IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier

Real-time Joint Angular-Acceleration Planning for Vision-based Kinematically Redundant Manipulator in Dynamic Environment

HUI ZHANG[1], HONGZHE JIN[1], Member, IEEE, MINGDA GE[1], YANHE ZHU[1], Member, IEEE, and JIE ZHAO[1], Member, IEEE

[1]School of Mechatronics Engineering, Harbin Institute of Technology, Harbin, 150001, China

Corresponding author: Jie Zhao (e-mail: jzhao@hit.edu.cn), Hongzhe Jin (e-mail: hongzhejin@hit.edu.cn), Yanhe Zhu (e-mail: yhzhu@hit.edu.cn).

This work was supported in part by the National Key R&D Program of China under Grant 2017YFF0108000, the Provincial Supporting Projects of the National Key R&D Program of China under Grant GX18A017, the Major Research Plan of the National Natural Science Foundation of China under Grant 91648201, the National Natural Science Foundation of China under Grant 61473102, the Joint Research Fund (U1713201) between the National Natural Science Foundation of China (NSFC) and Shen Zhen, and the Self-Planned Task of State Key Laboratory of Robotics and System (HIT) under Grant SKLRS201805B.

ABSTRACT This paper presents an angular acceleration planning for joint configuration spaces of vision-based kinematically redundant robot manipulators to achieve real-time tracking tasks in dynamic environment. In accordance with the proved proposition, the joint velocity formed by the inverse kinematics mapping of the defined pose error is regarded as the system error, and utilized to further deduce the planned joint angular acceleration and with kinematic redundancy for obstacle avoidance simultaneously. Highly-complex nonlinearities in the angular acceleration are equivalent to a system perturbation represented by a synthetic form, thereby simplifying the tedious calculations of the angular acceleration theoretically and converting the planning problem into a control problem. Hyperbolic tangent-based super twisting algorithm, as the control input of manipulator system, is designed to resist the synthetic perturbation. The collision-free movement of redundant manipulators for tracking tasks is thus achieved through the integration of the control input in real time. An error-based S-function is proposed as the internal parameter of the planned joint acceleration to prevent the integral saturation. Lyapunov theoretical analysis proves the stability of the proposed super twisting algorithm. Simulation and contrast experiments in dynamic-obstacles environments indicate that the proposed joint angular acceleration planning for kinematically redundant manipulators owns feasibility, smoothness and practicability.

INDEX TERMS Angular acceleration, synthetic perturbation, hyperbolic tangent-based super twisting algorithm, redundant manipulator.

I. INTRODUCTION

The development of artificial intelligence technology has aroused great interest in the research on vision-based robot manipulators for autonomous and intelligent operations/tasks during interacting with the dynamic environments [1], such as with human, robots or other automation equipment. In the process of the operations/tasks, real-time motion planning for vision-based robot manipulators not only ensures the safe and collision-free motion but also plans the smooth trajectories to avoid the system chattering that may seriously damage the actuators and degrade the tracking performance of robot [2, 3]. Furthermore, smooth trajectories can be executed faster and with a higher accuracy, and make vision-based robot manipulators complete operations /tasks with higher quality in dynamic environments. Hence, obtaining the smooth joint trajectory is key in the research of the real-time motion planning for vision-based robot manipulators.

The real-time motion planning methods for robot manipulators has been thoroughly studied in existing literatures and divided into real-time Cartesian motion planning [4–20] and real-time joint trajectory planning [21–33]. Real-time Cartesian motion planning is usually chosen priority because of the intuitive operation/task space, simple description of the path constraints, direct expression of the end-effector (EE) pose, and easy achievement for obstacle avoidance. In order to generate a smooth joint trajectory for robot manipulator in operations/tasks, many interpolation points on the moving

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

path are always defined via linear, polynomial or spline function in Cartesian space [4–7]. Meanwhile, the velocity and acceleration of the insertion points need to be determined to ensure the motion continuity [12, 13]. Although the interpolation points with constraints can get a smooth trajectory for the EE and joints, the real-time solutions of forward kinematics (FK) and inverse kinematics (IK) for each interpolation points have become the system intrinsic complexity and consumed a part of the computational effort. Additionally, force/moment, dynamics model, parameterized processing of velocity and acceleration, and limit of absolute jerk value, are also required in dynamics-based planning [6, 11, 13], which results in much more highly nonlinear model and tedious calculations. Some real-time planning methods based on kinematics is sample and practical, like fixed proportion-based [16], fixed clamp-based [17], pose error controller-based [18], and intelligent methods [10, 19]. However, these methods consider only position and velocity information in Cartesian space and fluctuations still exist in the path and joint trajectory. Overall, even though the smooth motion of manipulator could be achieved by using real-time Cartesian motion planning, the generated joint trajectories are not as smooth as those generated based on real-time joint trajectory planning.

Real-time joint trajectory planning methods can plan the joint motion trajectories of the manipulator directly and the motion of the manipulator is controlled by the planned joint inputs, such as the position, velocity, acceleration and jerk [20–24], or the torque [25–27] those are constrained through boundary condition, fitting function, optimization (or cost) function, or learning algorithm. Usually, the task space is non-intuitive, yet smooth motion planning is easy to realize and conducive to reducing the jitter of the joint and the manipulator. However, the joint trajectory planning easily causes the motion error of the EE and the unsafe distance between EE and obstacles [28]. The advanced methods, formulating the motion planning as quadratic programming problem, are flexible, accurate and efficient. The dual neural network scheme with convex–nonconvex constraints for redundant manipulator can achieve the real-time stable motion and owns high-accuracy and -efficiency in tasks [29]. The optimal criterion of joint angle and velocity is designed to prevent the occurrence of high joint velocity for redundant robot manipulators [30]. The varying-parameter neural network for solving the motion problem in joint space of redundant manipulator can avoid the joint-angular-drift in real time [31, 32]. In addition, the time-optimal joint trajectory planning method can obtain quite smooth joint acceleration and velocity [33], and accurate tracking of the EE. Nevertheless, the high-order dynamics and kinematics equation need be solved to acquire the planned joint control inputs, which brings the complex mathematical operations and difficulty in the application of real-time planning for low-cost and low-power microcontrollers.

Considering the real-time motion planning methods comprehensively, mostly methods, which relates to the interpolation, fitting or intelligent algorithm, aim to planning the continuous angular-acceleration to ensure the smoothness of the joint trajectory. Such ways can endow the manipulator with the good tracking performance and avoiding the system chattering simultaneously, but the real-time motion planning always involves the highly-complex nonlinear kinematics/dynamics models due to the differential for obtaining acceleration information and affects the real-time performance of the robot system. Nearly no research simplifies these models, especially the kinematics-based models. In addition, seldom method can directly plan the smooth joint trajectory and ensure intuitive constraints of Cartesian space tasks in the case of dynamic obstacles at the same time. Therefore, planning the continuous joint angular acceleration directly based on kinematics, owning intuitive task constraints in dynamic-obstacle environment, and simplifying the models will be considered in this paper.

On the basis of the previous contributions [6, 10, 11, 18, 20–24, 33], this paper proposes a simple and practical real-time planning method of angular acceleration in joint configuration spaces based on the kinematics, which can describe the operations tasks and motion constraints of Cartesian space directly in dynamic environment, and obtain smooth trajectories in joint space simultaneously.

The contributions and characteristics of the proposed real-time planning method those differ from the existing research are embodied as following aspects.

The vectors in IK for obstacle avoidance and tracking tasks are proved to be linear independent and orthogonal in the proposition so that the "joint velocity" formed by the IK mapping can be regarded as the "system error" and further derived to obtain the planned joint angular acceleration. Due to the null space vector in IK, the kinematic redundancy is still possessed for avoiding obstacles in the planned angular acceleration. i.e., the angular acceleration planning can also ensure collision-free motion of redundant manipulators in dynamic-obstacle environment during operations/tasks at the same time. The highly complex items in the angular acceleration equation are simplified and equivalent to the synthetic disturbance of the system, which theoretically reduces the system complexity and calculation cost. As a result, the real-time "planning problem" is converted into a nonlinear "control problem" about the joint angular acceleration, which means using the control technology to deal with the planning problem and is completely different from existing planning ideas. In the torque control design of nonlinear dynamics system, the traditional super twisting algorithm (STA) has been applied widely due to the great contributions, including the fast convergence in finite time, robustness, simplicity, high control accuracy and overcoming chattering of sliding mode control [34, 35]. Unfortunately, the chattering occurs when the traditional STA is used in real-time planning of kinematics-based robot manipulator system. So, as an improved method, the hyperbolic tangent-based STA is designed to expand the application of the traditional STA in motion planning. The designed STA is similar to sliding mode control and can effectively suppress the chattering in the planning

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

based on the traditional STA and resist the equivalent synthetic perturbation. Thus, the proposed angular acceleration planning method is formed by the real-time control of the designed STA, which does not rely heavily on models and can achieve the collision-free motion of redundant manipulator in dynamic environment. Besides, the proposed planning method can avoid the insufficiencies that the interpolation points with motion constraints need to be defined in real-time Cartesian motion planning and that the EE pose exists the non-intuitive task description and the unsafe distance with obstacles in real-time joint trajectory planning. The system stability is proved by Lyapunov theory. The analysis for the influence of parameters on the manipulator system, by using the hyperbolic tangent-based STA, is difficult due to the integral link and the hyperbolic tangent term. Then the designed Lyapunov function is further deduced to discuss the convergence performance and the motion in different cases.

Simulations and experiments of a moving object tracking in dynamic-obstacles environment fully demonstrate the feasibility, smoothness and practicability of the proposed real-time planning method of angular acceleration in joint configuration spaces.

The reminder of this paper is organized as follows. Section II discusses the real-time angular-acceleration planning, including the mapping from Cartesian space to joint space, the hyperbolic tangent-based STA and its stability analysis. Section III and Section IV provide the simulation and experimental results in different cases, respectively. Section V presents the conclusion.

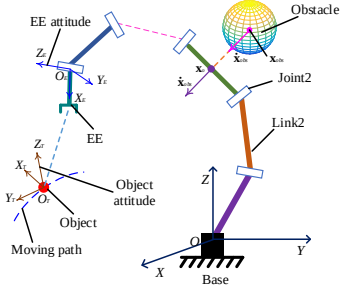


FIGURE 1. **s-DoF manipulator tracking a moving object in real time. *s*>6.**

II. REAL-TIME ANGULAR-ACCELERATION PLANNING

A. Mapping from Cartesian Space to Joint Space

This paper considers the real-time tracking an object as the tracking task. In Fig. 1, the Cartesian coordinate *O-XYZ* is the base coordinate system of the redundant manipulator. The coordinate system $\{X_T, Y_T, Z_T\}$ is the object attitude, which makes the EE face the direction of the coordinate axis X_T. $\boldsymbol{\alpha}$ is defined to directly describe the pose error and its first-order derivative between the EE and the object in Cartesian space as follows:

$$\boldsymbol{\alpha} = \eta_p(\mathbf{x}_r - \mathbf{x}) + \eta_d(\dot{\mathbf{x}}_r - \dot{\mathbf{x}}), \tag{1}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^{N\times 1}$ is used to obtain the higher-order differential information in the following theoretical derivation process. N equals to 3 for planar manipulator and N equals to 6 for spatial manipulator. η_p and η_d denote the constant coefficients. $\mathbf{x}_r = (x_r, y_r, z_r, \omega_r, \varphi_r, \gamma_r)^{\mathrm{T}}$ defines the object pose in Cartesian space. x_r, y_r and z_r express the position. ω_r, φ_r and γ_r refer to the attitude that is calculated by the equivalent angle-axis representation about the base coordinate system. $\dot{\mathbf{x}}_r$ is the corresponding velocity of $\mathbf{x}_r$. $\mathbf{x} = (x, y, z, \omega, \varphi, \gamma)^{\mathrm{T}}$ indicates the EE pose. x, y and z express the position. ω, φ and γ signify the attitude and are calculated by the equivalent angle-axis representation about the base coordinate system. $\dot{\mathbf{x}}$ is the corresponding velocity of $\mathbf{x}$.

The general IK of redundant robot manipulator [16] is utilized to map the pose error $\boldsymbol{\alpha}$ in Cartesian space to the joint space as follows:

$$\boldsymbol{\beta} \triangleq \mathbf{J}^*(\boldsymbol{\Theta})\boldsymbol{\alpha} + [\mathbf{I} - \mathbf{J}^*(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta})]\mathbf{v}, \tag{2}$$

where $\boldsymbol{\Theta}$ signifies the joint angles and $\boldsymbol{\Theta}=(\theta_1,\theta_2,\ldots,\theta_s)^{\mathrm{T}}$, $\boldsymbol{\Theta} \in \mathbb{R}^{s\times 1}$, and the degree of freedom (DoF) of manipulator is s. $\boldsymbol{\beta}$ is the joint velocity of manipulator in joint space that is formed by the IK mapping, $\boldsymbol{\beta}=(\beta_1, \beta_2, \ldots, \beta_s)^{\mathrm{T}}$. $\boldsymbol{\beta}$ is also regarded as the "system error" and the corresponding explanation is presented in last paragraph of Part A in Section II. $\mathbf{J}^*$ denotes the pseudo-inverse of Jacobian matrix $\mathbf{J}(\boldsymbol{\Theta})$ based on the damped least squares (DLS) method. $\mathbf{J}^*(\boldsymbol{\Theta}) \triangleq \mathbf{J}^{\mathrm{T}}[\mathbf{J}\mathbf{J}^{\mathrm{T}} + \lambda^2\mathbf{I}_1]^{-1}$ and $\mathbf{J} \in \mathbb{R}^{N\times s}$. The weighted pseudo-inverse is used to handle the joint limits. $\mathbf{I}_1$ is a unit matrix with the dimension $N\times N$. λ represents the damping factor that can handle the ill-conditioned $\mathbf{J}$ in the neighborhood of singular configurations for redundant manipulators and guarantee the EE with minimum possible deviation at all configurations [16]. $\mathbf{I}$ is a unit matrix with the dimension $s\times s$. $[\mathbf{I} - \mathbf{J}^*\mathbf{J}]\mathbf{v}$ is the null space and makes the position $\mathbf{x}_o$ on the arm of manipulator that closes to the position of the obstacle $\mathbf{x}_{obs}$ perform a specified velocity $\dot{\mathbf{x}}_o$ to avoid the obstacle as shown in Fig. 1.

$$\mathbf{v} = k_o\mathbf{J}_o^*\dot{\mathbf{x}}_o, \quad \|\dot{\mathbf{x}}_o\| > \|\dot{\mathbf{x}}_{obs}\|, \tag{3}$$

where k_o is obstacle avoidance gain. $k_o = \left(\frac{d_m}{\|\mathbf{x}_o - \mathbf{x}_{obs}\|}\right)^2 - 1$ if $\|\mathbf{x}_o - \mathbf{x}_{obs}\| < d_m$. $k_o = 0$ if $\|\mathbf{x}_o - \mathbf{x}_{obs}\| \geq d_m$. d_m is the critical distance to the obstacle. $\mathbf{J}_o^*$ is the pseudo-inverse of the Jacobian for avoiding the obstacle based on the DLS method, $\mathbf{J}_o^* = \mathbf{J}_o^{\mathrm{T}}(\mathbf{J}_o\mathbf{J}_o^{\mathrm{T}} + \lambda^2\mathbf{I}_o)^{-1}$. $\mathbf{I}_o$ is a unit matrix. The detailed explanations about obstacle avoidance are presented in [16]. The motion of the arm for obstacle avoidance has no influence on the tracking task because of the orthogonal and linearly independent relationship between $\mathbf{J}^*(\boldsymbol{\Theta})\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^*(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta})]\mathbf{v}$ in the case of λ=0.

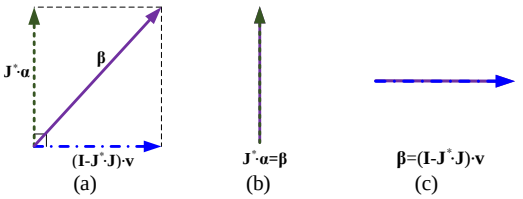


FIGURE 2. **The relationship among the vectors β, $\mathbf{J}^*\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^*\mathbf{J}]\mathbf{v}$. (a) Two vertical vectors $\mathbf{J}^*\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^*\mathbf{J}]\mathbf{v}$ are the components of β. (b) In the case of tracking task only, $\mathbf{J}^*\boldsymbol{\alpha} = \boldsymbol{\beta}$. (c) In the case of obstacle avoidance task only, $\boldsymbol{\beta} = [\mathbf{I} - \mathbf{J}^*\mathbf{J}]\mathbf{v}$.**

Proposition. The vectors $\mathbf{J}^*(\boldsymbol{\Theta})\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^*(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta})]\mathbf{v}$ in (2) are orthogonal and linearly independent in the case of λ=0.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

Proof. In accordance with the conception of pseudo-inverse matrix [16], there is

$$\mathbf{J}^{*} \overset{\text{def}}{=} \mathbf{J}^{\mathrm{T}} \cdot [\mathbf{J}\mathbf{J}^{\mathrm{T}}]^{-1} = \mathbf{J}^{\mathrm{T}} \cdot \mathbf{M}, \qquad \mathbf{J} \cdot \mathbf{J}^{*} \cdot \mathbf{J} = \mathbf{J}. \tag{4}$$

Let the vector $\mathbf{J}^{*}\boldsymbol{\alpha}$ multiply by $[\mathbf{I} - \mathbf{J}^{*}\mathbf{J}]\mathbf{v}$ as follows:

$$\begin{aligned} &(\mathbf{J}^{*}\boldsymbol{\alpha})^{\mathrm{T}} \cdot [\mathbf{I} - \mathbf{J}^{*}\mathbf{J}]\mathbf{v} \\ &= \boldsymbol{\alpha}^{\mathrm{T}} \cdot (\mathbf{J}^{*})^{\mathrm{T}} \cdot [\mathbf{I} - \mathbf{J}^{*}\mathbf{J}]\mathbf{v} \\ &= \boldsymbol{\alpha}^{\mathrm{T}} \cdot [\mathbf{M}^{\mathrm{T}} \cdot \mathbf{J} - \mathbf{M}^{\mathrm{T}} \cdot \mathbf{J} \cdot \mathbf{J}^{*} \cdot \mathbf{J}]\mathbf{v} \\ &= \boldsymbol{\alpha}^{\mathrm{T}} \cdot [\mathbf{M}^{\mathrm{T}} \cdot \mathbf{J} - \mathbf{M}^{\mathrm{T}} \cdot \mathbf{J}]\mathbf{v} = 0 \end{aligned} \tag{5}$$

Therefore, the vectors $\mathbf{J}^{*}(\boldsymbol{\Theta})\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^{*}(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta})]\mathbf{v}$ are orthogonal and linearly independent in the case of $\lambda = 0$.

According to ***Proposition***, the vector $\boldsymbol{\beta}$ is the sum of two vertical vectors $\mathbf{J}^{*}\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^{*}\mathbf{J}]\mathbf{v}$ as shown in Fig. 2(a), which indicates that the tracking and obstacle avoidance tasks of the redundant robot manipulator have no influence on each other. Both $\mathbf{J}^{*}\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^{*}\mathbf{J}]\mathbf{v}$ become $\mathbf{0}$ when $\boldsymbol{\beta}$ equals to $\mathbf{0}$. i.e., $\boldsymbol{\beta} = \mathbf{0}$ implies that both the tracking task and the obstacle avoidance task are accomplished simultaneously. The vectors $\boldsymbol{\beta}$ and $\mathbf{J}^{*}\boldsymbol{\alpha}$ are coincidence when the tracking task is executing and the avoiding obstacle task is achieved as shown in Fig. 2(b). The vectors $\boldsymbol{\beta}$ and $[\mathbf{I} - \mathbf{J}^{*}\mathbf{J}]\mathbf{v}$ are coincidence when the tracking task is completed and the avoiding obstacle task is performing as shown in Fig. 2(c). In according with the DLS method, $\mathbf{J}^{*}\boldsymbol{\alpha}$ and $[\mathbf{I} - \mathbf{J}^{*}\mathbf{J}]\mathbf{v}$ are approximate orthogonal in the case of $\lambda \neq 0$ and they still become $\mathbf{0}$ when $\boldsymbol{\beta}$ equals to $\mathbf{0}$. Therefore, the "joint velocity" $\boldsymbol{\beta}$ formed by the IK mapping could also be regarded as the "system error" to judge whether the operation tasks of the manipulator is completed in this paper. The following content of this paper uses "system error" to express $\boldsymbol{\beta}$.

B. Converting Planning Problem to Control Problem

To obtain the joint angular acceleration, the system error $\boldsymbol{\beta}$ of (2) is derived as follows:

$$\dot{\boldsymbol{\beta}} = \mathbf{J}^{*}(\boldsymbol{\Theta})\dot{\boldsymbol{\alpha}} + \dot{\mathbf{J}}^{*}(\boldsymbol{\Theta})\boldsymbol{\alpha} + \overbrace{[\mathbf{I} - \mathbf{J}^{*}(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta})]\dot{\mathbf{v}} - \left[\dot{\mathbf{J}}^{*}(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta}) + \mathbf{J}^{*}(\boldsymbol{\Theta})\dot{\mathbf{J}}(\boldsymbol{\Theta})\right]\mathbf{v}}^{\mathbf{K}(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\mathbf{v},\dot{\mathbf{v}})}. \tag{6}$$

Since $\dot{\boldsymbol{\alpha}} = \eta_p(\dot{\mathbf{x}}_r - \dot{\mathbf{x}}) + \eta_d(\ddot{\mathbf{x}}_r - \ddot{\mathbf{x}})$ and $\dot{\mathbf{x}} = \mathbf{J}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}}$, then

$$\dot{\boldsymbol{\alpha}} = \overbrace{\eta_p\left(\dot{\mathbf{x}}_r - \mathbf{J}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}}\right) + \eta_d\left(\ddot{\mathbf{x}}_r - \dot{\mathbf{J}}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}}\right)}^{\boldsymbol{\delta}(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\dot{\mathbf{x}}_r,\ddot{\mathbf{x}}_r)} - \eta_d\mathbf{J}(\boldsymbol{\Theta})\ddot{\boldsymbol{\Theta}}, \tag{7}$$

Thus (6) will become

$$\dot{\boldsymbol{\beta}} = \overbrace{\mathbf{J}^{*}(\boldsymbol{\Theta})\boldsymbol{\delta}\left(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\dot{\mathbf{x}}_r,\ddot{\mathbf{x}}_r\right) + \dot{\mathbf{J}}^{*}(\boldsymbol{\Theta})\boldsymbol{\alpha} + \boldsymbol{K}\left(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\mathbf{v},\dot{\mathbf{v}}\right)}^{\mathbf{A}(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\mathbf{v},\dot{\mathbf{v}},\dot{\mathbf{x}}_r,\ddot{\mathbf{x}}_r)} - \eta_d\mathbf{J}^{*}(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta})\ddot{\boldsymbol{\Theta}}, \tag{8}$$

where the items $\boldsymbol{\delta}$, $\dot{\mathbf{J}}^{*}$, $\boldsymbol{K}$, which are related to $\boldsymbol{\Theta}$, $\dot{\boldsymbol{\Theta}}$, $\mathbf{v}$, $\dot{\mathbf{v}}$, $\dot{\mathbf{x}}_r$ and $\ddot{\mathbf{x}}_r$, are highly complex nonlinear and difficult to obtain the accurate values due to the noise in the practical application. Meanwhile, these items consume a lot of computation when the DoFs of the manipulator are relatively large. Then the highly complex nonlinear items related to $\boldsymbol{\Theta}$, $\dot{\boldsymbol{\Theta}}$, $\mathbf{v}$, $\dot{\mathbf{v}}$, $\dot{\mathbf{x}}_r$ and $\ddot{\mathbf{x}}_r$, are simplified and considered as a synthetic system perturbation $\mathbf{A}$, $\mathbf{A} \in \mathbb{R}^{s\times 1}$. $\mathbf{A}$ is bounded, uncertain and not infinity, and approaches an arbitrarily small range near 0 for finite time in a real system. Thus, the real-time planning problem is turned into the nonlinear control problem about angular acceleration and $\mathbf{A}$ is compensated via the designed hyperbolic tangent-based STA in (11), which differs from the existing study evidently.

Because of the null-space vector in the system error $\boldsymbol{\beta}$ formed by IK, the null-space-based kinematic redundancy is still possessed for avoiding obstacles in the planned angular acceleration as provided in *Part* C of Section IV.

C. Design of Hyperbolic Tangent-based STA

Let $\mathbf{u} = \eta_d \cdot \mathbf{J}^{*}(\boldsymbol{\Theta})\mathbf{J}(\boldsymbol{\Theta})\ddot{\boldsymbol{\Theta}}$ in (8), then $\dot{\boldsymbol{\beta}}$ becomes

$$\dot{\boldsymbol{\beta}} = \mathbf{A}\left(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\mathbf{v},\dot{\mathbf{v}},\dot{\mathbf{x}}_r,\ddot{\mathbf{x}}_r\right) - \mathbf{u}, \tag{9}$$

where $\mathbf{u} \in \mathbb{R}^{s\times 1}$ refers to the control input, $\mathbf{u} = (u_1, u_2, \ldots, u_s)^{\mathrm{T}}$. The synthetic system perturbation $\mathbf{A} = (A_1, A_2, \ldots, A_s)^{\mathrm{T}}$. $\ddot{\boldsymbol{\Theta}}$ is the planned angular acceleration.

In the torque control of nonlinear dynamics system, the traditional STA has been widely applied and owns the fast convergence, robustness, simplicity and high control accuracy. And it could suppress the chattering of traditional sliding mode control effectively [34, 35]. When the traditional STA is used as the control input $\mathbf{u}$, per element in $\mathbf{u}$ can be described as follows:

$$u_i = k_a \cdot |\beta_i|^{\frac{1}{2}} \cdot \operatorname{sgn}(\beta_i) + \frac{k_r}{2}\int_0^t \operatorname{sgn}(\beta_i)\, dt\,, \tag{10}$$

where i = 1, 2, 3, …, s.

However, chattering occurs in real-time planning of kinematics-based robot manipulator system when the traditional STA is used in motion planning. To improve the situation and apply the STA in real-time motion planning, $\operatorname{sgn}(\beta_i)$ in (10) is replaced by the hyperbolic tangent function $\tanh(\rho\beta_i)$. The hyperbolic tangent-based STA is similar to sliding mode control and is utilized to make $\dot{\boldsymbol{\beta}}$ converge in this paper, so per element of the control input $\mathbf{u}$ becomes

$$u_i = k_a \cdot |\beta_i|^{\frac{1}{2}} \cdot \tanh(\rho\beta_i) + \frac{k_r}{2}\int_0^t \tanh(\rho\beta_i)\, dt\,, \tag{11}$$

where k_a, k_r and ρ denote the constant coefficients. The hyperbolic tangent-based STA can resist the equivalent synthetic perturbation and effectively suppress the chattering in planning. The contrast experiments are presented in *Part* B of Section IV.

***Remark* 1**. The description by the differential inclusion for hyperbolic tangent-based STA is

$$\begin{cases} \dot{x}_{i,1} = \dot{\beta}_i = -k_a \cdot |\beta_i|^{\frac{1}{2}} \cdot \tanh(\rho\beta_i) + x_{i,2} + A_i \\ \dot{x}_{i,2} = -\dfrac{k_r}{2}\tanh(\rho\beta_i) \end{cases},$$

where $x_{i,1}$, $x_{i,2}$ are the scalar state variables. The synthetic system perturbation A_i is bounded and uncertain, and could approach an arbitrarily small range near 0 in finite time. i = 1, 2, 3, …, s. The trajectories of the system errors cross the super-twisting surface gently, and stay on it when $\beta_i = 0$ has been reached. A Lyapunov function is proposed to ensure the convergence in finite time of all error trajectories to zero,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

when k_a and k_r are adequately selected, and for some kinds of perturbations as shown in (18).

With the consideration of $\mathbf{J}^*\mathbf{J} \approx \mathbf{I}$ in $\mathbf{u}$, $\mathbf{u} = \eta_d \cdot \ddot{\mathbf{\Theta}}$ and the control input $\mathbf{u}$ formed by the hyperbolic tangent-based STA is propositional to the angular acceleration. So $\mathbf{u}$ plans the joint angular acceleration directly and the planned joint velocity $\dot{\mathbf{\Theta}}_d$ is achieved by the integral equation of $\mathbf{u}$

$$\dot{\mathbf{\Theta}}_d = \varrho \cdot \int_0^t \mathbf{u}\,, \tag{12}$$

where ϱ, as the internal parameter, represents the error-based S-function for the manipulator and is used to prevent the integral saturation caused by real-time integration. ϱ is

$$\varrho = \frac{\|\boldsymbol{\alpha}\| + \|\mathbf{v}\|}{\eta_d(\|\boldsymbol{\alpha}\| + \|\mathbf{v}\| + \epsilon)}, \tag{13}$$

where ϵ is the positive real number for adjusting the S-function slope.

The planned joint position $\mathbf{\Theta}_d$ is obtained by the interval method as follows:

$$\mathbf{\Theta}_{d+1} := \mathbf{\Theta}_d + \Delta\mathbf{\Theta}_d, \tag{14}$$

where $\Delta\mathbf{\Theta}_d = \dot{\mathbf{\Theta}}_d \cdot \Delta\mathrm{T}$. $\Delta\mathrm{T}$ denotes the sampling time interval.

Based on the above theoretical derivation in Parts A and B, the proposed real-time angular acceleration planning in joint space is thus achieved and could be as a motion planning framework that considering the "joint velocity" as the "system error" to deduce the joint angular acceleration for the real-time operation/tracking tasks of redundant manipulator in dynamic environment as shown in Fig. 3.

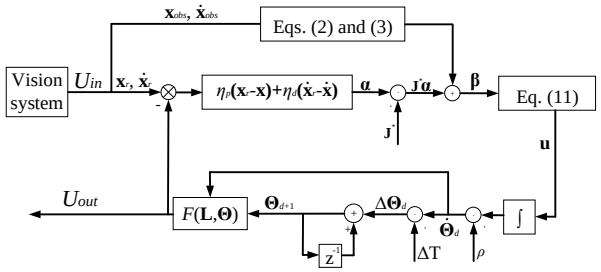


FIGURE 3. The angular acceleration planning method for joint configuration spaces of kinematically redundant robot manipulators in dynamic environment. $F(\mathbf{L},\mathbf{\Theta})$ denotes the FK. U_{in} defines $(\mathbf{x}_r, \dot{\mathbf{x}}_r, \mathbf{x}_{obs}, \dot{\mathbf{x}}_{obs})$. $\mathbf{x}_{obs}$, $\dot{\mathbf{x}}_{obs}$ refer to the position and velocity of obstacles, respectively. U_{out} represents the EE pose.

D. Stability Analysis

Let $\boldsymbol{\xi}$ be defined as follows:

$$\boldsymbol{\xi} = \begin{bmatrix}\xi_1 \\ \xi_2\end{bmatrix} = \begin{bmatrix} |\beta_i|^{\frac{1}{2}} \cdot \tanh(\rho\beta_i) \\ \dfrac{k_r}{2}\displaystyle\int_0^t \tanh(\rho\beta_i)dt \end{bmatrix}, \tag{15}$$

Then

$$\dot{\boldsymbol{\xi}} = \begin{bmatrix}\dot{\xi}_1 \\ \dot{\xi}_2\end{bmatrix} = \begin{bmatrix} \dfrac{|\beta_i|^{-\frac{1}{2}}}{2}\tanh(\rho|\beta_i|)\,\dot{\beta}_i + |\beta_i|^{\frac{1}{2}}[1-\tanh^2(\rho\beta_i)]\rho\dot{\beta}_i \\ \dfrac{k_r}{2}\tanh(\rho\beta_i) \end{bmatrix}$$

$$= \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \boldsymbol{\xi} + \begin{bmatrix} m \\ n \end{bmatrix}, \tag{16}$$

where

$$a = -|\beta_i|^{-\frac{1}{2}}\left\{\frac{k_a}{2}\tanh(\rho|\beta_i|) + |\beta_i|[1-\tanh^2(\rho\beta_i)]\rho k_a\right\},$$

$$b = -|\beta_i|^{-\frac{1}{2}}\overbrace{\left\{\frac{1}{2}\tanh(\rho|\beta_i|) + |\beta_i|[1-\tanh^2(\rho\beta_i)]\rho\right\}}^{b'},$$

$$c = \overbrace{|\beta_i|^{-\frac{1}{2}} \cdot \frac{k_r}{2}}^{c'},$$

$$d = 0,$$

$$m = \left\{\frac{|\beta_i|^{-\frac{1}{2}}}{2}\tanh(\rho|\beta_i|) + |\beta_i|^{\frac{1}{2}}[1-\tanh^2(\rho\beta_i)]\rho\right\}A_i,$$

$$n = 0.$$

Then the Lyapunov function $\mathbf{L}$ is defined as follows

$$\mathbf{L} = (L_1, L_2, L_3, \ldots, L_s)^{\mathrm{T}}, \qquad \mathbf{L} \in \mathbb{R}^{s\times 1}. \tag{17}$$

As the hyperbolic tangent-based STA is proposed on the basis of the traditional STA and similar to the high-order sliding mode, per element L_i (i = 1, 2, 3, …, s) in $\mathbf{L}$ is designed as follows:

$$L_i = \frac{k_r}{\rho}\ln[\cosh(\rho\beta_i)] + \frac{1}{2}\xi_2^2 + \frac{1}{2}(k_a\xi_1 + \xi_2)^2 \geq 0, \tag{18}$$

In (18), L_i is continuously differentiable for every point, and on those points the derivative can be calculated via applying Lyapunov's theorem. Also, L_i in (18) is beneficial to the system convergence. Then,

$$\dot{L}_i = \frac{k_r}{\rho}\tanh(\rho\beta_i)\,\rho\dot{\beta}_i + \xi_2\dot{\xi}_2 + (k_a\xi_1 + \xi_2)\cdot\left(k_a\dot{\xi}_1 + \dot{\xi}_2\right). \tag{19}$$

Assume that

$$|A_i| \leq \delta_2|\xi_2|, |A_i| \leq \delta_1|\xi_1|, |\xi_2||\xi_2| \leq \frac{1}{\delta_3}|\xi_1||\xi_1|,$$

where the synthetic system perturbation A_i is bounded, then

$$\dot{L}_i \leq -|\beta_i|^{-\frac{1}{2}} \cdot \boldsymbol{\xi}^T \cdot \overbrace{\begin{bmatrix} k_a^3 b' + \dfrac{k_r}{2}k_a - k_a b'\delta_2\dfrac{1}{\delta_3} - k_r\delta_1 - k_a^2 b'\delta_1 & k_a^2 b' \\ k_a^2 b' & k_a b' \end{bmatrix}}^{\mathbf{D}}\boldsymbol{\xi}, \tag{20}$$

In order to ensure that $\mathbf{D}$ is a positive definite matrix, the conditions $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ need to be satisfied as follows:

$$\boldsymbol{\mathcal{A}}: \quad k_a^3 b' + \frac{k_r}{2}k_a - k_a b'\delta_2\frac{1}{\delta_3} - k_r\delta_1 - k_a^2 b'\delta_1 > 0, \tag{21}$$

$$\boldsymbol{\mathcal{B}}: \quad \left(k_a^3 b' + \frac{k_r}{2}k_a - k_a b'\delta_2\frac{1}{\delta_3} - k_r\delta_1 - k_a^2 b'\delta_1\right)k_a b' - (k_a^2 b')^2 > 0, \tag{22}$$

Then k_a and k_r can be obtained as follows:

$$\begin{cases} k_a > 2\delta_1 \\ k_r > 1.6016 \cdot \dfrac{k_a^2\cdot\delta_1 + k_a\cdot\delta_2\cdot\dfrac{1}{\delta_3}}{k_a - 2\delta_1} \end{cases}, \tag{23}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

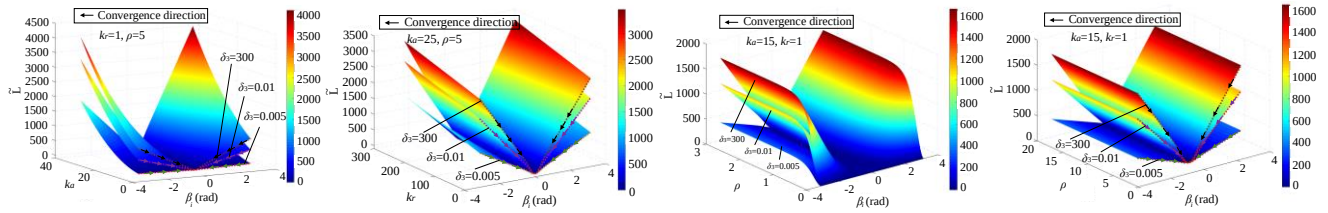


FIGURE 4. The convergence analysis for different parameters. (a) Influence of the k_a on $\tilde{L}$ at different β_i. (b) $\tilde{L}$ varies with k_r and β_i. (c) Relationship between ρ ($0<\rho<3$) and $\tilde{L}$ for different β_i. (d) Relationship between ρ ($\rho>3$) and $\tilde{L}$ for different β_i.

Thus,

$$\dot{\mathbf{L}} \le \mathbf{0}. \tag{24}$$

For the system, $\beta_i = 0$ is an equilibrium point that is strongly globally asymptotically stable if (23) satisfies. Therefore, the Lyapunov function **L** is convergent. The detailed process of (20)-(23) is provided in Appendix.

E. Influence of Parameters on Convergence

The direct analysis for the influence of parameters k_a, k_r, and ρ on the convergence of manipulator becomes quite difficult by using the designed STA due to the existence of integral link ξ_2 and the hyperbolic tangent terms. In order to display these parameters' influences on the convergence performance intuitively, L_i is further deduced to be

$$\begin{aligned} L_i &= \frac{k_r}{\rho}\cdot\ln[\cosh(\rho\beta_i)] + \frac{1}{2}\cdot\xi_2^2 + \frac{1}{2}\cdot(k_a\xi_1+\xi_2)^2 \\ &\le \frac{k_r}{\rho}\cdot\ln[\cosh(\rho\beta_i)] + \frac{1}{2}\cdot\frac{1}{\delta_3}\cdot\xi_1^2 + \frac{1}{2}\cdot(k_a|\xi_1|+|\xi_2|)^2 \\ &\le \frac{k_r}{\rho}\cdot\ln[\cosh(\rho\beta_i)] + \frac{1}{2}\frac{1}{\delta_3}\xi_1^2 + \frac{1}{2}\left(k_a|\xi_1| + \frac{1}{\sqrt{\delta_3}}\cdot|\xi_1|\right)^2 \\ &= \underbrace{\frac{k_r}{\rho}\cdot\ln[\cosh(\rho\beta_i)] + \frac{1}{2}\cdot\left[\frac{1}{\delta_3} + \left(k_a + \frac{1}{\sqrt{\delta_3}}\right)^2\right]\cdot\xi_1^2}_{\tilde{L}}, \end{aligned} \tag{25}$$

where $\tilde{L}$ is a function associated with β_i and defined as the upper boundary of L_i.

***Remark* 2**. For (25), $|\xi_2||\xi_2| \le \frac{1}{\delta_3}|\xi_1||\xi_1|$, $|\xi_2| \le \frac{1}{\sqrt{\delta_3}}|\xi_1|$.

Combining (18) and (25), the range of Lyapunov function L_i can be gained as follows:

$$0 \le L_i \le \tilde{L}. \tag{26}$$

According to (25), as β_i approaches 0, $\tilde{L}$ also equals to 0. Then for L_i in (26), when β_i approaches 0, there is

$$\lim_{\beta_i\to 0} L_i = \lim_{\beta_i\to 0} \tilde{L} = 0. \tag{27}$$

Thus, the upper boundary $\tilde{L}$ of L_i is used to indirectly analyze the influence of parameters on convergence performance in this paper. For different δ_3, the influences of parameters k_a, k_r, and ρ on the convergence of $\tilde{L}$ are shown in Fig. 4. Fig. 4(a) shows the influence of k_a on $\tilde{L}$ at different β_i. The convergence rate of $\tilde{L}$ increases gradually with the increase of k_a. When β_i becomes small, $\tilde{L}$ convergences slowly. As illustrated in Fig. 4(b), the convergence rate of $\tilde{L}$ increases slowly with the increase of k_r and is almost linear along with the changes of β_i. Figs. 4(c) and (d) display the relationship between ρ and $\tilde{L}$ for different β_i. ρ ($\rho\ge3$) has nearly the same convergence performance. The convergence rate of $\tilde{L}$ increases gradually with the increase of ρ ($0<\rho<3$). Therefore, a large value for k_a, k_r, and ρ is in favor of the fast convergence.

III. SIMULATION

The simulation of a 7-DoF manipulator demonstrates the feasibility and smoothness of the proposed angular-acceleration planning method by real-time tracking a static object in obstacle-free case as shown in Fig. 5(a). The influences of the parameters k_a, k_r, ρ, η_p and η_d on the length of EE motion path are analyzed when the object is given as shown in Figs. 5(b)-(f). Meanwhile, the real-time performance of the proposed method is discussed by combining with the convergence performance presented in Part E of Section II. In Fig. 5(b), k_a has great influence on path trajectory. The larger k_a is selected, the straighter the path trajectory of EE motion is. Fig. 5(c) shows that k_r should be a small value when other parameters are certain and k_r has less influence on the whole path trajectory of EE. Fig. 5(d) displays that the motion path of EE would get longer with ρ increasing and the trajectories of motion paths have roughly similar patterns for different ρ, which is corresponding to the Fig. 4(d). Fig. 5(e) provides that a small change of η_p can cause a great change in the path trajectory, which is similar to η_d as illustrated in Fig. 5(f). P_L increases with η_p increasing and decreases with η_d increasing. Figs. 5(b), (d) and (f) show that P_L becomes shorter gradually with the increase of k_a, ρ and η_d. Figs. 5(c) and (e) display that P_L get longer gradually with the growing of k_r and η_p. In practical application, the fast convergence, the short and smooth path of the EE and the smooth joint trajectory should be considered at the same time. The proposed angular-acceleration planning method could endow the manipulator system with smooth joint trajectory that can be executed fast and with a high accuracy as shown in Section IV, then the shorter and smooth path and the fast convergence rate are beneficial to the manipulator with better real-time motion performance and efficiency to accomplish the task. Considering Figs. 4 and 5 comprehensively, the large values of k_a, ρ, η_d and small values of k_r, η_p contribute to the real-time performance of the proposed method. Therefore, k_a, ρ and η_d should be chosen with a relatively larger value, and k_r and η_p should be set with a relatively smaller value to guarantee the real-time motion performance of the robot system under the condition of convergence.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

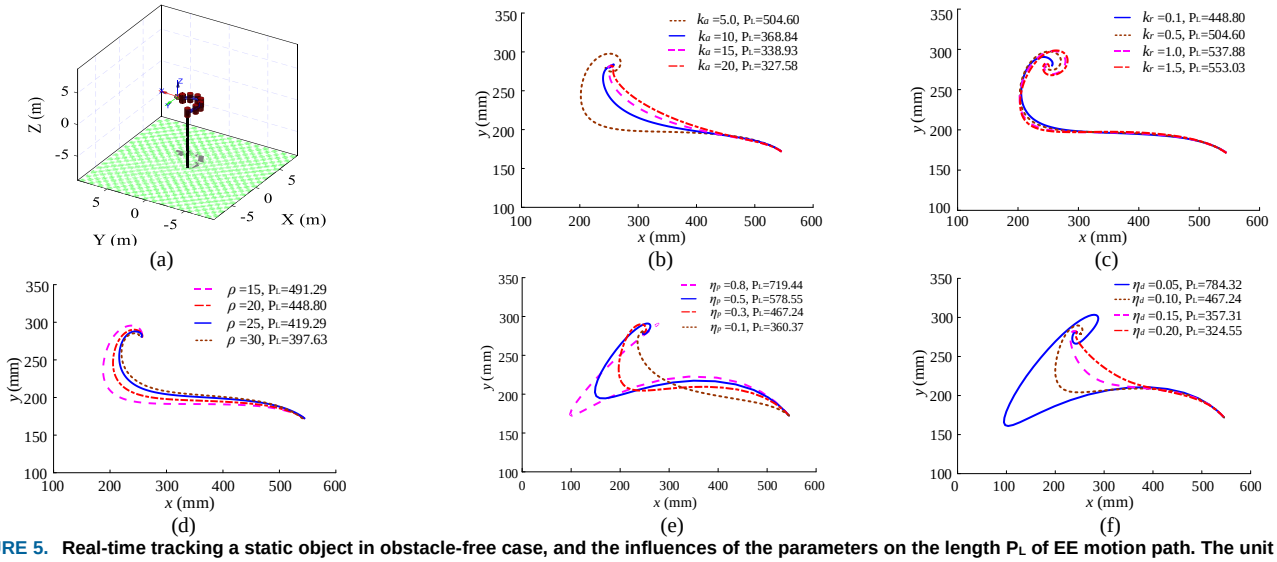


FIGURE 5. **Real-time tracking a static object in obstacle-free case, and the influences of the parameters on the length P_L of EE motion path. The unit of P_L is mm. The initial configuration is $\Theta_{initial}$ = (-5°, 5°, 5°, 10°, 10°, 20°, 20°)^T. The object position is x_r = (249 mm, 279mm)^T. (a) Real-time tracking a static object based on a 7-DoF manipulator. (b) The influence of k_a on the path length. k_r=0.5, ρ=20, η_p=0.1, η_d=0.1. (c) The influence of k_r on the path length. k_a=5, ρ=20, η_p=0.1, η_d=0.1. (d) The influence of ρ on the path length. k_a=5, k_r=0.1, η_p=0.1, η_d=0.1. (e) The influence of η_p on the path length. k_a=10, k_r=0.1, ρ=20, η_d=0.1. (f) The influence of η_d on the path length. k_a=10, k_r=0.1, ρ=20, η_p=0.3.**

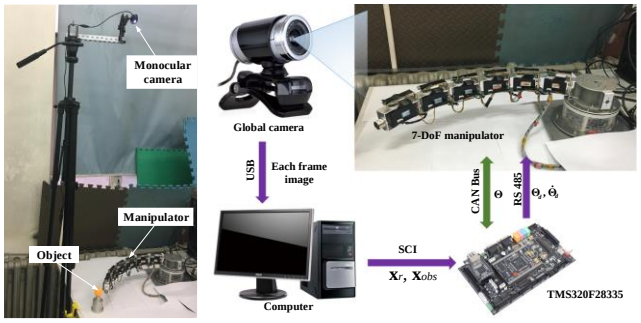


FIGURE 6. **Experimental setup. (a) Experimental platform. (b) Principle.**

IV. EXPERIMENTS

A. Experimental Setup Description

Fig. 6(a) shows the experimental platform. The motion control of the manipulator is achieved by applying the planned joint position and velocity commands. In Fig. 6(b), the principle of experimental setup can be briefly described as follows. 1) The global camera transfers the observed each frame image to the computer using universal serial bus (USB) in real time. The computer extracts the position information of the object and obstacles from each frame image, and transfers these position information to the TMS320F28335 controller through serial communication interface (SCI) bus. The position and velocity of the EE are calculated by the forward kinematics. The position and velocity of the obstacle and object are measured by the vision system. The vision processing procedures in the computer are developed based on morphology using Open Source Computer Vision Library. The measurement noise is suppressed by Gaussian filter. 2) The state feedback data $\boldsymbol{\Theta}$ of the manipulator run on controller area network (CAN) bus. The planned joint position $\boldsymbol{\Theta}_d$ and velocity $\dot{\boldsymbol{\Theta}}_d$ control the manipulator motion by RS485 bus. The baud rates of CAN bus, USB bus and SCI bus are set as 1 MHz. 3) The TMS320F28335 controller receives the data ($\mathbf{x}_r$, $\mathbf{x}_{obs}$, $\boldsymbol{\Theta}$). Meanwhile, the controller with a dominant frequency of 150 MHz carries and executes the program including FK, IK, motion planning algorithm. Tables 1 provides the parameters of the 7-DoF manipulator.

TABLE 1. **Parameters in the 7-DoF Manipulator.**

Parameters	l_1	l_2	l_3	l_4	l_5	l_6	l_7
Value (mm)	118.0	88.0	88.0	88.0	88.0	88.0	57.85

TABLE 2. **Parameters of the Proposed Algorithm.**

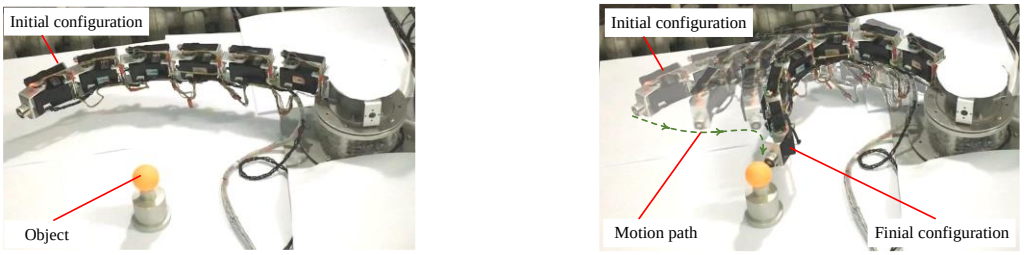
Parameters	k_a	k_r	ρ	η_p	η_d	λ^2	ϵ
Value	25.0	1.0	15.0	0.05	0.05	0.01	3

B. Static Object-Tracking in Obstacle-Free Environment

The control methods used for resisting the system synthetic perturbation **A** of the kinematically manipulator are compared by tracking a static object in obstacle-free environment, including the proposed STA and the traditional STA in (10) [34, 35]. The traditional fixed proportion-based planning method [16] is also used to compare with the proposed angular-acceleration planning method. To demonstrate the rapidity and smoothness of the proposed angular-acceleration planning method appropriately, the pose convergence time using the traditional STA-based, fixed proportion-based methods and the proposed method are set about 12s. The initial configuration of the manipulator and the tracked state of the object are set to the same. The parameters of the proposed angular-acceleration planning algorithm are shown in Table 2 and the experiments of tracking the static object are displayed in Fig. 7.

In Fig. 8(a) and Fig. 9(a), the proposed hyperbolic tangent-based STA and the traditional STA can plan smooth joint trajectories for the manipulator to track the static object and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access



(a) (b)

FIGURE 7. Tracking a static object in obstacle-free environment. (a) Initial configuration. (b) Real-time tracking based on the proposed method and final configuration.

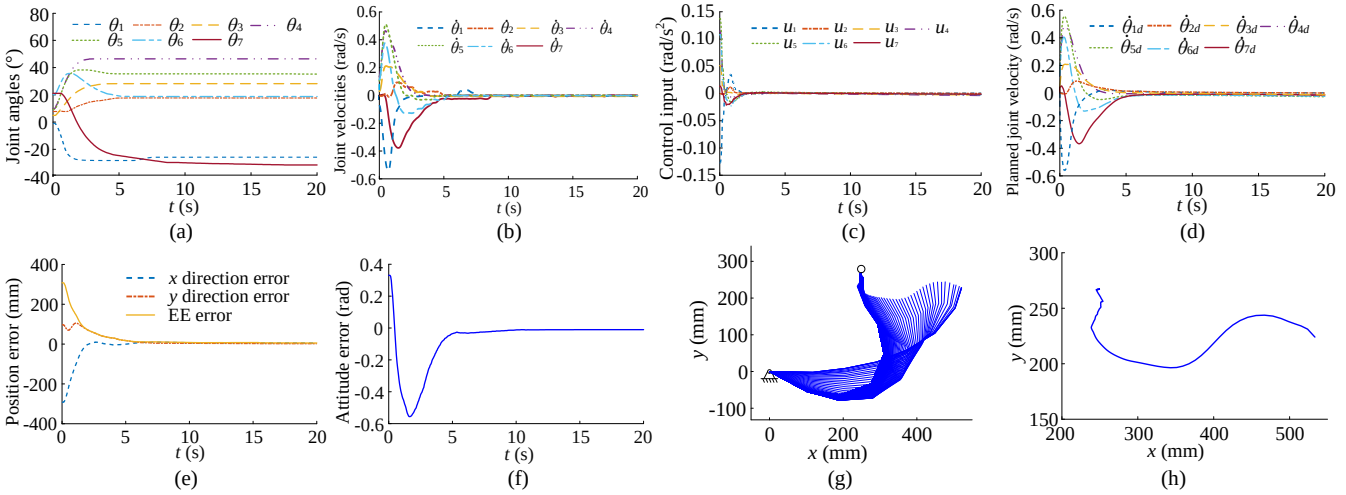


FIGURE 8. Object tracking in obstacle-free environment based on the proposed angular-acceleration planning method. Initial joint angle $\Theta_{initial}$ = (-1°, 5°, 8°, 10°, 10°, 20°, 20°)^T. x_r = (249 mm, 279mm, 1.605 rad)^T. (a) Joint angles. (b) Joint velocities. (c) Control input u, (= η_d × planned joint acceleration $\ddot{\Theta}$). (d) Planned joint velocity. (e) Position error of EE. (f) Attitude error of EE. (g) Motion of manipulator. (h) Motion path of EE.

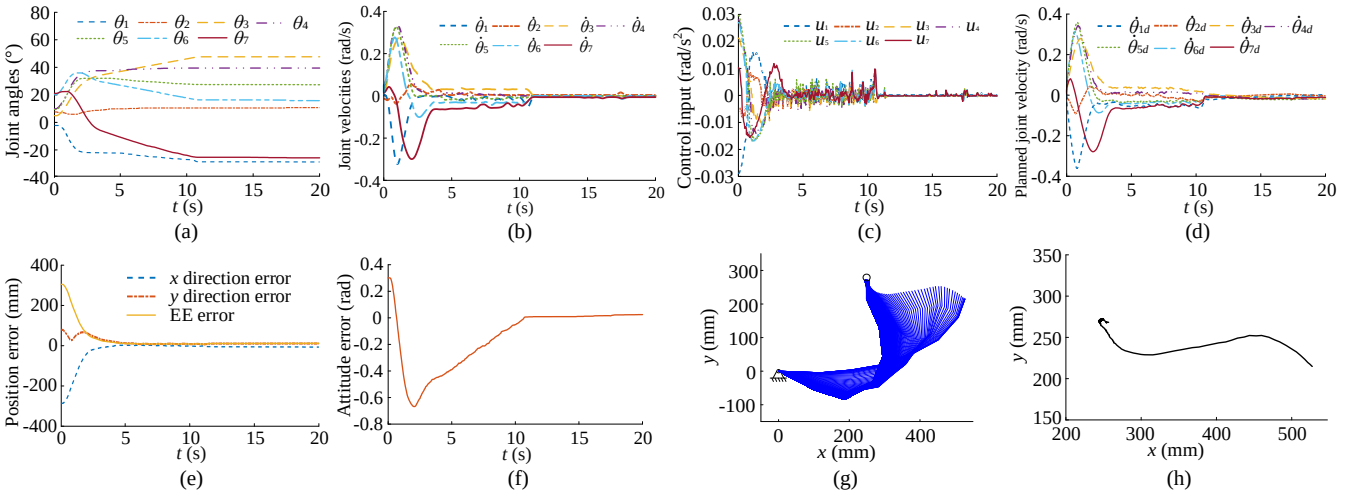


FIGURE 9. Object tracking in obstacle-free environment based on the traditional STA. Initial joint angle $\Theta_{initial}$ = (-1°, 5°, 8°, 10°, 10°, 20°, 20°)^T. x_r = (249 mm, 279mm, 1.605 rad)^T. (a) Joint angles. (b) Joint velocities. (c) Control input u, (= η_d × planned joint acceleration $\ddot{\Theta}$). (d) Planned joint velocity. (e) Position error of EE. (f) Attitude error of EE. (g) Motion of manipulator. (h) Motion path of EE.

the joint motion reaches a steady state eventually through the real-time integration of $\mathbf{u}$. According to (9) and $\mathbf{J}^*\mathbf{J} \approx \mathbf{I}$ in $\mathbf{u}$, $\mathbf{u} = \eta_d \cdot \ddot{\mathbf{\Theta}}$ and the contour shapes of $\mathbf{u}$ and planned joint acceleration $\ddot{\mathbf{\Theta}}$ are similar and proportional as shown in Figs 8(c) and 9(c). In Figs. 8(b)-(d) and Figs. 9(b)-(d), the joint velocities, control input (or planned angular acceleration) and planned joint velocities generated by using the proposed hyperbolic tangent-based STA are steadier and less jitter than those by using the traditional STA, even though their maximum values based on the proposed algorithm are much higher. The pose errors can convergence to 0 (or close to 0) in limited time as shown in Figs 8(e), 8(f), 9(e) and 9(f). The motions of manipulator are shown in Figs. 8(g) and 9(g). The corresponding motion paths of EE are provided in Figs. 8(h) and 9(h).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

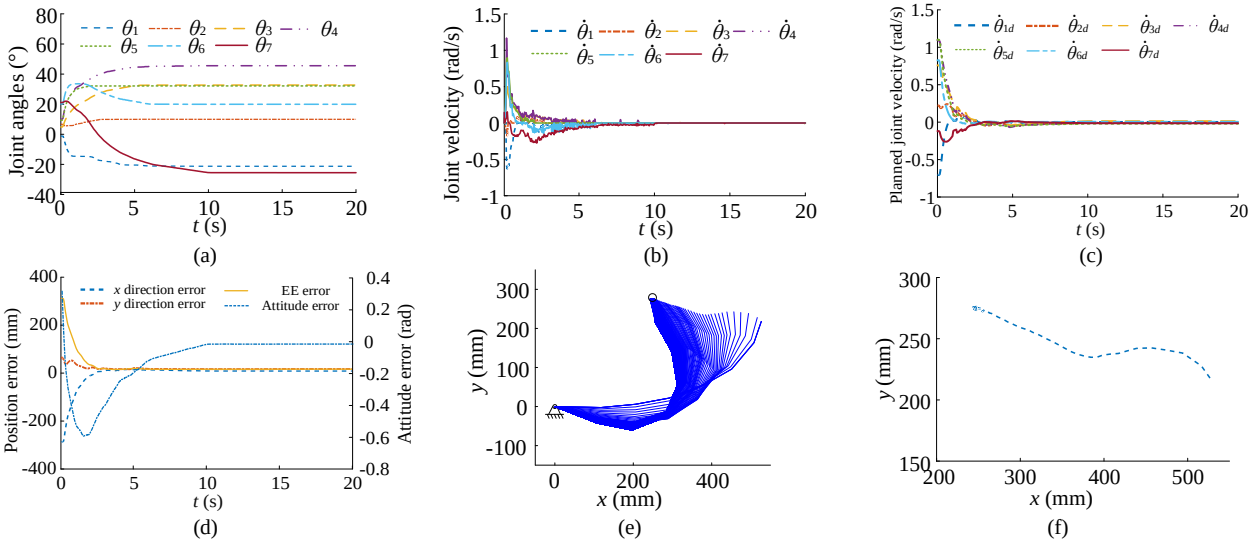


FIGURE 10. Object tracking in obstacle-free environment based on the fixed proportion method. Initial joint angle $\Theta_{initial}$ = (-1°, 5°, 8°, 10°, 10°, 20°, 20°)^T. x_r = (249 mm, 279mm, 1.605 rad)^T. (a) Joint angles. (b) Joint velocities. (c) Planned joint velocities. (d) Pose error of EE. (e) Motion of manipulator. (f) Motion path of EE.

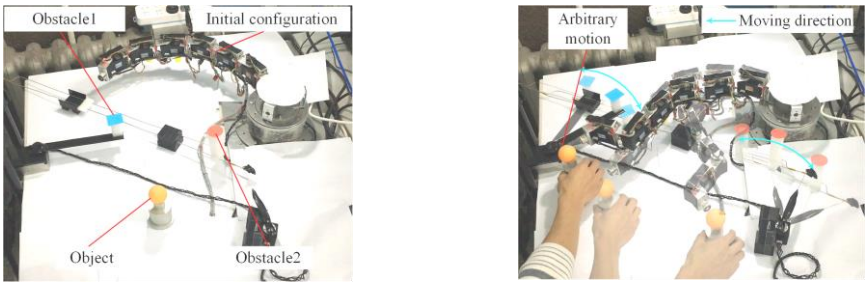


FIGURE 11. Real-time tracking of a moving object based on the proposed angular-acceleration planning method in dynamic-obstacle environment. Initial joint angle $\Theta_{initial}$ = (-45°, 12°, 8°, 25°, 13°, 25°, 35°)^T. x_r = (210 mm, 316 mm)^T. x_{obs1} = (416 mm, -172 mm)^T. x_{obs2} = (106 mm, 164 mm)^T. (a) Initial configuration of the manipulator, and layout of the object and the obstacles in the initial stage of motion. (b) The tracking process of the manipulator in real time.

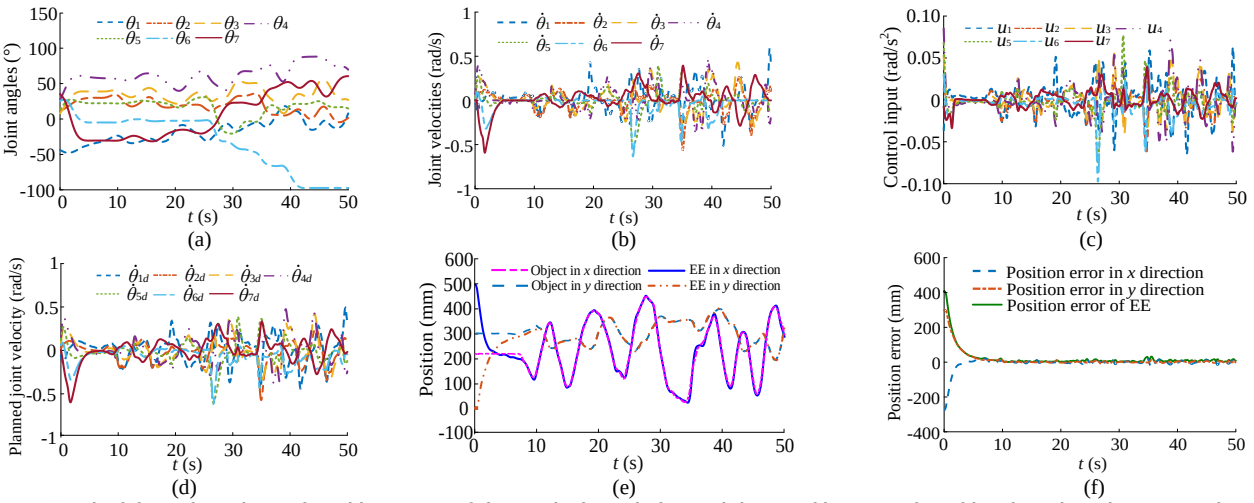


FIGURE 12. The joint trajectories and position errors of the manipulator during real-time tracking a moving object based on the proposed angular-acceleration planning method in dynamic environment. (a) Joint angle position. (b) Joint velocities. (c) Control input u, (= η_d × planned joint acceleration $\ddot{\theta}$). (d) Planned joint velocities. (e) The position of EE and object in Cartesian space. (f) Position error of EE.

Fig. 10 shows the real-time tracking of the static object based on the fixed proportion method. The joint angle trajectories are smooth, but the joint velocities have large steps in initial motion stage and have some jitters in the motion process as shown in Figs. 10(a)-(c). The pose errors are provided in Fig. 10(d). The motion process of manipulator and the corresponding EE motion path are shown in Figs. 10(e) and (f), respectively.

Comprehensive analysis of three methods, the lengths of the motion paths that are ranked from long to short are the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

proposed algorithm-based, the traditional algorithm-based, and the fixed proportion-based methods, respectively. Although the proposed method causes a little long path, its convergence rate and smoothness are much better than those of the other two methods. Therefore, the proposed angular acceleration planning method is feasible and smooth and owns certain application potential.

C. Moving Object-tracking in Dynamic-Obstacle Environment

The final experiment is used to demonstrate the feasibility, safety and smoothness of the proposed real-time angular-acceleration planning method in dynamic-obstacle environment. Table 3 provides the parameters of the proposed angular-acceleration planning method and Fig. 11 presents the real-time collision-free tracking experiment of a moving object in dynamic-obstacle environment. Fig. 11(a) shows the initial configuration of manipulator and the layout of object and obstacles in the initial stage of motion. Fig. 11(b) displays the tracking process of the manipulator in real time.

TABLE 3. Parameters of the Proposed Algorithm.

Parameters	k_a	k_r	ρ	η_p	η_d	λ^2	ϵ
Value	19.0	0.31	18.0	0.05	0.05	3	3

Within 6.5 seconds of the initial stage, the object remains static and the obstacles keep moving in according with certain rules as shown in Fig. 11(b) and Fig. 12(e). When the object is almost to be tracked, it then moves arbitrarily. The joint angles, joint velocities, control input (or planned angular acceleration) and planned joint velocities of the manipulator are provided in Figs. 12(a)-(d). Because the planned joint velocities of the proposed acceleration planning method is obtained by the real-time integration of control input $\mathbf{u}$, the joint angles and velocities of the manipulator are smooth and steady, which contribute to the real-time and smooth operations of the manipulator in practical application. Due to the great tracking ability of the commercial driver by applying the joint position and velocity commands, the planned joint velocities and the actual joint velocities are almost consistent and identical as shown in Figs. 12(b) and (d), respectively. The positions of the object and EE in Cartesian space are illustrated in Fig. 12(e). In the initial stage, the EE and the object remain a certain distance with each other. The initial position errors are -282.8 mm in x direction and 338.8 mm in y direction, respectively. With the continuous calculations of joint control input $\mathbf{u}$ in (11) and integration of control input $\mathbf{u}$ in (12), the real-time tracking of the object is achieved steadily by driving the manipulator. The position error of EE in Cartesian space convergences gradually as shown in Fig. 12(f). Due to the null space vector in (2) and (3), the redundancy for avoiding obstacles is still possessed in the planned angular acceleration. Thus the collision-free motion of the redundant manipulator is ensured as shown in Fig. 11(b). Therefore, the proposed angular-acceleration planning method with safety, smoothness and little-computations contributes to reducing the robot hardware costs and guaranteeing the stable and real-time safely tracking tasks of the manipulators in practical application.

V. CONCLUSION

This paper presented a practical real-time joint angular-acceleration planning method for vision-based kinematically redundant manipulator to achieve the tracking task in dynamic environment. The proved proposition considers the joint velocity formed by the IK mapping as the system error so that the planned joint angular-acceleration was further deduced. The angular-acceleration planning process with null space for the obstacle avoidance was simplified via a system synthetic perturbation instead of the highly-complex nonlinearities. Such considerations could avoid the indirectly obtained joint input for Cartesian motion planning and the non-intuitive the task description for joint trajectory planning. Meanwhile, the complexity of planning method was reduced greatly in theory. Different with the ideas of the existing methods, the real-time motion planning was converted into a control problem and dealt with by using the control technology. The hyperbolic tangent-based STA that was designed for the control input of manipulator was utilized to resist the synthetic perturbation and overcame the chattering in motion planning. The control input was integrated to obtain the real-time planned joint velocities of the manipulator and generate a smooth trajectory for a collision-free tracking task by combining with an error-based S-function. Theory analysis displayed the small computational cost and the convergence of the proposed planning method. Simulation and contrast experiments verified the feasibility, smoothness and practicability of the proposed joint angular acceleration planning method.

APPENDIX

A. Derivation of $\dot{L}_i$

According to (16), (19), $\dot{L}_i$ can be deduced as follow:

$$\dot{L}_i = \frac{k_r}{\rho}\tanh(\rho\beta_i)\,\rho\dot{\beta}_i + \xi_2\dot{\xi}_2 + (k_a\xi_1 + \xi_2)\cdot\left(k_a\dot{\xi}_1 + \dot{\xi}_2\right).$$

$$= |\beta_i|^{-\frac{1}{2}}\left\{\boldsymbol{\xi}^T\begin{bmatrix} k_a^3(-b') + k_a c' - k_r k_a & k_a^2(-b') + c' - \frac{k_r}{2} \\ k_a^2(-b') + c' - \frac{k_r}{2} & k_a(-b') \end{bmatrix}\boldsymbol{\xi} + k_a b' \cdot A_i \cdot \xi_2 + (k_r + k_a^2 \cdot b') \cdot A_i \cdot \xi_1\right\}$$

Since $|A_i| \le \delta_2|\xi_2|, |A_i| \le \delta_1|\xi_1|, |\xi_2||\xi_2| \le \frac{1}{\delta_3}|\xi_1||\xi_1|,$ then

$$\dot{L}_i \le |\beta_i|^{-\frac{1}{2}}\left\{\boldsymbol{\xi}^T\begin{bmatrix} k_a^3(-b') + k_a c' - k_r k_a & k_a^2(-b') + c' - \frac{k_r}{2} \\ k_a^2(-b') + c' - \frac{k_r}{2} & k_a(-b') \end{bmatrix}\boldsymbol{\xi} + k_a b' \cdot |A_i| \cdot |\xi_2| + (k_r + k_a^2 b') \cdot |A_i| \cdot |\xi_1|\right\}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

$$\leq |\beta_i|^{-\frac{1}{2}}\left\{\boldsymbol{\xi}^T\begin{bmatrix} k_a^3(-b')+k_a c'-k_r k_a & k_a^2(-b')+c'-\frac{k_r}{2} \\ k_a^2(-b')+c'-\frac{k_r}{2} & k_a(-b') \end{bmatrix}\boldsymbol{\xi} + k_a b'\delta_2\cdot|\xi_2||\xi_2| + (k_r+k_a^2\cdot b')\delta_1\cdot|\xi_1||\xi_1|\right\}$$

$$\leq |\beta_i|^{-\frac{1}{2}}\left\{\boldsymbol{\xi}^T\begin{bmatrix} k_a^3(-b')+k_a c'-k_r k_a & k_a^2(-b')+c'-\frac{k_r}{2} \\ k_a^2(-b')+c'-\frac{k_r}{2} & k_a(-b') \end{bmatrix}\boldsymbol{\xi} + k_a b'\delta_2\frac{1}{\delta_3}|\xi_1||\xi_1| + (k_r+k_a^2 b')\delta_1\cdot|\xi_1||\xi_1|\right\}$$

$$= -|\beta_i|^{-\frac{1}{2}}\cdot\boldsymbol{\xi}^T\begin{bmatrix} k_a^3 b'+\frac{k_r}{2}k_a-k_a b'\delta_2\frac{1}{\delta_3}-k_r\delta_1-k_a^2 b'\delta_1 & k_a^2 b' \\ k_a^2 b' & k_a b' \end{bmatrix}\boldsymbol{\xi}$$

Thus,

$$\dot{L}_i \leq -|\beta_i|^{-\frac{1}{2}}\cdot\boldsymbol{\xi}^T \cdot \overbrace{\begin{bmatrix} k_a^3 b'+\frac{k_r}{2}k_a-k_a b'\delta_2\frac{1}{\delta_3}-k_r\delta_1-k_a^2 b'\delta_1 & k_a^2 b' \\ k_a^2 b' & k_a b' \end{bmatrix}}^{\mathbf{D}}\boldsymbol{\xi}, \quad \text{(A-1)}$$

B. Definition of conditions $\mathcal{A}$ and $\mathcal{B}$

In accordance with the properties of positive definite matrix, all the order principal subexpressions of **D** are greater than zero.

The first order principal subexpression of **D** is greater than zero as follows.

$$\left|k_a^3 b'+\frac{k_r}{2}k_a-k_a b'\delta_2\frac{1}{\delta_3}-k_r\delta_1-k_a^2 b'\delta_1\right| = k_a^3 b'+\frac{k_r}{2}k_a-k_a b'\delta_2\frac{1}{\delta_3}-k_r\delta_1-k_a^2 b'\delta_1 > 0, \quad \text{(A-2)}$$

The second order principal subexpression of **D** is greater than zero as follows.

$$\begin{vmatrix} k_a^3 b'+\frac{k_r}{2}k_a-k_a b'\delta_2\frac{1}{\delta_3}-k_r\delta_1-k_a^2 b'\delta_1 & k_a^2 b' \\ k_a^2 b' & k_a b' \end{vmatrix} = \left(k_a^3\cdot b'+\frac{k_r}{2}\cdot k_a-k_a b'\cdot\delta_2\cdot\frac{1}{\delta_3}-k_r\cdot\delta_1-k_a^2\cdot b'\cdot\delta_1\right)\cdot k_a b'-(k_a^2\cdot b')^2 > 0. \quad \text{(A-3)}$$

In this paper, (A-2) and (A-3) are defined as the condition $\mathcal{A}$ and $\mathcal{B}$, respectively. Thus the conditions $\mathcal{A}$ and $\mathcal{B}$ need to be satisfied for the system stability.

C. Detailed proof of Lyapunov stability

1) DISCUSSION OF CONDIFIONS $\mathcal{A}$ AND $\mathcal{B}$

$\mathcal{A}$: $k_a^3\cdot b'+\frac{k_r}{2}\cdot k_a-k_a\cdot b'\cdot\delta_2\cdot\frac{1}{\delta_3}-k_r\cdot\delta_1-k_a^2\cdot b'\cdot\delta_1>0.$

$$\Rightarrow\ k_a\cdot b'\cdot\left(k_a^2-k_a\cdot\delta_1-\delta_2\cdot\frac{1}{\delta_3}\right) > \frac{k_r}{2}\cdot(2\cdot\delta_1-k_a). \quad \text{(A-4)}$$

Discussion 1.

In the case of $2\delta_1 > k_a$, we have

$$\begin{cases} \dfrac{k_a\cdot b'\cdot\left(k_a^2-k_a\cdot\delta_1-\delta_2\cdot\frac{1}{\delta_3}\right)}{2\cdot\delta_1-k_a} > \dfrac{k_r}{2} \\ k_a^2-k_a\cdot\delta_1-\delta_2\cdot\dfrac{1}{\delta_3}>0 \end{cases}, \quad \text{(A-5)}$$

Since $\frac{k_r}{2}>0$, then

$$\Rightarrow\quad k_a^2-k_a\cdot\delta_1-\delta_2\cdot\frac{1}{\delta_3}>0, \quad \text{(A-6)}$$

$$\Rightarrow\quad \left(k_a-\frac{\delta_1}{2}\right)\left(k_a-\frac{\delta_1}{2}\right)>\frac{\delta_1^2}{4}+\delta_2\cdot\frac{1}{\delta_3}, \quad \text{(A-7)}$$

Then

$$\begin{cases} k_a > \dfrac{\delta_1}{2}+\sqrt{\dfrac{\delta_1^2}{4}+\delta_2\cdot\dfrac{1}{\delta_3}} \\ k_a < \dfrac{\delta_1}{2}-\sqrt{\dfrac{\delta_1^2}{4}+\delta_2\cdot\dfrac{1}{\delta_3}}<0, \text{(Abandon)} \end{cases}, \quad \text{(A-8)}$$

Finally, we can get

$$\begin{cases} \dfrac{\delta_1}{2}+\sqrt{\dfrac{\delta_1^2}{4}+\delta_2\cdot\dfrac{1}{\delta_3}} < k_a < 2\cdot\delta_1 \\ 0<\dfrac{k_r}{2}<\dfrac{k_a\cdot b'\cdot\left(k_a^2-k_a\cdot\delta_1-\delta_2\cdot\frac{1}{\delta_3}\right)}{2\cdot\delta_1-k_a} \end{cases}, \quad \text{(A-9)}$$

Discussion 2.

In the case of $2\delta_1 < k_a$, we have

$$-\frac{k_a\cdot b'\cdot\left(k_a^2-k_a\cdot\delta_1-\delta_2\cdot\frac{1}{\delta_3}\right)}{k_a-2\cdot\delta_1}<\frac{k_r}{2}, \quad \text{(A-10)}$$

Case (1): In the case of $k_a^2-k_a\cdot\delta_1-\delta_2\cdot\frac{1}{\delta_3}>0$, we have

$$\frac{k_r}{2}>0, \quad \text{(A-11)}$$

$$\begin{cases} k_a > \max\left\{\dfrac{\delta_1}{2}+\sqrt{\dfrac{\delta_1^2}{4}+\delta_2\cdot\dfrac{1}{\delta_3}}, 2\delta_1\right\} \\ k_a < \dfrac{\delta_1}{2}-\sqrt{\dfrac{\delta_1^2}{4}+\delta_2\cdot\dfrac{1}{\delta_3}}<0, \text{(Abandon)} \end{cases}. \quad \text{(A-12)}$$

Case (2): In the case of $k_a^2-k_a\cdot\delta_1-\delta_2\cdot\frac{1}{\delta_3}\leq 0$, we have

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

$$\begin{cases} -\dfrac{k_a \cdot b' \cdot \left(k_a^2 - k_a \cdot \delta_1 - \delta_2 \cdot \dfrac{1}{\delta_3}\right)}{k_a - 2 \cdot \delta_1} < \dfrac{k_r}{2} \\ 2\delta_1 < k_a \le \dfrac{\delta_1}{2} + \sqrt{\dfrac{\delta_1^2}{4} + \delta_2 \cdot \dfrac{1}{\delta_3}} \end{cases}, \quad \text{(A-13)}$$

Finally, considering Cases (1) and (2) comprehensively, we can get

a1. When $2\delta_1 < \frac{\delta_1}{2} + \sqrt{\frac{\delta_1^2}{4} + \delta_2 \cdot \frac{1}{\delta_3}}$

$$\begin{cases} k_a > 2\delta_1 \\ \dfrac{k_r}{2} > \max\left\{ -\dfrac{k_a \cdot b' \cdot \left(k_a^2 - k_a \cdot \delta_1 - \delta_2 \cdot \dfrac{1}{\delta_3}\right)}{k_a - 2 \cdot \delta_1}, 0 \right\} \end{cases}. \quad \text{(A-14)}$$

b1. When $2\delta_1 > \frac{\delta_1}{2} + \sqrt{\frac{\delta_1^2}{4} + \delta_2 \cdot \frac{1}{\delta_3}}$

$$\begin{cases} \dfrac{k_r}{2} > 0 \\ k_a > 2\delta_1 \end{cases}. \quad \text{(A-15)}$$

Discussion 3.

In the case of $2\delta_1 = k_a$, we have

$$k_a \cdot b' \cdot \left(k_a^2 - k_a \cdot \delta_1 - \delta_2 \cdot \frac{1}{\delta_3}\right) > 0, \quad \text{(A-16)}$$

$$\Rightarrow \quad k_a^2 - k_a \cdot \delta_1 - \delta_2 \cdot \frac{1}{\delta_3} > 0, \quad \text{(A-17)}$$

Then

$$\begin{cases} k_a = 2\delta_1 > \sqrt{\dfrac{2\delta_2}{\delta_3}} \\ k_a = 2\delta_1 < -\sqrt{\dfrac{2\delta_2}{\delta_3}} < 0, \text{(Abandon)} \end{cases}, \quad \text{(A-18)}$$

Finally, we can get

$$\begin{cases} k_a > \sqrt{\dfrac{2\delta_2}{\delta_3}} \\ \dfrac{k_r}{2} > 0 \end{cases}, \quad \text{(A-19)}$$

$$\boldsymbol{\mathcal{B}}: \left(k_a^3 \cdot b' + \frac{k_r}{2} \cdot k_a - k_a b' \cdot \delta_2 \cdot \frac{1}{\delta_3} - k_r \cdot \delta_1 - k_a^2 \cdot b' \cdot \delta_1\right) \cdot k_a b' - (k_a^2 \cdot b')^2 > 0$$

$$\Rightarrow \frac{k_r}{2} \cdot (k_a - 2 \cdot \delta_1) > k_a^2 \cdot b' \cdot \delta_1 + k_a \cdot b' \cdot \delta_2 \cdot \frac{1}{\delta_3}, \quad \text{(A-20)}$$

Ensuring the establishment of $\boldsymbol{\mathcal{B}}$, then we can obtain

$$\begin{cases} k_a > 2 \cdot \delta_1 \\ \dfrac{k_r}{2} > \dfrac{k_a^2 \cdot b' \cdot \delta_1 + k_a \cdot b' \cdot \delta_2 \cdot \dfrac{1}{\delta_3}}{k_a - 2 \cdot \delta_1} \end{cases}, \quad \text{(A-21)}$$

Discussion 4.

In (A-14),

$$\frac{k_r}{2} > \max\left\{ -\frac{k_a \cdot b' \cdot \left(k_a^2 - k_a \cdot \delta_1 - \delta_2 \cdot \frac{1}{\delta_3}\right)}{k_a - 2 \cdot \delta_1}, 0 \right\}, \quad \text{(A-22)}$$

In (A-21),

$$\frac{k_r}{2} > \frac{k_a^2 \cdot b' \cdot \delta_1 + k_a \cdot b' \cdot \delta_2 \cdot \frac{1}{\delta_3}}{k_a - 2 \cdot \delta_1} > 0, \quad \text{(A-23)}$$

Let the following comparison by using equations (A-22) and (A-23),

$$\frac{k_a^2 \cdot b' \cdot \delta_1 + k_a \cdot b' \cdot \delta_2 \cdot \frac{1}{\delta_3}}{k_a - 2 \cdot \delta_1} + \frac{k_a \cdot b' \cdot \left(k_a^2 - k_a \cdot \delta_1 - \delta_2 \cdot \frac{1}{\delta_3}\right)}{k_a - 2 \cdot \delta_1} > 0, \quad \text{(A-24)}$$

Then the range of *kr* is

$$\frac{k_r}{2} > \frac{k_a^2 \cdot b' \cdot \delta_1 + k_a \cdot b' \cdot \delta_2 \cdot \frac{1}{\delta_3}}{k_a - 2 \cdot \delta_1}. \quad \text{(A-25)}$$

Considering $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{B}}$, ***Discussion 1***, ***Discussion 2***, ***Discussion 3*** and ***Discussion 4*** comprehensively, the range of values k_r and k_a can be specified as follows

$$\begin{cases} k_a > 2\delta_1 \\ \dfrac{k_r}{2} > \dfrac{k_a^2 \cdot b' \cdot \delta_1 + k_a \cdot b' \cdot \delta_2 \cdot \dfrac{1}{\delta_3}}{k_a - 2\delta_1} \end{cases}, \quad \text{(A-26)}$$

2) THE RANGE OF VALUES K_R AND K_A

Since

$$b' = \left(\frac{1}{2} \cdot \tanh(\rho|\beta_i|) + |\beta_i| \cdot [1 - \tanh^2(\rho\beta_i)] \cdot \rho\right). \quad \text{(A-27)}$$

Let

$$f(\rho|\beta_i|) = 2b', \qquad \rho|\beta_i| \in (0, +\infty). \quad \text{(A-28)}$$

Then

$$f(\rho|\beta_i|) = \tanh(\rho|\beta_i|) + 2\rho \cdot |\beta_i| \cdot [1 - \tanh^2(\rho\beta_i)], \quad \text{(A-29)}$$

When $f(\rho|\beta_i|)$ reaches the maximum,

$$\rho \cdot |\beta_i| \cdot \tanh(\rho|\beta_i|) = 0.75. \quad \text{(A-30)}$$

Thus, we have

$$f(\rho|\beta_i|)|_{max} = f(\varepsilon) = 1.6016. \quad \text{(A-31)}$$

So the scale of $f(\rho|\beta_i|)$ is

$$0 \le f(\rho|\beta_i|) \le 1.6016. \quad \text{(A-32)}$$

The scale of b' is

$$0 \le b' \le 0.8008. \quad \text{(A-33)}$$

Therefore, the range of values k_r and k_a can be re-specified as follows

$$\begin{cases} k_a > 2\delta_1 \\ k_r > 1.6016 \cdot \dfrac{k_a^2 \cdot \delta_1 + k_a \cdot \delta_2 \cdot \dfrac{1}{\delta_3}}{k_a - 2\delta_1} \end{cases}. \quad \text{(A-34)}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

REFERENCES

[1] L. Jin, S. Li, X. Luo, Y. M. Li, and B. Qin, "Neural dynamics for cooperative control of redundant robot manipulators," *IEEE Trans. Ind. Inform.*, vol. 14, no. 9, pp. 3812-3821, Sept. 2018.

[2] A. Gasparetto, P. Boscariol, A. Lanzutti and R. Vidoni, "Path Planning and Trajectory Planning Algorithms: A General Overview," *Motion and Operation Planning of Robotic Systems*, vol. 29, pp. 3-27, Jan. 2015.

[3] J.M. Rossell, J. Vicente-Rodrigo, J. Rubio-Massegu, and V. Barcons, "An effective strategy of real-time vision-based control for a Stewart platform," in *Proc. ICIT*, Feb. 20-22 2018, pp. 75-80.

[4] Z. Shiller and Y.R. Gwo, "Dynamic motion planning of autonomous vehicles," *IEEE Trans. Robotics and Automation*, vol. 7, no. 2, pp. 241-249, Apr. 1991.

[5] E. Cheung and V. Lumelsky, "Real-time path planning procedure for a whole-sensitive robot arm manipulator," *Robotica*, vol. 10, no. 4, pp. 339-349, Jul. 1992.

[6] A.S. Rana and A.M.S. Zalzala, "Near time-optimal collision-free motion planning of robotic manipulators using an evolutionary algorithm," *Robotica*, vol. 14, no. 6, pp. 621-632, Nov. 1996.

[7] S. Macfarlane, and E.A. Croft, "Jerk-bounded manipulator trajectory planning: design for real-time applications," *IEEE Trans. Robotics and Automation*, vol. 19, no. 1, pp. 42-52, Mar. 2003.

[8] R. Haschke, E. Weitnauer, and H. Ritter, "On-line planning of time-optimal, jerk-limited trajectories," in *Proc. IROS*, Sept. 22-26 2008, pp. 3248-3253.

[9] J. Vannoy and J. Xiao, "Real-time adaptive motion planning (RAMP) of mobile manipulators in dynamic environments with unforeseen changes," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1199-1212, Oct. 2008.

[10] H. Hajkarami and M.J. Sadigh, "On line path planning for minimum time motion of manipulators on non symmetric trajectories," in *Proc. ICM*, Apr. 13-15 2011, pp. 427-432.

[11] S.S. Pchelkin, A.S. Shiriaev, A. Robertsson, and L.B. Freidovich, "Integrated time-optimal trajectory planning and control design for industrial robot manipulator," in *Proc. IROS*, Nov. 3-7 2013, pp. 2521-2526.

[12] C.G.L. Bianco and F. Ghilardelli, "Real-time planner in the operational space for the automatic handling of kinematic constraints," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 3, pp. 730-739, Jul. 2014.

[13] A. Casalino, A.M. Zanchettin, and P. Rocco, "Online planning of optimal trajectories on assigned paths with dynamic constraints for robot manipulators," in *Proc. IROS*, Oct. 9-14 2016, pp. 979-985.

[14] N.A. Scott and C.R. Carignan, "A line-based obstacle avoidance technique for dexterous manipulator operations," in *Proc. ICRA*, May 19-23 2007, pp. 3353-3358.

[15] D.H.P. Nguyen, M. Hoffmann, A. Roncone, U. Pattacini, and G. Metta, "Compact real-time avoidance on a humanoid robot for human-robot interaction," in *Proc. ACM/HRI*, Jan. 2018, pp. 416-424.

[16] T. Petrič, A. Gams, N. Likar, and L. Zlajpah, "Obstacle avoidance with industrial robots," *Motion and Operation Planning of Robotic Systems*, Mar. 2015, pp. 113-145.

[17] R. Samuel Buss and J.S. Kim, "Selectively damped least squares for inverse kinematics," *Journal of Graphics Gpu & Game Tools*, vol. 10, no. 3, pp. 37-49, Jan. 2005.

[18] H. Z. Jin, H. Zhang, Z. X. Liu, D. C. Yang, D. Y. Bie, H. Zhang, G. Li, Y. H. Zhu, and J. Zhao, "A synthetic algorithm for tracking a moving object in a multiple-dynamic obstacles environment based on kinematically planar redundant manipulators," *Math. Probl. Eng.*, vol. 2017, no. 3, pp. 1-15, Apr. 2017.

[19] H. Zhang, H. Jin, Z. Liu, Y. Liu, Y. Zhu and J. Zhao, "Real-time kinematic control for redundant manipulators in a time-varying environment: multiple-dynamic obstacle avoidance and fast tracking of a moving object," *IEEE Trans. Ind. Inform.*, vol. 16, no. 1, pp. 28-41, Jan. 2020.

[20] S.A. Bazaz and B. Tondu, "Online computing of a robotic manipulator joint trajectory with velocity and acceleration constraints," in *Proc. ISATP*, Aug. 7-9 1997, pp. 1-6.

[21] E. Dyllong and A. Visioli, "Planning and real-time modifications of a trajectory using spline techniques," *Robotica*, vol. 21, no. 5, pp. 475-482, Sept. 2003.

[22] R. Lampariello, D. Nguyentuong, C. Castellini, G. Hirzinger, and J. Peters, "Trajectory planning for optimal robot catching in real-time," in *Proc. ICRA*, May 9-13 2011, pp. 3719-3726.

[23] D. Guo and Y. Zhang, "Acceleration-level inequality-based MAN scheme for obstacle avoidance of redundant robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 61, no. 12, pp. 6903-6914, Dec. 2014.

[24] J. Wang and X. Lei, "On-line kinematical optimal trajectory planning for manipulator," in *Proc. IHMSC*, Aug. 25-26 2018, pp. 319-323.

[25] M. Schlemmer and D. Oberpfaffenhofen, "On-line trajectory optimization for kinematically redundant robot-manipulators and avoidance of moving obstacles," in *Proc. ICRA*, Apr. 22-28 1996, pp. 474-479.

[26] K. Shin and N. Mckay, "A dynamic programming approach to trajectory planning of robotic manipulators," *IEEE Trans. Autom. Control*, vol. 31, no. 6, pp. 491-500, Jun. 2003.

[27] A.A. Ata and H. Johar, "Cubic-spline trajectory planning of a constrained flexible manipulator," in *Proc. ICRA*, Apr. 18-22 2005, pp. 3126-3130.

[28] T. Borangiu, F. Anton, and S. Anton, "Open architecture for vision-based robot motion planning and control," *Motion and Operation Planning of Robotic Systems*, pp. 63-89, Mar. 2015.

[29] Z. Zhang, S. Chen, S. Li, "Compatible Convex-Nonconvex Constrained QP-Based Dual Neural Networks for Motion Planning of Redundant Robot Manipulators," *IEEE Trans. Control Syst. Technol.*, vol.27, no. 3, pp. 1250 - 1258, May 2018.

[30] Z. Zhang, Y. Lin, S. Li, Y. Li, Z. Yu, Y. Luo, "Tricriteria Optimization-Coordination Motion of Dual-Redundant-Robot Manipulators for Complex Path Planning," *IEEE Trans. Control Syst. Technol.*, vo. 26, no. 4, pp. 1345 - 1357, Jul. 2017.

[31] Z. Zhang, Z. Yan, "A Varying Parameter Recurrent Neural Network for Solving Nonrepetitive Motion Problems of Redundant Robot Manipulators," *IEEE Trans. Control Syst. Technol.*, vol. 27, no.6, pp. 2680 - 2687, Nov. 2018.

[32] Z. Zhang, T. Fu, Z. Yan, L. Jin, Y. Sun, Z. Yu, Y. Li, "A Varying-Parameter Convergent-Differential Neural Network for Solving Joint-Angular-Drift Problems of Redundant Robot Manipulators," *IEEE Trans. Mech.*, vol. 23, no. 2, pp. 679 – 689, Apr. 2018.

[33] A. Reiter, A. Muller, and H. Gattringer, "On higher-order inverse kinematics methods in time-optimal trajectory planning for kinematically redundant manipulators," *IEEE Trans. Ind. Inform.*, vol. 14, no. 4, pp. 1681-1690, Apr. 2018.

[34] J. A. Moreno, and M. Osorio, "A Lyapunov approach to second-order sliding mode controllers and observers," in *Proc. ICDC*, Dec. 9-11 2008, pp. 2856-2861.

[35] L. Zhang, H. Obeid, S. Laghrouche, M. Cirrincione, "Second order sliding mode observer of linear induction motor," *IET Electr. Power Appl.*, vol. 13, no. 1, pp. 38 - 47, Feb. 2019.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001266, IEEE Access

HUI ZHANG received the B.S. in mechanical engineering from the Zhongyuan University of Technology, Zhengzhou, China, in 2013, and the M.S. degrees in mechanical engineering from Harbin Institute of Technology (Shenzhen), Shenzhen, China, in 2015.

He is currently working towards the PhD degree at the State Key Laboratory of Robotics and System, Harbin Institute of Technology. His research interests include robotic automated planning and cooperative manipulation.

HONGZHE JIN received the B.S. degree in instrument science and technology from Harbin Institute of Technology (HIT), Harbin, China, in 1999, and the Ph.D. degree in electronic engineering from the Pusan National University, Busan, South Korea, in 2009.

He is currently an Associate Professor with the School of Mechatronics Engineering, HIT. His research interests include modeling, analysis, control, identification of complex systems, and robotics.

MINGDA GE received the B.S. in mechanical engineering and automation from Harbin Institute of Technology, Harbin, China, in 2017, and the M.S. degrees in mechatronic engineering from Harbin Institute of Technology, Harbin, China, in 2019.

He is currently working towards the PhD degree at the State Key Laboratory of Robotics and System, Harbin Institute of Technology. His research interests include trajectory prediction and robotic automated planning.

YANHE ZHU received the B.S. and Ph.D. degrees in Mechatronics Engineering from Harbin Institute of Technology (HIT), Harbin, China, in 1998 and 2004, respectively.

He is currently a Professor and a Ph.D. supervisor of the School of Mechatronics Engineering, HIT. He also is the deputy director of the State Key Laboratory of Robotics and System. His research interests include reconfigurable modular robots and robotic exoskeletons.

JIE ZHAO received the B.S. and Ph.D. degrees in mechatronics engineering from the Harbin Institute of Technology (HIT), Harbin, China, in 1990 and 1996, respectively.

He is currently a Professor with the School of Mechatronics Engineering, HIT. He is the director of the State Key Laboratory of Robotics and System, HIT.

His research interests include industrial robots and bionic robots. Dr. Zhao is the leader of the Subject Matter Expert Group of Intelligent Robot in National 863 Program supervised by the Ministry of Science and Technology of China.