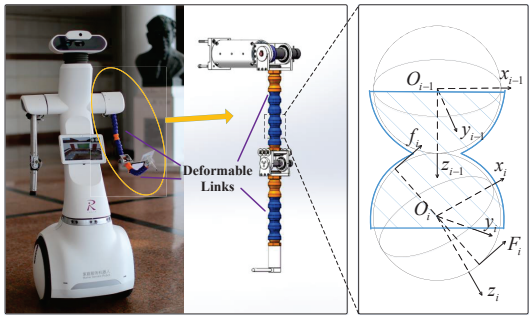
# A Hybrid Model and Kinematic-free Control Framework for a Low-cost Deformable Manipulator Using in Home Service

Gaofeng Li[1,2], Lei Sun[1,2], Shan Xu[1,2], Dezhen Song[3] and Jingtai Liu[1,2]

***Abstract*— Home service robot works in unstructured environments with various tasks, where a low-cost, dexterous, and intrinsically safe manipulator is important. Traditionally, a redundant manipulator with high degrees of freedom (DOFs) is employed. However, high DOFs bring problems like high cost, complex physical designing, and heavy weight. Here a novel low-cost deformable manipulator for home service robot, which is composed of rigid joints and deformable links, is introduced. The deformable manipulator can obtain relatively dexterous and extended workspace with fewer joints by bending its deformable links. Considering the typical hybrid behaviors caused by bending operation, a hybrid model and a practical kinematic-free control framework with no prior kinematic information are proposed to describe and control the deformable manipulator. The model and controller have been implemented on a 4-DOF deformable manipulator with two deformable links in simulation. The simulation results validate the proposed deformable manipulator and kinematic-free control.**




(a) (b)

Fig. 1. (a) A 4-DOF deformable manipulator with two deformable links. (b) The components of the deformable link can be simplified as a series of rigid spheres connected with each other.

## I. Introduction

### A. Background

Home service robots work in unstructured environments with various tasks. Hence a low-cost, dexterous and intrinsically safe manipulator is important for home service robots [1]. Traditionally, the links of manipulator are designed to be stiff to avoid kinematic error. However, stiff links result in fixed workspace and make it difficult to adapt to various tasks in home service. Moreover, rigid manipulators may cause some safety issues by collisions.

The traditional approach for this case is to design a redundant manipulator with high degrees of freedom (DOFs). However, high DOFs bring problems like complex physical designing, motion planning, and cost problem. Our manipulator is similar to continuum manipulators, such as the Soft Robotic Octopus Arm [2]–[4], the RobotinoXT [5]–[8] made by Festo-Didactic, the Concentric Tube Robot [9]–[12]. However, continuum manipulators also have the high cost problem due to their special materials or special actuators. As a result, they are not very suitable for home service.

The deformable manipulator proposed in this paper (as shown in Fig. 1) can obtain relatively more dexterous and extended workspace with fewer joints by bending its deformable links. However, frequent changes in links' shape bring difficulties to the model and control of the deformable manipulator. Hybrid systems are systems that display both continuous and discrete behavior [13]–[15]. The bending operation breaks up the continuous flow of the deformable manipulators, which is described by the manipulator's kinematic model, and leads to typical hybrid behaviors. The kinematic parameters experience a drastic change and become totally unknown. In our previous work [16]–[18], the kinematic parameters identification algorithms based on Denavit-Hartenber (DH) model and screw theory model are discussed. However, the reconstructed model can only describe the manipulator's performance between two adjacent bending operations. Moreover, the bending operation is frequent and it is relatively inefficient to run the identification algorithm every time.

Here a novel low-cost deformable manipulator, which is composed of rigid joints and deformable links, is introduced. The deformable links can be bent according to different tasks. One example of such material is the condenser pipe used in numerical lathe. It is very low-cost and easy to be obtained. Besides, the deformable links are compliant with impedance. Therefore, it is intrinsically safe and does not cause any damage to people and environments. Considering the typical hybrid behaviors, a hybrid model and a practical kinematic-free control framework with no prior kinematic information are proposed. The kinematic-free controller is based on the hypothesis that the Jacobian matrix is constant in a local region. It can be estimated by moving each actuator independently by an incremental amount and observing their effects on the robot's end-effector.

*This work is supported by National High-tech R&D Program of China (863 Program) under Grant 2012AA041403, National Natural Science Foundation of China (NSFC) under Grant No. 61375087, 61105096.

[1]Institute of Robotics and Automatic Information System, NanKai University, Tianjin, China, 300350. [2]Tianjin Key Laboratory of Intelligent Robotics, NanKai University, Tianjin, China, 300350. [3]Department of Computer Science and Engineering, Texas A&M University, College Station, Texas, United States, 77843.

Emails: {gaofengli, xu_shan}@mail.nankai.edu.cn, sunl@nankai.edu.cn, dzsong@cse.tamu.edu.

*Corresponding authors: liujt@nankai.edu.cn.

### B. Related Work

The deformable manipulator exhibits typical hybrid behaviors due to the effect of bending operations. There is still little related work in modelling a manipulator as a hybrid system. Chareyron *et al.* [19] model the contacts between the robots and their environments as discrete events that intertwine with the continuous dynamics, namely, a hybrid system. Ames *et al.* [20], [21] model the bipedal robot walking as a hybrid model with impulse effects when the robot's feet contacted with ground. Nevertheless, the distinction between our system and these hybrid systems is that the discrete events in our system need to be determined by ourselves, namely, when to trigger a bending operation need to be designed in our control framework.

Another group of related work is model-free or kinematic-free control methods. Kormushev *et al.* [22], [23] propose a radically new concept for controlling robots called Encoderless Robot Control (EnRoCo). EnRoCo uses an external camera to perceive the effects that the actuators have on the robot's motion and then uses learning algorithms to decide what actuation signals need to be sent to the actuators in order to achieve the desired robot motions. Another model-free controller is proposed by Yip *et al.* [24]. They use an optimal control strategy to generate the control signals and update the Jacobian matrix in each control period. Vikas *et al.* [25] also present an model-free, data-driven, reinforcement learning inspired approach. They represent the state transitions on a directed graph and then formulate the robot locomotion as a class of optimization problems on directed graphs, which is very close to our work.

However, the situation that a target is unreachable is not considered in all these model-free control methods. Thus these control methods cannot be applied directly to the deformable manipulator. In our control framework, the controller can tell whether a target is unreachable or not and trigger a bending operation without supervision.

## II. Problem Definition

The components of a deformable link can be simplified to a series of rigid spheres connected with each other (as shown in Fig. 1). Deformable links have the typical property: When the external torque is smaller than a threshold, they can be viewed as a rigid body. Their shapes are changed when the external torque is larger than the threshold.

Denote $\tau_i$, $i = 1, 2, \cdots, N$ as the external torque exerted to the $i$-th sphere, where $N$ is the number of spheres, $f_s$ and $f_d$ as the static and dynamic friction forces between the adjacent spheres, respectively, and $r$ as the radius of each sphere. $f_s$ is the force that locks the deformable link to hold a specific curve. $C_i$ is the frame attached to the $i$-th sphere and ${}^{i-1}_{i}T \in SE(3)$ is the transformation from frame $C_i$ to frame $C_{i-1}$. The aforementioned assumptions and properties can be formulated as following:

*Assumption 1:* The friction torque is always opposite to the external torque in direction. Also, we have $\tau_{s,max} > \tau_d$, where $\tau_{s,max} = f_{s,max} r$ is the maximum static friction torque and $\tau_d = f_d r$ is the dynamic friction torque.

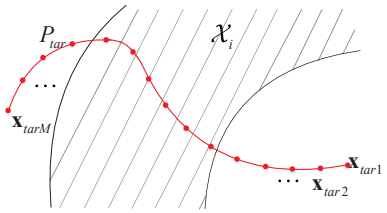



Fig. 2. A path $P_{tar}$ is given by a sequences of targets that generate a curve in 3-D space.

*Assumption 2:* The $i$-th sphere's motion is pure rotation under the action of external torque and the relative position expressed in frame $C_{i-1}$ is constant.

*Property 1:* If $\tau_i \leq \tau_{s,max}$ for all $i \in [1, N]$, ${}^{i-1}_{i}T$ is constant and there is no relative motion between adjacent spheres.

*Property 2:* If $\tau_i > \tau_{s,max}$ for some $i \in [1, N]$, the $i$-th sphere performs pure rotation and the resulting torque exerted to the $i$-th sphere is $\tau_i - \tau_d$.

Based on Assumptions 1, 2 and Properties 1, 2, the transformation ${}^{i-1}_{i}T$ can be formulated as the following:

$$ {}^{i-1}_{i}T = \begin{bmatrix} \boldsymbol{R}_i & \boldsymbol{r}_i \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{1} $$

where $\boldsymbol{R}_i = rotz(\alpha_i) roty(\beta_i) rotz(\gamma_i)$ is the rotation matrix and translation $\boldsymbol{r}_i = [0, 0, 2r]^T$ is constant. So the deformable link can be described by a function: $f(\cdots, \alpha_i, \beta_i, \gamma_i, \cdots)$, $i = 1, 2, \cdots, N$, where $\alpha_i$, $\beta_i$, $\gamma_i$ are Euler angles.

A deformable manipulator is a manipulator with several deformable links. For example, the deformable manipulator shown in Fig. 1(a) has two deformable links: one is the link between its shoulder and elbow and the other one is the link between its elbow and wrist.

An expected path is given by a sequence of targets that generate a curve in 3-D Euclidean space and should be reached by the end-effector. The path can be denoted by a queue: $P_{tar} = \{\mathbf{x}_{tar1}, \mathbf{x}_{tar2}, \cdots, \mathbf{x}_{tarM}\}$, where $M$ is the number of targets and $\mathbf{x}_{tarj}$, for $j = 1, 2, \cdots, M$, is one of the targets. In this context, the path is specified only in the task space and not in the time domain.

Let $\mathcal{X}_i$ denote the manipulator's workspace in a specific configuration. $P_{tar}$ is classified as a long path (as shown in Fig. 2) if it satisfies: (1) $P_{tar} \subseteq \bigcup_i \mathcal{X}_i$; (2) $\forall \mathcal{X}_i$, always $\exists \mathbf{x}_{tarj} \in P_{tar}$ such that $\mathbf{x}_{tarj} \notin \mathcal{X}_i$. A long path is difficult to be continuously tracked by traditional manipulators with stiff links due to joint limits, self collision, etc. However, the deformable manipulator provides a feasible scheme to track a long path with fewer joints by bending its deformable links. Nevertheless, the bending operation breaks up the continuous flow and leads to hybrid behaviors. We need to address the model and control problems in this paper.

*Problem Statement:* Given a long path $P_{tar} = \{\mathbf{x}_{tar1}, \mathbf{x}_{tar2}, \cdots, \mathbf{x}_{tarM}\}$, model and design a control policy for the deformable manipulator to generate a complete hybrid execution to track the path $P_{tar}$.

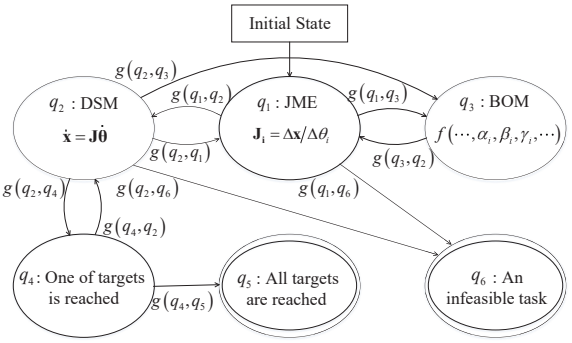



Fig. 3. The hybrid model of the deformable manipulator is represented by a directed graph.

## III. A Hybrid Model and Kinematic-free Control Framework for Deformable Manipulator

### A. A Hybrid Model Based on Kinematic-free Control

The bending operation leads to the manipulator's instantaneous change in kinematic model and makes the manipulator exhibit hybrid behaviors. Based on the kinematic-free control framework, the deformable manipulator is switched among three control modes, which are Jacobian Estimation Mode (JEM), Differential System Mode (DSM), and Bending Operation Mode (BOM), according to different conditions. In JEM, the system needs to estimate the Jacobian matrix. In DSM, the system is governed by its kinematic model. Therefore, the state of the deformable manipulator is changed continuously. In BOM, the system triggers a bending operation to obtain a better workspace. The goal is to control the deformable manipulator to accomplish a tracking task. It is straight forward to define a hybrid model, as shown in Fig. 3, to describe this process: $\mathcal{H} = (\mathcal{Q}, \mathcal{X}, f, \mathcal{I}, \mathcal{E}, \mathcal{G}, \mathcal{R})$, where

- $\mathcal{Q} = \{q_1, q_2, q_3, q_4, q_5, q_6\}$ is a set of discrete states. $q_1$, $q_2$ and $q_3$ denote that the system is in JEM, DSM and BOM, respectively. $q_4$ denotes that one target is reached by the end-effector. $q_5$ denotes that the tracking task is accomplished. $q_6$ denotes that the given path is an infeasible task for this deformable manipulator.
- $\mathcal{X} = \bigcup_i \mathcal{X}_i \subseteq \mathbb{R}^3$ is a set of continuous states. In this context, the end-effector's position is chosen as the continuous states.
- $f(q_2, x) : \mathcal{Q} \times \mathcal{X} \rightarrow \mathbb{R}^3$ describes the system's continuous flow in DSM.
- $\mathcal{I} \subseteq \mathcal{Q} \times \mathcal{X}$ is a set of initial states; Starting from a blank state, the system firstly estimates the Jacobian matrix, namely, $\mathcal{I} = \{q_1\} \times \mathcal{X}$.
- $\mathcal{E} = \{(q_i, q_j)\} \subseteq \mathcal{Q} \times \mathcal{Q}$, $i, j \in [1, 6]$ is a set of edges; There are 10 edges in this graph, as shown in Fig. 3.
- $\mathcal{G} = \{g(q_i, q_j)\}$, $i, j \in [1, 6]$ is a set of transition conditions corresponding to different edges; Obviously, $g(q_i, \cdot)$ is disjoint for all $i$.
- $\mathcal{R} = \{r(q_i, q_j, \mathbf{x})\}$, $i, j \in [1, 6]$ is a set of reset maps corresponding to different edges. $r(q_i, q_j, \mathbf{x}) = \mathbf{x}$ for all $i, j$ except for $i = 3 \quad and \quad j = 1$. $r(q_3, q_1, \mathbf{x}) = \mathbf{x}'$ because bending operation leads to instantaneous change for continuous state. $\mathbf{x}'$ can be measured.

The goal is to design $g(q_i, q_j)$, $i, j \in [1, 6]$ for each edge such that the system can reach $q_5$ or $q_6$ finally and this work is done by the kinematic-free control framework.

### B. Kinematic-free Control Framework

Without loss of generality, common notations and assumptions are defined here:

*Assumption 3:* The position of manipulator's base frame $\mathbf{r}_0$, position of the third joint $\mathbf{r}_3$, and position of the fourth joint $\mathbf{r}_4$, can be measured by a camera. The maximum length of two deformable links are known and denoted by $L_{max1}$ and $L_{max2}$, respectively.

This assumption is to judge whether a path tracking is an infeasible task or not. It is not used in the kinematic-free control process. For simiplificati, a strong condition is defined for the deformable manipulator shown in Fig. 1:

$$\begin{aligned} &g(q_1, q_6) = g(q_2, q_6) \\ &= \{\mathcal{Q} \times \mathcal{X} \,|\, \|\mathbf{x}_{tarj} - \mathbf{r}_0\| > L_{\max 1} + L_{\max 2}\}, \end{aligned} \tag{2}$$

which means that a target is too far away from the base. Whenever this condition is satisfied, the system switches to $q_6$ and declares that "This path is infeasible to be tracked".

*Assumption 4:* The position of the end-effector $\mathbf{x_{cur}}$ can be measured and the measurement is accurate enough.

A manipulator can be described by a Jacobian model: $\dot{\mathbf{x}} = \mathbf{J}\dot{\boldsymbol{\theta}}$. Analysing the elements of $\mathbf{J}$, it is deterministic and unknown in a specific position. Fortunately, $\mathbf{J}$ can be treated as a constant in a local region. $\mathbf{J}_i$, the $i$-th column of $\mathbf{J}$, represents the contribution of the $i$-th actuator, $\theta_i$, on the end-effector position. Then $\mathbf{J}$ can be estimated by moving each actuator independently by an incremental amount and observing their effects on the robot's end-effector.

The kinematic-free control framework can be summarized as a three-step process: *1)* Estimate the initial Jacobian matrix $\hat{\mathbf{J}}$, *2)* Generate control signals $\Delta\boldsymbol{\theta}$ in $q_2$, and *3)* Evaluation of Jacobian matrix according to visual feedback.

*1) Estimate the initial Jacobian matrix $\hat{\mathbf{J}}$:*

When the system is switched to $q_1$, $\mathbf{J}$ needs to be estimated. Starting from the initial set, each actuator is driven by an incremental amount $\Delta\theta_i$. Then the effect on end-effector, $\Delta\mathbf{x}$, is measured by a camera. Thus

$$\hat{\mathbf{J}}_i = \Delta\mathbf{x}/\Delta\theta_i, \tag{3}$$

$$\hat{\mathbf{J}} = \begin{bmatrix} \hat{\mathbf{J}}_1 & \cdots & \hat{\mathbf{J}}_n \end{bmatrix}. \tag{4}$$

$\hat{\mathbf{J}}$ can be viewed as a constant and be re-used in a local region. Thus it is not necessary to estimate the Jacobian matrix in every control period.

The error vector $\Delta\mathbf{x}$ is defined as

$$\Delta\mathbf{x} = \mathbf{x}_{tarj} - \mathbf{x}_{cur}. \tag{5}$$

**Theorem 1.** If a target $\mathbf{x}_{tarj}$ is unreachable in the current configuration, then each row in Jacobian, $\mathbf{J}_i$, is perpendicular to the error vector $\Delta\mathbf{x}$ at the point that is the closest to $\mathbf{x}_{tarj}$ in the current workspace.

Please feel free to contact us for the proof of Theorem 1. According to Theorem 1, we define the condition as:

$$Q_1 = \left\{ \{\mathcal{Q}\} \times \mathcal{X} \left| \sum_{i=1}^{n} \left| \frac{\hat{\mathbf{J}}_i \Delta \mathbf{x}}{\left\| \hat{\mathbf{J}}_i \right\| \cdot \|\Delta \mathbf{x}\|} \right| < n \cos \phi_{\max} \right. \right\}, \tag{6}$$

where $n$ denotes the manipulator's DOFs, $\phi_{\max}$ is a tolerable threshold for the angle between $\hat{\mathbf{J}}_i$ and $\Delta \mathbf{x}$.

Considering the joint limits and mechanical interface, we define the condition as:

$$Q_2 = \left\{ \{\mathcal{Q}\} \times \mathcal{X} \left| \bigcup_i (\theta_i > \theta_{i\max} \cup \theta_i < \theta_{i\min}) \right. \right\}. \tag{7}$$

Hence, $g(q_1, q_3)$, $g(q_2, q_3)$ and $g(q_1, q_2)$ are defined as:

$$g(q_1, q_3) = \{\{q_1\} \times \mathcal{X} \,|Q_1 \cup Q_2\} \setminus g(q_1, q_6); \tag{8}$$

$$g(q_2, q_3) = \{\{q_2\} \times \mathcal{X} \,|Q_1 \cup Q_2\} \setminus g(q_2, q_6); \tag{9}$$

$$g(q_1, q_2) = (\{\{q_1\} \times \mathcal{X}\} \setminus g(q_1, q_3)) \setminus g(q_1, q_6), \tag{10}$$

where $A \setminus B$ is the relative complement set of $B$ in $A$.

Here we can give a more precise definition for $r(q_3, q_1, \mathbf{x})$:

$$r(q_3, q_1, \mathbf{x}) = \begin{cases} h(\cdots, \theta_{c_i}, \cdots) & g(q_j, q_3) \in Q_2 \\ \mathbf{x}' & g(q_j, q_3) \notin Q_2 \end{cases}, \tag{11}$$

where $h(\cdot)$ is the kinematic function which is unknown to our controller and $\theta_{c_i} = \frac{(\theta_{i\max} + \theta_{i\min})}{2}$, in which $\theta_{i\max}$ and $\theta_{i\min}$ are the maximum and the minimum allowable joint values, respectively. This definition means that the $i$-th joint is set to its middle position when the bending operation is caused by joint limits.

If $g(q_1, q_3)$ or $g(q_2, q_3)$ is satisfied, namely, a target is unreachable or a joint crosses its limits, the system is switched to $q_3$. In discrete state $q_3$, the system needs to solve the problem that what configuration the deformable link should be bent to. In this paper, we adopt a heuristic method to get a feasible solution for a special case, which will be explained in Section IV.

After the bending process, the system is switched to $q_1$ again to re-estimate the Jacobian matrix. Thus $g(q_3, q_1)$ can be defined as:

$$g(q_3, q_1) = \{\{q_3\} \times \mathcal{X}\}. \tag{12}$$

*2) Generate control signals $\Delta\boldsymbol{\theta}$ in $q_2$:*

After the Jacobian matrix is estimated in $q_1$, the system is switched to $q_2$ to generate control signals. The Jacobian matrix estimated in the first step is only effective locally. Therefore, the error vector $\Delta \mathbf{x}$ cannot be used directly to generate the control signals. Hence the key of kinematic-free controller is to generate a tiny and temporarily desired displacement for the end-effector in each control period. Define $\Delta \mathbf{x}_d$ as the desired displacement:

$$\Delta \mathbf{x}_d = \frac{a(\mathbf{x}_{tarj} - \mathbf{x}_{cur})}{\|\mathbf{x}_{tarj} - \mathbf{x}_{cur}\|}, \tag{13}$$

where $a = \begin{cases} 3 & \|\Delta \mathbf{x}\| > 10 \\ 1 & 5 < \|\Delta \mathbf{x}\| < 10 \\ 0.5 & \|\Delta \mathbf{x}\| \le 5 \end{cases}$ is the step length.

Then the control signal is generated according to

$$\Delta \boldsymbol{\theta} = \mathbf{J}^{\dagger} \Delta \mathbf{x}_d + \left(\mathbf{I}_n - \mathbf{J}^{\dagger} \mathbf{J}\right) \mathbf{v}, \tag{14}$$

where $\mathbf{J}^{\dagger} = \hat{\mathbf{J}}^T \left(\hat{\mathbf{J}} \hat{\mathbf{J}}^T\right)^{-1}$ is the pseudoinverse of $\hat{\mathbf{J}}$, $\mathbf{v} = -\partial \boldsymbol{\Phi}(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}$ is the secondary task, $\mathbf{I}_n$ is the identity matrix. To avoid joint limits and self collision, $\boldsymbol{\Phi}(\boldsymbol{\theta})$ is defined as:

$$\boldsymbol{\Phi}(\boldsymbol{\theta}) = \frac{\lambda}{2} \sum_{i=1}^{n} w_i \frac{(\theta_i - \theta_{c_i})^2}{\theta_{i\max} - \theta_{i\min}}, \tag{15}$$

where $\lambda$ is the tuning coefficient, $w_i$ is the weight for each joint.

*3) Evaluation of $\hat{\mathbf{J}}$ according to visual feedback:*

The end-effector's position, after $\Delta\boldsymbol{\theta}$ is exerted to the system, is denoted as $\mathbf{x}_{cur+1}$. Then the actual displacement of end-effector, $\Delta \mathbf{x}_k$, is measured by

$$\Delta \mathbf{x}_k = \mathbf{x}_{cur+1} - \mathbf{x}_{cur}. \tag{16}$$

Then the deviation between $\Delta \mathbf{x}_k$ and $\Delta \mathbf{x}_d$ can be seen as the evaluation of the Jacobian matrix:

$$\varepsilon = \|\Delta \mathbf{x}'\| = \|\Delta \mathbf{x}_k - \Delta \mathbf{x}_d\|. \tag{17}$$

Hence, we can define $g(q_2, q_4)$, $g(q_2, q_1)$, $g(q_4, q_2)$, $g(q_4, q_5)$ as:

$$\begin{aligned} g(q_2, q_4) = &\{\{q_2\} \times \mathcal{X} \,|\|\mathbf{x}_{tarj} - \mathbf{x}_{cur+1}\| < r_{th}\} \\ &\setminus g(q_2, q_3) \setminus g(q_2, q_6); \end{aligned} \tag{18}$$

$$\begin{aligned} g(q_2, q_1) = &\{\{q_2\} \times \mathcal{X} \,|\varepsilon > a\varepsilon_{\max}\} \\ &\setminus g(q_2, q_4) \setminus g(q_2, q_3) \setminus g(q_2, q_6); \end{aligned} \tag{19}$$

$$g(q_4, q_2) = \{\{q_4\} \times \mathcal{X} \,|||P'_{tar}| > 0\}; \tag{20}$$

$$g(q_4, q_5) = \{\{q_4\} \times \mathcal{X} \,|||P'_{tar}| = 0\}, \tag{21}$$

where $r_{th}$ defines a ball region centered in the current target and $\varepsilon_{\max}$ is a tolerable upper bound for the inaccuracy of Jacobian. $|\cdot|$ denotes the cardinality of a set.

The interpretations of these conditions are intuitive: $g(q_2, q_4)$ means that a target is reached. Then the system checks whether $g(q_4, q_5)$ is satisfied, namely, all targets are reached. If $g(q_4, q_5)$ is satisfied, the long path tracking task is accomplished and the system is switched to $q_5$. Otherwise, the system comes back to $q_2$ and tracks the next target. $g(q_2, q_1)$ means that the deviation between actual and desired displacement is too large, namely, the Jacobian is not accurate enough. Thus the system is switched to $q_1$ to re-estimate the Jacobian matrix.

**Theorem 2.** Given a target $\mathbf{x}_{tarj}$, the end-effector is driven to a ball region , whose center is the target and its radius is $a$, or to a point that is the closest to $\mathbf{x}_{tarj}$ in the current workspace. The distance decreases at a rate which is no slower than $|a(1 - \varepsilon_{\max})|$, where $\varepsilon_{\max} < 1$.

Please feel free to contact us for the proof of Theorem 2. Please note that $\varepsilon_{\max}$ is an upper bound. Actually, we can assign a larger value to $\varepsilon_{\max}$ to avoid frequent re-estimation in actual use.

## IV. Experiments in Simulation

To verify the proposed model and controller, the experiments in simulation are conducted on a 4-DOF deformable manipulator with 2 deformable links. One deformable link, which is composed of 10 sphere components, connects the shoulder joint and elbow joint. The other one, which is composed of 5 sphere components, connects the elbow joint and wrist joint. The base frame of the deformable manipulator is given by:

$$T_0 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 200 \\ -1 & 0 & 0 & 400 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (22)$$

Any prior kinematic parameters are not given to the controller in any way. Since the experiments are performed in simulation, there is no need to use a camera. Instead, the end-effector's position is obtained internally from the simulator.

Two types of experiments are performed as follows:

1) Type 1: Simple Tracking Task. In type 1, all targets are distributed in the current workspace of the deformable manipulator. The deformable manipulator can track these targets without bending operation;

2) Type 2: Difficult Tracking Task. In type 2, the targets are distributed in a relatively big scope of the 3-D Euclidean space. For a traditional manipulators with the same DOFs (4-DOF in this paper) and same link lengths, its workspace is restricted due to joint limits, self collision, etc. Therefore, it may be impossible to design a manipulator whose workspace can cover the whole path. However, the deformable manipulator can obtain dexterous and extended workspace to track all targets by bending operation.

The experiments and their results are explained in detail below. The parameters of the controller used in these experiments are: $r_{th} = 3.0mm$, $\varepsilon_{\max} = 1.5mm$, $\boldsymbol{\theta}_{\min} = [-\pi, -\pi, -6\pi/7, -2\pi]^T$, $\boldsymbol{\theta}_{\max} = [\pi, \pi, 6\pi/7, 2\pi]^T$, $\phi_{\max} = 5\pi/12$, $W = [w_1, w_2, w_3, w_4] = [1, 1, 2, 1]$, $\beta = 2$.

### A. Type 1: Simple Tracking Task

In this experiments, a more strict threshold is given: $r_{th} = 1.5mm$. The controller is given a whole reference trajectory that needs to be tracked. The path, which is given by a set of fuchsia points, is composed of two parts. One is the square shape path and the other is the figure-8 shape path. The side length of the square is $200mm$. The figure-8 shape path is composed of two circles, whose diameters are both $100mm$. The reference trajectory is specified only in the task space and not in the time domain.

An attempt to track this trajectory is shown in Fig. 4. This experiment demonstrates that the kinematic-free controller can reuse the Jacobian in a big scope of the workspace which is reflected by the reduced time of re-estimation process. This experiment demonstrates that the controller can control the deformable manipulator to track a desired path or target with no prior information about the kinematic parameters, as well as the traditional controller.

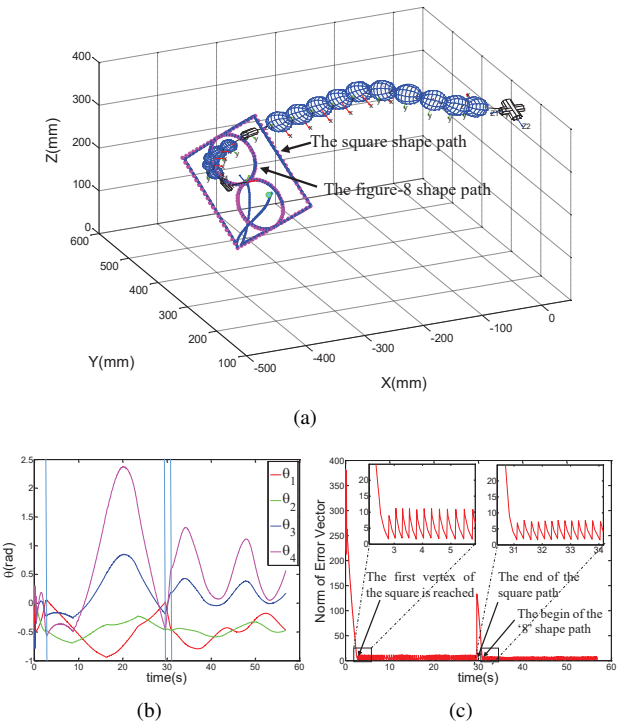



Fig. 4. (a) The path, (b) The recorded joint positions (in radian), and (c) The norms of error vectors of a simple tracking task

### B. Type 2: Difficult Tracking Task

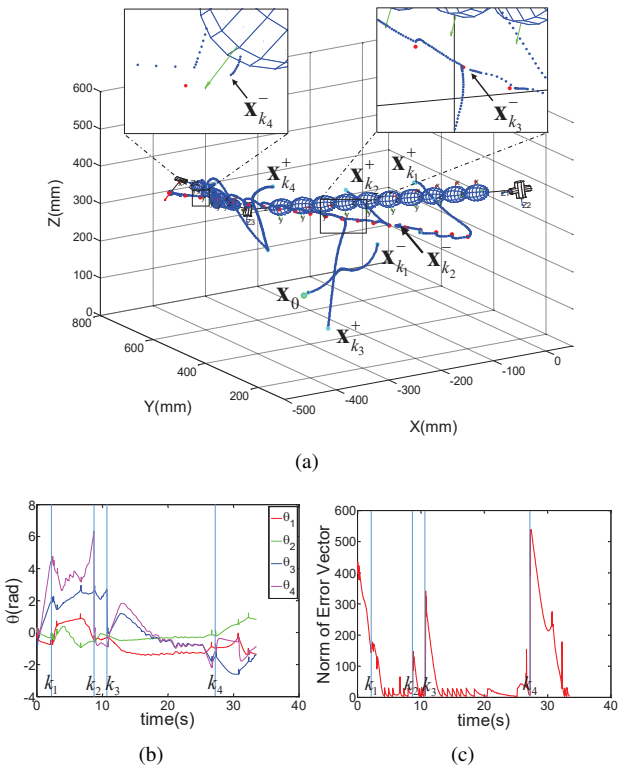
A given long path, $P_{tar} = \{\mathbf{x}_{tar1}, \mathbf{x}_{tar2}, \cdots, \mathbf{x}_{tar20}\}$, which is denoted by the red points in Fig. 5(a), is a line segment whose origin is $\mathbf{x}_{tar1} = [-100, 200, 300]^T$ $(mm)$ and end position is $\mathbf{x}_{tar20} = [-500, 600, 400]^T$ $(mm)$. The targets in this path are distributed in a relatively big scope of 3-D Euclidean space and are very difficult to be tracked by traditional manipulators with the same DOFs.

What configuration of the deformable links should be bent to is formulated as the following optimal problem:

$$\begin{matrix} \text{Max} \\ \alpha_i, \beta_i, \gamma_i \\ \text{subject} \end{matrix} \quad \begin{matrix} L_1 + L_2 \\ \left\{ \begin{matrix} L_1 - L_2 \le \|\mathbf{x}_{tarj} - \mathbf{r}_0\| \\ \alpha_i \le \alpha_{\max}, \beta_i \le \beta_{\max}, \gamma_i \le \gamma_{\max} \end{matrix} \right. \end{matrix}, \quad (23)$$

where $\mathbf{x}_{tarj}$ is the current target, $L_1 = \|\mathbf{r}_3 - \mathbf{r}_0\|$ and $L_2 = \|\mathbf{r}_4 - \mathbf{r}_3\|$ are the effective lengths of two deformable links, respectively, $\alpha_{max}$, $\beta_{max}$, and $\gamma_{max}$ are the maximum values for each sphere. $L_2$ remains constant in the optimal process.

The experiment results are shown in Fig. 5. Since $\mathbf{x}_{tar1}$ is unreachable in current configuration, the system is switched to $q_3$ when it reaches the point that is the closest to the target ($Q_1$ is satisfied). A bending operation is triggered to shorten the length of $L_1$. Then three bending operations are triggered successively to elongate $L_1$ when $\mathbf{x}_{tar6}$, $\mathbf{x}_{tar9}$ and $\mathbf{x}_{tar18}$ are tracked, in which two times are caused by $Q_2$ ($k_2$, $k_3$) and the other one is caused by $Q_1$ ($k_1$, $k_4$). It can be seen that the end-effector's position exhibits an instantaneous change from $\mathbf{x}_{k_i}^-$ to $\mathbf{x}_{k_i}^+$, which is a typical hybrid behavior (the time in Jacobian estimation and bending process is neglected).




(a)

(b) (c)

Fig. 5. (a) The path, (b) The recorded joint positions (in radian), and (c) The norms of error vectors to track a long path.

These simulation results demonstrate that the deformable manipulator can obtain dexterous and extended workspace with fewer joints by bending operations.

## V. Conclusion

In this paper, we proposed a novel deformable manipulator to provide dexterous workspace at low-cost. The deformable manipulator can obtain relatively dexterous and extended workspace with fewer joints by bending its deformable links. A hybrid model and kinematic-free control framework were proposed to describe and control the manipulator to track a long path, which is usually difficult to be tracked by conventional manipulators with the same DOFs. The simulation results demonstrated that the deformable manipulator can obtain dexterous and extended workspace with fewer joints by bending operations.

There are still two issues remaining to be investigated in the future work: First, what configuration should a deformable link be bent to; Second, how to bend it to a desired configuration. The practical real-world implementation will also need to be done in the future work.

## References


[1] http://www.ifr.org/service-robots/statistics.

[2] A. Kazakidi, X. Zabulis and D. P. Tsakiris, Vision-based 3D Motion Reconstruction of Octopus Arm Swimming and Comparison with an 8-arm Underwater Robot, *IEEE International Conference on Robotics and Automation*, Seattle, USA, 2015, pp. 1178-1183.

[3] M. Cianchetti, M. Follador, B. Mazzolai, P. Dario and C. Laschi, Design and Development of a Soft Robotic Octopus Arm Exploiting Embodied Intelligence, *IEEE International Conference on Robotics and Automation*, Saint Paul, USA, 2012, pp. 5271-5276.

[4] E. Guglielmino, L. Zullo, M. Cianchetti, M. Follador, D. Branson and D. G. Caldwell, The Applicaiton of Embodiment Theory to the Design and Control of an Octopus-like Robotic Arm, *IEEE International Conference on Robotics and Automation*, Saint Paul, USA, 2012, pp. 5277-5282.

[5] C. Escande, T. Chettibi, R. Merzouki, V. Coelen and P. M. Pathak, Kinematic Calibration of a Multisection Bionic Manipulator, *IEEE/ASME Transactions on Mechatronics*, 2015, 20(2), pp. 663-674.

[6] V. Falkenhahn, A. Hildebrandt, R. Neumann and O. Sawodny, Model-based Feedforward Position Control of Constant Curvature Continuum Robots using Feedback Linearization, *IEEE International Conference on Robotics and Automation*, Seattle, USA, 2015, pp. 762-767.

[7] T. Mahl, A. Hildebrandt and O. Sawodny, A Variable Curvature Continuum Kinematics for Kinematic Control of the Bionic Handling Assistant, *IEEE Transactions on Robotics*, 2014, 30(4), pp. 935-949.

[8] C. Escande, P. M. Pathak, R. Merzouki, and V. Coelen, Modelling of Multisection Bionic Manipulator: application to RobotinoXT, *IEEE International Conference on Robotics and Biomimetics*, Phuket, Thailand, 2011, pp. 92-97.

[9] C. Bergeles, A. H. Gosline, N. V. Vasilyev, P. J. Codd, P. J. Nido and P. E. Dupont, Concentric Tube Robot Design and Optimization Based on Task and Anatomical Constraints, *IEEE Transactions on Robotics*, 2015, 31(1), pp. 67-84.

[10] D. C. Rucker, B. A. Jones, R. J. Robert, A Model for Concentric Tube Continuum Robots Under Applied Wrenches, *IEEE International Conference on Robotics and Automation*, Anchorage, USA, 2010, pp. 1047-1052.

[11] D. C. Rucker, B. A. Jones and R. J. Webster, A Geometrically Exact Model for Externally Loaded Concentric-Tube Continuum Robots, *IEEE Transactions on Robotics*, 2010, 26(5), pp. 769-780.

[12] S. Neppalli, M. A. Csencsits, B. A. Jones and I. D. Walker, Closed-Form inverse kinematics for continuum manipulators, *Advanced Robotics*, 2009, 23(15), pp. 2077-2091.

[13] T. A. Henzinger, The Theory of Hybrid Automata, *Springer Berlin Heidelberg*, 2000, pp. 281-292.

[14] R. Goebel, R. G. Sanfelice and A. R. Teel, Hybrid Dynamical Systems, *IEEE Control System Magazine*, 2009, 29(2), pp. 28-93.

[15] J. Lygeros, K. H. Johansson, S. N. Simić, J. Zhang and S. S. Sastry, Dynamical Properties of Hybrid Automata, *IEEE Transactions on Automatic Control*, 2003, 48(1), pp. 2-17.

[16] X. Lu, J. Liu, J. Hao, S. Zhang and L. Sun, Self-Calibration of Deformable Arm with a Monocular Camera, *IEEE International Conference on Robotics and Biomimetics*, Bali, Indonesia, 2014, pp. 861-866.

[17] J. Hao, Research on Kinematics of Deformable Manipulator for Home Service Robots, Master Thesis, Nankai University, 2015.

[18] X. Lu, Research and Application on Object Recognition and Pose Measurement based on Monocular Camera, Ph.D. Thesis, Nankai University, 2015.

[19] S. Chareyron and P-B. Wieber, Position and Force Control of Nonsmooth Lagrangian Dynamical Systems Without Friction, *International Workshop on Hybrid Systems Computation and Control*, Zurich, Switzerland, 2005, pp. 215-225.

[20] J. W. Grizzle, C. Chevallereau, A. D. Ames and R. W. Sinnet, 3D Bipedal Robotic Walking: Models, Feedback Control, and Open Problems, *IFAC Symposium on Nonlinear Control Systems*, 2010, 2(3), pp. 8-35.

[21] A. D. Ames, K. Galloway, K. Sreenath and J. W. Grizzle, Rapidly Exponentially Stabilizing Control Lyapunov Functions and Hybrid Zero Dynamics, *IEEE Transactions on Automatic Control*, 2014, 59(4), pp. 876-891.

[22] P. Kormushev, Y. Demiris and D. G. Caldwell, Encoderless Position Control of a Two-Link Robot Manipulator, *IEEE International Conference on Robotics and Automation*, Seattle, USA, 2015, pp. 943-949.

[23] P. Kormushev, Y. Demiris and D. G. Caldwell, Kinematic-free Position Control of a 2-DOF Planar Robot Arm, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hamburg, Germany, 2015, pp. 5518-5525.

[24] M. C. Yip and D. B. Camarillo, Model-less Feedback Control of Continuum Manipulators in Constrained Environments, *IEEE Transactions on Robotics*, 2014, 30(4), pp.880-889.

[25] V. Vikas, P. Grover and B. Trimmer, Model-free Control Framework for Multi-limb Soft Robots, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hamburg, Germany, 2015, pp. 1111-1116.