Proceedings of the 2007 American Control Conference
Marriott Marquis Hotel at Times Square
New York City, USA, July 11-13, 2007

LMI-Based Additive Gain for Guaranteed Cost Control of Robotic Manipulator with Deadzone

Hiroaki Mukaidani, Takayoshi Umeda, Yoshiyuki Tanaka and Toshio Tsuji

Abstract— **In this paper, deadzone compensator with additive gain is proposed for robust tracking problem. After modeling a two-link manipulator via TS fuzzy model, the neural network is trained on-line to reduce the tracking error caused by deadzone. Based on the Linear Matrix Inequality (LMI) design approach, a class of a state feedback controller with additive gain such as neurocontroller is established. Also sufficient condition for the existence of guaranteed cost (GC) controller is derived. Under the proposed intelligent control scheme, it is shown that we succeed in reducing the tracking error.**

I. INTRODUCTION

It is well known that the robotic manipulators have highly nonlinear interconnection terms. As another nonlinearity, a deadzone exists when torque generated by the motor is assumed to be proportional to input voltage. Therefore, the exact dynamic model to compute the torque such that such nonlinear dynamics of the robotic manipulator system is compensated is necessary. So far, to estimate the deadzone, various reliable approaches such as neural networks, adaptive fuzzy logic control, and iterative learning control have been developed [1], [2], [3]. Particularly, stability proofs and design of deadzone compensator for two-link robot arm using a neural networks (NNs) or fuzzy logic are given in [2], [3], respectively. However, in these studies, much effort has been made towards finding a controller that guarantees robust stability. As the practical control system application, the robust stability and the adequate cost performance should be attained.

NNs and fuzzy logic control have been used extensively in feedback control systems. Most applications seem to be ad hoc with no demonstrations of stability for the overall closed-loop systems. Moreover, it is found that attempts to approximate jump functions such as deadzone using smooth activation functions require many NN nodes and many training iterations and still do not yield excellent results.

Recently, the Linear Matrix Inequality (LMI) approach has been applied to the guaranteed cost (GC) control problem [4]. This method has the advantage that not only the robust stability, but also an adequate upper bound of the performance index can be attained. Very recently, the LMI-based guaranteed cost stabilization for the uncertain discrete-time servo systems has been discussed [7]. However, it can be seen that due to the presence of the deadzone in the practical robotic manipulators, the stability of the closed-loop systems has been destroyed. Moreover, the cost performance would be degraded by these nonlinearity and payload variation.

In this paper, the LMI-based additive gain for the GC control problem of robotic manipulator with deadzone is investigated. The compensator consists of the TS fuzzy model and the additive gain. The first one is used to represent the deadzone and the other one is combined to reduce the large cost caused by the GC control. It is interesting to point out that the deadzone can be represented by using the TS fuzzy model. Additionally, the proposed additive gain can be used as the various artificial intelligence controller such as NNs, fuzzy logic control and PID tuning. The main contribution is that the tracking problem with the additive gain and the deadzone can be solved via the LMI. As a result, although the additive gain and the TS fuzzy model is considered in the robotic manipulator with the deadzone, the robust stability of the closed-loop system and the reduction of the cost are both attained. It is noteworthy that the GC control with the additive gain is applied to the robotic manipulator with the deadzone for the first time. Furthermore, adapting the weights of the cost function of the GC controller the transient response can be changed to the adequately desired trajectory. Hence, this result seems to be novelty. Finally, in order to demonstrate the efficiency of our design approach, the numerical example that is based on the robotic manipulator is given.

II. PROBLEM STATEMENT

The fuzzy dynamical model, proposed by Takagi and Sugeno [5], is described by fuzzy IF-THEN rules, which represented local linear input-output relations of a nonlinear system. In this paper, we represent a given nonlinear discrete-time system by the TS fuzzy model whose *i*th rule is of the following form:

$$\begin{aligned}&\textbf{Plantrule } i:\\&\textbf{IF } z_1(k) \text{ is } M_{i1} \text{ and ... and } z_g(k) \text{ is } M_{ig}\\&\textbf{THEN } \boldsymbol{x}(k+1)=\boldsymbol{A}_i\boldsymbol{x}(k)+\boldsymbol{B}_i\boldsymbol{u}(k),\ i=1,2,...,r, \quad (1)\end{aligned}$$

where $\boldsymbol{x}(k) \in \Re^n$ is the state vector, and $\boldsymbol{u}(k) \in \Re^m$ is the control input vector. $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are known constant matrices. M_{ij} is the fuzzy set, r is the number of IF-THEN rules. $z_1(k)$, ... , $z_g(k)$ are the premise variables. It is assumed in this paper that the premise variables do not explicitly depend on the input variables $\boldsymbol{u}(k)$.

H. Mukaidani is with Graduate School of Education, Hiroshima University, 1-1-1 Kagamiyama, Higashi-Hiroshima, 739-8524 Japan. mukaida@hiroshima-u.ac.jp

T. Umeda, Y. Tanaka and T. Tsuji are with Graduate School of Engineering, Hiroshima University, 1-4-1 Kagamiyama, Higashi-Hiroshima, 739-8527 Japan. [umeda, ytanaka, tsuji]@bsys.hiroshima-u.ac.jp

Given a pair of $(\boldsymbol{x}(k),\ \boldsymbol{u}(k))$, the final output of the fuzzy system is inferred as follows:

$$\begin{aligned}\boldsymbol{x}(k+1) &= \frac{\sum_{i=1}^{r}\mu_i(\boldsymbol{z}(k))[\boldsymbol{A}_i\boldsymbol{x}(k)+\boldsymbol{B}_i\boldsymbol{u}(k)]}{\sum_{i=1}^{r}\mu_i(\boldsymbol{z}(k))} \\ &= \sum_{i=1}^{r} h_i(\boldsymbol{z}(k))[\boldsymbol{A}_i\boldsymbol{x}(k)+\boldsymbol{B}_i\boldsymbol{u}(k)] \qquad (2)\end{aligned}$$

where $\boldsymbol{z}(k) = [z_1(k)\ z_2(k)\ ...\ z_g(k)]$ and

$$\begin{aligned}\mu_i(\boldsymbol{z}(k)) &= \prod_{j=1}^{g} M_{ij}(z_j(k)), \\ h_i(\boldsymbol{z}(k)) &= \frac{\mu_i(\boldsymbol{z}(k))}{\sum_{i=1}^{r}\mu_i(\boldsymbol{z}(k))}.\end{aligned}$$

$M_{ij}(z_j(k))$ is the grade of member ship of $z_j(k)$ in M_{ij}. It is assumed that

$$\mu_i(\boldsymbol{z}(k)) \geq 0,\ i=1,\ 2,\ ...,\ r \text{ and } \sum_{i=1}^{r}\mu_i(\boldsymbol{z}(k)) > 0 \qquad (3)$$

for all k. Then we can obtain the following conditions:

$$h_i(\boldsymbol{z}(k)) \geq 0,\ i=1,\ 2,\ ...,\ r \text{ and } \sum_{i=1}^{r} h_i(\boldsymbol{z}(k)) = 1 \qquad (4)$$

for all k.

It should be noted that the deadzone can be represented by relative few fuzzy rules, while the NNs need to have many nodes and many training iterations to denote the deadzone [3]. Thus, we will use TS fuzzy model.

The cost function associated with the system is given by

$$J = \sum_{k=0}^{\infty}(\boldsymbol{x}^{\mathrm{T}}(k)\boldsymbol{Q}\boldsymbol{x}(k) + \boldsymbol{u}^{\mathrm{T}}(k)\boldsymbol{R}\boldsymbol{u}(k)) \qquad (5)$$

where $\boldsymbol{Q} > 0$ and $\boldsymbol{R} > 0$ are given weighting matrices. Based on the fuzzy model (1), the following compensator is proposed to deal with the LQ control problem for a nonlinear discrete-time system that includes the deadzone with the cost function (5):

$$\boldsymbol{u}(k) = (\boldsymbol{K} + \Delta\boldsymbol{K}(k))\boldsymbol{x}(k), \qquad (6)$$

where

$$\Delta\boldsymbol{K}(k) = \boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k, \qquad (7)$$

where $\boldsymbol{D}_k$ and $\boldsymbol{E}_k$ are known constant matrices and $\boldsymbol{N}(k) \in \Re^{p\times q}$ is arbitrary function of additive gain. In this paper, NN will be used as additive gain. It is assumed that $\boldsymbol{N}(k)$ satisfies the norm condition

$$\boldsymbol{N}^{\mathrm{T}}(k)\boldsymbol{N}(k) \leq \boldsymbol{I}_q. \qquad (8)$$

Hence, from (2) and (6), we can obtain that the closed-loop system

$$\boldsymbol{x}(k+1) = \sum_{i=1}^{r} h_i(\boldsymbol{z}(k))\,[\boldsymbol{A}_i\boldsymbol{x}(k) + \boldsymbol{B}_i\boldsymbol{u}(k)] = \bar{\boldsymbol{A}}\boldsymbol{x}(k) \qquad (9)$$

where $\bar{\boldsymbol{A}} = \sum_{i=1}^{r} h_i(\boldsymbol{z}(k))(\boldsymbol{A}_i + \boldsymbol{B}_i(\boldsymbol{K} + \Delta\boldsymbol{K}(k)))$. In this case, cost function (5) is given by

$$J = \sum_{k=0}^{\infty}\boldsymbol{x}^{\mathrm{T}}(k)[\boldsymbol{Q} + (\boldsymbol{K}+\Delta\boldsymbol{K}(k))^{\mathrm{T}}\boldsymbol{R}(\boldsymbol{K}+\Delta\boldsymbol{K}(k))]\boldsymbol{x}(k). \qquad (10)$$

In this paper, the problem under consideration is to design a compensator (6) such that the resulting closed-loop overall fuzzy system (9) is globally asymptotically stable and the cost function (10) satisfies the bound $J \leq J^*$, where J^* is also termed a GC. Such a controller is called to be the reliable GC controller. Moreover, we want to design an optimal reliable GC fuzzy compensator in the sense of minimizing an upper bound.

III. RELIABLE GC CONTROL OF THE TS FUZZY MODEL

The following result presents a sufficient condition that guarantees the stability of the closed-loop system (9) and provides a GC for the cost function (10).

Theorem 1: Consider the closed-loop system (9) and cost function (10). Assume that there exist matrices $\boldsymbol{P} = \boldsymbol{P}^{\mathrm{T}} > 0$ that satisfy

$$\begin{bmatrix} -\boldsymbol{P} & \Phi^{\mathrm{T}} & \boldsymbol{Q}^{\frac{1}{2}} & \bar{\boldsymbol{A}}^{\mathrm{T}} \\ \Phi & -\boldsymbol{I}_n & 0 & 0 \\ \boldsymbol{Q}^{\frac{1}{2}} & 0 & -\boldsymbol{I}_n & 0 \\ \bar{\boldsymbol{A}} & 0 & 0 & -\boldsymbol{P}^{-1} \end{bmatrix} < 0, \qquad (11)$$

where $\Phi := \boldsymbol{R}^{\frac{1}{2}}(\boldsymbol{K} + \boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)$.

Then the closed-loop system (9) is globally asymptotically stable. In this case, the cost function (10) satisfies the bound

$$J \leq \boldsymbol{x}_0^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x}_0 := J^*, \qquad (12)$$

where $\boldsymbol{x}_0$ is the initial state.

Proof: Assume that the inequalities of (11) are satisfied for some $\boldsymbol{P} = \boldsymbol{P}^{\mathrm{T}}$. It follows from the Schur complement that

$$\begin{aligned}&\bar{\boldsymbol{A}}^{\mathrm{T}}\boldsymbol{P}\bar{\boldsymbol{A}} - \boldsymbol{P} + (\boldsymbol{K} + \boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)^{\mathrm{T}}\boldsymbol{R}(\boldsymbol{K} + \boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k) \\ &\quad + \boldsymbol{Q} < 0. \qquad (13)\end{aligned}$$

Consider the following quadratic functions as the Lyapunov function candidate

$$V(\boldsymbol{x}(k)) = \boldsymbol{x}^{\mathrm{T}}(k)\boldsymbol{P}\boldsymbol{x}(k). \qquad (14)$$

Along the state trajectory of (9), and using (13), we then have

$$\begin{aligned}&V(\boldsymbol{x}(k+1)) - V(\boldsymbol{x}(k)) \\ &= \boldsymbol{x}^{\mathrm{T}}(k)(\bar{\boldsymbol{A}}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A} - \boldsymbol{P})\boldsymbol{x}(k) \\ &< -\boldsymbol{x}^{\mathrm{T}}(k)[(\boldsymbol{K} + \boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)^{\mathrm{T}}\boldsymbol{R}(\boldsymbol{K} + \boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k) \\ &\quad + \boldsymbol{Q}]\boldsymbol{x}(k). \qquad (15)\end{aligned}$$

Obviously, $V(\boldsymbol{x}(k+1)) - V(\boldsymbol{x}(k)) < 0$. Hence, the closed-loop system (9) is globally asymptotically stable. In this case,

$V(\boldsymbol{x}(\infty)) = 0$. Summing the inequality (15) from $k = 0$ to $k = \infty$, we have

$$\begin{aligned} & V(\boldsymbol{x}(\infty)) - V(\boldsymbol{x}(0)) \\ &= -V(\boldsymbol{x}(0)) = -\boldsymbol{x}_0^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x}_0 \\ &< -\sum_{k=0}^{\infty} \boldsymbol{x}^{\mathrm{T}}(k)[\boldsymbol{Q} + (\boldsymbol{K} + \Delta\boldsymbol{K}(k))^{\mathrm{T}}\boldsymbol{R}(\boldsymbol{K} + \Delta\boldsymbol{K}(k))]\boldsymbol{x}(k) \\ &= -J. \end{aligned} \tag{16}$$

Thus, it follows from (16) that (12) holds. ■

Remark 1: Note that the GC of the cost function (10) obtained in Theorem 1 depends on the initial condition $\boldsymbol{x}_0$. To avoid this dependence on $\boldsymbol{x}_0$, we will assume $\boldsymbol{x}_0$ is a random variable satisfying

$$\mathrm{E}[\boldsymbol{x}_0] = 0 \text{ and } \mathrm{E}[\boldsymbol{x}_0\boldsymbol{x}_0^{\mathrm{T}}] = \boldsymbol{I}_n, \tag{17}$$

where $\mathrm{E}[\cdot]$ denotes the expectation operator. Then, the cost bound (12) leads to

$$\mathrm{E}[J] \leq \mathrm{E}[\boldsymbol{x}_0^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x}_0] = \mathrm{trace}[\boldsymbol{P}]. \tag{18}$$

In order to resolve the nonconvexity problem of (11), we will use the similar method of [8]. We introduce a new nonsingular matrix variable $\boldsymbol{Z} \in \Re^{n\times n}$ into (11) and consider the new extended LMIs (19). The following theorem presents a solution to the reliable GC control problem for the fuzzy system (1) in terms of LMIs.

Theorem 2: Consider the system (1) and cost function $\mathrm{E}[J]$ of (18). If there exist a common nonsingular matrix $\boldsymbol{Z}$, $\boldsymbol{X} = \boldsymbol{X}^{\mathrm{T}}$ and $\boldsymbol{Y}$, satisfying the LMI (19) for all $i = 1, 2, \ldots, r$, then there exists a reliable GC controller (6) such that the closed-loop system (9) is globally asymptotically stable and the cost function $\mathrm{E}[J]$ of (18) satisfies the bound

$$\mathrm{E}[J] \leq \mathrm{trace}[\boldsymbol{X}^{-1}]. \tag{24}$$

Furthermore, the state feedback gain matrix is given by

$$\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{Z}^{-1}. \tag{25}$$

In order to prove Theorem 2, the following Lemma will be used [9].

Lemma 1: Consider the appropriate matrix $\mathcal{F}$ which is satisfying $\mathcal{F}\mathcal{F}^T \leq \boldsymbol{I}_n$ and for any matrices $\mathcal{G}$ and $\mathcal{H}$, there exists the positive parameter $\lambda > 0$ such that the following inequality holds

$$\mathcal{G}\mathcal{F}\mathcal{H} + \mathcal{H}^T\mathcal{F}^T\mathcal{G}^T \leq \lambda\mathcal{G}\mathcal{G}^T + \lambda^{-1}\mathcal{H}^T\mathcal{H}. \tag{26}$$

Let us prove Theorem 2 by using the above Lemma 1.

Proof: Assume there exist a common nonsingular matrix $\boldsymbol{Z}$, $\boldsymbol{X} = \boldsymbol{X}^{\mathrm{T}}$, and $\boldsymbol{Y}$, satisfying the LMI (19). Let us define

$$\boldsymbol{Y} = \boldsymbol{K}\boldsymbol{Z}. \tag{27}$$

Applying the Schur complement [10] to (19) and using (27) result in $\boldsymbol{N}_i$. Using the standard matrix inequality (26) of Lemma 1, for all admissible additive gain, it is easy to verify that $\boldsymbol{M}_i < 0$. Multiplying $h_i(z(k)) > 0$ to $\boldsymbol{M}_i < 0$ and summing the inequality (20) from $i = 1$ to $i = r$, we have (21). It is clear from (21) that $\boldsymbol{X} > 0$ and $\boldsymbol{Z} + \boldsymbol{Z}^{\mathrm{T}} > \boldsymbol{X}$. Thus, $\boldsymbol{Z}$ is nonsingular. Using the inequality $\boldsymbol{X} - \boldsymbol{Z} - \boldsymbol{Z}^{\mathrm{T}} \geq -\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{X}^{-1}\boldsymbol{Z}$ that is obtained from $(\boldsymbol{Z}^{\mathrm{T}} - \boldsymbol{X})\boldsymbol{X}^{-1}(\boldsymbol{Z} - \boldsymbol{X}) \geq 0$, we have (22). Pre- and post- multiplying (22) by $\mathbf{block\ diag}\left(\boldsymbol{Z}^{-\mathrm{T}},\ \boldsymbol{I}_n,\ \boldsymbol{I}_n,\ \boldsymbol{I}_n\right)$ and $\mathbf{block\ diag}\left(\boldsymbol{Z}^{-1},\ \boldsymbol{I}_n,\ \boldsymbol{I}_n,\ \boldsymbol{I}_n\right)$, respectively, we have (23). Obviously, the inequality of (23) results in (11) under

$$\boldsymbol{P} = \boldsymbol{X}^{-1}. \tag{28}$$

It follows from Theorem 1 that the closed-loop system (9) is globally asymptotically stable and we have (24). Furthermore, by (27), we obtain (25). ■

It seems to be formidable to obtain the common LMIs solution that satisfy the (1). However, since the deadzone can be represented by relative few fuzzy rules, we can obtain the common reliable solutions $\boldsymbol{X}$, $\boldsymbol{Y}$ and $\boldsymbol{Z}$.

To minimize the upper bound on the cost function $\mathrm{E}[J]$ of (18), we seek to minimize an upper bound on the GC $\mathrm{trace}[\boldsymbol{X}^{-1}]$. With an auxiliary variable $\boldsymbol{V} \in \Re^{n\times n}$, which is an upper bound on $\boldsymbol{X}^{-1}$ and consider the following LMIs:

$$\begin{bmatrix} -\boldsymbol{V} & \boldsymbol{I}_n \\ \boldsymbol{I}_n & -\boldsymbol{X} \end{bmatrix} < 0, \tag{29}$$

where we want to minimize $\mathrm{trace}[\boldsymbol{V}]$ instead of

$$\mathrm{E}[J] \leq \mathrm{trace}[\boldsymbol{X}^{-1}] < \mathrm{trace}[\boldsymbol{V}]. \tag{30}$$

IV. ADDITIVE GAIN USING NEURAL NETWORK

The LMI design approach usually yields the conservative controller and large cost due to the presence of the additive gain $\boldsymbol{N}(k)$ such as the output of NN.

A. On-line learning algorithm of neurocontroller

It is expected that the reduction of the cost will be attained when the neurocontroller can manage that the error dynamics including deadzone converges to zero rapidly as possible. That is, the neurocontroller is required to reduce the response of system.

$$\mathcal{E}(k) := \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}(k+1)\boldsymbol{x}(k+1). \tag{31}$$

If $\mathcal{E}(k)$ can be reduced as small as possible, the position error $\boldsymbol{x}(k)$ would become small. That is, robotic manipulator will follow the desired position more precisely.

In the learning phase of NN, the weight updating rules can be described as

$$w_g^{ij}(k+1) = w_g^{ij}(k) + \Delta w_g^{ij}(k). \tag{32}$$

On the other hand, the modification of the weight coefficient $w_g^{ij}(k)$ is given by

$$\Delta w_g^{ij}(k) = -\eta\frac{\partial\mathcal{E}(k)}{\partial w_g^{ij}(k)}, \tag{33a}$$

$$\frac{\partial\mathcal{E}(k)}{\partial w_g^{ij}(k)} = \frac{\partial\mathcal{E}(k)}{\partial\boldsymbol{N}(k)} \cdot \frac{\partial\boldsymbol{N}(k)}{\partial w_g^{ij}(k)}, \tag{33b}$$

where η is the learning ratio.

$$\begin{bmatrix} \boldsymbol{X}-\boldsymbol{Z}-\boldsymbol{Z}^{\mathrm{T}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{E}_k^{\mathrm{T}} & \boldsymbol{Y}^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Q}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{A}_i^{\mathrm{T}}+\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} \\ \boldsymbol{E}_k\boldsymbol{Z} & -\epsilon_k\boldsymbol{I}_n & 0 & 0 & 0 \\ \boldsymbol{R}^{\frac{1}{2}}\boldsymbol{Y} & 0 & -\boldsymbol{I}_n+\epsilon_k\boldsymbol{R}^{\frac{1}{2}}\boldsymbol{D}_k\boldsymbol{D}_k^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & 0 & \epsilon_k\boldsymbol{R}^{\frac{1}{2}}\boldsymbol{D}_k\boldsymbol{D}_k^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} \\ \boldsymbol{Q}^{\frac{1}{2}}\boldsymbol{Z} & 0 & 0 & -\boldsymbol{I}_n & 0 \\ \boldsymbol{A}_i\boldsymbol{Z}+\boldsymbol{B}_i\boldsymbol{Y} & 0 & \epsilon_k\boldsymbol{B}_i\boldsymbol{D}_k\boldsymbol{D}_k^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & 0 & -\boldsymbol{X}+\epsilon_k\boldsymbol{B}_i\boldsymbol{D}_k\boldsymbol{D}_k^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} \end{bmatrix} < 0. \tag{19}$$

$$\begin{aligned} \boldsymbol{M}_i \;:=\; & \begin{bmatrix} \boldsymbol{X}-\boldsymbol{Z}-\boldsymbol{Z}^{\mathrm{T}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Q}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}(\boldsymbol{A}_i+\boldsymbol{B}_i\boldsymbol{K})^{\mathrm{T}} \\ \boldsymbol{R}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{Z} & -\boldsymbol{I}_n & 0 & 0 \\ \boldsymbol{Q}^{\frac{1}{2}}\boldsymbol{Z} & 0 & -\boldsymbol{I}_n & 0 \\ (\boldsymbol{A}_i+\boldsymbol{B}_i\boldsymbol{K})\boldsymbol{Z} & 0 & 0 & -\boldsymbol{X} \end{bmatrix} \\ & + \begin{bmatrix} 0 & \boldsymbol{Z}^{\mathrm{T}}(\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & 0 & \boldsymbol{Z}^{\mathrm{T}}(\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} \\ \boldsymbol{R}^{\frac{1}{2}}\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k\boldsymbol{Z} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \boldsymbol{B}_i\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k\boldsymbol{Z} & 0 & 0 & 0 \end{bmatrix} \\ \leq\; & \begin{bmatrix} \boldsymbol{X}-\boldsymbol{Z}-\boldsymbol{Z}^{\mathrm{T}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Q}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}(\boldsymbol{A}_i+\boldsymbol{B}_i\boldsymbol{K})^{\mathrm{T}} \\ \boldsymbol{R}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{Z} & -\boldsymbol{I}_n & 0 & 0 \\ \boldsymbol{Q}^{\frac{1}{2}}\boldsymbol{Z} & 0 & -\boldsymbol{I}_n & 0 \\ (\boldsymbol{A}_i+\boldsymbol{B}_i\boldsymbol{K})\boldsymbol{Z} & 0 & 0 & -\boldsymbol{X} \end{bmatrix} \\ & +\epsilon_k \begin{bmatrix} 0 \\ \boldsymbol{R}^{\frac{1}{2}}\boldsymbol{D}_k \\ 0 \\ \boldsymbol{B}_i\boldsymbol{D}_k \end{bmatrix} \begin{bmatrix} 0 & \boldsymbol{D}_k^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & 0 & \boldsymbol{D}_k^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} \end{bmatrix} + \epsilon_k^{-1} \begin{bmatrix} \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{E}_k^{\mathrm{T}} \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{E}_k\boldsymbol{Z} & 0 & 0 & 0 \end{bmatrix} = \boldsymbol{N}_i < 0. \end{aligned} \tag{20}$$

$$\sum_{i=1}^{r} h_i(\boldsymbol{z}(k))\boldsymbol{M}_i < 0 \Leftrightarrow \begin{bmatrix} \boldsymbol{X}-\boldsymbol{Z}-\boldsymbol{Z}^{\mathrm{T}} & \boldsymbol{Z}^{\mathrm{T}}(\boldsymbol{K}+\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Q}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\bar{\boldsymbol{A}}^{\mathrm{T}} \\ \boldsymbol{R}^{\frac{1}{2}}(\boldsymbol{K}+\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)\boldsymbol{Z} & -\boldsymbol{I}_n & 0 & 0 \\ \boldsymbol{Q}^{\frac{1}{2}}\boldsymbol{Z} & 0 & -\boldsymbol{I}_n & 0 \\ \bar{\boldsymbol{A}}\boldsymbol{Z} & 0 & 0 & -\boldsymbol{X} \end{bmatrix} < 0. \tag{21}$$

$$\begin{bmatrix} -\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{X}^{-1}\boldsymbol{Z} & \boldsymbol{Z}^{\mathrm{T}}(\boldsymbol{K}+\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Q}^{\frac{1}{2}} & \boldsymbol{Z}^{\mathrm{T}}\bar{\boldsymbol{A}}^{\mathrm{T}} \\ \boldsymbol{R}^{\frac{1}{2}}(\boldsymbol{K}+\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)\boldsymbol{Z} & -\boldsymbol{I}_n & 0 & 0 \\ \boldsymbol{Q}^{\frac{1}{2}}\boldsymbol{Z} & 0 & -\boldsymbol{I}_n & 0 \\ \bar{\boldsymbol{A}}\boldsymbol{Z} & 0 & 0 & -\boldsymbol{X} \end{bmatrix} < 0. \tag{22}$$

$$\begin{bmatrix} -\boldsymbol{X}^{-1} & (\boldsymbol{K}+\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k)^{\mathrm{T}}\boldsymbol{R}^{\frac{1}{2}\,\mathrm{T}} & \boldsymbol{Q}^{\frac{1}{2}} & \bar{\boldsymbol{A}}^{\mathrm{T}} \\ \boldsymbol{R}^{\frac{1}{2}}(\boldsymbol{K}+\boldsymbol{D}_k\boldsymbol{N}(k)\boldsymbol{E}_k) & -\boldsymbol{I}_n & 0 & 0 \\ \boldsymbol{Q}^{\frac{1}{2}} & 0 & -\boldsymbol{I}_n & 0 \\ \bar{\boldsymbol{A}} & 0 & 0 & -\boldsymbol{X} \end{bmatrix} < 0. \tag{23}$$

The term $\dfrac{\partial \mathcal{E}(k)}{\partial \boldsymbol{N}(k)}$ of the equation (33b) can be calculated from the energy function (31) as follows:

$$\frac{\partial \mathcal{E}(k)}{\partial \boldsymbol{N}(k)} = \boldsymbol{x}(k+1)\boldsymbol{B}\boldsymbol{D}_k\boldsymbol{E}_k\boldsymbol{x}(k). \tag{34}$$

Using (34), the NN can be trained so as to decrease the cost $\mathrm{E}[J]$ in on-line.

B. Multilayered Neural networks

The utilized NN is of a three–layer feed–forward network as shown in Fig. 1. The linear function is utilized in the neurons of the input and the hidden layers, and a sigmoid function in the output layer. The inputs and outputs of each layer can be described as follows.

$$s_g^i(k) := \begin{cases} U_i(k), & \{g=1(\text{input layer})\} \\ \sum w_1^{(i,j)}(k)o_1^j(k), & \{g=2(\text{hidden layer})\}, \\ \sum w_2^{(i,j)}(k)o_2^j(k), & \{g=3(\text{output layer})\} \end{cases}$$

$$o_g^i(k) := \begin{cases} s_1^i(k), & \{g=1(\text{input layer})\} \\ s_2^i(k)+\theta_1^{(i)}(k), & \{g=2(\text{hidden layer})\}, \\ \dfrac{1-e^{(-s_3^i(k)+\theta_2^{(i)}(k))}}{1+e^{(-s_3^i(k)+\theta_2^{(i)}(k))}}, & \{g=3(\text{output layer})\} \end{cases}$$

where $s_g^i(k)$ and $o_g^i(k)$ are the input and the output of the neuron i in the gth layer at the time t. $w_g^{i,j}(k)$ indicates the weight coefficient from the neuron j in the gth layer to the neuron i in the $(g+1)$th layer. $U_i(k)$ is the input of NN. $\theta_g^{(i)}(k)$ is a positive constant for the threshold of the

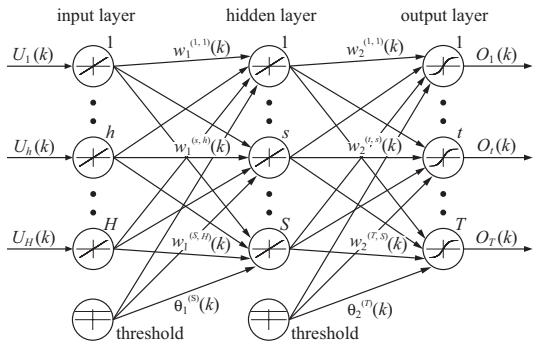

Fig. 1. Structure of the multilayered neural networks

neuron i in the $(g+1)$th layer. As the additive gain defined in the formula (7), the outputs of NN are set in the range of $[-1.0,\ 1.0]$.

V. A NUMERICAL EXAMPLE

In order to demonstrate the effectiveness of the proposed deadzone compensator, a numerical example is given. Consider the two-link robotic manipulator that is given in Fig. 2. The system parameters are chosen as link mass $m_1 = 7.43$[kg], $m_2 = 1.72$[kg], lengths $l_1 = l_2 = 0.2$[m], joint angular positions $\theta_1(k)$[rad], $\theta_2(k)$[rad].

Let us consider a following linearized equation of motion for the two-link robotic manipulator with the deadzone given by

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\mathcal{D}\boldsymbol{u}(k), \tag{35}$$

where

$$\mathcal{D}\boldsymbol{u}(k) = \boldsymbol{f}(\boldsymbol{u}(k)) := \begin{cases} \boldsymbol{u}(k), & |u_i(k)| > 0.2 \\ 0, & |u_i(k)| \le 0.2 \end{cases},$$
$$\boldsymbol{x}(k) = \begin{bmatrix} x_1(k) & x_2(k) & x_3(k) & x_4(k) \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} \theta_1(k) & \theta_2(k) & \dot{\theta}_1(k) & \dot{\theta}_2(k) \end{bmatrix}^{\mathrm{T}},$$
$$\boldsymbol{u}(k) := \begin{bmatrix} u_1^{\mathrm{T}}(k) & u_2^{\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}}.$$

Hence, taking the deadzone nonlinearity shown by Fig. 3 into account, it is assumed that the model of the nonlinear discrete-time system with deadzone is given by

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\begin{bmatrix} u_1^3(k) \\ u_2^3(k) \end{bmatrix} + \boldsymbol{C}\boldsymbol{u}(k), \tag{36}$$

where

$$\boldsymbol{A} = \begin{bmatrix} 1.0000 & 0 & 0.001 & 0 \\ 0 & 1.0000 & 0 & 0.001 \\ 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix},$$
$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.003 & 0 \\ 0 & 0.4055 \end{bmatrix},\ \boldsymbol{C} = 10^{-3}\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.003 & 0 \\ 0 & 0.4055 \end{bmatrix},$$
$$\boldsymbol{D}_k = \begin{bmatrix} 0.03 & 0.001 & 0.01 & 0.001 \\ 0.001 & 0.02 & 0.001 & 0.025 \end{bmatrix},\ E_k = 0.1,$$
$$\boldsymbol{N}(k) = \mathbf{diag}\left(N_1(k)\ N_2(k)\ N_3(k)\ N_4(k) \right),$$

with $N_i(k)$ denotes the NN input such that $|N_i(k)| \le 1$.

The novel idea is as follows. It should be noted that for sufficiently small constant C such deadzone can be approximate $f(u_i) = u_i^3 - Cu_i,\ i = 1,\ 2$. Thus, we choose this parameter as $C = 0.001$. Finally, we can apply the TS fuzzy model to this deadzone. To construct a TS fuzzy model for this system, it is assumed that $u_i(k) \in [-a, a]$, where a is a positive value. Since the nonlinear term is $u_i^3(k)$, using the same procedure as in [6], the nonlinear term can be represented as

$$u_i^3(k) = F_1^1(u_i(k)) \cdot 1 \cdot u_i(k) + F_1^2(u_i(k)) \cdot 0 \cdot u_i(k), \tag{37}$$

where $F_1^1,\ F_1^2 \in [0, 1]$, and $F_1^1(u_i(k)) + F_1^2(u_i(k)) = 1$. By solving the equations, they are obtained as follows:

$$F_1^1(u_i(k)) = u_i^2(k),\ F_1^2(u_i(k)) = 1 - u_i^2(k).$$

F_1^1 and F_1^2 can be interpreted as membership functions of fuzzy sets. By using these fuzzy sets, the nonlinear system can be represented by the following TS fuzzy model:

Plant rule 1:
If $u_i(k)$ is $F_1^1(u_i(k))$
Then $\boldsymbol{x}(k+1) = \boldsymbol{A}_1\boldsymbol{x}(k) + \boldsymbol{B}_1\boldsymbol{u}(k)$,
Plant rule 2:
If $u_i(k)$ is $F_1^2(u_i(k))$
Then $\boldsymbol{x}(k+1) = \boldsymbol{A}_2\boldsymbol{x}(k) + \boldsymbol{B}_2\boldsymbol{u}(k)$,

where $\boldsymbol{A}_1 = \boldsymbol{A}_2 := \boldsymbol{A}$, $\boldsymbol{B}_1 := \boldsymbol{B} + \boldsymbol{C}$ and $\boldsymbol{B}_2 := \boldsymbol{C}$.

Therefore, it is expected that the TS fuzzy model can exactly represent the nonlinear of the deadzone.

Remark 2: If the deadzone has the complex structure, the fuzzy logic rules have to be increased. However, the existence condition of the common LMIs solutions will be conservative.

The initial system condition is $x_1(0) = \frac{\pi}{3}$, $x_2(0) = \frac{\pi}{4}$, $x_3(0) = 0$, $x_4(0) = 0$, and the weighting matrices are $\boldsymbol{Q} = \mathbf{diag}(200,\ 200,\ 1,\ 1)$ and $\boldsymbol{R} = \mathbf{diag}(0.001,\ 1)$, respectively. Desired reference positions are $x_{1d} = 0$ and $x_{2d} = 0$. In the following simulation, it is assumed that $a = 1$ and when $-0.2 \le u_i(k) \le 0.2$, the system has a deadzone such that $\boldsymbol{u}(k) = 0$.

Solving the LMIs (19), the state feedback control gain $\boldsymbol{K}$ is given by

$$\boldsymbol{K} = \begin{bmatrix} -447.3798 & 0.3191 & -548.9171 & 0.1428 \\ 0.0042 & -7.2240 & -0.0070 & -4.8232 \end{bmatrix}. \tag{38}$$

To specify simulation results, the proposed method is compared with the existing GC approach [7] that does not consider the deadzone. The GC control gain $\hat{\boldsymbol{K}}$ based on the GC approach is given by

$$\hat{\boldsymbol{K}} = \begin{bmatrix} -215.0992 & -0.0305 & -24.6845 & -0.0028 \\ 0.0025 & -8.8177 & 0.0003 & -0.8361 \end{bmatrix}. \tag{39}$$

The neurocontroller consists of four neurons in the input and the output layers and 30 neurons in the hidden layers, respectively. The initial weights are set randomly in the range of $[-0.05,\ 0.05]$.

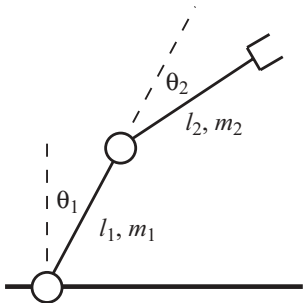

Fig. 2. The two-link robotic manipulator.

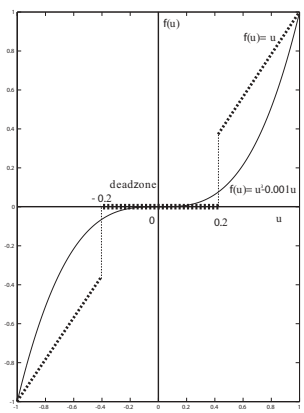

Fig. 3. Deadzone.

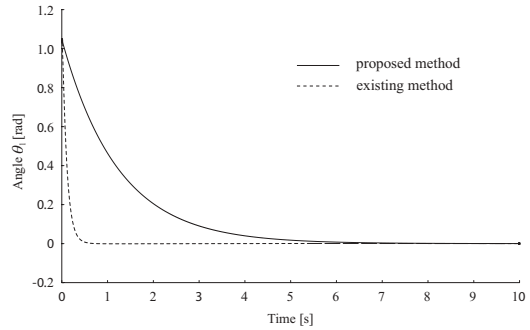

Fig. 4. Simulation result for position $x_1(k) = \theta_1(k)$ of link one.

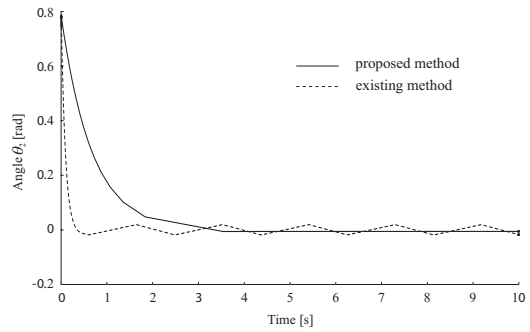

Fig. 5. Simulation result for position $x_2(k) = \theta_2(k)$ of link two.

The simulation results are shown in Figs. 4, 5. Note that the solid lines mean the response via the proposed method, while the dash lines mean the response via the existing method [7]. Although the responses of the proposed system are both stable, the control result of the existing method [7] is very oscillatory for the link two. This result shows that our proposed method can attain good tracking performance for the robotic manipulator system with deadzone system. However, because the proposal method obtains a high gain, the cost is higher than the conventional system.

VI. CONCLUSIONS

In this paper, the LMI-based deadzone compensator of the robotic manipulator has been investigated. Our novel idea is that the TS fuzzy model is used to represent the deadzone and the NN is combined to reduce the large cost. Furthermore, although the compensator consists of TS fuzzy model and NNs, the asymptotic stability of overall closed-loop system with deadzone and adequate cost bound are both attained. It is noteworthy that the proposed controller can be designed by means of the common solution of a few LMIs. Additionally, for the proposed additive gain, the various artificial intelligence control approach can be used. In this paper, although the NN is used, the fuzzy logic control may be used because many NN nodes and many training iterations are not needed compared with NN. It may be noted that the GC control with the additive gain has been applied to the robotic manipulator with the deadzone for the first time. As another important feature, adapting the weights of the cost function, the transient response can be changed appropriately. Finally, the numerical example has shown the excellent results.

In order to demonstrate the new control scheme such as output feedback control case, experimental result of two-link manipulator seems to be needed. This practical problem will be addressed in future investigations.

REFERENCES

[1] J.-X. Xu and V. Badrinath, "Adaptive Robust Iterative Learning Control with Dead Zone Scheme", *Automatica*, vol.36, no.1, 2000, pp. 91-99.

[2] F.L. Lewis, W.K. Tim, L.Z. Wang and Z.X. Li, "Deadzone Compensation in Motion Control Systems Using Adaptive Fuzzy Logic Control," *IEEE Trans. Contr. Syst. Technol.*, vol. 7, no. 6, 1999, pp. 731-742.

[3] R.R. Selmić and F.L. Lewis, "Deadzone Compensation in Motion Control Systems Using Neural Networks," *IEEE Trans. Automat. Contr.*, vol. 45, no. 4, 2000, pp. 602-613.

[4] L. Xie, and Y. C. Soh, "Guaranteed cost control of uncertain discrete-time systems," *Control Theory and Advanced Technology.*, vol. 10, 1995, pp. 1235-1250.

[5] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Trans. Syst., Man, Cybern.*, vol. 15, no. 1, 1985, pp. 116-132.

[6] H.-N. Wu, "Reliable LQ Fuzzy Control for Nonlinear Discrete-Time Systems via LMIs," *IEEE Trans. Syst., Man, Cybern.*, vol. 34, no. 2, 2004, pp. 1270-11275.

[7] H. Mukaidani, S. Sakaguchi, Y. Tanaka and T. Tsuji, "LMI-based neurocontroller for guaranteed cost control of uncertain servo system," *Proc. 2006 American Control Conference*, pp. 2963-2968, Minneapolis, June 2006.

[8] M.C. de Oliveira, J. Bernussou and J.C. Geromel, "A New Discrete-time Robust Stability Condition," *Systems and Control Letters*, vol. 37, no. 4, 1999, pp. 261-265.

[9] Y. Wang, D. J. Hill and G. Guo, "Robust Decentralized Control for Multimachine Power Systems," *IEEE Trans. Circuits and Systems I*, vol. 45, no. 3, 1998, pp. 271-279.

[10] K. Zhou, *Essentials of robust control*, Prentice-Hall, 1998.