Nonparametric Online Learning Control for Soft Continuum Robot: An Enabling Technique for Effective Endoscopic Navigation

Kit-Hang Lee[1], Denny K.C. Fu[1], Martin C.W. Leong[1], Marco Chow[1], Hing-Choi Fu[1], Kaspar Althoefer[2], K.Y. Sze[1], Chung-Kwong Yeung[3], and Ka-Wai Kwok[1]

[1]K.H. Lee, Denny K.C. Fu, Martin C.W. Leong, Marco Chow, H. C. Fu, K. Y. Sze, and Ka-Wai Kwok are with the Department of Mechanical Engineering, The University of Hong Kong, Hong Kong

[2]K. Althoefer is with School of Engineering and Materials Science, Queen Mary University of London, UK.

[3]C.K. Yeung is with NISI (HK) Limited, and Honorary Clinical Professor of Department of Surgery, Li Ka Shing Faculty of Medicine, The University of Hong Kong, Hong Kong.

Corresponding author: K.W. Kwok, e-mail: kwokkw@hku.hk

Abstract

Bio-inspired robotic structures composed of soft actuation units have attracted increasing research interest. Taking advantage of its inherent compliance, soft robots can assure safe interaction with external environments, provided that precise and effective manipulation could be achieved. Endoscopy is a typical application. However, previous model-based control approaches often require simplified geometric assumptions on the soft manipulator, but which could be very inaccurate in the presence of unmodeled external interaction forces. In this study, we propose a generic control framework based on nonparametric, online, as well as local training, in order to *learn* the inverse model directly, without prior knowledge of the robot's structural parameters. Detailed experimental evaluation was conducted on a soft robot prototype with control redundancy, performing trajectory tracking in dynamically constrained environments. Advanced element formulation of finite element analysis (FEA) is employed to initialize the control policy, hence eliminating the need for random exploration in the robot's workspace. The proposed control framework enabled a soft fluid-driven continuum robot to follow a 3D trajectory precisely, even under dynamic external disturbance. Such enhanced control accuracy and adaptability would facilitate effective endoscopic navigation in complex and changing environments.

Keywords: Endoscopic navigation, finite element analysis (FEA), inverse transition model, soft robot control.

I. Introduction

Design of nature-inspired manipulators actuated based on soft material properties has become one of the most engaged research areas in robotics [1]. Soft robots embedded with delicate chambers can be driven by fluidic input [1-4], resulting in functional deformations such as bending and elongation/shortening [5]. Accredited to the limber robotic structure, its manipulation assures high compliance within a confined region, facilitating versatile interaction with surrounding objects [6, 7]. These features introduce a potential impact to many robotic applications demanding for safe interaction within a dynamic environment, such as soft tissue in minimally invasive surgery (MIS) [8, 9]. Therefore, endoscopy is one of the timely applications. Conventional endoscopes predominately comprise of metallic skeleton driven by steel cables, governing the kinematics of a series of bending mechanisms. It inevitably induces high friction and is susceptible to fatigue failure upon prolonged duration of service. These metallic structures also come with a high rigidity at the scope tip, which may increase the risk of causing trauma or even perforation when the scope is forcefully pushed against the wall of a confined lumen or cavity [10]. This has motivated the development of soft robotic instruments for surgical interventions [11-14] which can also be disposable to ensure zero risk of endoscopy-related infection transmission. Endotics [11, 12] was the first system developed for the purpose of pain-free colonoscopy. Its novel locomotion scheme attempted to prevent the formation of complicated looping at the sigmoid/descending colon. As result, its single-segment bending is capable of omni-directional endoscopic exploration along the colon. Aer-O-Scope [13] was another commercial colonoscope relying on a simple approach making use of single-segment bending which is combined with effective locomotion. The STIFF-FLOP soft robot [9, 14] was another milestone in keyhole surgery to offer intra-cavitary exploration using a soft-material robot, validated in a cadaveric trial for the first time.

Soft robotic endoscopes have brought a few branches of research directions in the limelight. Various control approaches have also been developed to master the dexterity of such manipulators, giving rise to agile and responsive tele-manipulation. Paramount to surgical safety, having a decent control performance in the presence of a confined and dynamic environment is also essential. Therefore, much research effort [15-18] has been paid for deriving analytical models with the aim to describe or predict the robot kinematic/dynamic behavior [19], akin to controlling conventional rigid-link robots. However, these analytical models are complex due to the intrinsic non-linear hyper-elastic property of the soft elastomeric materials, which constitute the robot body. Any additional control dimensionality of the soft robot would further exacerbate the complexity of such kinematic equations [16].

To simplify the modeling process, the piecewise constant curvature (PCC) assumption is one of the widely-used techniques [15, 16, 18, 20] to obtain close-formed solutions [21, 22]. This enables real-time kinematic control of curvature discrepancy to attain the desired pose [23] and to perform dynamic motion primitives [24] for fluidically-driven soft continuum robots. The parameters that govern the analytical models can also be estimated online [25]. Other model-based methods have been proposed without taking the PCC assumption, such as approximation of trunk-like structures to infinite degree-of-freedom (DoF) system [26], and modeling spring-mass modeling techniques [27, 28] which can be incorporated in a hierarchical controller for generating stereotyped motions of an octopus-like manipulator [27]. Recently, the Cosserat theory [29] of elasticity has been used to predict underwater motion of a cable-driven, octopus-like soft robot [30] by deducing its geometrically exact formulations. Yet, external disturbance to the robot, such as gravity, payload and external interaction, can promptly invalidate those assumptions. These over-simplified assumptions would substantially degrade the model's reliability in real applications. Moreover, structural parameters in the kinematics have to be determined prior to the modeling process. The search for these invariant coefficients is heuristic in nature. This might induce further complications when mapping the robot motion analytically. In addition, such invariant can only hold upon slight modification of the robot, as they possess strong correlation with the robot's mechanical structure. Inevitably, the analytical model has to be revisited after any major change to the robot structure, further diminishing the effectiveness of such an approach.

With the foreseen difficulty of developing the analytical/kinematic model, research attempts were made to control the soft, pliable robot using non-parametric, learning-based approaches. The idea is to obtain forward/inverse mappings for kinematics/dynamics robot control based on measurement data only. Model-free control methods can also be developed based on direct modeling architecture [31], where the inverse mapping is directly obtained. This mapping depicts the inverse transition model of the robot, which could be a changing function due to the contact between the robot and the environments, such as soft tissue. The use of Neural Networks (NNs) has been proposed to globally approximate the inverse mapping between end-effector and robot actuation [32, 33]. Such an approach can compensate for uncertainties in robot dynamics [32], and has been demonstrated to yield even more reliable solutions when compared to using an analytical model of a cable-driven soft robot [33]. Previous studies of NNs mostly consider simplified scenarios, such as a non-redundant manipulator and contact-free situation [32, 33]. Although redundantly actuated robotic systems can be controlled in lower dimensionality in a hierarchical manner, it may require pre-defined movement patterns (primitives) for specific task goals [27]. Moreover, there has been a great demand on using machine learning approaches to address the change in inverse mapping of the hyper-elastic robot upon contact [1]. A Jacobian-based model-free controller has shown its capabilities to manipulate a planar cable-driven continuum robot in an environment with static constraints [34]. However, there are still no example that demonstrates manipulation of redundantly-actuated soft continuum robot in three-dimensional space, and is adaptive to unknown external disturbance.

In this paper, we propose a control framework based on nonparametric local learning technique. Nonparametric local learning methods, such as [35, 36], possess the ability to learn the high dimensional inverse transition of rigid-link robots. The essence of nonparametric local methods is to construct a batch of locally weighted models that collectively approximate the inverse mapping. Each of these models is spawned and updated in an independent manner, such that the overall architecture can be rapidly transformed to accommodate new input data. Meanwhile, the weighted global approximation can be optimized on the fly, and consistent with the desired control behavior [36]. Such nonparametric local learning approach can thus facilitate fast online correction of the learning model [37]. Therefore, the proposed framework is suitable for providing a rapid response to soft robot manipulation within constrained environments. Workspace exploration is a prerequisite to collect pre-training data for learning the proposed controller. It is desirable to have accurate enough kinematic data to initialize the controller offline, since it is impractical to carry out robot exploration in the confined trans-luminal workspace. We propose to use finite element analysis (FEA) to sample the kinematic data for the offline learning process. FEA has been widely used in the design optimization and miniaturization of soft robots [13]. Not only can the FEA accurately predict the highly deformable behaviors, but it can also provide data for characterization of inverse kinematic relations for control [38]. However, the application of FEA to robotic control has only been minimally investigated in continuum structure with small deformation [38, 39]. The major contributions of this work are:

- The first attempt to exploit online nonparametric local learning technique with the aim to directly approximate the inverse kinematics of a redundantly-actuated, fluid-driven endoscope prototype for soft robot control in 3-D space (**Section II**);
- Novel integration of FEA into the online learning method is implemented to initialize a reliable inverse model offline before deployment of the proposed controller in practical scenarios. (**Section III**);
- Experimental validation of the control performance and adaptability is conducted to demonstrate 3D trajectory tracking (mean error < 2.49º) of soft continuum robot even under dynamic external disturbance (**Section III**).

II. METHODS

A. Design of Soft Endoscope Prototype

A generic, fluidic-driven soft continuum robot made of RTV (Room Temperature Vulcanization) silicone rubber (Ecoflex 0050, Smooth-on Inc.) is designed and fabricated to evaluate the proposed framework for endoscopic navigation (**Fig. 1a**). The soft robot comprises of three cylindrical inflatable chambers, each covered by a helical Kevlar string layer with a pitch of 1mm. This fiber constrained structure is first proposed by Suzumori *et al.* [4, 40], in which the helical constraint layer enforces axial anisotropic expansion of inflatable chambers, so as to generate an effective bending moment when subject to pressure input. To enable effective endoscopic navigation, the three air chambers can be individually actuated by air or other fluid, facilitating a large panoramic workspace with a bending angle >150˚. The slender robot configuration with 13-mm outer diameter and 93-mm length are also compatible with conventional endoscopes, which is of importance to dexterous manipulation inside a confined trans-luminal workspace. Fabrication of the robot involves three major phases: **i)** Three cylindrical air chambers are cast with RTV silicone in inner molds; **ii)** Kevlar strings are wrapped densely in a single helical structure along each soft chamber; **iii)** Additional layers of silicone are cast to house the three inflatable chambers into one. This could fix the strings against dislocation, even after numerous bending actions.

B. Characterization of Robot Motion Transition

Gradual, smooth regulation of the fluidic flow rate allows steady bending of the presented soft manipulator. It also allows rapid reaching of fluid pressure equilibrium, minimizing the residual motion generated during such fluidic actuation. During endoscopic navigation within small and confined spaces (e.g. duodenum), such quasi-static motion characteristic [41] can facilitate effective, precise targeting of the endoscopic camera or interventional tools (e.g., biopsy forceps or brush cytology) at the surgical regions of interest, thereby avoiding inadvertent damage to delicate tissue and potential discomfort to the patient.

To mathematically describe the motion transition of the soft robot, let $\mathbf{u}_k \in U$ be the fluid pressure (at equilibrium) in the actuation chambers at time step k where U denotes the control space. Let $\boldsymbol{\theta}_k$ be the state of the robot when the chambers are filled with the pressure of $\mathbf{u}_k$ at equilibrium. This state corresponds to the distal tip position $\mathbf{p} \in \Re^3$ and orientation normal $\mathbf{n} \in \Re^3$ in the Cartesian space (**Fig. 2**), which are collectively represented by $\mathbf{x}_k = [\mathbf{p}, \mathbf{n}]^T \in \Re^6$. The forward transition model of the soft robot can be described by the following equation system:

$$\begin{cases} \boldsymbol{\theta}_{k+1} = f(\boldsymbol{\theta}_k, \Delta\mathbf{u}_k) \\ \mathbf{x}_k = h(\boldsymbol{\theta}_k) \end{cases} \quad (1)$$

where $\Delta\mathbf{u}_k = \mathbf{u}_{k+1} - \mathbf{u}_k$ is the difference of the fluid pressure. The motion transition function f is a continuous mapping that depends on the current state of the robot $\boldsymbol{\theta}_k$. Compared to rigid-link robots where the robot state can be well-defined by joint kinematics, it is difficult to describe the exact state of the soft robot. For example, model-based approaches approximate this robot state based on PCC [15, 16, 18, 20-25] and non-PCC [26-30] constraints. The nonlinear function h transforms robot state $\boldsymbol{\theta}_k$ to Cartesian representation $\mathbf{x}_k$.

Typical endoscopic navigation requires delicate articulation of the distal tip so as to provide accurate positioning and easy access to the soft tissue lesion. A micro-camera at the soft robot tip provides forward vision. Therefore, the operator can aim the distal tip at a lesion target on the luminal wall so as to guide the interventional instruments to deploy from the tip via the biopsy channel. This tele-manipulated endoscopic navigation gives rise to a robot task space coordinate $\mathbf{s}_k$ defined by its viewing direction (i.e., pitch and yaw angle). The system equation in Eq. (1) can hence be extended to an actuation to task space mapping f_s as follow:

$$\mathbf{s}_{k+1} = f_s(\boldsymbol{\theta}_k, \Delta\mathbf{u}_k) \quad (2)$$

where $\mathbf{s}_{k+1} = \mathbf{s}_k + \Delta\mathbf{s}_k$ is the task space coordinate at time step $k+1$ after the change in fluid pressure $\Delta\mathbf{u}_k$ is applied.

C. Inverse Problem for Online Learning of Task Space Control

Our control objective is to enable the operator to control the displacement of the robot directly in the task space coordinate $\Delta\mathbf{s}_k^*$ (i.e., the desired change in the robot tip orientation) with the use of a motion input device. The superscript '*' denotes the desired motion specified by the users or other reference input. Thus, the controller is designed to approximate the inverse of the motion transition f_s in (2), i.e. $\Delta\mathbf{u}_k = \tilde{\Phi}(\Delta\mathbf{s}_k^*, \boldsymbol{\theta}_k)$, in order to estimate the required change in control input $\Delta\mathbf{u}_k$ (as seen in **Fig. 5**). The inverse motion transition model $\tilde{\Phi}$ heavily depends on the current robot state. However, the exact state $\boldsymbol{\theta}_k$ cannot be directly measured due to its hyper-flexibility and the interactions with enclosed workspace inside a patient's cavity. We sought to adopt the task space coordinates $\mathbf{s}$ which would offer the updated clues about the current robot state. This approach is also of practical interest because these measurements are readily available in our control system. The task space coordinate $\mathbf{s}$ can be tracked using advanced positional tracking systems. For example, electromagnetic tracking systems are commonly used in medical application to provide sub-millimetre-level tracking [42, 43]. Together with the actuators input $\mathbf{u}_k$, these online acquired data are presented to the learning algorithms to update the inverse mapping Φ during robot run-time.

$$\Delta\mathbf{u}_k = \Phi(\Delta\mathbf{s}_k^*, \mathbf{s}_k, \mathbf{u}_k) \quad (3)$$

Note that Φ is the approximation of the true inverse mapping $\tilde{\Phi}$. If the dimensionality of the task space is smaller than that of the control space, theoretically there exists an infinite number of solutions of $\Delta\mathbf{u}_k$ that result in the same task space displacement $\Delta\mathbf{s}_k^*$. This leads to the *ill-posed* problem in learning the inverse mapping Φ.

D. Inverse model learning with multiple local controllers

Nonparametric local learning techniques have been applied to learn the ill-posed inverse problem, aiming to control redundantly actuated robots [31, 44, 45]. Referring to Peters *et al.* [36], the inverse model of a rigid link robot can be learnt using spatially localized nonparametric learning techniques, given that the robot state is well-defined by the joint kinematics. Here, the spatial localization refers to the robot state $\boldsymbol{\theta}_k$. Such localization scheme is motivated by the hypothesis that the inverse problem would be well-defined locally [36]. It is because nonparametric learning techniques essentially average out the sampled data. Model learning based on nonconvex training datasets would give invalid solutions [36]. However, in the vicinity of $(\mathbf{s},\mathbf{u})$, the average of $\Delta\mathbf{u}$ would be consistent with the average of the task space displacement $\Delta\mathbf{s}$ (**Fig. 2**). Therefore, in a local region of a given $(\mathbf{s},\mathbf{u})$, the training dataset $\{\Delta\mathbf{u},\Delta\mathbf{s},\mathbf{s},\mathbf{u}\}$ would become a convex set. This enables learning of the inverse mapping in the vicinity of $(\mathbf{s},\mathbf{u})$ (**Fig. 2**). We approximate the local inverse mapping from the desired task space displacement to the actuation command as follows:

$$\Delta\mathbf{u}_k = \Phi^i(\Delta\mathbf{s}_k^*, \mathbf{s}_k, \mathbf{u}_k) = [\Delta\mathbf{s}_k^*]^{\mathrm{T}}\beta^i \quad (4)$$

where β^i is the parameter of the local inverse model. Each mapping serves as a local controller. Compared to [36], we do not include an intercept/bias term, since the change of actuation command Δu should have zero mean. The computation of β^i will be explained in the later context.

E. Online Learning of the Global Controller

To approximate the global inverse mapping, we employ a linear combination of the locally learned mapping [46]:

$$\Delta\mathbf{u}_k = \frac{\sum_{i=1}^{n} w^i(\mathbf{s}_k,\mathbf{u}_k)\Phi^i(\Delta\mathbf{s}_k^*,\mathbf{s}_k,\mathbf{u}_k)}{\sum_{i=1}^{n} w^i(\mathbf{s}_k,\mathbf{u}_k)} = \frac{\sum_{i=1}^{n} w^i(\mathbf{s}_k,\mathbf{u}_k)[\Delta\mathbf{s}_k^*]^{\mathrm{T}}\beta^i}{\sum_{i=1}^{n} w^i(\mathbf{s}_k,\mathbf{u}_k)}. \quad (5)$$

This controller architecture allows straightforward, one-iteration computation in each time step, in contrast to indirect modeling approaches [34]. The number of local models n and the weight $w^i(\mathbf{s}_k,\mathbf{u}_k)$, as well as the local controllers $\Phi^i(\Delta\mathbf{s}_k^*,\mathbf{s}_k,\mathbf{u}_k)$ can be obtained in an online manner.

For this purpose, the local forward model is "learnt" using Locally Weighted Projection Regression (LWPR) [37] which offers piecewise linear function approximation, while simultaneously determines the appropriate local region of each linear model. Each local forward model performs a linear mapping as:

$$\Delta \mathbf{s}_k = f_{\mathbf{s}}^i(\mathbf{s}_k, \mathbf{u}_k, \Delta \mathbf{u}_k) = [\Delta \mathbf{u}_k]^{\mathrm{T}} \hat{\beta}^i \qquad (6)$$

where $\hat{\beta}^i$ denotes the corresponding parameter. Each local region, namely the receptive field (RF), is shaped based on the membership function:

$$w^i(\mathbf{s}_k, \mathbf{u}_k) = \exp\left(-0.5\left(\begin{bmatrix}\mathbf{s}_k\\ \mathbf{u}_k\end{bmatrix} - c^i\right)^{\mathrm{T}} \mathbf{D}^i \left(\begin{bmatrix}\mathbf{s}_k\\ \mathbf{u}_k\end{bmatrix} - c^i\right)\right) \qquad (7)$$

centered at c^i, where $\mathbf{D}^i$ is the distance metric. Each membership function weights the corresponding locally learned inverse model in the controller (6). One advantage of LWPR is that it can automatically spawn new linear models and the corresponding RF when new data laid outside all existing RF is presented. Meanwhile, the center c^i of RF is determined by the input space of new data through the incremental learning, so as the total number of local regions n **(Fig. 3)**. Each newly spawned RF is initialized with a diagonal distance metric $\mathbf{D}^i$ value. This $\mathbf{D}^i$ value will be updated throughout the incremental learning process to improve the overall regression accuracy and convergence rate. To prevent overfitting and allocation of too many numbers of RFs n, a smaller initial $\mathbf{D}^i$ value is preferred (i.e. larger receptive fields). Cross-validation is also employed in determining the initial $\mathbf{D}^i$, which is of important to ensure that the forward model can be accurately reflected by the piece-wise linear regression.

Despite the fact that each RF could fulfill the local convexity requirement, due to the redundancy in the robotic system, the solutions of the local controllers (4) could be inconsistent with the desired solutions [36]. Although this problem could be resolved by pre-processing the training data such that it only produce one particular solution, it lacks the generality and is difficult to apply in high dimensional systems [31]. Therefore, we employ another approach that reshape the local inverse models using constrained optimization, where the local controllers are enforced to provide consistent solutions from infinite possibilities in the null space of the control space. We then define the optimization problem as:

$$\min_{\Delta \mathbf{u}} C_k(\Delta \mathbf{u}_k) = (\Delta \mathbf{u}_k - \Delta \mathbf{u}_{0,k})^{\mathrm{T}} \mathbf{N} (\Delta \mathbf{u}_k - \Delta \mathbf{u}_{0,k}) \qquad (8)$$

$$\text{subject to } \Delta \mathbf{u}_k = \Phi(\Delta \mathbf{s}_k^*, \mathbf{s}_k, \mathbf{u}_k)$$

where the cost function C_k represents the user-defined optimality scaled by a diagonal matrix $\mathbf{N}$. $\Delta \mathbf{u}_{0,k} = \upsilon(\mathbf{s}_k, \mathbf{u}_k)$ is the user-defined null space behavior. One example of null space behavior could be minimizing the elongation of the robot, which results in smaller bending radius to facilitate dexterous motion inside enclosed cavity. Finally, the optimization constraint $\Delta \mathbf{u}_k = \Phi(\Delta \mathbf{s}_k^*, \mathbf{s}_k, \mathbf{u}_k)$ ensures the correctness of the inverse solution.

The constrained optimization problem can be solved by introducing a reward function (9) and a cost function (10):

$$r(\mathbf{u}_k) = \sigma_i \exp(-0.5\sigma_i^2 C_k) \quad (9)$$

$$E_i = \sum_{k=1}^{N} r(\mathbf{u}_k) w^i(\mathbf{s}_k, \mathbf{u}_k)(\Delta\mathbf{u}_k - [\Delta\mathbf{s}_k^{\mathrm{T}}]\boldsymbol{\beta}^i)^2 \quad (10)$$

The reward function $r(\mathbf{u}_k)$ is scaled by the mean cost σ_i to improve the learning efficiency [36]:

$$\sigma_i^2 = \sum_{h=1}^{k} w^i(\mathbf{s}_h, \mathbf{u}_h) C_h / \sum_{h=1}^{k} w^i(\mathbf{s}_h, \mathbf{u}_h) \quad (11)$$

The cost function is then minimized by means of reward-weighted regression, where each local model needed to be updated:

$$\boldsymbol{\beta}_{k+1}^i = (\mathbf{X}^{\mathrm{T}}\mathbf{W}^i\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{W}^i\mathbf{Y} \quad (12)$$

where $\mathbf{W}^i = diag(r(\mathbf{u}_1)w_1^i, \ldots, r(\mathbf{u}_k)w_k^i)$, $\mathbf{X} = [\Delta\mathbf{s}_k^*]$ and $\mathbf{Y} = [\Delta\mathbf{u}_k]$ are the training datasets. The overall procedures of the learning-based controller are summarized in **Algorithm 1**.

III. Experiments, Results & Discussion

The proposed control framework is implemented on a custom-made soft robot to investigate its performance and behavior under external dynamic constraints. We have also attempted to utilize FEA to simulate robot motion data for pre-training of an initial control policy. This can avoid the need for random exploration of its robot workspace to initialize online learning functions. Such exploration is usually time-consuming, and may not be practical, particularly for single-use purposes in surgical applications. Accuracy and stability of the proposed controller are examined via path following under various constrained environments. The interaction force with the external constraint is also measured throughout the experiments. The control block diagram of the overall robotic system, including the processing core and actuation system, is illustrated in **Fig. 6**.

A. Initialization of Online Learning by FEA-based model

Proper initialization of pre-training data is essential to many online learning techniques. These preceding data are dedicated to pre-training an initial control policy before the online learning begins. It is usually acquired by driving the robot with random input. Instead, we proposed to incorporate FEA, by which the robot deformation can be simulated with a hyper-elastic computation model. This simulation can generate comprehensive pre-training samples that cover the entire robot workspace at a high resolution, facilitating offline pre-training of the learning-based controller (**Fig. 5**).

The FEA model of the robot is constructed using ABAQUS [47] to predict the robot kinematics and workspace. The RTV silicone rubber is considered as incompressible hyper-elastic material formulated by Odgen material model [48]. It exhibits negligible volume change under hydrostatic compression and has a Poisson's ratio close to 0.5. Due to the incompressibility of silicone rubber and the large deformation nature of the simulation, the element formulation and the mesh quality pose a compelling effect on both the accuracy and convergence of the simulation. Therefore, hexahedral element (C3D8RH, **Fig. 1c**) based on u-p hybrid formulation with hourglass control [47] is chosen over the commonly used quadratic tetrahedral elements (**Fig. 1d**) in the FEA of our soft robotic manipulators. The C3D8RH element possesses eight displacement nodes and one interior pressure node. The combination of these displacement and pressure nodes are often close to optimal [49]. Such integration scheme improves, not only the element efficiency, but also the element accuracy under bending load. However, compared to tetrahedrons, automatic mesh generation of hexahedrons is relatively in effective, resulting in poor tessellation quality. To this end, the presented meshing has to be obtained by custom-designed protrusions, and all elements are right prisms initially. By restoring the mesh quality, the assemblage contains far fewer elements and is much more robust in convergence.

The presented manipulator model is tessellated with 12k linear hexahedral elements (C3D8RH, **Fig. 1c**). There are also 2,214 linear truss elements (T3D2) being placed along each actuation chamber in a layer-by-layer arrangement (**Fig. 1b**). Truss elements are used to model the helical strain-wrapping constraints that ensure the anisotropic expansion of the chambers upon a pressure actuation. Actuation and gravity loads are applied to the presented FEA model. The gradual change of the stress input, which is distributed across the surface mesh along the inner chamber surface, guarantees reliable convergence, giving rise to an equilibrium solution throughout all the time steps during the FEA. Quasi-static motion with negligible hysteresis can be achieved when the real robot prototype is manipulated while delicately regulating the inflation pressure into the chamber at high-resolution steps. It is worth noting that deformation/bending of both the FEA-modelled manipulator and the actual one are very similar corresponding to the same levels of inflation pressure simulated, as shown in **Fig. 4**. Over 1,000 simulated motion samples $\{\Delta\mathbf{u}, \Delta\mathbf{s}, \mathbf{s}, \mathbf{u}\}$ have been obtained using the FEA, covering the entire robot workspace (**Fig 5**). These simulated data are adopted to pre-train the online learning controller as described in the following sections.

B. Experimental Setup

To evaluate the proposed control performance, three motorized pneumatic units are employed to actuate the presented soft manipulator incorporated with our close-loop control testing platform (**Fig. 6)**. Each unit consists of a pneumatic cylinder coupled to a precise stepper motor through a leadscrew transmission. This facilitates accurate regulation of the air flow. Our soft robotic manipulator can be fully articulated in a dome-shaped workspace with a maximum curve angle of >150˚ in all directions.

An electromagnetic (EM) tracking system (NDI Medical Aurora) is employed to close the robot control loop by the continuous positional data feedback (**Fig. 7a**). This tracking system is commonly available in many image-guided intervention systems. It can track of the position and orientation of tiny EM coils in real time with RMS accuracy of 0.7mm and 0.2° at 40Hz. A tiny tracking coil is embedded at the robot distal tip. Online updating (at 20Hz) of the inverse mapping estimation $\Delta\mathbf{u}_k = \Phi(\Delta\mathbf{s}_k^*, \mathbf{s}_k, \mathbf{u}_k)$ by the local learning algorithm is achieved, where $\mathbf{s}_k$ is measured tip direction. The positional data is also recorded throughout the robot task so as to evaluate the overall control performance. The entire control framework is implemented in the MATLAB environment. The open source library of LWPR [50] is employed to incrementally learn the robot forward model, which determines the valid linearization of each local controllers.

A series of path following tasks is performed under various constraints scenarios to investigate how the online learning control approach reacts to such unknown interactions. At the beginning, the robot is allowed to move freely in its workspace without any interference. This serves as the control experiment to establish the baseline of controller performance. Subsequently, the robot is gently pushed by a plastic rod in order to simulate an unknown, dynamic interaction with the robot manipulation (**Fig. 7b**). The rod is actuated by a high precision stepping motor to generate repeatable contact with the robot body; meanwhile, the contact force is monitored by a force/torque sensor (ATI Industrial Automation: F/T Nano17). The tracking error is defined as the shortest distance between the robot targeting direction $\mathbf{s}_k$ and the desired trajectory.

C. Evaluation of Online Local Learning Controller

To realize accurate navigation under unknown constraints, the inverse model is adapted in the proposed learning-based controller, which has to be updated online based on the newly acquired motion data. In this study, we compared three types of data sources for the inverse models training: **i)** pre-trained by FEA data without using online data; **ii)** Initialized by random exploration with online learning data; and **iii)** Pre-trained by the FEA data, and then updated by online data. These online-updated inverse models are evaluated for resolved motion rate control [51] to track a pre-defined trajectory. Thus, the desired task space displacement $\Delta\mathbf{s}_k^*$ that tracks the reference input is obtained as follow:

$$\Delta\mathbf{s}_k^* = \Delta\mathbf{s}_k^{ref} + \mathbf{K}_p^{ref}(\mathbf{s}_k^{ref} - \mathbf{s}_k) \quad (13)$$

where $\Delta\mathbf{s}_k^{ref}$ and $\mathbf{s}_k^{ref}$ are the reference task space displacements and coordinates generated from interpolating a pre-defined trajectory. Note that the reference input can be replaced by manual control in actual endoscopic navigation scenario. We employed the same proportional–derivative (PD) gain $\mathbf{K}_p^{ref} = \mathbf{I}$ for all three settings to perform tracking along a reference trajectory. Thus, the actuation input $\Delta\mathbf{u}_k$ is estimated by the online learning inverse model as depicted Eq. (4).

To enforce the consistency of inverse mapping among all localized linear controllers, a standard null space behavior $\Delta\mathbf{u}_{0,k} = \upsilon(\mathbf{s}_k, \mathbf{u}_k)$ is defined. This gives rise to an immediate reward function $r(\Delta\mathbf{u}_k)$ to weigh the training data that best imitates the desired null space behavior (Eq. (9)). For the presented soft robot, we first choose a rest configuration to be $\mathbf{u}_{rest} = [0,0,0]^{\mathrm{T}}$, which can minimize the overall inflation pressure as well as the elongation of the manipulator. Then the robot is attracted towards to the rest configuration with a loose attractor function $\Delta\mathbf{u}_{0,k} = -\mathbf{K}_p(\mathbf{u}_k - \mathbf{u}_{rest})$, where $\mathbf{K}_p = 0.2\mathbf{I}$. We defined an identity metric $\mathbf{N} = \mathbf{I}$ as all three inflatable actuators of the robot are identical and should contribute the same in achieving the desired null space behavior.

It is also necessary to normalize the training dataset into the same scale component-wise so that the LWPR can learn the data variance properly. Min-max normalization is a simple but effective technique commonly used [52]:

$$\hat{q}_i = \frac{q_i - \min(q_i)}{\max(q_i) - \min(q_i)} \qquad (14)$$

However, the statistical $\max(q_i)$ and $\min(q_i)$ values would be sensitive to outliers; therefore, we define the min-max values according to the physical constraints of data, including the typical robot workspace and the maximum volume of the cylinder unit.

i) Pre-trained by FEA without using online data: In this setting, both the forward model and control policy are pre-trained solely by the FEA simulated data (**Section-IIIA**). The online data was not taken account in this setting. This acts as a control experiment to depict the actual influence of the external interactions. In the unconstrained experiment (**Fig. 8a**), it was observed that the controller could roughly follow the trajectory with a relatively large tracking error of ±1.79º and a maximum error of ±6.96º with the use of the feedback controller **(Table I)**. Despite the considerable discrepancy between the FEA-simulated and the actual configuration, this experiment still demonstrates that the FEA data is capable of pre-training a reasonable inverse model for rough path following.

In the later constrained experiment (**Fig. 9a**), the robot maintained tracking of the trajectory with similar accuracy at the beginning. When the external interaction is engaged at the moment of 25 second, the robot was pushed further away from the desired trajectory, resulting in an increased mean tracking error ±4.64º and a maximum error of ±14º (**Table II**). This indicates that the feedback controller cannot fully compensate the significant motion bias that is induced by the external disturbance. In the case of a conventional rigid-linked robot, this kind of error due to the interaction with the constraint is often considered as a perturbation. The error can hence be compensated by increasing the feedback control gain, given that the inverse model is readily available from the kinematics chain. However, such approach is not directly applicable to a soft robot due to their mechanical compliance that inevitably induces much larger positioning errors. In addition, the interaction force may also alter the force equilibrium of the robot and therefore, substantially degrading the reliability of the predetermined inverse model. The following experiments demonstrate how the proposed online algorithm can accommodate the influence of constrained environment, which is particularly demanding for the control of soft robots.

ii) Initialized by random exploration with online learning: The random exploration of robot workspace is a typical approach [34] to initialize a data-driven controller before its actual deployment. This kind of arbitrary movement is necessary to provide preceding data for setting up a learning model. It involves tracking 50 random input pressure waypoints $\mathbf{u}_k$ with a PD feedback controller. The deliberately-tuned PD gains can cause poor tracking of the random waypoints. Such babbling movement (green path in **Fig 8b** & **9b**) can facilitate faster learning rate as the robot sweep throughout a wider neighboring workspace. Pre-training with the exploration data resulted in a forward LWPR model with 110 receptive fields, which define the linearization for the piecewise linear inverse model in advance to actual deployment of the online learning.

Upon exploration, the online learning controller could follow the desired trajectory with an average error of ±1.13° in the first cycle under the constraint-free environment (**Fig. 8b**). The error was found to be significantly lower than the inverse model pre-trained by FEA simulated data. It is reasonable because the actual robot data was used. After a few cycles, the tracking error further decayed to an average of ±0.87° and maximum of ±1.92°, as having the online learning controller adapted with the trajectory.

Next, the feasibility of online inverse model adaptation was validated by engaging external force interactions (**Fig. 9b**). The online learning controller can compensate the bias, and hence minimize the error down to an average of ±2.35° within 5 seconds upon contact with the constraint. The external constraint is moved away after 30 seconds of contact. It is also worth noting that the controller could quickly update the inverse mapping online and follow the trajectory will high accuracy. No control instability is observed throughout the experiment. The pure online learning approach achieves the highest average accuracy among all settings, both for constrained and unconstrained scenarios **(Table I & II).** However, the need for initialization by "babbling motion" (green path in **Fig. 8b** and **9b**) should be avoided in clinical scenarios to prevent unnecessary interactions with patient anatomy.

iii) Pre-trained by FEA data, then updated by online data: To alleviate the need for random exploration, we attempted to pre-train the controller with FEA data and then update the inverse model by online learning. This approach combines the advantages of the both aforementioned settings, in which the inverse model can be initialized with FEA data. The robot can immediately begin navigation using this pre-trained model, without the need of the initialization through undesired babbling movement. The subsequent manipulation data are also acquired to incrementally train a more precise inverse model, so as to adapt to external interactions. This feature is demonstrated in **Fig. 8c**, in which the robot is allowed to move freely.

Although the robot begins with a relatively large tracking error of average ±2.21° and maximum of ±7.49° in the first cycle, the error is quickly compensated by the online learning and converged to an average of ±0.90° and maximum of ±2.80°. This tracking result is compared with the other two approaches in **Table I**. In the first cycle, the combined approach exhibit tracking error close to pre-training with FEM only (Avg. ±2.21° vs ±1.79°), because both inverse models are initialized with less accurate FEM data. The learning technique then correct the inverse model with online data, so that the tracking error decrease rapidly and become comparable with the pure online approach (Avg. ±0.90° vs ±0.87°). This shows that the combined approach can initialize a reasonable learning-based controller with less accurate FEM data, then further refine the inverse model while performing the tracking task. Note that the combined approach does not required random exploration (green path in **Fig. 8b** and **9b**) to obtain pre-training data, which is difficult to cover the entire robot workspace with sufficient density.

This combined approach is also capable of adapting to the unknown external interaction (**Fig. 9c**). The inverse model can quickly adapt the inverse mapping upon contact with the external interaction at 36s. It continues to follow the trajectory with a small mean absolute error of ±2.49°. The controller also remains stable and re-adapts after the removal of the constraints. Readers could also refer to the attached video for extra details about the robot behavior and the characteristics of constraint. Referring to **Section II-C**, we presented the challenge in learning an inverse model spatially localized by the unmeasurable robot state $\boldsymbol{\theta}_k$, as well as how this robot state can be retrieved indirectly from sensory measurements. These trajectory tracking experiments have shown that the inverse model could be successfully learnt by continuous updates of both the task space coordinate $\mathbf{s}_k$ and control input $\mathbf{u}_k$. Both are set as the localization parameters required in the inverse model. Therefore, the robot state $\boldsymbol{\theta}_k$ could be estimated sustainably by the learning algorithm. These 3-6D positional data updates are clinically practical. The comparable position tracking techniques designed for image-guided interventions are also under active research [53], one of which would be MRI-guided endoscopic retrograde cholangio-pancreatography (ERCP).

IV. Conclusion and Future work

We have proposed a model-free control framework which adopts an online nonparametric local learning technique for manipulation of a redundantly-actuated, fluid-driven soft continuum robot in the presence of a dynamic external disturbance. Nonparametric techniques are capable of constructing highly nonlinear functions by measurement data solely, which is particularly suitable for characterization of hyper-elastic robot structure. To accommodate the flexibility of soft robot body, we approximate the global inverse kinematics by a linear combination of many locally learnt inverse kinematic models. Our model-free controller employs this global approximation, where the behavior of the redundant actuator can be optimized by a user-defined criterion, and simultaneously fulfilling the control objective defined in task space coordinates. In addition, the controller is adaptive to changes in the environment, where each local model can be updated online independently according to newly acquired data. This equips the robot with the ability to maintain control accuracy under external dynamic disturbance. Our work is the *first* attempt of implementing such direct inverse modeling using online nonparametric learning technique to control a redundantly-actuated soft continuum robot.

We have also incorporated FEA into the learning control framework for proper initialization of the robot inverse model. It enables precise prediction of the hyper-elastic robot deformation under various actuation pressure, without the need for the over-simplified analytical model. It can also offer adequate sample data covering the entire workspace at high resolution. This avoids the need of time-consuming random exploration to initialize the learning model, which may not be practical in many surgical applications. The proposed controller can hence be initialized offline using FEA simulated data, ready for endoscopic navigation procedure.

The proposed novel control framework has been experimentally validated. In the constrained experiment, after FEA-based initialization of the controller, the endoscope prototype could follow a 3-D trajectory with an accuracy of mean ±2.21° and max. ±7.49°, and attained the almost the same tracking accuracy (mean ±2.49° and max. ±11.03°) after 5 seconds upon the addition/removal of external disturbance (max. 1N). This is also the *first* demonstration of realizing model-free close-loop control of a fluid-driven soft continuum in 3-D task space even under dynamic external disturbance.

The current form of our learning-based control method is first designed for a single segment manipulator. In our future work, we intend to extend the framework to address soft manipulation with multi-segments [54]. As a cascade of multiple actuation modules, it provides enhanced manipulation flexibility for interventional tools, facilitating more complicated operations in confined space. In this case, a generic optimization function will be developed to resolve the null-space control of hyper-redundant robot [55]. Further characterization of such multi-segment soft manipulators will be investigated. To address its hyper-redundancy, it will also require additional sensory systems or algorithms to parameterize the possible motion transition of robot configuration, thus estimating the inverse model for the higher DoF robot.

Acknowledgment

This work is supported in parts by the Croucher Foundation, the Research Grants Council (RGC) of Hong Kong (Ref. No. 27209151 and No. 17227616) and NISI (HK) Limited.

Author Disclosure Statement

No competing financial interests exist.

Reference

1. Trivedi, D., et al., *Soft robotics: Biological inspiration, state of the art, and future research.* Applied Bionics and Biomechanics, 2008. **5**(3): p. 99-117.

2. Mao, S., et al., *Gait study and pattern generation of a starfish-like soft robot with flexible rays actuated by SMAs.* Journal of Bionic Engineering, 2014. **11**(3): p. 400-411.

3. Sareh, S., et al. *Bio-inspired tactile sensor sleeve for surgical soft manipulators*. in *IEEE International Conference on Robotics and Automation (ICRA)*. 2014. IEEE.

4. Suzumori, K., S. Iikura, and H. Tanaka. *Development of flexible microactuator and its applications to robotic mechanisms*. in *The 1991 IEEE International Conference on Robotics and Automation*. 1991.

5. Wang, L. and F. Iida, *Deformation in soft-matter robotics.* IEEE Robotics & Automation Magazine, 2015. **22**(3): p. 125-139.

6. McMahan, W., et al. *Field trials and testing of the OctArm continuum manipulator*. in *IEEE International Conference on Robotics and Automation (ICRA)*. 2006. IEEE.

7. Runge, G., et al. *SpineMan: Design of a soft robotic spine-like manipulator for safe human-robot interaction*. in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2015. IEEE.

8. Maghooa, F., et al. *Tendon and pressure actuation for a bio-inspired manipulator based on an antagonistic principle*. in *IEEE International Conference on Robotics and Automation (ICRA)*. 2015. IEEE.

9. Cianchetti, M., et al., *Soft robotics technologies to address shortcomings in today's minimally invasive surgery: the STIFF-FLOP approach.* Soft Robotics, 2014. **1**(2): p. 122-131.

10. Lohsiriwat, V., *Colonoscopic perforation: Incidence, risk factors, management and outcome.* World Journal of Gastroenterology, 2010. **16**(4): p. 425.

11. Tumino, E., et al., *Endotics system vs colonoscopy for the detection of polyps.* World J Gastroenterol, 2010. **16**(43): p. 5452-5456.

12. Cosentino, F., et al., *Functional evaluation of the endotics system, a new disposable self-propelled robotic colonoscope: in vitro tests and clinical trial.* The International journal of artificial organs, 2009. **32**(8): p. 517-527.

13. Pfeffer, J., et al., *The Aer-O-Scope: proof of the concept of a pneumatic, skill-independent, self-propelling, self-navigating colonoscope in a pig model.* Endoscopy, 2006. **38**(02): p. 144-148.

14. Fras, J., et al. *New STIFF-FLOP module construction idea for improved actuation and sensing*. in *IEEE International Conference on Robotics and Automation (ICRA)*. 2015. IEEE.

15. Camarillo, D.B., et al., *Mechanics modeling of tendon-driven continuum manipulators.* IEEE Transactions on Robotics, 2008. **24**(6): p. 1262-1273.

16. Jones, B. and I.D. Walker, *Kinematics for multisection continuum robots.* IEEE Transactions on Robotics, 2006. **22**(1): p. 43-55.

17. Mahvash, M. and P.E. Dupont, *Stiffness control of surgical continuum manipulators.* IEEE Transactions on Robotics, 2011. **27**(2): p. 334-345.

18. Webster III, R.J. and B.A. Jones, *Design and kinematic modeling of constant curvature continuum robots: A review.* The International Journal of Robotics Research, 2010.

19. Ganji, Y. and F. Janabi-Sharifi, *Catheter kinematics for intracardiac navigation.* IEEE Transactions on Biomedical Engineering, 2009. **56**(3): p. 621-632.

20. Jones, B.A. and I.D. Walker. *A New Approach to Jacobian Formulation for a Class of Multi-Section Continuum Robots*. in *IEEE International Conference on Robotics and Automation (ICRA)*. 2005.

21. Webster III, R.J., et al., *Closed-form differential kinematics for concentric-tube continuum robots with application to visual servoing.* Experimental Robotics: The Eleventh International Symposium, 2009: p. 485-494.

22. Neppalli, S., et al., *Closed-form inverse kinematics for continuum manipulators.* Advanced Robotics, 2009. **23**(15): p. 2077-2091.

23. Marchese, A.D., et al., *Design and control of a soft and continuously deformable 2d robotic manipulation system*, in *The IEEE International Conference on Robotics and Automation*. 2014. p. 2189-2196.

24. Marchese, A.D., R. Tedrake, and D. Rus, *Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator.* The International Journal of Robotics Research, 2016. **35**(8): p. 1000-1019.

25. Wang, H., et al., *Visual servo control of cable-driven soft robotic manipulator*, in *The 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2013. p. 57-62.

26. Chirikjian, G.S., *Hyper-redundant manipulator dynamics: A continuum approximation.* J. Adv. Robot, 1995. **9**(3): p. 217-243.

27. Kang, R., et al., *Design, modeling and control of a pneumatically actuated manipulator inspired by biological continuum structures.* Bioinspiraton& Biomimetics, 2013. **8**: p. 036008.

28. Yekutieli, Y., et al., *Dynamic model of the octopus arm. I. biomechanics of the octopus reaching movement.* J. Neurophysiol, 2005. **94**(2): p. 1443-1458.

29. Giorelli, M., et al., *A two dimensional inverse kinetics model of a cable driven manipulator inspired by the octopus arm*, in *the 2012 IEEE International Conference on Robotics and Automous Systems*. 2012. p. 3819-3824.

30. Renda, F., et al., *Dynamic model of a multibending soft robot arm driven by cables.* IEEE Transactions on Robotics, 2014. **30**(5): p. 1109-1122.

31. Nguyen-Tuong, D. and J. Peters, *Model learning for robot control: a survey.* Cognitive processing, 2011. **12**(4): p. 319-340.

32. Braganza, D., et al., *A neural network controller for continuum robots.* Robotics, IEEE Transactions on, 2007. **23**(6): p. 1270-1277.

33. Giorelli, M., et al., *Neural Network and Jacobian Method for Solving the Inverse Statics of a Cable-Driven Soft Arm With Nonconstant Curvature.* Robotics, IEEE Transactions on, 2015. **31**(4): p. 823-834.

34. Yip, M.C. and D.B. Camarillo, *Model-less feedback control of continuum manipulators in unknown environments.* IEEE Trans. Robot, 2014. **30**(4): p. 880-889.

35. Nguyen-Tuong, D., M. Seeger, and J. Peters, *Model learning with local gaussian process regression.* Advanced Robotics, 2009. **23**(15): p. 2015-2034.

36. Peters, J. and S. Schaal, *Learning to control in operational space.* The International Journal of Robotics Research, 2008. **27**(2): p. 197-212.

37. Vijayakumar, S., A. D'souza, and S. Schaal, *Incremental online learning in high dimensions.* Neural computation, 2005. **17**(12): p. 2602-2634.

38. Largilliere, F., et al. *Real-time control of soft-robots using asynchronous finite element modeling*. in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. 2015. IEEE.

39. Duriez, C. *Control of elastic soft robots based on real-time finite element method*. in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. 2013. IEEE.

40. Faudzi, A.A.M., et al. *Development of bending soft actuator with different braided angles*. in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. 2012. IEEE.

41. Greigarn, T. and M.C. Cavusoglu. *Task-space motion planning of MRI-actuated catheters for catheter ablation of atrial fibrillation*. in *International Conference on Intelligent Robots and Systems (IROS 2014)*. 2014. IEEE.

42. Ko, S.Y., L. Frasson, and F.R. y Baena, *Closed-loop planar motion control of a steerable probe with a "programmable bevel" inspired by nature.* IEEE Transactions on Robotics, 2011. **27**(5): p. 970-983.

43. Xu, R., et al. *Position control of concentric-tube continuum robots using a modified Jacobian-based approach*. in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. 2013. IEEE.

44. Hartmann, C., et al. *Real-Time Inverse Dynamics Learning for Musculoskeletal Robots based on Echo State Gaussian Process Regression*. in *Robotics: Science and Systems*. 2012.

45. Sigaud, O., C. Salaün, and V. Padois, *On-line regression algorithms for learning mechanical models of robots: a survey.* Robotics and Autonomous Systems, 2011. **59**(12): p. 1115-1129.

46. Schaal, S., C.G. Atkeson, and S. Vijayakumar, *Scalable techniques from nonparametric statistics for real time robot learning.* Applied Intelligence, 2002. **17**(1): p. 49-60.

47. Simulia, D., *ABAQUS 6.13 User's Manual.* Dassault Systems, Providence, RI, 2013.

48. Başar, Y. and M. Itskov, *Finite element formulation of the Ogden material model with application to rubber-like shells.* International Journal for Numerical Methods in Engineering, 1998. **42**(7): p. 1279-1305.

49. Hughes, T.J., *The finite element method: linear static and dynamic finite element analysis*. 2012: Courier Corporation.

50. Klanke, S., S. Vijayakumar, and S. Schaal, *A library for locally weighted projection regression.* Journal of Machine Learning Research, 2008. **9**(Apr): p. 623-626.

51. Whitney, D.E., *Resolved Motion Rate Control of Manipulators and Human Prostheses.* IEEE Transactions on Man-Machine Systems, 1969. **10**(2): p. 47-53.

52. Jain, A., K. Nandakumar, and A. Ross, *Score normalization in multimodal biometric systems.* Pattern recognition, 2005. **38**(12): p. 2270-2285.

53. Chen, Y., et al., *Design and Fabrication of MR-Tracked Metallic Stylet for Gynecologic Brachytherapy.* IEEE/ASME Transactions on Mechatronics, 2015. **PP**(99): p. 1-1.

54. Sadati, S.H., et al., *Stiffness Control of Soft Robotic Manipulator for Minimally Invasive Surgery (MIS) Using Scale Jamming*, in *Intelligent Robotics and Applications*. 2015, Springer. p. 141-151.

55. Kwok, K.W., et al., *Dimensionality reduction in controlling articulated snake robot for endoscopy under dynamic active constraints.* IEEE Transactions on Robotics, 2013. **29**(1): p. 15-31.

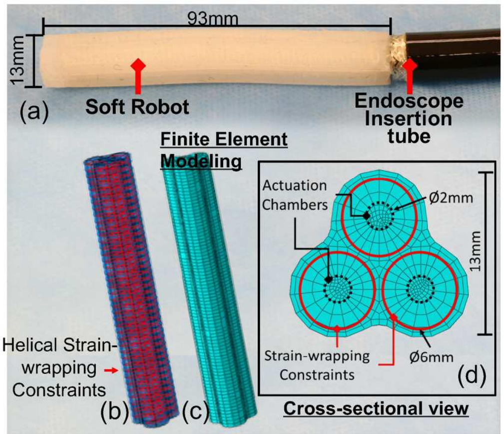

FIG. 1. (a) Soft robotic endoscope prototype made of silicone rubber. It has a dimension compatible with the insertion tube of conventional endoscope; (b) CAD/CAM model of the soft manipulator showing the simulated helical strain-wrapping constrains around its individual actuation chamber using linear truss, where the anisotropic expansion can be achieved; (c) Finite element model tessellated with 12,000 linear hexahedron elements. A total of 2,214 truss elements are defined to emulate the effect of strain-wrapping constraint; (d) Cross-sectional area tessellated by hexahedron meshing.

212x140mm (300 x 300 DPI)

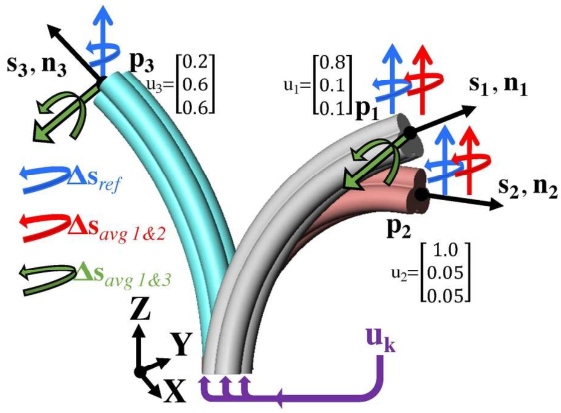

FIG. 2. Three robot configurations illustrating the example of localized inverse models. Assume that their tip directions $\mathbf{s}_i$ will undergo the same rotation $\Delta\mathbf{s}_{ref}$ (blue arrow) with the proper pressure changes $\Delta\mathbf{u}_i$ applied, where i=1…3. (a) In the case of configurations 1 and 2, the average of their control inputs $\Delta\mathbf{u}_1$ and $\Delta\mathbf{u}_2$ would still lead to a consistent $\Delta\mathbf{s}_{avg}$ (red arrow); (b) While two configurations, such as 1 and 3, are vastly different, this average of inputs $\Delta\mathbf{u}_1$ and $\Delta\mathbf{u}_3$ may lead to two dramatically different rotations $\Delta\mathbf{s}_{\mathrm{avg}}$, leading to undesired movement. Therefore, learning the inverse model directly with a global function approximator may lead to invalid solutions and unstable robot performance.

210x148mm (300 x 300 DPI)

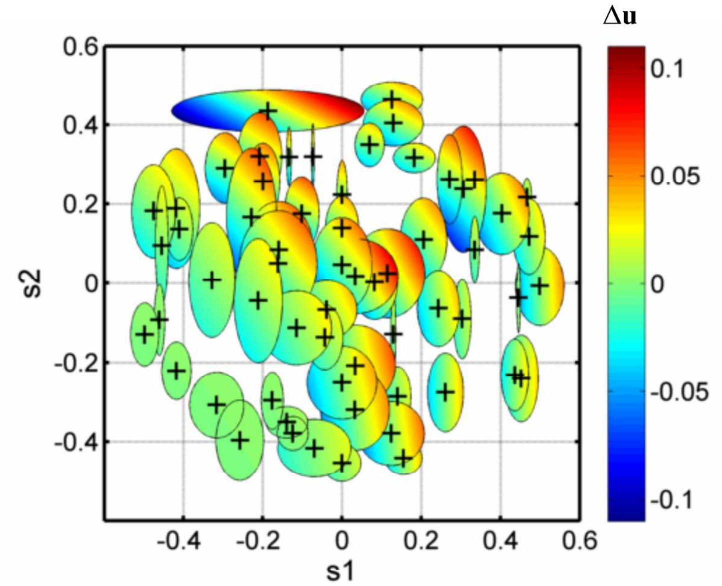

FIG. 3. Example set of localized linear controllers that approximate the nonlinear inverse mapping Φ of a 1-D actuation $\Delta\mathbf{u}$. The valid region of each spatially localized controller is centered at $\mathbf{c}_i$ (denoted by plus sign), with their range parameterized by $\mathbf{D}_i$ (colored ellipse) in the robot state space. The warm color depicts the actuation $\Delta\mathbf{u}$ predicted by the linear control law β^i, in order to achieve a particular movement $\Delta\mathbf{s}^*$ in task space.

204x188mm (300 x 300 DPI)

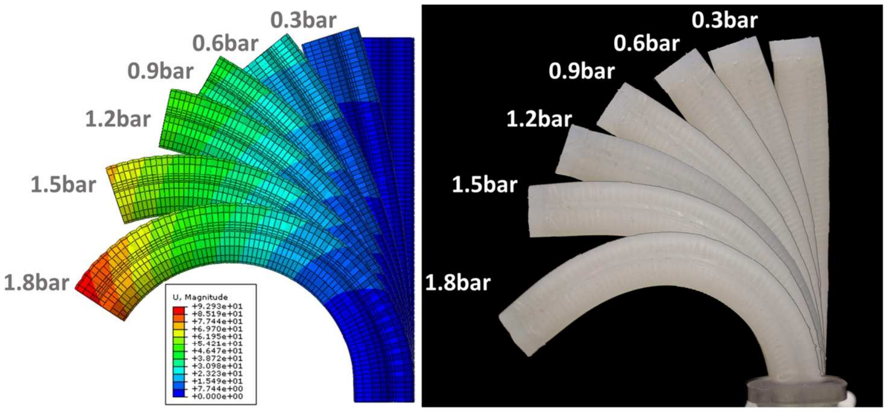

FIG. 4. FEA models (Left) simulated with 7 levels of inflation pressure in a single chamber. Similar deformation characteristics are exhibited in actual configurations of the soft manipulator (Right) under the same corresponding pressure levels.

172x93mm (300 x 300 DPI)

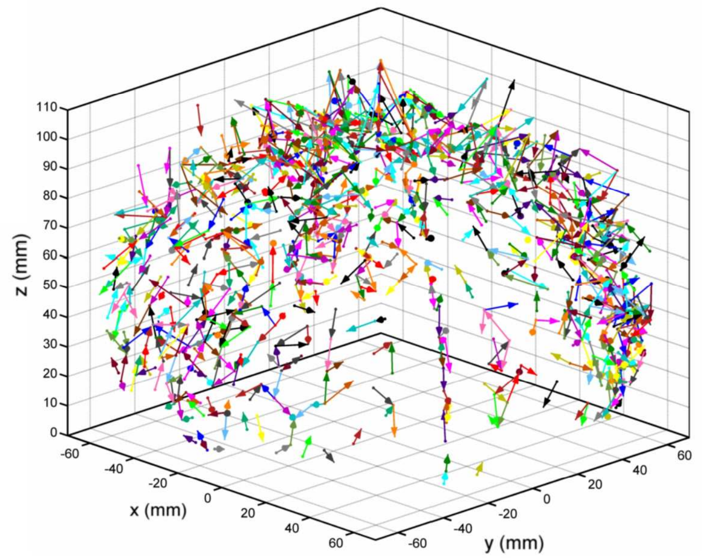

FIG. 5. FEA-simulated kinematics data covering the entire workspace of the soft robot. The arrows illustrate the predicted movement of the robot tip when an arbitrary pressure change $\mathbf{\Delta u}_k$ is applied. These data enable pre-training of a reasonable initial control policy before the online learning begins, without the need for undesired random movement (babbling).

188x153mm (300 x 300 DPI)

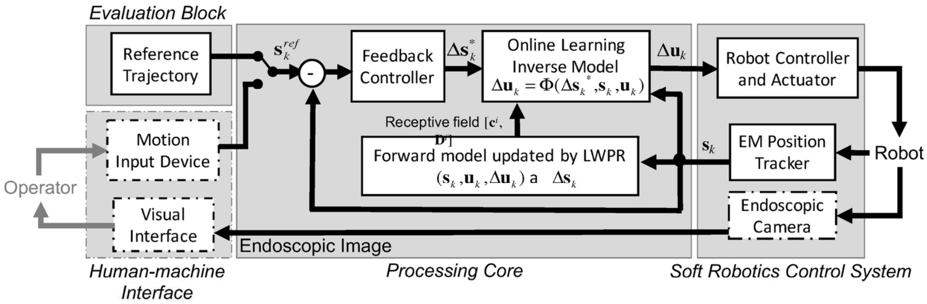

FIG. 6. System architecture of the proposed control framework depicting interconnection of the key components. The processing core is responsible for fast computation of inverse solution. The inverse model is also updated continuously by incorporating the online data in real time. The operator can specify the reference input s^{ref}_k via a motion input device for effective endoscopic navigation. In our experiments, this input is replaced by a pre-defined reference trajectory to evaluate the online learning performance of the inverse mapping.

161x80mm (300 x 300 DPI)

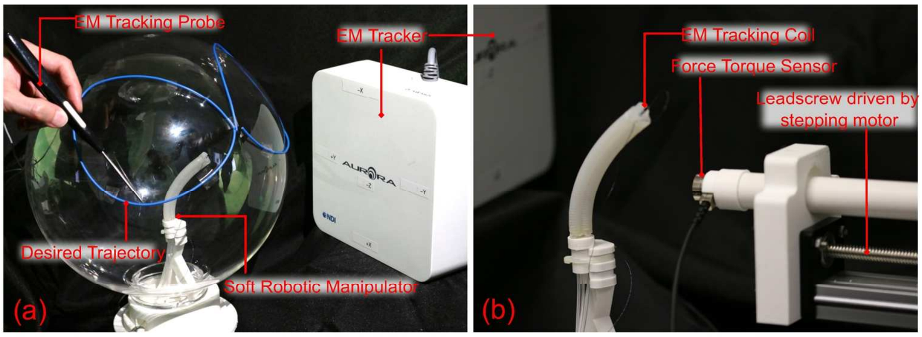

FIG. 7. (a) Registration process of the predefine trajectory using an EM position tracking system. Blue line on the transparent sphere illustrates the tracking trajectory on the task space; (b) Soft manipulator is commanded to follow the desired trajectory automatically. Its end-effector position is also measured by the tracking system to close the feedback loop under online learning control policy. Plastic rod actuated by a stepping motor pushes against the soft robot, generating the external constrains. The contact force is monitored by a force/torque sensor.

166x82mm (300 x 300 DPI)

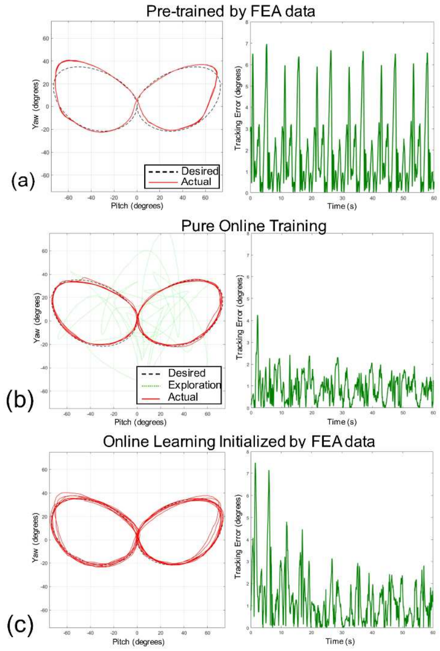

FIG. 8. Tracked trajectory plotted (Left), and the corresponding tracking error in time domain (Right). In the control experiment, the robot is allowed to move freely without any constraint. Control performance of the online learning controllers trained by three different data source is validated: (a) Pre-trained by FEA without using online data; (b) Pure online learning initialized by random exploration; (c) Pre-trained by FEA data and updated by online data. The online learning initialized by FEA data approach (c) combines the advantage of (a) and (b), in which random exploration (Green path in (b)) in not required, but its tracking error converge to similar accuracy as in pure online learning.

300x268mm (300 x 300 DPI)

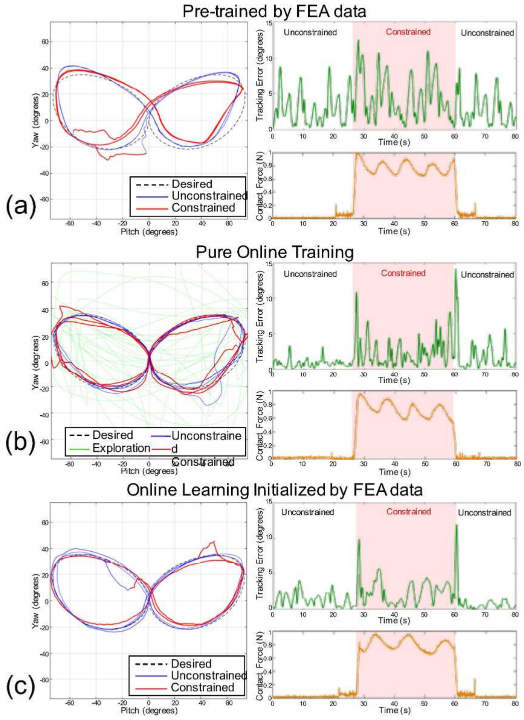

FIG. 9. Tracked trajectory plotted (Left), and the corresponding tracking error in time domain (Right). under external interactions. Control performance is validated in the three different conditions as in **Fig. 8**. It can be observed that the online learning for (b) and (c) is capable of compensating the external interaction with the tracking error reduced, as compared to the controller without using online data (a).

279x251mm (300 x 300 DPI)

Algorithm 1: Online Learning Algorithm of Inverse Mapping

1 **for each new input data sample**: $[\Delta\mathbf{s}_k^*, \mathbf{s}_k, \mathbf{u}_k, \Delta\mathbf{u}_k]$
2   Add $(\mathbf{s}_k, \mathbf{u}_k, \Delta\mathbf{u}_k) \rightarrow \Delta\mathbf{s}_k^*$ to the forward model LWPR.
3   Update the current number of models n and localization
  of the forward models $w^i(\mathbf{s}, \mathbf{u})$ for all input data
  Compute desired null-space behavior
4     $\Delta\mathbf{u}_{0,k} = \upsilon(\mathbf{s}_k, \mathbf{u}_k)$.
  Compute costs $C_k = (\Delta\mathbf{u}_{1,k})^{\mathrm{T}} \mathbf{N} \Delta\mathbf{u}_{1,k}$, where
5     $\Delta\mathbf{u}_{1,k} = \Delta\mathbf{u}_k - \Delta\mathbf{u}_{0,k}$
6   **for each model** $i = 1, 2, \ldots, n$
    Update the mean cost:
7       $\sigma_i^2 = \sum_{h=1}^{k} w^i(\mathbf{s}_h, \mathbf{u}_h) C_h / \sum_{h=1}^{k} w^i(\mathbf{s}_h, \mathbf{u}_h)$.
    Compute reward:
8       $r(\Delta\mathbf{u}_k) = \sigma_i \exp(-0.5\sigma_i^2 C_k)$
    Solve the following reward-weighted regression problem
    with step 10-13:
9       $E_i = \sum_{k=1}^{N} r(\mathbf{u}_k) w^i(\mathbf{s}_k, \mathbf{u}_k)(\Delta\mathbf{u}_k - [\Delta\mathbf{s}_k^{\mathrm{T}}]\boldsymbol{\beta}^i)^2$
    Add new data point to the weighted regression:
10       $\mathbf{X}_k = [\Delta\mathbf{s}_k^*]$
11       $\mathbf{Y}_k = [\Delta\mathbf{u}_k]$
12       $\mathbf{W}^i = diag(r(\mathbf{u}_1) w_1^i, \ldots, r(\mathbf{u}_k) w_k^i)$
13   Update the weighted regression of inverse mapping model
       $\boldsymbol{\beta}_{k+1}^i = (\mathbf{X}^{\mathrm{T}} \mathbf{W}^i \mathbf{X})^{-1} \mathbf{X}^{\mathrm{T}} \mathbf{W}^i \mathbf{Y}$
14   end
15 end

212x270mm (300 x 300 DPI)

TABLE I – TRAJECTORY TRACKING PERFORMANCE UNDER FREELY MOVEABLE ENVIRONMENT

Training Mode	Mean Abs. Error		Max. Abs. Error		Error SD σ	
	First Cycle	After	First Cycle	After	First Cycle	After
Pre-trained by FEA only	±1.79°	±1.82°	±6.96°	±6.56°	1.71°	1.66°
Pure Online Learning	±1.13°	±0.87°	±4.24°	±1.92°	0.86°	0.45°
Combined	±2.21°	±0.90°	±7.49°	±2.80°	1.81°	0.65°

84x19mm (300 x 300 DPI)

TABLE II – TRAJECTORY TRACKING PERFORMANCE UNDER CONSTRAINED ENVIRONMENT

Training Mode	Mean Abs. Error	Max. Abs. Error	Error SD σ	Mean Contact Force	Max. Contact Force
Pre-trained by FEA only	±4.64°	±14.00°	3.20 °	0.77N	1.06N
Pure Online Learning	±2.35°	±18.38°	2.47 °	0.78N	0.97N
Combined	±2.49°	±11.03°	1.74 °	0.80N	0.96N

76x14mm (300 x 300 DPI)