Indonesian Journal of Electrical Engineering and Informatics (IJEEI)
Vol. 6, No. 4, December 2018, pp. 458~470
ISSN: 2089-3272, DOI: 10.11591/ijeei.v6i1.738

Adaptive Fuzzy Control of Puma Robot Manipulator in Task Space with Unknown Dynamic and Uncertain Kinematic

Azita Azarfar
Department of Electrical and Computer Engineering, Islamic Azad University, Iran

Article Info

Article history:

Received Sep 29, 2018
Revised Nov 11, 2018
Accepted Dec 12, 2018

Keyword:

Adaptive fuzzy control
PUMA 560
Robot manipulator
Robot task space
Uncertain Jacobian matrix

ABSTRACT

A In this paper, an adaptive direct fuzzy control system is presented to control the robot manipulator in task space. It is assumed that robot system has unknown dynamic and uncertain kinematic. The control system and adaption mechanism are firstly designed for joint space tracking. Then by using inverse Jacobian strategy, it is generalized for task space. After that, to overcome the problem of Jacobian matrix uncertainty, an improved adaptive control system is designed. All the design steps are illustrated by simulations.

Copyright © 2018 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Azita Azarfar,
Department of Electrical and Computer Engineering,
Islamic Azad University, Teheran, Iran.
Email: azita.azarfar@gmail.com

1. INTRODUCTION

Most robot applications are in task space or Cartesian space. While a lot of research has been done on controlling the robot manipulators in task space [1]-[3], most of them require exact knowledge of the system. Robot manipulator systems are highly nonlinear, and the system dynamics change when robot picks up different tools. The robot kinematics and Jacobian matrix are hence uncertain, and the transferring information from joint space to Cartesian space is not completely correct. In control systems theory, to deal with such problems, adaptive control methods are usually proposed and used in all fields [4]-[7]. In robotics system control [6], [8]-[10] proposed adaptive sliding mode control to overcome the uncertainties in dynamics and kinematics. The sliding vector is the joint space error and the adaption law is usually extracted from Lyapunov functions to guarantee the stability. In [11], an adaptive back stepping method is presented to control the position of robot manipulator based on support vector machine; the main controller is a state feedback control system and the stability is proved by Lyapunov theory. Adaptive impedance control strategies are also presented for robotic systems [12], [13]. In [13], the system uncertainties are approximated by a radial basis function neural network. In [12], a neural network is designed to approximate the unknown dynamics and drive the robot without velocity sensor. In the last decade, using of intelligent methods such as neural networks, fuzzy systems, and intelligent optimization algorithms in robotic systems and other control systems has increased. These methods may be used as the main control system, or they may approximate a part of dynamics or uncertainties. Neural networks and fuzzy systems can approximate any arbitrary nonlinear function, so they have frequently been used in many robot control systems, as a system model or control strategy. In [14], an adaptive neural network control system is designed based on unified objective bound to decrease the effect of initialization error on transient response of the robot manipulator. A neural network control system is presented in [15] to deal with input dead zone and output constraints. The input dead zone and the robot model are approximated by a neural network, and the output constraint is considered

in Lyapunov function. Ref. [16] proposed an ANFIS-PD+I based hybrid force/position controller for unspecified robot dynamics in the presence of external disturbances. Fuzzy logic control (FLC) systems are widely used in robot control systems. The robustness of the control system is usually improved by using fuzzy control systems instead of classical control strategies, and the experts and operators' experiences can be added to the control system. Different fuzzy control methods, such as Simple FLC system [17], fuzzy PID control systems [18], fuzzy sliding mode control systems [19] and adaptive fuzzy systems [20] have been presented. A review on controlling a two-link robot manipulator by fuzzy techniques has been recently done in [21].

In our paper, an adaptive direct fuzzy control system is designed to control a robot manipulator in task space with unknown dynamic and uncertain kinematics. The control method is firstly designed for joint space control of the robot manipulator based on the direct adaptive fuzzy control approach presented in [22]. After that, the proposed method is generalized to task space control system by using inverse Jacobian methodology. The results are accepted when the Jacobian matrix is known, and it does not have acceptable results when there is uncertainty in Jacobian matrix. To deal with this problem, a new update law is proposed in the next step. The new presented update law guarantees the acceptable performance of robot manipulator systems with unknown dynamics and uncertain Jacobian matrix. The stability is proved by Lyapunov method. Four simulations have been done on an industrial PUMA robot to illustrate all the design steps and control system effectiveness.

The paper is structured as follows. Section 2 introduces the three-link industrial robot PUMA. Direct adaptive fuzzy control system for joint space control is designed in Section 3. A simulation on Puma robot in joint space in existence of disturbance is also provided in Section 3. Extending the control system to control the robot end effector has been discussed in Section 4. The new adaption law and improved control system have been presented in Section 5. Section 6 concludes the paper.

2. PUMA ROBOT MODEL

The PUMA 560 (Programmable Universal Manipulation Arm) is an industrial robot arm which has six degrees of freedom resulted from three rotational joints. It owns three joints for arm and three joints for wrist. Since the purpose of this paper is just to control the position of robot end point in task space, the robot arm joints are only considered in robot model. The wrist joints are responsible for end point final angle A PUMA schematic is shown in Figure 1. The three link robot dynamic equations can be written as [23]:

$$M(q)\ddot{q} + C(q,\dot{q}) + g(q) = Q \tag{1}$$

where $q \in R^3$ are the joint coordinates, $M(q) \in R^{3\times3}$ is the symmetric positive-definite inertia matrix, $C(q,\dot{q})$ represents the coriolis and centripetal terms, g(q) represents the gravity effects and Q is a control input vector in R^3 which represents the torques applied to three joints. The position of robot end –effector can be calculated from joint coordinates through robot kinematic equation which can be expressed as:

$$X = h(q) \tag{2}$$

where X is the position vector of robot end-effector in task space and $h(q)$ is a nonlinear function extracted from kinematic equations. The velocity of end-effector is related to joint velocity through Jacobian matrix $J(q) \in R^{3\times3}$ as

$$\dot{x} = J(q)\dot{q} \tag{3}$$

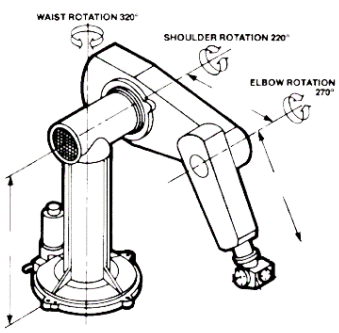

Figure 1. A schematic of PUMA 560 arm [24]

3. JOINT SPACE CONTROL SYSTEM

The control objective is to design a feedback controller $u = u(Z|\theta)$ based on fuzzy systems and an adaptation law for adjusting the parameter vector θ, such that the robot joint angles q follows the ideal output q_m, as close as possible. The fuzzy controller is designed with nine fuzzy rules in the form of:

$$L_j : if\ z_1\ is\ A_1^j\ and\ z_2\ is\ A_2^j\ THEN\ y = S^{L_j}$$

where L_j denotes the *j-th* fuzzy rule defined in Table 1. The inputs are $z_1 = q_d - q$ and $z_2 = \dot{q}_d - \dot{q}$ and $z = [z_1, z_2]^T$. A_i^j are the fuzzy input sets which are shown in Figure 2 and it is the same for both of inputs. y is the output, for $j = 1,2,\ldots,9$ and $i = 1,2$. Specifically, using the product inference engine, singleton fuzzifier and center average defuzzifier, the output $u(z) \in V \subset R$ of the fuzzy system is in form of

$$u(Z|\theta) = \frac{\sum_{j=1}^{9}(\bar{y}_u^{l_j})(\prod_{i=1}^{n}\mu_{A_i^j}(z_i))}{\sum_{j=1}^{9}(\prod_{i=1}^{2}\mu_{A_i^j}(z_i))} \tag{4}$$

where $\bar{y}_u^{l_j}$ is the center of output membership function for the j-*th* rule. Let $\bar{y}_u^{l_j}$ be the free parameters that are collected into a vector of $\theta \in R^9$. The output membership functions of the proposed fuzzy system with initialized center (θ_0) are shown in Figure 3. The objective is to update vector θ to improve the controller performance. Hence, the center of output membership function is changed adaptively to overcome the uncertainty and unknown dynamics. We can rewrite the $u(Z|\theta)$ in form of $u(Z|\theta) = \theta^T\varepsilon(Z)$ where $\varepsilon(Z)$ is a nine-dimensional vector in form of

$$\varepsilon_k(Z) = \frac{[\prod_{i=1}^{n}\mu_{A_i}^{l_k(z_i)}]}{\sum_{l_j=1}^{9}[\prod_{i=1}^{n}\mu_{A_i}^{l_j(z_i)}]} \qquad k=1, 2, \ldots, 9 \tag{5}$$

and parameters in vector θ will be changed during online operation. The next task is to design an adaptation law for θ, such that the tracking error e is minimized.

Table 1- Fuzzy control system rules

Error \ Chang of error	Negative	Zero	Positive
Positive	PM	PH	PVH
zero	NS	Zero	PS
negative	NVH	NH	NM

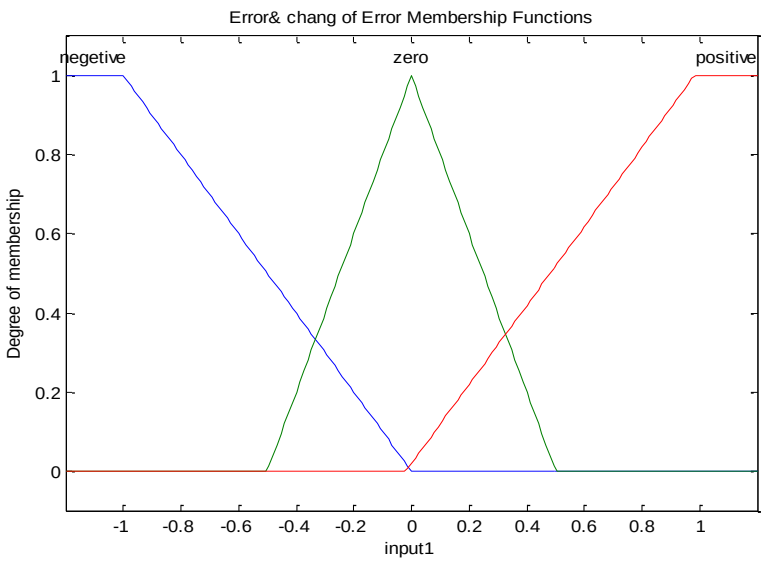

Figure 2. Main fuzzy control system input membership function

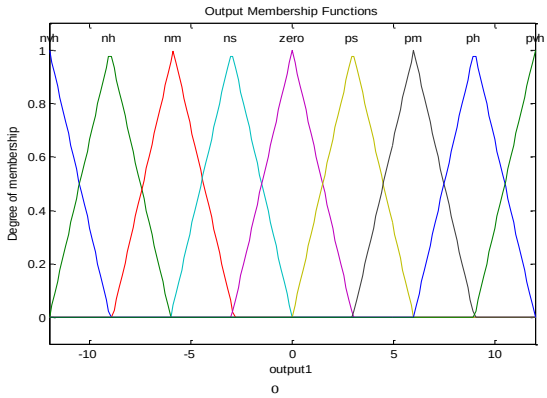

Figure 3. Main fuzzy control system output membership function with initial center

Adaption law:

The robot system can be rewritten as an unknown function in form of below

$$\ddot{q} = N(\mathrm{q}, \dot{\mathrm{q}}) + bu \tag{6}$$

according to linearization feedback theory, if we choose

$$u^* = \frac{1}{\mathrm{b}}[-N(\mathrm{q}, \dot{\mathrm{q}}) + \ddot{q_m} + \mathbf{K}^{\mathrm{T}}\mathbf{e}] \tag{7}$$

where $e = q_m - q$, $\mathbf{e} = (e, \dot{e})^T$, $\mathbf{K} = (k_1, k_2)^T$, then we obtain that closed-loop system stability is governed by

$$\ddot{\mathrm{e}} + \mathrm{k}_1\dot{\mathrm{e}} + \mathrm{k}_2\mathrm{e} = 0 \tag{8}$$

Since $N(\mathrm{q}, \dot{\mathrm{q}})$ are unknown, the ideal controller above cannot be implemented. Then the control objective changes to estimating u^* by an adaptive fuzzy controller $\mathrm{u}(Z|\theta)$. Substituting $u = u(Z|\theta)$ into (6) and by rearrangement, we obtain

$$\mathrm{e}^{\mathrm{n}} = -\mathbf{K}^{\mathrm{T}}\mathbf{e} + \mathrm{b}[\mathrm{u}^* - \mathrm{u}(Z|\theta)] \tag{9}$$

assume that

$$\Lambda = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \\ -k_n & -k_{n-1} & \dots & -k_1 \end{bmatrix} \mathbf{b} = \begin{bmatrix} 0 \\ \dots \\ 0 \\ \mathrm{b} \end{bmatrix} \tag{10}$$

the closed-loop dynamics can be written into the vector form

$$\dot{\boldsymbol{e}} = \Lambda\boldsymbol{e} + \boldsymbol{b}[\mathrm{u}^* - \mathrm{u}(Z|\theta)] \tag{11}$$

define the optimal parameters

$$\theta^* = \arg\min_{\theta\in\mathrm{R}^9}[\sup_{z\in\mathrm{R}^2}|\mathrm{u}(\mathrm{Z}|\theta) - \mathrm{u}^*|] \tag{12}$$

and the minimum approximation error

$$\omega = \mathrm{u}(\mathrm{Z}|\theta^*) - \mathrm{u}^* \tag{13}$$

we can rewrite the error (11) as

$$\dot{\boldsymbol{e}} = \Lambda\boldsymbol{e} + b(\theta^* - \theta)^T\varepsilon(Z) - \boldsymbol{b}\omega \tag{14}$$

Consider the Lyapunov function candidate

$$V = \frac{1}{2}\boldsymbol{e}^T P\boldsymbol{e} + \frac{\boldsymbol{b}}{2\gamma}(\theta^* - \theta)^T(\theta^* - \theta) \tag{15}$$

where P is a positive definite matrix satisfying the Lyapunov equation $\Lambda^{\mathrm{T}}\mathrm{P} + \mathrm{P}\Lambda = -\mathrm{Q}$, (Q is a an arbitrary *positive* definite matrix) and γ is a positive constant.

The time derivative of V along the closed-loop system trajectory is

$$\dot{V} = -\frac{1}{2}\mathbf{e}^{\mathrm{T}}\mathrm{Q}\mathbf{e} + \mathbf{e}^{\mathrm{T}}P\boldsymbol{b}[(\theta^{*} - \theta)^{T}\varepsilon(Z) - \omega] - \frac{b}{\gamma}(\theta^{*} - \theta)^{T}\dot{\theta} \tag{16}$$

Let p_n be the last column of P, then from $\boldsymbol{b}$ = (0, ..., 0, b) we have $\mathbf{e}^{\mathrm{T}}P\boldsymbol{b} = \mathbf{e}^{\mathrm{T}}p_n b$. So, we can rewrite $\dot{V}$ as

$$\dot{V} = -\frac{1}{2}\mathbf{e}^{\mathrm{T}}\mathrm{Q}\mathbf{e} + \frac{b}{\gamma}(\theta^{*} - \theta)^{T}\left[\gamma\mathbf{e}^{\mathrm{T}}p_n\varepsilon(Z) - \dot{\theta}\right] - \mathbf{e}^{\mathrm{T}}p_n b\omega \tag{17}$$

If we choose the adaptation law as

$$\dot{\theta} = \gamma\mathbf{e}^{\mathrm{T}}p_n\varepsilon(Z) \tag{18}$$

then

$$\dot{V} = -\frac{1}{2}\mathbf{e}^{\mathrm{T}}\mathrm{Q}\mathbf{e} - \mathbf{e}^{\mathrm{T}}p_n b\omega \tag{19}$$

Since $Q > 0$ and ω is the minimum approximation error, it is expected that by designing the fuzzy system $\mathrm{u}(\mathrm{Z}|\theta)$ with a sufficiently large number of rules, the ω would be small enough such that $|\mathbf{e}^{\mathrm{T}}p_n b\omega| < \frac{1}{2}\mathbf{e}^{\mathrm{T}}\mathrm{Q}\mathbf{e}$, which results in $\dot{V} < 0$.

So, the adaption law is extracted from Lyapunov function to guarantee the stability. The joint space control structure is displayed in Figure 4. The parameter γ is the convergence rate. By increasing γ, the error usually decreases faster, but if it is set as a large number, small change in error results in a large change in adjustable parameters which is not acceptable.

3.1. Joint space control simulation (simulation A)

To illustrate the effectiveness of the proposed controller in joint space, a simulation on PUMA 560 robot has been done in Matlab software. A desired trajectory shown in Figure 5 is considered for joint coordinates. The vector θ is the center of fuzzy output membership functions. The goal of adaptive controller is to update the vector θ according to (18) to improve the control performance. The initial center of fuzzy output membership functions is set to

$$\theta_0 = [-12\ \ -9\ \ -6\ \ -3\ 0\ 3\ 6\ 9\ 12] \tag{20}$$

To illustrate the effect of adaption law, one of controllers which is applied to joint 2 is non-adaptive control system. The center of fuzzy output membership function related to joint 2 controller is fixed and it is same as θ_0. The other two controllers applied to joint1 and joint 3 will be adapted by (18). A disturbance pulse is also added to output joint coordinates which is started in second 1, and it will be ended in second 2. The tracking errors of joints are displayed in Figure 6. It is clear that joints 1 and 3, in which the controllers are adapted by adaption law, have smaller errors and better performance in presence of disturbance. The center of "zero" membership functions related to fuzzy outputs is shown in Figure 7. This figure illustrates how the membership functions change adaptively to compensate the system error and to overcome the disturbance. From all the results, the proposed controller has acceptable performance in presence of unknown dynamics and disturbance.

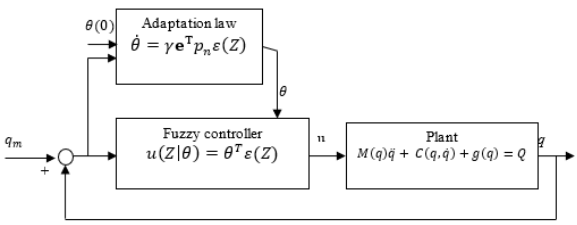

Figure 4. Proposed joint space control structure

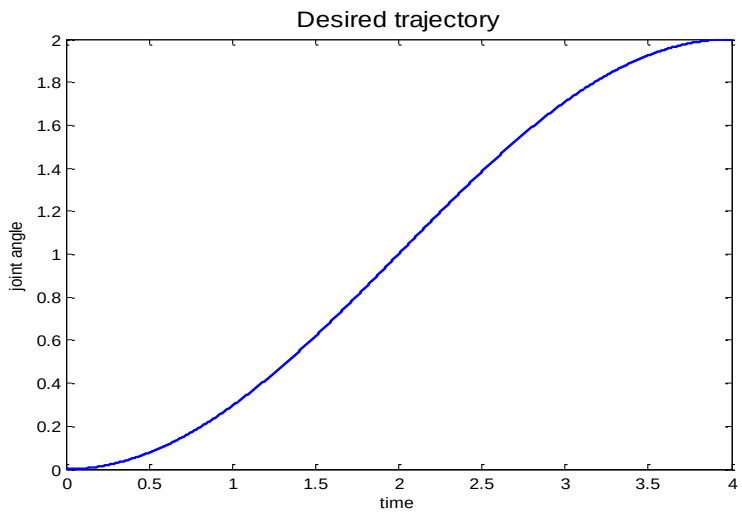

Figure 5. Desired trajectory for joints angle in simulation A

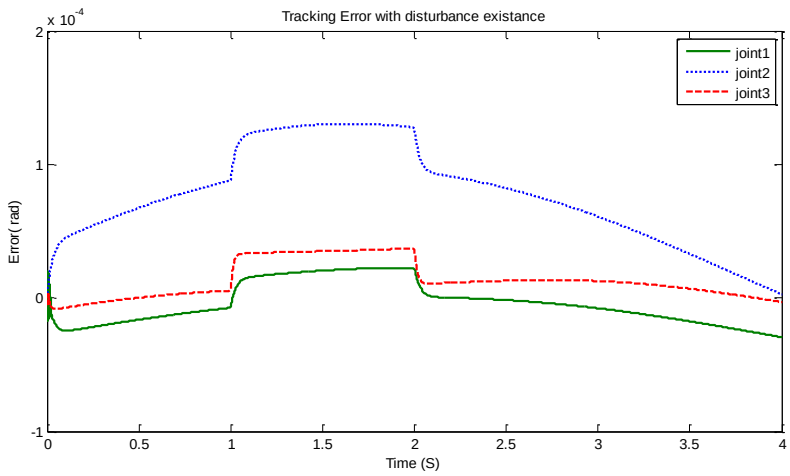

Figure 6. Robot joints tracking error in simulation A. Joint 2 has non-adaptive control system and a disturbance pulse is also added between seconds 1 and 2.

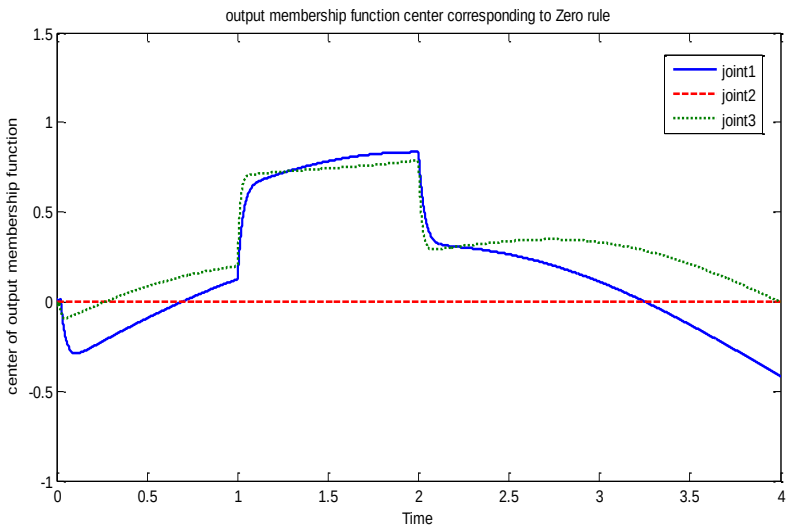

Figure 7. Center of fuzzy output “zero” membership functions in simulation A.

4. TASK SPACE CONTROL SYSTEM

In joint space control strategies, it is assumed that a desired trajectory is specified for joint variables such as joint position or velocity. But the robot motion is usually defined in task space or Cartesian space. The control methods in which the error is defined based on robot end-effector position and velocity in Cartesian space are named task space control schemes. There are some different control systems in this field, but one of the most known methods is inverse Jacobian strategy displayed in Figure 8. Cartesian error is mapped to joint space error through inverse Jacobian matrix based on (3). The main problem that limits the application of this method is uncertainties in robot kinematics and Jacobian matrix. Therefore, the control system would not have successful performance if the Jacobian matrix is not exact. The proposed method in the last section is used for task space control by using inverse Jacobian strategy. Two different simulations have been done with exact and non-exact Jacobian matrix. When the Jacobian matrix is exact, the control system has acceptable performance in task space although the robot dynamics are unknown. On the other hand, the proposed control system does not work perfectly when Jacobian matrix is uncertain. The proposed control structure in task space is displayed in Figure 9.

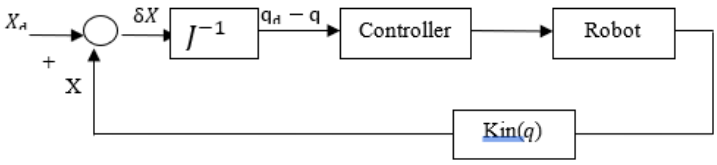

Figure 8. Inverse Jacobian strategy

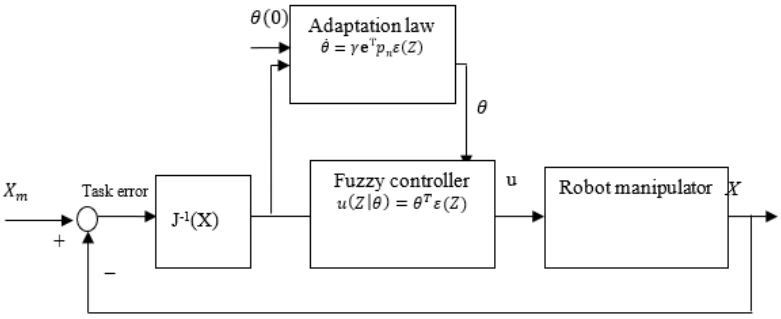

Figure 9. Task space adaptive direct fuzzy control structure

The presented control system is applied to PUMA robot to track a circle in Cartesian space with following equations:

$$\begin{cases} x = 0.203\sin(\pi t) \\ y = 0.203\cos(\pi t) \\ z = 0.4013 \end{cases} \tag{21}$$

simulations will be explained in following subsections.

4.1. Task space control system with exact Jacobian matrix simulation (simulation B)

In this simulation, it is considered that Jacobian matrix is exact and there is no kinematic uncertainty in robot manipulator. The error is calculated in Cartesian space, and it is transmitted to joint space correctly by using exact inverse of Jacobian matrix. The robot end effector position tracking the desired path is shown in Figure 10. Tracking is performed completely. The Cartesian error is displayed in Figure 11. The voltage of the motors which move the joints are shown in Figure 12. The voltages are limited and acceptable. From all the simulation results, that proposed control system has acceptable performance using exact Jacobian matrix.

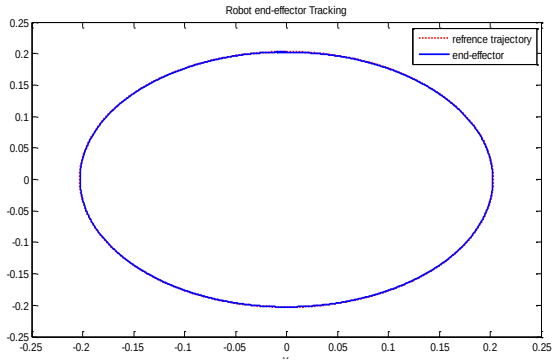

Figure 10. Robot end effector position tracking desired trajectory in simulation B.

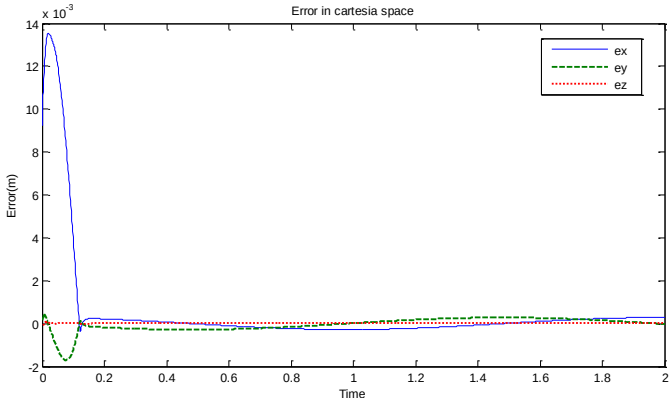

Figure 11. Cartesian space error in simulation B.

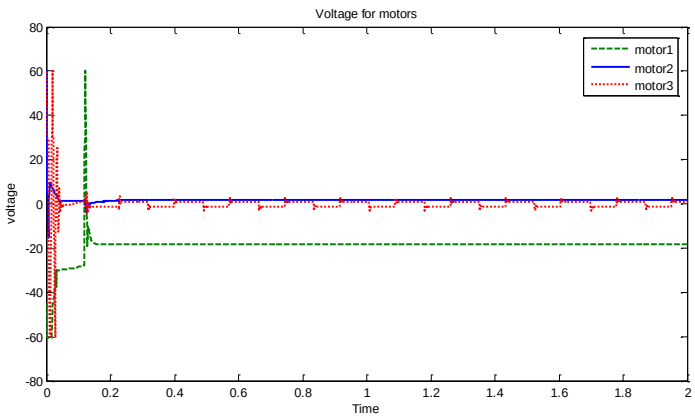

Figure 12- Input voltage of actuators which apply the torque to robot joints in simulation B

4.2. Task space control simulation with uncertain Jacobian matrix (simulation C)

To investigate the control performance using uncertain Jacobian matrix, 50 percent uncertainty is considered for kinematic parameters (Denavit Hartenberg parameters). The Jacobian matrix is hence inexact and imperfect error is transmitted to joint space. The control system is applied to PUMA robot to track a circle the same as the last subsection. Tracking error in Cartesian space is displayed in Figure 13, and the motor voltages are shown in Figure 14. Tracking error increases in comparison with simulation B. The motor voltages are increased in unstable manner to compensate the error. Therefore, the system is out of control. It can be found that the proposed control does not have acceptable performance if there exists uncertainty in Jacobian matrix. To overcome this problem, an improved adaptive fuzzy task space control is presented in the next section.

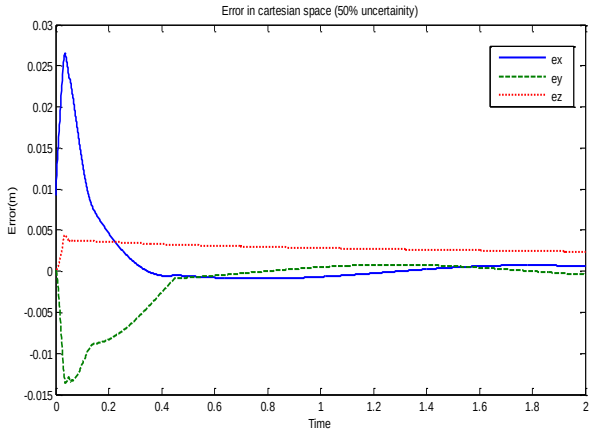

Figure 13. Tracking error in Cartesian space in simulation C. The error is increased in comparison with simulation B.

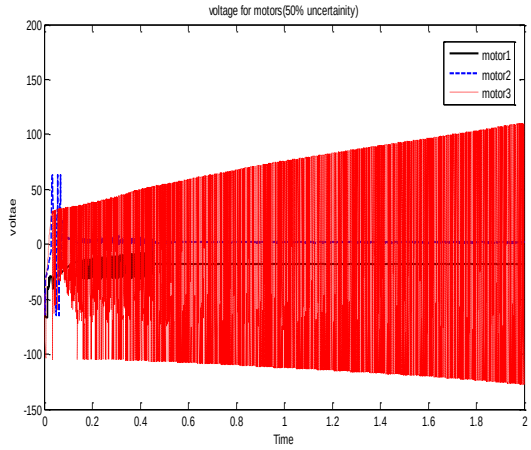

Figure 14. Input voltage for joint actuators in simulation C. It shows that the control system is not applicable when the Jacobian matrix is not exact.

5. IMPROVED TASK SPACE ADAPTIVE FUZZY CONTROL SYSTEM

To overcome the problem of uncertainty in Jacobian matrix, the presented control system should be improved. The controller is now designed directly in task space. The control objective is now to design a feedback law $u = u(Z|\theta)$ based on fuzzy systems and an adaptation law for adjusting the parameter vector θ, such that the robot end effector position X follows the ideal output X_m, as close as possible. The main fuzzy system is the same as Section 3, and vector θ is also the center of fuzzy output membership function. It is necessary to rewrite the robot dynamics in task space. Derivative of (3) would be

$$\ddot{\mathrm{X}} = \dot{\mathrm{J}}(q)\dot{q} + \mathrm{J}(q)\ddot{q} \tag{22}$$

by substituting $\ddot{q}$ from (6) into (22) we have

$$\ddot{\mathrm{X}} = \dot{\mathrm{J}}(q)\dot{q} + \mathrm{J}(q)\mathrm{N}(\mathrm{q},\dot{\mathrm{q}}) + \mathrm{J}(q)bu \tag{23}$$

based on feedback linearization theory, u can be written as

$$\mathrm{u} = \mathrm{H}(\mathrm{q},\dot{\mathrm{q}}) + \frac{1}{b}\mathrm{J}^{-1}V \tag{24}$$

where

$$\mathrm{H}(\mathrm{q},\dot{\mathrm{q}}) = \frac{1}{b}[-N(q,\dot{q}) - \mathrm{J}^{-1}\dot{J}\dot{q}] \tag{25}$$

so, the closed-loop task space robot dynamic by substituting (24) in (23) can be calculated as

$$\ddot{X} = \mathrm{V} \tag{26}$$

now by choosing V based on linear control systems as

$$V = \ddot{X}_d + k_1\left(\dot{X}_d - \dot{X}\right) + k_2(X_d - X) = \ddot{X}_d + \mathbf{K}^{\mathrm{T}}\mathbf{E}_{\boldsymbol{T}} \tag{27}$$

where $\mathbf{E}_{\boldsymbol{T}} = (e_{task}, \dot{e}_{task})^T$ is task space error, and $\mathbf{K} = (k_2, k_1)^T$, the ideal control law would be as in the following equation if robot dynamics are known:

$$u^* = [\mathrm{H}(\mathrm{q}, \dot{\mathrm{q}}) + \tfrac{1}{b}\mathrm{J}^{-1}\ddot{X}_d] + \tfrac{1}{b}\mathrm{J}^{-1}\mathbf{K}^{\mathrm{T}}\mathbf{E}_{\boldsymbol{T}} \tag{28}$$

Since the robot dynamic is unknown, the control objective is to design a fuzzy controller $u(Z|\theta)$ which estimate u^*. Z is vector of fuzzy system input. By applying $u(Z|\theta)$ to system (considering $(u(Z|\theta) = u^* + [u(Z|\theta) - u^*])$) , the error dynamic system would be

$$\ddot{\mathrm{e}}_{\mathrm{task}} = -\mathbf{K}^{\mathrm{T}}\mathbf{E}_{\boldsymbol{T}} + J\mathrm{b}[\mathrm{u}^* - \mathrm{u}(\mathrm{Z}|\theta)] \tag{29}$$

now by defining the matrixes Λ and B as

$$\Lambda = \begin{bmatrix} \bar{O} & \bar{1} \\ -K_2 & -K_1 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} O \\ J\mathrm{b} \end{bmatrix} \tag{30}$$

the error dynamics can be rewritten in matrix form as

$$\dot{\mathbf{E}}_{\boldsymbol{T}} = \Lambda\mathbf{E}_{\boldsymbol{T}} + \boldsymbol{B}[\mathrm{u}^* - \mathrm{u}(\mathrm{Z}|\theta)] \tag{31}$$

by using the same definitions for ideal parameter θ^* and least estimation error ω in (12) and (13), (31) can be written as

$$\dot{\mathbf{E}}_{\boldsymbol{T}} = \Lambda\mathbf{E}_{\boldsymbol{T}} + \boldsymbol{B}(\theta^* - \theta)^T\varepsilon(Z) - \mathbf{B}\omega \tag{32}$$

now, it is required to stabilize the error dynamic (32). The following Lyapunov is the candidate to guarantee the stability.

$$V = \tfrac{1}{2}\mathbf{E}_{\boldsymbol{T}}{}^T P\mathbf{E}_{\boldsymbol{T}} + \tfrac{b}{2\gamma}(\theta^* - \theta)^T(\theta^* - \theta) \tag{33}$$

where γ is a positive constant and P is a positive definite matrix which satisfies the following Lyapunov relation.

$$\Lambda^{\mathrm{T}}\mathrm{P} + \mathrm{P}\Lambda = -\mathrm{Q} \tag{34}$$

the derivative of (33) with respect to time would be

$$\dot{V} = -\tfrac{1}{2}\mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}\mathrm{Q}\mathbf{E}_{\boldsymbol{T}} + \mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}P\boldsymbol{B}[(\theta^* - \theta)^T\varepsilon(Z) - \omega] - \tfrac{b}{\gamma}(\theta^* - \theta)^T\dot{\theta} \tag{35}$$

same assumption like (16), (35) can be rewritten as

$$\dot{V} = -\tfrac{1}{2}\mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}\mathrm{Q}\mathbf{E}_{\boldsymbol{T}} + \tfrac{b}{\gamma}(\theta^* - \theta)^T\left[\gamma\mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}p_n J\varepsilon(Z) - \dot{\theta}\right] - \mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}p_n Jb\omega \tag{36}$$

so, by choosing the adaption law in form of

$$\dot{\theta} = \gamma\mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}p_n J\varepsilon(Z) \tag{37}$$

then the derivative of Lyapunov function will be

$$\dot{V} = -\tfrac{1}{2}\mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}\mathrm{Q}\mathbf{E}_{\boldsymbol{T}} - \mathbf{E}_{\boldsymbol{T}}{}^{\mathrm{T}}p_n Jb\omega \tag{38}$$

which is like Section 3. If the fuzzy system is designed well with small estimation error such that $|\mathbf{e}^{\mathrm{T}} p_n b\omega| < \frac{1}{2}\boldsymbol{e}^{\mathrm{T}}\mathrm{Q}\mathbf{e}$, then (38) results in $\dot{V} < 0$. The updating law is like (18) but now it depends on the task space error, and Jacobian matrix is added to the equation. Hence, the controller can compensate the uncertainty in the Jacobian matrix. The controller structure is shown in Figure 15. The proposed improved adaptive controller is applied to PUMA robot, and simulation results are investigated in the following subsection.

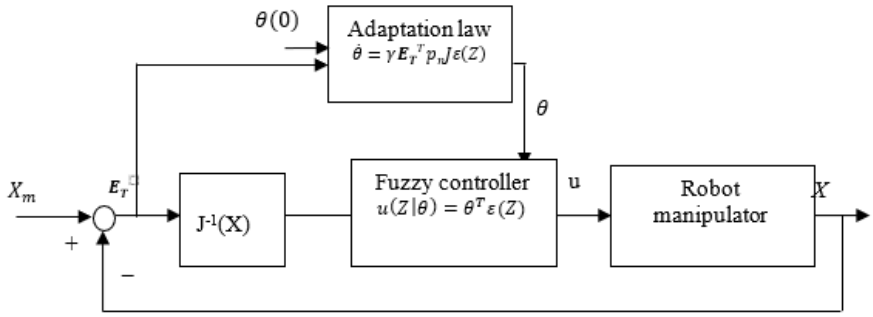

Figure 15. Improved task space adaptive fuzzy control structure

5.1. Improved task space adaptive control system simulation (simulation D)

To investigate the performance of the proposed controller, it is applied to PUMA 560 robot system to track a desired circle in Cartesian space. The tracking error is calculated in task space along the axes of x,y and z. These errors are used for adaption mechanism. Then by using inverse Jacobian, the joint error is calculated from measured task space error. The joint space error is then sent to fuzzy systems to control the joints. Like |Simulation C, the Jacobian matrix has 50 percent uncertainty. By considering $= 20$, the robot end effector position tracking desired trajectory is displayed in Figure 16. Tracking errors are shown in Figure 17. The error is smaller than 0.1 millimeter in steady state. The actuators voltages are displayed in Figure 18 which are limited and acceptable. Figure 19 illustrates the change of center of fuzzy output membership function during simulation. The rules which are fired with higher weight, the corresponding center of output function will be adapted more. From all the simulation results, the proposed control system has successful performance in presence of kinematic uncertainty.

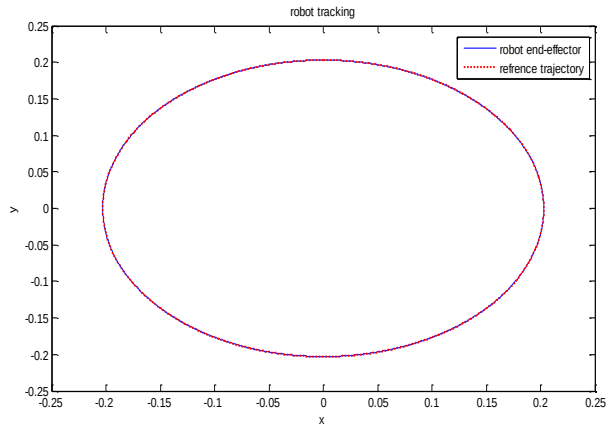

Figure 16. Position of robot end effector tracking the desired circle in simulation D.

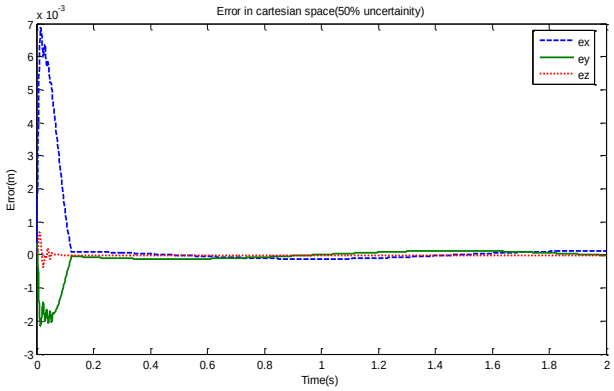

Figure 17. Task space tracking error in simulation D

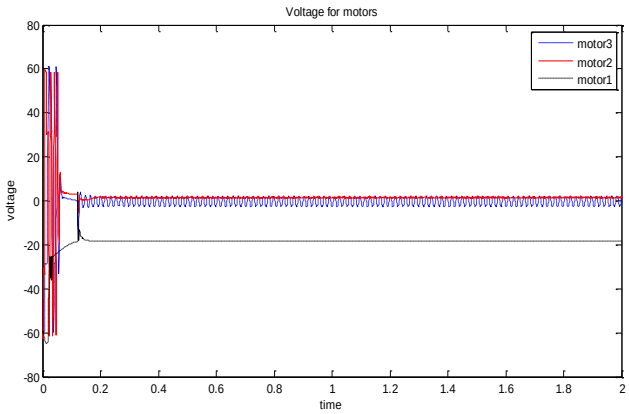

Figure 18. Actuator input voltages in Simulation D.

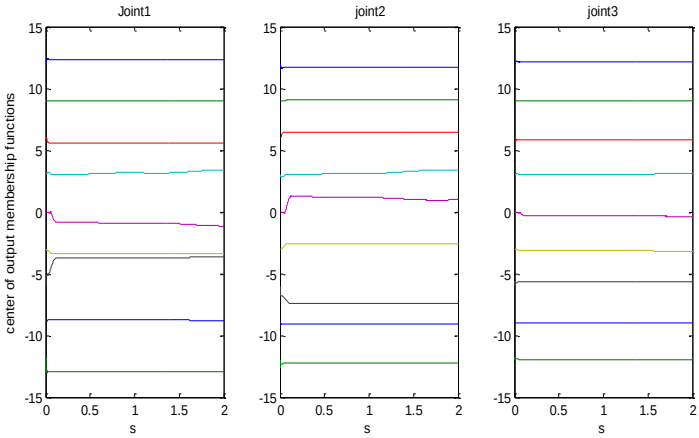

Figure 19. Center of fuzzy output membership functions (vector θ)

6. CONCLUSION

An adaptive fuzzy control system is proposed in this paper to control the robot manipulator in task space subjected to both dynamic and kinematic uncertainty. The proposed system is first designed for joint space tracking and then is improved to control the robot end effector in task space. Four different simulations have been conducted to illustrate the successful performance of the proposed controller in both joint and task spaces.

REFERENCES

[1] Y. Bouteraa, I. B. Abdallah, and J. Ghommam, "Task-space region-reaching control for medical robot manipulator," *Computers & Electrical Engineering,* 2017.

[2] M. Galicki, "Finite-time trajectory tracking control in a task space of robotic manipulators," *Automatica,* vol. 67, pp. 165-170, 2016.

[3] R. Kelly, "Regulation of manipulators in generic task space: An energy shaping plus damping injection approach," *IEEE Transactions on Robotics and Automation,* vol. 15, pp. 381-386, 1999.

[4] A. Azarfar, H. T. Shandiz, and M. Shafiee, "Adaptive feedback control for linear singular systems," *Turkish Journal of Electrical Engineering & Computer Sciences,* vol. 22, pp. 132-142, 2014.

[5] A. Azarfar, H. T. Shandiz, and M. Shafiee, "Adaptive Control for Nonlinear Singular Systems," *Information Technology And Control,* vol. 43, pp. 188-198, 2014.

[6] C.-C. Cheah, C. Liu, and J.-J. E. Slotine, "Adaptive tracking control for robots with unknown kinematic and dynamic properties," *The International Journal of Robotics Research,* vol. 25, pp. 283-296, 2006.

[7] W. E. Dixon, "Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics," *IEEE Transactions on Automatic Control,* vol. 52, pp. 488-493, 2007.

[8] C.-C. Cheah, C. Liu, and J.-J. E. Slotine, "Adaptive Jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models," *IEEE transactions on automatic control,* vol. 51, pp. 1024-1029, 2006.

[9] I. F. Jasim, P. W. Plapper, and H. Voos, "Adaptive sliding mode fuzzy control for unknown robots with arbitrarily-switched constraints," *Mechatronics,* vol. 30, pp. 174-186, 2015.

[10] H. Wang, "Adaptive control of robot manipulators with uncertain kinematics and dynamics," *IEEE Transactions on Automatic Control,* 2016.

[11] X. Huang, H. Gao, J. Li, R. Mao, and J. Wen, "Adaptive back-stepping tracking control of robot manipulators considering actuator dynamic," in *Advanced Intelligent Mechatronics (AIM), 2016 IEEE International Conference on,* 2016, pp. 941-946.

[12] B. Alqaudi, H. Modares, I. Ranatunga, S. M. Tousif, F. L. Lewis, and D. O. Popa, "Model reference adaptive impedance control for physical human-robot interaction," *Control Theory and Technology,* vol. 14, pp. 68-82, 2016.

[13] W. He, Y. Dong, and C. Sun, "Adaptive Neural Impedance Control of a Robotic Manipulator With Input Saturation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* vol. 46, pp. 334-344, 2016.

[14] X. Li and C. C. Cheah, "Adaptive neural network control of robot based on a unified objective bound," *IEEE Transactions on Control Systems Technology,* vol. 22, pp. 1032-1043, 2014.

[15] W. He, A. O. David, Z. Yin, and C. Sun, "Neural Network Control of a Robotic Manipulator With Input Deadzone and Output Constraint," *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* vol. 46, pp. 759-770, 2016.

[16] H. Chaudhary, V. Panwar, R. Prasad, and N. Sukavanam, "Adaptive neuro fuzzy based hybrid force/position control for an industrial robot manipulator," *Journal of Intelligent Manufacturing,* vol. 27, pp. 1299-1308, 2016.

[17] M. M. Fateh and A. Azarfar, "Improving fuzzy control of robot manipulators by increasing scaling factors," *ICIC Express Letters,* vol. 3, pp. 513-518, 2009.

[18] F.-C. Liu, L.-H. Liang, and J.-J. Gao, "Fuzzy PID control of space manipulator for both ground alignment and space applications," *International Journal of Automation and computing,* vol. 11, pp. 353-360, 2014.

[19] Y.-H. Chang, C.-Y. Yang, W.-S. Chan, H.-W. Lin, and C.-W. Chang, "Adaptive fuzzy sliding-mode formation controller design for multi-robot dynamic systems," *International Journal of Fuzzy Systems,* vol. 16, p. 121, 2014.

[20] Q. Zhou, H. Li, and P. Shi, "Decentralized adaptive fuzzy tracking control for robot finger dynamics," *IEEE Transactions on Fuzzy Systems,* vol. 23, pp. 501-510, 2015.

[21] K. Lochan and B. Roy, "Control of two-link 2-DOF robot manipulator using fuzzy logic techniques: A review," in *Proceedings of Fourth International Conference on Soft Computing for Problem Solving,* 2015, pp. 499-511.

[22] L.-X. Wang, *A course in fuzzy systems*: Prentice-Hall press, USA, 1999.

[23] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control* vol. 3: Wiley New York, 2006.

[24] M. Van, H.-J. Kang, Y.-S. Suh, and K.-S. Shin, "A robust fault diagnosis and accommodation scheme for robot manipulators," *International Journal of Control, Automation and Systems,* vol. 11, pp. 377-388, 2013.