Robotica (2007) volume 25, pp. 269–281. © 2006 Cambridge University Press
doi:10.1017/S0263574706003067 Printed in the United Kingdom

Locomotion control of a hydraulically actuated hexapod robot by robust adaptive fuzzy control and dead-zone compensation

Ranjit Kumar Barai*† and Kenzo Nonami‡

†*Graduate School of Science and Technology, Chiba University, 1-33 Yayoi-cho, Inage-ku, Chiba 2638522, Japan.*
‡*Department of Electronics and Mechanical Engineering, Chiba University, 1-33 Yayoi-cho, Inage-ku, Chiba 2638522, Japan.*

(Received in Final Form: July 31, 2006. First published online: October 12, 2006)

SUMMARY
This investigation presents locomotion control of a hydraulically actuated six-legged humanitarian demining robot by robust adaptive fuzzy control in conjunction with the dead zone compensation technique within independent joint control framework. For proper locomotion of the demining robot, accurate tracking of the desired joint trajectory is very important. However, high degree of nonlinearity, the uncertainties due to changing hydraulic properties, and delay due to the flow of oil and dead zone of the proportional electromagnetic control valve results into an inaccurate plant model for the hydraulically actuated robotic joints. Consequently, model-based classical control techniques result into a large tracking error. Therefore, adaptive fuzzy control technique, being a model independent control paradigm for complex and uncertain systems, is a good choice for such systems. In this work, a hydraulic dead zone compensated robust adaptive fuzzy control law has been proposed for locomotion control of hydraulically actuated hexapod demining robot. The experimental results exhibit a fairly accurate trajectory tracking of the leg joints and, consequently, very stable locomotion of the walking robot.

KEYWORDS: Robust adaptive fuzzy control (RAFC); Hydraulic actuator; Six-legged walking robot; Robot locomotion.

1. Introduction

During the last two decades, there has been a growing interest towards adaptive fuzzy control. This is mainly because of their smaller processing time as compared to the classical adaptive controllers and their capability to approximate any higher order nonlinear systems by applying a set of *IF-THEN* rules. Being a model independent and heuristics dependent control strategy, it can successfully handle the situations where uncertainty is involved in the system model. Wang[6] proved that fuzzy systems are universal approximators and the output of the system can be represented as a linear combination of the fuzzy regressor or *Basis Function*. This was indeed a breakthrough in the history of fuzzy systems. This discovery inspired many researchers to apply fuzzy control techniques in nonlinear systems.[4]

When there is uncertainty in the dynamics of the system, an accurate linear parameterizing model for the design of a classical adaptive control is very difficult to obtain. Fuzzy technique, being a model independent approach, can be successfully applied in such situations because it does not require the parameterization model. We will see later that the fuzzy model of the direct adaptive fuzzy controller in terms of the basis function is naturally represented in the same form as a linear parameterization model. However, since the fuzzy representation is approximate in nature, it may be insufficient to achieve the desired accuracy in tracking with mere adaptive fuzzy control. Further, the approximation error crept into the feedback loop makes it difficult to guarantee the stability of the closed-loop system.[2, 4] This necessitates the development of a robust adaptive fuzzy control law.

Few robust adaptive fuzzy controllers are reported in the literature. In ref. 2–4, robust adaptive fuzzy control methods are described. These are mainly Lyapunov method based designs to guarantee robust stability. However, they needed a supervisory controller-like component in the control input, which is basically a discrete switching function and may cause unnecessary chattering. An improvement has been suggested over the earlier methods in ref. 10, but due to the need for double adaptation, it is very difficult to implement it in real time. In ref. 11, a fuzzy robust control design has been presented that relies on Neural Network to determine the adaptation law and incorporates a projection operator to maintain stability, which is difficult to estimate in most of the practical applications.[12] A persistence excitation concept has been introduced in ref. 12 to design robust adaptive fuzzy control law. However, it needs a time-varying low-pass filter to produce smooth adaptation which adds extra complexity in the design and its implementation.

There are also a few implementations of classical adaptive control and fuzzy-based control for hydraulic robot mechanism or similar applications. A robust adaptive control law has been derived based on the classical method which needs an accurate model of the robotic mechanism.[5] In ref. 19, a detailed experimental implementation has been presented for a PD type fuzzy controller for controlling a class of hydraulically actuated industrial robots. An attempt has been made to cope up with the dead zone of similar nonlinear systems by the adaptive sliding mode technique.[20] A detailed account of our hydraulically actuated hexapod robot, COMET-III, controlled by preview sliding mode technique, has been presented in ref. 18. However, due to the presence of the sliding mode control block in the feedback loop, the occurrence of undesirable chattering is unavoidable.

* Corresponding author. E-mail: ranjit@graduate.chiba-u.jp

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

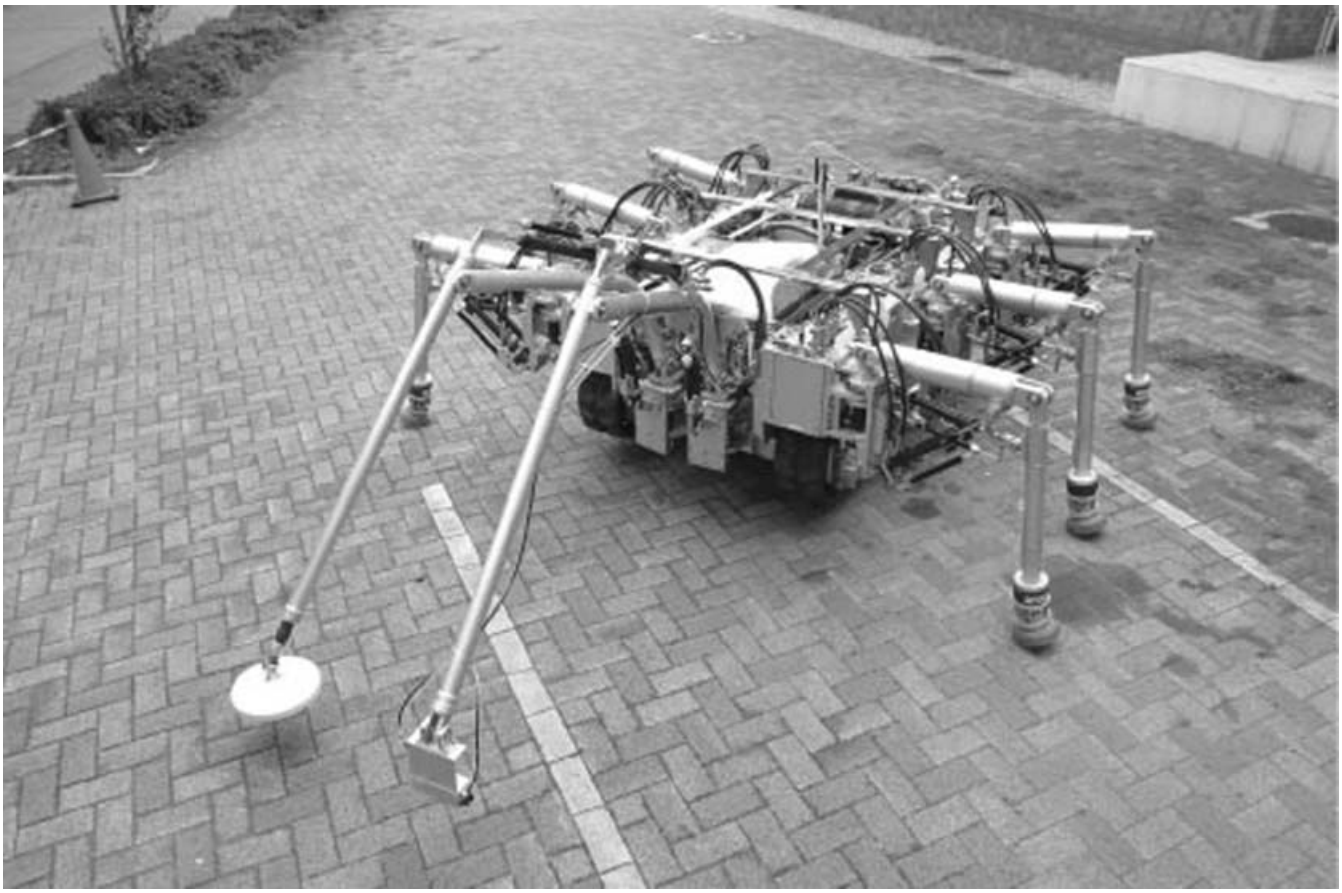


Fig. 1. Humanitarian demining robot COMET-III.

In this investigation, we will present the locomotion control of COMET-III by robust adaptive fuzzy control in conjunction with the hydraulic dead zone compensation technique within an independent joint control framework. COMET-III was built for humanitarian demining and actuated by electro-hydraulic servo system to provide sufficient power to walk over uneven terrain, for carrying the payload of the demining tools like mine detection manipulator and mine gripper (or mine marker), and for actuating the demining tools. Electro-hydraulic robotic systems are rich in various nonlinearities as well as uncertainties. Therefore, it is very difficult to obtain an accurate model of electro-hydraulic robotic system for the application of classical control principles. Moreover, when independent joint control method is applied in robotic systems, parameters like inertia, Corriolis, gravity, etc. act as disturbances to the system. While controlling the joints to follow the desired trajectory, delay due to flow of oil and dead zone of the proportional electromagnetic control valve pose a serious problem. Fuzzy control system being a model independent technique and universal approximator, adaptive fuzzy control technique is a proper choice for this situation. In this paper, a hydraulic dead zone compensated robust adaptive fuzzy control law is proposed for controlling the hydraulically actuated leg joints to achieve an accurate foot trajectory tracking. The adaptation law is designed by Lyapunov synthesis method and the dead zone compensation signal is designed by one-step-ahead[21] control principles. The experimental results exhibit a very stable locomotion system of the walking robot with a small tracking error. To the best of our knowledge, control of electro hydraulic servo system of the walking robots with robust adaptive fuzzy control technique and at the same time dead zone compensation to achieve better tracking of the commanded trajectory has not yet been addressed in the literature.

The rest of the paper is organized as follows: A brief description of the hydraulically actuated six-legged robot, COMET-III, is given in Section 2. The proposed robust adaptive fuzzy controller and the dead zone compensation are presented in Section 3. The real time implementation of the algorithm for the locomotion control of the walking robot COMET-III is presented in Section 4. In Section 5, we have shown the experimental results and, finally, in Section 6, the conclusion is presented.

2. Hydraulically Actuated Six-Legged Robot COMET-III

The humanitarian demining robot, COMET-III is a six-legged walking robot, each leg having three degrees of freedom which ensures stable walking on rough terrains.[8] The picture of COMET-III is shown in Fig. 1. The driving force is based on hydraulic power of 14 MPa pressure that is enough to generate sufficient hydraulic power to walk over irregular terrain[8] with the payload of the demining tools and good speed of locomotion.[18] The demining tools like mine detection manipulator and the mine gripper (or mine marker) shares the same source of hydraulic power for their actuation.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

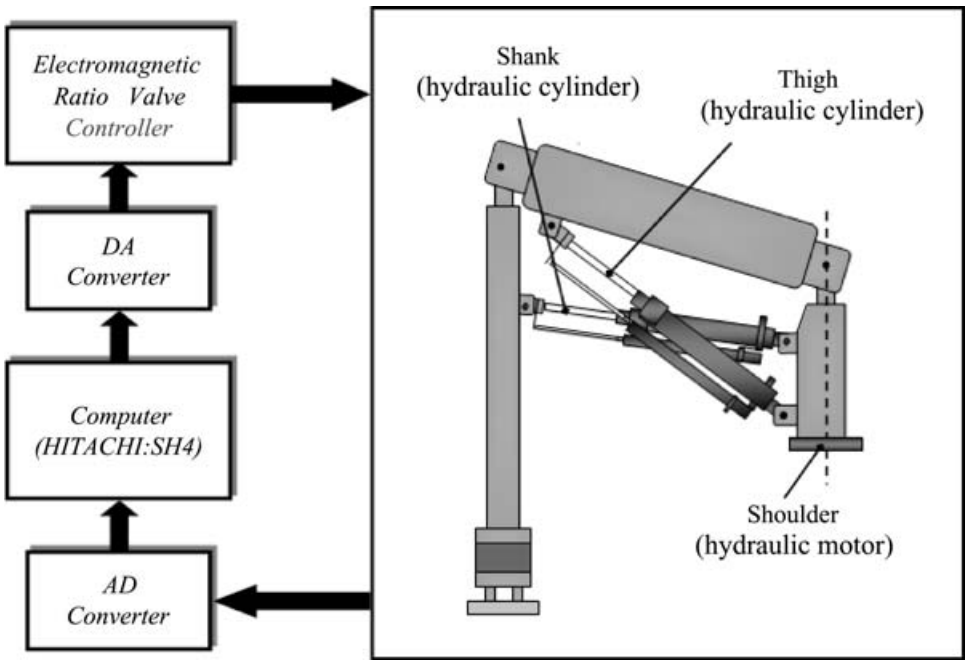


Fig. 2. Schematic diagram of COMET-III leg mechanism and its controller electronics.

The shoulder joints of each leg are driven by hydraulic motors. The thigh and shank joints of each leg are driven by hydraulic cylinders. Therefore, the angular positions of the hydraulic motors are measured by rotary potentiometers, whereas the positions of each cylinder are measured by linear potentiometers. The schematic diagram of COMET-III leg mechanism and its controller electronics is presented in Fig. 2. COMET-III has another mode of locomotion—a rubber crawler driven by hydraulic motors. The locomotion speed by the walking mode of legs and running mode of the crawler are 300 m and 3 km per hour, respectively. COMET-III can climb up a slope of 30° by crawler as well as by legs. Table I summarizes the main features of COMET-III.

In this work, we have restricted the locomotion of COMET-III to backward-forward walking with a tetrapod gait. The schematic diagram of the tetrapod gait is shown in Fig. 3. The gait command is issued by the operator through a tele-operation over a wireless serial data link. Three embedded computers (SH4) are connected in TCP/IP to form a local area network. This enables a simple and faster distributed control architecture. The locomotion control algorithm is executed by the SH4 computer-2, SH4 computer-1 is entrusted with the task of tele-operation, and the SH4 computer-3 is entrusted with the task of manipulation of the mine detection and removal (or mine marking) arms. The gait command is received by the SH4 computer-1 from the user over the wireless serial link and is passed to the SH4 computer-2 over TCP/IP network. From the gait command received, it first generates a reference trajectory in the Cartesian task space. Then, the task space trajectory is converted to joint space trajectory for each joint of all the legs. This joint space trajectory becomes the input command to the controllers of each joint. The control architecture of COMET-III is shown in Fig. 4. Here, during implementation, we have adopted an independent joint control scheme[16] for it's simplicity. There are several literatures on robotics that discuss about the various control schemes of robot manipulators. These control schemes primarily consider the equation of motion derived by either Newton–Euler's formulation or Lagrange's formulation and calculate the torque required by each joint

Table I. Hardware specification of COMET-III.

Height	870 mm
Width	2550 mm
Length	4154 mm
Weight	1200 kg
Payload	300 kg
Material	Aluminium alloy, SUS304
Crawler	Rubber crawler
Hydraulic tank	40 L
Gasoline tank	20 L
Control valves	26
Engine	1
Displacement	653 cc
Max. output	22 ps/3600 rpm
Max. torque	4.65 kgf/m
Legs	6
Shoulder	315 mm
Thigh	615 mm
Shank	870 mm
Range of movement of the shoulder	$-80^\circ \sim +80^\circ$
Range of movement of the thigh	$-10^\circ \sim +30.7^\circ$
Range of movement of the shank	$-15.9^\circ \sim +30^\circ$
Embedded microcomputer	3
Data bus	Compact PCI
Processor	SH4
Clock frequency	198 MHz
Data width	32 bit
Cache memory	Command 8 K + Data 16 K
Network protocol	TCP/IP

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

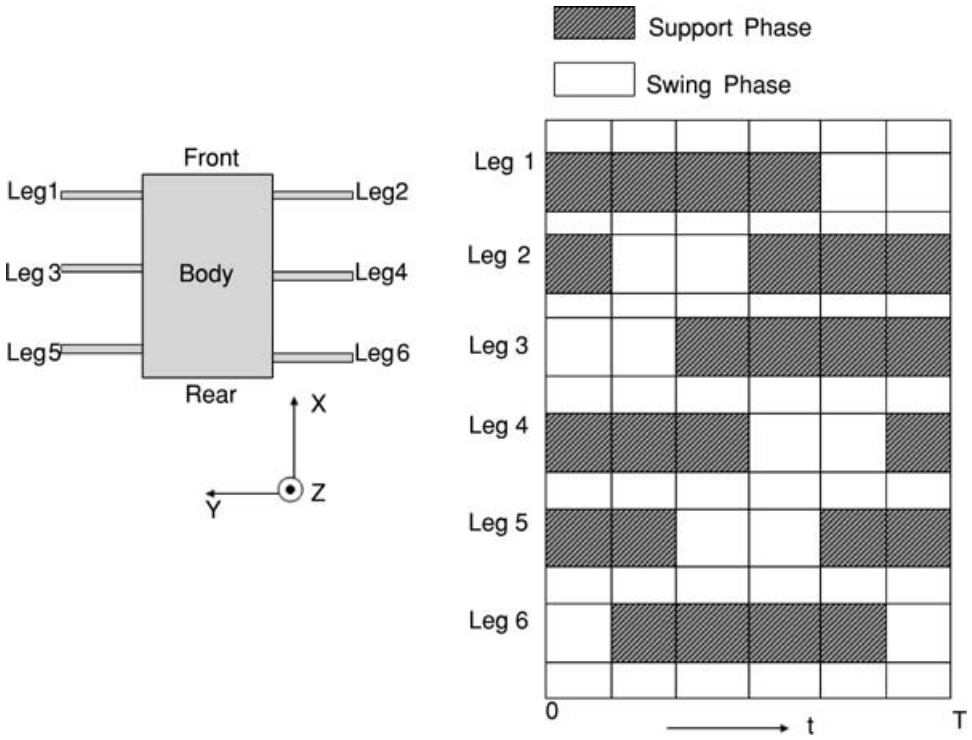


Fig. 3. Schematic diagram of the tetrapod gait of COMET-III.

to generate the necessary velocity and acceleration in order to follow the reference trajectory. However, these schemes require accurate model parameters of the robotic manipulator like inertia, Corriolis and centrifugal terms, and gravity parameter matrix; all of them not only nonlinear but also highly coupled and quite difficult to determine.

In our case, the leg joints are actuated by hydraulic cylinders or hydraulic motors. Compared to the conventional walking robots, where the leg joints are driven by electric motors, the controller design for the leg joints driven by hydraulic cylinders and hydraulic motors are more difficult. This is because hydraulic actuators contribute towards additional nonlinearities and stronger coupling among the dynamics of various joints.[5] Moreover, the hydraulic actuators are subjected to nonsmooth and discontinuous nonlinearities due to directional changes of valve opening

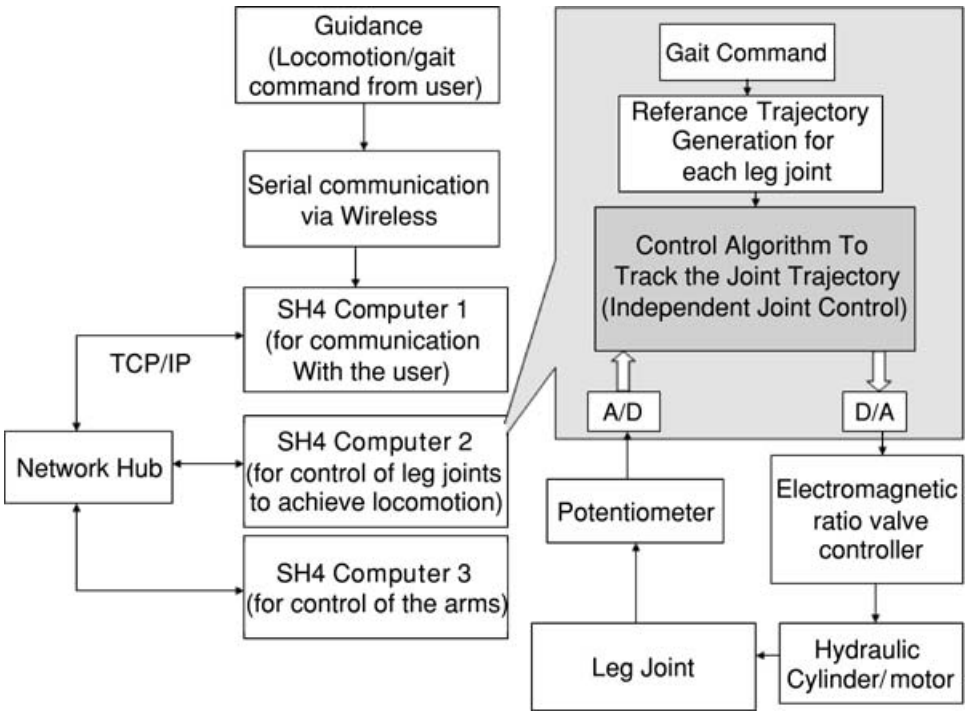


Fig. 4. Control architecture of COMET-III.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

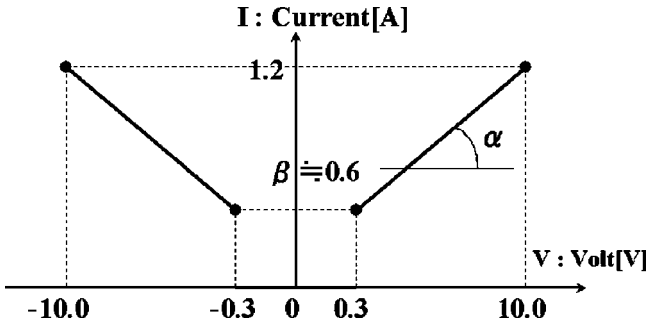


Fig. 5. Proportional electromagnetic valve characteristics.

and frictions. Finally, a hydraulically actuated leg mechanism experiences a large amount of model uncertainties, including the large changes in the load on the foot, large variation of parameters (like bulk modulus), oil leakage, air bubbles, and temperature.[18] Therefore, the situation is very complex to determine an accurate model with high degree of accuracy. As already explained, the situation is even more complicated due to the presence of dead zone of the proportional electromagnetic control valves, as is obvious from it's characteristics shown in Fig. 5. If conventional control methods like proportional-integral-derivative (PID) is employed, several problems related to hydraulic robot mechanism control like the time delay associated with the oil flow and valve dead-zone remain unsolved and the tracking performance is poor. In order to cope up with the non-linearities and uncertainties associated with the hydraulic mechanisms, we have developed a robust adaptive fuzzy controller and a hydraulic dead zone compensation scheme, and applied it for the locomotion control of COMET-III within the independent joint control framework. Because fuzzy controllers are model independent and if we consider the various time-varying nonlinear and coupling terms as "disturbances and uncertainties" penetrating into the "plant" (leg joint), these can be tackled very well by robust adaptation of the controller parameters. In the next section, we will derive the robust adaptive fuzzy control law.

3. Design of Control Law

Adaptive fuzzy controllers fall into two categories: direct adaptive fuzzy controller and indirect adaptive fuzzy controller. In the direct adaptive fuzzy control, the fuzzy rules are directly incorporated into the controller itself. Thus the rules describe the control input, making it easier to implement, whereas in indirect adaptive fuzzy control, the fuzzy rules describe the plant and then, the control input is derived from that plant description.

Here, we have proposed the direct robust adaptive fuzzy control law and its two components. One component is responsible for feedback control and is responsible for handling parameter variation and various uncertainties associated with the controlled object. The other component is responsible for solving the problem of the time delay in the output response that is caused by the hydraulic oil flow and the dead zone of the proportional electromagnetic control valves. We will present these components in the following subsections.

3.1. Design of direct robust adaptive fuzzy controller

In this section, first we will present the theoretical development of our proposed direct robust adaptive fuzzy controller that can control a class of nonlinear systems, like the electro-hydraulic robotic system. First we consider the nth order nonlinear dynamic system of the following form:

$$x^n = f\left(x, \dot{x}, \ldots, x^{(n-1)}\right) + g\left(x, \dot{x}, \ldots, x^{(n-1)}\right)u \quad y = x \tag{1}$$

where f is an unknown but bounded continuous function, g is assumed to be a known function, and $u \in R$ and $y \in R$ are the input and output of the system respectively. Let, $x = (x, \dot{x}, \ddot{x}, \ldots, x^{(n-1)})^T \in R^n$ be the state vector of the system which is assumed to be available. The control objective is to force y to follow a given bounded reference signal y_m under the constraint that all signals must be bounded. Hence, we have determined a feedback control law based on fuzzy logic systems and also an adaptation law for adjusting the parameters of the fuzzy logic systems to satisfy the following conditions:

(i) The closed-loop system must be globally stable, in the sense that all variables must be uniformly bounded.
(ii) The tracking error, $e = y - y_m$ should be as small as possible under the constraint in (i).

Thus, we would like to ensure robustness by ensuring global stability and lowering the tracking error. Then, our design objective is to impose an adaptive fuzzy control algorithm so that the following asymptotically stable tracking is achieved:

$$e^{(n)} + k_1 e^{(n-1)} + \cdots + k_n e = 0 \tag{2}$$

where the polynomial $h(s) = s^n + k_1 s^{n-1} + \cdots + k_{n-1}s + k_n$ is a Hurwitz polynomial and is represented by the characteristic equation given in Eq. (2). The roots of the $h(s)$ should lie in the open left half of the s-plane via the adequate choice of $k = (k_1, k_2, \ldots, k_n)$. In Eq. (2), if we incorporate the parameter error $\phi = \theta - \theta^*$, where θ is the actual parameter and θ^* is the estimated parameter, then the error differential equation takes the following form:

$$e^{(n)} = -k_1 e^{(n-1)} - k_2 e^{(n-2)} - \cdots - k_n e + K\phi^T(\xi(x) \tag{3}$$

where K is a positive constant and $\xi(x)$ is an arbitrary but known signal.

Figure 6 shows a general description of an adaptive fuzzy control system.[4] An adaptive fuzzy system is defined as a fuzzy logic system equipped with a learning algorithm, where the fuzzy system is constructed from a set of fuzzy IF-THEN rules using fuzzy logic principles, and the learning algorithm adjusts the parameters of the fuzzy system based on the training information.

The fuzzy logic system performs a mapping from $U \in R^n$ (universe of discourse of input fuzzy sets, in fuzzy logic literature) to $V \in R$ (universe of discourse of output fuzzy sets, in fuzzy logic literature). It comprises of four components: fuzzifier, fuzzy rule base, fuzzy inference engine, and defuzzifier. A detail review of fuzzy system and fuzzy mathematics can be found in ref. 9. In case of an adaptive fuzzy system, an additional block called "adaptation

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

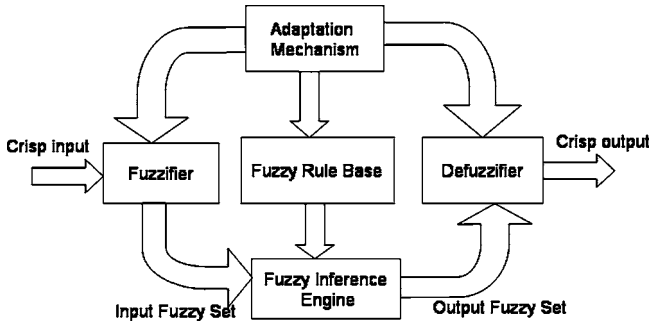


Fig. 6. Adaptive fuzzy systems.

block" is necessary that changes the parameters of the fuzzy controller. In this paper, we have considered that the fuzzy rule base consists of M number of rules for a multi-input single-output fuzzy system and has the following form:

$$R^l : \text{IF } x_1 \text{ is } \mathrm{F}_1^l \text{ AND} \ldots \text{AND, } x_n \text{ is } \mathrm{F}_n^l \text{ THEN y is } G^l \qquad (4)$$

where $x = (x_1, x_2, \ldots, x_n)^T \in U$, and $y \in V \subset R$ are the input and output variables of the fuzzy system, respectively, F_i^l and G^l are fuzzy sets and are characterized by linguistic terms like "BIG", "SMALL", etc., and $l = 1, \ldots, M$. By using singleton fuzzifier, product inference, and center-average defuzzification, the fuzzy logic system can be expressed as:

$$y(x) = \frac{\sum_{l=1}^{M} \bar{y}^l \left(\prod_{i=1}^{n} \mu_{F_i^l}(x_i) \right)}{\sum_{l=1}^{M} \left(\prod_{i=1}^{n} \mu_{F_i^l}(x_i) \right)}. \qquad (5)$$

Here, $\bar{y}^1$ is the point where μ_G^l achieves its maximum value, therefore, we can assume that $\mu_G^l(\bar{y}^l) = 1$.

Then, Eq. (4) can be written as[3]

$$y(x) = \theta^T \xi(x) \qquad (6)$$

Where $\theta = (\bar{y}^1, \ldots, \bar{y}^M)^T$ is the parameter vector, and $\xi(x) = (\xi^1(x), \ldots, \xi^M(x))^T$ is called a *Basis Function*[6] and is defined as

$$\xi(x) = \frac{\prod_{i=1}^{n} \mu_{F_i^l}(x_i)}{\sum_{l=1}^{M} \left(\prod_{i=1}^{n} \mu_{F_i^l}(x_i) \right)}. \qquad (7)$$

There are two main reasons for using the fuzzy logic system of Eq. (6) as the basic building block[3,4] for adaptive fuzzy controller. First, the fuzzy logic system in the form of Eq. (6) is proven in[6] as a universal approximator, i.e., for any given real continuous function f on the compact set U, there exists a fuzzy logic system in the form of Eq. (6) such that it can uniformly approximate f over U to an arbitrary accuracy. Therefore, the fuzzy logic system of Eq. (6) is qualified as the basic building block of adaptive controllers of nonlinear systems. Second, the fuzzy logic system of Eq. (6) is constructed from the fuzzy IF-THEN rules of Eq. (4) using some specific fuzzy inference, fuzzification, and defuzzification strategies. Therefore, linguistic information about a physical system can be directly incorporated in the controllers. It must be noted here that the form of Eq. (6) is just the same as the linear parameterization model.[17]

If both $f(x)$ and $g(x)$ in Eq. (1) are known and there are no external disturbances, then the control law of Eq. (8) can be applied to the nonlinear system [Eq. (1)] to asymptotically achieve the error dynamic Eq. (2).

$$u = \frac{1}{g(x)} \left[-f(x) + y_m^{(n)} + k^T e \right]. \qquad (8)$$

However, the functions $f(x)$ and $g(x)$ are actually not known, therefore, the control law [Eq. (8)] could be approximated by a fuzzy logic system of the form represented in [Eq. (6)]. Then the control law is

$$u = \theta^T \xi(x). \qquad (9)$$

In Eq. (9), the elements of the parameter vector of the controller$= (y^{-1}, \ldots, y^{-M})^T$ are varied based on some adaptation law to cope up with uncertain plant parameter variation. Here, we will develop an adaptation scheme that will enable the overall system to be endowed with robustness and, at the same time, with faster adaptation to plant parameter variation. We can develop an adaptation law in the same line as discussed in ref. 17, because after an approximation of the system [Eq. (1)] by the fuzzy technique, the system takes the form of a linear parametric model that is required by the classical adaptive control.

Let the adaptation law be given by

$$\dot{\theta} = -\gamma \xi(x) B^T P e \qquad (10)$$

where γ is the adaptation rate, P is a symmetric positive definite matrix and is the solution of the following Riccati-like equation

$$PA + A^T P + Q - \frac{2}{r} PBB^T P + \frac{1}{\rho^2} PBB^T P = 0 \qquad (11)$$

where r is a positive weighing factor and ρ is an attenuation level. A and B matrices are given as

$$A = \begin{bmatrix} 0 & 1 & 0 & \ldots\ldots & 0 \\ 0 & 0 & 1 & \ldots\ldots & 0 \\ 0 & 0 & 0 & \ldots\ldots & 0 \\ .. & .. & .. & \ldots\ldots & .. \\ -k_n & -k_{n-1} & -k_{n-2} & \ldots\ldots & -k_1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ . \\ . \\ 1 \end{bmatrix}. \qquad (12)$$

Now, due to the approximation error caused by the fuzzy system, modification of the adaptation law is necessary[10] so that the system does not become unstable due to a large approximation error. We will modify Eq. (10) with a leakage term,[1,17] called continuous switching leakage. Then, the

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

modified adaptation law is

$$\dot{\theta} = -\gamma \xi(x) B^T P e - \gamma \sigma_s \theta \tag{13}$$

where σ_s is called continuous switching function and is represented as

$$\sigma_s = \begin{cases} 0, & \text{if } |\theta(t)| < M_0 \\ \sigma_0 \left(\dfrac{|\theta(t)| - M_0}{|\theta(t)|} \right), & \text{if } M_0 \le |\theta(t)| \le 2M_0 \\ \sigma_0, & \text{if } |\theta(t)| > 2M_0 \end{cases} \tag{14}$$

where ρ_0 and M_0 are design constants. The constraints for the design of σ_0 and M_0 are $\sigma_0 > 0$, $M_0 \ge |\theta^*|$, and $|\theta(0)| \le M_0$. During implementation, these design constants are chosen by various trial and errors. The advantage of using the continuous switching leakage is that the adaptation law is not a discontinuous one.

Theorem 1: If the adaptive system described by Eqs. (9)–(14),

(i) is satisfying [Eq. (3)],
(ii) having the adaptation law of the form given in Eq. (13),
(iii) with P being a positive definitive matrix fulfilling the Eq. (11) for some positive definite matrix Q,

then a stable tracking performance is achieved and the tracking error converges to the neighborhood of zero as time $t \to \infty$.

The proof of Theorem 1 is given in the Appendix.

3.2. Design of electro-hydraulic dead-zone compensator

The system of Eq. (1) assumes that there is no time delay associated with it. With this assumption, the tracking error is expected to be in the neighborhood of zero as proved in theorem 1. However, when dealing with hydraulic systems, this assumption should be relaxed. Therefore, even if stability of the system could be achieved due to bounded closed-loop signals, the overall tracking performance is not good owing to the presence of flattery delay[18] in the output. One may be tempted to increase the feedback gain to get better performance, but mere feedback compensation results into actuator saturation; the steady-state deviation by time delay also remains.

Here, we have proposed to add one compensatory signal with the control input [Eq. (9)] so that it eliminates the effect of time delay caused by the hydraulic system. Then, we can assume that the delay compensated system is controlled by the robust adaptive fuzzy control law.

The delay compensation signal is designed from the principles of one-step-ahead control method.[21] Here we estimate the control input necessary at the next sampling ($u_{est}(k+1)$) step from the reference and the output at the present sampling instant k. If we design a control input that is proportional to this $u_{est}(k+1)$, it is very effective in eliminating the delay from the sampling instants $k = 1, 2, \ldots, \infty$. This is because we are driving the system that would be at the instant $(k+1)$ to the instant k.

Therefore, the hydraulic delay compensation control input can be written in the form

$$u_{hyd}(k) = \eta[r(k+1) - y(k)] \tag{15}$$

where $u_{hyd}(k)$ is the hydraulic delay compensation control input at the kth sampling instant, $r(k+1)$ is the reference at the $(k+1)$th instant, $y(k)$ is the output at the kth instant, and η is a design constant. The value of η is chosen by trial and error during real-time implementation.

The overall control input in the discrete form can be written as

$$u(k) = \theta^T(k)\xi(x(k)) + u_{hyd}(k). \tag{16}$$

4. Real-Time Implementation of the Robust Adaptive Fuzzy Control Algorithm in Embedded System

For the real-time implementation of the proposed adaptive controller, the first step is to choose a suitable membership function amongst the pool of known membership functions. In our case, we have chosen symmetric triangles with equal width and 50% overlap with neighboring triangles, as the membership functions, because it is the most natural and unbiased type of membership function.[7] Implementation of these types of membership functions is also easier, since fewer computations are involved and are suitable for embedded systems. The input and output membership functions for the feedback controller with normalized bases are shown in Fig. 7.

In order to implement the controller represented by Eq. (9), we considered that ξ is a function of error $e = y - y_m$ and rate of change of error $\dot{e}$. The parameters of the controller (θ) are chosen as the centre of each output membership functions. Thus, during adaptation, the center points of each output membership functions are changed following the robust adaptation law presented in Eq. (13). This effectively makes the output membership functions crowded in the same

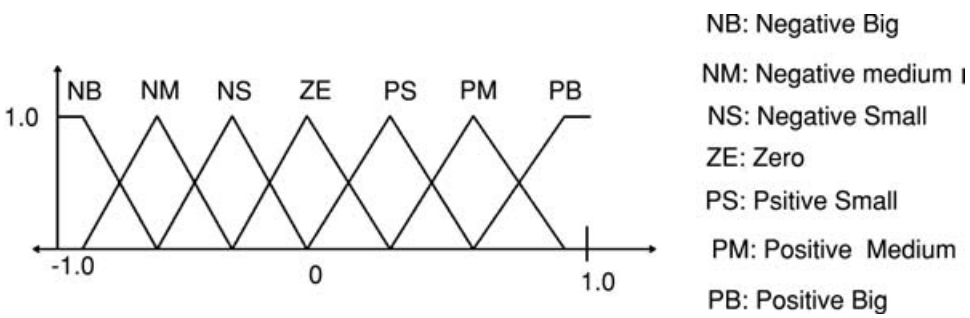


Fig. 7. Input–output membership functions with normalized base.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

e / Δe	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

Fig. 8. FAM table for PI-like fuzzy controller.

direction as the filtered error signal Pe, producing smaller control input if the error is small and larger control input if the error is larger. Thus the system can track the parameter variation at a much faster rate.

We have implemented the controller represented by Eq. (9) as a simple proportional integral (PI) like the fuzzy controller. The fuzzy rule base of such controller is represented as fuzzy associative memory (FAM) table and is given in Fig. 8.

For realistic estimation of the positive definite matrix P from Eq. (11) for the COMET-III leg joints, we determined an approximate transfer function models of shoulder, thigh, and shank joints by frequency response technique[18] as given in Eq. (17). From these transfer functions, we can calculate the value of the matrices A and B. Considering Q as a unity matrix, we calculated the value of P by using Matlab.

$$G(s) = \frac{\alpha\omega_n^2}{s^2 + 2\varsigma\omega_n s + \omega_n^2} \frac{1 - \frac{1}{2}sT}{1 + \frac{1}{2}sT} \tag{17}$$

where $\{\alpha, \varsigma, \omega_n, T\}_{\text{shoulder}} = \{1, 1.500, 2.000, 0.3\}$
$\{\alpha, \varsigma, \omega_n, T\}_{\text{thigh}} = \{1, 2.928, 4.477, 0.2\}$
$\{\alpha, \varsigma, \omega_n, T\}_{\text{shank}} = \{1, 3.028, 5.507, 0.2\}$

The C language-like pseudo code of the algorithm is given in Fig. 9.

```
Step-1: Obtain e and edot   // error and rate of change of error.

Step-2: Initialize the center[i], width[i] , i=1,…,7 , of the input membership functions

Step-3: Calculate mf1[i] and mf2[j]  for all i=1,…,7; j=1,…,7. //membership values for e and edot

Step-4: Obtain fuzzy rule base mat_fb[i][j] for the robust adaptive fuzzy feedback
        controller.

Step-5: Obtain gamma_adapt   // nominal value of rate of adaptation
        Obtain P={p1, p2}      // solution from Riccati like equation
        Obtain M0,  0          // for leakage terms
        Obtain neu             // hydraulic delay compensation coefficient


Step-6: Set zai_num[7][7]=0.0, and zai_den=0.0  // initialize the numerator and denominator of
                                                // the Fuzzy Basis function
Step-7: For i=0 to 6, For i=0 to 6,
        zai_num[i][j]=mf1[i]*mf2[j];            //Product inference
        zai_den +=zai_num[i][j];

        zai[i][j]=zai_num[i][j]/zai_den;        // Fuzzy Basis function

Step-8: e_bar=p1*e+p2*edot;  // e_bar is the filtered error signal

Step-9: For i=0 to 6, For i=0 to 6,
        theta_dot[i][j] +=gamma_adapt*e_bar*zai[i][j] -leakage;  // adaptation rule

Step-10: theta [i+1]=theta[i]+theta_dot[i+1] ;  // integration of the adaptive parameter of controller,
                                                // "theta" is actually the centre of the membership
                                                //functions of the output membership functions of
                                                //the feedback controller
Step-11:  For i=0 to 6, For i=0 to 6,
          del_u += theta[i]*zai[i][j];          // rate of change of control input

Step-12:  u[i+1]=u[i]+del_u[i+1]                //control input

Step-13:  uhyd=[i+1]=neu*(ref[i+2]-y[i+1])    // hydraulic delay compensation

Step-14:  u_total[i+1]=u[i+1]+uhyd[i+1]

Step-15: Go to step-1.
```

Fig. 9. Pseudo code of the control algorithm.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

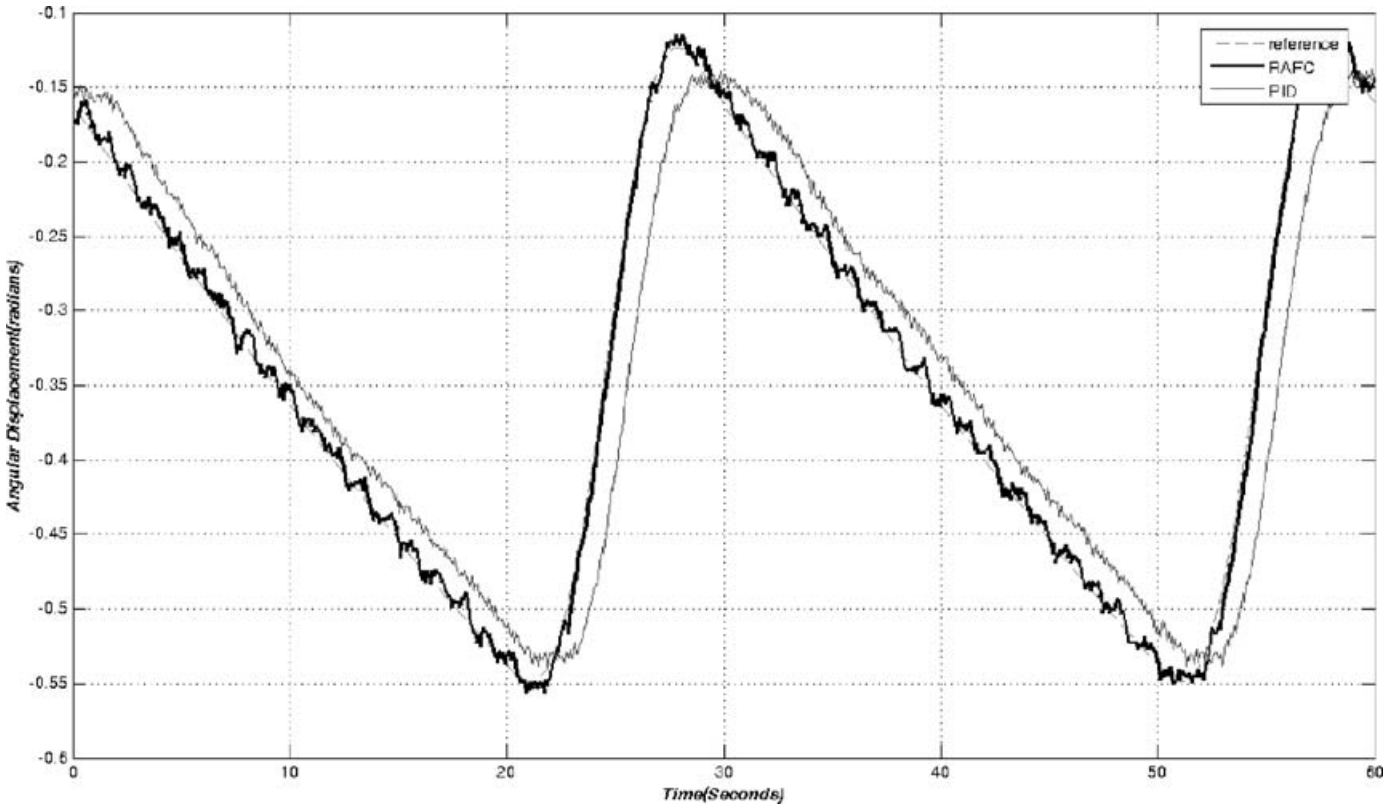


Fig. 10. Tracking performance of the shoulder joint.

The source code is written in C language and compiled in exeGCC, a GNU C compiler for SH series applications. The compiled object file .mot is downloaded to the SH4 FlashROM by the development system provided by M/s Kyoto Micro System for the embedded application with the Compact-PCI type SH4 microcomputers.

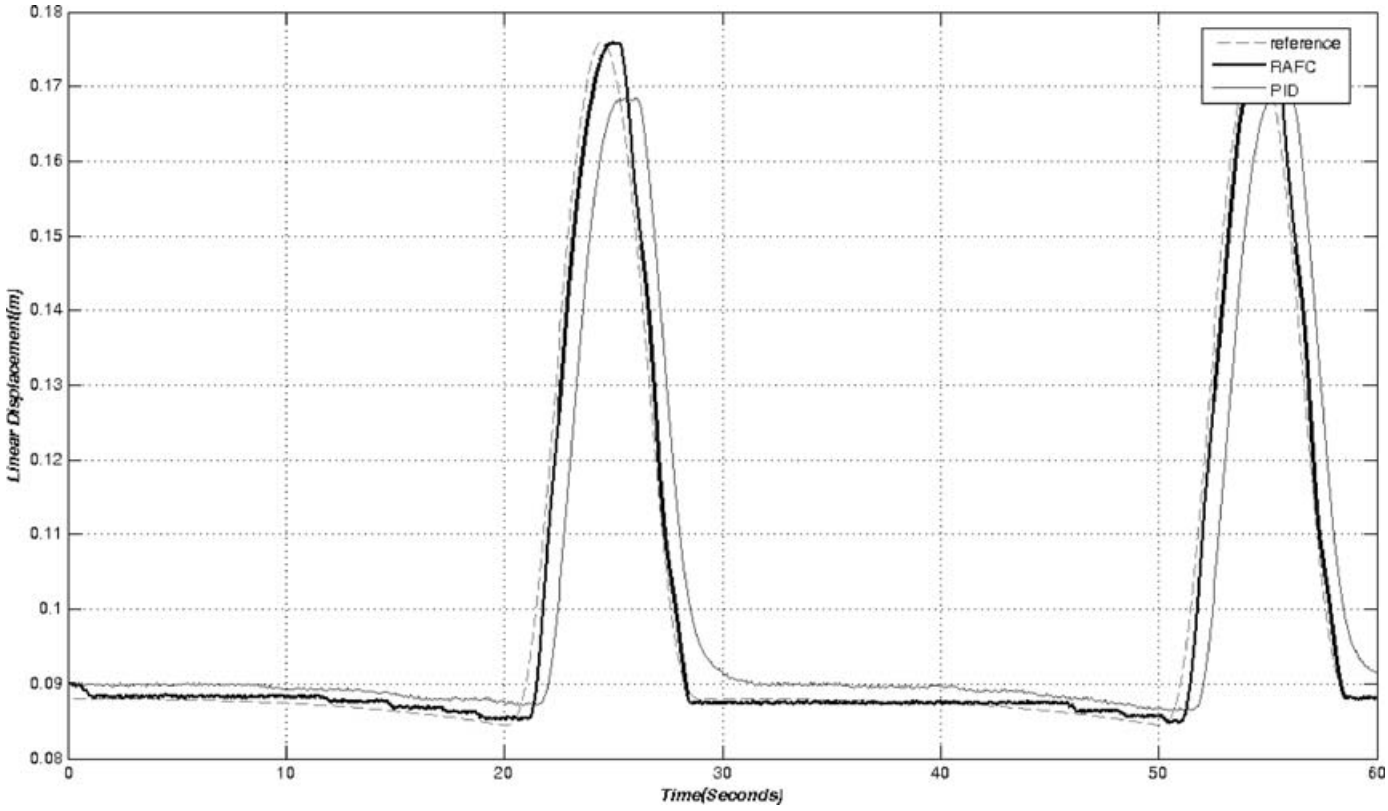


Fig. 11. Tracking performance of the thigh joint.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

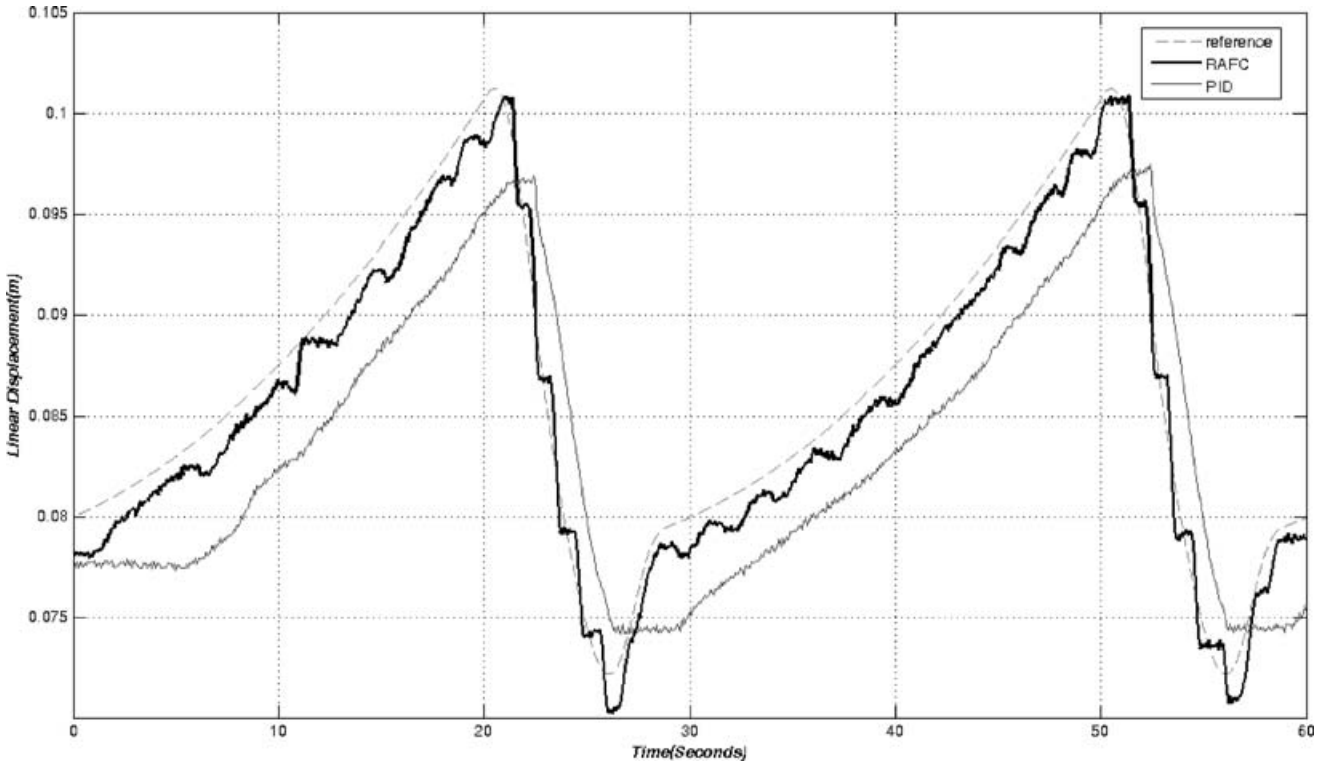


Fig. 12. Tracking performance of the shank.

5. Experimental Results

With the proposed control algorithm, walking experiments have been conducted on COMET-III. As already mentioned, tetrapod gait has been implemented for forward and backward walking of COMET-III. Here, we have assumed that the ground is slightly slippery, but does not have much obstacles and roughness. The actual experiment has been conducted over a very smooth but slippery vinyl floor. The experimental data showing the tracking performance of the reference joint trajectories, when COMET-III is walking, are shown in Figs. 10–12 for shoulder, thigh, and shank respectively. For comparison,

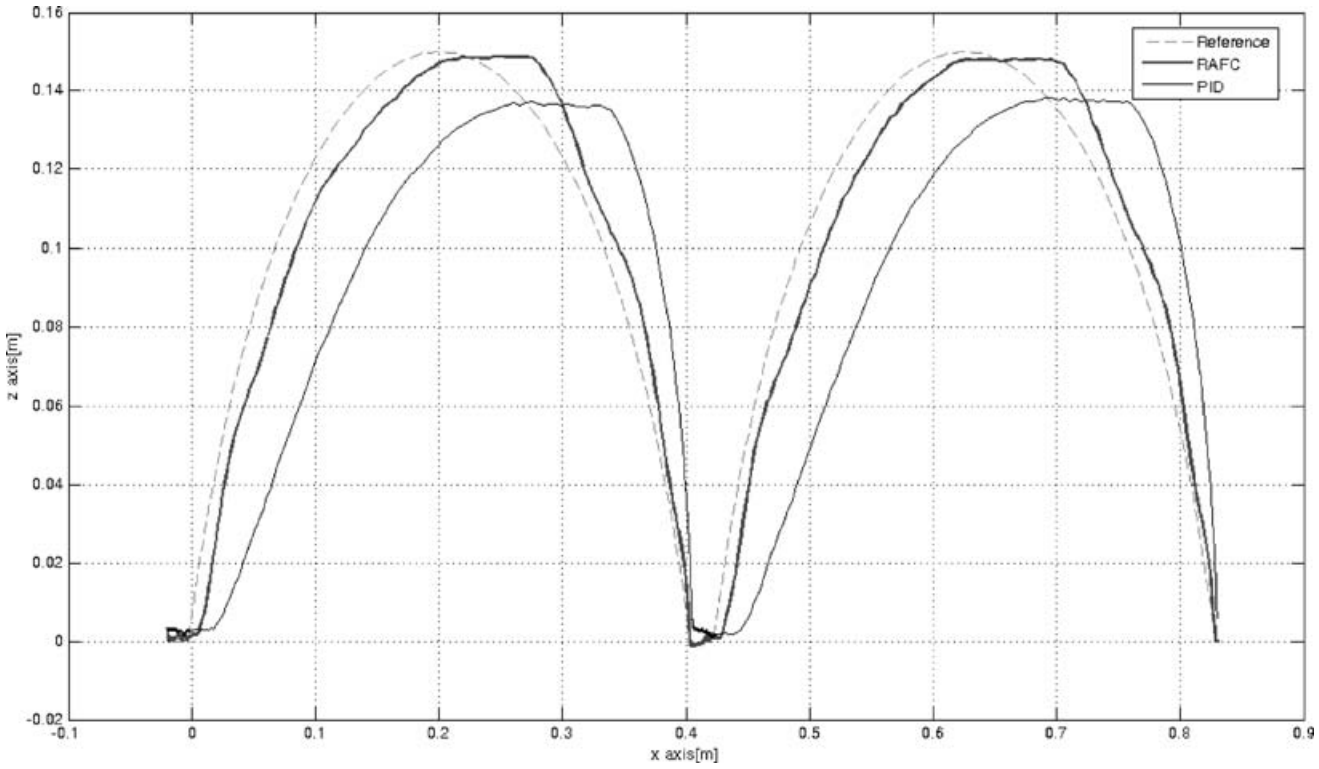


Fig. 13. The foot trajectory in the *X*–*Z* plane.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

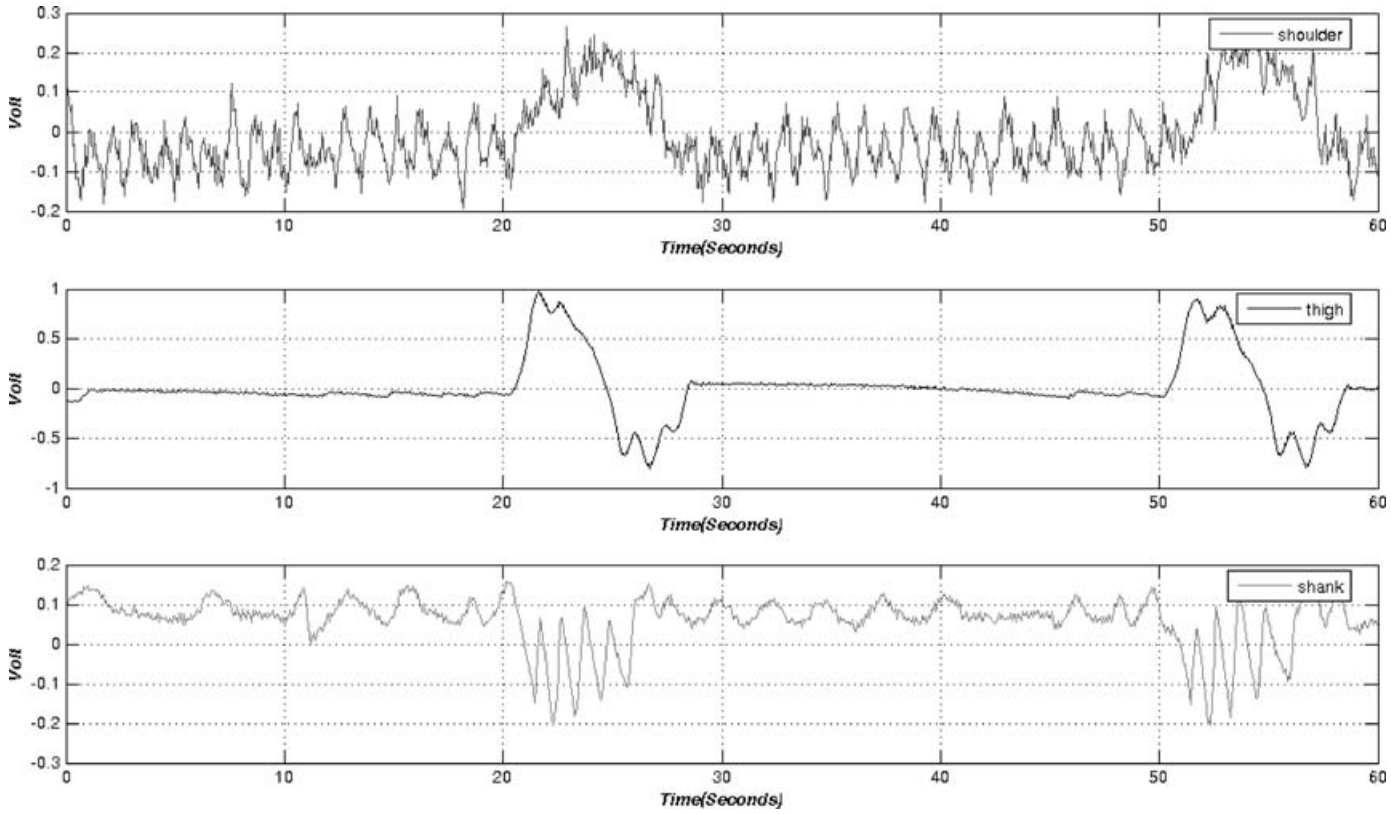


Fig. 14. Hydraulic delay compensatory one-step-ahead control input.

the data from PID control is also plotted on the same axis.

For successful implementation of any fuzzy controller, the most challenging part is to estimate the maximum swing of the input and output signals of the fuzzy controller. This is important because if the interval over which the fuzzy sets are defined does not match properly with the range of signal variation, the performance of the fuzzy controller

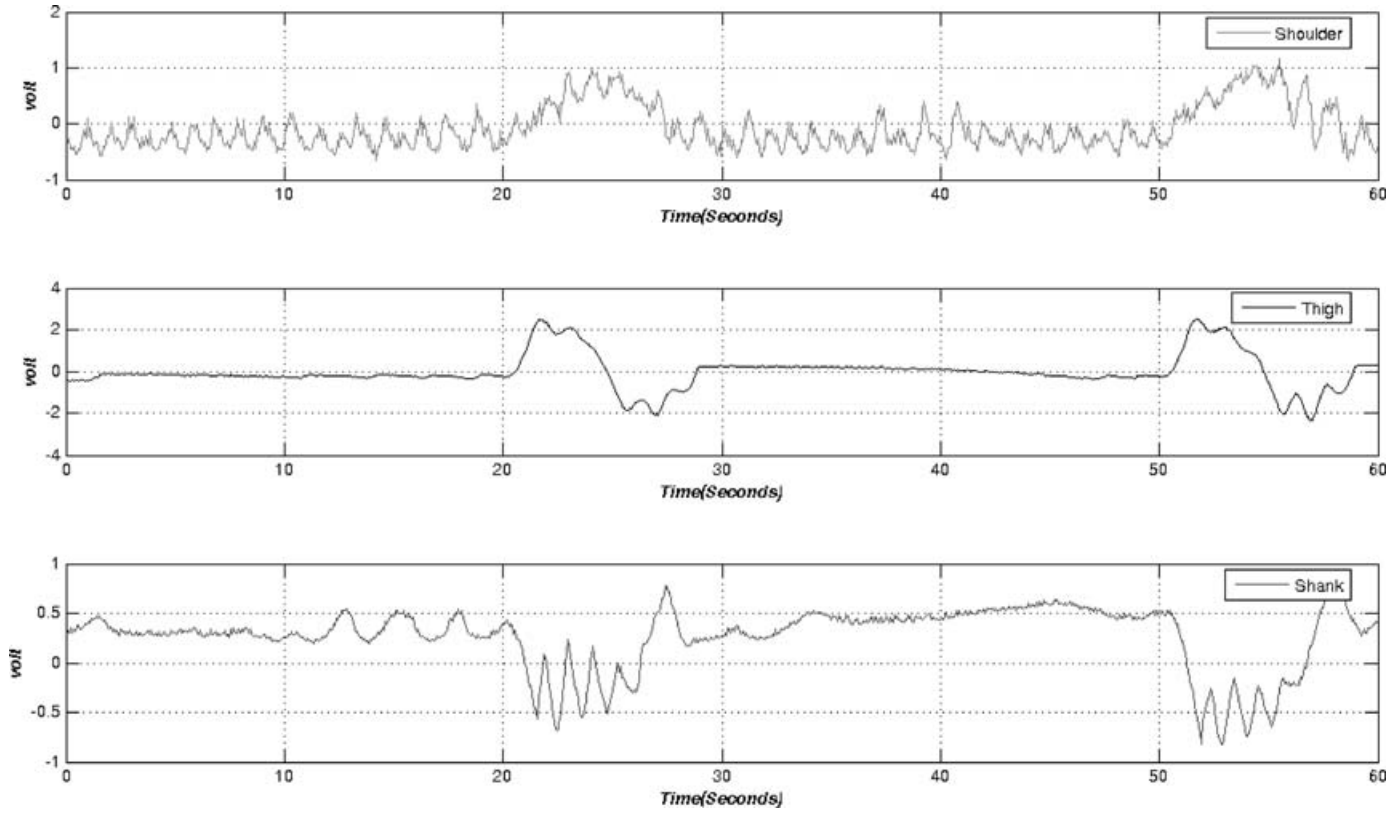


Fig. 15. Overall control input.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

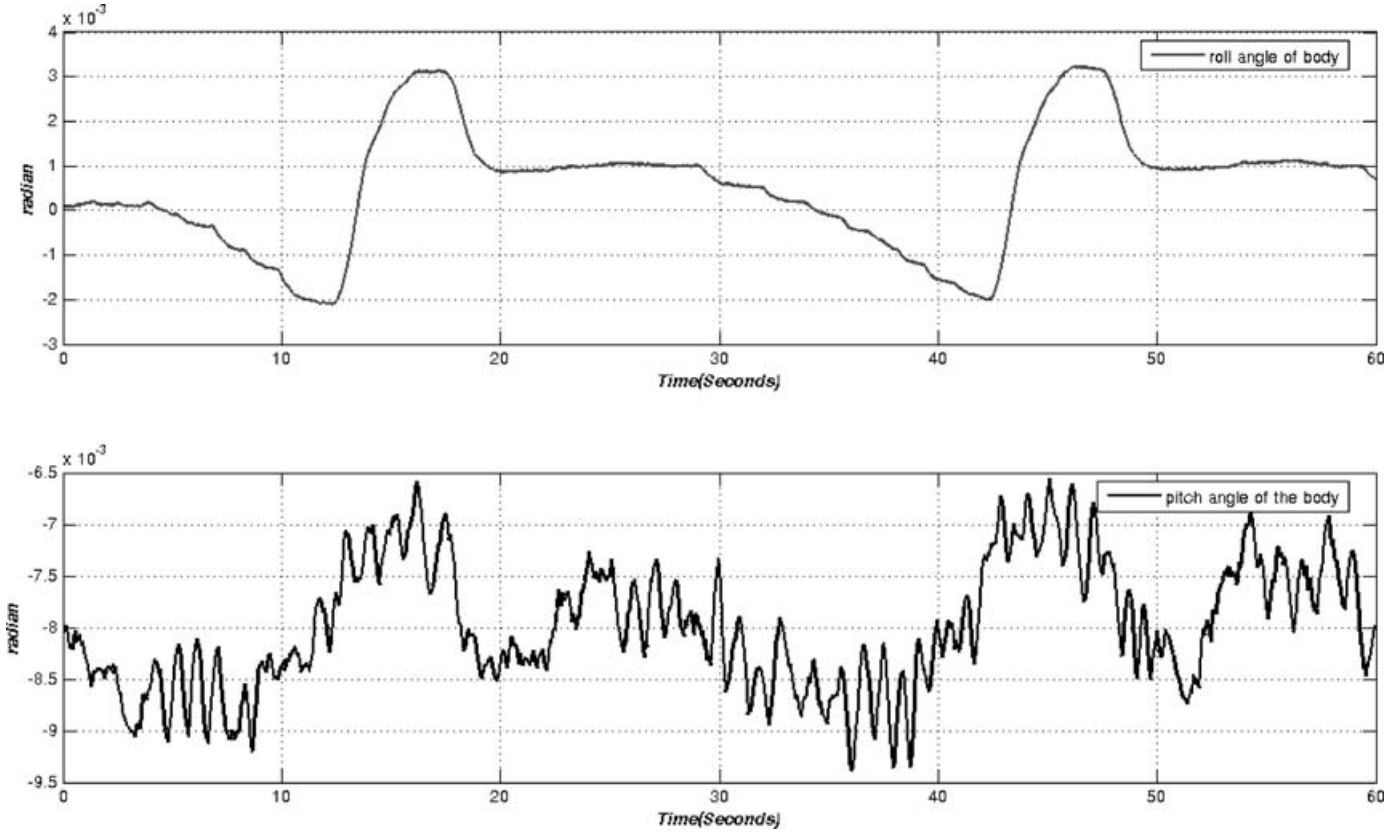


Fig. 16. Roll angle and pitch angle variation of body during walk.

will be ambiguous. While working with the state variables like error and rate of change of error, the situation is more complex. This is because during the tuning between the fuzzy set interval and the actual error signals swings, the error further decreases as they come closer. Thus, it needs many trials and errors while tuning the fuzzy set intervals with the actual signal swing. In order to facilitate this tuning process, we define the input fuzzy sets over a normalized interval of $(-1$ to $+1)$ and the input signals are multiplied by the scale-factors. As we have taken fuzzy singletons as the output fuzzy sets of the controller, we took a liberty of choosing the interval from $(-3$ to $+3)$ so that all the seven membership functions are unit distance apart. During the experiment, we have chosen the scale-factors for the input error signals of 14.5, 25, and 160 for shoulder, thigh and shank joint, respectively. The scale-factors for the input rate of change of error signal are 1, 2, and 10 for shoulder, thigh and shank joint, respectively. The output scaling factors are taken as 0.25, 0.475 and 0.14, respectively. These values are derived after several experimental trials.

The associated foot trajectory in Cartesian coordinate is shown in Fig. 13. From Figs. 11 to 13, it is evident that the proposed controller is quite effective for eliminating much of the flattering time delay caused by the hydraulic system. The control input generated for compensating the hydraulic delay is shown in Fig. 14. The total control input is shown in Fig. 15.

The roll and pitch angle variation of the robot body is shown in Fig. 16. We can see from this data that the locomotion is quite stable. The little undulation of the roll data is due to walking on the slippery floor and the small-magnitude high-frequency part is due to the noise generated by the vibration caused by the internal combustion engine in the vicinity of the attitude sensor.

6. Conclusion

In this investigation, we have proposed a hydraulic dead zone compensated robust adaptive fuzzy controller for a class of hydraulically actuated robotic mechanisms, containing flattery time delay caused by the dead zone associated with the electromagnetic proportional control valve. The proposed controller has been designed for the hydraulically actuated hexapod robot COMET-III and tested in real-time in the SH4 embedded system. The proposed algorithm runs very well in the embedded system with hard real-time constraints for a sampling time of 0.06 s. The experimental results exhibit that the robust adaptive fuzzy controller performs fairly good tracking of the joint trajectories compared to the conventional PID controller. The flattering delay imposed by the dead zone of the proportional electromagnetic valves and the time delay due to flow of oil has also been eliminated to a great extent. The proposed control scheme results in a very stable locomotion as shown by the body attitude response.

References

1. P. A. Ioannou, "Robust adaptive control: A unified approach," *Proc. IEEE* **79**(12), 1736–1767 (1991).
2. L. X. Wang, "Stable adaptive fuzzy control of nonlinear systems," *IEEE Trans. Fuzzy Syst.* **1**(2), 146–155 (1993).
3. B. S. Chen, C. H. Lee and Y. C. Chang, "H∞ tracking design of uncertain nonlinear SISO systems: Adaptive fuzzy approach," *IEEE Trans. Fuzzy Syst.* **4**(1), 32–43 (1996).

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press

4. J. H. Park, S. J. Seo and G. T. Park, "Robust adaptive fuzzy controller for nonlinear system using estimation of bounds for approximation errors," *Fuzzy Sets Syst.* **133**, 19–36 (2003).
5. F. Bu and B. Yao, "Nonlinear model based coordinated adaptive robust control of electro-hydraulic arms via overparameterizing method," *Proceedings of the IEEE International Conference on Robotics and Automation* (2001) pp. 3459–3464.
6. L. X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least-square learning," *IEEE Trans. Neural Netw.* **3**(5), 807–814 (1992).
7. R. K. Mudi and N. R. Pal, "A self-tuning fuzzy PI-controller," *Fuzzy Sets Syst.* **115**, 327–338 (2000).
8. K. Nonami, Q. Huang, D. Komizo, Y. Fukao, Y. Asai, Y. Shiraishi, M. Fujimoto and Y. Ikedo, "Development and control of mine detection robot COMET-II and COMET-III," *JSME Int. J. Ser. C* **46**(3), 881–890 (2003).
9. D. Driankov, H. Hellendoorn and M. Reinfrank, *An Introduction to Fuzzy Control* (Springer-Verlag, Berlin, Germany, 1996).
10. K. Fischle and D. Schröder, "An improved stable adaptive fuzzy control method," *IEEE Trans. Fuzzy Syst.* **7**(1), 27–40 (1999).
11. S. S. Ge and J. Wang, "Robust adaptive neural control for a class of perturbed strict feedback nonlinear systems," *IEEE Trans. Neural Netw.* **13**(6), 1409–1419 (2002).
12. Y. T. Kim and Z. Z. Bien, "Robust adaptive fuzzy control in the presence of external disturbance and approximation error," *Fuzzy Sets Syst.* **148**, 377–393 (2004).
13. T. Corbet, N. Sepehri and P. D. Lawrence, "Fuzzy control of a class of hydraulically actuated industrial robots," *IEEE Trans. Control Syst. Technol.* **4**(4), 419–426 (1996).
14. H. O. Wang and K. Tanaka, "An approach to fuzzy control of nonlinear systems: Stability and design issues," *IEEE Trans. Fuzzy Syst.* **4**(1), 14–23 (1996).
15. K. Tanaka and M. Sugeno, "Stability and design of fuzzy control systems," *Fuzzy Sets Syst.* **45**, 135–156 (1992).
16. M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control* (Wiley, New York, NY, 1989).
17. P. A. Ioannou and J. Sung, *Robust Adaptive Control* (Prentice-Hall, Upper Saddle River, NJ, 1996).
18. K. Nonami and Y. Ikedo, "Walking control of COMET-III using discrete time preview sliding mode control," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2004) pp. 3219–3225.
19. T. Corbet, N. Sepehri and P. D. Lawrence, "Fuzzy control of a class of hydraulically actuated industrial robot," *IEEE Trans. Control Syst. Technol.* **4**(4), 419–426 (1996).
20. X. S. Wang, C. Y. Su and H. Hong, "Robust adaptive control of a class of nonlinear systems with unknown dead-zone," *Proceedings of the 40th IEEE Conference on Decision and Control* (2001) pp. 1627–1632.
21. C. R. Johnson, "Adaptive implementation of one-step-ahead optimal control via input matching," *IEEE Trans. Autom. Control* **AC-23**, 865–872 (1978).
22. J. Slotine and W. Li, *Applied Nonlinear Control* (Prentice-Hall, Englewood Cliffs, NJ, 1991).

Appendix

Proof of Theorem 1:

The stability analysis method presented in ref. 15 focuses mainly on systems having a Sugeno type fuzzy model, although it gives a greater insight into the theoretical basis of analyzing the stability of the fuzzy system. However, obtaining a Sugeno-type fuzzy model is not always so easy in practical situations. Therefore, we adopted a Mamdani-type fuzzy model in our paper.

The vector $\xi(x)$ in Eq. (6) contains the normalized firing strengths of the fuzzy rules, which depend on the controller inputs. Since firing strength lies in the interval (0,1), it can be guaranteed that $\xi(x)$ is bounded.[10]

We choose

$$V = \frac{1}{2}\left(e^T P e + K\frac{1}{\gamma}\phi^T\phi\right) \tag{18}$$

as the Lyapunov function, where P is the solution of the Eq. (11) for some positive definite matrix Q.

Derivative of V with respect to time yields:

$$\begin{aligned}\dot{V} = {} & \frac{1}{2}\left(\dot{e}^T P e + e^T P\dot{e} + K\frac{1}{\gamma}\dot{\phi}^T\phi + K\frac{1}{\gamma}\phi^T\dot{\phi}\right) \\ & \frac{1}{2}\Bigg(e^T A^T P e + k\xi^T\phi B^T P e + e^T P A e + K e^T P B\phi^T\xi \\ & + K\frac{1}{\gamma}\dot{\phi}^T\phi + K\frac{1}{\gamma}\phi^T\dot{\phi}\Bigg) - \frac{1}{2}e^T Q e + K e^T P B\phi^T\xi \\ & + K\frac{1}{\gamma}\phi^T\dot{\phi}.\end{aligned}$$

If the adaptive law [Eq. (10)] is used, we have

$$\dot{\phi} = \dot{\theta} = -\gamma\xi(x)B^T P e$$

and, hence

$$\begin{aligned}\dot{V} = {} & -\frac{1}{2}e^T Q e + K e^T P B\phi^T\xi(x) - K\frac{1}{\gamma}\phi^T(\gamma\xi(x)B^T P e) \\ & - \frac{1}{2}e^T Q e.\end{aligned}$$

Since Q is a positive definite, $\dot{V} < 0$, this proves the global stability of the proposed algorithm.

To prove the convergence of the tracking error, we need the following Lemma,[22] which is derived from Barbalat's lemma.

Lemma 1: If a scalar function $\tilde{V}(t)$ satisfies the following conditions:

(i) $\tilde{V}(t)$ is lower bounded;
(ii) $\dot{\tilde{V}}(t)$ is negative semi-definitive;
(iii) $\dot{\tilde{V}}(t)$ is uniformly continuous in time;

then $\lim_{t\to\infty}\dot{\tilde{V}}(t) = 0$.

Let us choose $\tilde{V} = V - \int_0^t (K e^T P B\phi^T\xi + K\frac{1}{\gamma}\phi^T\dot{\phi})\,d\tau$, this function is lower bounded at 0. We obtain

$$\dot{\tilde{V}} = -\frac{1}{2}e^T Q e$$

$$\ddot{\tilde{V}} = -\frac{1}{2}(e^T A^T Q e + e^T Q A e) - e^T Q B K\phi^T\xi(x).$$

From this, it is evident that $\dot{\tilde{V}}(t)$ is negative semi-definite. If $\xi(x)$ is bounded, $\ddot{\tilde{V}}(t)$ is also bounded and hence, $\dot{\tilde{V}}(t)$ is uniformly continuous in time. Thus, it can be concluded that if $\xi(x)$ is bounded, $\lim_{t\to\infty}\dot{\tilde{V}}(t) = 0$ and $\lim_{t\to\infty} e(t) = 0$.

https://doi.org/10.1017/S0263574706003067 Published online by Cambridge University Press