Available online at www.sciencedirect.com
SCIENCE DIRECT®
Fuzzy Sets and Systems 154 (2005) 208–226
FUZZY sets and systems
www.elsevier.com/locate/fss

A computed torque controller for uncertain robotic manipulator systems: Fuzzy approach

Zuoshi Song*, Jianqiang Yi, Dongbin Zhao, Xinchun Li

Laboratory of Complex Systems and Intelligence Science, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, P.R. China

Received 6 February 2004; received in revised form 4 March 2005; accepted 17 March 2005
Available online 8 April 2005

Abstract

Computed Torque Control (CTC) is an effective motion control strategy for robotic manipulator systems, which can ensure globally asymptotic stability. However, CTC scheme requires precise dynamical models of robotic manipulators. To handle this impossibility, in this paper, a new approach combing CTC and Fuzzy Control (FC) is developed for trajectory tracking problems of robotic manipulators with structured uncertainty and/or unstructured uncertainty. Fuzzy part with a set of tunable parameters is employed to approximate lumped uncertainty due to parameters variations, unmodeled dynamics and so on in robotic manipulators. Based on Lyapunov stability theorem, it is shown that the proposed controller can guarantee stability of closed-loop systems and satisfactory tracking performances. The proposed approach indicates that CTC method is also valid for controlling uncertain robotic manipulators as long as compensative controller is appropriately designed. Finally, computer simulation results on a two-link elbow planar robotic manipulator are presented to show tracking capability and effectiveness of the proposed scheme.

Keywords: Robotic manipulators; Computed torque control; Fuzzy control; Fuzzy basis function

1. Introduction

It is well known that robotic manipulators are complicated, dynamically coupled, highly time-varying, highly nonlinear systems that are extensively used in tasks such as welding, paint spraying, accurate

* Corresponding author. Tel.: +86 10 8261 5422; fax: +86 10 6265 8815.
E-mail addresses: zssong_cas@yahoo.com (Z. Song), jianqiang.yi@mail.ia.ac.cn (J. Yi).

0165-0114/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.fss.2005.03.007

positioning systems and so on. In these tasks, end-effectors of robotic manipulators are commended to move from one place to another, or to follow some given trajectories as close as possible. Therefore, trajectory tracking problem is the most significant and fundamental task in control of robotic manipulators. Motivated by requirements such as a high degree of automation and fast speed operation from industry, in the past decades, various control methods are introduced in the publications such as proportional, integration, derivative (PID) control [9], feed-forward compensation control [5], adaptive control [17,18], variable structure control [19], neural networks control [12,13,16], fuzzy control [2,26] and so on.

As a predominant method in industrial robotic manipulators, traditional PID control has simple structure and convenient implementation [9]. However, some strong assumptions are required to be made, which involve that each joint of robotic manipulators is decoupled from others and the system has to be in the status of slow motion. Control performance degrades quickly as operating speed increases. Therefore, a robotic manipulator controlled in this way is only appropriate for relatively slow motion.

Robotic manipulator systems are inevitably subject to structured and unstructured uncertainty. Structured uncertainty is characterized by a correct dynamical model with parameters variations, which results from difference in weights, sizes and mass distributions of payloads manipulated by robotic manipulators, difference in links properties of robotic manipulators, difference in inaccuracies on torque constants of actuators and so on. Unstructured uncertainty is characterized by unmodeled dynamics, which is due to the presence of external disturbances, high-frequency modes of robotic manipulators, neglected time-delays and nonlinear frictions and so on.

Structured uncertainty can result in imprecision of dynamical models of robotic manipulators, and controllers designed for nominal parameters may not properly work for all changes in parameters. Adaptive control techniques [17,18] can be used in this case. However, adaptive control law is unable to handle unstructured uncertainty. To overcome this difficulty, variable structure control [19] that can simultaneously attenuate influences of both structured and unstructured uncertainty is employed. Unfortunately, undesirable chattering on sliding surface due to high frequent switching can deteriorate system performances, which cannot be eliminated completely.

Neural networks-based controllers for robotic manipulators are also extensively studied from early CMAC controllers [12] and simple backpropagation neural networks controllers without theoretical analysis and stability security to stable-neural networks adaptive controllers with rigorous stability analysis using Lyapunov stability theory or passive theory, see [16] and the references therein. Justification for using neural networks for robotic control lies in their excellent capability in learning any complicated mapping from training examples such that designed controllers are able to respond to unexpected situations. However, size of such neural networks becomes so large that powerful computational facilities are required.

For practical and complex control problem of robotic manipulators, various control approaches mentioned above naturally occur in our mind, but traditional and effective schemes also cannot be ignored. *Computed Torque Control* [11] (here abbreviated as CTC) is worth noting, because CTC is easily understood and of good performances. Briefly speaking, CTC is a linear control method to linearize and decouple robotic dynamics by using perfect dynamical models of robotic manipulator systems in order that motion of each joint can be individually controlled using other well-developed linear control strategies. However, CTC method for robotic manipulators suffers from two difficulties. First, CTC requires exact dynamical knowledge of robotic manipulators, which is apparently impossible in practical situations. Second, CTC is not robust to structured uncertainty and/or unstructured uncertainty, which may result in performance devaluation. Can we say that CTC is invalid for actual robotic manipulators according to the drawbacks above? The answer is "*No*", of course. As a matter of fact, many researches have made great efforts in improvement of CTC-based control methods for uncertain robotic manipulators.

Wijesoma [25] considered a variable structure controller as a compensator for CTC, which required complex solution to inverse of inertial matrix. Neural networks as a function approximator to learn lumped uncertainty of robotic manipulator systems was added to CTC in [10]. Tso [21] viewed error dynamical system as a reference model and designed a variable structure model reference adaptive controller to compensate for CTC. All of these schemes demonstrated that modified CTC scheme applying to uncertain robotic manipulator systems was also effective.

Fuzzy Logical System (here abbreviated as FLS) was introduced by Zadeh [27] in 1965. One of successful fuzzy applications is to model complex nonlinear systems by a set of fuzzy rules. One important property of fuzzy modeling approaches is that FLS is a universal approximator [24]. In other words, FLS can approximate virtually any nonlinear functions to arbitrary accuracy provided that enough rules are given. FLS for control, i.e. Fuzzy Control (here abbreviated as FC) can integrate expertise of skilled personnel into control procedure and mathematical model is not required. Over the last few years, FC for complex nonlinear systems have been developed extensively [1,4,7,22].

Recently, much attention has been devoted to FC for robotic manipulators. Sun [15] combined FC and variable structure control to construct a controller, where FLS was greatly simplified by using system representative point and its derivative as inputs. Control laws designed by Hsu [3] consisted of a regular fuzzy controller and a supervisory control term, which ensured stability of closed-loop systems. In [8], two FC schemes for a class of uncertain continuous-time multi-input multi-output nonlinear dynamical systems were derived. Satisfactory performances were achieved by applying them to robotic manipulators. The latest survey on FC for robotic manipulators can be found in [14] and references cited therein.

In this paper, it is supposed that robotic manipulator systems with structured uncertainty and/or unstructured uncertainty can be separated as two subsystems: nominal system with precise dynamical knowledge and uncertain system with unknown knowledge. An approach of CTC plus FC compensator is proposed. The nominal system is controlled using CTC and for uncertain system, a fuzzy controller is designed. Here the fuzzy controller acts as compensator for CTC. Parameters updating laws of the fuzzy controller are derived using Lyapunov stability theorem. In this way, uniformly ultimate boundedness and asymptotic stability of closed-loop systems are achieved. The validity of the proposed control scheme is verified by simulations on a two-link elbow planar robotic manipulator.

This paper is organized into five sections. Following the introduction, in Section 2, some preliminaries are addressed, which consist of mathematical notations, mathematical definitions, stability lemmas, brief introduction of FLS, dynamical models of robotic manipulators with uncertainties, and detailed explanation related to CTC for robotic manipulators. Section 3 is devoted to controller design based on CTC plus fuzzy compensator and stability proof based on Lyapunov stability theory. Section 4 includes simulation results to illustrate feasibility of the proposed strategy. Some conclusion remarks are finally included in Section 5.

2. Preliminaries

2.1. Mathematical notations, definitions and stability lemmas

The following notations and definitions will be used extensively throughout the paper.

Let R, R^n, $R^{n\times m}$ be real numbers, real n-vectors, and real $n \times m$ matrices, respectively. $\lambda_{\min}(Q)$ and $\lambda_{\max}(Q)$ indicate the minimum and the maximum eigenvalue of matrix $Q \in R^{n\times n}$, respectively. Let A^{T} denote the transpose of matrix A and $diag[\cdot]$ denote a diagonal matrix. It is assumed that all matrices are assumed compatible in dimension for the purpose of computations.

Definition 1. The trace of a square matrix $A = [a_{ij}] \in R^{n\times n}$, denoted by $Tr(A)$, is defined as the sum of diagonal elements of the matrix A.

Corollary 1. *The trace of a matrix and the trace of its transpose are the same, i.e.* $Tr(A) = Tr(A^{\mathrm{T}})$ *for* $\forall A \in R^{n\times n}$.

Corollary 2. *The trace of a scalar is equal to itself, i.e.* $Tr(y) = y$, *for* $\forall y \in R$. *Specially, if* $x \in R^n$, $y \in R^m$ *and* $A \in R^{n\times m}$, *then* $x^{\mathrm{T}}Ay$ *is a scalar satisfying* $Tr(x^{\mathrm{T}}Ay) = x^{\mathrm{T}}Ay$.

Corollary 3. *If* $x, y \in R^n$, *trace of scalar* $x^{\mathrm{T}}y$ *is equal to that of matrix* yx^{T}, *i.e.* $Tr(x^{\mathrm{T}}y) = Tr(yx^{\mathrm{T}})$.

Corollary 4. *The derivative of the trace of a time-varying matrix is equal to the trace of the derivative of the matrix. Specially, for a time-varying matrix* $A(t) \in R^{n\times n}$ *and a constant matrix* $\Phi \in R^{n\times n}$, *the relationship* $\mathrm{d}(Tr(A^{\mathrm{T}}\Phi A))/\mathrm{d}t = 2Tr(A^{\mathrm{T}}\Phi\dot{A})$ *naturally holds.*

Definition 2. The norm of a vector $x \in R^n$ is defined as $\|x\| = (\sum_{i=1}^{n} x_i^2)^{1/2}$ and the Frobenius norm of a matrix $A = [a_{ij}] \in R^{n\times n}$ is defined by $\|A\| = [Tr(A^{\mathrm{T}}A)]^{1/2}$. In the case where x is a scalar, $\|x\|$ denotes its absolute value.

Definition 3. Let $f(t)$ be a function on $[0,\infty)$ of signal spaces. L_2 space and L_∞ space are defined as $L_2 = \{f : R^+ \to R \mid \|f\|_2 = \int_0^\infty |f|^2\,\mathrm{d}t < \infty\}$ and $L_\infty = \{f : R^+ \to R \mid \|f\|_\infty = \int_0^\infty \sup_{t\in[0,\infty)} |f|\,\mathrm{d}t < \infty\}$, respectively.

From a signal point of view, the signal in L_2 space has finite energy and that in L_∞ space is bounded.

Definition 4. The solutions of $\dot{x} = f(x,t)$ are said to uniformly ultimately bounded (UUB) if there exist constants b, c and $T(\alpha)$ for every $\alpha \in (0,c)$ such that: $|x(t_0)| < \alpha \Rightarrow |x(t)| < b,\ \forall t > t_0 + T$. They are globally UUB if above condition holds for an arbitrarily large α.

Lemma 1. *Let* $f(t)$ *be a time-varying function. If* $f(t), \dot{f}(t) \in L_\infty$ *and* $f(t) \in L_2$, *then* $f(t) \to 0$ *as* $t \to \infty$.

2.2. Description of fuzzy logic systems

An FLS is composed of four main components: a fuzzifier, a fuzzy rule base, a fuzzy inference engine and a defuzzfier as shown in Fig. 1. The fuzzifier maps non-fuzzy input space to fuzzy sets

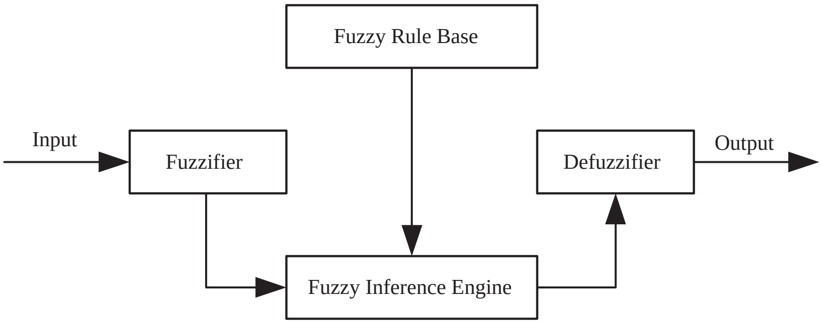

Fig. 1. Configuration of fuzzy logic systems.

and the defuzzifier performs an opposite task to map fuzzy sets to non-fuzzy output space. Fuzzy sets are characterized by their membership functions $\mu(\cdot)$'s. The fuzzy inference engine performs a mapping from fuzzy sets in input space $U \in R^n$ to fuzzy sets in output space $V \in R^m$ based on fuzzy rules. Finally, it is assumed that: $U = U_1 \times U_2 \times \cdots \times U_n, U_i \in R,\ (i = 1, 2, \ldots, n)$ and $V = V_1 \times V_2 \times \cdots \times V_m$, $V_k \in R,\ (k = 1, 2, \ldots, m)$. There are many choices for each part of an FLS. Therefore, a particular FLS could be any combination of these choices. In this paper, the following FLS is under the consideration.

The fuzzy rule base consists of a collection of fuzzy IF-THEN rules:

$$\begin{aligned} R^j &: \text{if } x_1 \text{ is } A_1^j \text{ and } x_2 \text{ is } A_2^j \text{ and } \cdots\cdots \text{ and } x_n \text{ is } A_n^j, \\ &\text{then } y_1 \text{ is } B_1^j \text{ and } y_2 \text{ is } B_2^j \text{ and } \cdots\cdots \text{ and } y_m \text{ is } B_m^j, \end{aligned} \tag{1}$$

where $x = [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$ and $y = [y_1, y_2, \ldots, y_m]^{\mathrm{T}}$ are input vectors and output vectors of the FLS, respectively. $A_i^j (i = 1, 2, \ldots, n)$ and $B_k^j (k = 1, 2, \ldots, m)$ are linguistic variables of fuzzy sets in subspace U_i and V_k, described by their membership functions $\mu_{A_i^j}(x_i)$ and $\mu_{B_k^j}(y_k),\ j = 1, 2, \ldots, M$. M is total number of the fuzzy rules. By using product inference, singleton-fuzzifier, and center-average defuzzifier strategies, output of the FLS can be expressed as

$$y(x) = \hat{\Omega}(x|\hat{\Theta}) = \frac{\sum_{j=1}^{M} \bar{y}^j \left(\prod_{i=1}^{n} \mu_{A_i^j}(x_i)\right)}{\sum_{j=1}^{M} \left(\prod_{i=1}^{n} \mu_{A_i^j}(x_i)\right)} = \hat{\Theta}\xi(x), \tag{2}$$

where $\mu_{A_i^j}(x_i)$ is membership function value of fuzzy variable x_i, $\bar{y}^j$ is the point at which $\mu_{B_k^j}(y_k)$ achieves its maximum value, and it is assumed that $\mu_{B_k^j}(\bar{y}^j) = 1$.

$$\hat{\Theta} = \begin{pmatrix} \bar{y}_1^1 & \bar{y}_1^2 & \cdots & \bar{y}_1^M \\ \bar{y}_2^1 & \bar{y}_2^2 & \cdots & \bar{y}_2^M \\ \vdots & \vdots & \ddots & \vdots \\ \bar{y}_m^1 & \bar{y}_m^2 & \cdots & \bar{y}_m^M \end{pmatrix}, \tag{3}$$

is adjustable parameter matrix and $\xi(x) = [\xi^1(x), \xi^2(x), \ldots, \xi^M(x)]^{\mathrm{T}}$ is fuzzy basis function vector, in which $\xi^j(x)$ is defined as

$$\xi^j(x) = \frac{\prod_{i=1}^n \mu_{A_i^j}(x_i)}{\sum_{j=1}^M \left(\prod_{i=1}^n \mu_{A_i^j}(x_i)\right)} \quad (j = 1, 2, \ldots, M). \tag{4}$$

FLS is selected to control nonlinear and uncertain robotic manipulator systems for two reasons. First, FLS has been proven to be capable of universally approximating any well-defined nonlinear functions over a compact set to arbitrary accuracy. Second, there exist other types of universal approximators such as multi-layer neural networks, radial basis function neural networks and so on. However, only FLS is constructed from IF-THEN rules using some specific fuzzy inference, fuzzification, and defuzzification strategies. Therefore, linguistic information from human experts can be incorporated into FLS in a systematic way [23].

2.3. Problem formulation

In this subsection, dynamical models of robotic manipulators with uncertainties and idea of CTC for robotic manipulators will be presented in detail.

According to Lagrange theory [26], dynamical equations of robotic manipulators with n serial links incorporating external disturbances can be expressed as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + f(q, \dot{q}) = \tau, \tag{5}$$

where $\ddot{q}, \dot{q}, q \in R^n$ are vectors of joint accelerations, velocities, and positions, respectively. $M(q) \in R^{n\times n}$ is a symmetric positive definite inertia matrix; $C(q, \dot{q})\dot{q} \in R^n$ being a vector of centripetal and Coriolis forces. $G(q) \in R^n$ denotes gravity vectors and $f(q, \dot{q}) \in R^n$ is unmodeled dynamics including friction terms and external disturbances and so on. $\tau \in R^n$ represents torque vectors exerting on joints. The structural properties of robotic manipulators hold such as boundedness of $M(q)$, $C(q, \dot{q})$ and $f(q, \dot{q})$.

For convenience, dynamical model (5) can be rewritten as the following compact form:

$$M(q)\ddot{q} + H(q, \dot{q}) + f(q, \dot{q}) = \tau, \tag{6}$$

where $H(q, \dot{q}) = C(q, \dot{q})\dot{q} + G(q)$. The task of trajectory tracking for robotic manipulators can be describe as: given desired trajectories of manipulator joints $q_{\mathrm{d}} \in R^n$, find control input τ such that actual trajectories q of manipulator joints in (6) can tend to q_{d} as time goes to infinity. In this paper, the following assumption is required.

Assumption 1. The desired trajectories q_{d} are continuous and bounded known functions of time with bounded known derivatives up to the second order.

The parameters $M(q)$, $C(q, \dot{q})$ and $G(q)$ in dynamical model (6) are functions of physical parameters of manipulators like links masses, links lengths, moments of inertial and so on. The precise values of these parameters are difficult to acquire due to measuring errors, environment and payloads variations. Therefore, here it is assumed that actual values $M(q)$, $C(q, \dot{q})$ and $G(q)$ can be separated as nominal parts denoted by $M_0(q), C_0(q,\dot{q}), G_0(q)$ and uncertain parts denoted by $\Delta M(q), \Delta C(q,\dot{q}), \Delta G(q)$, respectively. These variables satisfy the following relationships:

$$\begin{aligned} M(q) &= M_0(q) - \Delta M(q), \\ C(q,\dot{q}) &= C_0(q,\dot{q}) - \Delta C(q,\dot{q}), \\ G(q) &= G_0(q) - \Delta G(q). \end{aligned} \tag{7}$$

Suppose that dynamical models of robotic manipulators are known precisely and unmodeled dynamics are excluded, i.e. $\Delta M(q), \Delta C(q,\dot{q}), \Delta G(q)$ in (7) and $f(q,\dot{q})$ in (6) are all zeros. At this time, dynamical models (6) can be converted into the following nominal models:

$$M_0(q)\ddot{q} + H_0(q,\dot{q}) = \tau, \tag{8}$$

where $H_0(q,\dot{q}) = C_0(q,\dot{q})\dot{q} + G_0(q)$. Now considering the control law

$$\tau = M_0(q)(\ddot{q}_{\mathrm{d}} - K_{\mathrm{v}}\dot{e} - K_{\mathrm{p}}e) + H_0(q,\dot{q}), \tag{9}$$

where $\ddot{q}_{\mathrm{d},}\dot{q}_{\mathrm{d}}, q_{\mathrm{d}} \in R^n$ are desired trajectories vectors of joint accelerations, velocities, and positions, respectively. $e = q - q_{\mathrm{d}}$ is defined as trajectory tracking error vectors. K_{p} and K_{v} are proportional and derivative constant matrices, respectively. Substituting (9) into (8) yields

$$\ddot{e} + K_{\mathrm{v}}\dot{e} + K_{\mathrm{p}}e = 0. \tag{10}$$

It is obvious that errors will asymptotically tend to zero if proportional gain K_{p} and derivative gain K_{v} are chosen in the favorable situation. The control law (9) is termed as *Computed Torque Control* [11].

Remark 1. It is simple to understand that CTC approach is based on feedback linearization technique, which results in a linear time-invariant closed-loop system (10), implying acquirement of globally asymptotical stability. Furthermore, explicit conditions for performance indices may be obtained in terms of controller gain matrices. More specifically, globally asymptotical stability is guaranteed provided that K_{p} and K_{v} in (9) are symmetric and positive definite constant matrices.

According to analysis above, CTC strategy relies on strong assumptions that exact knowledge of robotic dynamics is precisely known and unmodeled dynamics has to be ignored, which is impossible in practical engineering. Motivated by separation (7), idea of CTC introduced above, and capability for approximating nonlinear functions of FLS, one can image that CTC is used to control nominal system and another FLS-based controller attaching to CTC for uncertain system can be designed. In this way, applying (7) and (9) to practical manipulator systems (6) yields

$$\ddot{e} + K_{\mathrm{v}}\dot{e} + K_{\mathrm{p}}e = \rho, \tag{11}$$

where $\rho = M_0^{-1}[\Delta M(q)\ddot{q} + \Delta H(q,\dot{q}) - f(q,\dot{q})]$, in which $\Delta H(q,\dot{q})$ satisfies $H(q,\dot{q}) = H_0(q,\dot{q}) - \Delta H(q,\dot{q})$. ρ is a function of joint variables, physical parameters, parameters variations, unmodeled dynamics and so on. On the other hand, ρ denotes the sum of structured uncertainty and unstructured uncertainty. Thus, here ρ is termed as lumped uncertain function of robotic dynamics.

Remark 2. From expression of ρ, it is easily seen that lumped uncertainty results from parameters variations including inertia uncertainty, gravity uncertainty, Coriolis uncertainty and unmodeled dynamics such as external disturbances. The existence of ρ causes CTC invalid and makes stability of robotic manipulators difficult to acquire.

Up to now, control objective can be restated as: seek a control law based on FLS as a compensator for CTC such that joint motions of robotic systems (6) can follow desired trajectories. Overall control law becomes

$$\tau = \tau_0 + \tau_{\mathrm{c}}, \tag{12}$$

where τ_0 is CTC defined like (9) and τ_{c} is a compensating torque to be determined below.

3. Design of fuzzy compensator controller

In this section, a fuzzy compensator controller serving as a compensator for CTC is considered and designed in detail. Firstly, it is assumed that the right-hand side in Eq. (11) can be represented by an ideal FLS as

$$\rho(z) = W^{*}\xi(z) + \varepsilon(z), \tag{13}$$

where $\varepsilon(z)$ is reconstruction error of FLS, $\xi(z) = [\xi^1(z), \xi^2(z), \ldots, \xi^N(z)]^{\mathrm{T}}$ is fuzzy basis function vector fixed by the designer and $\xi^i(z) = (i = 1, 2, \ldots, N)$ is corresponding to Eq. (4), and $z(t)$ is given by

$$z(t) = [\ddot{q}^{\mathrm{T}}, \dot{q}^{\mathrm{T}}, q^{\mathrm{T}}]^{\mathrm{T}}. \tag{14}$$

$W^{*} = [w_{ij}] \in R^{n\times N}$ in Eq. (13) is an optimal weight matrix satisfying that

$$W^{*} = \arg\min_{W} \left\{ \operatorname*{Sup}_{x\in D_z} |\hat{\rho}(z|\hat{W}) - \rho(z)| \right\}. \tag{15}$$

Here, D_z denotes the sets of suitable bounds of z. It is assumed that z never reaches boundary of D_z. In (15), $\hat{\rho}(z|\hat{W})$ is an estimation of $\rho(z)$, which can be approximated using an FLS as (2):

$$\hat{\rho}(z|\hat{W}) = \hat{W}\xi(z), \tag{16}$$

where $\hat{W}$ is adjustable weight matrix corresponding to Eq. (3). Define compensative control law τ_{c} in (12) as

$$\tau_{\mathrm{c}} = -M_0(q)\hat{\rho}(z|\hat{W}). \tag{17}$$

Using control law (12), closed-loop system becomes:

$$\ddot{e} + K_{\mathrm{v}}\dot{e} + K_{\mathrm{p}}e = \tilde{\rho}(z), \tag{18}$$

where expression of $\tilde{\rho}(z)$ is

$$\tilde{\rho}(z) = \rho(z) - \hat{\rho}(z|\hat{W}) = \varepsilon(z) + \tilde{W}\zeta(z), \tag{19}$$

with $\tilde{W} = W^{*} - \hat{W}$ denoting error of weight matrix.

Supposed that state vector is defined as $x = [e^{\mathrm{T}}, \dot{e}^{\mathrm{T}}]^{\mathrm{T}}$, the state-space equation has form as

$$\dot{x} = Ax + B\tilde{\rho}(z), \tag{20}$$

where $A = \begin{bmatrix} 0 & I \\ -K_{\mathrm{p}} & -K_{\mathrm{v}} \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ I \end{bmatrix}$. To proceed with subsequent development and closed-loop stability analysis, the following assumptions are made:

Assumption 2. The reconstruction error $\varepsilon(z)$ is bounded, i.e. $\|\varepsilon\| \leqslant c_\varepsilon$ for $\forall z \in D_z$ with known c_ε.

Assumption 3. The norm of optimal weight matrix are bounded so that $\|W^*\| \leqslant c_w$.

Remark 3. According to universal approximation theorem [23], the state variables of an FLS have to be limited in a given compact set so that optimal parameter exists. In this paper, compact set D_z in Eq. (15) is assumed to be large enough so that state variables remain within D_z under closed-loop control. Thus, Assumption 2 is reasonably made. Moreover, the upper bound c_ε of reconstruction error can be arbitrarily reduced by increasing number of fuzzy rules. Assumption 3 is used for stability analysis below [14].

Theorem. *Consider robotic manipulator systems with structured uncertainty and/or unstructured uncertainty* (5), *supposed that Assumptions* 1–3 *are satisfied. The control law is provided by* (12), *which consists of CTC part* (9) *and fuzzy part* (17). *The parameters in fuzzy part are tuned by*

$$\dot{\hat{W}} = \Gamma^{-1} B^{\mathrm{T}} P x \xi^{\mathrm{T}}, \tag{21}$$

where $\Gamma = diag(\Gamma_1, \Gamma_2, \ldots, \Gamma_n)$, $\Gamma_i > 0$ $(i = 1, 2, \ldots, n)$ *is gain matrix and* P *is positive definite solution of the following Riccati equation*:

$$A^{\mathrm{T}} P + PA + P^{\mathrm{T}} B B^{\mathrm{T}} P + Q = 0, \tag{22}$$

where Q *is a constant matrix with appropriate dimensions given in advance. Then*:

(1) $x(t)$ *in closed-loop systems are uniformly ultimately bounded.*
(2) *If reconstruction error* $\varepsilon \in L_2$, *i.e.* $\int_0^\infty \varepsilon^2(t)\,\mathrm{d}t < \infty$, *trajectory tracking errors of robotic manipulators tend to zero as time goes to infinity.*

Proof. (1) Consider the following Lyapunov-function candidate:

$$V = x^{\mathrm{T}} P x + Tr(\tilde{W}^{\mathrm{T}} \Gamma \tilde{W}), \tag{23}$$

with $\tilde{W} = W^* - \hat{W}$. Taking the time derivative of V along (20) results in

$$\begin{aligned} \dot{V} &= x^{\mathrm{T}}(A^{\mathrm{T}} P + PA)x + \tilde{\rho}^{\mathrm{T}} B^{\mathrm{T}} P x + x^{\mathrm{T}} P B \tilde{\rho} + 2Tr(\tilde{W}^{\mathrm{T}} \Gamma \dot{\tilde{W}}) \\ &= x^{\mathrm{T}}(A^{\mathrm{T}} P + PA)x + 2x^{\mathrm{T}} P B \tilde{\rho} - 2Tr(\dot{\hat{W}}^{\mathrm{T}} \Gamma \tilde{W}). \end{aligned} \tag{24}$$

Applying repeatedly the properties of trace of matrix and substituting (21) and (22) into (24) leads to

$$\begin{aligned} \dot{V} &= -x^{\mathrm{T}}(P^{\mathrm{T}} B B^{\mathrm{T}} P + Q)x + 2x^{\mathrm{T}} P B(\varepsilon + \tilde{W}\xi) - 2Tr(\tilde{W}^{\mathrm{T}} \Gamma \dot{\hat{W}}) \\ &= -x^{\mathrm{T}}(P^{\mathrm{T}} B B^{\mathrm{T}} P + Q)x + 2x^{\mathrm{T}} P B \varepsilon + 2x^{\mathrm{T}} P B \tilde{W} \xi - 2Tr(\tilde{W}^{\mathrm{T}} \Gamma \dot{\hat{W}}) \end{aligned}$$

$$
\begin{aligned}
&= -x^{\mathrm{T}}(^{\mathrm{T}}BB^{\mathrm{T}}P + Q)x + 2x^{\mathrm{T}}PB\varepsilon + 2Tr(\xi^{\mathrm{T}}\tilde{W}^{\mathrm{T}}B^{\mathrm{T}}Px) - 2Tr(\tilde{W}^{\mathrm{T}}\Gamma\dot{\hat{W}}) \\
&= -x^{\mathrm{T}}(P^{\mathrm{T}}BB^{\mathrm{T}}P + Q)x + 2x^{\mathrm{T}}PB\varepsilon + 2Tr(\tilde{W}^{\mathrm{T}}B^{\mathrm{T}}Px\xi^{\mathrm{T}}) - 2Tr(\tilde{W}^{\mathrm{T}}\Gamma\dot{\hat{W}}) \\
&= -x^{\mathrm{T}}(P^{\mathrm{T}}BB^{\mathrm{T}}P + Q)x + 2x^{\mathrm{T}}PB\varepsilon + 2Tr[\tilde{W}^{\mathrm{T}}(B^{\mathrm{T}}Px\xi^{\mathrm{T}} - \Gamma\dot{\hat{W}})] \\
&= -x^{\mathrm{T}}(P^{\mathrm{T}}BB^{\mathrm{T}}P + Q)x + 2x^{\mathrm{T}}PB\varepsilon + 2Tr[\tilde{W}^{\mathrm{T}}(B^{\mathrm{T}}Px\xi^{\mathrm{T}} - \Gamma(\Gamma^{-1}B^{\mathrm{T}}Px\xi^{\mathrm{T}}))].
\end{aligned} \tag{25}
$$

Then the equation above becomes

$$
\begin{aligned}
\dot{V} &= -x^{\mathrm{T}}(P^{\mathrm{T}}BB^{\mathrm{T}}P + Q)x + 2x^{\mathrm{T}}PB\varepsilon \\
&= -x^{\mathrm{T}}Qx - (B^{\mathrm{T}}Px - \varepsilon)^{\mathrm{T}}(B^{\mathrm{T}}Px - \varepsilon) + \varepsilon^{\mathrm{T}}\varepsilon \\
&\leqslant -x^{\mathrm{T}}Qx + \varepsilon^{\mathrm{T}}\varepsilon.
\end{aligned} \tag{26}
$$

It is easy to get $\dot{V} \leqslant -\lambda_{\min}(Q)\|x\|^2 + \|\varepsilon\|^2$, thus $V(x, \tilde{W})$ is negative outside the following compact set Σ_x:

$$
\Sigma_x = \left\{ x(t) | 0 \leqslant \|x(t)\| \leqslant \sqrt{\frac{1}{\lambda_{\min}(Q)}}\|\varepsilon\| \right\}. \tag{27}
$$

Assumption 3 implies that W is bounded (equivalently, $\tilde{W}$ is bounded), then $x(t)$ in closed-loop systems are uniformly ultimately bounded.

(2) Integration of Eq. (26) from $t = 0$ to ∞ can be rewritten as

$$
\int_0^\infty x^{\mathrm{T}}Qx \,\mathrm{d}t \leqslant \int_0^\infty \varepsilon^{\mathrm{T}}\varepsilon \,\mathrm{d}t + V(0) - V(\infty). \tag{28}
$$

Then

$$
\int_0^\infty \|x\|^2 \,\mathrm{d}t \leqslant k/\lambda_{\min}(Q), \tag{29}
$$

where $k = \int_0^\infty \varepsilon^{\mathrm{T}}\varepsilon \,\mathrm{d}t + V(0) - V(\infty)$. Noting that $V(t)$ is a non-increasing function of time and low bounded, this implies $V(0) - V(\infty) < \infty$ and $\int_0^\infty \varepsilon^2(\mathrm{t})\,\mathrm{d}t < \infty$, resulting in $k < \infty$. From (29), it is clear that $x \in L_2$ is satisfied. The boundedness of $x(t)$ above implies $x \in L_\infty$. From closed-loop dynamic equation (20) and boundedness of $x(t)$, $\tilde{W}(t)$, and $\varepsilon(t)$, one can get $\dot{x} \in L_\infty$. i.e. $x \in L_2 \cap L_\infty$, $\dot{x} \in L_\infty$. Thus, $\lim_{t\to\infty} x(t) = 0$ is achieved from Lemma 1. Therefore, whole closed-loop system is asymptotically stable, i.e. trajectory tracking errors converge to zero as time goes to infinity.

According to above analysis, the architecture of closed-loop system is shown in Fig. 2.

Remark 4. The core idea of the proposed scheme above lies in that lumped uncertainty can be approximated by an FLS. The updating law (21) of weight parameters is obtained using Lyapunov stability theorem. In expression (21), ξ is fuzzy basis function calculated by selected membership functions according to aforementioned Eq. (4).

Remark 5. The membership functions are designed by trial-and-play in the proposed scheme. In addition, the optimal weight is not required for fuzzy compensator as updating law (21) for parameter matrix of the FLS can result in convergence of trajectory tracking errors according to the proof above. What we need to know is only the existence of optimal weight, which is apparent from Remark 3.

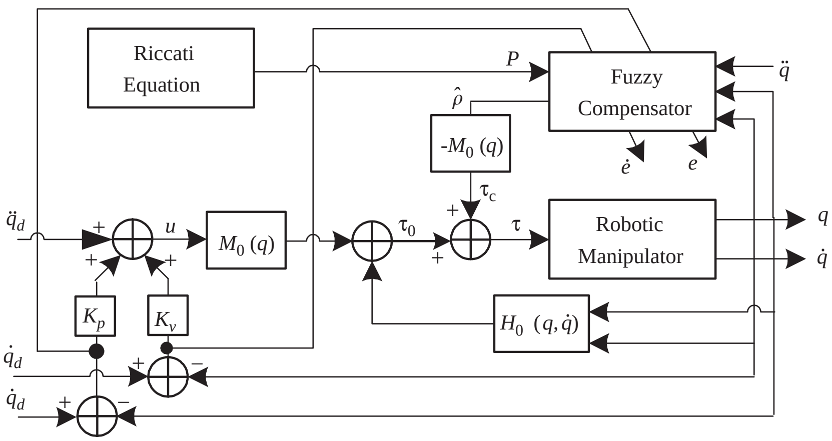

Fig. 2. Diagram of closed-loop systems.

Remark 6. All the parameters in the proposed scheme may be uncertain, which is true in practical situation. In some publications like [6], it is clearly unrealistic that only Coriolis terms, gravity terms, and friction terms may be uncertain, whereas inertia matrix was assumed known exactly.

Remark 7. Noted that the inputs of designed FLS include acceleration signal $\ddot{q}$ and CTC part also requires acceleration signal $\ddot{q}_\mathrm{d}$, here it is assumed that accelerations are measurable.

4. Simulation results

In this section, to illustrate validity of the proposed method, a two-link planar elbow manipulator [20] shown in Fig. 3 is simulated, whose dynamical model can be described by Eq. (5). With respect to the manipulator, the matrices in (5) are listed as follows:

$$M(q) = \begin{bmatrix} \theta_1 + \theta_3 + 2\theta_2 \cos(q_2) & \theta_3 + \theta_2 \cos(q_2) \\ \theta_3 + \theta_2 \cos(q_2) & \theta_3 \end{bmatrix},$$

$$C(q,\dot{q}) = \begin{bmatrix} -\theta_2 \sin(q_2)\dot{q}_2 & -\theta_2 \sin(q_2)\dot{q}_1 - \theta_2 \sin(q_2)\dot{q}_2 \\ \theta_2 \sin(q_2)\dot{q}_1 & 0 \end{bmatrix},$$

$$G(q) = \begin{bmatrix} g_{111} + g_{112} \\ g_{12} \end{bmatrix},$$

where

$$\theta_1 = I_1 + \left(\frac{m_1}{4} + m_2\right) l_1^2, \quad \theta_2 = \frac{m_2 l_1 l_2}{2}, \quad \theta_3 = I_2 + \frac{m_2 l_2^2}{4},$$

$$g_{111} = \left(\frac{1}{2} m_1 l_1 + m_2 l_1\right) g \cos(q_1), \quad g_{112} = g_{12} = \frac{1}{2} m_2 l_2 g \cos(q_1 + q_2),$$

with $q_i (i = 1, 2)$ denote angular positions (rad), $m_i (i = 1, 2)$ are links masses (kg), $l_i (i = 1, 2)$ represent links lengths (m), $I_i (i = 1, 2)$ are inertia moment of link i about corresponding center of

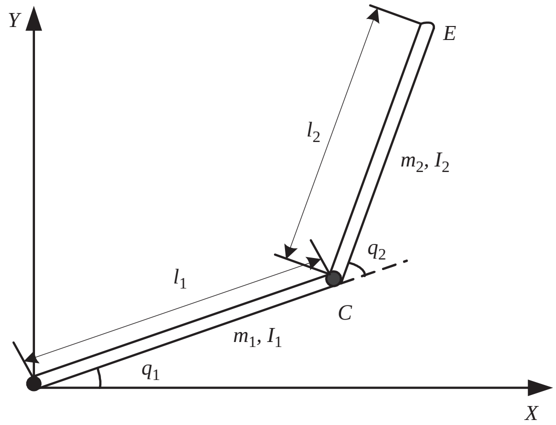

Fig. 3. Two-link planar elbow manipulator.

Table 1
Simulation parameters

	Mass (m_i/kg)		Length (l_i/m)		Inertial of Moment (I_i/kg m^2)	
	Nominal value	Actual value	Nominal value	Actual value	Nominal value	Actual value
Link 1	4.0	8.0	0.5	0.5	0.2	0.4
Link 2	4.0	8.0	0.5	0.5	0.2	0.4

gravity (kg m^2). $g = 10\,\mathrm{m/s}^2$ is acceleration of gravity. Point E in Fig. 3 denotes the end-effector of the robotic manipulator. Table 1 shows parameters used in simulations. For the purpose of comparison, simulation studies in three cases are conducted.

The object is to design control input in order to force joint variables $q = [q_1, q_2]^{\mathrm{T}}$ to track desired trajectories as time goes to infinity. Here, desired joint trajectories to be tracked are assumed to be $q_{\mathrm{d}} = [q_{1\mathrm{d}}, q_{2\mathrm{d}}]^{\mathrm{T}}$ with $q_{1\mathrm{d}} = 0.5\cos(t) + 0.2\sin(3t)$ and $q_{2\mathrm{d}} = -0.2\sin(2t) - 0.5\cos(t)$. In the following three cases, it is all assumed that initial positions of joints are $q_1(0) = q_2(0) = 4$ and initial velocities of joints are zeros.

Case 1: Firstly, CTC scheme applying to robotic manipulators with precise dynamical models is considered. The parameters of robotic manipulators are equal to nominal values in Table I. The designed parameters in CTC (9) are picked as

$$K_{\mathrm{v}} = diag(5.5, 5.5) \quad \text{and} \quad K_{\mathrm{p}} = diag(8, 8).$$

Figs. 4 and 5 show tracking performances of two joints in this case, respectively. In the two figures, the solid lines depict desired trajectories and the dotted lines denote actual trajectories, respectively. As can be seen from Figs. 4 and 5, tracking errors of two joints occur at the very beginning of the process and after a few seconds, actual trajectories and desired trajectories almost overlap each other. Thus, CTC method exhibits *Excellent* tracking performances when precise dynamical knowledge of robotic manipulators without unmodeled dynamics is known.

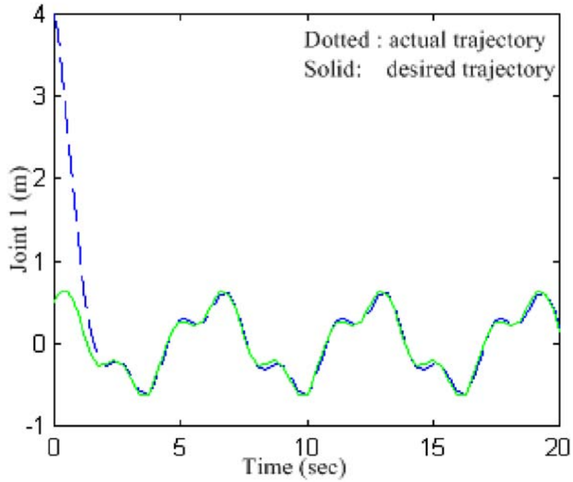

Fig. 4. Tracking of joint 1 when CTC applying to robotic manipulators without uncertainties.

Remark 8. The motivations of adopting control gains as above include two aspects. First, from Routh–Shure criterion, asymptotical stability can be guaranteed provided both K_{v} and K_{p} are positive definite. On the other hand, the values of gains are set up by trial-and-play based on static and transient performances from simulation results.

Remark 9. From expression (9), CTC approach is a linear control technique and relies on exact cancellation of nonlinear dynamics of robotic manipulators. Therefore, precise parameters in dynamical models have to be known. Additionally, there are only two design parameters so that they are easily adjusted according to system responses. However, requirements that exact knowledge of dynamics and ignorance of unmodeled dynamics make CTC method unattractive to industry.

Case 2: Here the same control laws with previous case in both forms and parameters are applied to robotic manipulators, which is supposed to be influenced by structured uncertainty (parameters variations) and unstructured uncertainty (external disturbances). In other words, the control schemes are designed according to nominal parameters instead of actual parameters. Moreover, external disturbances are considered and assumed to be $f(q,\dot{q}) = [\sin(t), \cos(t)]^{\mathrm{T}}$. The simulation results of tracking performances for the first link and those for the second link are shown in Figs. 6 and 7, respectively. In Figs. 6 and 7, desired trajectories and actual trajectories are depicted with solid lines and dotted lines, respectively. Compared with previous case, it can be easily acquired that tracking performance of CTC method greatly deteriorates when there is no exact dynamical knowledge of robotic manipulators available and unmodeled dynamics of robotic manipulators are taken into considerations.

Remark 10. Here, the simulation results of robotic manipulators subjected to simultaneous structured uncertainty and unstructured uncertainty are shown. As a matter of fact, it is found from lots of conducted simulations in this case that if unmodeled dynamics, i.e. external disturbances, are neglected and

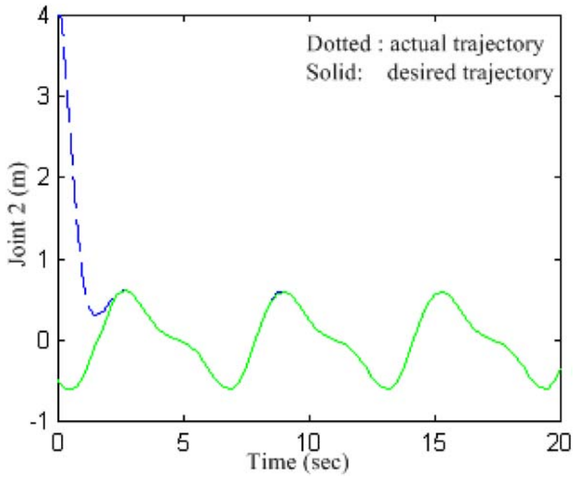

Fig. 5. Tracking of joint 2 when CTC applying to robotic manipulators without uncertainties.

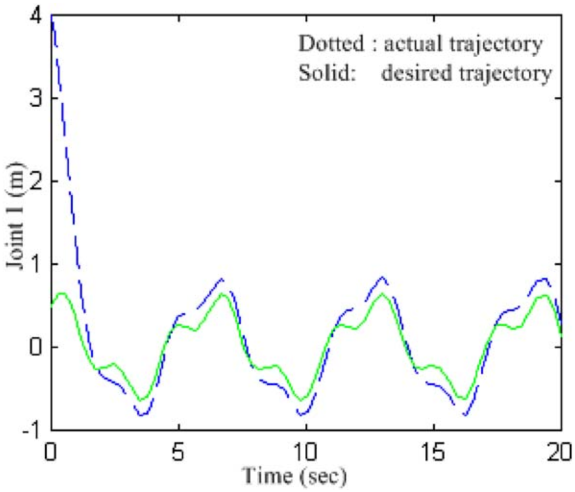

Fig. 6. Tracking of joint 1 when CTC applying to robotic manipulators with uncertainties.

parameters variations are considered, or opposite case, parameters variations are excluded and unmodeled dynamics are considered, then tracking performances are also much worse than case 1. Therefore, it is safely said that CTC approach is of good performances only when dynamical models of robotic manipulators are known precisely and free from unmodeled dynamics.

Case 3: From theoretical analysis and simulations above, if CTC is applied to uncertain robotic manipulator, then additional controller compensating for CTC should be required and appropriately designed. In this case, the proposed CTC plus fuzzy compensator is used to control robotic manipulators with uncertainties. Here, FLS is used for estimating lumped uncertainty function ρ in (11) due to parameters variations and unmodeled dynamics. The FLS has joint accelerations $\ddot{q}$, joint velocities $\dot{q}$, joint positions

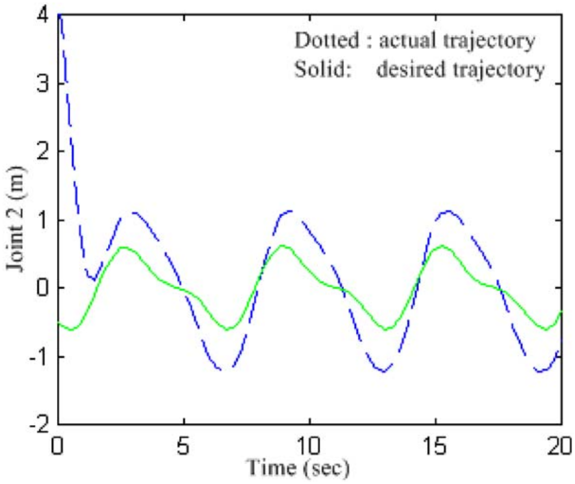

Fig. 7. Tracking of joint 2 when CTC applying to robotic manipulators with uncertainties.

q as inputs. Namely, input vectors of the FLS are of form like $z = [z_1^{\mathrm{T}}, z_2^{\mathrm{T}}, z_3^{\mathrm{T}}]^{\mathrm{T}}$ with $z_1^{\mathrm{T}} = [z_{11}, z_{12}]$, $z_2^{\mathrm{T}} = [z_{21}, z_{22}]$ and $z_3^{\mathrm{T}} = [z_{31}, z_{32}]$, where $z_{1i} = \ddot{q}_i$, $z_{2i} = \dot{q}_i$ and $z_{3i} = q_i$ for $i = 1, 2$. Additionally there are two joint angles. Therefore, it is clear that the dimension of the inputs and outputs of the FLS is 6 and 2, respectively. For each input vector, five Gaussian membership functions are defined as

$$\mu_{F_{ij}^1}(z_{ij}) = \exp\left(-\left(\frac{z_{ij}+2}{0.5}\right)^2\right), \quad \mu_{F_{ij}^2}(z_{ij}) = \exp\left(-\left(\frac{z_{ij}+1}{0.5}\right)^2\right),$$

$$\mu_{F_{ij}^3}(z_{ij}) = \exp\left(-\left(\frac{z_{ij}}{0.5}\right)^2\right), \quad \mu_{F_{ij}^4}(z_{ij}) = \exp\left(-\left(\frac{z_{ij}-1}{0.5}\right)^2\right),$$

$$\mu_{F_{ij}^5}(z_{ij}) = \exp\left(-\left(\frac{z_{ij}-2}{0.5}\right)^2\right) \quad i = 1, 2; \;\; j = 1, 2, 3.$$

To reduce the number of fuzzy rules, like [1], 30 ($2 \times 3 \times 5$) fuzzy rules are employed as follows:

$$R_{ij}^k : \text{if } z_{ij} \text{ is } F_{ij}^k \quad \text{then } \rho_1 \text{ is } B_1^k \quad \text{and} \quad \rho_2 \text{ is } B_2^k,$$

where $i = 1, 2;\; j = 1, 2, 3$ and $k = 1, 2, \ldots, 5$. ρ_1 and ρ_2 are two outputs of the FLS satisfying $\rho = [\rho_1, \rho_2]^{\mathrm{T}}$. In this case, the initial values of weight parameters $\hat{W}(0)$ are set equal to zero matrix. The positive definite matrix Q in Riccati equation (22) and the gain matrix Γ in updating law (21) are selected as $Q = diag(15, 15, 15, 15)$ and $\Gamma = diag(9, 9, 9, 9)$, respectively. The external disturbances are assumed to be $f(q, \dot{q}) = [\sin(t), \cos(t)]^{\mathrm{T}}$, also the same as case 2. The tracking performances for two joints of robotic manipulators with variations in parameters and unmodeled dynamics are shown in Figs. 8 and 9, respectively. The control efforts for joints 1 and 2 are shown in Figs. 10 and 11, respectively.

Remark 11. The *exclusive difference* between case 2 and this case lies in that designed controller in this case includes a fuzzy compensator, however, used control law in case 2 does not have fuzzy part. From

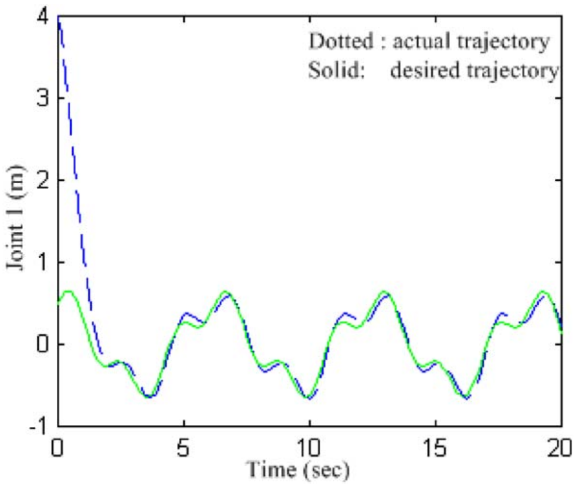

Fig. 8. Tracking of joint 1 when CTC plus fuzzy compensator applying to robotic manipulator with uncertainties.

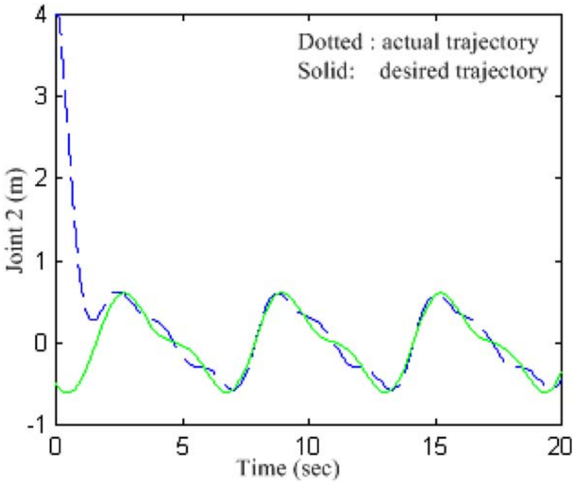

Fig. 9. Tracking of joint 2 when CTC plus fuzzy compensator applying to robotic manipulator with uncertainties.

the comparison between Figs. 8 and 6 and comparison between Figs. 9 and 7,it can be seen that fuzzy compensator greatly improves tracking performances.

Remark 12. The approaches combing FC and variable structure control are very popular in FC for robotic manipulators [3,15]. It is well known that there exists inevitable chattering due to introduction of variable structure control. However, one can say from control input signals in Figs. 10 and 11 that there is almost no high-frequent chattering even in presence of external noisy disturbances and fast varying parameters. Noted that simulation time here is taken as 20 s unlike relatively short simulation time in other publications.

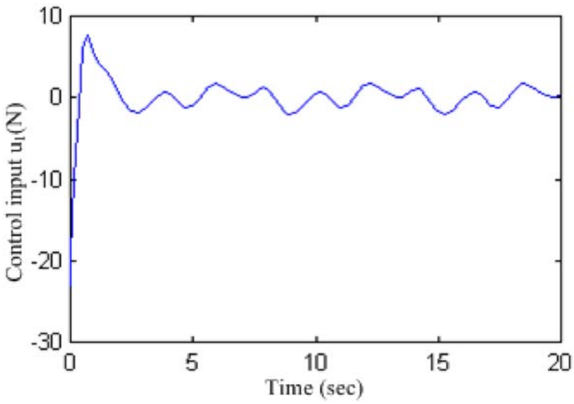

Fig. 10. Control input u_1 for joint 1.

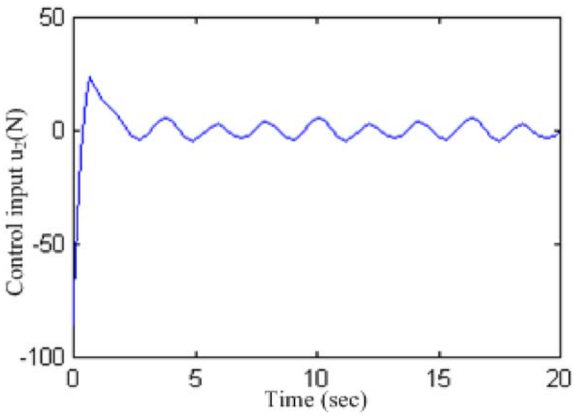

Fig. 11. Control input u_2 for joint 2.

5. Conclusion

This paper addresses trajectory tracking problems of robotic manipulators with structured uncertainty and unstructured uncertainty. Firstly, fundamental ideas and restrictions of conventional control method for robotic manipulators, so-called Computed Torque Control scheme, were presented in detail. Based on CTC as a nominal controller, a fuzzy compensation is designed to handle inevitable uncertainties. In the proposed scheme, the fuzzy compensative controller is used for approximating lumped uncertainty, whose weight parameters are adjusted based on Lyapunov stability theorem. In this way, stability of closed-loop systems can be guaranteed and tracking performances can be achieved. In order to illustrate feasibility of the proposed scheme, application of the proposed scheme to a two-link elbow planar manipulator is provided.

Furthermore, one contribution of this paper is that it provides a novel idea and method, i.e. CTC plus compensator, to design tracking controllers for robotic manipulators with uncertainties. It is indicated that tradition methods could not be neglected in designing controllers for complex plants. On the other hand, extensive applicable fields of FLS are also shown so that FC can handle control problems of complex systems by combining with other well-developed algorithms.

Acknowledgements

This work was partly supported by NSFC Projects (Grant Nos. 60334020, 60440420130, and 60475030), and MOST Projects (Grant Nos. 2003CB517106 and 2004DFB02100), China.

References

[1] P.T. Chan, A.B. Rad, J. Wang, Indirect adaptive fuzzy sliding model control: part two: parameter projection with supervisory control, Fuzzy Sets and Systems 122 (2001) 31–43.
[2] B.S. Chen, H.J. Uang, C.S. Tseng, Robust tracking enhancement of robot systems including motor dynamics: a fuzzy-based dynamic game approach, IEEE Trans. Fuzzy Systems 6 (1998) 538–552.
[3] Y.-C. Hsu, G. Chen, H.-X. Li, A fuzzy adaptive variable structure controller with applications to robot manipulator, IEEE Trans. Systems Man Cybernet. 31 (2001) 331–340.
[4] C. C Hua, X. Guan, G. Duan, Variable structure adaptive fuzzy control for a class of nonlinear time delay systems, Fuzzy Sets and Systems 148 (2004) 453–468.
[5] P.K. Khosla, T. Kanade, Experimental evaluation of nonlinear feedback and feedforward control schemes for manipulator, Int. J. Robot. Res. 7 (1988) 18–28.
[6] Y.T. Kim, Z.Z. Bien, Robust adaptive control in the presence of external disturbance and approximation error, Fuzzy Sets and Systems 148 (2004) 377–393.
[7] Y.H. Kim, F.L. Lewis, Neural network output feedback control of robot manipulators, IEEE Trans. Robot. Automat 15 (1999) 301–309.
[8] S. Labiod, M.S. Boucherit, T.M. Guerra, Adaptive fuzzy control of a class of MIMO nonlinear systems, Fuzzy Sets and Systems 15 (1) (2005) 59–77 http://www.elsevier.com/locate/fss.
[9] J.Y.S. Luh, Conventional controller design for Industrial robots-a tutorial, IEEE Trans. Systems Man Cybernet. 13 (1983) 298–316.
[10] Z. Man, X. Yu, K. Ashraghian, A robust adaptive sliding mode tracking control using an RBF neural network for robotic manipulators, Proc. IEEE Int. Conf. Neural Networks 5 (1995) 2403–2408.
[11] R.H. Middleton, G.C. Goodwin, Adaptive computed torque control for rigid link manipulators, System Control Lett. 10 (1988) 9–16.
[12] W.T. Miller, F.H. Glanz, L.G. Kraft, Application of a general learning algorithm to the control of robotic manipulators, Int. J. Robot. Res. 6 (1987) 84–98.
[13] T. Ozaki, T. Suzuki, T. Furuhashi, S. Okuma, Y. Uchikawa, Trajectory control of robotic manipulator using neural networks, IEEE Trans. Ind. Electron. 38 (1991) 195–202.
[14] S. Purwar, I.N. Kar, A.N. Jha, Adaptive control of robot manipulators using fuzzy logic systems under actuator constraints, Fuzzy Sets and Systems; in press. http://www.elsevier.com/locate/fss.
[15] F. Sun, Z. Sun, G. Feng, An adaptive fuzzy controller based on sliding mode for robot manipulator, IEEE Trans. Systems Man Cybernet. 29 (1999) 661–667.
[16] F. Sun, Z. Sun, P.Y. Woo, Neural network-based adaptive controller design of robotic manipulators with an observer, IEEE Trans. Neural Networks 12 (2001) 54–67.
[17] J.J.E. Slotine, W. Li, Adaptive manipulator control: a case study, IEEE Trans. Automat. Control 33 (1988) 995–1003.
[18] J.J.E. Slotine, W. Li, Applied Nonlinear Control, Prentice-Hall International Editions, New York, 1990.
[19] J.J.E. Slotine, S.S. Sastry, Tracking control of nonlinear systems using sliding surface with application to robot manipulator, Int. J. Control 38 (1983) 465–492.
[20] M.W. Spong, M. Vidyasagar, Robot Dynamics and Control, Wiley, New York, 1989.
[21] S.K. Tso, Y.S. Xu, H.Y. Shum, Variable structure model reference adaptive control of robot manipulators, Proc. IEEE Int. Conf. Robot. Automat. 1 (1991) 2148–2153.

[22] L.X. Wang, Stable adaptive fuzzy control of nonlinear systems, IEEE Trans. Fuzzy Systems 1 (1993) 146–155.
[23] J. Wang, A.B. Rad, P.T. Chan, Indirect adaptive fuzzy sliding model control: part one: fuzzy switching, Fuzzy Sets and Systems 122 (2001) 21–30.
[24] L.X. Wang, J.M. Mendel, Fuzzy basis function universal approximation, and orthogonal least square learning, IEEE Trans. Neural Networks 3 (1992) 807–814.
[25] S.W. Wijesoma, R.J. Richards, Robust trajectory following of robots using computed torque structure with VSS, Int. J. Control 52 (1990) 935–962.
[26] S.Y. Yi, M.J. Chung, A robust fuzzy logic controller for robot manipulators with uncertainties, IEEE Trans. Systems Man Cybernet. 27 (1997) 706–713.
[27] L. Zadeh, Fuzzy sets, Inform. Control 8 (1965) 338–353.