Tracking Control of Robot Manipulators with Unknown Models: A Jacobian-Matrix-Adaption Method

Dechao Chen, Yunong Zhang, *Member, IEEE*, and Shuai Li, *Senior Member, IEEE*

***Abstract*—Tracking control of robot manipulators is a fundamental and significant problem in robotic industry. As a conventional solution, the Jacobian-matrix-pseudo-inverse (JMPI) method suffers from two major limitations: one is the requirement on known information of the robot model such as parameter and structure; the other is the position error accumulation phenomenon caused by the open-loop nature. To overcome such two limitations, this paper proposes a novel Jacobian-matrix-adaption (JMA) method for the tracking control of robot manipulators via the zeroing dynamics. Unlike existing works requiring the information of the known robot model, the proposed JMA method uses only the input-output information to control the robot with unknown model. The solution based on the JMA method transforms the internal, implicit, and unmeasurable model information to the external, explicit, and measurable input-output information. Moreover, simulation studies including comparisons and tests substantiate the efficacy and superiority of the proposed JMA method for the tracking control of robot manipulators subject to unknown models.**

***Index Terms*—Input-output information, Jacobian-matrix-adaption (JMA), robot manipulators, unknown models, zeroing dynamics.**

I. Introduction

TRACKING control of industrial robots aims at the computation of control actions to steer the motion of the end-effector along user-defined or desired paths in the robot workspace [1]–[7]. The tracking control involves a wide range of industrial robot applications, such as welding [8], painting process [9], and part assembly [10]. It is known that the tracking control of nonlinear or complex models with unknown information is a critical and challenging problem [11], [12]. Thus, the tracking control of robot mechanisms, especially robot manipulators, has been intensively studied by robotic researchers [1], [3], [8], [9], [13]. Generally, the robot tracking control can be divided into two categories, i.e., the tracking control related to dynamic model [2], [14], and the tracking control related to kinematic model (which is investigated in this paper). For example, in the presence of unknown dynamics of robot model, Yang *et al.* [2] employed an effective neural network approximation technique to compensate for uncertainties of robot model. For the tracking control with robot kinematic model, Zhang *et al.* [1] proposed a scheme to generate coordinated head-arm motion for a humanoid robot with two degrees-of-freedom (DOF) for the head and seven for each arm. The robot model in [1] can effectively track external targets and body parts. Hu *et al.* [3] developed an adaptive homography-based visual servo tracking controller for the camera-in-hand problem using a quaternion formulation to represent rotation tracking error. Tarokh and Zhang [8] creatively investigated a genetic algorithm for the real-time motion tracking of robot manipulators. Zhao *et al.* [9] addressed the problem of path tracking of industrial robots by correcting a preplanned path through an iterative learning control method.

Despite the great advantages offered by advanced robot manipulators in modern industry, developing an effective scheme for their tracking control still remains a challenging problem. Some approaches for such a problem have been reported, including neural-network-based method [1], adaptive control method [15], barrier-Lyapunov-function-based method [16], and Jacobian-matrix-pseudo-inverse (JMPI) method [17]. Specifically, the adaptive control method can effectively handle the model uncertainty and thus has been widely studied by researchers [18]–[24]. For example, Luo and Chen proposed an effective algorithm for human pose estimation by using the adaptive control law with point-cloud-based limb regression approach. By utilizing the barrier Lyapunov function, the tracking control problems in the presence of output constraints can be successfully handled [16], [25]–[27]. In addition, the JMPI method is a conventional control method to find the joint variable vector by first computing the inverse or the pseudo-inverse of the Jacobian matrix. Then the reference trajectories in the joint space can be obtained from the user-defined or desired path of the end-effector. Due to the simplicity, the control system based on the JMPI method has been widely investigated to

Manuscript received September 21, 2017; accepted October 21, 2017. Date of publication October 25, 2017; date of current version July 2, 2018. This work is supported in part by the National Natural Science Foundation of China under Grant 61473323 and Grant 61401385, by the Science and Technology Program of Guangzhou, China under Grant 2014J4100057, in part by Hong Kong Research Grants Council Early Career Scheme under Grant 25214015, and in part by Departmental General Research Fund of Hong Kong Polytechnic University (G.61.37.UA7L). Paper no. TII-17-2194. *(Corresponding author: Shuai Li.)*

D. Chen and Y. Zhang are with the School of Information Science and Technology, Sun Yat-sen University (SYSU), Guangzhou 510006, China (e-mail: chdchao@sina.com; zhynong@mail.sysu.edu.cn).

S. Li is with the Department of Computing, the Hong Kong Polytechnic University, Hung Hom, Hong Kong (e-mail: shuaili@polyu.edu.hk).

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

Digital Object Identifier 10.1109/TII.2017.2766455

resolve the redundancy of robot manipulators [17], [28]–[30]. Jin and Zhang [29] introduced two Taylor-type discrete-time Zhang neural network models via a new numerical-differentiation formula for the time-varying Jacobian matrix pseudoinversion of a planar robot manipulator. In [30], a neural network activated by the Li function for the tracking control of a PA10 robot manipulator via Jacobian matrix pseudoinversion was investigated in detail.

Control systems using the conventional JMPI method suffer from two major limitations: one is the requirement on the known information of the robot model; the other is the position error accumulation (PEA) phenomenon caused by the open-loop nature of the JMPI method. It is known that the conventional JMPI method needs to continuously perform the pseudo-inverse of Jacobian matrix over time, which is closely related to the robot's specific characteristic, structure, and parameter. In other words, the type of the used robot, the length of each link, and the Denavit–Hartenberg (DH) parameters should be precisely known for the control design [31], [32]. Even though the kinematic parameters are precisely known, the impacts on long time operation, e.g., wearing, friction, fatigue, may cause the variation of kinematic parameters of a robot manipulator, resulting in the difference between their real values and the nominal ones. In addition, the kinematic parameters usually vary from robot to robot, making the JMPI method less portable. Moreover, the position error of the JMPI method may accumulate over time, which prohibits this method from being used in practical applications involving tasks that should be run for a long time. The PEA phenomenon may also induce degradation in accuracy to perform primary tasks during every cyclic motion [28]. To overcome such two major limitations of the JMPI method, later works [33], [34] have been reported by estimating or approximating the Jacobian matrix of robot manipulators. For example, Wanasinghe *et al.* [33] creatively proposed a Jacobian free approach for relative localization of a multirobotics system. In addition, Luo *et al.* [34] successfully solved the model-free optimal tracking control problem of nonaffine nonlinear discrete-time systems.

Zeroing dynamics (also known as Zhang dynamics) [35], [36] has been proposed to handle time-varying problems with multiple state dimensions [37], [38]. Specifically, this new neural dynamics approach zeros out each element of the error function in a neural-dynamic manner. It can be viewed as a systematic and methodological approach for solving various time-varying problems (including tracking control problems) [35], [36]. In spite of the great success of zeroing dynamics in time-varying problems solving, existing works based on the zeroing dynamics require the information of the known model involved. For example, the zeroing-type control laws for the population control of a specific Lotka–Volterra model were presented in [39]. In addition, based on the zeroing dynamics and gradient dynamics, Zhang *et al.* [40] developed and investigated a kind of zeroing-gradient controllers for achieving the tracking control purpose of the inverted pendulum on a cart system with a known model. Note that existing control systems using the zeroing dynamics are all the model-based control systems requiring the information of the known model involved. The tracking control of robot systems subject to unknown models has been a major problem overlooked for years. To make progress along the direction of zeroing dynamics on time-varying control problem solving, a novel and effective Jacobian-matrix-adaption (JMA) method for the tracking control of robot manipulators is proposed in this paper. It is the first time to investigate and handle the unknown model of the robot manipulator in the framework of the zeroing dynamics.

The remainder of this paper is organized in four sections. In Section II, the tracking control problem is formulated. In Section III, the control system is designed via the proposed JMA method with the theoretical analysis. Section IV illustrates the simulation studies including comprehensive comparisons and robustness tests. Section V concludes this paper with final remarks. Before ending this introductory section, it is worth pointing out the main contributions of this paper as follows.

1) To make progress along the direction of the zeroing dynamics on time-varying problem solving, this paper proposes a novel JMA method for the tracking control of robot manipulators. It is an important breakthrough of the zeroing dynamics in time-varying robot control research concerning the unknown model.
2) The core of the proposed JMA method features the full utilization of the real-time input-output information to remedy the unknown model information of the robot.
3) The control system derived by the proposed JMA method successfully handles the PEA phenomenon and shows an outstanding tracking performance even without knowing the robot model.
4) Simulation studies including comparisons and tests substantiate the efficacy and superiority of the proposed JMA method for the tracking control of robot manipulators.

II. Problem Formulation

The inverse-kinematics problem in robotics is to find the values of the joint state vector in the joint space given the position-and-orientation vector of the end-effector in the Cartesian space relative to the known parameter and structure of the robot model. Specifically, the kinematic equation for robot manipulators is depicted as follows:

$$\mathbf{f}(\theta(t)) = \mathbf{r}_{\mathrm{a}}(t)$$

where $\theta(t) \in \mathbb{R}^n$ denotes the joint state (e.g., the joint angle for serial robot manipulators) vector at time t, and $\mathbf{r}_{\mathrm{a}}(t) \in \mathbb{R}^m$ denotes the end-effector's position-and-orientation vector. For a specific robot manipulator, $\mathbf{f}(\cdot) : \mathbb{R}^n \to \mathbb{R}^m$ is a continuous nonlinear forward-kinematics mapping with known parameter and structure of the robot model. The objective of the tracking control of robot manipulators is to generate the corresponding joint state signal $\theta(t)$ in real-time t by considering the user-defined or desired Cartesian path $\mathbf{r}_{\mathrm{d}}(t) \in \mathbb{R}^m$, such that $\mathbf{f}(\theta(t)) = \mathbf{r}_{\mathrm{a}}(t) \to \mathbf{r}_{\mathrm{d}}(t)$. With the tracking objective, by differentiating $\mathbf{r}_{\mathrm{a}}(t)$ with respect to time t, it can obtain the relation between the end-effector Cartesian velocity vector $\dot{\mathbf{r}}_{\mathrm{a}}(t) \in \mathbb{R}^m$ and the joint velocity (or termed joint control signal) $\dot{\theta}(t) \in \mathbb{R}^n$ as follows:

$$J(\theta(t))\dot{\theta}(t) = \dot{\mathbf{r}}_{\mathrm{a}}(t) \to \dot{\mathbf{r}}_{\mathrm{d}}(t) \tag{1}$$

where $J(\theta(t)) \in \mathbb{R}^{m\times n}$ is the Jacobian matrix of end-effector defined as $J(\theta(t)) = \partial \mathbf{f}(\theta(t))/\partial\theta(t)$, and $\dot{\mathbf{r}}_{\mathrm{d}}(t)$ is the desired velocity vector of the end-effector. The conventional JMPI based control system [17] at velocity level is written as

$$\mathbf{u}(t) = \dot{\theta}(t) = J^{\dagger}(\theta(t))\dot{\mathbf{r}}_{\mathrm{d}}(t) \tag{2}$$

where $J^{\dagger}(\theta(t)) \in \mathbb{R}^{n\times m}$ denotes the pseudo-inverse of Jacobian matrix $J(\theta(t))$. In addition, $\mathbf{u}(t)$ denotes the control input of robot manipulators using the joint velocity signal.

Remark 1: As observed in the conventional control system (2), the information, e.g., parameter and structure, of the robot model is required to be known in prior for formulating $J^{\dagger}(\theta(t))$. Note that different robot manipulators possess different parameters and structures, which makes the conventional control system (2) less portable in practical applications. In addition, physical modeling of different robot manipulators with unknown models consumes extra time in such a conventional control system. Due to the lack of the information on robot models in some applications, the tracking control process may be destroyed by unknown models. Moreover, the conventional control system (2) is essentially an open-loop system which does not possess the feedback information of execution, and results in the accumulation of position error over time. As for some closed-loop robot systems which can compensate for the accumulation of position error, external proportional-integral-derivative (PID) controllers are required additionally [41]. Therefore, it is urgently needed for a highly portable solution which works without knowing the information of the robot model and external PID controllers, as well as simultaneously overcomes PEA phenomenon appearing in JMPI based control system. Moreover, the research on robot manipulators with unknown models in the framework of the zeroing dynamics is also vacant. These construct motivation that we investigate the tracking control of robot manipulators with unknown models by using the JMA method based on the zeroing dynamics.

III. Control System Design

In this section, by presenting the design processes of tracking control and JMA, we propose a control system for the tracking control of robot manipulators by transforming the robot model information to the input-output information. Before proposing the control system, the concept of the zeroing dynamics is introduced as the background.

Concept: As a new neural dynamics approach, zeroing dynamics (also known as Zhang dynamics) has been proposed and developed to solve various time-varying (dynamic) problems [37]–[40]. The main principle of this new neural dynamics approach is based on an indefinite matrix-valued or vector-valued error function, and exploits the time-derivative information of time-varying coefficients. Such a neural dynamics approach is developed from Hopfield neural network, which is a typical recurrent neural network [42]. Given that this neural dynamic approach zeros out each element of the error function and was proposed by Zhang *et al.* [35], [36], it was named as the zeroing dynamics. As recognized by Marco *et al.* [38] that the only systematic approach proposed so far for the solution to time-varying problems is the zeroing dynamics.

A. Tracking Control

First, to monitor and control the process of the tracking problem solving of robot manipulators, a vector-valued error function (for the measurement of the difference between the desired path and the actual trajectory of the end-effector in real-time t) is defined as follows:

$$\mathbf{e}(t) = \mathbf{r}_{\mathrm{d}}(t) - \mathbf{r}_{\mathrm{a}}(t) \tag{3}$$

To make each element $e_i(t)$ (with $i = 1, 2, \ldots, m$) of the error function (3) converge to zero, by utilizing the zeroing-dynamics [35], [36] design formula

$$\dot{\mathbf{e}}(t) = -\gamma\mathbf{e}(t) \tag{4}$$

with $\gamma \in \mathbb{R}^{+}$ being a design parameter to scale the convergence rate, we have

$$\dot{\mathbf{r}}_{\mathrm{d}}(t) - J(\theta(t))\dot{\theta}(t) = -\gamma(\mathbf{r}_{\mathrm{d}}(t) - \mathbf{r}_{\mathrm{a}}(t)). \tag{5}$$

Note that dynamical (5) can be rewritten as an explicit form with control input signal $\mathbf{u}(t)$ at velocity level as follows

$$\mathbf{u}(t) = \dot{\theta}(t) = J^{\dagger}(\theta(t))(\dot{\mathbf{r}}_{\mathrm{d}}(t) + \gamma(\mathbf{r}_{\mathrm{d}}(t) - \mathbf{r}_{\mathrm{a}}(t))). \tag{6}$$

As observed from control (6), the feedback information $\mathbf{r}_{\mathrm{a}}(t)$ comes from the output of the primary task execution measured by the sensors in the end-effector workspace, which makes the control system to be closed-loop.

Differing from the conventional control systems by using known parameter and structure of the robot model, in this paper, we adapt and update the Jacobian matrix with unknown parameters and structure in real-time t. Therefore, control (6) with the unknown pseudo-inverse of Jacobian matrix can be written as

$$\mathbf{u}(t) = \dot{\theta}(t) = \hat{J}^{\dagger}(t)(\dot{\mathbf{r}}_{\mathrm{d}}(t) + \gamma(\mathbf{r}_{\mathrm{d}}(t) - \mathbf{r}_{\mathrm{a}}(t))) \tag{7}$$

where $\hat{J}^{\dagger}(t) \in \mathbb{R}^{n\times m}$ is the unknown pseudo-inverse of Jacobian matrix to be adapted.

B. Jacobian Matrix Adaption

In this subsection, by applying the zeroing-dynamics approach again, the unknown Jacobian matrix is adapted in real-time t for the effective tracking control of robot manipulators. During the JMA process, we assume that the robot system is always at a nonsingular configuration.

For the adaption of Jacobian matrix of robot manipulators, a vector-valued error function with the unknown Jacobian matrix $\hat{J}(t)$ is defined as

$$\epsilon(t) = \dot{\mathbf{r}}_{\mathrm{a}}(t) - \hat{J}(t)\dot{\theta}(t) \tag{8}$$

with vector $\epsilon(t) \in \mathbb{R}^{m}$. Then, by applying the zeroing-dynamics design formula

$$\dot{\epsilon}(t) = -\mu\epsilon(t) \tag{9}$$

where $\mu \in \mathbb{R}^{+}$ is another design parameter to scale the convergence rate, we have the following dynamical equation for the

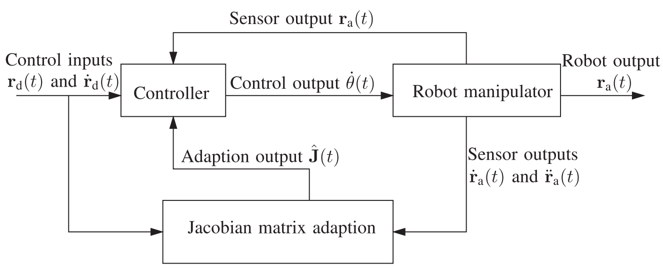

Fig. 1. Block diagram and information transformation of the control system based on the proposed JMA method for the tracking control of robot manipulators with unknown models.

adaption of the unknown Jacobian matrix:

$$\ddot{\mathbf{r}}_\mathrm{a}(t) - \dot{\hat{J}}(t)\dot{\theta}(t) - \hat{J}(t)\ddot{\theta}(t) = -\mu(\dot{\mathbf{r}}_\mathrm{a}(t) - \hat{J}(t)\dot{\theta}(t)) \quad (10)$$

which can be explicitly rewritten as

$$\dot{\hat{J}}(t) = \big(\ddot{\mathbf{r}}_\mathrm{a}(t) - \hat{J}(t)\ddot{\theta}(t) + \mu(\dot{\mathbf{r}}_\mathrm{a}(t) - \hat{J}(t)\dot{\theta}(t))\big)\dot{\theta}^{\dagger}(t). \quad (11)$$

Equation (11) is the JMA equation of the control system.

Remark 2: Note that the pseudo-inverse of joint velocity vector, i.e., $\dot{\theta}^{\dagger}(t)$, needs to be calculated in (11). The pseudo-inverse of a vector is a special case for Moore–Penrose pseudo-inverse [43]. The pseudo-inverse of a null (or say, all zero) joint velocity vector is the transpose of the null joint velocity vector. The pseudo-inverse of a nonnull joint velocity vector is calculated as $\dot{\theta}^{\dagger}(t) = \dot{\theta}^{\mathrm{T}}(t)/\big(\dot{\theta}^{\mathrm{T}}(t)\dot{\theta}(t)\big)$ with $\dot{\theta}^{\mathrm{T}}(t)$ denoting the transpose of the joint velocity vector.

Remark 3: This work mainly focuses on the proposal of the JMA method for the tracking control of the robot manipulator with unknown models via zeroing dynamics provided that the robot manipulator is always at a nonsingular configuration. When a robot manipulator is at a kinematic singularity configuration, the Jacobian matrix becomes ill conditioned (or say, rank-deficient) [44]. Note that the control system by using the proposed JMA method calculates the pseudo-inverse of the unknown Jacobian matrix, i.e., $\hat{J}^{\dagger}(t)$. For the remedy of the singularity problem, $(\kappa I + \hat{J}^{\mathrm{T}}(t)\hat{J}(t))^{-1}\hat{J}^{\mathrm{T}}(t)$ with $\kappa > 0$ being a sufficiently small constant and I being an identity matrix, can be used to approximate the analytical solution of $\hat{J}^{\dagger}(t)$ in engineering applications. We know that the eigenvalue of matrix $\kappa I + \hat{J}^{\mathrm{T}}(t)\hat{J}(t)$ is always greater than zero for $\kappa > 0$ and thus avoids singular state. Therefore, by introducing the above remedy, the proposed JMA method is still feasible even the robot in the situation of singular configuration. This remedy for the singularity problem is the so-called regularization in the fields of machine learning and inverse problems, which is widely used in engineering applications [45].

For better understanding, the block diagram of the control system via the proposed JMA method for the tracking control of the robot manipulator with the unknown model is presented in Fig. 1. Due to the closed-loop nature of the whole control system, the proposed JMA method needs the real-time measurements of the end-effector position $\mathbf{r}_\mathrm{a}(t)$, velocity $\dot{\mathbf{r}}_\mathrm{a}(t)$, and acceleration $\ddot{\mathbf{r}}_\mathrm{a}(t)$ as the feedback. In practical applications, some advanced robots, e.g., the KUKA KR60-2 industrial robot [46], can measure the real-time feedback signals of the end-effector via high accuracy sensors (including the position, velocity and acceleration sensors) [47]. The real-time feedback information of the robot end-effector measured by the corresponding sensors is fully utilized to learn the unknown Jacobian matrix adaptively. By taking advantage of the output-feedback information of the end-effector measured by sensors equipped in the advanced robot manipulator, an outstanding tracking control performance can thus be achieved even in the situation that the robot model is unknown. One key novelty of this work is that the solution based on the proposed JMA method transforms the internal, implicit, and unmeasurable model information to the external, explicit, and measurable input-output information. The block diagram in Fig. 1 with the information transformation illustrates the rationale of the proposed control method, and the reason why the control system possesses an outstanding tracking performance and overcomes the PEA phenomenon.

C. Theoretical Analysis

To verify the effectiveness of the proposed JMA method, the following theoretical analysis including the stability and convergence of the closed-loop control system is presented.

1) Stability Analysis: The stability analysis is provided by the following theorem.

Theorem 1: For the tracking control of a robot manipulator described by (1), starting from an initial position $\mathbf{r}_\mathrm{a}(0)$, the control system depicted in control (7) and JMA (11) is stable in the sense of Lyapunov.

Proof: For the tracking control of a robot manipulator described by (1), the dynamical (7) and (11) of the closed-loop control system are the equivalent expansion of the following equations:

$$\dot{\mathbf{e}}(t) = -\gamma\mathbf{e}(t)$$
$$\dot{\epsilon}(t) = -\mu\epsilon(t)$$

where $\mathbf{e}(t) = \mathbf{r}_\mathrm{d}(t) - \mathbf{r}_\mathrm{a}(t)$ and $\epsilon(t) = \dot{\mathbf{r}}_\mathrm{a}(t) - \hat{J}(t)\dot{\theta}(t)$. Define a Lyapunov function candidate as follows

$$l(t) = \frac{\mathbf{e}^{\mathrm{T}}(t)\mathbf{e}(t)}{2} + \frac{\epsilon^{\mathrm{T}}(t)\epsilon(t)}{2}.$$

We know that $l(t)$ is positive-definite in view of $l(t) > 0$ for $\mathbf{e}(t) \neq 0$ or $\epsilon(t) \neq 0$, and $l(t) = 0$ for both $\mathbf{e}(t) = 0$ and $\epsilon(t) = 0$ only. Then, we can obtain the time-derivative of $l(t)$ as follows:

$$\dot{l}(t) = -\gamma\mathbf{e}^{\mathrm{T}}(t)\mathbf{e}(t) - \mu\epsilon^{\mathrm{T}}(t)\epsilon(t).$$

Therefore, we have the result that $\dot{l}(t)$ is negative-definite for design parameters $\gamma > 0$ and $\mu > 0$. Based on the Lyapunov stability theory [48], the control system depicted in control (7) and JMA (11) is stable. ■

2) Convergence Analysis: The convergence analysis is provided by the following definition and theorem.

Definition 1: For the tracking control of a robot manipulator, starting from an initial position $\mathbf{r}_\mathrm{a}(0)$, an end-effector trajectory $\mathbf{r}_\mathrm{a}(t)$ at time $t \geq 0$ in Cartesian space synthesized by a control system is said to be convergent to the desired path $\mathbf{r}_\mathrm{d}(t)$ if it satisfies

$$\mathbf{r}_\mathrm{a}(t) \to \mathbf{r}_\mathrm{d}(t), \text{ as } t \to \infty. \quad (12)$$

Theorem 2: For the tracking control of a robot manipulator described by (1), starting from an initial position $\mathbf{r}_{\mathrm{a}}(0)$, the end-effector trajectory $\mathbf{r}_{\mathrm{a}}(t)$ in Cartesian space synthesized by control system depicted in control (7) and JMA (11) is convergent to the desired path $\mathbf{r}_{\mathrm{d}}(t)$.

Proof: According to the zeroing-dynamics design formula (9), we have

$$\epsilon(t) = \epsilon(0)\exp(-\mu t). \tag{13}$$

The steady-state of the error is obtained as

$$\lim_{t\to\infty} \epsilon(t) = 0. \tag{14}$$

According to error function (8), we obtain

$$\lim_{t\to\infty} \dot{\mathbf{r}}_{\mathrm{a}}(t) = \lim_{t\to\infty} \hat{J}(t)\dot{\theta}(t). \tag{15}$$

Revisit control (7) in the following form:

$$\dot{\mathbf{r}}_{\mathrm{d}}(t) - \hat{J}(t)\dot{\theta}(t) = -\gamma(\mathbf{r}_{\mathrm{d}}(t) - \mathbf{r}_{\mathrm{a}}(t)). \tag{16}$$

By substituting (15) into (16), we have

$$\lim_{t\to\infty}(\dot{\mathbf{r}}_{\mathrm{d}}(t) - \dot{\mathbf{r}}_{\mathrm{a}}(t)) = -\gamma \lim_{t\to\infty}(\mathbf{r}_{\mathrm{d}}(t) - \mathbf{r}_{\mathrm{a}}(t)) \tag{17}$$

which leads to

$$\lim_{t\to\infty}(\mathbf{r}_{\mathrm{d}}(t) - \mathbf{r}_{\mathrm{a}}(t)) = \lim_{t\to\infty}(\mathbf{r}_{\mathrm{d}}(0) - \mathbf{r}_{\mathrm{a}}(0))\exp(-\gamma t) \tag{18}$$

and we finally have the result as

$$\mathbf{r}_{\mathrm{a}}(t) \to \mathbf{r}_{\mathrm{d}}(t), \text{ as } t \to \infty. \tag{19}$$

According to Definition 1, we have the result that the end-effector trajectory $\mathbf{r}_{\mathrm{a}}(t)$ in Cartesian space synthesized by the proposed control system is convergent to the desired path $\mathbf{r}_{\mathrm{d}}(t)$. ■

Remark 4: It is worth pointing out that the proposed JMA (11) requires the time derivative of joint control signal, i.e., $\ddot{\theta}(t)$. Mathematically, the time derivative of joint control signal can be obtained on the basis of the Euler type difference rule $\ddot{\theta}(t) \approx (\dot{\theta}(t+\Delta t) - \dot{\theta}(t))/\Delta t \approx (\theta(t+2\Delta t) - 2\theta(t+\Delta t) + \theta(t))/(\Delta t)^2$ with sufficient small sampling time Δt. Considering that the sampling time usually can not be arbitrarily small in practical implementation, users can employ the Euler difference rule with more data points according to the requirement of accuracy. In principle, it can be approximated with any desired accuracy by employing an appropriate difference rule with sufficient number of data points. Such an approximation approach possesses the characteristics of high precision and convenience of software and hardware implementations. Practically, the joint acceleration can be also measured by sensors equipped in advanced robots as the output-feedback information.

Remark 5: It is worth pointing out that the tracking control of robot manipulators focused in this paper is a typical kinematics problem. The solutions of the proposed JMA method are the related joint variables (i.e., θ and $\dot{\theta}$). In some applications, when the motors of the robot joints are driven by pulse commands transmitted from the host computer, such joint variables are finally converted into pulses per second [28] as the joint control signals, which are sent to drive the servo and stepper motors.

IV. Simulations, Comparisons, and Tests

Note that PUMA 560 is a typical kind of robot manipulators with 6 independently controlled joints, and is widely used in robotic researches [31], [32]. In addition, the end-effector of PUMA 560 can reach any position in its workspace at any orientation. In this paper, we consider the three-dimensional (3-D) position tracking control of the end-effector. Therefore, the PUMA 560 serves as a functional redundant manipulator. Its corresponding parameters and structure can be found in [31], [32]. Actually, for an inverse-kinematics problem solving, the redundant robot manipulator, e.g., the PUMA 560 robot manipulator, is a typical platform of overactuated robots [49]. Due to the feature of multiple solutions to a redundant robot manipulation, based on the convergence theorem in Section III-C, the state of the redundant robot manipulator by using the proposed JMA method converges to one of feasible solutions, when the number of actuated joints is more than that of output DOF. Note that the tracking control problem solving by the proposed JMA method essentially is robotic redundancy resolution [49]. The control method established in this paper is not directly applicable to underactuated robots due to the lack of redundancy for them [50]. The JMA method is applicable to overactuated (or specifically say, redundant) robots.

In this section, simulations, comparisons, and tests based on the PUMA 560 manipulator are conducted to show the efficacy and superiority of the proposed JMA method for handling the tracking control problem. Without loss of generality, the motion-task duration is set to be $T_{\mathrm{d}} = 20$ s. The initial value of joint angle vector of the robot manipulator is set to be $\theta(0) = [0, -\pi/4, 0, 2\pi/3, -\pi/4, 0]^{\mathrm{T}}$ rad, and the initial value of joint velocity vector of the robot manipulator is set to be $\dot{\theta}(0) = [0, 0, 0, 0, 0, 0]^{\mathrm{T}}$ rad/s. The initial value of the Jacobian matrix for the adaption is set as

$$\hat{J}(0) = \begin{bmatrix} 0.007 & -0.178 & -0.483 & -0.111 & 0.192 & 0 \\ 0.689 & 0 & 0 & -0.091 & -0.157 & 0 \\ 0 & 0.689 & 0.384 & 0.111 & 0.064 & 0 \end{bmatrix}.$$

In addition, the design parameters are set as $\gamma = 100$ and $\mu = 100$ throughout this section.

Remark 6: As for the selection of the initial value of the Jacobian matrix $\hat{J}(0)$ for the adaption, without loss of generality, we can predefine a rough value according to the engineering experience. Based on Theorem 2, the Jacobian matrix $\hat{J}(t)$ is adaptively updated in real-time t to make the end-effector trajectory converge to the desired path. As for the predefined design parameters γ and μ, theoretically, arbitrary values satisfying $\gamma > 0$ and $\mu > 0$ can be set. In practical applications, for the purpose of rapid convergence, the values of design parameters γ and μ can be set as appropriately large as the hardware would permit [51].

A. Circular Path Tracking Control

In this example, we consider the end-effector of the robot manipulator to track a circular path. During the task execution, i.e., $t \in [0, T_{\mathrm{d}}]$, the *X*-, *Y*-, and *Z*-axes of the user-defined or

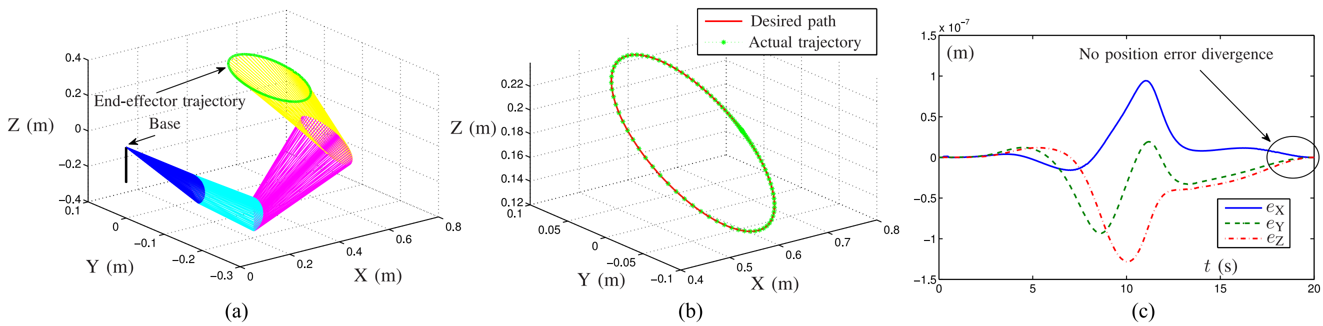

Fig. 2. Synthesized motion results by the control system via the proposed JMA method when the end-effector of the robot manipulator tracks the circular path. (a) Motion trajectories of the robot manipulator. (b) Profiles of the actual trajectory and the desired path. (c) Profiles of the position error.

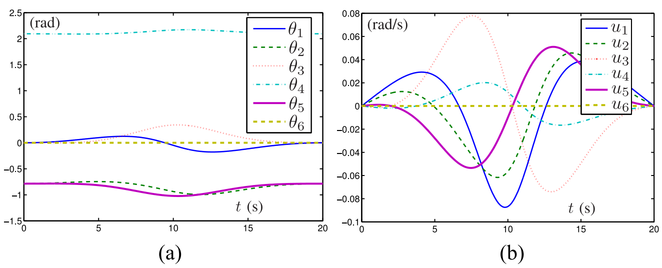

Fig. 3. Synthesized joint variables by the control system via the proposed JMA method when the end-effector of the robot manipulator tracks the circular path. (a) Profiles of joint angles. (b) Profiles of joint control signals.

desired path $\mathbf{r}_\mathrm{d}(t)$ for the end-effector are respectively set as

$$\mathbf{r}_{\mathrm{dX}}(t) = \iota\cos(2\pi\sin^2(0.5\pi t/T_\mathrm{d})) - \iota + 0.6891$$
$$\mathbf{r}_{\mathrm{dY}}(t) = \iota\cos(\pi/6)\sin(2\pi\sin^2(0.5\pi t/T_\mathrm{d})) + 0.0069$$
$$\mathbf{r}_{\mathrm{dZ}}(t) = \iota\sin(\pi/6)\sin(2\pi\sin^2(0.5\pi t/T_\mathrm{d})) + 0.1778$$

where the geometry parameter is set as $\iota = 0.1$ m in this example. The corresponding simulation results of the 6 DOF robot manipulator to track the circular path synthesized by the proposed control system are illustrated in Figs. 2 and 3. Specifically, Fig. 2(a) shows the joint motions of the 6 DOF robot manipulator in a 3-D plane during the task duration. The actual trajectory of the end-effector is a precise circle. As detailedly shown in Fig. 2(b), the actual trajectory of the end-effector of the robot manipulator is sufficiently close to the desired circle-shaped path. From Fig. 2(c), the maximal absolute value of end-effector position error $\mathbf{e} = [e_\mathrm{X}, e_\mathrm{Y}, e_\mathrm{Z}]^\mathrm{T}$ is about 1.4×10^{-7} m, which illustrates the great advantage of the proposed control system. Such results show that the end-effector's primary task is completed well. Moreover, as shown in Fig. 2(c), we can readily find that the divergence tendency of the position error does not occur. In other words, the PEA phenomenon is remedied successfully by using the proposed control system. This is in contrast to the results by using existing control paradigms that usually suffer from the PEA phenomenon [17], [28]. In addition, the profiles of corresponding joint angles and control signals of the robot are presented in Fig. 3. All joint angles and joint control signals are smooth and stable during the whole tracking process.

B. Rhodonea Path Tracking Control

In this example, we consider the end-effector of the robot manipulator to track a Rhodonea path. During the task execution, the *X*-, *Y*-, and *Z*-axes of the user-defined or desired path $\mathbf{r}_\mathrm{d}(t)$ for the end-effector are respectively set as

$$\mathbf{r}_{\mathrm{dX}}(t) = \varsigma\cos(4\pi\sin^2(0.5\pi t/T_\mathrm{d}))\cos(2\pi\sin^2(0.5\pi t/T_\mathrm{d})) - \varsigma + 0.6891$$
$$\mathbf{r}_{\mathrm{dY}}(t) = \varsigma\cos(\pi/6)\cos(4\pi\sin^2(0.5\pi t/T_\mathrm{d})) \times \sin^2(2\pi\sin(0.5\pi t/T_\mathrm{d})) + 0.0069$$
$$\mathbf{r}_{\mathrm{dZ}}(t) = \varsigma\sin(\pi/6)\cos(4\pi\sin^2(0.5\pi t/T_\mathrm{d})) \times \sin(2\pi\sin^2(0.5\pi t/T_\mathrm{d})) + 0.1778$$

where the geometry parameter is set as $\varsigma = 0.1$ m in this example. The corresponding simulation results of the 6 DOF robot manipulator to track the Rhodonea path synthesized by the proposed control system are illustrated in Figs. 4 and 5. Specifically, Fig. 4(a) shows the joint motion trajectories of the 6 DOF robot manipulator in a 3-D plane in the task duration. We can readily find that the actual trajectory of the end-effector is a precise Rhodonea. As seen from Fig. 4(b), the actual trajectory of the end-effector of the robot manipulator in this example is also sufficiently close to the desired Rhodonea path. In addition, the maximal absolute value of end-effector position error $\mathbf{e} = [e_\mathrm{X}, e_\mathrm{Y}, e_\mathrm{Z}]^\mathrm{T}$ shown in Fig. 4(c) is about 7.8×10^{-7} m. The results indicate that the end-effector's primary task to track the Rhodonea path is completed well. As shown in Fig. 4(c), we can similarly find that the divergence tendency of the position error in this example does not occur. In other words, the PEA phenomenon is also remedied successfully by using the proposed control system. The profiles of corresponding joint angles and control signals of the robot are presented in Fig. 5. Similar to the results when tracking a circular trajectory, all joint angles and control signals are smooth and stable. The above-mentioned simulation results in such two path tracking examples substantiate the effectiveness of the proposed control system for the tracking control of the robot manipulator with unknown parameters and structure, and the PEA phenomenon conquering.

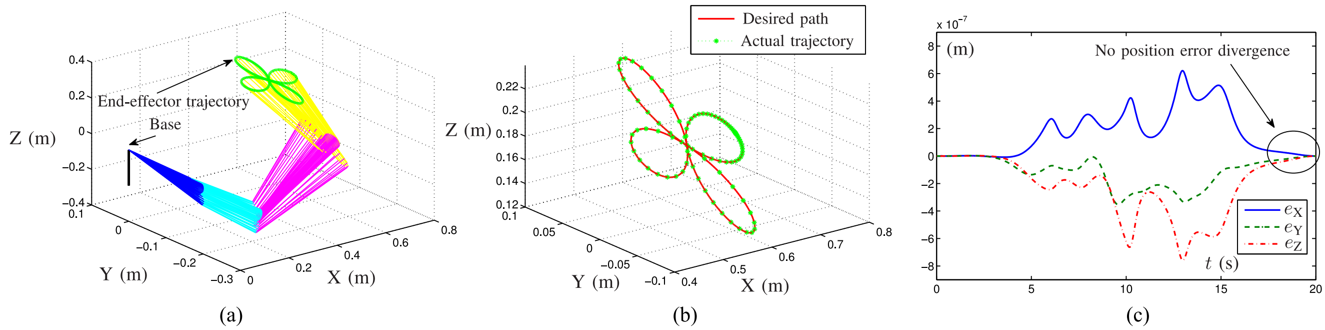

Fig. 4. Synthesized motion results by the control system via the proposed JMA method when the end-effector of the robot manipulator tracks the Rhodonea path. (a) Motion trajectories of the robot manipulator. (b) Profiles of the actual trajectory and the desired path. (c) Profiles of the position error.

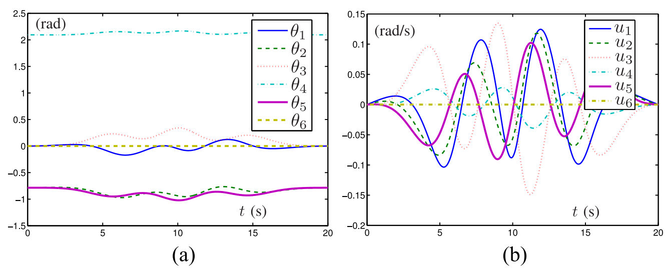

Fig. 5. Synthesized joint variables by the control system via the proposed JMA method when the end-effector of the robot manipulator tracks the Rhodonea path. (a) Profiles of joint angles. (b) Profiles of joint control signals.

C. Comparisons

As discussed in the introduction, the JMPI method, as a conventional control strategy, has been widely used in robotic applications [52]. It can be generally represented as one minimum-norm particular solution plus a homogeneous solution. Specifically, the general form of the JMPI scheme at joint velocity level can be described as follows:

$$\mathbf{u}(t) = \dot{\theta}(t) = J^{\dagger}(\theta(t))\dot{\mathbf{r}}_{\rm d}(t) + (I - J^{\dagger}(\theta(t))J(\theta(t)))\mathbf{v}(t)$$

where $I \in \mathbb{R}^{n\times n}$ is an identity matrix and $\mathbf{v}(t) \in \mathbb{R}^n$ is an arbitrary vector usually selected by using some optimization criteria or performance indexes. In the comparison, $\mathbf{v}(t) = 0$ is set for simplicity. Thus, the control system for comparison is exactly the control system depicted in (2) of Section II. It is worth pointing out that parameter and structure of the robot model are required to be known for the JMPI method calculating the pseudo-inverse of Jacobian matrix $J(\theta(t))$ in this comparative simulation. The DH parameters used for the conventional control system via JMPI method can be seen in Table I [31]. In addition, the corresponding robot structure can also be found in [31] and [32]. In contrast, the DH parameters and structure are not required to be known by using the proposed control system for the tracking tasks in Sections IV-A and IV-B, which is exactly one of the advantages of the control system via the proposed JMA method compared with the conventional control systems via the JMPI method.

The comparative simulation is conducted on the same 6 DOF robot manipulator. For a better comparison, all simulation

TABLE I
DH PARAMETERS OF THE PUMA 560 ROBOT MANIPULATOR

♯	a (m)	d (m)	α (rad)
1	0	0	$\pi/2$
2	0.4318	0	0
3	0.0203	0.15005	$-\pi/2$
4	0	0.4318	$\pi/2$
5	0	0	$-\pi/2$
6	0	0.25625	0

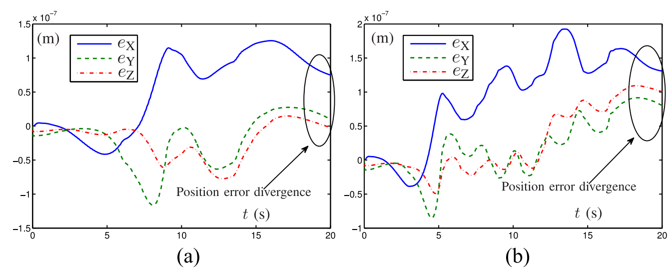

Fig. 6. Synthesized position errors by the control system via the conventional JMPI method when the end-effector of the robot manipulator tracks circular and Rhodonea paths. (a) Profiles of the position error of the circular path tracking. (b) Profiles of the position error of the Rhodonea path tracking.

conditions are set identical to those in Section IV-A and Section IV-B. The comparative position errors synthesized by the JMPI control system (2) for the same circular and Rhodonea paths tracking control are presented in Fig. 6. As seen from Fig. 6, synthesized by the conventional JMPI control system (2), the maximal absolute value of end-effector position errors is sufficiently small, which implies that the robot manipulator completes the tasks successfully. However, the divergences of position errors occur for both cases at the end of the task durations. In other words, the PEA phenomenon occurs when using the JMPI control system (2) presented in [17]. As discussed in the introduction, the PEA phenomenon prohibits the control system from being used in applications involving tasks that have to run for a long time or run for many cycles. Therefore, the JMPI control system (2) presented in [17] is disadvantageous for complicated applications with long durations.

Moreover, to further show the advantages and superiorities of the control system via the proposed method, comprehensive

TABLE II
COMPARISONS OF DIFFERENT METHODS FOR THE TRACKING CONTROL OF ROBOT MANIPULATORS

Method	Parameter limitation◇	Error accumulation	Real-time control	Robot restriction	Accuracy
This paper	No	No	Yes	No	High
[17]	Yes	Yes	Yes	Planar robot manipulator only	High
[28]	Yes	Yes	Yes	Planar robot manipulator only	High
[53]	Yes	No	Yes	Planar robot manipulator only	High
[54]	Yes	No	No	Humanoid robot only	Low

Note: ◇If the control system has the limitation that the information of robot model should be known in prior.

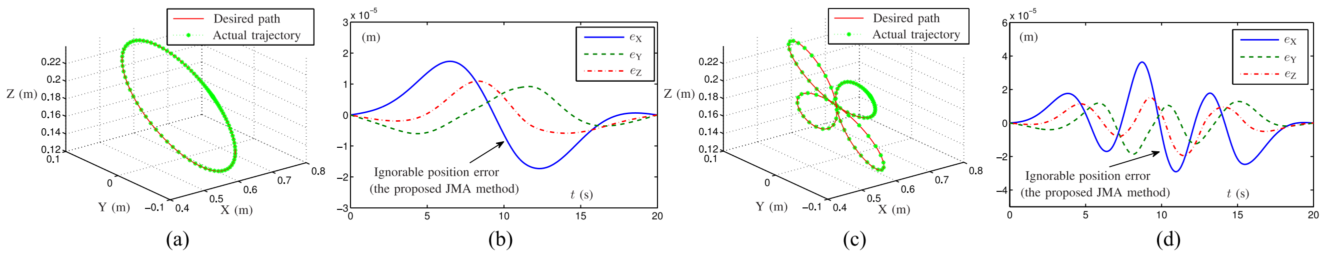

Fig. 7. Synthesized motion results by the control system via the proposed JMA method for the robot manipulator with inaccurate DH parameters to track desired paths. (a) Profiles of the actual trajectory and the desired circular path. (b) Profiles of the position error of the circular path tracking. (c) Profiles of the actual trajectory and the desired Rhodonea path. (d) Profiles of the position error of the Rhodonea path tracking.

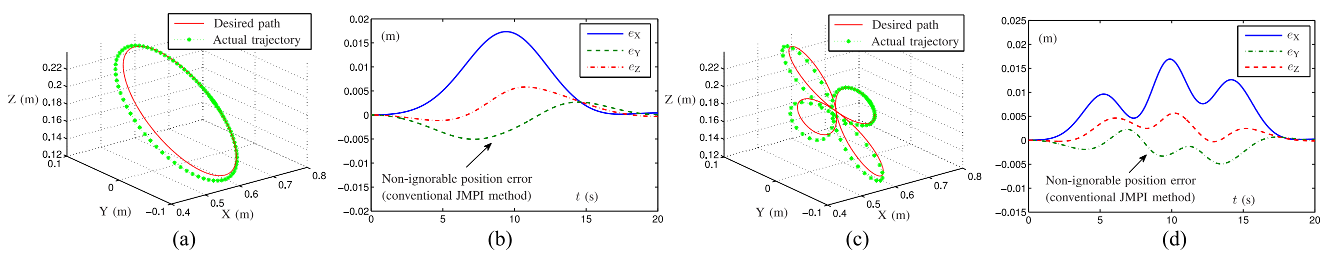

Fig. 8. Synthesized motion results by the control system via the conventional JMPI method for the robot manipulator with inaccurate DH parameters to track desired paths. (a) Profiles of the actual trajectory and the desired circular path. (b) Profiles of the position error of the circular path tracking. (c) Profiles of the actual trajectory and the desired Rhodonea path. (d) Profiles of the position error of the Rhodonea path tracking.

performance comparisons with different methods for the tracking control of the robot manipulator are summarized in Table II. As seen and compared with the tracking performance of different control systems, we can find that the DH parameters are not required to be known in prior by using the proposed control method. However, the parameter limitation occurs in other conventional control methods [17], [28], [53], [54]. Therefore, the proposed method is free of parameter limitation and robot restriction, and is greatly convenient and flexible for practitioners working on the robot system development. In addition, the control system by using the proposed JMA method in this paper can be similarly applied to many different robotic models. Another major advantage, as indicated in Table II, is that the proposed control method conquers the PEA phenomenon that commonly occurs in conventional control methods [17], [28], and is able to complete the design task with high accuracy (see Table II).

D. Robustness Tests

Note that the control system by using the proposed JMA method does not require to know DH parameters of robot model, and thus is less sensitive to the variation of robot parameters, and features self-adaption in the control process. To further show such an advantage of the control system, robustness tests are designed by introducing uncertainty of the robot model. The tests are comparatively conducted via the proposed JMA based control system and the conventional JMPI based control system. In some practical applications, it is difficult to obtain DH parameters in advance. In addition, factors like wearing, friction, fatigue, may also cause the variation of parameters of a robot manipulator with time, resulting in the difference between their real values and the nominal ones. Without loss of generality, we conduct the tests based on inaccurate DH parameters by introducing uncertain biases, i.e., $a_2 = 0.4318 + \Delta\sigma_1$ and $d_6 = 0.25625 + \Delta\sigma_2$ with $\Delta\sigma_1$ and $\Delta\sigma_2$ being the uncertain biases for the unknown robot model, which can be deemed as the situation of physical abrasion or any other uncertainty of robot model in many practical robot applications. The corresponding simulation results are shown in Figs. 7 and 8. Specifically, Fig. 7 shows the motion results synthesized by the proposed control system for the robot manipulator to track desired circular and Rhodonea paths. As one can readily see in the figure, the end-effector of the robot manipulator still achieves desirable tracking performance with enough small position errors even considering inaccurate DH parameters with uncertainty. Comparatively, Fig. 8 shows the motion results synthesized by the conventional control system via JMPI method for the robot manipulator to track desired circular and Rhodonea paths. It can be readily found that the end-effector of the robot manipulator can not accurately track the desired paths with the inaccurate DH parameters by introducing uncertainty. The corresponding position errors are also comparatively shown in the figures. As observed, the proposed control method is superior to the JMPI method in terms of the parameter sensitivity and uncertainty. The control system by using the proposed JMA method fully utilizes the output-feedback information of the end-effector, and transforms the internal, implicit, and unmeasurable model information to the external, explicit, and measurable input-output information. That is the main reason why the control system by using the proposed JMA method can successfully handle the robot model uncertainty, and achieves the outstanding tracking control performance.

V. Conclusion

This paper has achieved the tracking control of robot manipulators by using the proposed JMA method based on zeroing dynamics. The solution of the proposed JMA method has transformed the internal, implicit and unmeasurable model information to the external, explicit, and measurable input-output information. The control system by using the proposed JMA method has shown the ability to overcome the limitations of the conventional JMPI method in error accumulation and dependence on accurate parameter information of the known model. Moreover, theoretical analysis, simulation studies, comprehensive comparisons, and robustness tests have substantiated the efficacy, portability, and superiority of the proposed method. Future work lies in developing a complete experimental environment equipped with the advanced robot mechanism for the physical application of the control system by using the proposed JMA method. The extension of the JMA method to other robot mechanisms will also be considered in our future work.

Acknowledgment

Kindly note that all authors of the paper are jointly of the first authorship.

References

[1] Z. Zhang, A. Beck, and N. Magnenat-Thalmann, "Human-like behavior generation based on head-arms model for tracking external targets and body parts," *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1390–1400, Aug. 2015.

[2] C. Yang, Y. Jiang, Z. Li, W. He, and C.-Y. Su, "Neural control of bimanual robots with guaranteed global stability and motion precision," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1162–1171, Jun. 2017.

[3] G. Hu, N. Gans, N. Fitz-Coy, and W. Dixon, "Adaptive homography-based visual servo tracking control via a quaternion formulation," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 1, pp. 128–135, Jan. 2010.

[4] C. Yang, Z. Li, and J. Li, "Trajectory planning and optimized adaptive control for a class of wheeled inverted pendulum vehicle models," *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 24–36, Feb. 2013.

[5] K. Tchoń, A. Ratajczak, and I. Góral, "Lagrangian Jacobian inverse for nonholonomic robotic systems," *Nonlinear Dyn.* vol. 82, no. 4, pp. 1923–1932, Dec. 2015.

[6] Y.-J. Liu and S. Tong, "Adaptive NN tracking control of uncertain nonlinear discrete-time systems with nonaffine dead-zone input," *IEEE Trans. Cybern.*, vol. 45, no. 3, pp. 497–505, Mar. 2015.

[7] H. Gao and T. Chen, "Network-based $\mathcal{H}_\infty$ output tracking control," *IEEE Trans. Autom. Control*, vol. 53, no. 3, pp. 655–667, Apr. 2008.

[8] M. Tarokh and X. Zhang, "Real-time motion tracking of robot manipulators using adaptive genetic algorithms," *J. Intell. Robot. Syst.*, vol. 74, no. 3, pp. 697–708, Jun. 2014.

[9] Y. M. Zhao, Y. Lin, F. Xi, and S. Guo, "Calibration-based iterative learning control for path tracking of industrial robots," *IEEE Trans. Ind. Electron.*, vol. 62, no. 5, pp. 2921–2929, May 2015.

[10] L. Johannsmeier and S. Haddadin, "A hierarchical human-robot interaction-planning framework for task allocation in collaborative industrial assembly processes," *IEEE Rob. Autom. Lett.*, vol. 2, no. 1, pp. 41–48, Jan. 2017.

[11] H. Gao and T. Chen, "New results on stability of discrete-time systems with time-varying state delay," *IEEE Trans. Autom. Control*, vol. 52, no. 2, pp. 328–334, Feb. 2007.

[12] J. Na, Q. Chen, X. Ren, and Y. Guo, "Adaptive prescribed performance motion control of servo mechanisms with friction compensation," *IEEE Trans. Ind. Electron.*, vol. 61, no. 1, pp. 486–494, Jan. 2014.

[13] Y.-J. Liu, S. Lu, and S. Tong, "Neural network controller design for an uncertain robot with time-varying output constraint," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2060–2068, Aug. 2017.

[14] C. Yang, X. Wang, Z. Li, Y. Li, and C.-Y. Su, "Teleoperation control based on combination of wave variable and neural networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2125–2136, Aug. 2017.

[15] J. Na, X. Ren, and D. Zheng, "Adaptive control for nonlinear pure-feedback systems with high-order sliding mode observer," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 3, pp. 370–382, Mar. 2013.

[16] W. He, Z. Yin, and C. Sun, "Adaptive neural network control of a marine vessel with constraints using the asymmetric barrier Lyapunov function," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1641–1651, Jul. 2017.

[17] B. Liao and W. Liu, "Pseudoinverse-type bi-criteria minimization scheme for redundancy resolution of robot manipulators," *Robotica*, vol. 33, no. 10, pp. 2100–2113, Dec. 2015.

[18] Y.-J. Liu and S. Tong, "Barrier Lyapunov functions for Nussbaum gain adaptive control of full state constrained nonlinear systems," *Automatica*, vol. 76, pp. 143–152, Feb. 2017.

[19] R. C. Luo and S. Y. Chen, "Human pose estimation in 3-D space using adaptive control law with point-cloud-based limb regression approach," *IEEE Trans. Ind. Informat.*, vol. 12, no. 1, pp. 51–58, Feb. 2016.

[20] W. He, Y. Ouyang, and J. Hong, "Vibration control of a flexible robotic manipulator in the presence of input deadzone," *IEEE Trans. Ind. Informat.*, vol. 13, no. 1, pp. 48–59, Feb. 2017.

[21] W. He, S. Zhang, and S. S. Ge, "Adaptive control of a flexible crane system with the boundary output constraint," *IEEE Trans. Ind. Electron.*, vol. 61, no. 8, pp. 4126–4133, Aug. 2014.

[22] F. Guo, H. Kodamana, Y. Zhao, B. Huang, and Y. Ding, "Robust identification of nonlinear errors-in-variables systems with parameter uncertainties using variational Bayesian approach," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2017.2712743.

[23] Y.-J. Liu, S. Tong, C. L. P. Chen, and D.-J. Li, "Neural controller design-based adaptive control for nonlinear MIMO systems with unknown hysteresis inputs," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 9–19, Jan. 2016.

[24] C. Yang, K. Huang, H. Cheng, Y. Li, and C.-Y. Su, "Haptic identification by ELM-controlled uncertain manipulator," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2398–2409, Aug. 2017.

[25] Y.-J. Liu, J. Li, S. Tong, and C. L. P. Chen, "Neural network control-based adaptive learning design for nonlinear systems with full-state constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 7, pp. 1562–1571, Jul. 2016.

[26] W. He, C. Sun, and S. S. Ge, "Top tension control of a flexible marine riser by using integral-barrier Lyapunov function," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 2, pp. 497–505, Apr. 2015.

[27] Z. Chen, Z. Li, and C. L. P. Chen, "Adaptive neural control of uncertain MIMO nonlinear systems with state and input constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 6, pp. 1318–1330, Jun. 2017.

[28] D. Chen and Y. Zhang, "Minimum jerk norm scheme applied to obstacle avoidance of redundant robot arm with jerk bounded and feedback control," *IET Control Theory Appl.*, vol. 10, no. 15, pp. 1896–1903. Oct. 2016.

[29] L. Jin and Y. Zhang, "Discrete-time Zhang neural network of $O(\tau^3)$ pattern for time-varying matrix pseudoinversion with application to manipulator," *Neurocomputing*, vol. 142, pp. 165–173, Oct. 2014.

[30] D. Guo and Y. Zhang, "Li-function activated ZNN with finite-time convergence applied to redundant-manipulator kinematic control via time-varying Jacobian matrix pseudoinversion," *Appl. Soft Comput.*, vol. 24, pp. 158–168, Nov. 2014.
[31] Y. Zhang and J. Wang, "A dual neural network for constrained joint torque optimization of kinematically redundant manipulators," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 32, no. 5, pp. 654–662, Oct. 2002.
[32] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2243–2254, Oct. 2017.
[33] T. R. Wanasinghe, G. K. I. Mannm, and R. G. Gosine, "A Jacobian free approach for multi-robot relative localization," in *Proc. Can. Conf. Electr. Comput. Eng.*, Toronto, ON, Canada, 2014, pp. 1–6.
[34] B. Luo, D. Liu, T. Huang, and D. Wang, "Model-free optimal tracking control via critic-only Q-learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 10, pp. 2134–2144, Oct. 2016.
[35] Y. Zhang, L. Xiao, Z. Xiao, and M. Mao, *Zeroing Dynamics, Gradient Dynamics, and Newton Iterations.* Boca Raton, FL, USA: CRC Press, 2015.
[36] Y. Zhang and D. Guo, *Zhang Functions and Various Models.* Berlin, Germany: Springer-Verlag, 2015.
[37] A. Benchabane, A. Bennia, F. Charif, and A. Taleb-Ahmed, "Multi-dimensional capon spectral estimation using discrete Zhang neural networks," *Multidimensional Syst. Sign. Process.* vol. 24, no, 3, pp. 583–598, Sep. 2013.
[38] M. D. Marco, M. Forti, P. Nistri, and L. Pancioni, "Discontinuous neural networks for finite-time solution of time-dependent linear equations," *IEEE Trans. Cybern.*, vol. 46, no. 11, pp. 2509–2520, Nov. 2016.
[39] Y. Zhang, X. Yan, B. Liao, Y. Zhang, and Y. Ding, "Z-type control of populations for Lotka-Volterra model with exponential convergence," *Math. Biosci.*, vol. 272, pp. 15–23, Feb. 2016.
[40] Y. Zhang, X. Yu, Y. Yin, C. Peng, and Z. Fan, "Singularity-conquering ZG controllers of z2g1 type for tracking control of the IPC system," *Int. J. Control*, vol. 87, no. 9, pp. 1729–1746, Mar. 2014.
[41] T. Shibata and T. Murakami, "Null space motion control by PID control considering passivity in redundant manipulator," *IEEE Trans. Ind. Informat.*, vol. 4, no. 4, pp. 261–270, Nov. 2008.
[42] Y. Zhang, D. Jiang, and J. Wang, "A recurrent neural network for solving sylvester equation with time-varying coefficients," *IEEE Trans. Neural Netw.*, vol. 13, no. 5, pp. 1053–1063, Sep. 2002.
[43] E. H. Moore, "On the reciprocal of the general algebraic matrix," *B. Am. Math. Soc.*, vol. 26, no. 9, pp. 394–395, Jan. 1920.
[44] L. Jin, S. Li, H. M. La, and X. Luo, "Manipulability optimization of redundant manipulators using dynamic neural networks," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 4710–4720, Jun. 2017.
[45] P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems*. Philadelphia, PA, USA: SIAM, 1998.
[46] H. Koch, A. König, A. Weigl-Seitz, K. Kleinmann, and J. Suchý, "Multi-sensor contour following with vision, force, and acceleration sensors for an industrial robot," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 2, pp. 268–280, Feb. 2013.
[47] T. Faulwasser, T. Weber, P. Zometa, and R. Findeisen, "Implementation of nonlinear model predictive path-following control for an industrial robot," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 4, pp. 1505–1511, Jul. 2017.
[48] B. C. Kuo, *Automatic Control System*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1991.
[49] L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulators*. London, U.K.: Springer-Verlag, 2000.
[50] C. Yang, Z. Li, R. Cui, and B. Xu, "Neural network-based motion control of underactuated wheeled inverted pendulum models," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 11, pp. 2004–2016, Nov. 2014.
[51] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA, USA: Addison-Wesley, 1989.
[52] C. A. Klein and S. Ahmed, "Repeatable pseudoinverse control for planar kinematically redundant manipulators," *IEEE Trans. Syst. Man, Cybern.*, vol. 25, no. 12, pp. 1657–1662, May 1995.
[53] Y. Wang, X. Yan, L. He, H. Tan, and Y. Zhang, "Inverse-free solution of Z1G1 type to acceleration-level inverse kinematics of redundant robot manipulators," in *Proc. Int. Conf. Adv. Comput. Intell.*, Wuyi, China, 2015, pp. 57–62.
[54] B. Aghbali, A. Yousefi-Koma, A. G. Toudeshki, and A. Shahrokhshahi, "ZMP trajectory control of a humanoid robot using different controllers based on an off line trajoectory generation," in *Proc. Int. Conf. Robot. Mechatronics*, Tehran, Iran, 2013, pp. 530–534.

Dechao Chen received the B.S. degree in electronic information science and technology from Guangdong University of Technology, Guangzhou, China, in 2013. He is currently working toward the Ph.D. degree in communication and information systems at School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China, under the direction of Professor Yunong Zhang.

In addition, he is also with the SYSU-CMU Shunde International Joint Research Institute, Foshan, China, for cooperative research. His research interests include robotics, neural networks, nonlinear dynamics systems.

Yunong Zhang (S'02–M'03) received the B.S. degree in automatic control engineering from Huazhong University of Science and Technology, Wuhan, China, in 1996, the M.S. degree in automatic control engineering from South China University of Technology, Guangzhou, China, in 1999, and the Ph.D. degree in mechanical and automation engineering from Chinese University of Hong Kong, Shatin, Hong Kong, China, in 2003.

He is currently a professor in the School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China. Before joining Sun Yat-sen University in 2006, he had been with National University of Singapore, University of Strathclyde, and National University of Ireland at Maynooth, since 2003. In addition, he is also currently with the SYSU-CMU Shunde International Joint Research Institute, Foshan, China, for cooperative research. His main research interests include robotics, neural networks, computation and optimization. His web-page is now available at http://sdcs.sysu.edu.cn/content/2477.

Shuai Li (M'14–SM'17) received the B.E. degree in precision mechanical engineering from Hefei University of Technology, Hefei, China, in 2005, the M.E. degree in automatic control engineering from University of Science and Technology of China, Hefei, China, in 2008, and the Ph.D. degree in electrical and computer engineering from Stevens Institute of Technology, Hoboken, NJ, USA, in 2014.

After graduation, he joined the Department of Computing, Hong Kong Polytechnic University, Hung Hom, Hong Kong. His current research interests include dynamic neural networks, wireless sensor networks, robotic networks, machine learning, and other dynamic problems defined on a graph.

Dr. Li is currently on the editorial board of the *Neural Computing and Applications* and the *International Journal of Distributed Sensor Networks*.