Mechanism and Machine Theory 201 (2024) 105736

Contents lists available at ScienceDirect

Mechanism and Machine Theory

journal homepage: www.elsevier.com/locate/mechmt

Research paper

An adaptive lumped-mass dynamic model and its control application for continuum robots

Xu Zhang [a], Chenghao Yang [a,b], Zhibin Song [a], Mojtaba A. Khanesar [c], David T Branson [c], Jian S. Dai [a,d], Rongjie Kang [a,*]

[a] *Key Laboratory of Mechanism Theory and Equipment Design of the Ministry of Education, Centre for Advanced Mechanisms and Robotics, Tianjin University, PR China*
[b] *School of Mechanical Engineering, Hebei University of Technology, PR China*
[c] *Advanced Manufacturing Technology Research Group, Faculty of Engineering, University of Nottingham, UK*
[d] *Shenzhen Key Laboratory of Intelligent Robotics and Flexible Manufacturing Systems, Southern University of Science and Technology, Shenzhen, PR China*

ARTICLE INFO

Keywords:
Continuum robot
Lumped-mass model
Dynamics
Parameter estimation

ABSTRACT

Dynamic modeling for continuum robots remains challenging due to their large nonlinear deformation and the variation of dynamic parameters during movement. In this paper, a lumped-mass dynamic model (LMD) for a continuum robot is constructed including elastic and viscous parameters in the robotic joints. Then the appropriate dynamic parameters (e.g. spring and damping coefficients of the LMD) with respect to the motion status (e.g. position and velocity of the robot) are estimated using a Genetic Algorithm (GA). Based on the obtained data set, a Multi-Layer Perception (MLP) is trained to establish a direct mapping from the motion status to the dynamic parameters, so the LMD can tune its parameters in real-time when moving within the workspace, resulting an adaptive lumped-mass dynamic model (ALMD). Compared to the fixed-parameter LMD, the modeling error of the ALMD is reduced by up to 60.2 %. Finally, a feed-forward controller is implemented to control a continuum robotic prototype using the presented ALMD, reducing the maximum tracking error by 67.5 %.

1. Introduction

Compared with traditional discrete rigid-joint robots, continuum robots have the advantages of high dexterity [1]. Although there have been many kinds of materials and actuators used in continuum robots, the body structures of this kind of robots are basically the same, generally consisting of several serially stacked modules. Each of these modules usually contains three or four longitudinal actuators arranged in parallel [2,3]. By tuning the length of those actuators, the module can bend or even elongate. The more modules included in a continuum robot, the higher dexterity the robot can achieve. In these systems, the actuators and supporting structure (e.g., the backbone) are usually made of flexible materials, such as silicone [3-6], NiTi alloy [7–11], spring [12,13] and so on. Due to the elasticity in materials and the rheonomic constraints between structures (e.g., the driving rod and the constraint disk), the system has strong nonlinearity and uncertainty [14,15], making dynamic modeling of the continuum robot very challenging.

There has been some previous work establishing dynamic models for continuum robots. Some researchers developed the dynamic

* Corresponding author.
E-mail address: rjkang@tju.edu.cn (R. Kang).

https://doi.org/10.1016/j.mechmachtheory.2024.105736
Received 28 December 2023; Received in revised form 28 May 2024; Accepted 8 July 2024
Available online 14 July 2024

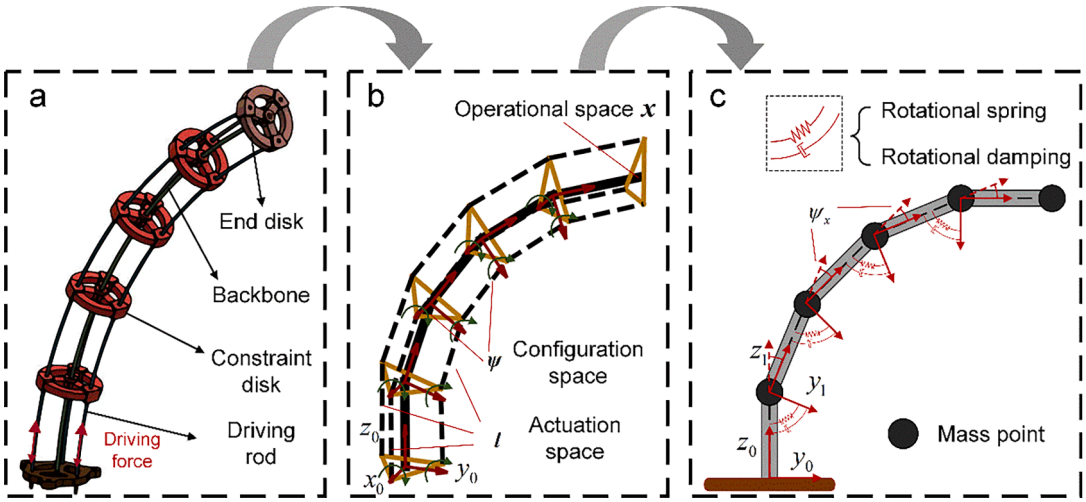

Fig. 1. Modeling of a continuum robot: (a) the structure of the robot, (b) the kinematic model, (c) the LMD.

model by using rod theories in combination with specific geometry assumptions. Godage et al. considered the continuum robot as a flexible rod with piecewise equal curvature, and established the dynamic model based on spatial integration [16,17]. Renda et al. considered that the deformation of the continuum robot is uniform along the axis of the arm, and modeled it based on the Cosserat rod theory. The dynamic model can accurately describe the deformation and torsional deformation [18,19]. Chen et al. divided the continuum robot into several units and modeled them using beam constraint model (BCM) to accurately calculate the deformation of the large deflection flexible arm [20]. These methods make the dynamic model computationally efficient, but the use of constant material parameters cannot describe the changes in the dynamics under different motion statuses such as position and velocity of the arm tip. Some researchers built comprehensive dynamic models by considering more factors such as friction or gravity. Rone et al. considered the friction between the driving wire and the constraint disks, and formulated a dynamic model based on the principle of virtual work [21]. Yang et al. analyzed the influence of friction, gravity, and geometric constraints on the motion of continuum robots [2]. However, the complex modeling methods may reduce the computational efficiency, and therefore make it difficult to apply the theoretical model to practical control. The computational speed of these models was often limited to less than 5 Hz, making them unsuitable for real-time control applications [22]. Other works established the dynamic model by converting the flexible body to a combination of mass-damper-spring elements. Yekutieli et al. used the mass-spring elements to model octopus-like arms in 2-D space [23,24]. Kang et al. established a 3-dimensional dynamic model for a pneumatic continuum robot based on a mass-damper-spring network [25]. Although these equivalent structures can describe the flexibility characteristics of the continuum robots in a concise and uniform way, the parameters of those mass-damper-spring elements are difficult to obtain [14].

Some parameter estimation methods have been proposed for rigid robots. Atkeson et al. identified the inertial parameters of a PUMA-600 robot based on the least squares method [26]. Swevers et al. simplified the parameter estimation process by using a periodic excitation method for a KUKA IR 361 robot [27]. Kozlowski et al. presented a joint planning method to decouple the inertia parameters of a ASEA IRp-6 robot, so the inertia matrix of each joint can be solved separately [28]. These methods focused on the inertia parameters of the rigid robots, but it is more critical for the continuum robot to identify the elastic and damping coefficients due to the complex interactions between the components [25]. Della Santina et al. proposed a scheme to equivalent the dynamic model of a soft robot to a rigid robot, and considered the parameters estimation as a regression problem [29–31]. Inspired by this work, we further investigate the variation of the dynamic parameters during robotic movements. For example, the dynamic stiffening effect will influence the elastic level [32] of the robot body, and the bending curvature will change the internal friction of the robot [2]. Therefore, it is necessary to use an adaptive algorithm to predict the dynamic parameters of a continuum robot in real time.

In this paper, a lumped-mass dynamic model (LMD) is established for continuum robots that have intrinsic compliance in their body structure. Then, an off-line parameter estimation scheme is proposed to determine the equivalent elastic and damping coefficients in the dynamic model with given postures and velocity based on a genetic algorithm (GA). Following this, a multilayer perceptron (MLP) is trained and used to select the suitable dynamic parameters according to the varying position and velocity of the continuum robot in real time. The LMD and the off-line/on-line combined parameter estimation strategy result in a dynamic model that can adapt to different working conditions with high accuracy and computing efficiency, i.e., adaptive lumped-mass dynamic model (ALMD). Based on the adaptive dynamic model, a feedforward control scheme for the continuum robot is further developed and validated in this work.

The paper is organized as follows: the adaptive lumped-mass dynamic model of a continuum robot and its parameter estimation method based on GA are presented in Section 2. An MLP is trained to predict the dynamic parameters based on the motion status (position and velocity of the arm tip) of the continuum robot in Section 3. Section 4 implements the dynamic model into a feed-forward control scheme and validates it on a continuum robotic prototype. Conclusions are given in Section 5.

2. Lumped-mass dynamic model for continuum robots

In this section, a dynamic model based on lumped-mass method for continuum robots is established [33]. The lumped-mass assumption allows for improvement in computational efficiency by characterizing the main dynamics of robot (i.e. dynamics of bending motion). Subsequently, a parameter estimation method based on minimizing the modeling error in the operational space is proposed.

2.1. A lumped-mass dynamic model for continuum robots

Continuum robots are usually composed of one or multiple modules, each of which contains three or four parallelly arranged actuators and a number of constraint elements (e.g. constraint disks) to prevent the actuators from radial displacement when performing bending motion [34]. In this paper, we use the typical module of a rod driven continuum robot to demonstrate our dynamic modeling method. As shown in Fig. 1(a), the driving rods are fixed to the end disk and moving through the other constraint disks freely. By controlling the displacements of the driving rods, the continuum robot can achieve 2°-of-freedom (DOF) bending motion. In this paper, three driving rods distributed by 120° apart in a circle are used to achieve desired motion.

The kinematics of the continuum module is regarded as a series of parallel mechanism with identical rotation angle $\boldsymbol{\psi} = \left[\psi_x, \psi_y\right]^T$, where ψ_x, ψ_y are the Euler angles of the moving platform about the x and y axes [35], respectively, as shown in Fig. 1(b). The rotation angle $\boldsymbol{\psi}$ is controlled by the length of the driving rods $\mathbf{l}$. The relationship between the rotation angle $\boldsymbol{\psi}$ of the upper plane of the parallel mechanism and the length of the driving rods is detailed in [36,37].

The stacked parallel mechanisms are further considered as a series of rigid links with identical rotation angle $\boldsymbol{\psi} = \left[\psi_x, \psi_y\right]^T$, as shown in Fig. 1(c). Each link can be described by a lumped mass point (the mass of each equivalent mass point is defined as m_e) on one end while the joint is represented by the rotational spring (k_x and k_y) and damping (c_x and c_y) about the x and y axis (i.e., $x_0, x_1 \ldots$ and $y_0, y_1 \ldots$ in Fig. 1). The driving force $\mathbf{f}$ is converted to an equivalent torque $\mathbf{T} = \left[T_x, T_y\right]^T$ through the matrix $\mathbf{B}$ [38].

$$\mathbf{T} = \mathbf{B}\mathbf{f} \tag{1}$$

Based on the above definitions, we can use the Lagrangian method to derive the dynamic equations:

(a) Kinetic energy. The coordinates of each equivalent mass point can be obtained from the kinematic model of the continuum robot. So the kinetic energy is

$$K = \frac{1}{2} m_e \sum_{i=0}^{n} \dot{\mathbf{x}}_i^{\,2} \tag{2}$$

(b) Gravitational potential energy. The coordinates of each equivalent mass point can be obtained from the kinematic model of the continuum robot. So, the gravitational potential energy can be obtained as

$$P_z = m_e g \sum_{i=1}^{n} h_i \tag{3}$$

(c) Elastic potential energy. The elastic potential energy of the system is the energy of each torsion spring. So the elastic potential energy can be obtained as

$$P_t = \sum_{i=1}^{n} \left(\frac{1}{2} k_x \psi_x^{\,2} + \frac{1}{2} k_y \psi_y^{\,2} \right) \tag{4}$$

Then the Lagrange equation can be obtained:

$$L = K - P_z - P_t \tag{5}$$

So the driving torque $\mathbf{T} = \left[T_x, T_y\right]^T$ can be written as:

$$\left.\begin{aligned} T_x - c_x \dot{\psi}_x &= \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{\psi}_x}\right) - \frac{\partial L}{\partial \psi_x} \\ T_y - c_y \dot{\psi}_y &= \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{\psi}_y}\right) - \frac{\partial L}{\partial \psi_y} \end{aligned}\right\} \tag{6}$$

Then, the dynamics of this system has been established based on the Lagrange method [35]:

Table 1
Nomenclature used in this paper.

Symbol	Description
$\boldsymbol{X}$	The position in the operational space, i.e. the position of end-tip
$\boldsymbol{\psi}$	The position in the configuration space, i.e. the rotation angle of the joints
$\boldsymbol{L}$	The position in the actuation space, i.e. the length of driving rods
$\boldsymbol{F}$	The force on the driving rods
$\boldsymbol{P}$	The combination of dynamic parameters $[k_x, k_y, c_x, c_y]$

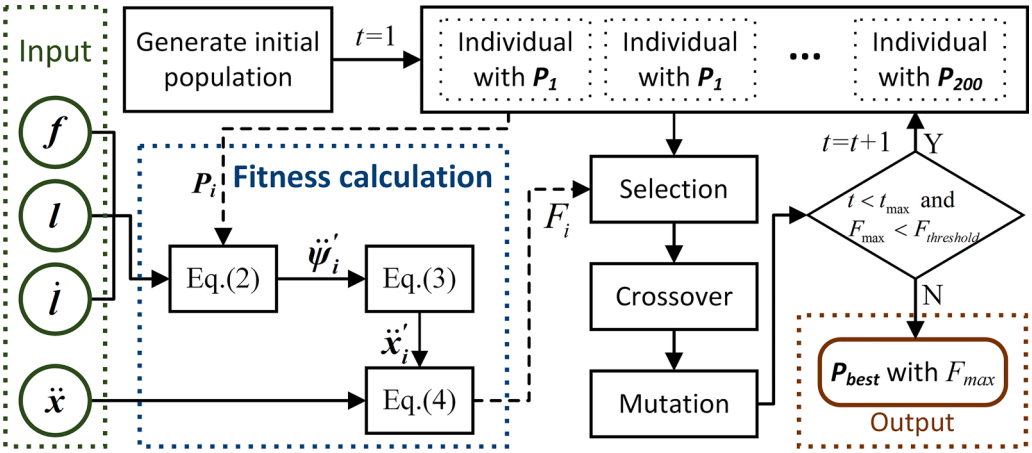

Fig. 2. Parameter estimation method based on GA.

$$\boldsymbol{T} = \mathbf{Bf} = \mathbf{M}(\boldsymbol{\psi})\ddot{\psi} + \mathbf{H}(\boldsymbol{\psi}, \dot{\psi}) + \mathbf{G}(\boldsymbol{\psi}) + \mathbf{K}(\mathbf{P})\boldsymbol{\psi} + \mathbf{C}(\mathbf{P})\dot{\psi} \tag{7}$$

where **M** is the inertia matrix:

$$\mathbf{M} = \begin{bmatrix} \frac{\partial}{\partial \dot{\psi}_x}\left(\frac{\partial K}{\partial \dot{\psi}_x}\right) & \frac{\partial}{\partial \dot{\psi}_y}\left(\frac{\partial K}{\partial \dot{\psi}_x}\right) \\ \frac{\partial}{\partial \dot{\psi}_x}\left(\frac{\partial K}{\partial \dot{\psi}_y}\right) & \frac{\partial}{\partial \dot{\psi}_y}\left(\frac{\partial K}{\partial \dot{\psi}_y}\right) \end{bmatrix} \tag{8}$$

H is the Coriolis and centrifugal terms:

$$\mathbf{H} = \begin{bmatrix} \frac{\partial}{\partial \psi_x}\left(\frac{\partial K}{\partial \dot{\psi}_x}\right) & \frac{\partial}{\partial \psi_y}\left(\frac{\partial K}{\partial \dot{\psi}_x}\right) \\ \frac{\partial}{\partial \psi_x}\left(\frac{\partial K}{\partial \dot{\psi}_y}\right) & \frac{\partial}{\partial \psi_y}\left(\frac{\partial K}{\partial \dot{\psi}_y}\right) \end{bmatrix} \tag{9}$$

G is the vector of gravity parts:

$$\mathbf{G} = \begin{bmatrix} \frac{\partial P_z}{\partial \psi_x} & \frac{\partial P_z}{\partial \psi_y} \end{bmatrix}^T \tag{10}$$

K is the elastic matrix $diag(k_x, k_y)$ and k_x, k_y are the elasticity coefficients along the x and y directions, respectively. **C** is the damping matrix $diag(c_x, c_y)$ and c_x, c_y are the elasticity coefficients along the x and y directions, respectively. And $\boldsymbol{\psi}$ is considered as the rotation angle of the joints here, which means the position of the robot in the configuration space. The detailed parameters description of the proposed model is listed in Table 1.

2.2. Parameter estimation based on GA

In Section 2.1, a dynamic model for continuum robots is established, but the parameters in the model need to be determined. Some of these parameters can be explicitly calculated. For instance, the length of each link is averaged by the total length of the robot and the mass of each mass point is averaged by the total mass. However, for parameters like the rotational spring and damping coefficients, a parameter estimation method is necessary. In this section, the parameter estimation is regarded as an optimization problem to

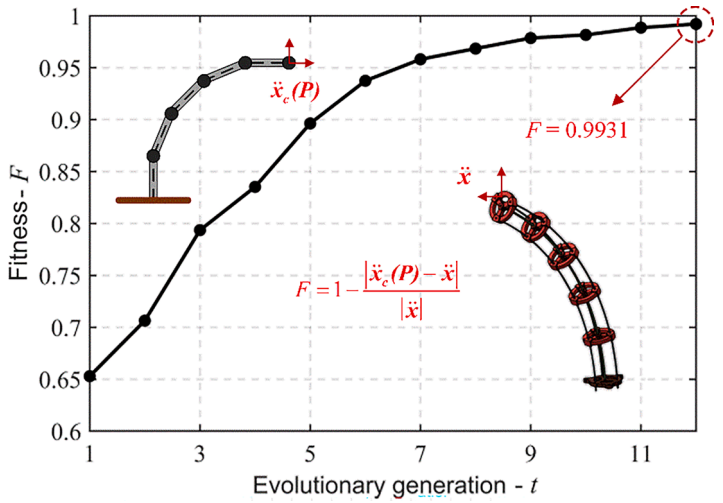

Fig. 3. Fitness evolution curve of genetic algorithm.

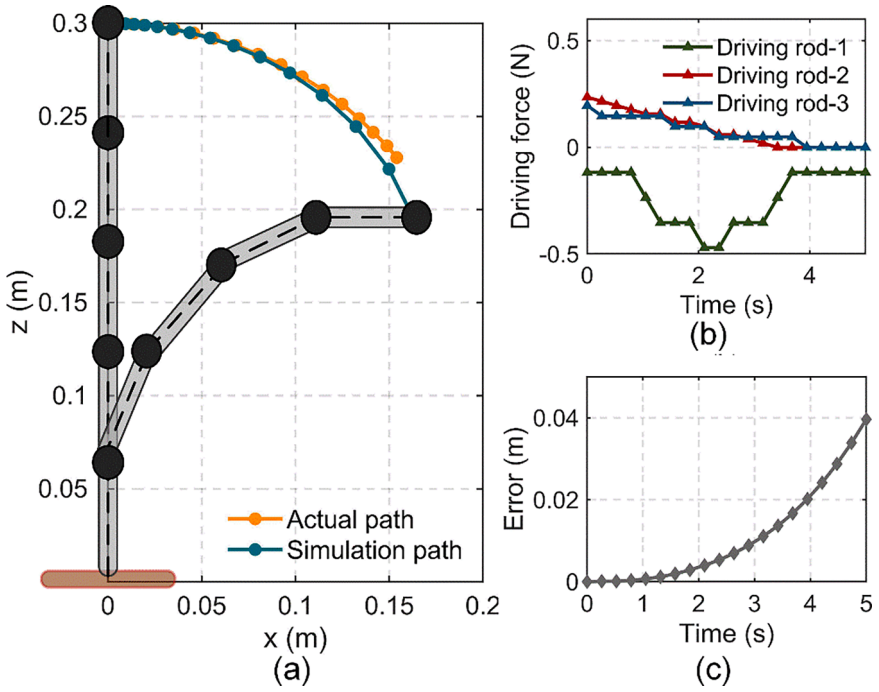

Fig. 4. Comparison of simulation and experimental results: (a) a comparison of the path at the arm tip, (b) the applied driving force, (c) the error between the LMD and prototype.

minimize the error between the actual acceleration in the operational space and the calculated acceleration obtained from the dynamic model. This optimization problem is then solved using a GA [39].

As shown in Fig. 2, individuals with different dynamic parameters $\mathbf{P_i}$ in the population are applied to the dynamic model. After multiple generations of selection, crossover and mutation operators [39], the one with the highest fitness is selected. The required inputs include the actual acceleration of the arm tip, $\ddot{\mathbf{x}}$, driving force, f, the displacement of the driving rod, $\mathbf{l}$ and its velocity $\dot{\mathbf{l}}$. These inputs can be collected by the sensor system of the continuum robot. A detailed description of the robotic prototype and sensor system are introduced in Section 4. Taking these inputs, the position and velocity in the configuration space, $\boldsymbol{\psi}(\mathbf{l})$ and $\dot{\psi}(\mathbf{l},\dot{\mathbf{l}})$, can be obtained through the kinematics of the continuum robot [35]. According to Eq. (7), the calculated value of the acceleration in the configuration space $\ddot{\psi}_{\mathbf{i,c}}$ satisfies the following equation.

$$\ddot{\psi}_{\mathbf{i,c}} = \mathbf{M}^{-1}(\mathbf{Bf} - \mathbf{H} - \mathbf{G} - \mathbf{K}(\mathbf{P})\boldsymbol{\psi} - \mathbf{C}(\mathbf{P})\dot{\psi}) \tag{11}$$

Table 2
Data of a parameter estimation example.

	Symbol	Value	Unit
Input	$\ddot{\boldsymbol{x}}$	[0.1325,0.2076,0.0113]	m/s^2
	$\boldsymbol{l}$	[0.3000,0.3000,0.3000]	m
	$\dot{\mathbf{l}}$	[0.0000,0.0000,0.0000]	m/s
	$\boldsymbol{f}$	[0.3025,0.3263,−0.1265]	N
output	$\boldsymbol{P}$	[0.1008,0.1085,0.0218,0.0298]	N•m

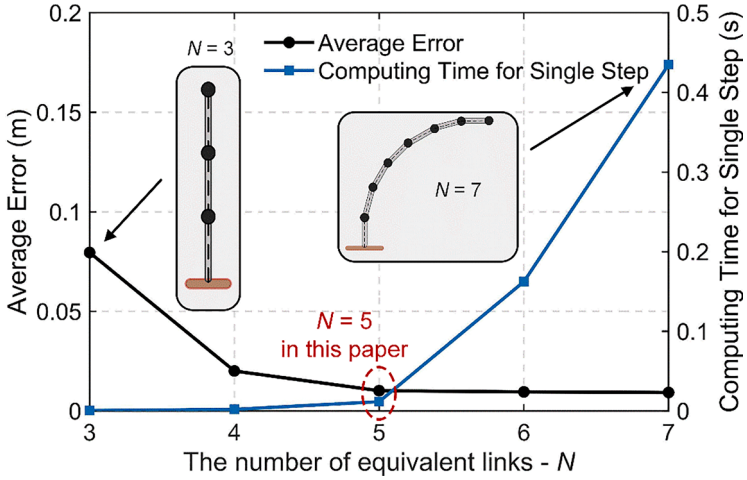

Fig. 5. Fitness evolution curve of genetic algorithm.

Furthermore, the calculated acceleration in the operational space $\ddot{\mathbf{x}}_{\mathbf{i,c}}$ can be expressed by

$$\ddot{\mathbf{x}}_{\mathbf{i,c}} = \mathbf{J} \cdot \ddot{\psi}_{\mathbf{i,c}} + \dot{\mathbf{J}} \cdot \dot{\psi}_{\mathbf{i,c}} \tag{12}$$

where $\mathbf{J}$ is the Jacobian matrix of the continuum robot. It can be seen that the calculated value of the acceleration $\ddot{\mathbf{x}}_{\mathbf{i,c}}(\psi, \dot{\psi}, \mathbf{P_i})$ is a function of $\boldsymbol{\psi}, \dot{\psi}$, and $\mathbf{P_i}$. Thus, for a given position and velocity in the configuration space, $\boldsymbol{\psi}$ and $\dot{\psi}$, the dynamic parameters $\mathbf{P_{best}}$ can be estimated using GA by looking for the minimum error between the calculated value $\ddot{\mathbf{x}}_{\mathbf{c}}$ and the actual value $\ddot{\mathbf{x}}$.

In Fig. 2, the optimization objective is normalized to set the fitness function as

$$F_i = 1 - \frac{\left|\ddot{\mathbf{x}}_{\mathbf{i,c}}(\mathbf{P_i}) - \ddot{\mathbf{x}}\right|}{|\ddot{\mathbf{x}}|} \tag{13}$$

The population size for our GA is 200, the maximum GA iteration $t_{\max}$ is 50 and the threshold of the fitness $F_{threshold}$ is 0.99. As shown in Fig. 3, a parameter estimation example was carried out. When applying a driving force to a 2-DOF robotic prototype with initial position and velocity $\mathbf{l}$ = [0.3,0.3,0.3], $\dot{\mathbf{l}}$= [0.0,0.0,0.0], the individual with maximum fitness Fmax=0.9931 is obtained after 12 iterations, meaning the error between the calculated value $\ddot{\mathbf{x}}(\mathbf{P})$ and the actual value $\ddot{\mathbf{x}}$ is less than 1 %. By using this method, a set of dynamic parameters $\mathbf{P}$ that corresponds to different initial motion statuses can be obtained. Note that, for continuum robots with multiple modules [2], it is necessary to collect motion statuses for each module and perform parameter estimation one by one.

To verify the accuracy of the dynamic model with the parameter $\mathbf{P}$ obtained from GA, a new set of driving forces, Fig. 4(a) and (b), were applied to the continuum robotic prototype whose initial position and velocity, $\mathbf{l}$ and $\dot{\mathbf{l}}$, are the same as the example shown in Table 2. The end position of this updated robotic prototype in the operational space were first collected. Then the driving force was also applied to the dynamic model with the same initial status, and the simulated results of the end position compared with the actual ones collected from the prototype. As shown in Fig. 4(c), the model and prototype are consistent at the beginning, but the error increases gradually. This is mainly because the dynamic parameters used in the simulated model are the estimated ones from the initial configuration. However, as the continuum robot continues to move, the use of those constant parameters no longer matches the actual dynamics of the robot in other configurations. This phenomenon will be analyzed in detail in Section 2.3.

Another important factor affecting modeling accuracy and computational speed is the number of equivalent links, N, in the dynamic model. Fig. 5 shows the average error and computational time of the dynamic model for different numbers of equivalent links. The simulation platform used is MATLAB (MATLAB 2020a, The MathWorks Inc., Natick, MA), and simulations are performed on a 3.40 GHz AMD Ryzen-5–2600 6-Core CPU. As the number of equivalent links increases, the computational time for a single step (which

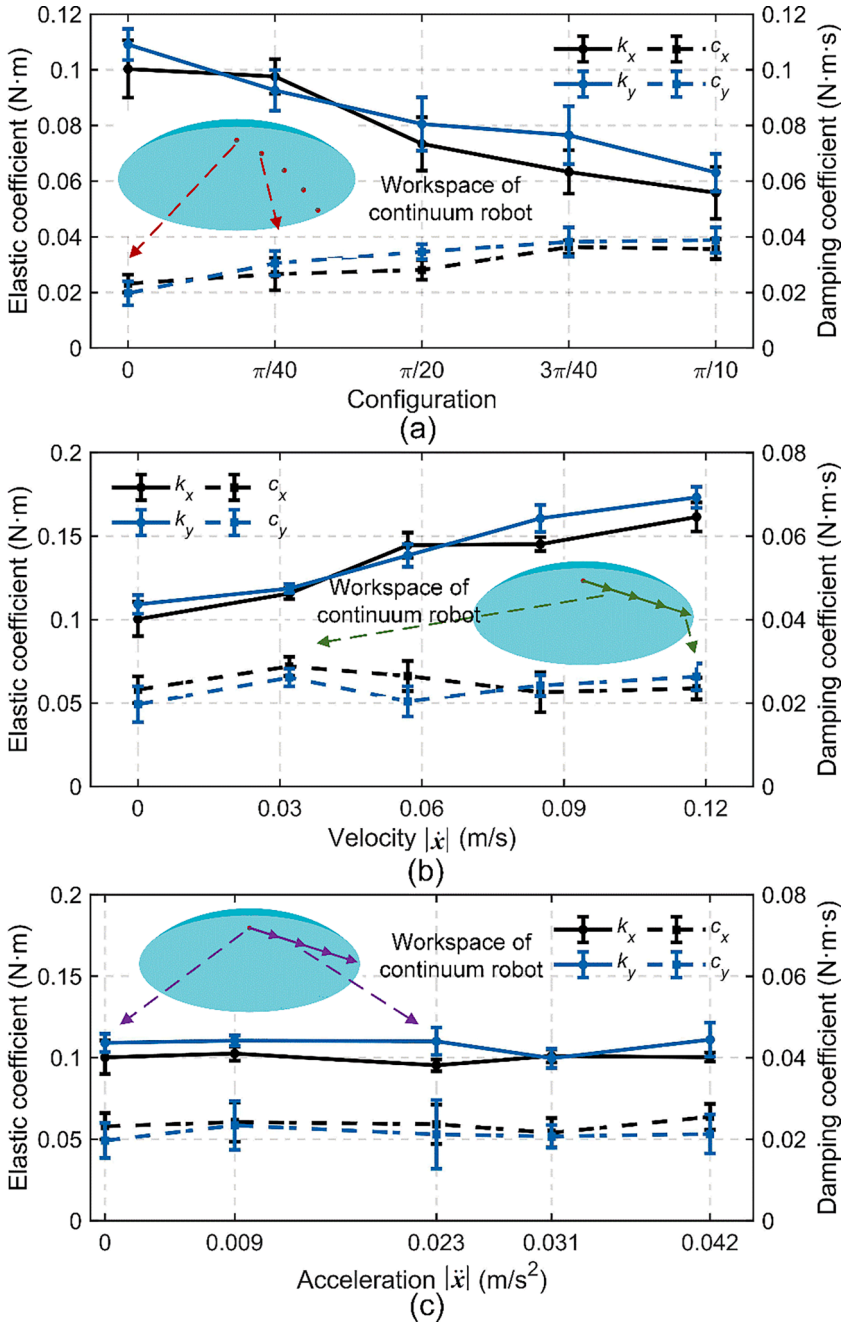

Fig. 6. The change of dynamic parameters with different statuses:(a) the change of dynamic parameters with different configuration, (b) the change of dynamic parameters with different velocity, (c) the change of dynamic parameters with different acceleration.

means the dynamic equation Eq. (11) is calculated once each) increases because the model is getting more complex. On the other hand, the accuracy of the dynamics is improved when the number of equivalent links increases. Considering the possibility of applying the dynamic model to a real-time control system, the number of equivalent links is selected as $N = 5$ in this work to ensure that a control frequency higher than 50 Hz. As a comparison, other dynamic models generally work with a control frequency less than 5 Hz [22,38].

2.3. Variable dynamic parameters

In Fig. 4, it was proven that the constant parameter dynamic model cannot be applied to all continuum robot statuses by demonstrating the tracking error along a path. Therefore, we need to investigate what factors affect dynamic parameters the most

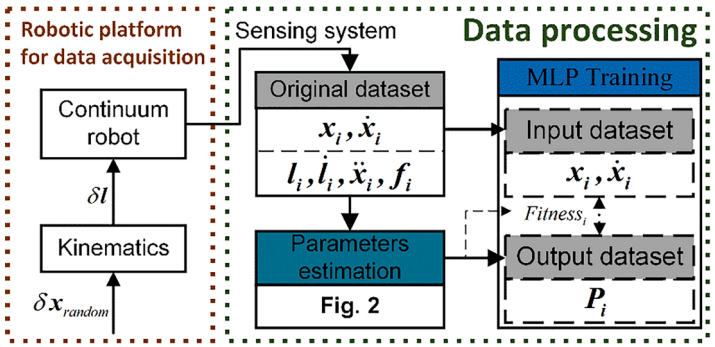

Fig. 7. Data acquisition and processing scheme.

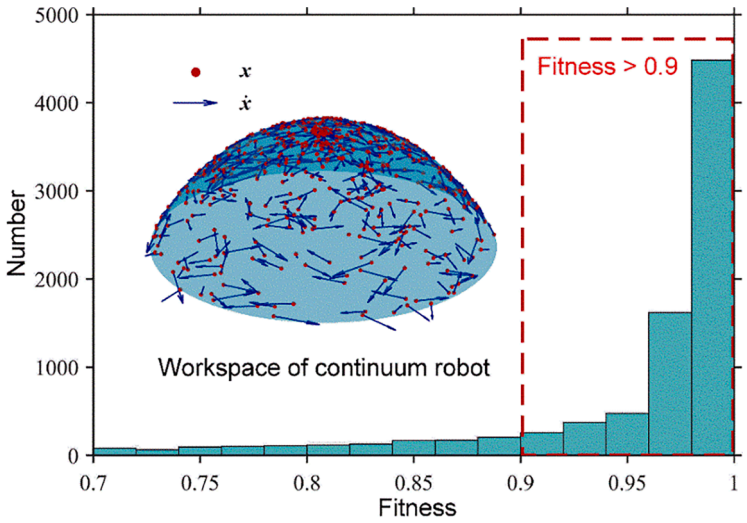

Fig. 8. Position, velocity and fitness of the samples.

during a robotic movement.

As shown in Fig. 6, three sets of experiments were performed for the continuum robot, and the dynamic parameters are estimated by using the method presented in Section 2.2. Fig. 6(a) shows the change of the dynamic parameters P (i.e. $[k_x, k_y, c_x, c_y]$) with varying configurations (i.e. different end positions) yet the same velocity ($\dot{\mathbf{x}} = [0,0,0]$m /s) and driving force ($\mathbf{f} = [0.31, 0.33, -0.15]$N). When the configuration angle $|\psi|$ becomes larger, the elastic coefficients about the x and y directions become smaller and the damping coefficients about the x and y directions become larger. The decrease of the elastic coefficient is mainly due to the fact that the backbone and driving rods reach their yield limit as the bending becomes larger [40]. The increased damping coefficient is mainly due to the increase in friction generated by the increased contact (more contact points and higher contact force) between the driving rod and the through-hole in the constraint disks when the robot bends with a large curvature [2]. Although the configuration change here is in a 2D plane. It proves that the dynamic parameters will change when the robot moves.

Fig. 6(b) shows the change of dynamic parameters with different velocities in the operational space while maintaining the same configuration ($\psi = [0,0]$) and driving force ($\mathbf{f} = [0.31, 0.33, -0.15]$N). When the velocity $|\dot{\mathbf{x}}|$ becomes larger, the elastic coefficients about the x and y directions become larger yet the damping coefficients about the x and y directions have very little change. The increase of the elastic coefficient is mainly due to the dynamic stiffening effect of the materials [32].

Fig. 6(c) shows the change of the dynamic parameters with different acceleration in the operational space, while maintaining the same configuration ($\psi = [0,0]$) and velocity ($\dot{\mathbf{x}} = [0,0,0]$m /s) in the operational space. When the acceleration $|\ddot{\mathbf{x}}|$ becomes larger, both the elastic and damping coefficients change little. This indicates that the acceleration $|\ddot{\mathbf{x}}|$ has little influence on the change of dynamic performance, and the influence of acceleration can be ignored when discussing the factors affecting dynamic parameters.

In view of Fig. 6, it can be seen that the elastic and damping coefficients are dependent on the configuration and velocity of the robot. The presence of multi-factor correlation will further increase the difficulty in obtaining the analytical expressions of the dynamic coefficients. Thus, we utilize a neural network method to achieve the prediction of these coefficients.

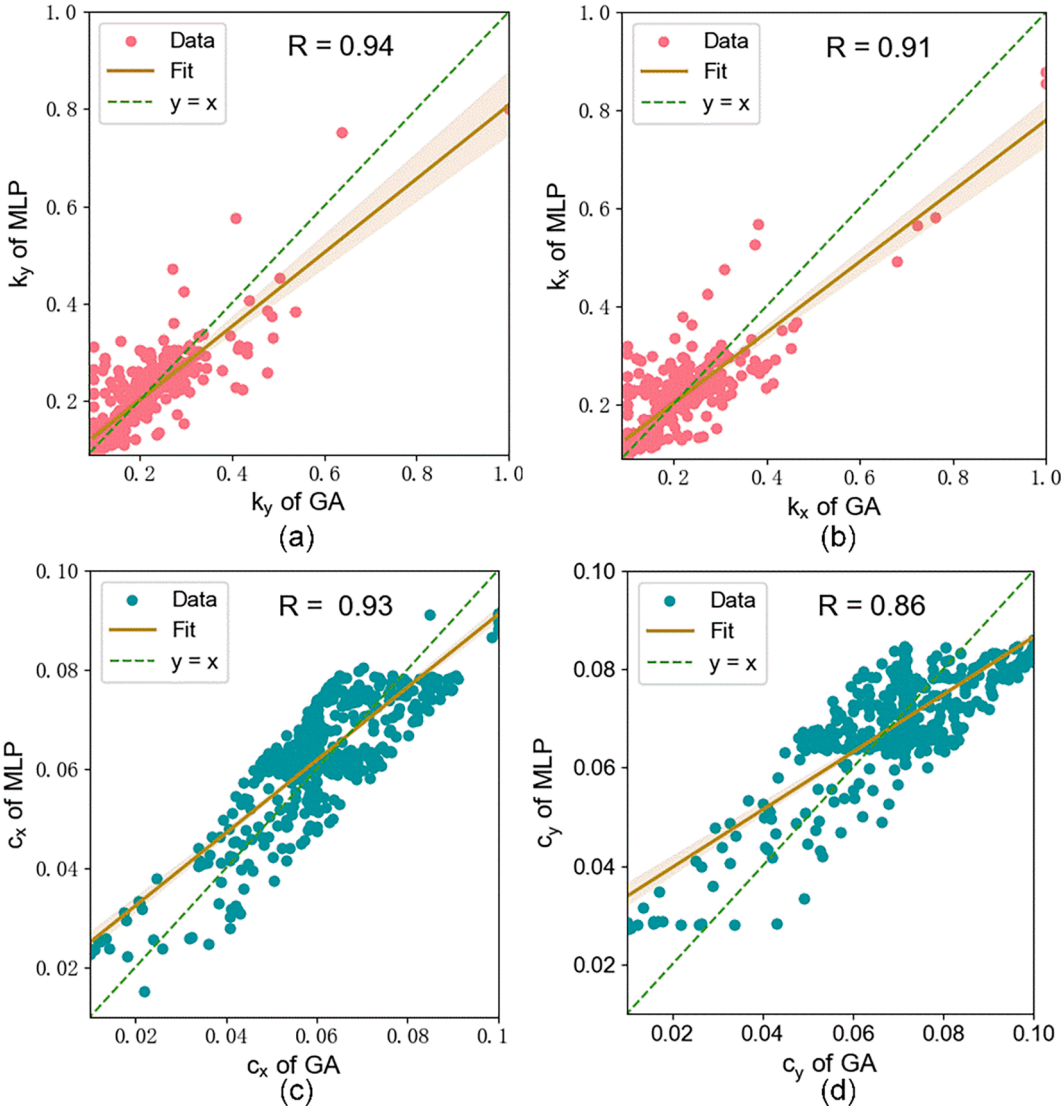

Fig. 9. Comparison between the outputs of MLP and GA: (a)k_x, (b)k_y, (c)c_x, (d)c_y.

3. Parameter prediction based on MLP

In this section, a MLP is trained to predict the appropriate dynamic parameters according to the position and velocity of the continuum robot. The accuracy of the obtained dynamic model containing those adaptive parameters is then verified by experiments.

3.1. Data collection and processing

The main factors affecting the dynamic parameters are configuration and velocity, which have been proved in Section 2.3. Therefore, the input layer of the MLP contains the configuration angle and velocity while the output layer is the corresponding dynamic parameter P. However, the configuration angle is difficult to measure directly, so it is replaced by the position in the operational space **x** in our single module robot. As shown in Fig. 7, a robotic prototype based on inverse kinematics is used to realize different positions and velocities for the robot in the operational space [35].

In our case, a dataset containing 10,635 sample pairs (input: the position $\mathbf{x_i}$ and the velocity $\dot{\mathbf{x}}_\mathbf{i}$ in the operational space; output: the dynamic parameters $\mathbf{P_i}(i = 1, 2, ..., 10635)$) has been collected by driving the continuum robot to achieve different motion statuses through random vectors $\delta\mathbf{x}_{random}$ in the operational space. Fig. 8 shows the samples are distributed in different positions, and the velocities are in different directions and norms. This makes the neural network trained from the data set have better generalization ability rather than being only applied to a specific range of positions and velocities [41]. For each sample, we used the GA based estimation method for the corresponding dynamic parameters $\mathbf{P_i}$, as well as its fitness. The histogram in the figure reflects the fitness

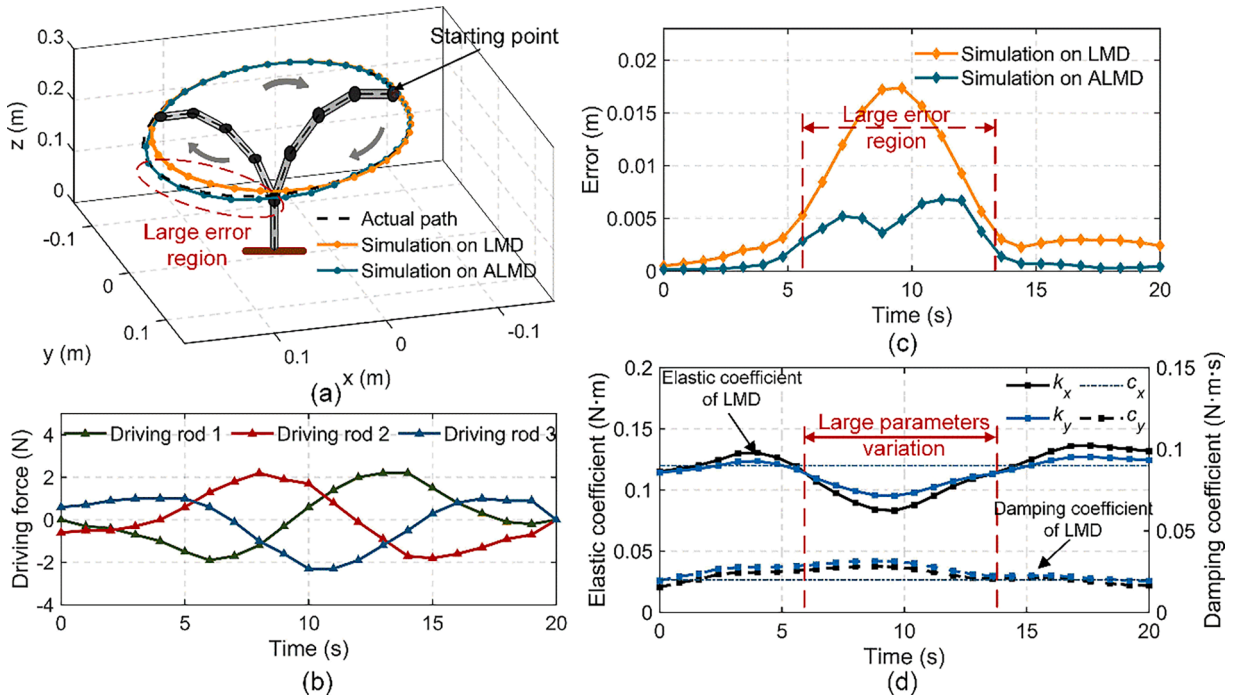

Fig. 10. Comparison of simulated and experimental results:(a) a comparison of the path of the robot tip, (b) the driving force, (c) the error of the LMD and ALMD, (d) the change of dynamic parameters in the ALMD.

distribution of the samples. GA cannot always identify accurate results due to factors such as falling into local optimum or measurement errors. So, a sample filter is utilized to delete the data with fitness less than 0.9. After filtering, 7214 samples remained in the dataset.

3.2. Training MLP

Based on the dataset, a neural network MLP fitting the dynamic parameters and motion status (the position and velocity in the operational space) is trained. The MLP used to predict dynamic parameters is a neural network with three layers. The input layer of the MLP has six neurons (the position and velocity in the operational space are three-dimensional vectors, respectively), and the output layer has four neurons (the dynamic parameters **P** is a four-dimensional vectors). The MLP has a hidden layer with five fully connected neurons [41,42]. The log-sigmoid function and mean square error are utilized as an activation function and loss function for the MLP, respectively. The Bayesian regularization approach is used as the training algorithm [43]. The dataset is divided into 70 % training set and 30 % testing set and the network is developed using MATLAB.

To improve the quality of the training, a filter based on roulette wheel selection algorithm is used to make the data with high fitness have a high probability to participate in the MLP training process [44]. Fig. 9 shows a comparison between the output of the MLP and the original data from the GA in the testing set. Most of the data are concentrated near the line $y = x$ and the correlation coefficient $R > 0.85$, meaning the MLP fits the relationship between the motion status (i.e. position and velocity) of the continuum robot and the dynamic parameters [45].

3.3. Adaptive lumped-mass dynamic model

An adaptive dynamic model is established using the MLP to predict system parameters. An experiment was carried out to verify the difference between the ALMD and LMD. As shown in Fig. 10(a) and (b), the continuum robotic prototype was moving in the shown operational space while driving force and position information was collected. In this section, the expression of desired path is as follows:

$$\begin{cases} \psi_x = \dfrac{2\pi + \cos\left(\dfrac{t}{10} - \pi\right)\pi}{6N} \cdot \sin\left(\dfrac{t}{10} - \pi\right) \\ \psi_y = \dfrac{2\pi + \cos\left(\dfrac{t}{10} - \pi\right)\pi}{6N} \cdot \cos\left(\dfrac{t}{10} - \pi\right) \end{cases}, 0 \le t \le 20s \tag{14}$$

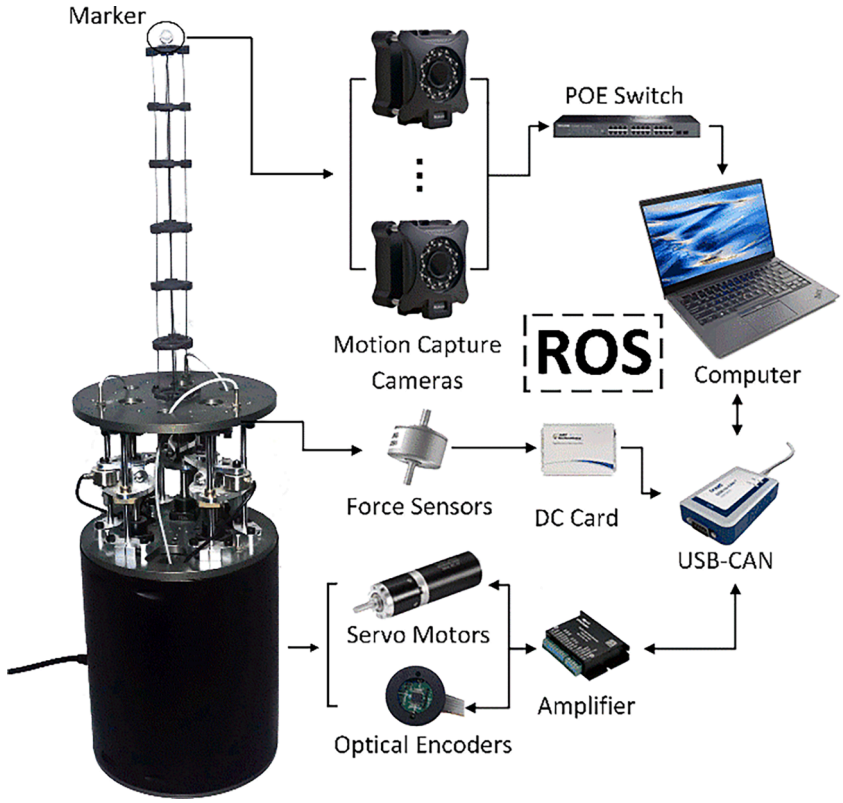

Fig. 11. Prototype of the continuum robot.

where $\boldsymbol{\psi} = [\psi_x, \psi_y]^T$ is the rotation angles of the continuum robot in the configuration space. The driving force was then applied to both the ALMD and LMD. The resulting simulated paths are compared with the actual path in Fig. 10(c), which shows the maximum error between the LMD and prototype is 60.2 % larger than that between the ALMD and prototype. This is due to that the maximum value of $|\boldsymbol{\psi}|$ is achieved around $t = 10$ s, according to Eq.(14), which indicates the maximum bending of the continuum robot. So, the dynamics of the continuum robot changed greatly in this period, and the LMD cannot reflect the large variation of dynamic parameters (mainly the elastic coefficient about the x and y direction) as shown in Fig. 10(d).

4. Experiment

In this section, the design of the continuum robot prototype are first summarized. Then a feedforward control method based on the adaptive parameter dynamic model is implemented into the robotic system to improve control accuracy.

4.1. Experimental platform

As shown in Fig. 11, the prototype of the continuum robot is composed of a 2- DOF continuum arm and the sensing/driving system

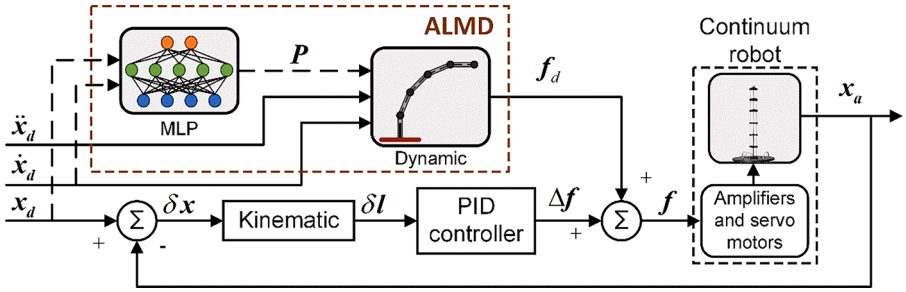

Fig. 12. The feedforward control based on ALMD.

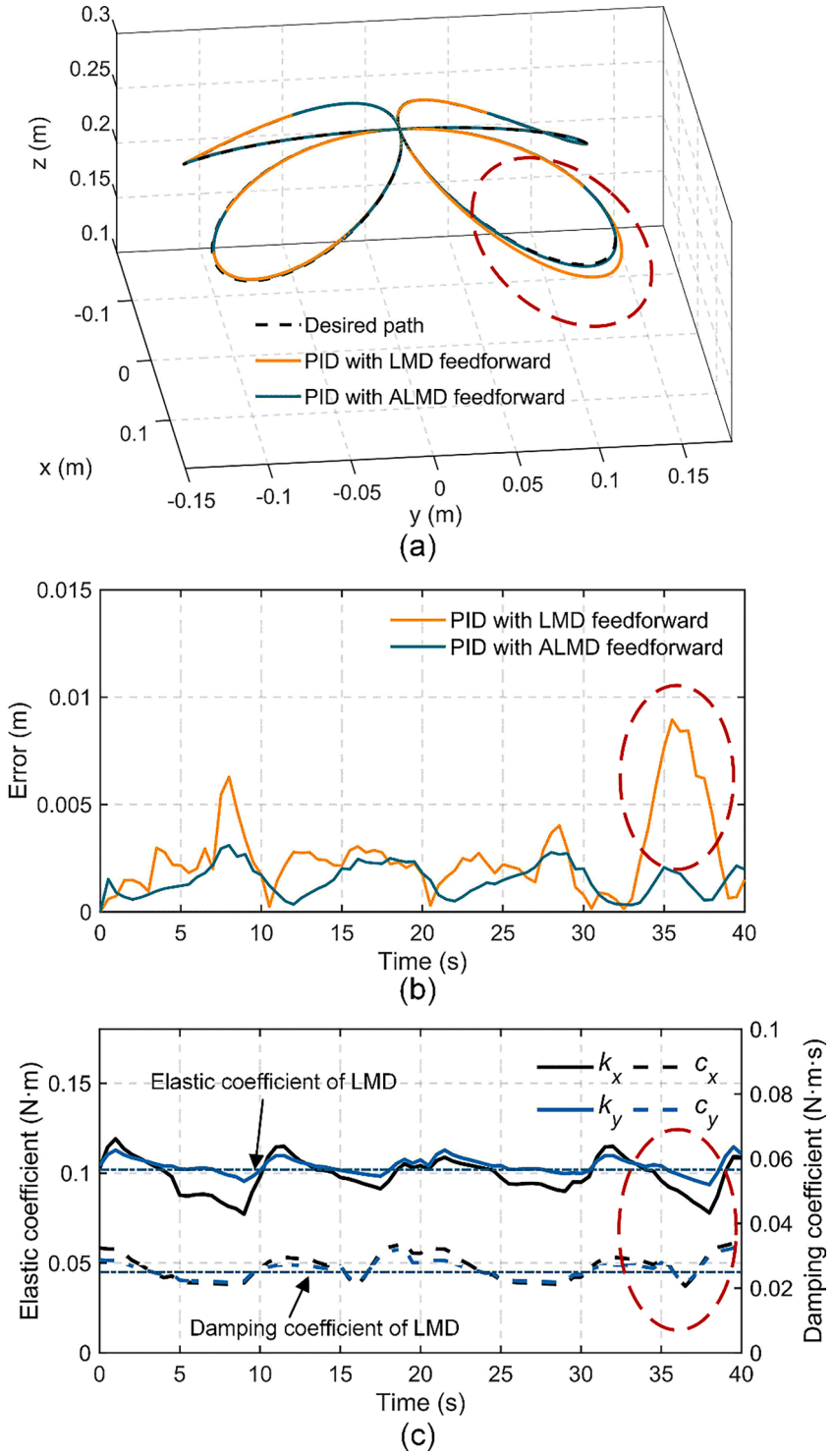

Fig. 13. Experimental results:(a) a comparison of the path, (b) the tracking error in different control schemes, (c) the change of the dynamic parameters in the ALMD.

[2]. The continuum arm contains a backbone, three driving rods distributed 120° apart and 7 constraint disks. The total length of the arm is 300 mm, and the sets of servo motors and ball screws to pull or push the driving rods. An optical motion capture system including four cameras and a marker (The NOKOV Inc., China) is used to measure the position of the arm tip within its operational space. The arm tip velocities and accelerations are provided by calculating the first order and second order time derivatives of its

position. Each servo motor is equipped with an optical encoder to measure the rotation angle for the use of calculating the actual length of the driving rod. A combination of force sensors and equal-arm lever mechanisms realize the measurement of the driving force [2]. The sensing and driving systems are connected to the central PC through a POE (Power Over Ethernet) switch and CAN (Controller Area Network) bus. The control program is based on ROS in multiple nodes to achieve coordinated control for those devices.

4.2. The feedforward control based on the adaptive dynamic model

As shown in Fig. 12, a feedforward channel based on the ALMD is combined with a kinematic PID control scheme for the continuum robot. To evaluate its performance, the control effects with LMD feedforward and pure PID are compared [46]. Due to the efficiency of the equivalent dynamics model, the frequency of the control scheme can reach 50 Hz.

The kinematic controller converts the position error $\Delta\mathbf{x}$ in the operational space to the actuator displacement $\delta\mathbf{l}$ in the actuation space. The PID controller generates the driving force $\Delta\mathbf{f}$ to compensate for the positioning error. In addition, the driving force $\mathbf{f}_d$ is used to compensate the dynamic effects of the continuum robot which is obtained through a dynamic model (could be either LMD or ALMD). The sum of the kinematics based PID control $\Delta\mathbf{f}$, and the dynamic feedforward compensation $\mathbf{f}_d$, is used as the total driving force $\mathbf{f}$ to the robot (Note that, as the output of the controller, the driving force $\mathbf{f}$ is represented in a form of electrical signal, which is then converted to mechanical force with the help of the amplifier and servo motor). This control scheme can improve the dynamic performance by considering the inertia, elasticity and damping of the continuum robot. A set of experiments were carried out to validate the performance of different control schemes, as shown in Fig. 13. The PID with LMD feedforward and PID with ALMD feedforward were used to track a given path. The same PID parameters ($k_p = 9.6$, $k_i = 1.5$ and $k_d = 1.2$, obtained by the ISTE criterion [47]) were used in the two schemes so we can clearly compare the effects of different feedforward compensation methods. Compared with PID with LMD feedforward, adding ALMD based feedforward compensation will significantly reduce the maximum tracking error by 67.5 % (denoted by the red dashed circle in Fig. 13). This is because the LMD is unable to reflect the actual dynamics of the robot and provides an inaccuracy compensation force to the robot. This experiment shows the use of adaptive dynamic parameters can achieve more accurate feedforward compensation.

5. Conclusion

In this paper, an equivalent dynamic model for continuum robots is established. Then a GA-based dynamic parameter estimation method is proposed to identify the equivalent elastic and damping coefficients in the model, which is related to the motion status, e.g. configuration and velocity. Next a multi-layer perceptron is trained to predict the dynamic parameters of the continuum robot according to the motion status. The combination of the off-line parameter estimation and on-line parameter prediction methods ensures computational efficiency, while maintaining high accuracy for the dynamics model. A set of simulations were carried out to prove that the maximum error of the ALMD with adaptive parameters is reduced by 60.2 % compared with the LMD with constant parameters. Finally, on the experimental platform, a feedforward controller employing the ALMD was developed to compensate for the kinematic PID control scheme, and contributed in the reduction in the maximum error by 67.5 %.

This work constructed a computationally efficient dynamic model with satisfactory accuracy for real time control and considered the influence of the motion status (i.e. position and velocity) of the continuum robot on its dynamic performance. Future work will include environmental interactions into the modeling and control process to extend the applications of the presented methods.

CRediT authorship contribution statement

Xu Zhang: Writing – original draft, Methodology. **Chenghao Yang:** Validation, Methodology, Data curation. **Zhibin Song:** Writing – review & editing, Validation, Formal analysis. **Mojtaba A. Khanesar:** Writing – review & editing, Validation, Methodology. **David T Branson:** Writing – review & editing, Investigation. **Jian S. Dai:** Writing – review & editing, Supervision. **Rongjie Kang:** Writing – review & editing, Supervision, Project administration, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (grant no. 52375023, 51975401).

References

[1] R.J. Webster, B.A. Jones, Design and kinematic modeling of constant curvature continuum robots: a review, Int. J. Rob. Res. 29 (13) (2010) 1661–1683.
[2] C. Yang, S. Geng, I. Walker, D.T. Branson, J. Liu, J.S. Dai, R. Kang, Geometric constraint-based modeling and analysis of a novel continuum robot with Shape Memory Alloy initiated variable stiffness, Int. J. Rob. Res. 39 (14) (2020) 1620–1634.
[3] J.S. Dai, N. Holland, D.R. Kerr, Finite twist mapping and its application to planar serial manipulators with revolute joints, Proceed. Institut. Mech. Eng., Part C: J. Mech. Eng. Sci. 209 (4) (1995) 263–271.
[4] B.A. Jones, I.D. Walker, Kinematics for multisection continuum robots, IEEE Transact. Robot. 22 (1) (2006) 43–55.
[5] F. Renda, M. Cianchetti, M. Giorelli, A. Arienti, C. Laschi, A 3D steady-state model of a tendon-driven continuum soft manipulator inspired by the octopus arm, Bioinspir. Biomim. 7 (2) (2012) 025006.
[6] E. Tatlicioglu, I.D. Walker, D.M. Dawson, Dynamic modelling for planar extensible continuum robot manipulators, in: IEEE International Conference on Robotics and Automation, Apr. 2007, pp. 1357–1362.
[7] M. Rolf, J.J. Steil, Efficient exploratory learning of inverse kinematics on a bionic elephant trunk, IEEE. Trans. Neural. Netw. Learn. Syst. 25 (6) (2014) 1147–1160.
[8] X. Dong, D. Axinte, D. Palmer, S. Cobos, M. Raffles, A. Rabani, J. Kell, Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines, Robot. Comput. Integr. Manuf. 44 (2017) 218–229.
[9] Z. Ping, T. Zhang, C. Zhang, J. Liu, S. Zuo, Design of contact-aided compliant flexure hinge mechanism using superelastic nitinol, J. Mech. Des. 143 (11) (2021) 114501.
[10] K. Xu, N. Simaan, An Investigation of the Intrinsic Force Sensing Capabilities of Continuum Robots, IEEE. Transact. Robot. 24 (3) (2008) 576–587.
[11] A. Bajo, N. Simaan, Hybrid motion/force control of multi-backbone continuum robots, Int. J. Rob. Res. 35 (4) (2016) 422–434.
[12] H.-S. Yoon, B.-J. Yi, A 4-DOF flexible continuum robot using a spring backbone, in: International Conference on Mechatronics and Automation, Aug. 2009, pp. 1249–1254.
[13] D.-G. Choi, B.-J. Yi, W.-K. Kim, Design of a spring backbone micro endoscope, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, Oct. 2007, pp. 1815–1821.
[14] R. Kang, D.T. Branson, E. Guglielmino, D.G. Caldwell, Dynamic modeling and control of an octopus inspired multiple continuum arm robot, Comput. Math. Applic. 64 (5) (2012) 1004–1016.
[15] J.S. Dai, Euler–Rodrigues formula variations, quaternion conjugation and intrinsic connections, Mech. Mach. Theory. 92 (2015) 144–152.
[16] I.S. Godage, D.T. Branson, E. Guglielmino, G.A. Medrano-Cerda, D.G. Caldwell, Dynamics for biomimetic continuum arms: a modal approach, in: IEEE International Conference on Robotics and Biomimetics, Dec. 2011, pp. 104–109.
[17] I.A. Gravagne, C.D. Rahn, I.D. Walker, Large deflection dynamics and control for planar continuum robots, IEEE/ASME. Transact. Mechatron. 8 (2) (2003) 299–307.
[18] F. Renda, F. Boyer, J. Dias, L. Seneviratne, Discrete Cosserat Approach for Multisection Soft Manipulator Dynamics, IEEE. Transact. Robot. 34 (6) (2018) 1518–1533.
[19] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, C. Laschi, Dynamic model of a multibending soft robot arm driven by cables, IEEE. Transact. Robot. 30 (5) (2014) 1109–1122.
[20] F. Ma, G. Chen, Modeling large planar deflections of flexible beams in compliant mechanisms using chained beam-constraint-model1, J. Mech. Robot. 8 (2) (2016) 021018.
[21] W.S. Rone, P. Ben-Tzvi, Continuum robot dynamics utilizing the principle of virtual power, IEEE. Transact. Robot. 30 (1) (2014) 275–287.
[22] S.M.H. Sadati, S.E. Naghibi, A. Shiva, B. Michael, L. Renson, M. Howard, C.D. Rucker, K. Althoefer, T. Nanayakkara, S. Zschaler, C. Bergeles, H. Hauser, I. D. Walker, TMTDyn: a matlab package for modeling and control of hybrid rigid–continuum robots based on discretized lumped systems and reduced-order models, Int. J. Rob. Res. 40 (1) (2021) 296–347.
[23] Y. Yekutieli, R. Sagiv-Zohar, R. Aharonov, Y. Engel, B. Hochner, T. Flash, Dynamic model of the octopus arm. I. Biomechanics of the octopus reaching movement, J. Neurophysiol. 94 (2) (2005) 1443–1458.
[24] N. Giri, I.D. Walker, Three module lumped element model of a continuum arm section, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011, pp. 4060–4065. Sep.
[25] R. Kang, Y. Guo, L. Chen, D.T. Branson, J.S. Dai, Design of a pneumatic muscle based continuum robot with embedded tendons, IEEE/ASME. Transact. Mechatron. 22 (2) (2017) 751–761.
[26] C.G. Atkeson, C.H. An, J.M. Hollerbach, Estimation of inertial parameters of manipulator loads and links, Int. J. Rob. Res. 5 (3) (1986) 101–119.
[27] J. Swevers, W. Verdonck, J. De Schutter, Dynamic model identification for industrial robots, IEEE. Control. Syst. Magaz. 27 (5) (2007) 58–71.
[28] K.R. Kozlowski, P. Dutkiewicz, Experimental Identification of Robot and Load Dynamics, IFAC. Proceed. 29 (1) (1996) 397–402.
[29] C.D. Santina, R.K. Katzschmann, A. Bicchi, D. Rus, Model-based dynamic feedback control of a planar soft robot: trajectory tracking and interaction with the environment, Int. J. Rob. Res. 39 (4) (2020) 490–513.
[30] C.D. Santina, R.K. Katzschmann, A. Biechi, D. Rus, Dynamic control of soft robots interacting with the environment, in: IEEE International Conference on Soft Robotics, 2018, pp. 46–53. Apr.
[31] M. Trumić, C. della Santina, K. Jovanović, A. Fagiolini, Adaptive control of soft robots based on an enhanced 3d augmented rigid robot matching, in: American Control Conference, 2021, pp. 4991–4996. May.
[32] T.R. Kane, R.R. Ryan, A.K. Banerjeer, Dynamics of a cantilever beam attached to a moving base, J. Guid. Cont. Dyn. 10 (2) (1987) 139–151.
[33] Y.T. Chai, K.S. Varyani, N.D.P. Barltrop, Three-dimensional Lump-Mass formulation of a catenary riser with bending, torsion and irregular seabed interaction effect, Ocean Eng. 29 (12) (2002) 1503–1525.
[34] J.A. Saglia, N.G. Tsagarakis, J.S. Dai, D.G. Caldwell, A high-performance redundantly actuated parallel mechanism for ankle rehabilitation, Int. J. Rob. Res. 28 (9) (2009) 1216–1227.
[35] X. Zhang, Y. Liu, D.T. Branson, C. Yang, J.S. Dai, R. Kang, Variable-gain control for continuum robots based on velocity sensitivity, Mech. Mach. Theory. 168 (2022) 104618.
[36] J.S. Dai, D. Wang, Geometric analysis and synthesis of the metamorphic robotic hand, J. Mech. Des. 129 (11) (2006) 1191–1197.
[37] J.A. Saglia, J.S. Dai, D.G. Caldwell, Geometry and kinematic analysis of a redundantly actuated parallel mechanism that eliminates singularities and improves dexterity, J. Mech. Des. 130 (12) (2008).
[38] R. Kang, D.T. Branson, T. Zheng, E. Guglielmino, D.G. Caldwell, Design, modeling and control of a pneumatically actuated manipulator inspired by biological continuum structures, Bioinspir. Biomim. 8 (3) (2013) 036008.
[39] J.H. Holland, Adaptation in Natural and Artificial systems: an Introductory Analysis With Applications to biology, control, and Artificial Intelligence, MIT press, 1992.
[40] B.J. Goodno, J.M. Gere, Mechanics of Materials, Cengage learning, 2020.
[41] D. Partridge, Network generalization differences quantified, Neur. Netw. 9 (2) (1996) 263–271.
[42] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 7553. Art. no.
[43] F.D. Foresee, M.T. Hagan, Gauss-Newton approximation to Bayesian learning, Proceed. Int. Conferen. Neur. Netw. 3 (1997) 1930–1935.
[44] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[45] L.G. Brown, A survey of image registration techniques, ACM. Comput. Surv. 24 (4) (1992) 325–376.
[46] R. Kelly, R. Salgado, PD control with computed feedforward of robot manipulators: a design procedure, IEEE Transact. Robot. Automat. 10 (4) (1994) 566–571.
[47] M. Zhuang, D.P. Atherton, Automatic tuning of optimum PID controllers, IEE Proceed. D (Contr. Theor. Applic.) 140 (3) (1993) 216–224.