing the parameters, in each time instant $(t + \Delta t)$, given the estimated deformation Jacobian matrix $\hat{J}_{comb}(t)$ determined by (13) and the CM actuation control input $\Delta \theta(t)$ calculated from (11), the CM is moved. Then, the modified CM Jacobian $\hat{J}_{comb}(t + \Delta t)$ is calculated using the actual measured displacement of the feature point(s) on the CM (i.e. $v_i(t)$) in the image plane, which considering Theorem (1) will result in a reduced error compared to the time instant $t$. This algorithm iterates while a predefined error threshold $\epsilon$ is satisfied. Algorithm 1 summarizes these steps.

Algorithm 1: DD-PMI Control of CMs

Initialization:
1. $\Gamma, \Gamma_d \leftarrow$ choose the Feedback Feature Vector function from (6) or (7) as well as their desired values
2. $A, b, \Delta \theta_{min}, \Delta \theta_{max} \leftarrow$ define constraints
3. $\hat{J}_{comb}(0) \leftarrow$ initialize the combined Jacobian matrix
4. $\epsilon \leftarrow$ define the error threshold
5. $\beta \leftarrow$ define the rate of change of $\hat{J}_{comb}(0)$ in (13)

Control loop:
6. while $\|\hat{\Gamma}\| > \epsilon$ do
7. $\Delta \Gamma \leftarrow$ update the error vector using (10)
8. $\Delta \theta \leftarrow$ (11) calculate the actuation control inputs
9. Command the CM with the calculated $\Delta \theta$
10. $\Delta q \leftarrow$ update the vector of the feature points (5)
11. $\hat{J}_{comb} \leftarrow$ (13) update the combined Jacobian matrix
12. end

3 Experimental Setup

To evaluate the DD-PMI method, we integrated two different types of continuum manipulators with the da Vinci Research Kit (Fig. 2). The following sections briefly describe our experimental setup and procedure as well as the considered evaluation measures.

3.1 The da Vinci Research Kit (dVRK)

The dVRK consists of mechanical parts from the da Vinci Classic Surgical System donated by Intuitive Surgical Inc., and open source electronics and software (cisst/SAW libraries) developed by researchers at Johns Hopkins University

8
Figure 2: The setup used for the DD-PMI control of continuum manipulators using the dVRK system. We actuated a 5mm EndoWrist DeBakey Forcep and a custom-designed 3mm continuum manipulator with the da Vinci Instrument’s actuation box to perform our experiments.

This surgical robot has three Patient Side Manipulators (PSMs), one Endoscopic Camera Manipulator (ECM), and two Master Tool Manipulators. For our evaluation experiments, we integrated the following CMs with one of the PSMs, and used an endoscope mounted on the ECM for obtaining the visual feedback (Fig. 2).

3.2 5 mm EndoWrist DeBakey Forceps

Integration of this da Vinci EndoWrist Instrument (Intuitive surgical, Inc., California, USA) with the dVRK, similar to human wrist articulation, provides 7 DoF at the instrument’s distal end. Of note, the instrument solely has 4 DoF, which is controlled by its actuation box. Despite other multiport EndoWrist instruments, this type of forceps incorporate a “snake style” wrist to provide more dexterity to the surgeon (Fig. 2). The flexible part of the instrument is 14 mm long and can bend in two orthogonal planes. It also can have a rotational motion about the central axis of the instrument’s rigid shaft. The fourth degree of freedom of the instrument is used for grasping objects. In this paper, we only use the 4 DoFs provided by the instrument to control the flexible part of this continuum instrument in free and obstructed environments.

3.3 3 mm Continuum Manipulator

This continuum manipulator with 3.3 mm outer diameter and 1.8 mm tool channel diameter has been designed to assist surgeons in treatment of petrous apex lesions [33]. It is made of nitinol with 40 mm working length and has an endoscope channel with a diameter of 0.7 mm (Fig. 2). To actuate this tendon-driven robot using the dVRK and obtain C- and S-shapes, we modified the da Vinci Instrument’s actuation box [40]. Utilizing this actuation box, the robot can be actuated in one plane and two directions and rotated around its rigid shaft axis. In contrast to the 5 mm instrument, the 3 mm manipulator shows hysteresis in its structure, due to the fabrication process and its delicate structure. It also has slack in its actuation mechanism.

3.4 Experimental Procedure and Evaluation Measures

Before performing each experiment, we first integrated the CM with the dVRK and assigned feature point(s) on the body of the CM. The endoscope position and orientation was arbitrarily chosen such that it provided enough field of view. Hence, we are not calibrating the robots and camera before performing experiments. To track displacement of the feature point(s) using endoscope, we implemented a visual tracking algorithm with the Optical Flow package of the OpenCV library [41], which implements the Lucas-Kanade algorithm [42]. To control the movement of the CM using dVRK, we used cisst/SAW libraries in C++ and MATLAB (MathWorks, Inc.). We also used a MATLAB-ROS bridge [43] to communicate between these environments. To solve (11), we used an active set algorithm [44] - i.e. the lsqlin function in MATLAB - which is used for solving constrained linear least-squares problems. The rate of camera
feedback was 25 Hz and the update rate of the total control loop (including the feedback and other calculations using MATLAB) was 15 Hz for all experiments.

As mentioned in Assumption[1] the visual feedback provided by the vision system can be impeded by measurement noise. To mitigate this issue, we first filtered the measurements using the following first-order low-pass filter and then used them in (13) and (11):

$$\dot{x}_f = -\Lambda(x_f - x)$$

where $x$ and $x_f$ represent the original and filtered signals, respectively. The diagonal matrix $\Lambda = \text{diag}(\lambda_1, \lambda_2, \cdots, \lambda_k)$ allows us to adjust the dissipative properties of the filtered signal.

To analyze the results of each experiment and the performance of the DD-PMI method, we used two measures as follows. To study the convergence for each experiment, we plotted the trajectory of the CM’s end effector in the image plane, and calculated the Euclidean distance between the desired target point and the end-effector position at each time step. We call this as the Euclidean distance error (EDE). We also applied the Yoshikawa’s manipulability measure (YMM) $\Phi$[45] to study the performance of the Broyden update rule in real-time estimation of the combined deformation Jacobian $\hat{J}_{\text{comb}}$:

$$\Phi = \sqrt{\det(\hat{J}_{\text{comb}} \hat{J}_{\text{comb}}^T)}$$

where $\det(\cdot)$ denotes determinant of a matrix.

This quantity measures the difficulty of moving the end-effector in a certain direction for conventional redundant rigid link robots. Although we cannot guarantee the estimated combined Jacobian behaves as the true Jacobian, the sudden variations in this measure can provide information regarding the CM response when interacting with environment.

4 Results

Fig. 3[1] summarizes type of experiments we performed to evaluate the DD-PMI algorithm on the mentioned CMs. As shown, to demonstrate versatility and capability of our method, we implemented our algorithm in position control of the CM in free and obstructed environments (with various geometry and stiffness) as well as manipulation of a deformable object with unknown behavior under two imposed boundary condition. In addition, to investigate the viability of the DD-PMI algorithm on ex-vivo tissue, we performed an experiment on a lamb heart. We also performed experiments to evaluate the parameters affecting the convergence of the DD-PMI algorithm. The following sections describe these experiments in detail.
4.1 End-Effector Position Control In Free Environment

We evaluated the performance of the DD-PMI framework in controlling the end-effector position of each CM in free environment. To this end, we first assigned a feasible desired target point $\Gamma_d \in \mathbb{R}^2$ (marked with a red point in Fig. 4a and Fig. 5a) in the image plane, and then arbitrarily initialized $\hat{J}_{comb}(0) \in \mathbb{R}^{2 \times 3}$ with a matrix of all ones. For both robots, we performed several preliminary experiments to obtain the optimal values of the following parameters: $\beta = 0.7$, $\Lambda_1 = \text{diag}(2, 2, 2)$ for filtering the PSM’s joint angles signals provided by the dVRK system, and $\Lambda_2 = \text{diag}(2, 2)$ for the visual feedbacks of the image feature point provided by the ECM. We also set $\epsilon = 1$ pixel as the error threshold. Of note, we repeated this set of experiments for both robots and with different initial conditions and desired target points.
Figure 5: (a) overlay of the traversed trajectory of the 3mm CM’s end-effector with respect to the robot’s initial configuration during its position control in free environment. The green and red points represent the start and end positions of the CM’s tip in the image plane, respectively. (b) the Euclidean distance error (EDE) between the end-effector position and its desired position. (c) the YMM of the estimated Jacobian along the end-effector trajectory.

4.1.1 5mm EndoWrist DeBakey Forceps:

Fig. 4 demonstrates the traversed trajectory of the end-effector in the image plane (in pixel) as well as the EDE between the end-effector position and its desired location in each time instant. The Jacobian-related measure (i.e. the YMM of the estimated Jacobian along the end-effector trajectory) is also plotted in this figure. As shown in the EDE plot, it takes about 124 steps for the algorithm to identify and learn the Jacobian of the CM considering the initialized Jacobian. After this learning phase, the robot quickly approaches to the desired target location (converging phase). Investigation of the YMM measure during the learning phase also demonstrate the quick changes in the Jacobian to capture the deformation behavior of the instrument. Of note, during the converging phase, the rate of changes in the YMM measure drastically decreases, which shows the accuracy of the estimated Jacobian during the learning phase.
We used the 3mm CM to evaluate the efficacy of our method in the DD-PMI shape control. We assigned three feature points on the body of the robot (shown by green points in Fig. 6a), and used the inverse of the FFV function expressed in (7) to fit a constant curvature curve (shown by a red curve in Fig. 6a) to these points in each time instant. We used 5 desired radius of curvatures- as the desired feature vector $\Gamma_d \in \mathbb{R}^5$ and utilized the same experimental parameters and error threshold (i.e. $\epsilon = 1$ pixel) as the previous set of experiments. We arbitrarily initialized $\hat{J}_{comb}(0) \in \mathbb{R}^{1 \times 3}$ with a matrix of all ones. For the visual feedbacks of the image feature point provided by the dVRK system, and $\Lambda_2 = \text{diag}(2, 2, 2)$ for filtering the PSM’s joint angles signals (0). Further, corresponding EDEs and the YMMs are shown in Fig. 6b and Fig. 6c, respectively. According to these figures, for all experiments, it took about 50 steps for the robot to identify and learn the Jacobian and successfully converge to the desired shape. It is notable that the length of learning phase is dependent on the initialization of $\hat{J}_{comb}$. Since we started all experiments with the same initial Jacobian, we see almost the same learning and converging phases.

### 4.2 Continuum Manipulator Shape Control

We used the 3mm CM to evaluate the efficacy of our method in the DD-PMI shape control. We assigned three feature points on the body of the robot (shown by green points in Fig. 6b), and used the inverse of the FFV function expressed in (7) to fit a constant curvature curve (shown by a red curve in Fig. 6b) to these points in each time instant. We used 5 desired radius of curvatures- as the desired feature vector $\Gamma_d \in \mathbb{R}^5$ and utilized the same experimental parameters and error threshold (i.e. $\epsilon = 1$ pixel) as the previous set of experiments. We arbitrarily initialized $\hat{J}_{comb}(0) \in \mathbb{R}^{1 \times 3}$ with a matrix of all ones. For the visual feedbacks of the image feature point provided by the dVRK system, and $\Lambda_2 = \text{diag}(2, 2, 2)$ for filtering the PSM’s joint angles signals (0). Further, corresponding EDEs and the YMMs are shown in Fig. 6b and Fig. 6c, respectively. According to these figures, for all experiments, it took about 50 steps for the robot to identify and learn the Jacobian and successfully converge to the desired shape. It is notable that the length of learning phase is dependent on the initialization of $\hat{J}_{comb}$. Since we started all experiments with the same initial Jacobian, we see almost the same learning and converging phases.

### 4.3 End-Effector Position Control In Obstructed Environments

The interaction of the CM with an obstructed environment may affect the CM’s kinematics and dynamics as compared to the cases of free motion in the space. Further, deformation behavior of the CM is dependent on the location, magnitude, and type (e.g. point load or distributed load) of the contact force as well as the geometry (e.g. flat, concave or convex) and stiffness (e.g. rigid, deformable, or variable) of the unknown obstructed environment. Given these uncertainties, therefore, the implementation of a model-based control method in unknown obstructed environments may not be feasible. Examples of these situations include interaction of various types of continuum manipulators interacting either with deformable tissues (e.g. in natural orificeTrans-luminal endoscopic surgery [19], and endoscopic submucosal dissection [46] or rigid tissues (e.g. in MIS treatment of femur osteonecrosis [47] and pelvic osteolysis in hip revision surgery [48]) with various unknown geometry and stiffness.

Motivated by the mentioned applications, the goal of the experiments was to evaluate the effectiveness of our method for autonomous maneuver of the CM in the presence of unknown obstacles without prior knowledge of both their material properties and geometries. It is worth noting that in the case of PMI control in an obstructed environment, as described in Section 2, deformation behavior of both the CM and the environment have been implicitly included in our formulation. Hence, the algorithm learns both deformation Jacobian of the CM and the environment during the interaction phases. Accurate learning and estimation of the deformation Jacobian matrix, in fact, affects and assigns the next movement of the CM. Convergence of the CM to its predefined error threshold is highly dependent on the learning phase and the estimated deformation Jacobian matrix. In order to evaluate the versatility and performance of the DD-PMI framework in learning unknown environments, we considered two rigid and three deformable obstacles with flat, convex, and variable-shape geometries (as shown in Fig. 7). We chose the target point (marked as red) ensuring the collision of the CM with the obstacle on its trajectory toward the target point.

In this set of experiments, we used both CMs and arbitrarily initialized $\hat{J}_{comb}(0) \in \mathbb{R}^{2 \times 3}$ with a matrix of all ones. For both robots, we used the following parameters: $\beta = 0.7$, $\Lambda_1 = \text{diag}(2, 2, 2)$ for filtering the PSM’s joint angles signals provided by the dVRK system, and $\Lambda_2 = \text{diag}(2, 2)$ for the visual feedbacks of the image feature point provided by the ECM. We also set $\epsilon = 1$ pixel as the error threshold.
Figure 6: (a) overlay of the traversed trajectory of the 3mm CM’s end-effector with respect to the robot’s initial configuration during its shape control in free environment with the desired radius of curvature of $\Gamma_d = \kappa_d^{-1} = 300$ pixel. The green points represent the location of feature points on the surface of the CM and red point represents the end position of the CM’s tip after achieving the desired curvature. (b) the Euclidean distance error (EDE) between the end-effector position and its desired position. (c) the YMM of the estimated Jacobian along the end-effector trajectory.

4.3.1 5mm instrument interacting with the rigid and flat environment:

Fig. 8a demonstrates the DD-PMI control of the 5mm instrument and its interaction with a flat rigid obstacle (Fig. 7a). As shown in Fig. 8b, in the first 60 steps, due to the inaccurate initialization of the Jacobian matrix $\hat{J}_{comb}(0)$, the Euclidean distance error increased. However, after this learning phase (step 61 to 165), the EDE drastically decreased (from 216 to 4 pixel) during the (convergence) phase. We call steps 166 to 600 as the singularity phase. In this phase, regardless of the control inputs (i.e. pull of the actuation cables), the rate of changes in the displacement of the instrument’s tip is small due to the interaction with rigid obstacle. The vector of the instrument’s tip displacement is almost normal to the obstacle surface (i.e. pushing the flat and rigid surface), which results in a small or negligible displacement of the instrument. This incorrect movement arises from the lack of knowledge about the environment geometry and stiffness which results in a dynamic change in the CM Jacobian during interaction with the environment. Of note, during this phase, the DD-PMI algorithm identifies and learns the new deformation behavior of the CM together
Figure 7: Snapshots of the obstructed environments and type of the CMs used to evaluate the performance of the DD-PMI control framework. (a) a rigid and flat environment with 5mm CM. (b) a rigid and flat environment with 3mm CM. (c) a rigid and convex environment with 5mm CM. (d) a deformable and flat environment with 5mm CM. (e) a deformable and convex environment with 5mm CM. (f) a variable stiffness and geometry environment with 5mm CM. (g) an ex-vivo phantom (lamb heart) with 5mm CM.

with the environment in real time. Investigation of the YMM plot (Fig. 8c) supports this claim, where multiple peaks in this measure demonstrate the drastic changes in the Jacobian matrix in order to identify the environment and move the CM out from the singularity situation. The CM pushes and releases the obstacle multiple times (as can be seen in the provided media), to identify and learn the deformation behavior of environment and move out of the singularity configurations. After step 600, the EDE decreased and the robot moved out from the singularity phase and completed the task (convergence phase).

An important information that can be inferred from the YMM figure (Fig. 8c) is detection of the interaction phase between the CM and environment. Investigation of this figure shows that step 35 is the beginning of the interaction between instrument and the flat obstacle. This interaction with obstacle can be identified by comparing the YMM of this experiment with other obstacle free experiments. As we can see, due to the texture and friction of the obstacle surface, the frequency of the changes in the YMM is higher than the obstacle-free cases. This is due to the fact that friction and interaction forces between the instrument and the environment dynamically changing the deformation behavior of the CM, while in the no interaction cases, these forces do not exist and therefore these rapid changes are not detected. In fact, the surface texture has also caused small variations in the traversed trajectory of the CM as well as its EDE (as shown in Fig. 8b).

4.3.2 3mm CM interacting with the rigid and flat environment:

We repeated the previous experiment with the 3mm CM to study the performance of the DD-PMI method on this robot. Fig. 9 demonstrates the traversed trajectory of this CM with respect to the location of the obstacle in the environment as well as the corresponding EDE and YMM during interaction with this flat and rigid obstacle. As shown in Fig. 9b, the first two phases (i.e. learning and converging) can be inferred between steps 0 to 22 and 23 to 42, respectively. After these phases, a singularity phase occurs (steps 43 to 119). Despite the previous experiment which the singularity happened during the interaction of the instrument with the rigid obstacle, here the source of singularity is not solely the collision. As mentioned, the 3mm CM has slack in its actuation mechanism, which means despite the movement of its actuation cables, the CM may not bend. The disparity between the actuating inputs and displacements can be inferred as another type of singularity. The YMM plot during this phase demonstrates the change in the Jacobian to learn the
Figure 8: (a) overlay of the traversed trajectory of the 5mm CM’s end-effector with respect to the robot’s initial configuration during its position control in the flat and rigid obstructed environment. The green and red points represent the start and end positions of the CM’s tip in the image plane, respectively. (b) the Euclidean distance error (EDE) between the end-effector position and its desired position. (c) the YMM of the estimated Jacobian along the end-effector trajectory.

Jacobian and move toward the target point (Fig. 9b). Steps 120 to 146 corresponds to the converging phase, where the EDE decreases quickly before the CM collides with the obstacle. Similar to the previous case, a short singularity phase occurs due to interaction of the CM with obstacle (steps 147 to 156), where the EDE does not change significantly. At step 157, the CM moves away from the singularity and towards the desired target point.

4.3.3 5mm instrument interacting with the rigid and convex environment:

We selected a convex geometry (Fig. 7c) to study our estimation method in the case of interaction of the CM with 1) a non-flat geometry, and 2) an unknown stiff environment with local deformation.
Figure 9: (a) overlay of the traversed trajectory of the 3mm CM’s end-effector with respect to the robot’s initial configuration during its position control in the flat and rigid obstructed environment. The green and red points represent the start and end positions of the CM’s tip in the image plane, respectively. (b) the Euclidean distance error (EDE) between the end-effector position and its desired position. (c) the YMM of the estimated Jacobian along the end-effector trajectory.

Fig. 10a illustrates the traversed trajectory of the 5mm instrument with respect to the location of the obstacle in the environment as well as the corresponding EDE and YMM during interaction with the convex rigid obstacle. Fig. 10b demonstrates three introduced phases of learning, singularity, and convergence. Steps 0 to 18 corresponds to the learning phase, where the EDE increases. Subsequently, we see convergence towards the target (steps 18 to 30). Similar to the previous experiment, due to the interaction with the convex obstacle, the singularity phase occurs and lasts for 134 steps. During this phase, the robot pushes and releases the obstacle multiple times, as can be seen in the YMM and EDE plots (Fig. 10a-b). The oscillations in the EDE plot are due to the local deformations of the obstacle during push and release sequences. The corresponding changes are also visible through the YMM plot, where quick and oscillating changes in this measure denote the DD-PMI control effort to identify the new deformation behavior during
Figure 10: (a) overlay of the traversed trajectory of the 5mm CM’s end-effector with respect to the robot’s initial configuration during its position control in the convex and rigid obstructed environment. The green and red points represent the start and end positions of the CM’s tip in the image plane, respectively. (b) the Euclidean distance error (EDE) between the end-effector position and its desired position. (c) the YMM of the estimated Jacobian along the end-effector trajectory.

This push-release singularity phase and to move away from this singularity. After this phase, the robot moves toward the desired target point in 33 iterations (converging phase).
4.3.4 5mm instrument interacting with the deformable and flat environment:

For this experiment, we used the platinum-catalyzed silicon (Ecoflex 00-50, Smooth-On, Inc.), as a soft, tear resistant, and stretchy material with uniform physical properties, to make a deformable rectangular obstacle (100mm x 70mm x 10mm). Fig. [11a] demonstrates the traversed trajectory of the 5 mm instrument in interaction with this obstacle toward approaching the desired target point. Further, Fig. [11b] and [11c] demonstrate the corresponding EDE and YMM plots of this experiment, respectively. The EDE plot shows three converging phases (i.e. steps 48 to 118, steps 305 to 346, and steps 427 to 526) separated by two singularity phases (i.e. steps 118 to 305 and steps 346 to 427). From step 346 to the end of experiment, the instrument interacts with the deformable obstacle. During this period, as can be seen in the EDE plot, the CM pushes and releases the obstacle multiple times, which results in a slight decrease and increase in the EDE.
Figure 12: (a) overlay of the traversed trajectory of the 5mm CM’s end-effector with respect to the robot’s initial configuration during its position control in the convex and deformable obstructed environment. The green and red points represent the start and end positions of the CM’s tip in the image plane, respectively. (b) the Euclidean distance error (EDE) between the end-effector position and its desired position. (c) the YMM of the estimated Jacobian along the end-effector trajectory.

plot through all this period. Further, the corresponding quick changes in the estimated Jacobian can be observed in the YMM plot, which is due to the direct contact between the instrument and the obstacle.

4.3.5 5mm instrument interacting with the deformable and convex environment:

Fig. 12a, demonstrates the interaction of the instrument with a baseball representing a deformable and convex environment. In contrast to other cases, we started the experiment when the instrument was pushing the baseball. As shown in the EDE plot (Fig. 12b), between steps 27 to 260, the instrument pushes and releases the ball multiple times to identify the deformation behavior of the CM while interacting with the environment. After this phase, the instrument moves toward the goal by pushing the ball and approaching the desired target. Similar to the previous cases, the YMM
Figure 13: (a) overlay of the traversed trajectory of the 5mm CM’s end-effector with respect to the robot’s initial configuration during its position control in the variable-stiffness and -geometry obstructed environment. The green and red points represent the start and end positions of the CM’s tip in the image plane, respectively. (b) the Euclidean distance error (EDE) between the end-effector position and its desired position. (c) the YMM of the estimated Jacobian along the end-effector trajectory.

measure (i.e. Fig. 12c) shows a high frequency oscillatory response due to the interaction of the CM with the ball and the contact forces.

4.3.6 5mm instrument interacting with a variable stiffness and geometry deformable environment:

To investigate the performance of the DD-PMI algorithm in real-time estimation of the deformation behavior of a variable stiffness and geometry environment, we chose an air cushion bag used for packaging (Fig. 13b). The air cushion bag is very soft such that can be pushed up to a certain depth. However, after reaching the mentioned depth, it becomes stiff only in the direction of push- due to the trapped air in the cushion. In other words, because of this variable stiffness, a very small movement in the joint space of the robot may result in a big unexpected movement in
the Cartesian space. Of note, during the experiment, due to the variable trapped air region inside the cushion, the stiff regions vary constantly and the estimator needs to adjust itself in order to predict the deformation behavior by the Jacobian matrix.

Fig. [13] demonstrates the traversed trajectory of the CM, the EDE, and YMM for this experiment. As shown in the YMM plot, despite the other cases, the amplitude and frequency of the YMM oscillatory responses are much higher due to the variable stiffness behavior of the air cushion (Fig. [13]). According to this plot, at step 55 the air cushion’s stiffness begins to drastically change- due to the trapped air- and this variation in the stiffness continues during the experiment. Accordingly, we see the corresponding changes in the EDE plot (Fig. [13b])- starting at the step 55- which is detectable through sudden increase and decrease in this error. The DD-PMI algorithm robustly estimated the deformation behavior in real-time without a prior knowledge and moved toward the target.

4.4 Effect of Parameters Initialization on the DD-PMI Algorithm Convergence

As we mentioned in the previous sections, convergence of the DD-PMI algorithm depends on the coefficient \( \beta \) defined in [13] as well as the initialization of the combined Jacobian \( \hat{J}_{comb}(0) \). To better clarify the effect of these parameters on the convergence of our algorithm, we performed two experiments. In these experiments, we used the 5 mm instrument and repeated the experiment described in Section [4.1.3] with similar initial configuration (i.e. end-effector position and orientation) and the desired position.

In the first experiment, we fixed \( \beta = 0.7 \) and arbitrarily varied the initial Jacobian matrix and studied the convergence time and performance (Fig. [14]).

Considering this Figure, we used identical \( J_{comb}(0) \) as described in the previous sections and arbitrarily scaled this matrix by a factor of 10, 100, 1000, and 10000. As shown in Fig. [14], two observations can be understood: (i) the algorithm convergence does not depend on the matrix initialization; (ii) the convergence time can dramatically be affected by this initialization. For instance, among the selected initializations, scaling factor of 100 resulted in the fastest convergence time (i.e. \( \sim 160 \) iterations) while the original initialization increased the convergence iterations/times to about three times larger (i.e. \( \sim 440 \) iterations).

In the second experiment, we used the best obtained initialization in the previous experiment (i.e. 100 \( J_{comb}(0) \)) and selected 6 different values for \( \beta = \{0.1, 0.3, 0.5, 0.7, 0.9, 1\} \). Fig. [15] demonstrates the results of this experiment and the effect of this parameter on the convergence time. As can be observed, similar to the other experiment, for all cases the algorithm converges but with different convergence rates. For this particular experiment, \( \beta = 0.5 \) and \( \beta = 0.7 \) had the best and the worst results, respectively.

5 Discussion

We investigated the performance of the DD-PMI framework for controlling different type of continuum manipulators using three different category of experiments- as summarized in Fig. [5]. We showed that the performance of the DD-PMI was robust to the inaccurate initialization of the estimated deformation Jacobian matrix and initial configuration of the CMs. Prior PMI control studies mainly focused on position control of the CMs [4][22][6]. Using the introduced general concept of the feedback feature vector function, however, we demonstrated that various control objectives can easily be implemented depending upon the available feedback (e.g. S-shape control). To show this versatility, we performed both position and C-shape control of the CMs using the DD-PMI framework. Further, we successfully showed the scalability of our method using two different types of CMs designed for medical applications. The framework performance is, therefore, independent of the manipulator type and can potentially be extended to other CMs such as concentric tubes [13] and multi backbone continuum robots [19]. Furthermore, although we used visual feedback for our control framework, the presented formulation can be adjusted for other types of sensors (e.g FBG optical or magnetic sensors).

As shown in the results section, generally two phases of learning and convergence are common in all performed experiments (i.e. free, obstructed, and manipulation experiments). In the learning phase, the EDE error first increases due to the inaccurate initialization of the deformation Jacobian matrix, then it decreases in the convergence phase after learning the Jacobian of the CM or the system of CM/environment. In addition to these phases, a singularity phase is observed in the experiments pertaining robot and environment interactions. In this phase, despite the tension in the actuation cables, no movement in the CM’s end-effector is observed due to the interaction of the robot with the obstructed environment (mostly via the robot’s tip). Here, the change in the EDE measure is very small and usually there is a drastic change in the YMM measure- either frequency or magnitude (e.g. Fig. [11] and Fig. [12]. In fact, this sudden change in the EDE and YMM measures can also indicate the initial contact instance as well as the contact interval between the CM and the obstacles in the obstructed environment. The increase in the EDE measure during
Figure 14: Comparison of the performance of the DD-PMI algorithm in the case of different initialization of the combined Jacobian matrix. (a) the Euclidean distance error (EDE) between the end-effector position and its desired position; (b) the YMM of the estimated Jacobian along the end-effector trajectory. $J_0$ is a matrix with entries of all ones.

The singularity phase demonstrates the end of interaction between the CM and environment and moving away from its surface.

The singularity situation may impede the motion of the robot toward the target goal and may cause the robot to get stuck in this situation [5]. To avoid this, we showed that the DD-PMI algorithm could identify and learn in real-time both the CM’s and environment’s deformation behavior during interaction (using push-release movements) and eventually move out of these situations by finding an alternate path to the desired target point. Of note, this singularity phase depends on the geometry and stiffness of the unknown obstacles in the obstructed environment. For instance, in environments including deformable obstacles (Fig. 11 and Fig. 12) or rigid obstacles with local deformation (Fig. 10), the CM may get stuck in various singularity points due to incorrect data-driven estimation of Jacobian- caused by local deformation of the obstacles during interaction with the CM. It’s worth noting that the estimated deformation Jacobian maps the joint space movements of the robot (e.g. actuation of tendons) to the movement of desired location(s) of the interest on the body of the CM (e.g. CM’s tip). Hence, local deformation of an environment interacting with the robot is interpreted as a potential path toward the desired target goal by the data-driven algorithm, which in reality is not a possible solution. As shown in the previous section, the DD-PMI method could address this singularity situation by accurate estimation of the environment stiffness together with the CM deformation behavior. Of note, this is not guaranteed with the proposed approach by [4]. In Section 2, we discussed and proved that this capability is due to the type of formulation of the deformation Jacobian matrix and features of the Broyden update method (i.e. minimum possible Frobenius norm of the estimated Jacobian with its previous value as well as satisfying the Secant condition). To perform similar experiments on flat obstacles, [5] implemented a hybrid force-position PMI controller to avoid singularity in interaction.
with the obstacles. Here, we showed the performance of our method in moving out of singularity situations on different stiffnesses and geometries without the need for the contact force information (i.e., direction and magnitude). Further, in the case of experiment with the air cushion bag, we demonstrated the performance of the DD-PMI algorithm on a variable-geometry and -stiffness environment without the use of contact force information.

The implemented EDE and YMM measures can also express information about the surface stiffness, where the rising and falling (e.g., Fig. 11 and Fig. 12) in the singularity phase are pertinent to the local deformation during interaction of the CM with the environment. Higher magnitudes of these oscillations can be perceived from a deformable obstructed environment and lower magnitudes may be related to local deformations of a rigid environment. In addition, higher frequency in the change of YMM measure demonstrates the rate of variation of the stiffness in the environment (compare Fig. 13 with other experiments in Section 4.3).

The presented results in Section 4.4 on the effects of Jacobian matrix initialization and coefficient $\beta$, demonstrated the robust performance of the DD-PMI algorithm against random initialization of the mentioned parameters. It also showed that optimizing these values can remarkably affect the convergence time, which needs further investigation in our future works. Of note, if we include prior knowledge regarding the CM and environment deformation behavior, the convergence rate can be enhanced. In all of the experiments performed, we assumed no prior knowledge simulating the worst case scenario.
6 Conclusion

In this paper, we presented a versatile data-driven priori-model-independent control framework to manipulate different type of CMs and soft robots in unknown free and obstructed environments without using the interaction force information. The main component of our approach is the use of Broyden update rule for estimating the unknown combined Jacobian matrix similar to the techniques used in the visual servoing literature (e.g. [32]). The main idea of using this update rule together with a visual feedback in this study, however, is not merely visual servoing but intelligently utilizing the features of the Broyden update rule to address the limitations of the methods proposed in the literature for model-independent control of CMs. Our main contribution is the development of an algorithm that can not only perform visual servoing tasks, but also 1) works with other type of uncalibrated sensors beyond vision (e.g. FBG [49, 50] and magnetic tracker sensors [16]); 2) performs shape and end effector position control in both free and obstructed environments without the use of load cells; 3) avoids singularities due to the end effector interaction with unknown environments (without the use of force feedback) as opposed to [5]; and 4) results in the best possible solution to the optimization problem for the model-less control proposed by [4]. Indeed by using Broyden update rule we have theoretically and experimentally shown all of the above can be simultaneously achieved. Furthermore, to the best of our knowledge, visual servoing literature does not address and deeply investigate the combined problem of uncalibrated visual servoing and model-independent control of a CM. We evaluated the DD-PMI framework performance using three different experimental schemes. We demonstrated that performance of the DD-PMI method is versatile in terms of independence to the CM type and initial configuration as well as initialization of the estimated deformation Jacobian matrix. Further, we demonstrated the efficacy of the DD-PMI approach in manipulation of the CM in 6 various obstructed environments with unknown obstacle geometry and stiffness. In addition, the capability of the DD-PMI approach was successfully studied in manipulation of the CM under unknown dynamic disturbances during manipulation of a deformable object. We also demonstrated a successful implementation of the proposed algorithm on an ex-vivo phantom.

Some limitations of this study are as follows. We limited the shape control of our CMs to only constant curvature C-shape configurations. The success of S-shape control as well as the use of other potential feedback feature vector functions will need additional investigations. The framework was evaluated using visual feedback. Future studies may focus on implementing the DD-PMI algorithm using a CM equipped with FBG for minimally-invasive treatment of pelvic osteolysis [48, 51, 52]. The study of osteolysis will also require addressing combined PMI control of CMs in integration with robotic manipulators [53]. Excessive interaction force between the CM and the tissue during the learning phase of the algorithm may create some safety concerns. One potential remedy to this issue can be the use of FBG sensors for simultaneous shape [54, 55] and force sensing [56] for continuous measurement of interaction forces. Another potential extension of this study can be autonomous and semi-autonomous tissue debridement, tissue suturing, and tumor dissection via indirect manipulation of unknown deformable tissues utilizing CMs [57, 58, 59, 60, 61, 62]. The future studies will also include performing in-vivo animal experiments to mimic more realistic tissue properties and clinical settings.

As shown in Section 4.4, convergence of the DD-PMI algorithm depends on initialization of the involved parameters. To enhance the algorithm’s learning performance, future work must address intelligent selection of the involved parameters before and during the experiments. This may include the use of obtained priori and/or posteriori information of the CM and/or environment. Further, in the case of detection of an interaction with an unknown environment, the algorithm may utilize a memory factor to speed up learning. This feature will especially be useful when repetitive interactions occur within the same environment.

References

[1] Shigeo Hirose and Makoto Mori. Biologically inspired snake-like robots. In IEEE Int. Conf. Rob. Biomimetics, pages 1–7, 2004.
[2] Deepak Trivedi, Christopher D Rahn, William M Kier, and Ian D Walker. Soft robotics: Biological inspiration, state of the art, and future research. Appl. Bionics Biomech., 5(3):99–117, 2008.
[3] David B Camarillo, Thomas M Krummel, and J Kenneth Salisbury. Robotic technology in surgery: past, present, and future. Am. J. Surg., 188(4):2–15, 2004.
[4] Michael C Yip and David B Camarillo. Model-less feedback control of continuum manipulators in constrained environments. IEEE Trans. Rob., 30(4):880–889, 2014.
[5] Michael C Yip and David B Camarillo. Model-less hybrid position/force control: a minimalist approach for continuum manipulators in unknown, constrained environments. IEEE Rob. Autom. Lett., 1(2):844–851, 2016.
[6] Thomas George Thuruthel, Egidio Falotico, Mariangela Manti, Andrea Pratesi, Matteo Cianchetti, and Cecilia Laschi. Learning closed loop kinematic controllers for continuum manipulators in unstructured environments. *Soft Rob.*, 4(3):285–296, 2017.

[7] David B Camarillo, Christopher F Milne, Christopher R Carlson, Michael R Zinn, and J Kenneth Salisbury. Mechanics modeling of tendon-driven continuum manipulators. *IEEE Trans. Rob.*, 24(6):1262–1273, 2008.

[8] Robert J Webster III, Jin Seob Kim, Noah J Cowan, Gregory S Chirikjian, and Allison M Okamura. Nonholonomic modeling of needle steering. *Int. J. Rob. Res.*, 25(5-6):509–525, 2006.

[9] D Caleb Rucker and Robert J Webster III. Statics and dynamics of continuum robots with general tendon routing and external loading. *IEEE Trans. Rob.*, 27(6):1033–1044, 2011.

[10] D Caleb Rucker, Robert J Webster III, Gregory S Chirikjian, and Noah J Cowan. Equilibrium conformations of concentric-tube continuum robots. *Int. J. Rob. Res.*, 29(10):1263–1280, 2010.

[11] Kai Xu and Nabil Simaan. Analytic formulation for kinematics, statics, and shape restoration of multibackbone continuum robots via elliptic integrals. *J. Mech. Rob.*, 2(1):011006, 2010.

[12] Valentin Falkenhahn, Alexander Hildebrandt, Rüdiger Neumann, and Oliver Sawodny. Dynamic control of the bionic handling assistant. *IEEE/ASME Trans. Mechatron.*, 22(1):6–17, 2017.

[13] Robert J Webster III and Bryan A Jones. Design and kinematic modeling of constant curvature continuum robots: A review. *Int. J. Rob. Res.*, 29(13):1661–1683, 2010.

[14] Ryan J Murphy, Michael DM Kutzer, Sean M Segreti, Blake C Lucas, and Mehran Armand. Design and kinematic characterization of a surgical manipulator with a focus on treating osteolysis. *Robotica*, 32(6):835–850, 2014.

[15] Tobias Mahl, Alexander Hildebrandt, and Oliver Sawodny. A variable curvature continuum kinematics for kinematic control of the bionic handling assistant. *IEEE Trans. Rob.*, 30(4):935–949, 2014.

[16] Mohsen Mahvash and Pierre E Dupont. Stiffness control of surgical continuum manipulators. *IEEE Trans. Rob.*, 27(2):334–345, 2011.

[17] Robert J Webster III, Joseph M Romano, and Noah J Cowan. Mechanics of precurved-tube continuum robots. *IEEE Trans. Rob.*, 25(1):67–78, 2009.

[18] Nabil Simaan. Snake-like units using flexible backbones and actuation redundancy for enhanced miniaturization. In *Proc. IEEE Int. Conf. Rob. Autom.*, pages 3012–3017, 2005.

[19] Nabil Simaan, Kai Xu, Wei Wei, Ankur Kapoor, Peter Kazanzides, Russell Taylor, and Paul Flint. Design and integration of a telerobotic system for minimally invasive surgery of the throat. *Int. J. Rob. Res.*, 28(9):1134–1153, 2009.

[20] Anzhu Gao, Ryan J Murphy, Hao Liu, Iulian I Iordachita, and Mehran Armand. Mechanical model of dexterous continuum manipulators with compliant joints and tendon/external force interactions. *IEEE/ASME Trans. Mechatron.*, 22(1):465–475, 2017.

[21] John H Crews and Gregory D Buckner. Design optimization of a shape memory alloy–actuated robotic catheter. *J. Intell. Mater. Syst. Struct.*, 23(5):545–562, 2012.

[22] Achille Melingui, Joseph Jean-Baptiste Mvogo Ahanda, Othman Lakhal, Jean Bosco Mbede, and Rochdi Merzouki. Adaptive algorithms for performance improvement of a class of continuum manipulators. *IEEE Trans. Syst. Man Cybern. Syst.*, 2017.

[23] Andrea Bajo and Nabil Simaan. Kinematics-based detection and localization of contacts along multisegment continuum robots. *IEEE Trans. Rob.*, 28(2):291–302, 2012.

[24] Achille Melingui, Othman Lakhal, Boubaker Daachi, Jean Bosco Mbede, and Rochdi Merzouki. Adaptive neural network control of a compact bionic handling arm. *IEEE/ASME Trans. Mechatron.*, 20(6):2862–2875, 2015.

[25] Michele Giorelli, Federico Renda, Marcello Calisti, Andrea Arienti, Gabriele Ferri, and Cecilia Laschi. Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature. *IEEE Trans. Rob.*, 31(4):823–834, 2015.

[26] David Braganza, Darren M Dawson, Ian D Walker, and Nintendra Nath. A neural network controller for continuum robots. *IEEE Trans. Rob.*, 23(6):1270–1277, 2007.

[27] Thomas George Thuruthel, Egidio Falotico, Federico Renda, and Cecilia Laschi. Learning dynamic models for open loop predictive control of soft robotic manipulators. *Bioinspiration biomimetics*, 12(6):066003, 2017.

[28] Yaakov Engel, Peter Szabo, and Dmitry Volkinshtein. Learning to control an octopus arm with gaussian process temporal difference methods. In *Adv. Neural Inf. Process. Syst.*, pages 347–354, 2006.
[29] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *ICML*, pages 387–395, 2014.

[30] Thomas George Thuruthel, Yasmin Ansari, Egidio Falotico, and Cecilia Laschi. Control strategies for soft robotic manipulators: A survey. *Soft Rob.*, 5(2):149–163, 2018.

[31] Andrey V Kudryavtsev, Mohamed Taha Chikhaoui, Aleksandr Liadov, Patrick Rougeot, Fabien Spindler, Kanty Rabenorosoa, Jessica Burgner-Kahrs, Brahim Tamadazte, and Nicolas Andreff. Eye-in-hand visual servoing of concentric tube robots. *IEEE Rob. Autom. Lett.*, 3(3):2315–2321, 2018.

[32] Jenelle Armstrong Piepmeier, Gary V McMurray, and Harvey Lipkin. Uncalibrated dynamic visual servoing. *IEEE Trans. Rob. Autom.*, 20(1):143–147, 2004.

[33] Suat Coemert, Anzhu Gao, John P Carey, Mattias F Traeger, Russell H Taylor, Tim C Lueth, and Mehran Armand. Development of a snake-like dexterous manipulator for skull base surgery. In *IEEE Annu. Int. Conf. Eng. Med. Biol. Soc.*, pages 5087–5090, 2016.

[34] Peter Kazanzides, Zihan Chen, Anton Deguet, Gregory S Fischer, Russell H Taylor, and Simon P DiMaio. An open-source research kit for the da vinci® surgical system. In *IEEE Int. Conf. Rob. Autom.*, pages 6434–6439, 2014.

[35] Hao Liu, Amirhossein Farvardin, Robert Grupp, Ryan J Murphy, Russell H Taylor, Iulian Iordachita, and Mehran Armand. Shape tracking of a dexterous continuum manipulator utilizing two large deflection shape sensors. *IEEE Sens. J.*, 15(10):5494–5503, 2015.

[36] David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. Prentice Hall Prof. Tech. Ref., 2002.

[37] Clark Kimberling. *Triangle centers and central triangles*. Utilitas Math. Publishing Incorporated, 1998.

[38] Charles G Broyden. A class of methods for solving nonlinear simultaneous equations. *Math. Comput.*, 19(92):577–593, 1965.

[39] Stephen Wright and Jorge Nocedal. Numerical optimization. *Springer Science*, 35:67–68, 1999.

[40] Philip E Gill, Walter Murray, and Margaret H Wright. *Practical optimization*. Academic press, 1981.

[41] Tsuneo Yoshikawa. Manipulability of robotic mechanisms. *Int. J. Rob. Res.*, 4(3):3–9, 1985.

[42] Nisha Patel, Carlo A Seneci, Jianzhong Shang, Konrad Leibrandt, Guang-Zhong Yang, Ara Darzi, and Julian Teare. Evaluation of a novel flexible snake robot for endoluminal surgery. *Surg. Endosc.*, 29(11):3349–3355, 2015.

[43] Farshid Alambeigi, Yu Wang, Shahriar Sefati, Cong Gao, Ryan J Murphy, Iulian Iordachita, Russel H Taylor, Harpal Kahanu, and Mehran Armand. A curved-drilling approach in core decompression of the femoral head osteonecrosis using a continuum manipulator. *IEEE Rob. Autom. Lett.*, 2(3):1480–1487, 2017.

[44] Farshid Alambeigi, Shahriar Sefati, Ryan J Murphy, Iulian Iordachita, and Mehran Armand. Design and characterization of a debriding tool in robot-assisted treatment of osteolysis. In *IEEE Int. Conf. Robot. Autom.*, pages 5664–5669, 2016.

[45] Hao Liu, Amirhossein Farvardin, Sahba Aghajani Pedram, Iulian Iordachita, Russell H Taylor, and Mehran Armand. Large deflection shape sensing of a continuum manipulator for minimally-invasive surgery. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 201–206. IEEE, 2015.

[46] Shahriar Sefati, Rachel Hegeman, Farshid Alambeigi, Iulian Iordachita, and Mehran Armand. Fbg-based position estimation of highly deformable continuum manipulators: Model-dependent vs. data-driven approaches. In *2019 International Symposium on Medical Robotics (ISMR)*, pages 1–6. IEEE, 2019.

[47] Farshid Alambeigi, Shahriar Sefati, and Mehran Armand. A convex optimization framework for constrained concurrent motion control of a hybrid redundant surgical system. In *2018 Annual American Control Conference (ACC)*, pages 1158–1165. IEEE, 2018.
[52] Farshid Alambeigi, Mahsan Bakhtiarinejad, Shahriar Sefati, Rachel Hegeman, Iulian Iordachita, Harpal Khanuja, and Mehran Armand. On the use of a continuum manipulator and a bendable medical screw for minimally invasive interventions in orthopedic surgery. *IEEE Transactions on Medical Robotics and Bionics*, 1(1):14–21, 2019.

[53] Paul Wilkening, Farshid Alambeigi, Ryan J Murphy, Russell H Taylor, and Mehran Armand. Development and experimental evaluation of concurrent control of a robotic arm and continuum manipulator for osteolytic lesion treatment. *IEEE Rob. Autom. Lett.*, 2(3):1625–1631, 2017.

[54] Shahriar Sefati, Michael Pozin, Farshid Alambeigi, Iulian Iordachita, Russell H Taylor, and Mehran Armand. A highly sensitive fiber bragg grating shape sensor for continuum manipulators with large deflections. In *IEEE Sens.*., pages 1–3, 2017.

[55] Shahriar Sefati, Farshid Alambeigi, Iulian Iordachita, Mehran Armand, and Ryan J Murphy. Fbg-based large deflection shape sensing of a continuum manipulator: Manufacturing optimization. In *IEEE Sens.*., pages 1–3, 2016.

[56] Berk Gonenc and Iulian Iordachita. Fbg-based transverse and axial force-sensing micro-forceps for retinal microsurgery. In *IEEE Sens.*., pages 1–3, 2016.

[57] Sahba Aghajani Pedram, Peter Ferguson, Ji Ma, Erik Dutson, and Jacob Rosen. Autonomous suturing via surgical robot: An algorithm for optimal selection of needle diameter, shape, and path. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2391–2398. IEEE, 2017.

[58] Changyeob Shin, Peter Walker Ferguson, Sahba Aghajani Pedram, Ji Ma, Erik P Dutson, and Jacob Rosen. Autonomous tissue manipulation via surgical robot using learning based model predictive control. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3875–3881. IEEE, 2019.

[59] Sahba Aghajani Pedram, Peter Walker Ferguson, Changyeob Shin, Ankur Mehta, Erik P Dutson, Farshid Alambeigi, and Jacob Rosen. Toward synergetic learning for autonomous manipulation of deformable tissues via surgical robots: An approximate q-learning approach. *arXiv preprint arXiv:1910.03398* 2019.

[60] Farshid Alambeigi, Zerui Wang, Rachel Hegeman, Yun-Hui Liu, and Mehran Armand. Autonomous data-driven manipulation of unknown anisotropic deformable tissues using unmodelled continuum manipulators. *IEEE Robotics and Automation Letters*, 4(2):254–261, 2018.

[61] Farshid Alambeigi, Zerui Wang, Rachel Hegeman, Yun-Hui Liu, and Mehran Armand. A robust data-driven approach for online learning and manipulation of unmodeled 3-d heterogeneous compliant objects. *IEEE Robotics and Automation Letters*, 3(4):4140–4147, 2018.

[62] Farshid Alambeigi, Zerui Wang, Yun-hui Liu, Russell H Taylor, and Mehran Armand. Toward semi-autonomous cryoablation of kidney tumors via model-independent deformable tissue manipulation technique. *Annals of biomedical engineering*, 46(10):1650–1662, 2018.