Robotica (2022), **40**, pp. 1098–1124
doi:10.1017/S026357472100093X

CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Dynamics and path tracking of continuum robotic arms using data-driven identification tools

Aida Parvaresh and S. Ali A. Moosavian*

Center of Excellence in Robotics and Control, Advanced Robotics & Automated systems (ARAS) Laboratory, Dept of Mechanical Engineering, K. N. Toosi University of Technology, Tehran, Iran
***Corresponding author.** E-mail: moosavian@kntu.ac.ir

Received: 4 October 2019; **Revised:** 13 June 2021; **Accepted:** 14 June 2021; **First published online:** 9 August 2021

Keywords: Continuum robotic arm; path tracking; forward dynamics problem; inverse dynamics problem; system identification; load carrying maneuver

Abstract

In this paper, forward/inverse dynamics of a continuum robotic arm is developed using a data-driven approach, which could tackle uncertainties and extreme nonlinearities to obtain reliable solutions. By establishing a direct mapping between the actuator and task spaces, the unnecessary mappings of actuator-to-configuration then configuration-to-task are eliminated, to reduce extra computational cost. The proposed approach is validated through simulation (based on Cosserat rod theory) and experimental tests on RoboArm. Next, path tracking in the presence/absence of obstacles as well as load carrying maneuver are investigated. Finally, the obtained results concerning repeatability, scalability, and disturbance rejection performance of the approach are discussed.

1. Introduction

In contrast to rigid-link robotic arms, continuum arms are flexible structures with the ability of continuous bending along the backbone. These arms are inspired from intrinsic capabilities of natural organisms like snakes, octopus arms, elephant trunks, and mammalian tongues. The unique features of these arms like manipulating in unstructured environments, grasping unknown objects, safe operation as well as dexterity make them a popular research area [1].

Obtaining dynamics (forward dynamics (FD) and inverse dynamics (ID)) and kinematics (forward kinematics (FK) and inverse kinematics (IK)) in continuum manipulators encounter several challenges. These challenges are originated from high nonlinearities, uncertainties, friction, shear, and so on, which limit modeling accuracy and increase computation time. Modeling can be considered as the first and most fundamental step regarding its influence in further applications, including path planning [2], optimization [3], grasping [4], and control [5]. Despite the extensive published papers in the field of kinematics modeling, only a few investigations have been dedicated to dynamics modeling of these manipulators considering high complexity, particularly in spatial motions. Most of the conducted researches were based on the development of corresponding kinematics models, through which higher errors would be generated [6]. Generally, prior investigations in modeling field can be categorized into three groups of analytical, numerical, and data-driven approaches, which would be explained in the following sections.

1.1. Analytical methods

These methods are referred to the methods, through which analytical solution can be obtained from closed-form equations by assuming some simplifying assumptions. Among the most commonly used simplifications, it can be mentioned to planar motion, constant curvature (CC), and neglection of nonlinear terms assumptions. Bamdad et al. [7] introduced an analytical procedure using modified Denavit–Hartenberg (D–H) technique for obtaining FK of continuum manipulators with piecewise CC assumption and used it for real-time control of manipulator. In another study by Neppalli [8], a closed-form analytical solution was developed for a multi-section continuum manipulator by discretizing the manipulator into several single-section manipulators. However, error propagation for multi-section case was not addressed. Falkenhahn et al. [9] used a lumped-mass and CC assumptions along with the analytical derivatives in order to establish a balance between forces and energies. Both FD and ID models were obtained; however, only planar case with neglecting torsional effects was studied. Generally, key advantage of analytical approaches is the provision of closed-form solution, which can be implemented easily in further applications, especially for control purposes. Nevertheless, problems resulted from simplifying assumptions, including poor performance, deviation from experimental results, and incapability in providing exact model can be considered as disadvantages.

1.2. Numerical methods

To solve the mentioned deficiencies and difficulties, numerical approaches have been suggested by many researchers. In a research by Chikhaoui et al. [10], FK of a tendon-driven continuum manipulator with extensible segments was determined numerically using two procedures of Cosserat rod theory and beam theory. Despite the higher accuracy of former procedure, latter one provided lower computational cost. Peyron et al. [11] obtained the kinematics model of a continuum manipulator by solving boundary value problem. In this study, the manipulator was discretized along its backbone and obtained nonlinear equations for each section were solved numerically. Besides, experimental tests were executed on a magnetic continuum manipulator for the sake of validation. However, no comparison was performed to evaluate the effectiveness of the proposed approach with other numerical approaches. In a study by Kang et al. [12], a pneumatically driven continuum manipulator was discretized to a number of serially connected parallel mechanisms and the kinematics/dynamics were derived with the help of rigid body dynamics. Habibi et al. [13] introduced a lumped-mass model for flexible surfaces subjected to large deflection as a result of actuating with two continuum manipulators. The equations of motion for discretized masses were derived through general Newtonian principle. In another study, Kim et al. [14] obtained the FK of three-section tendon-driven continuum manipulator through circular arcs assumption. The derived equations from twist method were solved numerically and validated through experimental tests. In refs. [15,16], Dehgani et al. presented a dynamic modeling approach for continuum manipulator, which used Cosserat rod theory with some realistic simplifying assumptions to reduce computational cost. In addition, a method for canceling the error of numerical integrations was introduced to enhance the computations speed. In general, main advantage of numerical approach is the provision of more exact model in which some nonlinearities and uncertainties can be considered. However, despite the improvement in accuracy, system complexity and computational cost would be increased. Hence, suitability of these approaches for online purposes would be weakened. Moreover, convergence is not guaranteed in most of the numerical approaches, so the solutions are probable to be trapped in the local solution.

1.3. Data-driven methods

Regarding the aforementioned challenges in previous approaches, recently, data-driven modeling (DDM) has been proposed. This type of modeling can be defined as establishing a relation between inputs and outputs of system without using explicit mathematical equations [17]. In a study by Li et al. [18], a DDM procedure was applied to determine a mapping from actuator to position signals of a pneumatically driven continuum manipulator. The estimation of online Jacobian was accomplished using adaptive Kalman filter with the ability of capturing instantaneous changes in Jacobian. In another research by Tan et al. [19], parameters of kinematics model were identified experimentally to cover existing uncertainties in parameters and joint inputs. However, the proposed scheme was validated only

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

for in-plane motions. Recently, Jackes et al. [20] proposed a recurrent neural network-based approach. This approach avoided the complexity of mathematical modeling, while capturing the mechanical and electrical nonlinear effects of the system. The proposed model related the actuation signals to the force measurement; however, it was validated only for a single-segment three-tendon continuum manipulator. In a study by Thuruthel et al. [6], a methodology was suggested for learning the FD of soft manipulator with the use of model-free neural network-based approach for the first time. They declared that their method provides easier implementation and improved accuracy compared to other dynamics models. It should be noted that learning inverse model in redundant manipulators often encounters some problem regarding the nonuniqueness of solutions. Several approaches have been proposed for overcoming this problem, such as distal supervised learning [21], learning in velocity level [22], path-based sampling approach (goal babbling) [23], and multiple local learning procedure [24]. Basically, the main benefit of DDM is capturing the whole system behavior which was partially discarded in previous approaches. In addition, adaptability to a wide range of manipulators, or in other words, robot-independent structure is the other potential advantage. However, requirement for higher datasets to attain an appropriate model may affect the computational time.

Inclusion of contact forces in modeling is an important issue that should be considered. Involved forces can be classified to small-valued and large-valued forces in terms of their magnitude regarding the stiffness of manipulator, inertia, velocity, duration, and location of contact [25]. In former group, small undesired contact forces can be covered with small approximation error by neural networks [26]. However, addition of second-class forces imposes higher complexity in analytical and numerical procedures; in this regard, some data-driven methods were presented [20, 25]. Thuruthel et al. [25] suggested a data-driven technique using force data obtained from single load cell embedded on the end effector of manipulator. With the help of gathered data, they established a neural network-based learning model for predicting the force.

By reviewing the available literature, lack of a general framework for obtaining kinematics and dynamics with applicability for different types/scales of continuum manipulators and capability of dealing with uncertainties, nonlinearities, and load carrying is sensed. Moreover, in most of the conducted researches, ID problem was mostly discarded considering its higher complexity. To overcome these deficiencies, in this paper, a DDM approach is proposed to consider the dynamic motion of manipulator using nonlinear autoregressive network with exogenous inputs (NARX) model.
The main contributions of the present paper can be summarized as follows:

- Proposing a general framework with maximal applicability for dynamics modeling (FD/ID) of continuum manipulators, which can be also used for kinematics modeling (FK/IK).
- Eliminating the need for solving complex equations by the use of data-driven approach.
- Capturing existing nonlinearities and uncertainties in parameters and the structure of continuum manipulator.
- Providing a simple and real-time solution with high consistency to real experimental setup in spatial motions for further applications, especially control schemes.
- Proposing a straightforward algorithm for path tracking of continuum manipulator in the presence and absence of obstacles as well as load carrying maneuver.
- Investigating the scalability of proposed procedure for manipulators with different numbers of sections.
- Implementation of proposed procedure on experimental setup of RoboArm and simulation setup based on Cosserat rod theory.

The rest of the paper is organized as follows: in Section 2, the problem is defined and the proposed procedure for kinematics/dynamics modeling of continuum manipulator in three-dimensional space is described. In the following section, the inverse problem is explored; additionally, path tracking and load carrying features are evaluated. Section 4 is dedicated to the description of studied systems: simulation setup based on Cosserat rod theory and the experimental setup of RoboArm. The results of

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

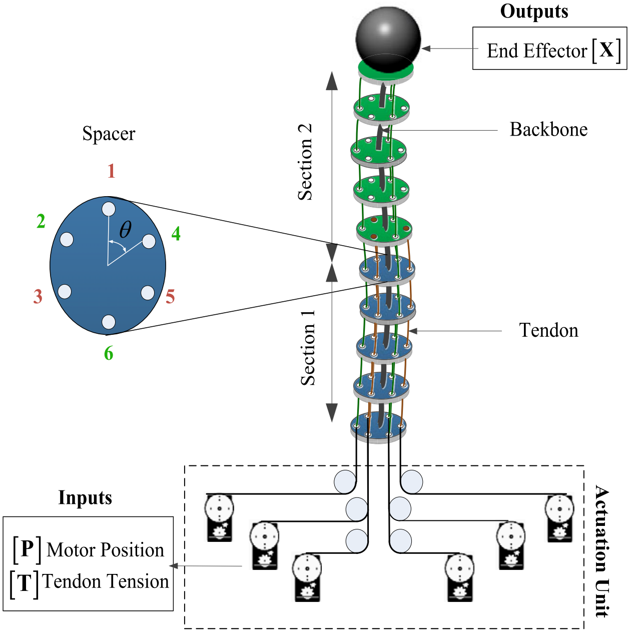

Figure 1. *Schematic of a two-section continuum manipulator consisting of a backbone, six servomotors, six tendons, and several spacers. The motor position [**P**] and tendon tension [**T**] are the system inputs and the position of end effector [**X**] is the system output.*

implementing the proposed procedure on both simulation and experimental setups are provided and discussed in Section 5. Finally, main findings of this investigation are provided in the conclusion section.

2. Problem Definition

The considered system in this study is a continuum manipulator representing a multi-input multi-output (MIMO) system with n sections, r inputs, and m outputs. The schematic of a two-section continuum manipulator is illustrated in Fig. 1. According to this figure, the system is consisted of a backbone, tendons for actuating manipulator, and spacers for keeping tendons parallel with respect to the backbone. The manipulator is a two-section system (n=2) and six servomotors (r=6) are used for positioning the end effector (m=3). The inputs are motor position/tendon tension $[\mathbf{P/T}]_{\mathbf{j}=1,2,\ldots,n}$ of j$^{\text{th}}$ section ($j \in \{1, \ldots, n\}$), and the outputs are position/orientation of end effector, [**X**]. The goal is to develop the best model (**M***) for a continuum manipulator among the candidate models as follows:

$$\mathbf{M}^* = \{\mathbf{M}(\theta) \,|\theta \in D_m\} \tag{1}$$

where **M**(θ) is the model structure, which can be ARX, NARX, and ARMAX, etc., and should be selected on the basis of an appropriate criterion. In addition, D_m is an index set of interest. To this aim, a data-driven system identification (SI) procedure is proposed.

2.1. Data-driven identification procedure for modeling

The proposed procedure for modeling continuum manipulator is depicted in Fig. 2. According to this flowchart, proposed SI contains four main steps. These steps are known as the determination of model inputs/excitation signals, selecting model structure/parameters, identification procedure, and evaluation steps, which are detailed in the following subsection (for more details, refer to refs. [27–29]).

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

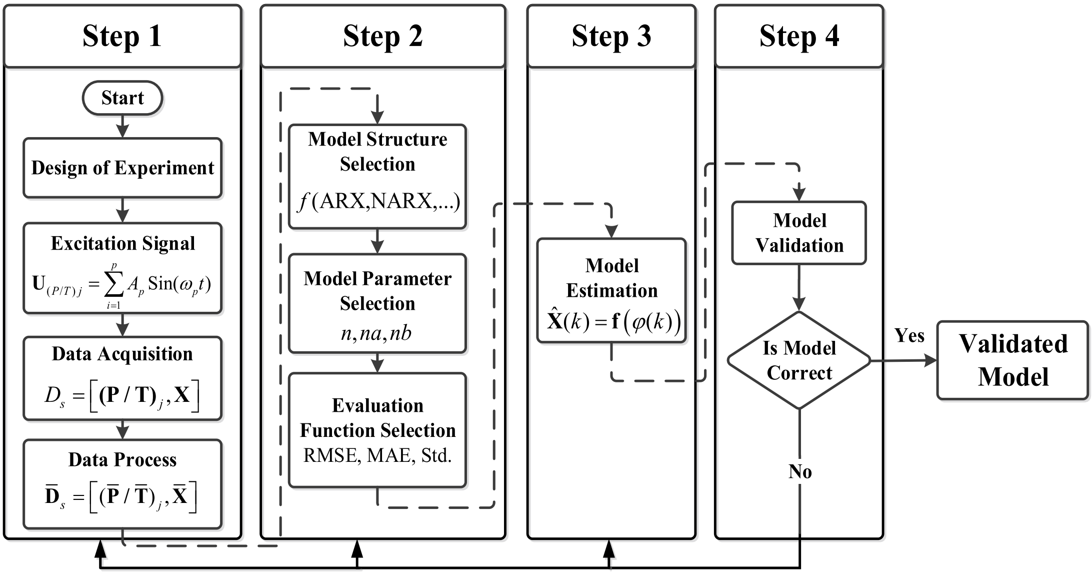
Step 1
Step 2
Step 3
Step 4
Start
Design of Experiment
Excitation Signal
$\mathbf{U}_{(P/T)j} = \sum_{i=1}^{p} A_p \,\mathrm{Sin}(\omega_p t)$
Data Acquisition
$D_s = \left[(\mathbf{P}/\mathbf{T})_j, \mathbf{X}\right]$
Data Process
$\bar{\mathbf{D}}_s = \left[(\bar{\mathbf{P}}/\bar{\mathbf{T}})_j, \bar{\mathbf{X}}\right]$
Model Structure Selection
f(ARX,NARX,...)
Model Parameter Selection
n, na, nb
Evaluation Function Selection
RMSE, MAE, Std.
Model Estimation
$\hat{\mathbf{X}}(k) = \mathbf{f}\left(\varphi(k)\right)$
Model Validation
Is Model Correct
Yes
No
Validated Model

Figure 2. *The main steps for modeling continuum manipulator through data-driven system identification procedure for obtaining a validated model.*

2.1.1. Inputs and excitation signals

Identification procedure is conducted based on processed input–output data, $\bar{\mathbf{D}}_s = [(\bar{\mathbf{P}}/\bar{\mathbf{T}})_J, \bar{\mathbf{X}}]$, from continuum manipulator. If reasonable and appropriate distribution is assumed for data, high-precision model would be obtained with higher amount of training data. The overall real operating workspace of manipulator should be covered during data acquisition process. In addition, sampling time should be considered small enough to capture small changes in model output so that

$$\text{for } x_m \in \mathbf{X}\text{:}\;\; x_m(k-i) \simeq x_m(k-i-1) \tag{2}$$

where $x_m(k-i)$ and $x_m(k-i-1)$ are two subsequent outputs and k is the time sample. For n-section manipulator with three tendons in each section, we have

$$[\mathbf{P}/\;\mathbf{T}]_{j=1,2,\ldots,n} = \begin{bmatrix} (\mathbf{P/T})_1 \\ \vdots \\ (\mathbf{P/T})_\mathbf{j} \end{bmatrix} = \begin{bmatrix} (P/T)_{1,1} & (P/T)_{1,3} & (P/T)_{1,5} \\ \vdots & \vdots & \vdots \\ (P/T)_{j,j} & (P/T)_{j,j+2} & (P/T)_{j,j+4} \end{bmatrix} \tag{3}$$

For simplicity, from now on, inputs are shown by [**T**] instead of [**P/T**]. In studied manipulator, which is a two-section, six-tendon manipulator, input vector is defined as:

$$[\mathbf{T}]_{j=1,2} = \begin{bmatrix} \mathbf{T}_1 \\ \mathbf{T}_2 \end{bmatrix} = \begin{bmatrix} T_{1,1} & T_{1,3} & T_{1,5} \\ T_{2,2} & T_{2,4} & T_{2,6} \end{bmatrix} \tag{4}$$

Additionally, outputs vector is

$$\mathbf{X} = \begin{bmatrix} x_{ee} & y_{ee} & z_{ee} \end{bmatrix} \tag{5}$$

In the above equation, x_{ee}, y_{ee} and z_{ee} are the positions of end effector along X-, Y-, and Z-axis with respect to the manipulator base, $\mathrm{O} = \{\mathrm{x}_0, \mathrm{y}_0, \mathrm{z}_0\}$. Therefore, collected datasets for n_{data} independent points are defined as $\left[\, (\mathbf{T})_{j=1,2} \quad \mathbf{X} \,\right]_{l=1,2,\ldots,n_{data}}$. Different excitation signals can be used for identifying unknown systems. According to ref. [30], model identified with sine excitation has an excellent quality in the excited

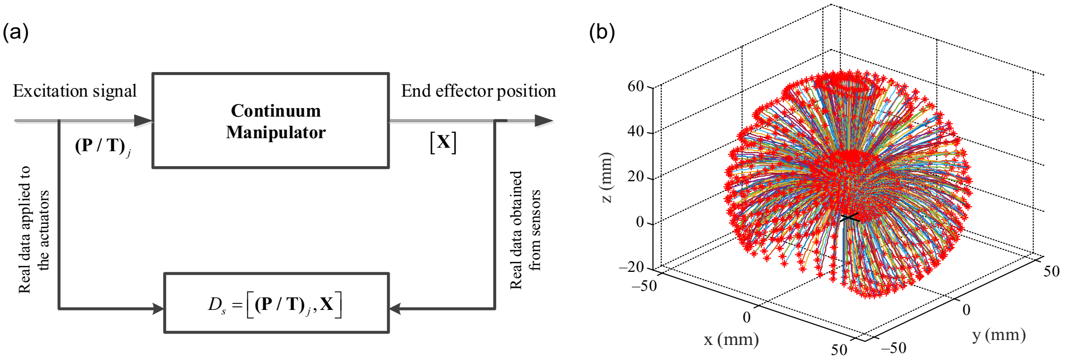
(a)
Excitation signal
Continuum Manipulator
End effector position
$(\mathbf{P}/\mathbf{T})_j$
$[\mathbf{X}]$
Real data applied to the actuators
Real data obtained from sensors
$D_s = [(\mathbf{P}/\mathbf{T})_j, \mathbf{X}]$
(b)
z (mm)
x (mm)
y (mm)
60
40
20
0
−20
−50
0
50
−50
0
50

Figure 3. (a) Data gathering procedure of required dataset for learning and (b) three-dimensional workspace of the continuum manipulator resulted from excitation signals.

frequencies. So, a combination of sine signals would be an appropriate selection for our continuum manipulator as shown in Equation (6):

$$\mathbf{U}_{(\mathbf{T})_j} = A_1\,\mathrm{Sin}(\omega_1 t) + A_2\,\mathrm{Sin}(\omega_2 t) + \ldots \quad = \sum_{i=1}^{p} A_p\,\mathrm{Sin}(\omega_p t) \tag{6}$$

where p is the number of sine signals, A_p and ω_p are the corresponding amplitude and frequency of p$^{\text{th}}$ signal, respectively. The workspace of excited manipulator besides data gathering procedure is shown in Fig. 3.

2.1.2. Model structure

Choosing an appropriate structure for model has an important role in determining the number of required parameters, convergence, and computational cost. The model can be described by a mapping $\mathbf{M}(\mathbb{R}^r \mapsto \mathbb{R}^m)$ from inputs space $(\mathbb{R}^r)$ to outputs space $(\mathbb{R}^m)$ as:

$$[\mathbf{X}(k+1)]_{m\times 1} = \mathbf{M}\begin{pmatrix} \mathbf{T}(k), \ldots, \mathbf{T}(k-na), \\ \mathbf{X}(k), \ldots \mathbf{X}(k-nb) \end{pmatrix}_{(m+r)\times 1} \tag{7}$$

In the above equation, $\begin{bmatrix}\mathbf{X}(k) & \ldots & \mathbf{X}(k-nb)\end{bmatrix}$ are current/previous outputs and $\begin{bmatrix}\mathbf{T}(k) & \ldots & \mathbf{T}(k-na)\end{bmatrix}$ represent current/previous inputs. In addition, na and nb are the respective numbers of previous inputs and outputs. Among different structures that can be used for $\mathbf{M}$, recurrent dynamic network of NARX model, which is a nonlinear expansion of ARX model, is selected considering the nonlinear and dynamic nature of the system. The ARX representation for single-input single-output (SISO) system is defined as:

$$\hat{X}(k) = b_1 T(k-1) + \ldots. + b_l T(k-na) \; - a_1 X(k-1) - \ldots - a_l X(k-nb) \tag{8}$$

where $\hat{X}(k)$ is the estimation of model output in current time regarding previous information of $T(k-na) \in \mathbb{R}$ and $x(k-nb) \in \mathbb{R}$. Also, a_l and b_l are respective constant coefficients for outputs and inputs that should be specified through identification. By substituting the known linear function by a nonlinear unknown function of $f(.)$, NARX representation can be obtained as follows:

$$\hat{X}(k) = f\,(\varphi(k)) \tag{9}$$

where $\varphi(k)$ is the regression vector and defined as:

$$\varphi(k) = [T(k-1) \ldots T(k-na)\, X(k-1) \ldots X(k-nb)]^T \tag{10}$$

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

So, Equation (9) can be rewritten as:

$$\hat{X}(k) = f\,(T(k-1), \dots, T(k-na),\ X(k-1), \dots, X(k-nb)) \tag{11}$$

By extending this formula to r-input m-output MIMO system, we have

$$\left[\hat{\mathbf{X}}(k)\right] = \mathbf{f}\begin{pmatrix} T_1(k-1), \dots, T_1(k-na), \dots, T_r(k-1), \dots, T_r(k-na), \\ X_1(k-1), \dots .X_1(k-nb), \dots, X_m(k-1), \dots, X_m(k-nb) \end{pmatrix} \tag{12}$$

where f(.) $\in \mathbb{R}^{m\times r}$ is a matrix of $m \times r$ nonlinear functions from $\mathbb{R}^r \to \mathbb{R}^m$. Equation (12), which is written in scalar form, can be represented in a vector form for discrete-time nonlinear MIMO system as:

$$\left[\hat{\mathbf{X}}(k)\right] = \mathbf{f}\,(\mathbf{T}(k-1), \dots, \mathbf{T}(k-na), \mathbf{X}(k-1), \dots .\mathbf{X}(k-nb)) \tag{13}$$

where **X** and **T** are $m \times 1$ and $r \times 1$ vectors and defined as:

$$\begin{aligned} \mathbf{X}(k) &= \begin{bmatrix} x_1(k) & \dots & x_m(k) \end{bmatrix}_{m\times 1}^{T} \\ \mathbf{T}(k) &= \begin{bmatrix} T_1(k) & \dots & T_r(k) \end{bmatrix}_{r\times 1}^{T} \end{aligned} \tag{14}$$

So, in this case, instead of approximating a_l and b_l coefficients for ARX model, nonlinear function of **f(.)** should be approximated. The NARX model can be represented in more general form as:

$$\hat{\mathbf{X}}(k) = \mathbf{f}\begin{pmatrix} \mathbf{T}(k-1),\ \mathbf{\Delta T}(k-1), \dots,\ \mathbf{\Delta}^{d_1-1}\mathbf{T}(k-1), \\ \mathbf{X}(k-1),\ \mathbf{\Delta X}(k-1), \dots,\ \mathbf{\Delta}^{d_2-1}\mathbf{X}(k-1) \end{pmatrix} \tag{15}$$

In this representation, d_1 and d_2 are the power orders of operation, and Δ is an operation and is defined as:

$$\Delta = 1 - q^{-1}, \Delta^2 = \left(1 - q^{-1}\right)^2, \dots, \Delta^d = \left(1 - q^{-1}\right)^d \tag{16}$$

For the second-order system, where $d_{1,2} = 2$, we have

$$\hat{\mathbf{X}}(k) = \mathbf{f}\begin{pmatrix} \mathbf{T}(k-1), \mathbf{T}(k-1) - \mathbf{T}(k-2), \\ \mathbf{X}(k-1), \mathbf{X}(k-1) - \mathbf{X}(k-2)) \end{pmatrix} \tag{17}$$

2.1.3. Identification procedure

NARX model is trained in series–parallel configuration by neural networks. Accordingly, inputs for a neuron are multiplied by corresponding weights, then resulted products are summed together and applied to a transfer function to produce output. This procedure can be stated as the following equation:

$$\hat{\mathbf{X}} = \sum_{f=0}^{e} \psi_f \Phi_f \left(\sum_{h=0}^{g} \psi_{fh} \mathbf{T}_f \right) \tag{18}$$

In the above equation, ψ_f and ψ_{fh} are the corresponding weights of output and hidden layers and Φ_f is the bias function. Output weights are linear functions and are defined as $\partial \mathbf{x}/\partial \psi_f = \Phi_f$, while hidden weights are nonlinear functions and updated as:

$$\psi_{fh}(k+1) = \psi_{fh}(k) - \Delta\psi_{fh}(k) = \psi_{fh}(k) - \eta_P(k)\frac{\partial J(k)}{\partial \psi_{fh}(k)} \tag{19}$$

where $\Delta\psi_{fh}(k)$ denotes the increment of $\psi_{fh}(k)$ and $\eta_P(k)$ is the learning rate. $J(k)$ is the training criteria and defined as $J(k) = \sqrt{{}^{1}\!/_{k} \sum_{k=1}^{k} (e(k))^2}$. The network should be selected by comparing the performance of different configurations with different numbers of neurons in hidden and output layers, different transfer function, and various iterations. Levenberg–Marquardt algorithm is used for training, which profits the speed convergence of Gauss–Newton algorithm besides the stability advantage of the steepest descent method.

2.1.4. Model evaluation

In this section, correctness of the model is evaluated in terms of different criteria. Firstly, validation should be conducted with fresh datasets. The quality and performance of learned model can be evaluated through error-based criteria, including, root mean square error (RMSE), mean absolute error (MAE), error mean (E_{mean}), and standard deviation (Std). These criteria are defined as:

$$RMSE = \sqrt{\left(\sum_{k=1}^{n_{data}} \left\| (\mathbf{X}(k) - \hat{\mathbf{X}}(k))^2 \right\|_2\right) / n_{data}} \quad MAE = \frac{1}{n_{data}} \sum_{k=1}^{n_{data}} \left| \mathbf{X}(k) - \hat{\mathbf{X}}(k) \right|$$

$$E_{mean} = \frac{1}{n_{data}} \sum_{k=1}^{n_{data}} \left(\mathbf{X}(k) - \hat{\mathbf{X}}(k) \right) \quad Std = \frac{1}{n_{data}} \sqrt{\sum_{k=1}^{n_{data}} (\mathbf{X}(k) - E_{mean})^2} \tag{20}$$

Training should be continued until achieving desired values which are determined according to the system dynamics, proposed application, and required accuracy. Increasing model accuracy is accompanied by higher computational cost which is not preferred in real-time applications. Hence, a balance between computational cost and accuracy should be established. Additionally, for comprehending modeling error in more realistic manner, normalized error is defined with respect to the length of manipulator (L) as:

$$e_{norm} = \left(\frac{\mathbf{X} - \hat{\mathbf{X}}}{\mathbf{X}} \right) \Big/ L \tag{21}$$

3. IK/ID and Path Tracking

The main contribution of this paper in the field of inverse modeling is to overcome the complexity of solving nonlinear equations analytically and high computational cost of numerical method using an algorithm for learning IK/ID. It is worth mentioning that real-time implementation is possible due to the low computational cost of proposed method.

3.1. Inverse kinematics/dynamics

IK/ID of continuum manipulator is defined as obtaining required [**T**] for a given [**X**] through similar data-driven approach as proposed in FK/FD. As mentioned before, learning inverse models in redundant manipulators suffer from the nonuniqueness of solution. Accordingly, first, additional constraints on tendon tensions and backbone curvature are considered. Through preliminary examination of trained model with gathered data, it is observed that nonuniqueness problem was solved. However, for more assurance, an optimization problem was defined to select desired solution by minimizing energy consumption. It is worth mentioning that since in training procedure, previous values are also considered in addition to current values of inputs and outputs, most of redundant solutions would be eliminated during initial phase of training. The required dataset for inverse problem is $\begin{bmatrix} \mathbf{X} & \mathbf{T} \end{bmatrix}$. The inverse problem can be solved through the following formulation in a vector form:

$$\hat{\mathbf{T}}(k) = \mathbf{f}' \left(\boldsymbol{\varphi}'(k) \right) \tag{22}$$

where $\mathbf{f}'$ is the nonlinear function of $[\mathbf{X}] \to [\mathbf{T}]$ and $\boldsymbol{\varphi}'(k)$ is the regression vector and given as:

$$\boldsymbol{\varphi}'(k) = \begin{pmatrix} X_1(k-1), \ldots .X_1(k-nb), \ldots, X_m(k-1), \ldots, X_m(k-nb) \\ T_1(k-1), \ldots, T_1(k-na), \ldots, T_r(k-1), \ldots, T_r(k-na), \end{pmatrix} \tag{23}$$

For studied system, which is a nonlinear three-input, six-output system, the estimation is defined as:

$$\mathbf{T}(k) = \mathbf{f}' \begin{pmatrix} X_1(k-1), \ldots, X_1(k-na), \ldots, X_3(k-1), \ldots, X_3(k-nb), \\ T_1(k-1), \ldots .T_1(k-na), \ldots, T_6(k-1), \ldots, T_6(k-na) \end{pmatrix} \tag{24}$$

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

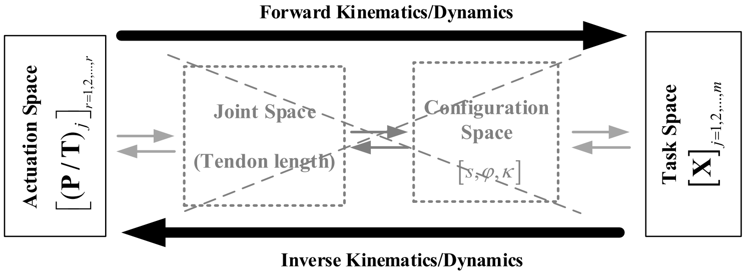
Forward Kinematics/Dynamics
Actuation Space
$\left[(\mathbf{P}/\mathbf{T})_j\right]_{r=1,2,\ldots,r}$
Joint Space
(Tendon length)
Configuration Space
$[s,\varphi,\kappa]$
Task Space
$[\mathbf{X}]_{j=1,2,\ldots,m}$
Inverse Kinematics/Dynamics

Figure 4. *The relation between actuation, joint, configuration, and task spaces for obtaining FK/IK and FD/ID of a continuum robotic arm.*

Forward/inverse mapping for modeling continuum manipulator is depicted in Fig. 4. According to this figure, this approach provides a direct mapping between actuation and task spaces with the elimination of additional mappings between actuator-to-joint, joint-to configuration, and configuration-to-task spaces. So, the computational cost can be reduced significantly. Furthermore, application of this approach is not limited to the special type of continuum manipulator and provides a general procedure; however, its applicability in various types should be examined.

3.2. Path tracking

For more evaluation of proposed approach for inverse modeling, path tracking feature in the presence/absence of obstacles is investigated. Designing a proper path can play an important role in obstacle avoidance feature of continuum robotic arm [31]. In this regard, after determining the desired path (set of $[\mathbf{X}]$), inverse modeling calculation is required for obtaining $[\mathbf{T}]_j$. The general methodology is described in the flowchart of Fig. 5.

3.2.1. Path tracking in the absence of obstacles

In the absence of obstacles, desired path parameters such as starting point $\mathrm{S_i}\left(\mathrm{x_i}\ \ \mathrm{y_i}\ \ \mathrm{z_i}\right)$, target point $\mathrm{S_f}\left(\mathrm{x_f}\ \ \mathrm{y_f}\ \ \mathrm{z_f}\right)$, midpoints, shape (circular, square. . .), and corresponding parameters (origin, radius, corners, etc.) should be imported to the algorithm to generate a path. Then, the generated path is discretized into several points (set of $[\mathbf{X}]$) and corresponding $[\mathbf{T}]_j$ is achieved through inverse procedure. After this step, an appropriate command is sent to the microcontroller and fed to the servomotors to track the desired path. Finally, the position of the end effector is collected using image processing technique.

3.2.2. Path tracking in the presence of obstacles

The problem is defined as steering the end effector from $\mathrm{S_i}\left(\mathrm{x_i}\ \ \mathrm{y_i}\ \ \mathrm{z_i}\right)$ to $\mathrm{S_f}\left(\mathrm{x_f}\ \ \mathrm{y_f}\ \ \mathrm{z_f}\right)$ in the presence of obstacles as depicted in Fig. 6. The related information of obstacles, including number of obstacles, m_{ob}, center position of obstacles, $\mathrm{S_{ob}}\left(\mathrm{x_{ob}}\ \ \mathrm{y_{ob}}\ \ \mathrm{z_{ob}}\right)$, diameter of obstacle, d_{ob} and desired safety index (ε), are determined priorly.

Then, effective obstacles are defined as those which their safety zones intersect reference (desired) path P_{des} connecting $\mathrm{S_i}\left(\mathrm{x_i}\ \ \mathrm{y_i}\ \ \mathrm{z_i}\right)$ to $\mathrm{S_f}\left(\mathrm{x_f}\ \ \mathrm{y_f}\ \ \mathrm{z_f}\right)$, see Fig. 6. After that, a perpendicular line to the desired path is plotted from the center of the effective obstacles, which intersects the safety zone at two points, $\mathrm{C}_{\mathrm{j},w}(\mathrm{j}=1,2,\ldots,2\mathrm{m_{ob}})$. Next, w points with the position of $\mathrm{S}_{\mathrm{cj,w}}\left(\mathrm{x_{cj,w}}\ \ \mathrm{y_{cj,w}}\ \ \mathrm{z_{cj,w}}\right)$ are selected from $C_{j,w}$ set. There are 2^w choices for selecting control points. Hence, the path planning problem is transformed to the planning of a polynomial path with midpoints.

Number of required conditions determines the polynomial degree. For w control points besides known start and target points, polynomial with the degree of w+1 can be fitted as:

$$P(s) = a_0 + a_1 t + a_2 t^2 + \ldots + a_{w+1} t^{w+1} \tag{25}$$

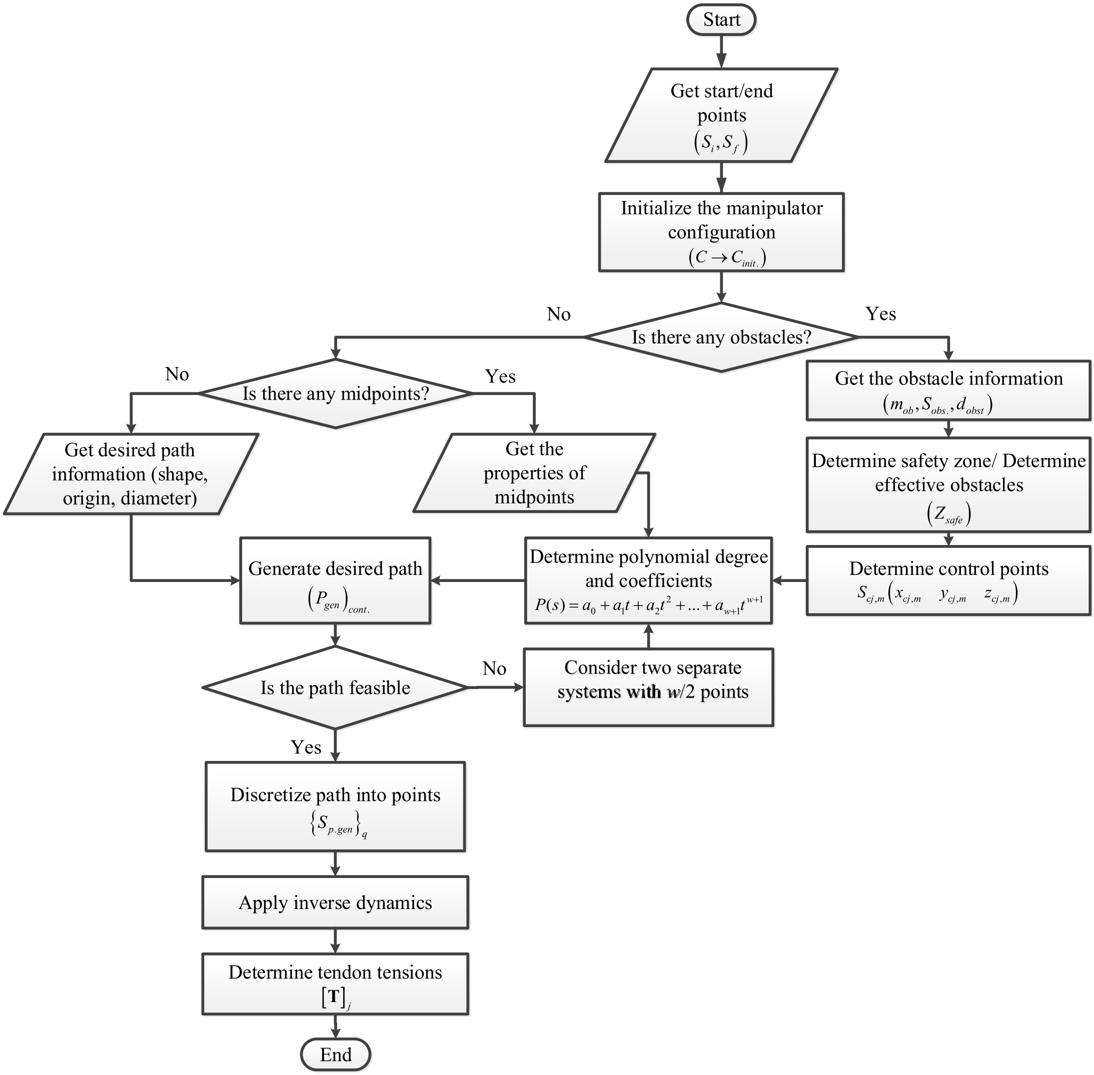
Start
Get start/end points (S_i, S_f)
Initialize the manipulator configuration $(C \rightarrow C_{init.})$
Is there any obstacles?
No
Yes
Is there any midpoints?
No
Yes
Get the obstacle information $(m_{ob}, S_{obs.}, d_{obst})$
Get desired path information (shape, origin, diameter)
Get the properties of midpoints
Determine safety zone/ Determine effective obstacles (Z_{safe})
Generate desired path $(P_{gen})_{cont.}$
Determine polynomial degree and coefficients $P(s) = a_0 + a_1 t + a_2 t^2 + ... + a_{w+1} t^{w+1}$
Determine control points $S_{cj,m}(x_{cj,m} \quad y_{cj,m} \quad z_{cj,m})$
Is the path feasible
No
Consider two separate systems with $w/2$ points
Yes
Discretize path into points $\{S_{p.gen}\}_q$
Apply inverse dynamics
Determine tendon tensions $[\mathbf{T}]_j$
End

Figure 5. *The proposed algorithm for generating the desired path in the presence/absence of obstacles and obtaining the corresponding inputs through inverse problem.*

The unknown coefficients can be determined by solving a set of linear algebraic equations. The most efficient path can be selected considering different criteria such as the length of path, energy consumption, smoothness of path, and other criteria. If the generated path is infeasible, w points should be divided into two subsections and the algorithm should be continued considering the continuity condition in mutual point. After the generation of the desired path, $(P_{gen})_{cont.}$, it is discretized into several points, $\{S_{p.gen}\}_q$, is required tendon tensions/motor positions, $[\mathbf{P/T}]_j$, and would be obtained through implementing the inverse problem.

3.3. Load carrying maneuver

According to ref. [25], force estimation can be achieved through direct and indirect sensing approaches. In former class, sensor is located in contact area. Despite the straightforward force estimation (independent of studied system) in this approach, limitations in the utilized type of sensors and their placement are the probable disadvantages. The latter approach is concerned with the estimation of

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

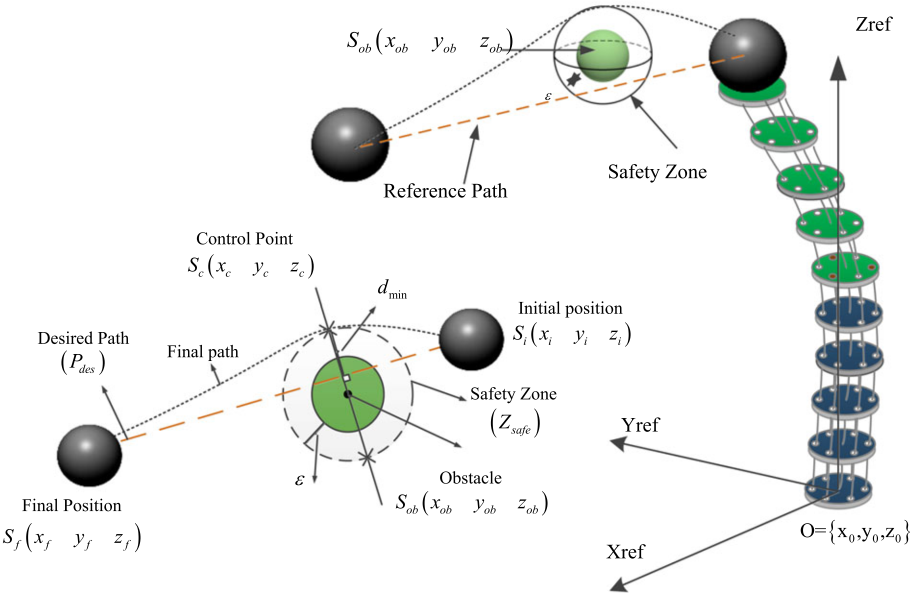

Figure 6. *Path tracking in the presence of obstacle located in $S_{ob}(x_{ob}\ y_{ob}\ z_{ob})$ from the initial point $S_i(x_i\ y_i\ z_i)$ of to the final point of $S_f(x_f\ y_f\ z_f)$.*

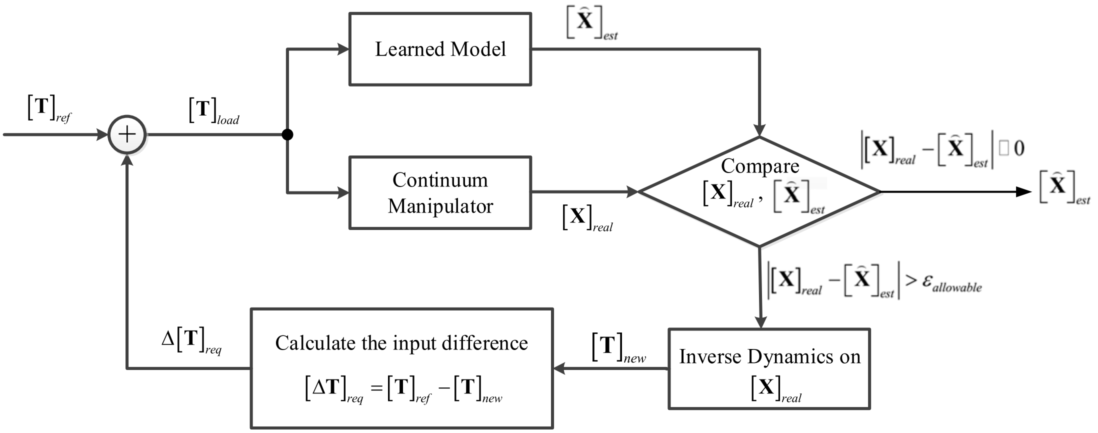
$[\mathbf{T}]_{ref}$
$[\mathbf{T}]_{load}$
Learned Model
$[\hat{\mathbf{X}}]_{est}$
Continuum Manipulator
$[\mathbf{X}]_{real}$
Compare
$[\mathbf{X}]_{real}, [\hat{\mathbf{X}}]_{est}$
$\left|[\mathbf{X}]_{real} - [\hat{\mathbf{X}}]_{est}\right| □ 0$
$[\hat{\mathbf{X}}]_{est}$
$\left|[\mathbf{X}]_{real} - [\hat{\mathbf{X}}]_{est}\right| > \varepsilon_{allowable}$
$\Delta[\mathbf{T}]_{req}$
Calculate the input difference
$[\Delta\mathbf{T}]_{req} = [\mathbf{T}]_{ref} - [\mathbf{T}]_{new}$
$[\mathbf{T}]_{new}$
Inverse Dynamics on
$[\mathbf{X}]_{real}$

Figure 7. *The proposed algorithm for detecting load carrying case for carrying unknown loads.*

force on the basis of gathered data from other sensors, which facilitates the selection of sensor type and placement. It should be noted that in the case of load carrying maneuver, different kinematics and dynamics may be achieved compared to free motion [32]. Hence, the effect of force should be involved.

In our proposed approach for detecting load carrying phase, only available sensory information, including motor positions/tendon tensions and visual feedback of end effector position, would be used. The proposed algorithm for this case can be seen in Fig. 7. According to this algorithm, real output of manipulator, $[\mathbf{X}]_{real}$, should be compared with estimated output from the learned model, $\left[\hat{\mathbf{X}}\right]_{est}$, for the same input of $[\mathbf{T}]_{ref}$. If the difference between these values exceeds the maximum allowable value ($\left|[\mathbf{X}]_{real} - \left[\hat{\mathbf{X}}\right]_{est}\right| > \varepsilon_{allowable}$), load carrying case is detected. The next step is performing the inverse problem on $[\mathbf{X}]_{real}$ and obtaining the corresponding $[\mathbf{T}]_{new}$. The difference between these inputs

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

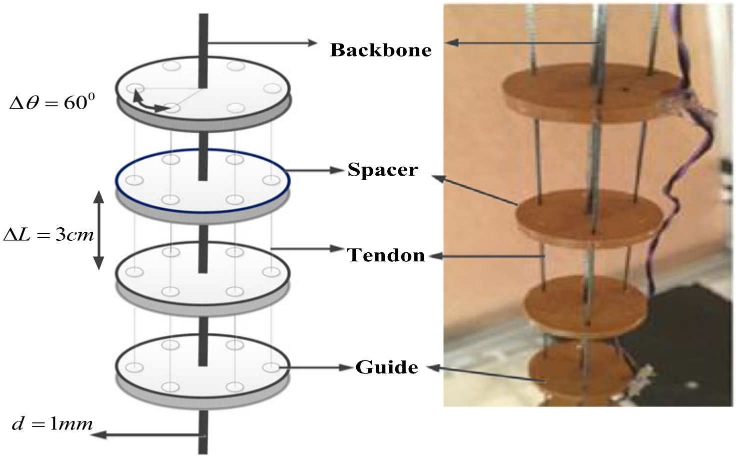

Figure 8. *Schematic and real RoboArm: experimental setup of constructed continuum manipulator with backbone, tendon, spacers, and guides.*

($[\Delta\mathbf{T}]_{req} = [\mathbf{T}]_{ref} - [\mathbf{T}]_{new}$) should be added to the unloaded input, and simulations should be continued with modified values for the loaded condition ($[\mathbf{T}]_{load}$). The $[\Delta\mathbf{T}]_{req}$ is calculated once the load carrying case is detected and added to $[\mathbf{T}]_{ref}$ in subsequent steps without performing further calculations.

The same procedure can be used for the case of interacting with an unknown rigid obstacle in which, regardless of inputs changes, $[\Delta\mathbf{T}]$, displacement of manipulator end effector would be very insignificant ($\Delta\mathbf{X} \approx \mathbf{0}$). Hence, interaction with an undefined obstacle would be identified by comparing the input/output datasets with corresponding free motion.

4. Description of Experimental and Simulation Setups

The proposed data-driven procedure for modeling would be verified on two datasets gathered from the experimental setup of RoboArm and simulation setup based on Cosserat rod model. Descriptions of these setups are provided in the following sections.

4.1. Experimental setup of RoboArm

RoboArm is a continuum manipulator that constructed in ARAS (Advanced Robotics and Automated System) center [27, 28]. This manipulator is a tendon-driven two-section arm with flexible backbone in which one end is clamped and the other end is free. Each section has three tendons that are kept parallel with respect to each other and backbone by the use of several spacers. The schematic and real RoboArm is depicted in Fig. 8. The whole constructed structure which includes RoboArm, actuation system, and sensing system is represented in Fig. 9, and the associated properties are also provided in Table I. Moreover, schematic of system components and their connection are illustrated in Fig. 10. In this structure, tensions are applied by rolling up the tendons by servomotors. The tendons pass through guiding pulleys and load cells so that their tensions can be measured. It is important to keep tendon tensions in safe range. Higher tensions would lead to snap and generation of higher friction, while slack resulted from lower tensions may lead to invalid measurement [20]. The prior problem is solved by setting limits in actuation; besides, low-stiffness springs are embedded before the motors to generate an insignificant pre-tensioning in tendons. Data are collected from two systems: image processing system and load cells. Two A4-tech PK7 cameras in appropriate adjustable distances are implemented to capture the three-dimensional position of RoboArm.

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

Table I. *Characteristics of RoboArm, actuation, and sensing systems.*

Characteristic	**Value**	**Characteristic**	**Value**
Robot type	Tendon-driven	Section length	295 mm
Type of actuation	Externally	Total length	590 mm
Number of sections	2	Backbone diameter	1 mm
Radial distance of tendons	120 degrees	Motor type	Dynamixel AX-12
Number of tendons in each section	3	Distance between spacers	30 mm
Total degrees of freedom (actuation)	6	Backbone Elasticity Modulus	203 Gpa
Dimensions of actuation box	500 × 500 × 300mm	Camera type	A4Tech PK7

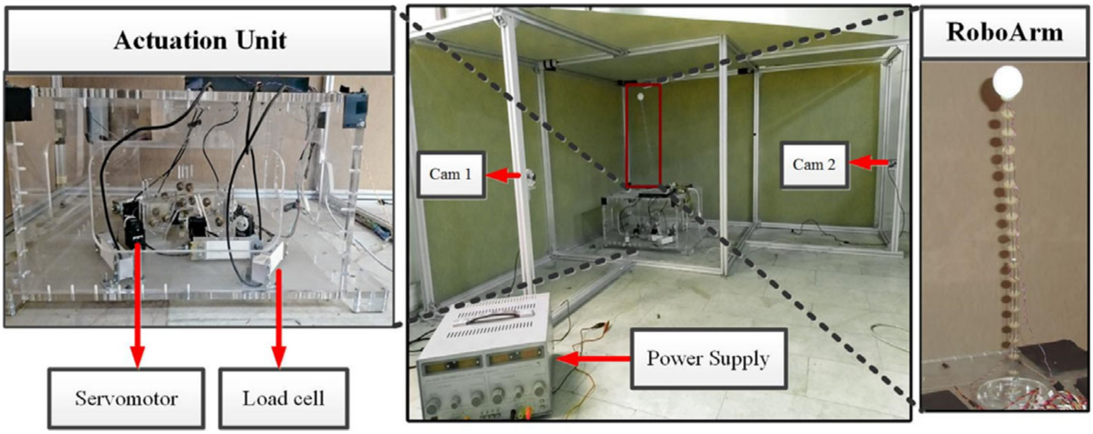

Figure 9. *The real setup of whole structure, including RoboArm, actuation system, power supply, and sensing system along with the location of calibration LEDs.*

4.1.1. Setup initialization

The home position is defined as the straight configuration of manipulator along the Z-axis from zero reference. In this configuration, no tension (except minimum insignificant pre-tension for slack prevention) is generated in tendons and the positions of servo motors are set at 500. After initializing the manipulator, the calibration of image processing system is conducted. As can be seen in Fig. 9, four LEDs are implemented on image processing box (LEDs #1, 2, 3, and 4) and three LEDs (#5, 6, and 7) are located on actuation box; in addition, one LED is also positioned on the end effector. The LEDs are turned on one by one, and images are captured by two cameras and compared to correct the possible deviations. This procedure is conducted before each run. In order to minimize the noise-induced errors in data, some filters, including reducing exposure, RGB to B&W, reducing brightness, and increasing contrast, are implemented in the internal setting of cameras to reduce the computational cost of image processing. The other processing works, including box blur, finding object center, rotation correction, lens distortion correction, and pixel to mm transformation, are done by the proposed image processing algorithm.

4.1.2. Data gathering

The samples are gathered by generating excitation signal in the computer and transferring to PIC 32 microcontroller by RS 232 cable. After that, the signal is sent to servomotors to produce the motion. The

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

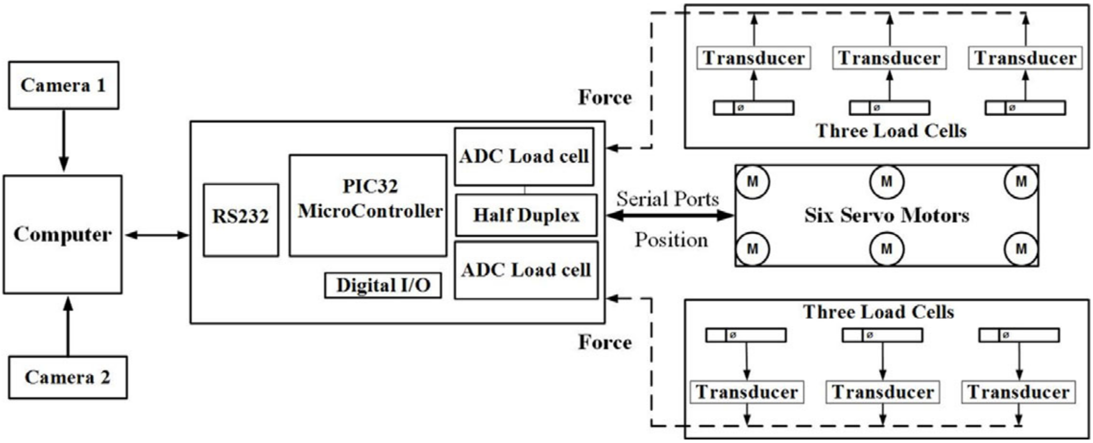

Figure 10. *Schematic of system specifications: cameras, servomotors, load cells, and their connection.*

motion of end effector is then recorded by image processing system. Hence, $\left[\mathbf{T} \quad \mathbf{X}\right]_{l=1,2,\ldots,n_{data}}$ dataset is gathered. Totally, 6000 samples are collected; 90% (5400 samples) is employed for training and the remaining (10%; 600 samples) is used for testing and validating. The samples are collected at certain frequencies during continuous motion of manipulator. It is worth mentioning that no waiting time is considered between motions and data gathering is conducted in dynamic manner. The sampling time is considered to be 0.03 s regarding the limitations in image processing system (30 fps).

4.1.3. Repeatability of system

Repeatability can be expressed as the ability of robot to reach a specified point successively with an acceptable closeness [16]. Different factors, including nonlinearity of structure, environmental conditions, gear backlash, calibration and precision of measuring components, can affect repeatability. In this paper, system repeatability is crucial for the reliability of data acquisition procedure according to the requirement for numerous data which may be collected in several runs and different conditions. For this purpose, the system was subjected to five similar inputs in different conditions (continuous run, run after a day, run after stopping the system, run after system shut down, and run after several runs with different inputs). Then, the obtained results for the position of end effector were compared and repeatability of system was confirmed.

4.2. Simulation setup based on Cosserat rod theory

In order to gather required data for training a model through proposed data-driven procedure, a quasi-static model developed by Dehghani et al. [16, 33] on the basis of Cosserat rod theory is explained. By neglecting shear deformations and extensibility, this method provides a suitable approach for numerical calculations. Free body diagram of a continuum manipulator is depicted in Fig. 11. Each point along the rod is specified by s with Cartesian position of $r(s) = \left[x(s) \quad y(s) \quad z(s)\right]^T$. It is assumed that the point force of F and the torque of τ are applied on end effector. Additionally, $f(s)$ and $\psi(s)$ represent the distributed forces and torques, respectively. The force and moment balances for an element of manipulator can be written as:

$$\begin{cases} n(b) - n(a) + \int_a^b f(s)ds = 0 \\ m(b) - m(a) + r(b) \times n(b) - r(a) \times n(a) + \int_a^b (r(s) \times f(s) + \psi(s))ds = 0 \end{cases} \tag{26}$$

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

In the above equation, n and m are the summation of forces and moments, respectively. By using Hooke's law for pre-curved manipulator, m in local coordinate system can be defined as:

$$m^l = \begin{bmatrix} EI_{xx} & 0 & 0 \\ 0 & EI_{yy} & 0 \\ 0 & 0 & GJ \end{bmatrix} \left(\Omega^l - \Omega^{*l}\right) = C\left(\Omega^l - \Omega^{*l}\right) \tag{27}$$

where Ω is the orientation variation and Ω^{*l} is the undeformed rate of rotation along the rod. Additionally, I_{xx} and I_{yy} denote second moments of area and J represents polar second moment of area. Moreover, E and G are elastic and shear modulus, respectively. For uniform rod assumption, C is a constant matrix. In this paper, gravitational force is considered as distributed force, $f(s) = \rho Ag$. After solving Equations (26), replacing resultant derivatives, and performing some operations, mathematical model of elastic rod can be represented as follows:

$$\begin{aligned} &\text{(a) } n = \begin{bmatrix} F_x & F_y & \rho Ags - \rho AgL + F_z \end{bmatrix}^T \\ &(b) \begin{cases} \Omega^{l'} = -C^{-1}\left[\Omega^L\right]^{\times} C\left(\Omega^L - \Omega^{*L}\right) - C^{-1}R^T\left(-\left[n^{\times}\right] \times R\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T + \psi\right) + \Omega^{*l'} \\ R' = R\left[\Omega^l\right]^{\times} \\ r' = R\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \end{cases} \end{aligned} \tag{28}$$

where R is the rotation matrix represented as $R(s) = \begin{bmatrix} i(s) & j(s) & k(s) \end{bmatrix}$, where $i(s)$, $j(s)$ and $k(s)$ are the unit vectors as depicted in Fig. 11. In order to solve Equation (28), numerical methods with appropriate boundary conditions should be considered. Since the robot is fixed at its base, therefore position and orientation are known at $s = 0$. Besides, force and moments are also known at the tip of manipulator ($s = s_f$). Hence, the required boundary conditions are defined as follows:

$$\begin{cases} \vec{\mathbf{r}}(0) = \vec{\mathbf{r}}_0 \\ \mathbf{R}(0) = \mathbf{R_0} \\ \vec{\mathbf{\Omega}}^l(s_f) = \vec{\mathbf{\Omega}}^{*l} + \mathbf{C}^{-1}\mathbf{R}^T(s_f)\vec{\boldsymbol{\tau}} \end{cases} \tag{29}$$

The common approach for solving these equations is shooting method, which starts with the selection of an initial estimation for $\hat{\Omega}^l(0)$. The procedure continues by comparing $e_{bv} = \hat{\Omega}^l(s_f) - \Omega^l(s_f)$ error; as illustrated in Fig. 12. The above-mentioned equation sets are the general form of Cosserat model for continuum manipulators with preferred actuation. For a tendon-driven manipulator, equations should be modified. After establishing force and moment balances for an element with three tendons as depicted in Fig. 13, overall mathematical model of tendon-driven manipulator would be obtained as:

$$\begin{aligned} &\text{(a) } n = \begin{bmatrix} F_x & F_y & \rho Ags - \rho AgL + F_z \end{bmatrix}^T - \sum T_i^l \\ &(b) \begin{cases} \Omega^{l'} = -C^{-1}\left[\Omega^L\right]^{\times} C\left(\Omega^L - \Omega^{*L}\right) - C^{-1}\left(-\left[n^l\right]^{\times} r'^{\,l} + \psi\right) + \Omega^{*l'} - \ldots \\ \ldots - C^{-1}\left[\Omega^L\right]^{\times} \sum \left[\rho_i^l\right]^{\times} T_i^l - C^{-1}\tfrac{d}{ds}\left(-\left[\rho_i^l\right]^{\times} T_i^l \psi\right) \\ R' = R\left[\Omega^l\right]^{\times} \\ r' = R\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \end{cases} \end{aligned} \tag{30}$$

In the above equations, T_i is the i^th tendon tension. By solving Equation (30) along with the boundary conditions in Equation (29), relation between tendon tensions and position of end effector can be realized. For testing the proposed data-driven procedure, the excitation signal is imported as input and the output would be collected.

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

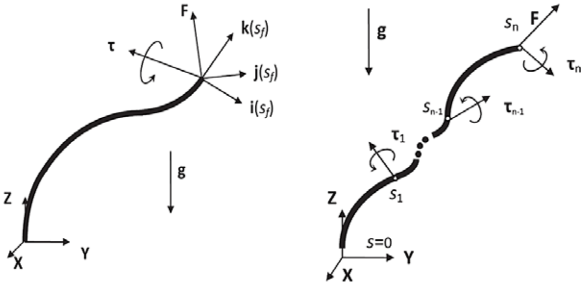

__Figure 11.__ Free body diagram of the continuum robotic arm introduced in ref. [33].

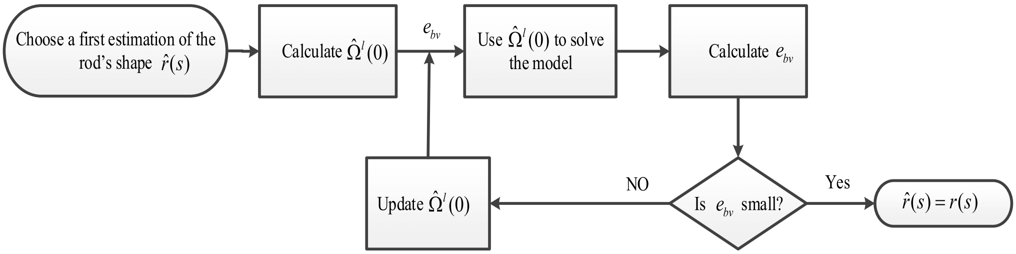
Choose a first estimation of the rod's shape $\hat{r}(s)$
Calculate $\hat{\Omega}^l(0)$
e_{bv}
Use $\hat{\Omega}^l(0)$ to solve the model
Calculate e_{bv}
Is e_{bv} small?
NO
Yes
Update $\hat{\Omega}^l(0)$
$\hat{r}(s)=r(s)$

__Figure 12.__ Flowchart of used numerical procedure for obtaining the mathematical model of the continuum arm.

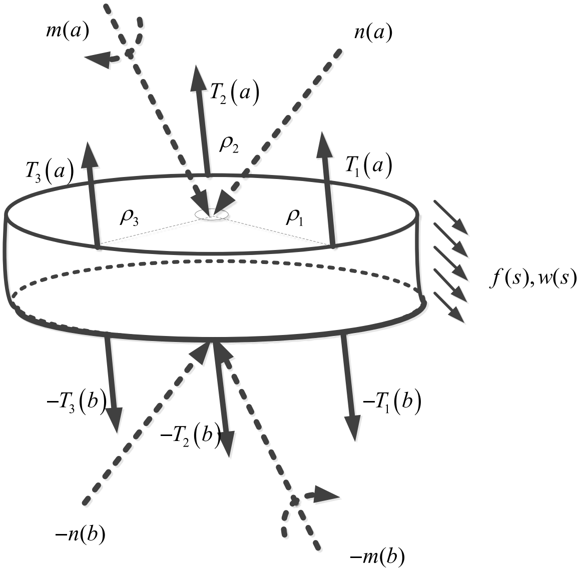

__Figure 13.__ Free body diagram for a slice of tendon-driven continuum manipulator.

5. Results and Discussions

This section provides the results of implementing proposed data-driven approach on the aforementioned experimental and simulation setups as well as investigating other features of algorithm, including scalability, disturbance rejection, and load carrying maneuver.

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

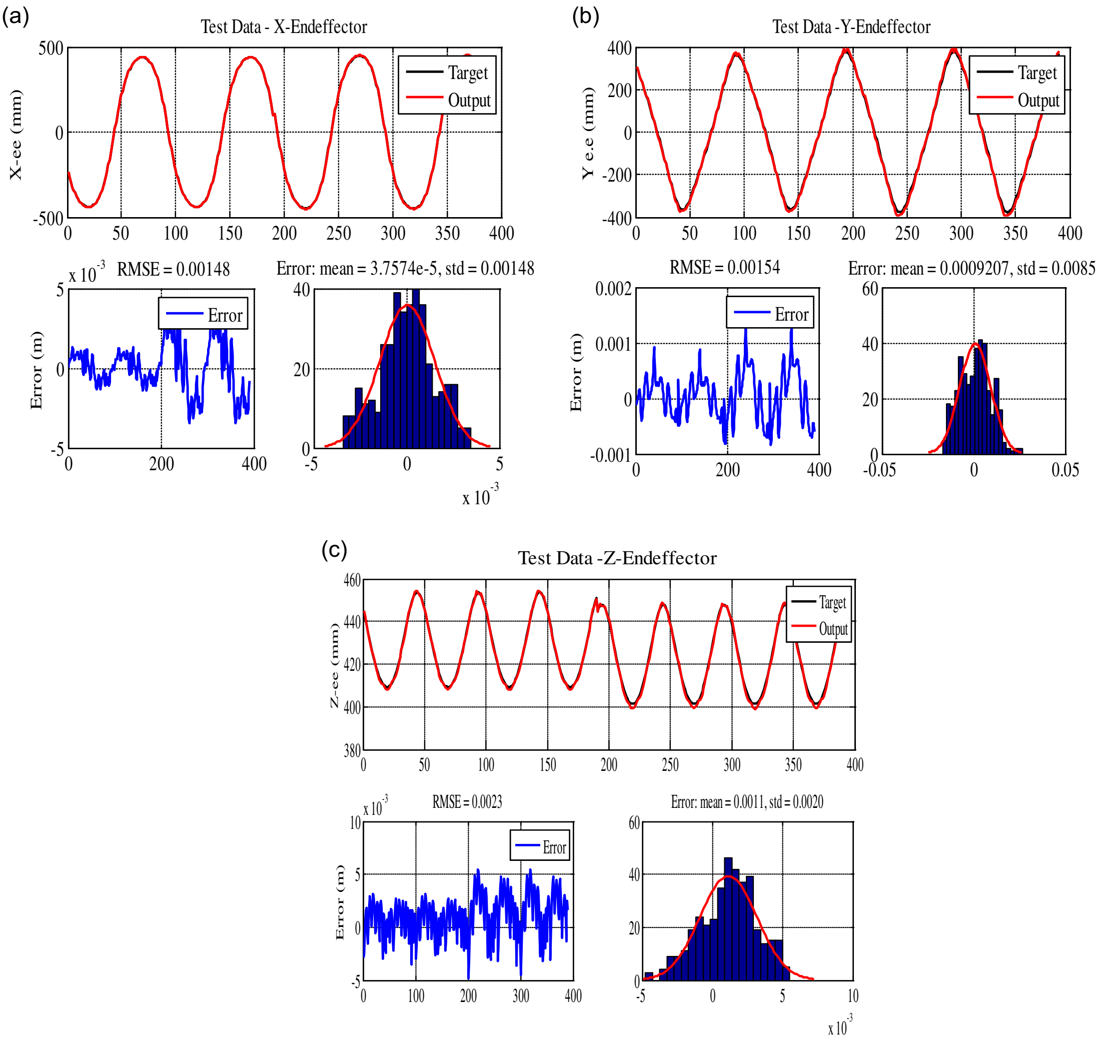

Figure 14. *FK results for simulation setup based on Cosserat rod theory; first diagram provides a comparison of estimated and real values, second diagram represents the prediction error, and third diagram depicts error mean and Std for (a) X-e.e, (b) Y-e.e, and (c) Z-e.e.*

5.1. Results of proposed data-driven approach for simulation setup based on Cosserat rod theory

5.1.1. FK modeling

The results of implementing the proposed data-driven approach on the data obtained from Cosserat rod simulation setup were investigated in this section. The comparison of real results from simulation setup and estimated results from learned model is provided in Fig. 14. According to this figure for FK, acceptable estimation accuracy can be perceived which confirmed the appropriateness of proposed method. In the first diagram of this figure, black line indicates real results obtained from simulation setup and red line represents estimated values through proposed DDM approach. In the second diagram, the deviation of estimated and real values was provided. This deviation can be characterized in terms of RMSE which was very small in all directions (0.00148, 0.00154, and 0.00230 m for x, y, and z positions of end effector). In the third diagram, error mean and standard deviation can be seen. The distribution of error close to zero revealed the accuracy of estimation. In addition, MAE values for the end effector

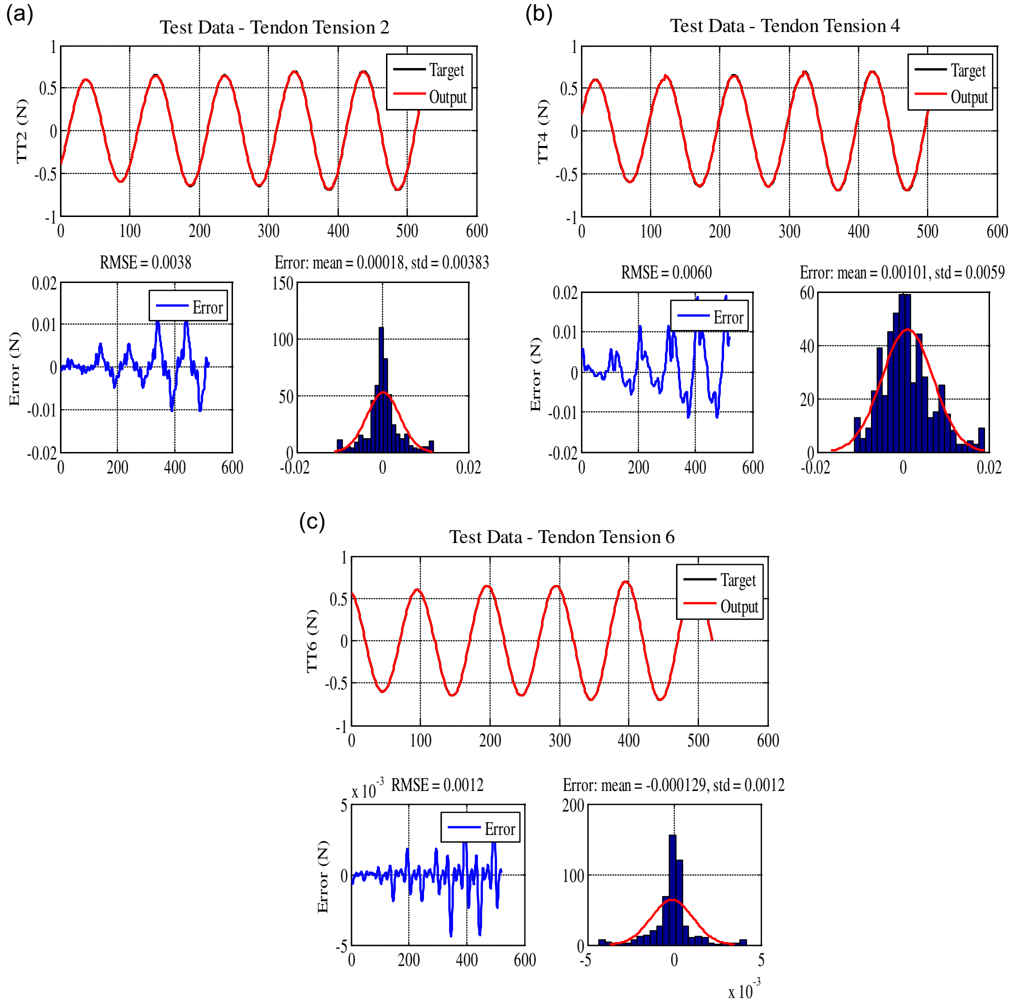

Figure 15. *IK results for simulation setup based on Cosserat rod theory; first diagram provides a comparison of estimated and real values, second diagram represents prediction error, and third diagram depicts error mean and Std for (a) second tendon, (b) fourth tendon, and (c) sixth tendon.*

position were 5.4, 4.1, and 3.0 mm, corresponding to the normalized error of 0.91, 0.69, and 0.50%. Consequently, appropriate performance of the proposed model in obtaining FK can be realized.

5.1.2. IK modeling

As mentioned, obtaining analytical solution for IK of nonlinear and redundant manipulators is very complex and high deviation from the real results was probable. Furthermore, in numerical approaches, high computational time and trapping in local solution were the challenges that should be overcome. Data-driven approaches also suffer from solution redundancy. The advantages of proposed procedure in this paper are simplicity, low computational cost, and provision of unique solution. Results of IK through proposed data-driven scheme are presented in Fig. 15. For the sake of brevity, results are provided only for second, fourth and sixth tendons. As can be seen, there was a good agreement between real and estimated values. The maximum errors in tensions of second, fourth, and sixth tendons were equal to 0.012, 0.018, and 0.004 N, respectively. Therefore, it can be concluded that IK of manipulator was fully captured with an insignificant error. Hence, appropriateness of proposed model for obtaining IK can be inferred.

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

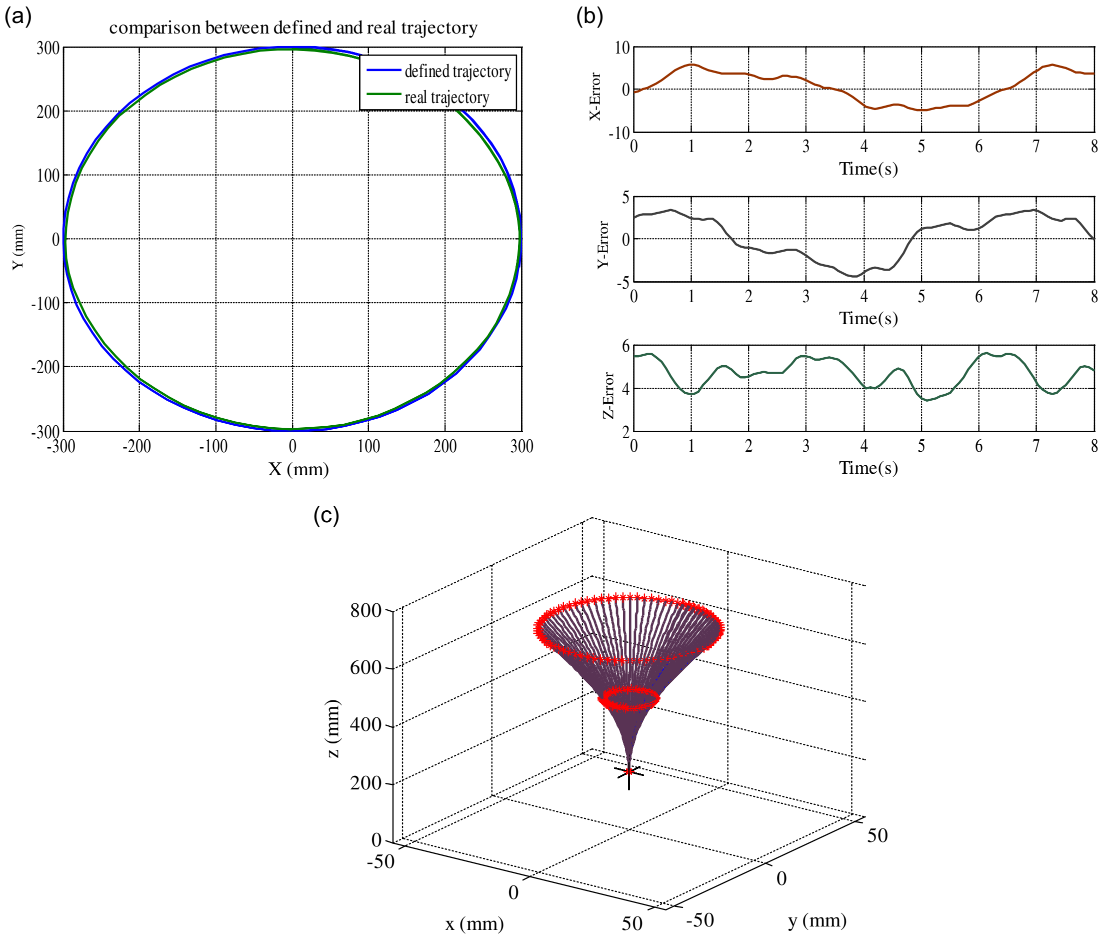

Figure 16. Tracking circular path by simulation setup: (a) defined and real path by proposed model, (b) path tracking error in (mm), and (c) configuration of manipulator during tracking circular path.

5.1.3. Path tracking

First, tracking a circular path with a center of $\mathbf{O} = \begin{bmatrix} 0 & 0 & 500 \end{bmatrix}$mm and radius of 300 mm in the absence of obstacles was evaluated. After determining desired path as depicted by blue line in Fig. 16(a), it was discretized into 100 points and required tendon tensions were specified through IK. Tensions were imported as inputs to simulation setup, and real path was generated as green line. The errors between these two paths are displayed in Fig. 16 (b). Accordingly, the capability of manipulator in tracking defined path with insignificant normalized errors of 1.1, 0.5, and 0. 9% in X, Y, and Z directions can be realized. The configuration of manipulator during path tracking is also represented in Fig. 16 (c). In the second step, the planning of an appropriate path from initial position of $S_i(12 \quad 17 \quad 500)$mm to final position of $S_f(82 \quad 85 \quad 500)$ mm in the presence of a 10-mm diameter obstacle in $S_{ob}(45 \quad 50 \quad 500)$mm was studied. Width of safety zone was assumed as 12 mm. The control point was obtained as $S_c(42 \quad 72 \quad 500)$mm. Hence, a polynomial was fitted to initial, control, and final points as depicted by blue line in Fig. 17, and discretized into 100 points. The corresponding tendon tensions were obtained through IK and then applied to model to generate resultant path as green line. According to this figure, good path tracking feature can be comprehended.

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

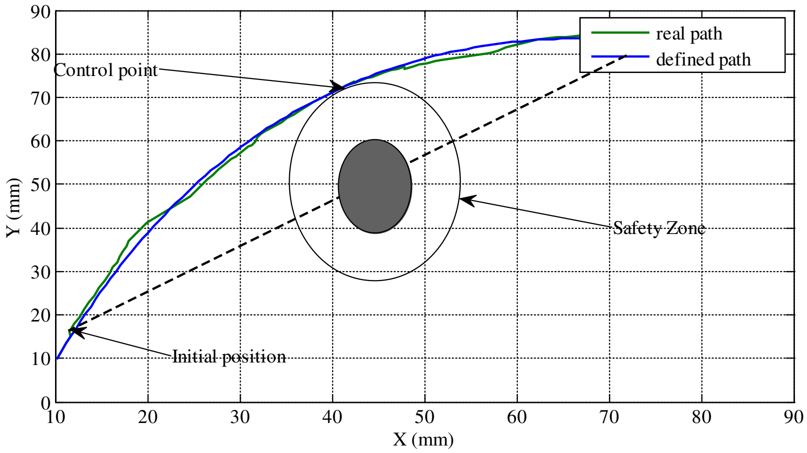

Figure 17. *Real and defined path of end effector in the presence of a circular obstacle in* S_{ob}*(45 50 500)mm.*

5.2. Results of proposed data-driven approach for experimental setup of RoboArm

5.2.1. FD modeling

Experimental validation of proposed procedure for FD was performed using data gathered from RoboArm. The provided results in Fig. 18 reveal that the model captured system dynamics with an acceptable accuracy. The maximum error values were 22, 21, and 18 mm corresponding to normalized errors of 3.72, 3.55, and 3.04%. Comparison of results for implementing proposed data-driven approach for FD of experimental setup and FK of simulation setup is provided in Table II. Higher estimation errors in the experimental setup were attributed to dynamic motion of RoboArm. The consequent unwanted and uncontrolled vibrations resulted from slender structure of RoboArm affected the richness of collected data. To enhance the quality of collected data, compensation of vibrations through appropriate controllers was suggested. This model provided best fitted model since it was learned with real data of RoboArm considering all effective parameters that were usually ignored in conventional methods.

5.2.2. ID modeling

Results for ID estimation of second, fourth, and sixth tendons are presented in Fig. 19. As it is obvious, proposed ID model for experimental setup of RoboArm provided an acceptable performance. The existing small deviations between real and estimated values were attributed to the intrinsic vibrations of RoboArm during dynamic motions. According to Fig. 19, maximum error in motor position was 50 units (approximately 1%). It should be noted that regarding the hysteresis nature of torque control in these servomotors, proposed algorithm was implemented on motors position instead of tendons tension.

5.2.3. Path tracking

In this section, path tracking feature of RoboArm was evaluated for similar circular path as surveyed in Section 5.1.3. The obtained result for tracking desired path by RoboArm is depicted in Fig. 20 (a). Additionally, deviations between defined and real paths are also provided in Fig. 20 (b). According to this figure, maximum normalized positioning errors in X, Y, and Z directions were 8.3, 7.1, and 1.5 %, respectively, with corresponding MAE values of 5.1, 4.1, and 1%. Furthermore, the comparison of path tracking feature for simulation and experimental setups is provided in Table III. The long and flexible structure besides sudden movements and rapid commands (by external or internal forces) in continuum manipulators disposed them to both positioning errors and undesirable vibrations. These errors are more prominent in dynamic motions owing to dominancy of friction and hysteresis [6, 34].

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

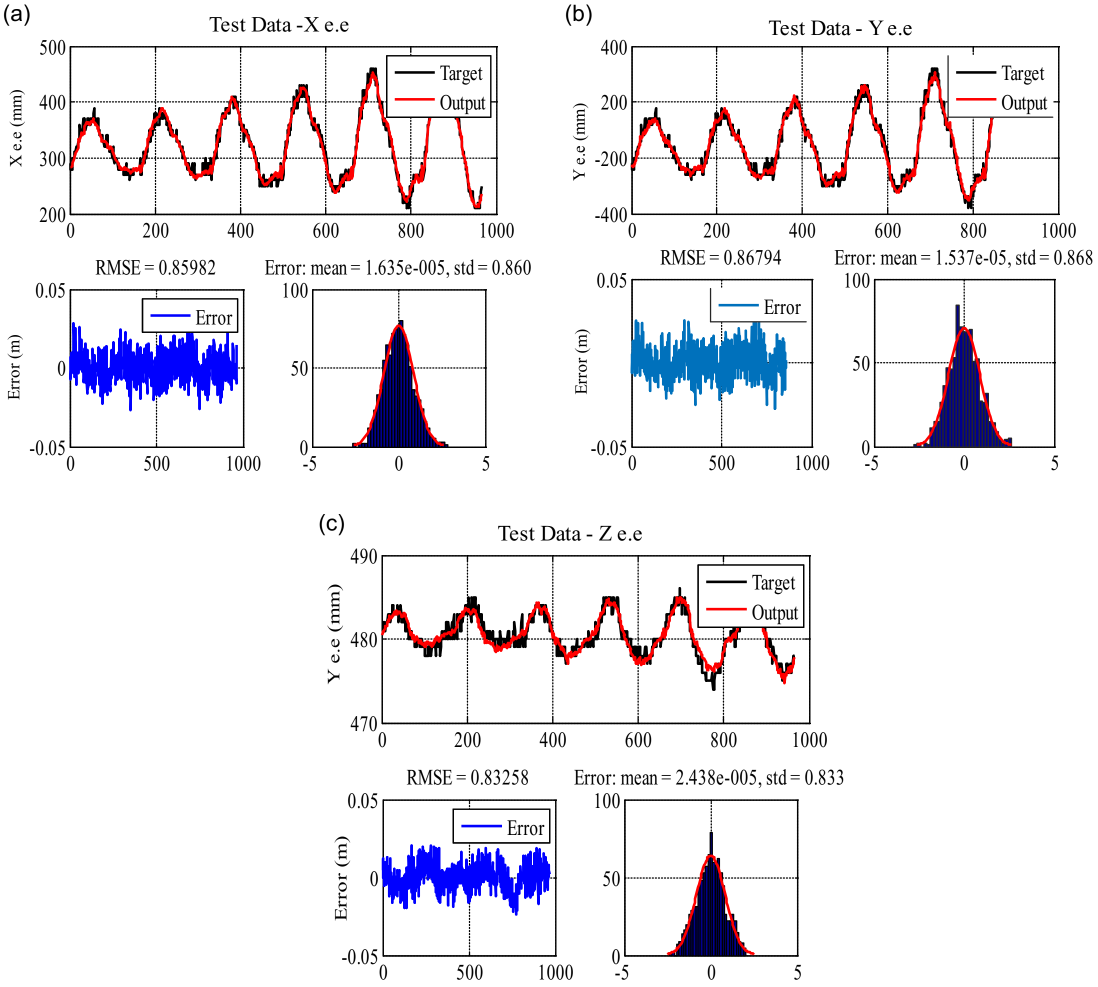

Figure 18. *FD for experimental setup of RoboArm; first diagram provides a comparison of estimated and real values, second diagram represents prediction error, and third diagram depicts error mean and Std for (a) X-e.e, (b) Y-e.e, and (c) Z-e.e.*

Occurrence of undesired oscillatory motions is common in the fast point-to-point tracking which should be suppressed to enhance the performance and accuracy [35, 36]. Accordingly, long motion cycles, quasi-static motion, slow trajectories, and canceling vibrations have been proposed [37]. In ref. [38], a controller was proposed to handle both path tracking and vibration suppression. In our study, further improvement in path tracking feature can be achieved using an appropriate controller for undesired vibrations.

It should be mentioned that as these manipulators are mainly used in medical operations, vibration would not be problematic due to the damping of vibration in ttissue. Moreover, no closed-loop controller was implemented in our study and the errors in previous steps may affect the subsequent points. Hence, by a suitable closed-loop controller, error may be improved. For better understanding, the circular trajectory of this work (radius of 300 mm) was compared with the circular trajectory of ref. [39] (radius of 63 m). The tracking error in the latter case was 22.5 mm for open-loop state, which was reduced to 2.8 mm with closed-loop control scheme. The corresponding open-loop error in our study was equal to 50 mm (corresponding to normalized error of 8.3%), which is claimed to be improved with closed-loop controller. It is worth mentioning that in continuous path tracking, regarding the discretization of path

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

Table II. *Comparison of modeling result for FK of simulation setup and FD of experimental setup.*

	Simulation model (FK)	**Experimental model (FD)**
Max error x (m)	4×10^{-3} (0.67%)	0.22×10^{-1} (3.72%)
Max error y (m)	1.8×10^{-3} (0.30%)	0.21×10^{-1} (3.55%)
Max error z (m)	5.2×10^{-3} (0.88%)	0.18×10^{-1} (3.04%)
Error mean x (m)	0.04×10^{-3}	0.016×10^{-3}
Error mean y (m)	0.92×10^{-3}	0.015×10^{-3}
Error mean z (m)	1.11×10^{-3}	0.024×10^{-3}
RMSE x (m)	1.48×10^{-3}	8.59×10^{-1}
RMSE y (m)	1.54×10^{-3}	8.67×10^{-1}
RMSE z (m)	2.30×10^{-3}	8.32×10^{-1}

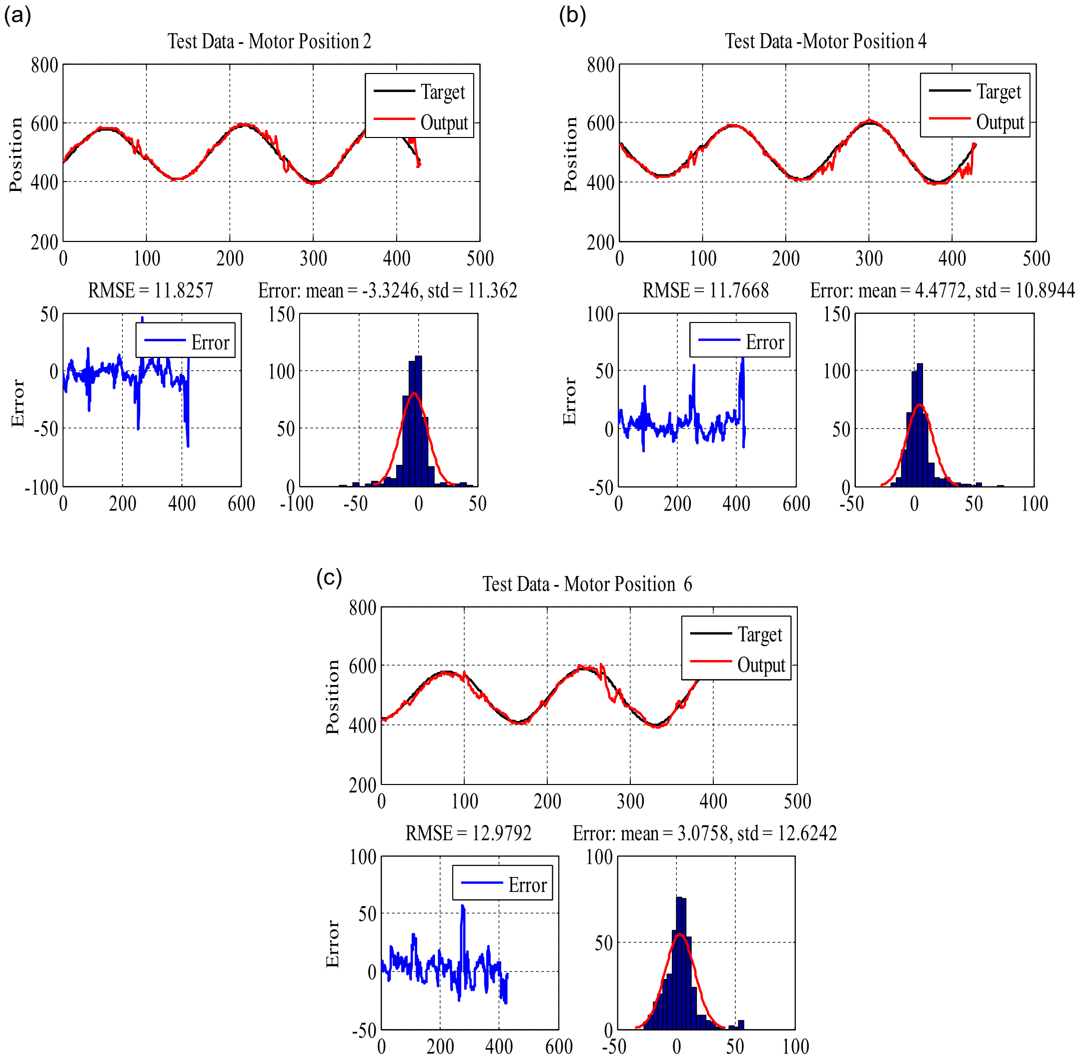

Figure 19. *ID results for experimental setup of RoboArm; first diagram provides a comparison of estimated and real values, second diagram represents prediction error, and third diagram depicts error mean and Std for RoboArm (a) first tendon, (b) fourth tendon, and (c) sixth tendon.*

Table III. *Error comparison of path tracking in both simulation and experimental setups.*

		Simulation model	**Experimental model**
Path tracking error	Max error x (m)	6.6×10^{-3} (1.10%)	50.0×10^{-3} (8.30%)
	Max error y (m)	4.2×10^{-3} (0.50%)	43.0×10^{-3} (7.10%)
	Max error z (m)	5.4×10^{-3} (0.90%)	9.0×10^{-3} (1.50 %)
	MAE x (m)	4.0×10^{-3} (0.66%)	31.0×10^{-3} (5.10%)
	MAE y (m)	3.2×10^{-3} (0.53%)	25.0×10^{-3} (4.10 %)
	MAE z (m)	3.0×10^{-3} (0.50%)	6×10^{-3} (1.00%)

Table IV. *Comparison of end effector error in different references for dynamics modeling.*

Ref	**Type**	**Length (m)**	**Tip error (m)**	**Tip error (%)**
[6]	Cable-driven	0.203	$(0.70 - 1.80) \times 10^{-2}$	$3.44^{*}-8.66^{**}$
[41]	Cable-driven	0.418	3.10×10^{-2}	7.30%
[42]	Cable-driven	0.311	$\approx 1.50 \times 10^{-2}$	≈5.00%
[43]	Pneumatic	1.000	5.00×10^{-2}	5.00%
Our study (simulation setup)	Tendon-driven	0.590	0.52×10^{-2}	0.88%
Our study (RoboArm)	Tendon-driven	0.590	2.2×10^{-2}	3.72%

*For the environment with higher viscosity (water) in simulation

**For the environment with lower viscosity (air) in simulation

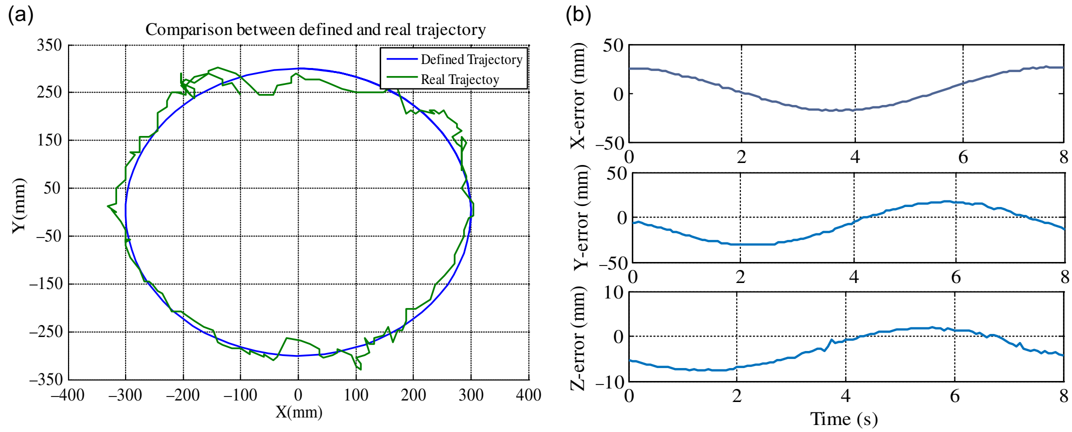

Figure 20. *Tracking circular path by experimental setup of RoboArm: (a) defined and obtained path by proposed approach and (b) tracking error of circular path.*

into several points and considering the update of target in each step, higher errors would be generated compared to single-point tracking case [40].

5.3. Error Discussion

The comparison of errors for dynamic model of the continuum manipulator with other available references is provided in Table IV. According to this table, a very good accuracy of the proposed procedure compared to other studies involving dynamic modeling can be inferred. As mentioned in ref. [2], even more improved models can be achieved by increasing the training time in expense of increased computational cost.

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

Table V. *Comparison of FK results in the presence and absence of external disturbances.*

		With disturbance	**Without disturbance**
Max error	Max error-X (m)	21.0×10^{-3} (3.55%)	4×10^{-3} (0.67%)
	Max error-Y (m)	13.0×10^{-3}(2.19%)	1.8×10^{-3} (0.30%)
	Max error-Z (m)	10.0×10^{-3}(1.69%)	5.2×10^{-3} (0.88%)
Error mean	Error mean-X (m)	-0.87×10^{-3}	0.037×10^{-3}
	Error mean-Y (m)	0.44×10^{-3}	0.92×10^{-3}
	Error mean-Z (m)	1.21×10^{-3}	1.11×10^{-3}
RMSE	RMSE-X (m)	12.9×10^{-3}	1.48×10^{-3}
	RMSE -Y (m)	3.9×10^{-3}	1.54×10^{-3}
	RMSE -Z (m)	7.7×10^{-3}	2.32×10^{-3}

Table VI. *Comparison of FK model for single, two, and three-section continuum manipulator.*

		MAE (m)	**RMSE (m)**	**Error mean (m)**
One-section continuum manipulator	X-e.e	2.00×10^{-3} (0.33%)	0.33×10^{-3}	-0.29×10^{-3}
	Y-e.e	0.90×10^{-3}(0.15%)	0.11×10^{-3}	0.26×10^{-3}
	Z-e.e	1.70×10^{-3}(0.28%)	0.24×10^{-3}	1.01×10^{-3}
Two-section continuum manipulator	X-e.e	5.40×10^{-3}(0.91%)	1.48×10^{-3}	0.04×10^{-3}
	Y-e.e	4.10×10^{-3}(0.69%)	1.54×10^{-3}	0.92×10^{-3}
	Z-e.e	3.00×10^{-3}(0.50%)	2.30×10^{-3}	1.11×10^{-3}
Three-section continuum manipulator	X-e.e	15.40×10^{-3}(2.50%)	18.40×10^{-3}	1.63×10^{-3}
	Y-e.e	14.70×10^{-3}(2.40%)	17.80×10^{-3}	1.20×10^{-3}
	Z-e.e	10.70×10^{-3}(1.78%)	12.50×10^{-3}	5.90×10^{-3}

5.4. *Performance in the presence of disturbances*

The applicability of proposed procedure in the presence of disturbances was examined in this section. In addition to the disturbances caused by environmental effects, small-valued contact forces can be also considered as disturbances. To evaluate this feature, a random disturbance (ratio of maximum disturbance amplitude to maximum input amplitude was equal to 11%) was added to the input data. The provided results in Table V confirmed good estimation in the presence of disturbance. As can be seen, the maximum errors in X, Y, and Z directions were equal to 8.9×10^{-3}, 10.2×10^{-3} and 9.1×10^{-3}m, corresponding to normalized errors of 1.40, 1.70, and 1.51% with respect to the manipulator length. Hence, it can be concluded that external disturbances can be compensated well by the proposed procedure, which was expected regarding disturbance rejection feature of neural network structure.

5.5. *Scalability*

For generalizing the proposed procedure for manipulators with different number of sections, several factors should be evaluated. According to ref. [10], error is generally amplified with increasing the manipulator's length regarding friction and coupling effects between sections. For data-driven procedure, despite the strengthened effects by length increase, no significant error was expected according to the inclusion of these effects in gathered data.

The procedure was implemented on single-section, two-section, and three-section manipulators in simulation setup based on Cosserat rod theory. However, for experimental validation, a new setup should be constructed. The provided results in Table VI disclosed better results for single-section manipulator. For the two-section manipulator, the prediction error of end effector position was about 2.72 times of the corresponding value for single-section counterpart. For the three-section manipulator, the mentioned value was approximately 3.2 times of two-section counterpart. Accordingly, scalability of manipulator

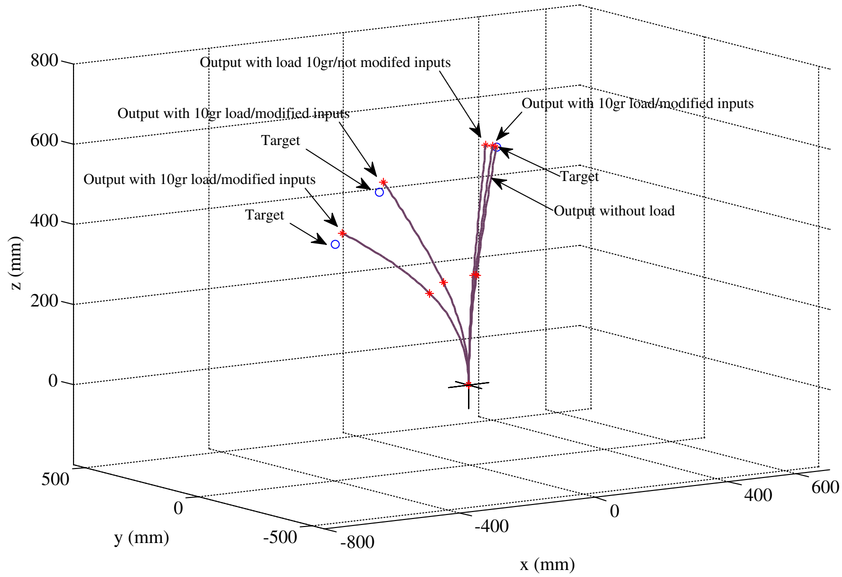

Figure 21. *Performance of manipulator in unloaded and loaded case (addition of 10 g mass).*

can be inferred. In this design, since the diameters of all sections were considered to be equal, only the scalability of manipulator in terms of length was provided; however, the scalability was also assessed for different diameters and almost similar results were obtained.

5.6. Load carrying maneuver

This section was dedicated to performance evaluation of proposed algorithm for load carrying maneuver. Accordingly, outputs (estimated) and target values with load are depicted in Fig. 21 for some cases. In the first case, output and target values of the end effector in unloaded free motion were shown, which were almost coincident (normalized error of 0.01% with respect to manipulator length). Then, a mass of 10 g was added and load carrying case was recognized by available sensory information. After that, required input for compensating load was calculated and added to reference input. For the second case, $S_1(380\quad -180\quad 585)$mm, estimated values of $\hat{S}_1(361 -169\quad 561)$mm were obtained in loaded condition, which were corresponding to normalized errors of 3.21, 1.86, and 4.07 %, respectively. Moreover, estimated values by algorithm for reaching $S_1(-150\quad 200\quad 410)$ and $S_1(-250\quad 100\quad 290)$ points revealed 2.9, 3.2, 3.7 and 3.2, 3.5, 3.8 % normalized error, respectively. For further compensation of applied load, additional sensory information and appropriate controllers are required.

6. Conclusions

In this paper, the development of kinematics/dynamics models through data-driven identification algorithm was addressed. Through the proposed procedure, a direct mapping was established between the actuation and task spaces. Therefore, complexity and computational costs were reduced by eliminating the additional actuator-to-configuration and configuration-to-task mappings. The proposed algorithm was implemented on gathered data from two setups: simulation setup based on Cosserat rod theory and experimental setup of RoboArm for validating its performance. According to the obtained results of prior setup, superior estimation of data-driven FK was realized in which maximum normalized errors for X, Y, and Z components of end effector were equal to 0.67, 0.30, and 0.88 %, respectively. The corresponding results for FD estimation of experimental setup were the respective values of 3.72, 3.55, and 3.04%. Higher estimation errors in latter setup were attributed to the consideration of dynamic

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

motion instead of quasi-static motion in the simulation setup. Accordingly, unwanted and uncontrolled vibrations affected the richness of collected data. To overcome this problem, compensation of vibrations through an appropriate controller was suggested to enhance the quality of collected data. By addressing the existing challenges in developing the inverse models for continuum manipulators, a data-driven procedure was proposed for inverse modeling problem, which was successfully implemented on simulation and experimental setups.

The simplicity and confidence in obtaining FK/FD and IK/ID as well as accuracy of obtained models are the advantageous features of proposed data-driven procedure. Capturing every details of real system, including uncertainties, friction, backlash, and so on, is the other probable advantage which influences the further applications, especially path planning and control schemes. Then, using the obtained model, path tracking in the presence/absence of obstacles was examined on both setups. The maximum normalized errors for X, Y, and Z components of end effector were 1.1, 0.5, and 0.9% for simulation setup and 8.3, 7.1, and 1.5% for experimental setup, respectively. Additionally, disturbance rejection was evaluated, and well compensation of disturbances by proposed scheme was comprehended. Furthermore, the generalization of procedure for continuum arms with lower and higher number of sections was studied in simulation. Despite the increased errors in higher number of sections originated from friction and coupling effects, the developed model still provided good estimation. Moreover, the performance of proposed algorithm in load carrying maneuver was assessed, and an acceptable prediction with the use of only available sensory information was achieved.

References

1. W. Saab, W. S. Rone, and P. Ben-Tzvi, “Robotic tails: A state-of-the-art review,” *Robotica.* **36**(9), 1263–1277 (2018).
2. J. Li and J. Xiao, “An efficient algorithm for real time collision detection involving a continuum manipulator with multiple uniform-curvature sections,” *Robotica.* **34**(7), 1566–1586 (2016).
3. K. Xu, J. Zhao, and X. Zheng, “Configuration comparison among kinematically optimized continuum manipulators for robotic surgeries through a single access port,” *Robotica.* **33**(10), 2025–2044 (2015).
4. D. Braganza, D. M. Dawson, I. D. Walker, and N. Nath, “Neural network grasping controller for continuum robots,” *Proceedings of the 45th IEEE Conference on Decision and Control*, (2006) pp. 6445–6449.
5. S. Cobos-Guzman, D. Palmer, and D. Axinte, “Kinematic model to control the end-effector of a continuum robot for multi-axis processing,” *Robotica.* **35**(1), 224–240 (2017).
6. T. G. Thuruthel, E. Falotico, F. Renda, and C. Laschi, “Learning dynamic models for open loop predictive control of soft robotic manipulators,” *Bioinspir. Biomim.* **12**(6), 66003 (2017).
7. M. Bamdad and M. M. Bahri, “Kinematics and manipulability analysis of a highly articulated soft robotic manipulator,” *Robotica.* **37**(5), 868–882 (2019).
8. S. Neppalli, M. A. Csencsits, B. A. Jones, and I. D. Walker, “Closed-form inverse kinematics for continuum manipulators,” *Adv. Robot.* **23**(15), 2077–2091 (2009).
9. V. Falkenhahn, T. Mahl, A. Hildebrandt, R. Neumann, and O. Sawodny, “Dynamic modeling of bellows-actuated continuum robots using the Euler–Lagrange formalism,” *IEEE Trans. Robot.* **31**(6), 1483–1496 (2015).
10. M. T. Chikhaoui, S. Lilge, S. Kleinschmidt, and J. Burgner-Kahrs, “Comparison of modeling approaches for a tendon actuated continuum robot with three extensible segments,” *IEEE Robot. Autom. Lett.* **4**(2), 989–996 (2019).
11. Q. Peyron, Q. Boehler, K. Rabenorosoa, B. J. Nelson, P. Renaud, and N. Andreff, “Kinematic analysis of magnetic continuum robots using continuation method and bifurcation analysis,” *IEEE Robot. Autom. Lett.* **3**(4), 3646–3653 (2018).
12. R. Kang, D. T. Branson, T. Zheng, E. Guglielmino, and D. G. Caldwell, “Design, modeling and control of a pneumatically actuated manipulator inspired by biological continuum structures,” *Bioinspir. Biomim.* **8**(3), 36008 (2013).
13. H. Habibi, C. Yang, I. S. Godage, R. Kang, I. D. Walker, and D. T. Branson, “A lumped-mass model for large deformation continuum surfaces actuated by continuum robotic arms,” *J. Mech. Robot.* **12**(1), 011014 (2020).
14. Y. Kim, S. S. Cheng, M. Diakite, R. P. Gullapalli, J. M. Simard, and J. P. Desai, “Toward the development of a flexible mesoscale MRI-compatible neurosurgical continuum robot,” *IEEE Trans. Robot.* **33**(6), 1386–1397 (2017).
15. M. Dehghani and S. A. A. Moosavian, “Compact modeling of spatial continuum robotic arms towards real-time control,” *Adv. Robot.* **28**(1), 15–26 (2014).
16. M. Dehghani and S. A. A. Moosavian, “Modeling and control of a planar continuum robot,” *Advanced Intelligent Mechatronics (AIM), 2011 IEEE/ASME International Conference on* (2011) pp. 966–971.
17. D. Solomatine, L. M. See, and R. J. Abrahart, “Data-driven modelling: concepts, approaches and experiences,” In: *Practical hydroinformatics*, edited by R. J. Abrahart (Springer-Verlag Berlin Heidelberg, Germany, 2009) pp. 17–30.
18. M. Li, R. Kang, D. T. Branson, and J. S. Dai, “Model-free control for continuum robots based on an adaptive Kalman filter,” *IEEE/ASME Trans. Mechatronics.* **23**(1), 286–297 (2017).

https://doi.org/10.1017/S026357472100093X Published online by Cambridge University Press

19. N. Tan, X. Gu, and H. Ren, "Pose characterization and analysis of soft continuum robots with modeling uncertainties based on interval arithmetic," *IEEE Trans. Autom. Sci. Eng.* **16**(2), 570–584 (2018).
20. D. Jakes, Z. Ge, and L. Wu, "Model-less Active Compliance for Continuum Robots using Recurrent Neural Networks," *arXiv Prepr. arXiv1902.08943* (2019).
21. A. Melingui, R. Merzouki, J. B. Mbede, C. Escande, B. Daachi, and N. Benoudjit, "Qualitative approach for inverse kinematic modeling of a compact bionic handling assistant trunk," in *Neural Networks (IJCNN), 2014 International Joint Conference on* (2014) pp. 754–761.
22. A. D'Souza, S. Vijayakumar, and S. Schaal, "Learning inverse kinematics," *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)* (2001) pp. 298–303.
23. M. Rolf and J. J. Steil, "Efficient exploratory learning of inverse kinematics on a bionic elephant trunk," *IEEE Trans. Neural Networks Learn. Syst.* **25**(6), 1147–1160 (2013).
24. L. Vannucci, E. Falotico, N. Di Lecce, P. Dario, and C. Laschi, "Integrating feedback and predictive control in a bio-inspired model of visual pursuit implemented on a humanoid robot," *Conference on Biomimetic and Biohybrid Systems* (2015) pp. 256–267.
25. T. G. Thuruthel, B. Shih, C. Laschi, and M. T. Tolley, "Soft robot perception using embedded soft sensors and recurrent neural networks," *Sci. Robot.* **4**(26), 1–9 (2019).
26. R. Grassmann, V. Modes, and J. Burgner-Kahrs, "Learning the forward and inverse kinematics of a 6-DOF concentric tube continuum robot in SE (3)," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2018) pp. 5125–5132.
27. A. Parvaresh, S. A. Moosavi, and S. A. A. Moosavian, "Identification and Position Control of a Continuum Robotic Arm," *2017 5th RSI International Conference on Robotics and Mechatronics (ICRoM)* (2017) pp. 310–315.
28. A. Parvaresh and S. A. A. Moosavian, "Modeling and Model-free Fuzzy Control of a Continuum Robotic Arm," *2018 6th RSI International Conference on Robotics and Mechatronics (IcRoM)* (2018) pp. 501–506.
29. A. Parvaresh and S. A. A. Moosavian, "Experimental identification and hybrid PID-fuzzy position control of continuum robotic arms," *Int. J. Robot. Theory Appl.* **6**(1), 53–63 (2020).
30. O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks, Fuzzy Models, and Gaussian Processes* (Springer-Verlag Berlin Heidelberg GmbH, Germany, 2020).
31. B. Ouyang, Y. Liu, H.-Y. Tam, and D. Sun, "Design of an interactive control system for a multisection continuum robot," *IEEE/ASME Trans. Mechatronics.* **23**(5), 2379–2389 (2018).
32. F. Alambeigi, Z. Wang, Y.-H. Liu, R. H. Taylor, and M. Armand, "A versatile data-driven framework for model-independent control of continuum manipulators interacting with obstructed environments with unknown geometry and stiffness," *arXiv Prepr. arXiv2005.01951* (2020).
33. M. Dehghani, "Dynamics modeling and control for continuum robot," K. N. Toosi University of Technology (2014).
34. I. A. Gravagne, C. D. Rahn, and I. D. Walker, "Large deflection dynamics and control for planar continuum robots," *IEEE/ASME Trans. Mechatronics.* **8**(2), 299–307 (2003).
35. T. G. Thuruthel, M. Manti, E. Falotico, M. Cianchetti, and C. Laschi, "Induced Vibrations of Soft Robotic Manipulators for Controller Design and Stiffness Estimation," *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)* (2018) pp. 550–555.
36. T. Yamaguchi et al., "A mechanical approach to suppress the oscillation of a long continuum robot flying with water jets," *IEEE Robot. Autom. Lett.* **4**(4), 4346–4353 (2019).
37. G. Gao, L. Li, H. Wang, and Y. Du, "Study on Vibration Analyze and Control of Continuum Robot," *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)* (2018) pp. 675–679.
38. R. S. Penning and M. R. Zinn, "A combined modal-joint space control approach for continuum manipulators," *Adv. Robot.* **28**(16), 1091–1108 (2014).
39. M. Li, R. Kang, S. Geng, and E. Guglielmino, "Design and control of a tendon-driven continuum robot," *Trans. Inst. Meas. Control.* **40**(11), 3263–3272 (2018).
40. T. George Thuruthel, E. Falotico, M. Manti, A. Pratesi, M. Cianchetti, and C. Laschi, "Learning closed loop kinematic controllers for continuum manipulators in unstructured environments," *Soft Robot.* **4**(3), 285–296 (2017).
41. F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, "Dynamic model of a multibending soft robot arm driven by cables," *IEEE Trans. Robot.* **30**(5), 1109–1122 (2014).
42. F. Renda, F. Boyer, J. Dias, and L. Seneviratne, "Discrete cosserat approach for multisection soft manipulator dynamics," *IEEE Trans. Robot.* **34**(6), 1518–1533 (2018).
43. D. Trivedi, A. Lotfi, and C. D. Rahn, "Geometrically exact dynamic models for soft robotic manipulators," *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2007) pp. 1497–1502.

Cite this article: A. Parvaresh and S. A. A. Moosavian (2022). "Dynamics and path tracking of continuum robotic arms using data-driven identification tools", Robotica **40**, 1098–1124. https://doi.org/10.1017/S026357472100093X