

# Path Tracking of a Cable-Driven Snake Robot With a Two-Level Motion Planning Method

Lei Tang, Jie Huang, Li-Min Zhu , Xiangyang Zhu , and Guoying Gu 

**Abstract**—Due to high degree of freedoms (DOFs), motion planning of the cable-driven snake robot is challenging and usually tackled by the iterative Jacobian-based method. This iterative method adopts discontinuous velocity and tolerance to determine intermediate poses, which deteriorate the tracking performance of robots. In this paper, a two-level motion planning method is proposed to schedule the motion of cable-driven snake robot owning a serial–parallel hybrid structure. First, kinematics model of the robot among actuators, joints, and end effectors are introduced. Second, in the upper level, a jerk-bounded algorithm is employed to plan the position, velocity, and acceleration profiles of the end effector for a given path, whereas in the lower level, the joint angles and cable length changes are calculated by the inverse kinematics of the robot according to the position and velocity profiles of the end effector from the upper level. Therefore, the iterative process is eliminated by the presented method, which enhances the tracking performance and motion smoothness of the robot. Finally, simulations and experiments on a 24 DOFs snake robot prototype demonstrate the effectiveness of the presented method. According to the experimental results, the two-level method improves the repeatability and accuracy of the prototype by about 60% and 6%, respectively, compared with the iterative method.

**Index Terms**—Cable-driven, kinematics, motion planning, path tracking, snake robot.

## I. INTRODUCTION

DEVELOPMENTS of cable-driven snake robots have surged in recent years, for their advantages over traditional industrial robots such as flexibility and capability of conducting operations in confined environments [1]–[4]. Larger numbers

Manuscript received January 11, 2018; revised June 9, 2018, September 18, 2018, and January 14, 2019; accepted March 23, 2019. Date of publication April 9, 2019; date of current version June 14, 2019. Recommended by Technical Editor Y. Guo. This work was supported in part by the National Natural Science Foundation of China under Grants 51435010 and 51622506. (Corresponding author: Guoying Gu.)

L. Tang and G. Gu are with the State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the Soft Robotics and Biodesign Lab, Robotics Institute, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: 20090430414@sju.edu.cn; guguoying@sju.edu.cn).

J. Huang, L.-M. Zhu, and X. Zhu are with the State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: thk2dth@sju.edu.cn; zhulm@sju.edu.cn; mexyzhu@sju.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMECH.2019.2909758

of DOFs make them own the potential use in inspection and maintenance of intricate situations and instruments. For example, Dong *et al.* [1] designed a cable-driven continuum snake robot for on-wing inspection/repair of gas turbine engines. A snakelike robot was implemented to automatically charge for electric cars by Tesla [3]. Some cable-driven snake arm systems have been developed for inspection and maintenance in aircraft wings [5], nuclear reactors [2], [6], [7], and pipelines [8]. Cable-driven snake robots can also be used for minimally invasive surgical instruments [9]–[13], such as the famous da Vinci Surgical System [14].

The cable-driven snake robot has the ability to slither into narrow or confined space to do inspection owing to slender structure, high DOFs, and dexterity. However, the hybrid serial–parallel structure and the high DOFs make the inverse kinematics and motion planning of such robot quite difficult [15]–[18]. For high DOFs serial mechanism, the closed-form inverse kinematics is difficult, while the differential inverse kinematics is convenient.

Based on the differential inverse kinematics, the iterative Jacobian-based method [19]–[22] is generally utilized to control the motion of cable-driven snake robots [23]–[26]. This method iteratively generates the displacements of actuators at next moment. By repeating this process until the difference between the current and desired positions of the robot is less than the given tolerance, the robot ceases or moves to the next target point. The tolerance leads to tracking errors of the robot. Moreover, as addressed in [21], the position of the end effector is directly interpolated, which may give rise to errors and velocity fluctuations that deteriorate the tracking accuracy and make tasks difficult when the end effector of the robot is equipped with cameras, grippers, or laser cutters.

In order to make the end effector keep track of desired path accurately, many efforts have been made [27]–[31]. In general, feedforward control is usually used to compensate the friction of robot joints [27]. Besides, feedback control is also utilized to improve the accuracy of robot. However, with the increase of the number of axes, it becomes less effective for reducing tracking errors by feedback control of individual axis [28]. Thus, the cross-coupled control is employed for multi-axes robot [32]. As discussed in [28], with the increase of the number of axes, this method can be very complicated. A more commonly used and easy way is to plan the velocity of the end effector of the robot. To reduce the impact on the hardware, the interpolation method is generally adopted to ensure that the position, velocity, and acceleration profiles of the desired trajectory are smooth. Tang and Robert [28] discussed the selection of the above approaches according to the requirements of precision and travel

speed range. If high contour precision and high speed of end effector are required, the cross-coupled control can be used. If the speed is low, the interpolation and feedback control are preferred. It should be noted that, although OC robotics, a General Electric Aviation company, has reported a commercial cable-driven snake robot, it generally moves at a low velocity without providing any report of trajectory planning method. In the literature, there is still rare study to handle the velocity discontinuity of cable-driven snake robots.

In summary, motion planning for the cable-driven snake robot is challenging due to the high DOFs and multi-level relationships among the motors, cables, joints, and end effector. The commonly used iterative Jacobian-based method [19]–[23] generally results in the problem of velocity discontinuity. Although some interpolation methods [21] have been recently employed to plan the position profiles of some robots, trajectory planning for the cable-driven snake robot to improve its tracking accuracy and motion smoothness has not been achieved. Therefore, this work is motivated and contributed to develop a two-level motion planning method for the accurate path tracking of the cable-driven snake robot with experimental validation. In the upper level, the position, velocity, and acceleration profiles of the end effector are planned ahead with a jerk-bounded algorithm. Thus, the problem of velocity fluctuation by the traditional iterative Jacobian-based method can be improved. However, in the lower level, the motor control inputs and joint angles are calculated by the inverse kinematics of the cable-driven snake robot according to the position and velocity profiles from the upper level. Therefore, the tolerance which is essential in the iterative method is unnecessary, which further improves accuracy of the robot. Moreover, the velocity and acceleration profiles of the end effector, which make the motion of the robot smoother, are taken into consideration.

The rest of the paper is organized as follows. Section II briefly generalizes the design and kinematics of the cable-driven snake robot. The iterative Jacobian-based motion planning method and problems of this method are presented in Section III. To solve these problems, a two-level planning method is developed. The velocity and acceleration profiles of the robot are planned in advance with the jerk-bounded algorithm, which is also introduced in Section III. Simulations of these two methods are implemented on the cable-driven snake robot for tracking square paths. The simulation results are presented in Section IV, in which the tracking accuracy and motion smoothness are discussed. Section V describes the prototype platform and control system of the snake robot. Experiments are conducted on that platform with the help of a laser motion tracker. Section VI concludes this paper.

## II. CABLE-DRIVEN SNAKE ROBOT

The cable-driven snake robot system is briefly presented here. For detailed information, the reader may refer to [15]. A two-step inverse kinematics of the cable-driven snake robot is developed to lay the foundation for its motion planning.

### A. Overview of the Robot System

The cable-driven snake robot presented in this paper has 24 DOFs. The prototype in Fig. 1(a) is composed of the robotic

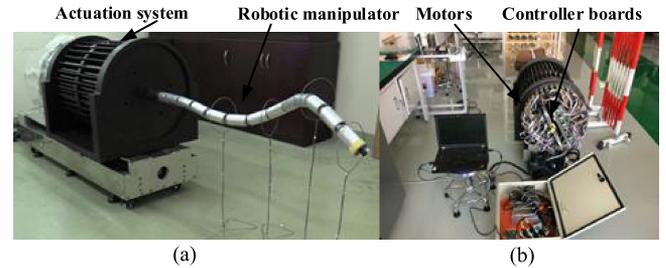


Fig. 1. Prototype of the developed cable-driven snake robot: (a) Front view. (b) Rear view.

manipulator and external actuation system. The manipulator with a length of 1235 mm and a diameter of 45 mm has 12 sections, and consists of joints, links, and actuation cables. All the sections are modular and equipped with two perpendicular intersected revolution joints. To reduce weight, all the components have central holes that allow electrical wires to go through. As shown in Fig. 1(b), the manipulator is driven by motors through cables. The motors and control system are at the rear of the robot. Thirty-six motors (Maxon EC-MAX30) and 36 EPOS2 cards are integrated into the actuation mechanism of the robot, which makes the control desk of the robot quite compact. The power distribution cabinet supplies electricity for all components.

### B. Kinematics of the Cable-Driven Snake Robot

The cable-driven snake robot consists of driven cables, joints, and end effector. The lengths of cables, rotation angles, and the position and orientation of the end effector are notified as actuator space, joint space, and task space, respectively. The product of exponentials (POE) formula is utilized to establish the kinematics among above spaces. Forward and inverse kinematics of the snake robot are presented below.

#### • Forward kinematics

The forward kinematics is to construct cascaded models from the actuator space to joint space and joint space to task space. Since the model from actuator space to joint space is unnecessary for the development of two-level method, we only present the model from joint space to task space. The modular design of the robot ensures that each section has the same structure. Thus, the coordinate frames can be set in the same way as shown in Fig. 2. Each section can be viewed as a parallel mechanism. In order to describe the relationship between adjacent sections, local POE [33] is utilized. The spatial frame is aligned with the lower platform of the section. The twists of the joints in this section are listed in Table I.

TY and TX denote the twists of rotation about  $Y_{i2}$  and  $X_{i1}$ , respectively. TI is the twist of initial transformation. The transformation matrix can be determined by (1), in which  $\alpha_i$  and  $\beta_i$  are the angles rotated about axes  $Y_{i2}$  and  $X_{i1}$ , respectively. The initial distance between the lower and higher platforms is  $|OO_1| = 2h$ . All the links are designed modular and they have the same length, denoted by  $L$ . The number of sections is  $n$ .

$${}_{i-1}T = e^{\hat{S}_{y,i} \alpha_i} e^{\hat{S}_{x,i} \beta_i} e^{\hat{S}_{st,i}}, i = 2, 3, \dots, n \quad (1)$$

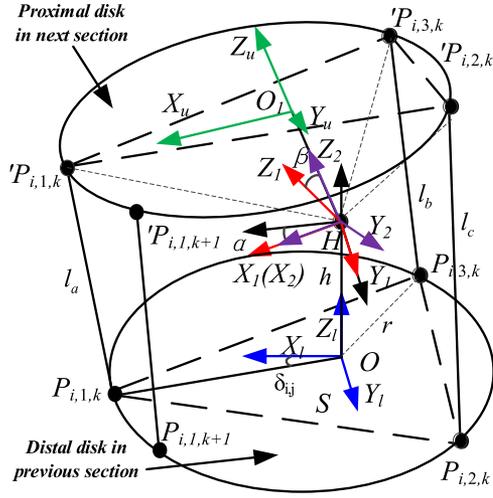


Fig. 2. Coordinate frames establishment of the snake robot.

TABLE I  
PARAMETERS OF LOCAL POE

Description	Symbol	Value
TY in local frame	$\xi_{y,i}^{i-1}$	$(0,1,0,0,0,0)^T$
TX in local frame	$\xi_{x,i}^{i-1}$	$(1,0,0,0,0,0)^T$
TI in local frame	$\xi_{st,i}^{i-1}$	$(0,0,0,0,0,2h+L)^T$

TABLE II  
PARAMETERS OF GLOBAL POE

Description	Symbol	Value
TY in world frame	$\xi_{y,i}^1$	$(0,1,0, -(i-1)(2h+L), 0, 0)^T$
TX in world frame	$\xi_{x,i}^{i-1}$	$(1,0,0,0, (i-1)(2h+L), 0)^T$
TI in world frame	$\xi_{st,i}^{i-1}$	$(0,0,0,0,0, (n-1)(2h+L))^T$

The global POE [33] is utilized to establish the relationship between the joint space and task space with the world frame coincided with the lower platform of the first section. Thus, the twists of the  $i$ th section can be denoted in Table II. With the above parameters, the POE formula from the first joint to the end effector of the robot is (2).

$${}^n T_2 = e^{\hat{\xi}_{y1}^1 \alpha_1} e^{\hat{\xi}_{x1}^1 \beta_1} \dots e^{\hat{\xi}_{yi}^1 \alpha_i} e^{\hat{\xi}_{xi}^1 \beta_i} \dots e^{\hat{\xi}_{yn}^1 \alpha_n} e^{\hat{\xi}_{xn}^1 \beta_n} e^{\hat{\xi}_{st}^1} \quad (2)$$

#### • Inverse kinematics

The robot owns a hybrid structure: the 12 sections are connected in a serial manner, and each section is a parallel mechanism. Specifically, each section is a four-chains parallel mechanism consisting of three spherical–prismatic–spherical (SPS) chains and a universal joint chain (3-SPS–U). Thus, the inverse kinematics of the robot can be solved by two steps. In the first step, the differential inverse kinematics denotes the relationship between the infinitely small changes of the end effector and the joint angles. In the second step, with the known joint angles, the cable length changes (CLCS) can be solved analytically. The differential relationship between changes of the end effector and

the joint angles can be described as (3).

$$dx = J_h d\theta. \quad (3)$$

The position vector of the end effector is  $x$ , and the vector of the joint angles is  $\theta$ . Hence,  $J_h$  is the velocity Jacobian matrix of  $x$  with respect to  $\theta$ , and can be calculated as follows. At first, the spatial Jacobian of the end effector  $J_s$  is calculated. By taking partial derivative of (2) with respect to joint angles, the spatial Jacobian matrix can be obtained as (4), in which each column is the instantaneous twist obtained by adjoint transformation (Ad) of the initial twist which can be calculated as (5).

$$J_s(\theta) = [\xi'_{y1} \quad \xi'_{x1} \quad \dots \quad \xi'_{yi} \quad \xi'_{xi} \quad \dots \quad \xi'_{yn} \quad \xi'_{xn}]. \quad (4)$$

$$\xi'_{yi} = \text{Ad} \left( e^{\hat{\xi}_{y1}^1 \alpha_1} e^{\hat{\xi}_{x1}^1 \beta_1} \dots e^{\hat{\xi}_{yi-1}^1 \alpha_{i-1}} e^{\hat{\xi}_{xi-1}^1 \beta_{i-1}} \right) \xi_{yi}^1$$

$$\xi'_{xi} = \text{Ad} \left( e^{\hat{\xi}_{y1}^1 \alpha_1} e^{\hat{\xi}_{x1}^1 \beta_1} \dots e^{\hat{\xi}_{xi-1}^1 \beta_{i-1}} e^{\hat{\xi}_{yi}^1 \alpha_i} \right) \xi_{xi}^1. \quad (5)$$

Ad in (5) is the adjoint transformation that is usually acted on a twist  $\xi$  by a homogeneous transformation matrix  $g$ . For a  $g$  expressed as in (6), the adjoint transformation can be denoted by (7).

$$g = \begin{bmatrix} R & b \\ 0^T & 1 \end{bmatrix} \quad (6)$$

$$\text{Ad}(g)\xi = \begin{bmatrix} R & 0 \\ b^{\wedge} R & R \end{bmatrix} \xi \quad (7)$$

where  $R$  and  $b$  are the rotation matrix and translation vector of  $g$ , respectively.

However, the spatial Jacobian is different from the velocity Jacobian. The position of the end effector in the spatial world frame is denoted as  $P_{st}$ . The spatial velocity is  $V_{st}^s = [(\omega_{st}^s)^T (-\omega_{st}^s \times P_{st} + \dot{P}_{st})^T]^T$ , and the velocity of the end effector is  $V_{st}^h = [(\omega_{st}^s)^T (\dot{P}_{st})^T]^T$ . The translation velocity of the spatial velocity has been coupled by the rotation velocity. Therefore, the relationship between them is expressed as (8). Thus, the end effector velocity Jacobian can be computed by (9).

$$V_{st}^s = \begin{bmatrix} I_{3 \times 3} & \hat{P}_{st} \\ 0 & I_{3 \times 3} \end{bmatrix} V_{st}^h. \quad (8)$$

$$J_h = \begin{bmatrix} I_{3 \times 3} & -\hat{P}_{st} \\ 0 & I_{3 \times 3} \end{bmatrix} J_s. \quad (9)$$

The differential inverse kinematics model between the position of the end effector and the joint angles can be calculated by (10) by combining (3)–(9).

$$d\theta = J_h^{-1} dx. \quad (10)$$

However, inverse solution of (10) may have the problem of singularity. To avoid singularity, the damped least square method [34] is utilized, which can be represented by (11)

$$d\theta = W J_h^T (J_h W J_h^T + \lambda I_{6 \times 6})^{-1} dx \quad (11)$$

where  $W$  is the weigh matrix, and is set as the identity matrix of size 24.  $\lambda$  is a positive damping constant with the value of 1.

In the second step, the relationships between changes of the joint angles and lengths of cables should be calculated. The three cables driving the  $k$ th (" $k = 1, 2, \dots, n$ ") section are distributed circumferentially with an interval of 120 degrees. The angle between the  $j$ th (" $j = 1, 2, \dots, 3$ ") cable and the  $X$  axis of the world frame of the  $i$ th (" $i = 1, 2, \dots, k$ ") section is  $\delta_{i,j}$ . Thus,  $\delta_{i,j+1} = \delta_{i,j} + 2\pi/3$ . All the connection points between the cables and the disks are on a circle with a diameter of  $r$ . With the above definitions, the coordinates of the connect points between the  $j$ th cable and the distal disk of  $i$ th section is  $P_{i,j} = (r \cos \delta_{i,j}, r \sin \delta_{i,j}, 0, 1)^T$ . The corresponding point  $'P_{i,j}$  is the connection point between  $j$ th cable and the proximal disk of the next section with its coordinate being  $'P_{i,j} = {}^{i-1}T * P_{i,j}$ . Therefore, the length of the cables between the upper and lower disks is  $L_{i,j} = \|\mathit{'P}_{i,j} - P_{i,j}\|$ . Given all the joint angles  $\theta(t) = (\alpha_1, \beta_1, \dots, \alpha_i, \beta_i, \dots, \alpha_n, \beta_n)^T$ , the length of the  $j$ th driven cable of the  $k$ th section can be determined by (12).

$$L_{k,j}(\theta(t)) = \sum_{i=1}^k |P_{i,j} - \mathit{'P}_{i,j}| + kL. \quad (12)$$

The lengths of cables can be represented by the joint angles in a closed-form way of (12). Therefore, with the incremental and current joint angles ( $\Delta\theta$  and  $\theta(t)$ ), the value of the joint angles of the next moment  $\theta(t+1)$  can be updated.  $\Delta$  represents the subtraction between the related values at the moments of  $t+1$  and  $t$ . Given the joint angles of two adjacent moments, the change of cable length  $\Delta L_{k,j}$  can be determined by (13).

$$\Delta L_{k,j} = L_{k,j}(\theta(t+1)) - L_{k,j}(\theta(t)). \quad (13)$$

### III. TWO-LEVEL MOTION PLANNING METHOD OF CABLE-DRIVEN SNAKE ROBOT

The iterative Jacobian method is commonly used to plan the motion of the snake robots. Although it is convenient to use, its iterative process generally makes the velocity of the end effector discontinuous, which impacts the whole system, including the joints, cables, and motors. Moreover, the tolerance in the process deteriorates the tracking accuracy of the robot. Thus, a two-level motion planning method is developed to enhance the accuracy and smoothness of the motion of the cable-driven snake robot. In this section, first, the iterative Jacobian method is introduced. Second, the problems of this method are discussed. Third, the two-level motion planning method is presented.

#### A. Iterative Jacobian-Based Motion Planning Method

For a defined path, the Jacobian-based iterative method is a common and convenient method to generate motions for path-tracking tasks. As for the cable-driven snake robot, the control inputs of cables and motors should also be taken into consideration along with those of the joints. The detailed steps are as follows [35]:

*Inputs:* Current joint angles  $\theta_c$ , target positions  $P_t$ , and orientations  $R_t$ , tolerance of the positions and orientations  $SP$  and  $SR$ , control period  $T$ .

*Outputs:* Changes of driven cable lengths and control inputs of the motors.

*Step 1:* With the current joint angles  $\theta_c$ , the current position  $P_c$  and orientation  $R_c$  of the end effector are calculated according to (2).

*Step 2:* Calculate the position error  $dp = P_t - P_c$  and the orientation error  $dr = rot^{-1}(\mu, \phi)$ , in which  $rot(\mu, \phi) = R_t(R_c)^T$ , and  $\mu$  and  $\phi$  are the axis and angle of the orientation difference between  $R_c$  and  $R_t$ .

*Step 3:* If  $dp \leq SP$  and  $dr \leq SR$ , go to step 1; else, go to step 5.

*Step 4:* If the current target point is the last one, exit the loop; else, go to step 9.

*Step 5:* Calculate the expected velocity  $\dot{x} = [\omega, v]^T$ . The linear and angular velocities are  $v = k_1 * (dp/||dp||)$  and  $\omega = k_2 * rot^{-1}(\mu, \phi)$ , respectively.  $k_1$  and  $k_2$  are constants with their value of 0.75 and 0.5, respectively.

*Step 6:* Calculate the Jacobian of the current moment according to (4)–(9), and the joint velocity can be determined according to (11).

*Step 7:* The joint angles of the next moment are updated by  $\theta(t+1) = \theta(t) + \Delta\theta * T$ , in which  $T$  is the control period.

*Step 8:* The CLCS can be obtained according to (12)–(13) with the known joint angles of the current and next moment.

*Step 9:* All the variables of the current moment are updated by the next moment and go to step 1.

In the iterative Jacobian-based method, whether the end effector reaches the desired point is determined by the position and orientation errors and the tolerance. If the errors are smaller than the tolerance, then the velocity of the end effector should be set as the initial value, and all the above process should be repeated until the end effector reaches the last desired point. Usually, the iterative Jacobian-based method only does some interpolation of the position, or uses constant velocity for the end effector as discussed in [21]. Such simple process gives rise to inaccuracy to the end effector, and makes the motion of the motors and joints unsmooth.

#### B. Problems of Iterative Jacobian-Based Method

There are mainly two problems of the iterative Jacobian-based method, namely motion inaccuracy and trajectory discontinuity. First, the tolerance cannot guarantee that the end effector reaches the desired point, leading to inaccuracy to the robot. Especially when the trajectory has many via points, the error of the last segment will be accumulated to the next one. For example, when the snake robot follows a square trajectory, the tracking errors types vary with the tolerance. When the tolerance is large, there are two undesirable situations: (1) The robot moves towards the next desired point while it has not reached the current desired point, as shown in Fig. 3(a). (2) The robot passes the desired point and move towards the next one. The actual trajectory under this condition is similar to Fig. 3(b). However, when the tolerance is too small, the robot oscillates around the desired point. Second, the robot approximates the target point in an iterative way, which makes the velocities of the end effector fluctuate. These velocity fluctuations will transmit to the joints,

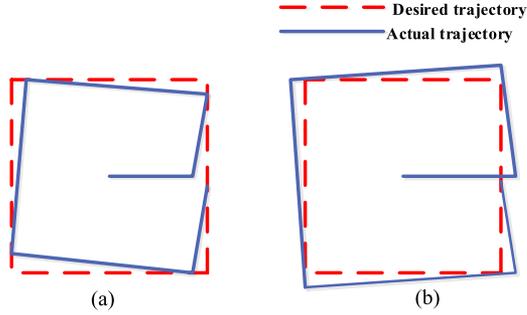


Fig. 3. Different error types caused by tolerance. (a) Tolerance is big. (b) Tolerance is small.

cables, and motors, which deteriorates the tracking performance and shortens the life of the snake robot.

### C. Two-Level Motion Planning Method

A two-level motion planning method is developed to eliminate the inaccuracy and velocity discontinuity generated by iterative Jacobian-based method. Usually, the snake robot has many actuator motors, and slithers into the complicate terrain at low speed. According to the principles presented by Tang and Robert [28], in this scenario of so many axes and low speed, the advance interpolation of the velocity of the end effector is more useful. Hence, in the upper level, the position, velocity, and acceleration profiles of the robot are determined with the jerk-bounded algorithm. In this algorithm, in order to follow the path smoothly, the rest-to-rest jerk-bounded trajectory is applied to each line of the path. It is worthy of mentioning that many trajectories, such as the jerk-bounded trajectory (i.e., the double  $S$  curve) [36] and jounce-bounded trajectory (jounce is the derivative of the jerk with respect to time) [37], can be used to plan the position of the end effector for the snake robot. Without losing generality, we employ the commonly used jerk-bounded trajectory in this work to verify the effectiveness of the developed two-level motion planning method.

*Remark 1:* It should be noted that the concept of two-level path tracking has been developed in the field of industry robots and machine tools, where an explicit inverse kinematics solution is generally required. However, this condition cannot be satisfied, if not impossible, for the cable-driven snake robot, which has a parallel–serial hybrid structure and hyper-redundant DOFs. Therefore, we first investigate the serial and parallel mechanism of the cable-driven snake robot and develop the kinematics model in three spaces, i.e., actuator space, joint space, and task space. In this sense, we can solve problem of the inverse kinematics from the task space to the actuator space by combining the differential and analytical kinematic analysis of the serial and parallel structure, respectively.

We use piecewise polynomial trajectories to follow the specified path in task space. The adopted trajectory is time-optimal in the sense that in any time instance the velocity, acceleration, or jerk attains its corresponding limit, as depicted in Fig. 4. The path in this paper is a square with a side length of 120 mm. Since the tangent at the vertex is discontinuous, the velocity at each vertex should be zero. In (14),  $j_{\max}$  is the specified maximum

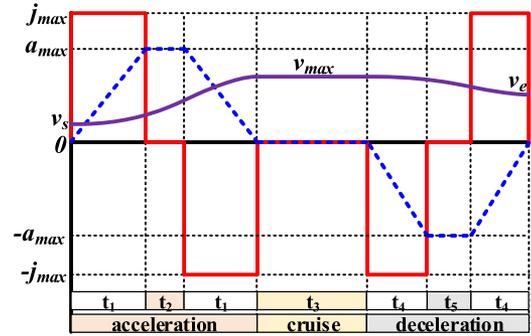


Fig. 4. Illustration of jerk-bounded trajectory.

jerk;  $t_1$ ,  $t_2$ , and  $t_3$  are the durations for non-zero jerk, for constant acceleration, and for constant velocity, respectively. The notation  $\| \cdot \|$  is the Euclidean distance in task space. By integrating the jerk, the acceleration  $a(t)$ , the velocity  $v(t)$ , and the displacement  $s(t)$  can be obtained. For a rest-to-rest trajectory, we have the boundary conditions as in (15).  $T = 4t_1 + 2t_2 + t_3$  is the total duration of the trajectory, which is also the control period. Buffers and control strategy are utilized to make sure that all the 36 motors move synchronously in a control period. The control strategy takes about 120 ms to check the state of the motors and buffers. In this work, we set the control period of 150 ms. According to the boundary conditions, the maximum acceleration  $a_{\max}$  and the maximum velocity  $v_{\max}$ , the time parameters  $t_1$ ,  $t_2$ , and  $t_3$  in (14) can be determined. The detailed discussion and formula derivation of the different trajectories can be found in [36]. After determining the time parameters of the trajectories for each line, the velocity at the  $j$ th control period can be obtained as  $v(jT)$ . Once the velocities at all control period are determined, the reference points can be determined as (16).

$$j(t) = \begin{cases} j_{\max}, & \leq t < t_1 \\ 0, & t_1 \leq t < t_1 + t_2 \\ -j_{\max}, & t_1 + t_2 \leq t < 2t_1 + t_2 \\ 0, & 2t_1 + t_2 \leq t < 2t_1 + t_2 + t_3 \\ -j_{\max}, & 2t_1 + t_2 + t_3 \leq t < 3t_1 + t_2 + t_3 \\ 0, & 3t_1 + t_2 + t_3 \leq t < 3t_1 + 2t_2 + t_3 \\ j_{\max}, & 3t_1 + 2t_2 + t_3 \leq t < 4t_1 + 2t_2 + t_3 \end{cases} \quad (14)$$

$$a(0) = a(T) = 0; v(0) = v_s; v(T) = v_e$$

$$s(T) - s(0) = \|P_{t+1} - P_t\| \quad (15)$$

$$P(j+1) = P(j) + v(jT) T \frac{P_{t+1} - P_t}{\|P_{t+1} - P_t\|}. \quad (16)$$

With all the positions and velocities of the reference points known, the established two-step inverse kinematics can be employed to determine the joint configurations and CLCS in each control period. The control inputs are given to the motors to drive the robot to follow the specified path. The flowchart of the two-level method is given in Fig. 5.

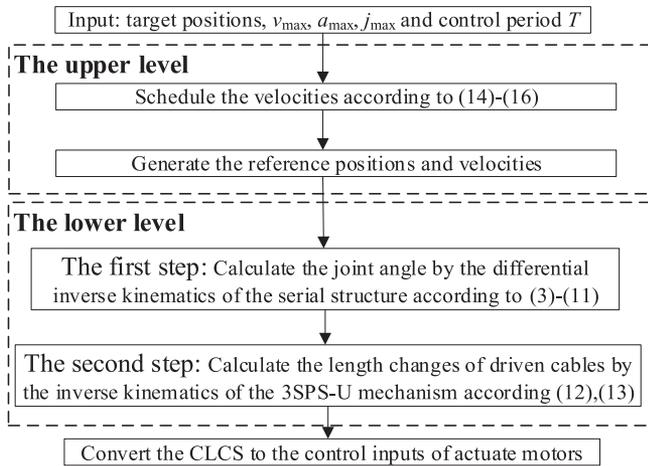


Fig. 5. Flowchart of the two-level motion planning method.

#### IV. SIMULATION OF THE ITERATIVE AND TWO-LEVEL METHODS

Simulations are performed on the MATLAB platform to illustrate the problems of the iterative method and the advantages of the developed two-level method. In these simulations, the end effector of the snake robot follows a square trajectory with a side length of 120 mm. The tracking errors, the velocities, and accelerations are calculated to evaluate the tracking performance and motion smoothness of the robot.

*Remark 2:* We should mention that we focus on the path-tracking planning for the cable-driven snake robot to improve its tracking accuracy and motion smoothness. The path is predefined in this work. Without losing generality, we use a  $120 \times 120$  mm square path to validate the effectiveness of the presented two-level method.

##### A. Simulation Trajectory Results of the Two Methods

First, the snake robot is simulated to track the given square trajectory by the iterative method. Fig. 6(a) shows a schematic description of the square trajectory tracking on the MATLAB platform. To show the influence of the tolerances in the iterative method, we perform two simulation trials with different tolerances (i.e., 2.5 and 0.6 mm) when the velocity profiles of the robot are the same (Fig. 6(b)). In Fig. 6(c) and (d), the simulation results indicate that the tracking performance of the robot relies on the values of the tolerances in the iterative method. Detailed tracking errors of these two simulations will be analyzed and discussed in the following sections.

Second, the simulated robot follows the same square trajectory by the developed two-level method. The maximum velocity and acceleration of the two-level method are set the same as those of the iterative method. According to (14)–(16), velocity, acceleration, jerk profiles, and distance between via points of the robot in task space are shown in Fig. 7. As shown in Fig. 7, we can see that the iterative method suffers from the problem of velocity fluctuation. With our two-level method, this velocity fluctuation can be well mitigated.

The actual and desired trajectories of the end effector are shown in Fig. 8(a). Fig. 8(b) shows the 3D view of the

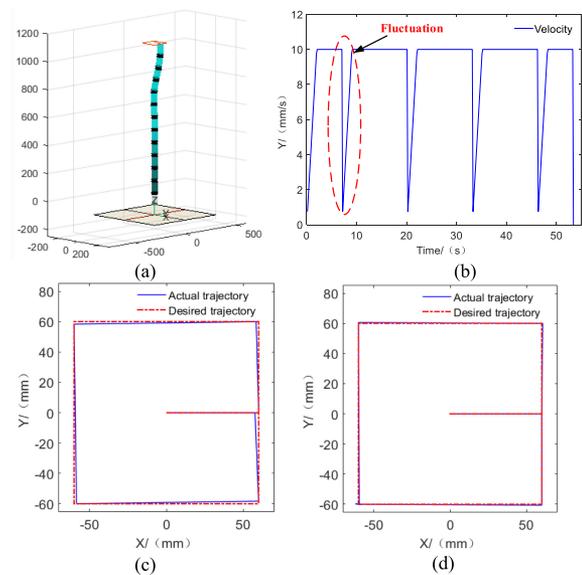


Fig. 6. Simulation tracking results of the iterative method. (a) Schematic of simulation process. (b) Velocity profiles with  $v_{\max} = 10$  mm/s,  $a_{\max} = 5$  mm/s<sup>2</sup>. (c) Trajectory results with tolerance of 2.5 mm. (d) Trajectory results with tolerance of 0.6 mm.

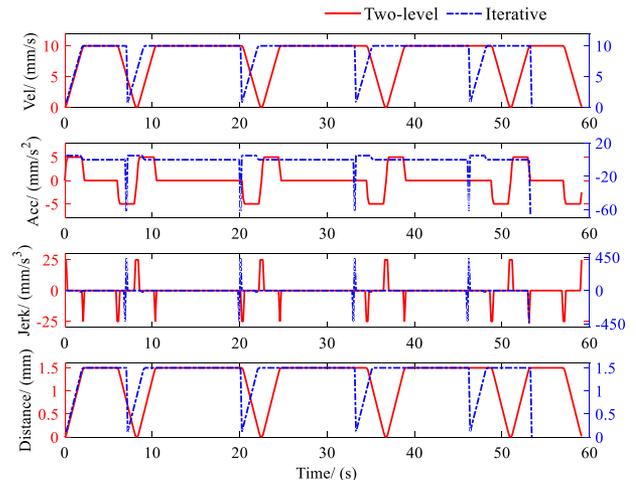


Fig. 7. Velocity, acceleration, jerk profiles, and distance between via points of the robot in task space of the iterative and two-level methods.

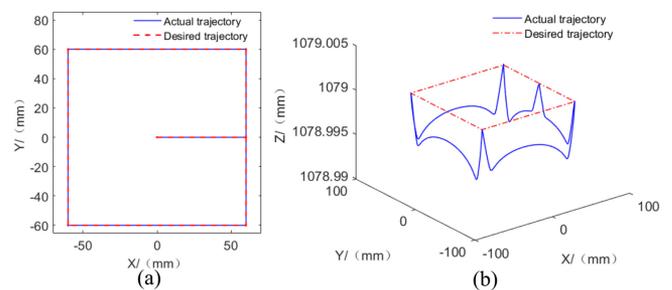
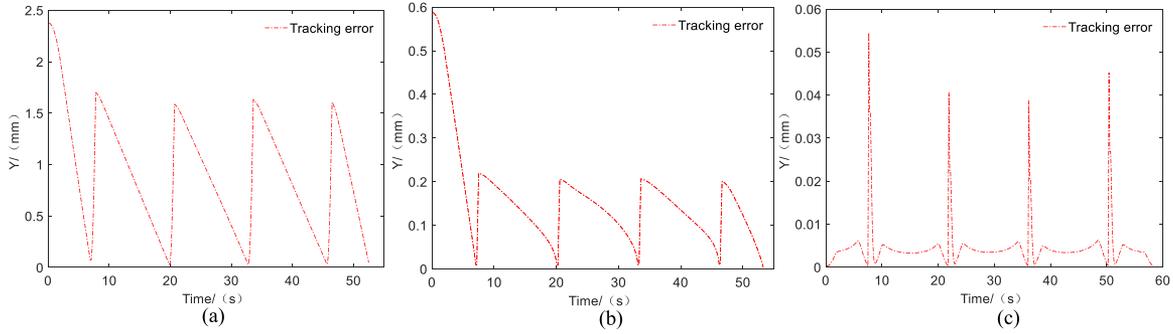


Fig. 8. Simulation tracking results by the presented two-level method: (a) Two-dimensional view. (b) Three-dimensional view.



**Fig. 9.** Simulation tracking errors during keeping track of the square. (a) and (b) Tracking errors by the iterative method with tolerance of 2.5 and 0.6 mm, respectively. (c) Tracking errors by the presented two-level method with maximum velocity and acceleration being the same with the iterative method.

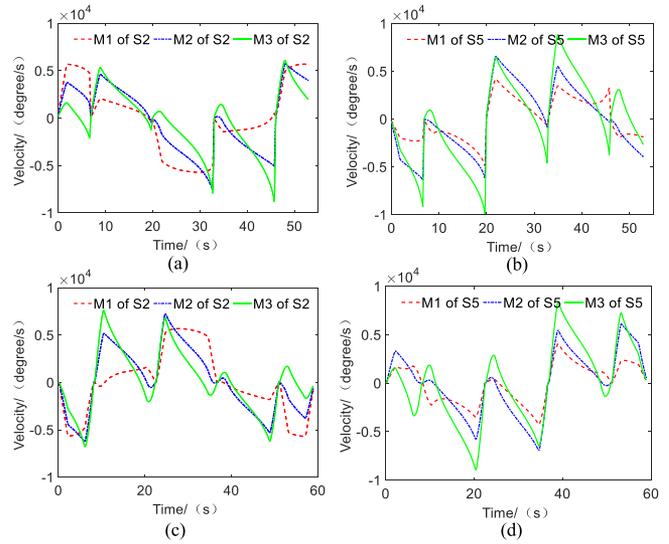
trajectories. The maximum error in Z direction is less than 0.005 mm, which is much less than errors in other directions. The tracking errors and the velocity and acceleration profiles of the motors, joints, and CLCS are calculated to analyze and compare accuracy and smoothness of the motions planned by the iterative and two-level methods.

### B. Tracking Errors of the Two Methods in Simulation

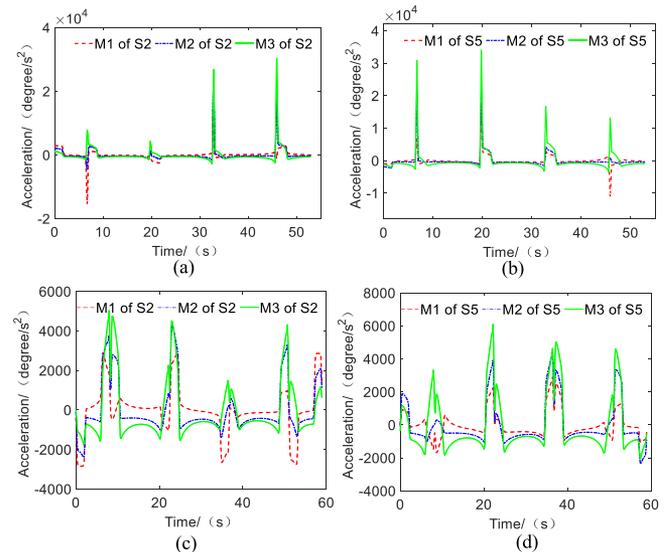
To evaluate the accuracy of the path planned by the iterative and presented two-level method, the tracking errors are calculated according to the method developed by Erkorkmaz *et al.* [38]. Fig. 9(a) and (b) shows the tracking errors by the iterative method with tolerances of 2.5 and 0.6 mm, respectively. There is tracking error at the beginning, because the robot goes from the origin to the first point of the square. Hence, the errors at the beginning are very similar to the tolerance. However, the developed two-level method nearly has no error at via points. Hence, it can move to the desired point accurately. It can be seen from Fig. 9 that tracking errors of the two-level method are significantly smaller than those of the iterative method.

### C. Velocities and Accelerations of Motors, Joints, and CLCS

The iterative method does not take the acceleration and deceleration into consideration, which makes the motion of robot discontinuous. The velocity and acceleration profiles of the motors, joints, and CLCS are the outputs of the methods. They should be plotted to investigate the motion smoothness of the iterative and two-level methods. Since each section has similar motion, only the motors, joints, and CLCS in sections 2 and 5 are plotted to compare the motion smoothness. The maximum velocity and acceleration of the two methods are the same to make the comparison convincing. Detailed parameters are  $v_{\max} = 10$  mm/s,  $a_{\max} = 5$  mm/s<sup>2</sup>, SP = 0.6 mm. There are three motors (M1, M2, and M3), two rotational joints (X and Y), and three cables (C1, C2, and C3) in a section of the robot. The velocity and acceleration profiles of the motors in sections 2 and 5 by the two methods are shown in Figs. 10 and 11. The maximum acceleration of the motors by the iterative method can reach as high as 34,000 degree/s<sup>2</sup>, whereas the maximum acceleration by the presented two-level method is less than 6000 degree/s<sup>2</sup>, which is only about 17.6% of the maximum



**Fig. 10.** Simulation motor velocity of sections 2 and 5 by the two methods. (a) and (b) Iterative method. (c) and (d) Two-level method.



**Fig. 11.** Simulation motor acceleration of sections 2 and 5 by the two methods. (a) and (b) Iterative method. (c) and (d) Two-level method.

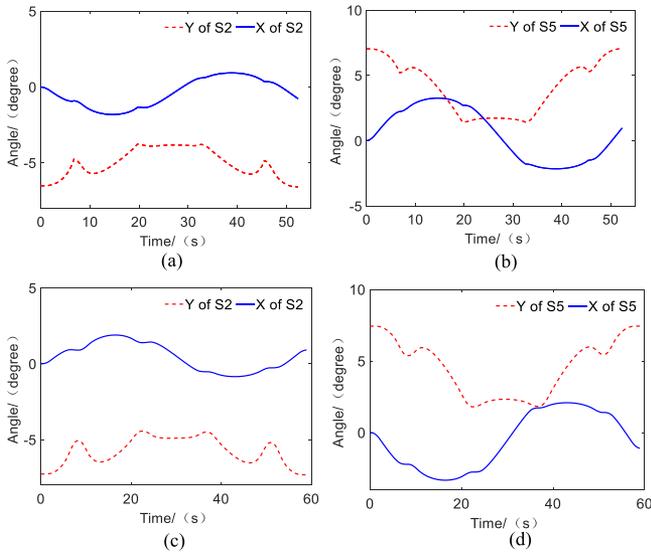


Fig. 12. Simulation joint angles of sections 2 and 5 by the two methods. (a) and (b) Iterative method. (c) and (d) Two-level method.

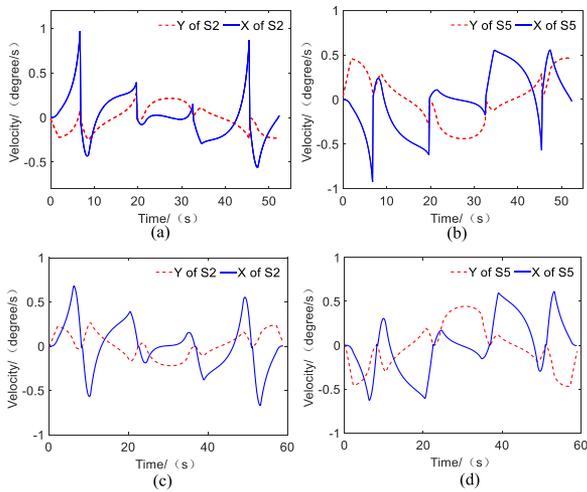


Fig. 13. Simulation joint velocity of sections 2 and 5 by the two methods. (a) and (b) Iterative method. (c) and (d) Two-level method.

acceleration by the iterative method. The position, velocity, and acceleration profiles of the joints in sections 2 and 5 by the two methods are shown in Figs. 12–14. From the figures, we can see that the maximum joint accelerations of the iterative and two-level methods are 3.3 and 0.7 degree/s<sup>2</sup>, respectively, which means the fluctuation of joint velocity is reduced by 78.7%. The velocity and acceleration of the CLCS in sections 2 and 5 by the two methods are shown in Figs. 15 and 16. The results show that the velocity and acceleration of the CLCS and motors change at the same rate. Therefore, the maximum acceleration of the CLCS by the presented two-level method is reduced to 17.6% of the maximum acceleration by the iterative method.

From the above results, we can see that the developed two-level method improves the tracking performance of cable-driven snake robot in terms of accuracy and motion smoothness compared with the iterative Jacobian-based method.

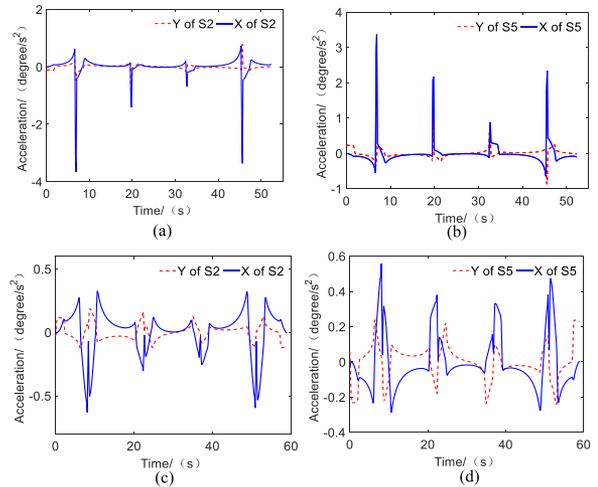


Fig. 14. Simulation joint acceleration of sections 2 and 5 by the two methods. (a) and (b) Iterative method. (c) and (d) Two-level method.

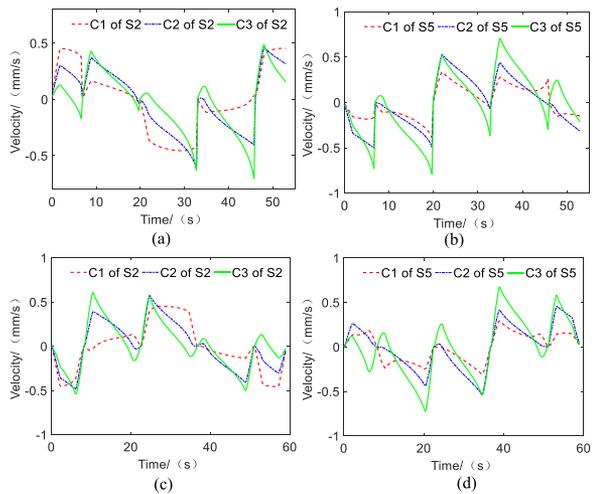


Fig. 15. Simulation CLCS velocity of sections 2 and 5 by the two methods. (a) and (b) Iterative method. (c) and (d) Two-level method.

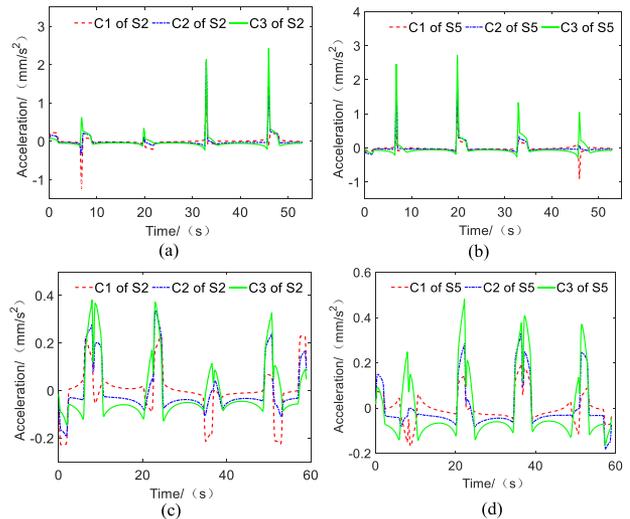


Fig. 16. Simulation CLCS acceleration of sections 2 and 5 by the two methods. (a) and (b) Iterative method. (c) and (d) Two-level method.

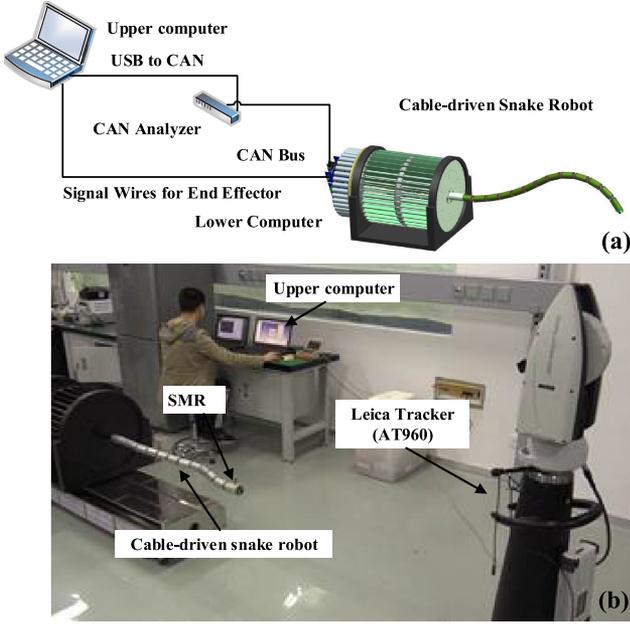


Fig. 17. Illustration of the experimental platform. (a) Diagram of the experimental platform. (b) Prototype of the experimental platform.

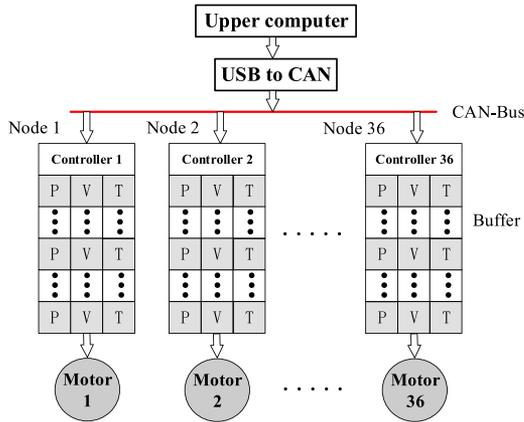


Fig. 18. Structure of the control system of the platform.

## V. EXPERIMENTAL VALIDATION

In this section, experiments are conducted on a prototype platform for further verification of the developed two-level method. The prototype is controlled to follow the same square trajectory as in the simulation. The motion data of the robot is recorded by the Leica Tracker Pilot (AT960) to analyze the performance of the presented method for the snake robot.

### A. Hardware of the Experimental Platform

The experimental platform is shown in Fig. 17(a), which consists of mechanical mechanism and control desk of the robot. The control desk mainly includes the upper computer and lower computer, which is composed of 36 EPOS2 cards. The cards are connected by the Controller Area Network (CAN) bus. The CAN analyzer (IXXAT) joins the CAN bus with the upper computer through USB to CAN portal. The signal wires of the

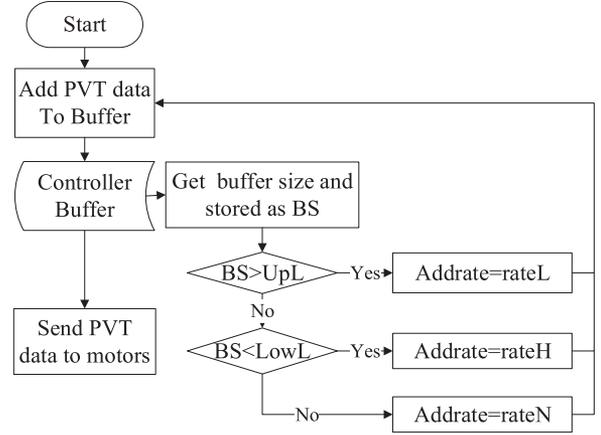


Fig. 19. Control strategy of the data and buffer.

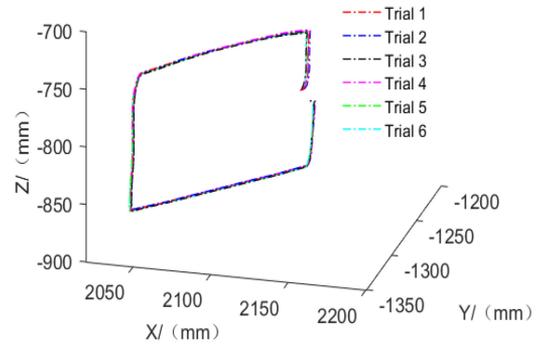


Fig. 20. Repeatability evaluation of different trial trajectories.

end effector also connect to the upper computer. The prototype of the experimental platform is shown in Fig. 17(b). All the computation and planning are done in the laptop (upper computer), and it distributes the control data to the EPOS2 cards (lower computer) which drive the motor to control the motion of the robot. The laser tracker (AT960) has a nominal accuracy of  $\pm 10 \mu\text{m} + 5 \mu\text{m}/\text{m}$ , and is implemented within two meters from the robot. A spherically mounted reflector (SMR) attached on the end effector of the robot is used to record trajectories of the end effector. In the experiments, the laser tracker real-time measures the displacements of the SMR at a rate of 15 points/s.

### B. Software of the Experimental Platform

The software of the control board is written in C++ with framework Qt. First, the control input data is generated by the simulation of the motion planning. Second, the control program calls the dynamic link library (DLL) of the MAXON to issue instructions to the controllers. The difficulty of controlling the motors is how to make 36 motors move cooperatively and coordinately. In this work, a buffer with first in first out (FIFO) structure is designed to improve the synchronization of motor operation. All the motors are controlled in the Interpolated Position Mode (IPM), which requires information including position, velocity profiles, and control period. The structure of the control system of the robot is shown in Fig. 18. The control inputs of the laptop are converted into CAN information by the CAN analyzer. There are 36 nodes on the CAN

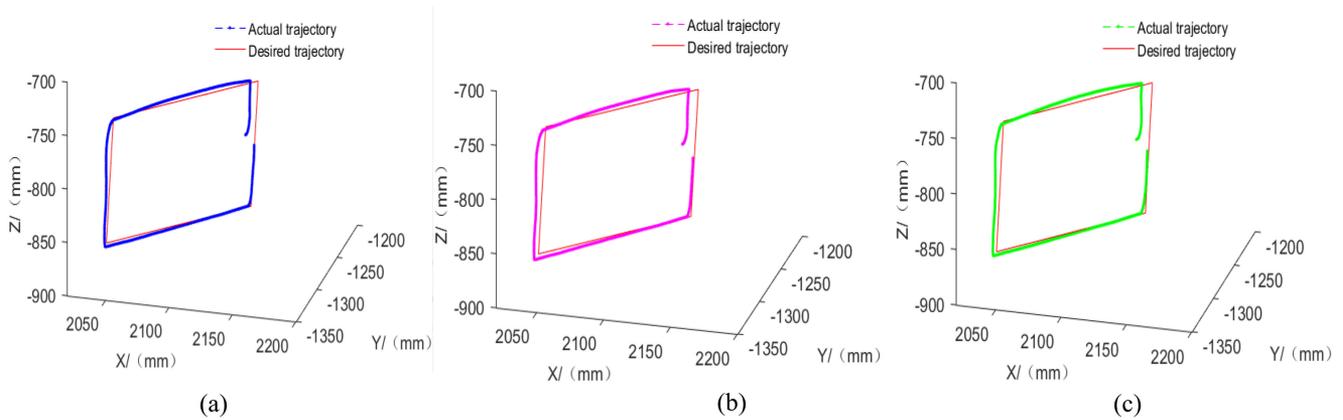


Fig. 21. Experiment trajectory results. (a), (b), and (c) are the actual and desired trajectories of trials 2, 4, and 6, respectively.

**TABLE III**  
REPEATABILITY PERFORMANCE COMPARISONS OF THE ITERATIVE METHOD (TRIALS 1 AND 2) AND OUR TWO-LEVEL METHOD WITH DIFFERENT ACCELERATION VALUES (TRIALS 3 AND 4 WITH  $5 \text{ mm/s}^2$ , AND TRIALS 5 AND 6 WITH  $2 \text{ mm/s}^2$ )

	Trail 1&2	Trail 3&4	Trail 5&6
RMSE (mm)	5.32	1.77	1.40
VVE ( $\text{mm}^2$ )	3.20	0.42	0.24

bus, and each node represents a controller. When the data-filling rate of the laptop and the data-consuming rate of the motors are inconsistent, the buffer will be gradually run out or filled up, which will cause error of the program. Hence, it is necessary to design an algorithm to dynamically adjust the size of buffer. The control strategy of the buffer is shown in Fig. 19.

### C. Results of the Experiments

The experimental results of the snake robot tracking square trajectories using the iterative and two-level methods including repeatability and accuracy are reported.

#### • Evaluation of repeatability

To evaluate the repeatability of the robot, the square trajectory has been followed six times, and half of them are chosen as the reference trajectory. The other trajectories have been compared with the corresponding reference one to compute the errors among corresponding points. Then, the root mean square error (RMSE) and variance value error (VVE) are utilized as indexes to quantify the repeatability of the robot. The results of the experiments recorded by the tracker are shown in Fig. 20. Trajectories of trials 1 and 2 are drawn by the iterative method with tolerance of 0.6 mm. Trajectories of trials 3, 4 and 5, 6 are drawn by the developed two-level method with maximum accelerations of 5 and  $2 \text{ mm/s}^2$ , respectively, and the maximum velocity of them are all 10 mm/s. A video of the experimental results of tracking a square path by the presented two-level method with maximum acceleration of  $5 \text{ mm/s}^2$  is available in the supplementary materials.

The RMSE and VVE are listed in Table III. It can be seen that the RMSE of the repeatability of the snake robot are 1–2 mm

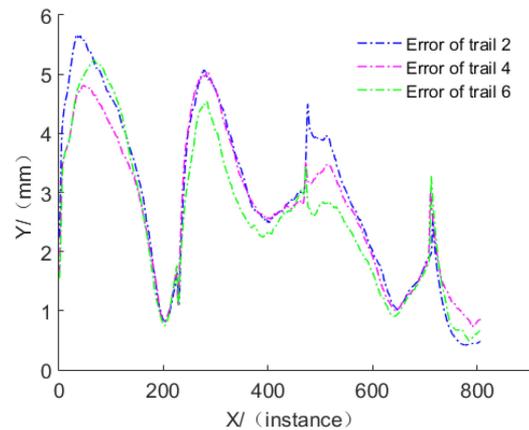


Fig. 22. Tracking errors of different trials.

by the two-level method, which are much smaller than that of the iterative method. The two-level method reduces the errors between corresponding points in different trials (Trials 1 and 2) to 33.3% of the errors of the iterative method (Trials 3 and 4), which means the presented method can significantly improve the repeatability of the robot compared with the iterative Jacobian-based method.

#### • Evaluation of accuracy

The accuracy of the robot should be represented by the errors between the recorded and desired trajectories. The desired trajectory is found by the following way. First, the normal vector of the plane of the square is fitting by the least square method. Second, the center and tangent vector of the square is acquired by optimization method. Trials 2, 4, and 6 are chosen to evaluate the tracking accuracy of the robot. The actual trajectories and the desired trajectories are shown in Fig. 21. The tracking errors between them are calculated according to the method discussed in [38]. Fig. 22 shows the detailed tracking errors along all points. To evaluate the accuracy of the robot with different methods, we use two performance indexes in terms of RMSE and VVE. In summary, the RMSE and VVE are listed in Table IV. The results indicate that RMSE and VVE of our two-level method (Trial 4) are reduced by about 6.5%, and 18.9%, respectively, compared with the traditional iterative Jacobian-based method

**TABLE IV**  
ACCURACY PERFORMANCE COMPARISONS OF THE ITERATIVE METHOD (TRIAL 2) AND OUR TWO-LEVEL METHOD WITH DIFFERENT ACCELERATION VALUES (TRIAL 4 WITH 5 mm/s<sup>2</sup>, AND TRIAL 6 WITH 2 mm/s<sup>2</sup>)

	Trail 2	Trail 4	Trail 6
RMSE (mm)	3.25	3.04	2.89
VVE (mm <sup>2</sup> )	2.16	1.75	1.68

(Trial 2). In addition, with decrease of acceleration (Trial 6), the RMSE and VVE can be further reduced.

## VI. CONCLUSION

In this paper, a two-level motion planning method is proposed for the path tracking of the cable-driven snake robot. In the upper level, the position, velocity, and acceleration profiles of the end effector are planned by a jerk-bounded algorithm. In this sense, the problem of the velocity fluctuations with the traditional iterative Jacobian-based method can be improved. While in the lower level, the control inputs for motor and joint angles are calculated by the inverse kinematics of the snake robot according to the position, velocity, and acceleration profiles from the upper level. The developed two-level method eliminates the iterative process during the motion planning which is essential in the traditional iterative method. Thus, it can significantly improve the motion smoothness and the tracking performance of the snake robots. For a prototype with 24 DOFs, simulation results show that the two-level method reduces the acceleration of the motors to only 17.6% of that produced by the iterative way. According to the experimental results, the repeatability and accuracy of the prototype are improved by about 60% and 6%, respectively, compared with the iterative method. The presented method achieves a high repeatability of 1–2 mm and a fair accuracy of 2–3 mm, validating the improvement of the tracking performance of the cable-driven snake robot. As future works, we will focus on dynamics modeling and closed-loop control design for the cable-driven snake robots to further improve their motion performance.

## REFERENCES

- [1] X. Dong, C. F. Quate, and C. Gerber, "Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines," *Robot. Comput.-Integr. Manuf.*, vol. 44, pp. 218–229, Apr. 2017.
- [2] R. O. Buckingham and A. C. Graham, "Dexterous manipulators for nuclear inspection and maintenance; Case study," in *Proc. 1st Int. Conf. Appl. Robot. Power Industry*, Montreal, QC, Canada, Oct. 2010, pp. 1–6.
- [3] E. Palermo, "Tesla unveils snakelike robot charger for electric cars," *LiveScience*. Available: <http://www.livescience.com/51791-tesla-electric-car-robot-charger.html>. Accessed: Apr. 17, 2017.
- [4] E. Kelasidi, P. Liljeback, K. Y. Petterson, and J. T. Gravidahl, "Innovation in underwater robots: Biologically inspired swimming snake robots," *IEEE Robot. Autom. Mag.*, vol. 23, no. 1, pp. 44–62, Mar. 2016.
- [5] R. Buckingham *et al.*, "Snake-arm robots: A new approach to aircraft assembly," *SAE Tech. Paper*, 2007, pp. 01–3870.
- [6] S. Ma, S. Hirose, and H. Yoshinada, "Development of a hyper-redundant multijoint manipulator for maintenance of nuclear reactors," *Adv. Robot.*, vol. 9, no. 3, pp. 281–300, Jan. 1994.
- [7] O. C. Robotics, "Snake-arm robots access the inaccessible," *Nucl. Technol. Int.*, vol. 1, pp. 92–94, 2008.
- [8] D. D. L. Mascarenas *et al.*, "A compliant mechanism for inspecting extremely confined spaces," *Smart Mater. Struct.*, vol. 26, no. 11, 2017, Art. no. 115028.
- [9] K. Xu and N. Simaan, "Actuation compensation for flexible surgical snake-like robots with redundant remote actuation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, FL, USA, May, 2006, pp. 4148–4154.
- [10] T. Ota *et al.*, "A novel highly articulated robotic surgical system for epicardial ablation," in *Proc. IEEE Conf. Eng. Medicine Biol. Soc.*, Vancouver, BC, Canada, Aug. 2008, pp. 250–253.
- [11] E. J. Lehr, E. Rodriguez, and W. R. Chitwood, "Applications of surgical robotics in cardiac surgery," in *Surgical Robotics*, J. Rosen, B. Hannaford, and R. M. Satava, Eds. Boston, MA, USA: Springer, 2011, pp. 701–722.
- [12] N. Simaan, R. Taylor, and P. Flint, "High dexterity snake-like robotic slaves for minimally invasive telesurgery of the upper airway," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI*, C. Barillot, D. R. Haynor, and P. Hellier, Eds., Berlin, Heidelberg: Springer, 2014, pp. 17–24.
- [13] Z. Li, L. Wu, H. Ren, and H. Yu, "Kinematic comparison of surgical tendon-driven manipulators and concentric tube manipulators," *Mech. Mach. Theory*, vol. 107, pp. 148–165, Jan. 2017.
- [14] D. V. Surgern, "Robotic-assisted surgery," [Online]. Available: <http://www.davincisurgery.com/>. Accessed: Oct. 31, 2017.
- [15] L. Tang, J. Wang, Y. Zheng, G. Gu, L. Zhu, and X. Zhu, "Design of a cable-driven hyper-redundant robot with experimental validation," *Int. J. Adv. Robot. Syst.*, vol. 14, no. 5, Sep. 2017, Art. no. 1729881417734458.
- [16] S. K. Mustafa, G. Yang, S. H. Yeo, W. Lin, and I. M. Chen, "Self-calibration of a biologically inspired 7 DOF cable-driven robotic arm," *IEEE/ASME Trans. Mechatron.*, vol. 13, no. 1, pp. 66–75, Feb. 2008.
- [17] S. K. Mustafa, W. B. Lim, G. Yang, S. H. Yeo, W. Lin, and S. K. Agrawal, "Cable-driven robots," in *Handbook of Manufacturing Engineering and Technology*, A. Y. C. Nee, Ed., London: Springer, 2015, pp. 2169–2228.
- [18] D. Lau, D. Oetomo, and S. K. Halgamuge, "Inverse dynamics of multilink cable-driven manipulators with the consideration of joint interaction forces and moments," *Robot. IEEE Trans.*, vol. 31, no. 2, pp. 479–488, Apr. 2015.
- [19] S. Cobos-Guzman, D. Palmer, and D. Axinte, "Kinematic model to control the end-effector of a continuum robot for multi-axis processing," *Robotica*, vol. 35, no. 1, pp. 224–240, Jan. 2017.
- [20] S. Chiaverini, G. Oriolo, and I. D. Walker, "Kinematically redundant manipulators," in *Springer Handbook of Robotics*, B. S. Prof and O. K. Prof, Eds., Berlin, Heidelberg: Springer, 2008, pp. 245–268.
- [21] T. Liu *et al.*, "Iterative jacobian-based inverse kinematics and open-loop control of an MRI-guided magnetically actuated steerable catheter system," *IEEE/ASME Trans. Mechatron.*, vol. 22, no. 4, pp. 1765–1776, Aug. 2017.
- [22] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Trans. Robot.*, vol. 22, no. 1, pp. 43–55, Feb. 2006.
- [23] N. Simaan, "Snake-like units using flexible backbones and actuation redundancy for enhanced miniaturization," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Barcelona, Spain, Apr. 2005, pp. 3012–3017.
- [24] I. D. Walker, H. Choset, and G. S. Chirikjian, "Snake-like and continuum robots," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Cham, Switzerland: Springer, 2016, pp. 481–498.
- [25] F. Trebuña, I. Virgala, M. Pástor, T. Lipták, and I. Miková, "An inspection of pipe by snake robot," *Int. J. Adv. Robot. Syst.*, vol. 13, no. 5, Jan. 2016, Art. no. 1729881416663668.
- [26] M. Li, D. Branson, J. S. Dai, and R. Kang, "Model-free control for continuum robots based on an adaptive Kalman filter," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 1, pp. 286–297, Feb. 2018.
- [27] H. Zhang and R. G. Landers, "Precision motion control methodology for complex contours," *J. Manuf. Sci. Eng.*, vol. 129, no. 6, pp. 1060–1068, Jun. 2007.
- [28] L. Tang and R. G. Landers, "Multi-axis contour control; the state of the art," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 6, pp. 1997–2010, Nov. 2013.
- [29] Q.-C. Pham, "A general, fast, and robust implementation of the time-optimal path parameterization algorithm," *IEEE Trans. Robot.*, vol. 30, no. 6, pp. 1533–1540, Dec. 2014.
- [30] M. Yuan, Z. Chen, B. Yao, and X. Zhu, "Time optimal contouring control of industrial biaxial gantry: A highly efficient analytical solution of trajectory planning," *IEEE/ASME Trans. Mechatron.*, vol. 22, no. 1, pp. 247–257, Feb. 2017.
- [31] J. Dong, P. M. Ferreira, and J. A. Stori, "Feed-rate optimization with jerk constraints for generating minimum-time trajectories," *Int. J. Mach. Tools Manuf.*, vol. 47, no. 12–13, pp. 1941–1955, Oct. 2007.

- [32] S. S. Yeh and P. L. Hsu, "Estimation of the contouring error vector for the cross-coupled control design," *IEEE/ASME Trans. Mechatron.*, vol. 7, no. 1, pp. 44–51, Mar. 2002.
- [33] I. M. Chen, G. Yang, C. T. Tan, and S. H. Yeo, "Local POE model for robot kinematic calibration," *Mech. Mach. Theory*, vol. 36, no. 11–12, pp. 1215–1239, Nov. 2001.
- [34] T. F. Chan and R. V. Dubey, "A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators," *IEEE Trans. Robot. Autom.*, vol. 11, no. 2, pp. 286–292, Apr. 1995.
- [35] W. Khalil and E. Dombre, *Modeling, Identification and Control of Robots*, Paris: Hermes Sciences, 2003.
- [36] X. Du, J. Huang, and L.-M. Zhu, "A complete S-shape feed rate scheduling approach for NURBS interpolator," *J. Comput. Des. Eng.*, vol. 2, no. 4, pp. 206–217, Oct. 2015.
- [37] W. Fan, X.-S. Gao, W. Yan, and C.-M. Yuan, "Interpolation of parametric CNC machining path under confined jounce," *Int. J. Adv. Manuf. Technol.*, vol. 62, no. 5–8, pp. 719–739, Sep. 2012.
- [38] K. Erkorkmaz, C. H. Yeung, and Y. Altintas, "Virtual CNC system. Part II. High speed contouring application," *Int. J. Mach. Tools Manuf.*, vol. 46, no. 10, pp. 1124–1138, 2006.



**Lei Tang** received the B.E. degree in mechanical design, manufacturing, and automation from the Hunan University, Changsha, China, in 2013. He is currently working toward the Ph.D. degree in mechanical engineering at the Shanghai Jiao Tong University, Shanghai, China.

His research interests include design, modeling, planning, and control of snake-like robots.



**Jie Huang** received the B.E. degree from the Wuhan University, Wuhan, China, in 2012. He is currently working towards the Ph.D. degree in mechanical engineering at Shanghai Jiao Tong University, Shanghai, China.

His research interests include modeling and control of feed drive systems, as well as trajectory planning in high-speed CNC machine tools.



**Li-Min Zhu** (M'12) received the B.E. degree (with honors) and the Ph.D. degree in mechanical engineering from Southeast University, Dhaka, Bangladesh, in 1994 and 1999, respectively.

From November 1999 to January 2002, he worked as a postdoctoral fellow with Huazhong University of Science and Technology, Wuhan, China. Since March 2002, he has been with the Shanghai Jiao Tong University, Shanghai, China, where he is currently the Cheung Kong

Chair Professor, Head of the Department of Mechanical Engineering and Vice Director of the State Key Laboratory of Mechanical System and Vibration. He has held the visitorship in the Monash University, Clayton, Australia (from September 1997 to May 1998) and The City University of Hong Kong, Kowloon Tong, Hong Kong (from December 2000 to March 2001). His research interests include (1) multi-axis CNC machining technology and machine tool, (2) coordinate metrology and measurement, and (3) control, sensing, and instrumentation for micro/nano manufacturing. He has published one monograph and more than 150 international journal papers.

Prof. Zhu was the recipient of the National Science Fund for Distinguished Young Scholars in 2013 and selected into the National High-level Personnel of Special Support Program in 2016. He has been an Associate Editor for the *IEEE Transactions on Automation Science and Engineering*. He is now a Technical Editor for the *IEEE/ASME Transactions on Mechatronics*, and Editorial Board Members of the *Proceedings of the Institution of Mechanical Engineer (IMechE)*, *Part B: Journal of Engineering Manufacture*, and *Chinese Journal of Mechanical Engineering*.



**Xiangyang Zhu** received the B.S. degree from the Department of Automatic Control Engineering, Nanjing Institute of Technology, Nanjing, China, in 1985, the M.Phil. degree in instrumentation engineering and the Ph.D. degree in automatic control engineering, both from the Southeast University, Nanjing, China, in 1989 and 1992, respectively.

From 1993 to 1994, he was a Postdoctoral Research Fellow with Huazhong University of Science and Technology, Wuhan, China. He joined the Department of Mechanical Engineering as an Associate Professor, Southeast University, Dhaka, Bangladesh, in 1995. Since June 2002, he has been with the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China, where he is currently a Changjiang Chair Professor and the Director of the Robotics Institute. His current research interests include robotic manipulation planning, human-machine interfacing, and biomechanics.

Dr. Zhu received the National Science Fund for Distinguished Young Scholars in 2005.



**Guoying Gu** (S'10–M'13) received the B.E. degree (with honors) in electronic science and technology, and the Ph.D. degree (with honors) in mechatronic engineering from the Shanghai Jiao Tong University, Shanghai, China, in 2006 and 2012, respectively.

Since October 2012, he was at the Shanghai Jiao Tong University, where he is currently appointed as a Professor of School of Mechanical Engineering. He was a Humboldt Postdoc Fellow with the University of Oldenburg, Germany.

He was a Visiting Scholar with the Massachusetts Institute of Technology, National University of Singapore, Singapore and Concordia University, Canada. His research interests include soft robotics, smart materials actuated systems, bioinspired robot design, and motion control. He is the author or co-author of over 80 publications, which have appeared in *Science Robotics*, *IEEE/ASME Transactions*, *Advanced Functional Materials*, *Soft Robotics*, etc., as book chapters and in conference proceedings.

Dr. Gu is the winner of multiple awards including Young Changjiang Scholar of the Ministry of Education, National Science Fund for Excellent Young Scholars, the first prize of natural science of Ministry of Education, Best Paper Award at the 2016 International Conference on Intelligent Robotics and Applications (ICIRA 2016) and 2011 IEEE International Conference on Information and Automation (IEEE-ICIA 2011). He is a member of the ASME. Now he serves as an Editorial Board Member of *International Journal of Advanced Robotic Systems*, and Topic Editor of *Frontiers in Robotics and AI*. He has also served as General Chair of 2017 International Symposium on Theory and Technology of Soft Robotics, and several international conferences/symposiums as Chair, Associate Editor, or Program Committee Member.