Guaranteed Cost Control of Polynomial Fuzzy Systems via a Sum of Squares Approach

著者（英）	Kazuo Tanaka, Hiroshi Ohtake, Hua O. Wang
journal or publication title	IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)
volume	39
number	2
page range	561-567
year	2009-04
URL	http://id.nii.ac.jp/1438/00009302/

doi: 10.1109/TSMCB.2008.2006639

Guaranteed Cost Control of Polynomial Fuzzy Systems via a Sum of Squares Approach

Kazuo Tanaka, *Member, IEEE,* Hiroshi Ohtake, *Member, IEEE,* and Hua O. Wang, *Senior Member, IEEE*

***Abstract*—This paper presents guaranteed cost control of polynomial fuzzy systems via a sum of squares (SOS) approach. First, we present a polynomial fuzzy model and controller that are more general representation of the well-known Takagi-Sugeno (T-S) fuzzy model and controller, respectively. Secondly, we derive a guaranteed cost control design condition based on polynomial Lyapunov functions. Hence, the design approach discussed in this paper is more general than the existing LMI approaches (to T-S fuzzy control system designs) based on quadratic Lyapunov functions. The design condition realizes guaranteed cost control by minimizing the upper bound of a given performance function. In addition, the design condition in the proposed approach can be represented in terms of SOS and is numerically (partially symbolically) solved via the recent developed SOSTOOLS. To illustrate the validity of the design approach, two design examples are provided. The first example deals with a complicated nonlinear system. The second example presents micro helicopter control. Both the examples show that our approach provides more extensive design results for the existing LMI approach.**

***Index Terms*—polynomial fuzzy control system, guaranteed cost control, sum of squares, polynomial Lyapunov function, stability.**

I. INTRODUCTION

THE Takagi-Sugeno (T-S) fuzzy model-based control methodology [1] has received a great deal of attention over the last two decades [2]-[6]. There is no loss of generality in adopting the T-S fuzzy model based control design framework as it has been established that any smooth nonlinear control systems can be approximated by the T-S fuzzy models (with liner model consequence) [7], [8]. Recently, we presented a sum of squares (SOS) approach [10], [11] to stability and stabilizability of polynomial fuzzy systems. This is a completely different approach from the existing LMI approaches [1], [9]. To the best of our knowledge, the paper [10] presented the first attempt at applying an SOS to fuzzy systems. Our SOS approach [10], [11] provided more extensive results for the existing LMI approaches to T-S fuzzy model and control.

This paper presents guaranteed cost control of polynomial fuzzy systems via a sum of squares (SOS) approach. First, we present a polynomial fuzzy model and controller that are more general representation of the well-known T-S fuzzy model and controller, respectively. Secondly, we derive a guaranteed cost control design condition based on polynomial Lyapunov functions. Hence, the design approach discussed in this paper is more general than the existing LMI approaches (to T-S fuzzy control system designs) based on quadratic Lyapunov functions. The design condition realizes guaranteed cost control by minimizing the upper bound of a given performance function. In addition, the design condition in the proposed approach can be represented in terms of SOS and is numerically (partially symbolically) solved via the recent developed SOSTOOLS [12]. To illustrate the validity of the design approach, two design examples are provided. The first example deals with a complicated nonlinear system. For this nonlinear system, any globally stabilizing T-S fuzzy controllers can not be designed via the existing LMI approach. The second example presents micro helicopter control from the application points of view. Even for the helicopter dynamics represented by a Takagi-Sugeno fuzzy model, we will show that the SOS control approach is better than the existing LMI approach. Both the examples show that our approach provides more extensive design results for the existing LMI approach.

II. GUARANTEED COST CONTROL

In [10], we proposed a new type of fuzzy model with polynomial model consequence, i.e., fuzzy model whose consequent parts are represented by polynomials. First, we briefly summarize the polynomial fuzzy model and controller.

It is well known that stability conditions for the T-S fuzzy system and the quadratic Lyapunov function reduce to LMIs, e.g., [1]. Hence, the stability conditions can be solved numerically and efficiently by interior point algorithms, e.g., by the Robust Control Toolbox of MATLAB[1]. On the other hand, stability [10] and stabilization conditions [11] for polynomial fuzzy systems and polynomial Lyapunov functions reduce to SOS problems. Clearly, the problem is never solved by LMI solvers and can be solved via SOSTOOLS [12]. Thus, SOS can be regarded as an extensive representation of LMIs.

The computational method used in this paper relies on the sum of squares decomposition of multivariate polynomials. A multivariate polynomial $f(\boldsymbol{x}(t))$ (where $\boldsymbol{x}(t) \in R^n$) is a sum of squares (SOS, for brevity) if there exist polynomials $f_1(\boldsymbol{x}(t)), \cdots, f_k(\boldsymbol{x}(t))$ such that $f(\boldsymbol{x}(t)) = \sum_{i=1}^{k} f_i^2(\boldsymbol{x}(t))$. It is clear that $f(\boldsymbol{x}(t))$ being an SOS naturally implies $f(\boldsymbol{x}(t)) \geq 0$ for all $\boldsymbol{x}(t) \in R^n$. For more details for SOS, see [10], [11]. A monomial in $\boldsymbol{x}(t)$ is a function of the form $x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$,

Manuscript received April 20, 2007; revised November 18, 2007. This work was supported in part by a Grant-in-Aid for Scientific Research (C) 18560244 from the Ministry of Education, Science and Culture of Japan.

Kazuo Tanaka and Hiroshi Ohtake are with the Department of Mechanical Engineering and Intelligent Systems, The University of Electro-Communications, Chofu, Tokyo 182-8585 Japan (email: ktanaka@mce.uec.ac.jp; hohtake@mce.uec.ac.jp).

Hua O. Wang is with the Department of Aerospace and Mechanical Engineering, Boston University, Boston, MA 02215 USA (email: wangh@bu.edu).

[1] A registered trademark of MathWorks, Inc.

where α_1, α_2, $\cdots$, α_n are nonnegative integers. In this case, the degree of the monomial is given by $\alpha_1 + \alpha_2 + \cdots + \alpha_n$.

A. Polynomial fuzzy model and controller

Consider the following nonlinear system:

$$\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t)), \tag{1}$$

where f is a nonlinear function. $\boldsymbol{x}(t) = [x_1(t) \quad x_2(t) \quad \cdots \quad x_n(t)]^T$ is the state vector and $\boldsymbol{u}(t) = [u_1(t)\ u_2(t)\ \cdots\ u_m(t)]^T$ is the input vector.

A polynomial fuzzy model has been proposed in [10]. Using the sector nonlinearity concept, we exactly represent (1) with the following polynomial fuzzy model (2). The main difference between the T-S fuzzy model [13] and the polynomial fuzzy model is consequent part representation. The fuzzy model (2) has a polynomial model consequence.

Model Rule i:

$$\begin{aligned} &If\ z_1(t)\ is\ M_{i1}\ and \cdots\ and\ \ z_p(t)\ is\ M_{ip} \\ &then\ \dot{\boldsymbol{x}}(t) = \boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t), \end{aligned} \tag{2}$$

where $i = 1, 2, \cdots, r$. $z_j(t)$ $(j = 1, 2, \cdots, p)$ is the premise variable. The membership function associated with the ith $Model\ Rule$ and jth premise variable component is denoted by M_{ij}. r denotes the number of $Model\ Rules$. Each $z_j(t)$ is a measurable time-varying quantity that may be states, measurable external variables and/or time. $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$ is a column vector whose entries are all monomials in $\boldsymbol{x}(t)$. That is, $\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) \in \boldsymbol{R}^N$ is an $N \times 1$ vector of monomials in $\boldsymbol{x}(t)$. $\boldsymbol{A}_i(\boldsymbol{x}(t)) \in \boldsymbol{R}^{n \times N}$ and $\boldsymbol{B}_i(\boldsymbol{x}(t)) \in \boldsymbol{R}^{n \times m}$ are polynomial matrices in $\boldsymbol{x}(t)$. Therefore, $\boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t)$ is a polynomial vector. Thus, the polynomial fuzzy model (2) has a polynomial in each consequent part. The details of $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$ is given in Proposition 1 of [11]. We assume that $\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) = 0$ iff $\boldsymbol{x}(t) = 0$ throughout this paper.

The defuzzification process of the model (2) can be represented as

$$\dot{\boldsymbol{x}}(t) = \sum_{i=1}^{r} h_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t)\}, \tag{3}$$

where

$$h_i(\boldsymbol{z}(t)) = \frac{\prod_{j=1}^{p} M_{ij}(z_j(t))}{\sum_{k=1}^{r}\prod_{j=1}^{p} M_{kj}(z_j(t))}.$$

It should be noted from the properties of membership functions that $h_i(\boldsymbol{z}(t)) \geq 0$ for all i and $\sum_{i=1}^{r} h_i(\boldsymbol{z}(t)) = 1$. Thus, the overall fuzzy model is achieved by fuzzy blending of the polynomial system models. A stability condition for the polynomial fuzzy systems without the inputs (i.e., $\boldsymbol{u}(t) = 0$) was derived in [10].

Since the parallel distributed compensation (PDC) mirrors the structure of the fuzzy model of a system, a fuzzy controller with polynomial rule consequence can be constructed from the given polynomial fuzzy model (2).

Control Rule i:

$$\begin{aligned} &If\ z_1(t)\ is\ M_{i1}\ and \cdots\ and\ \ z_p(t)\ is\ M_{ip} \\ &then\ \boldsymbol{u}(t) = -\boldsymbol{F}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) \quad i = 1, 2, \cdots, r \end{aligned} \tag{4}$$

The overall fuzzy controller can be calculated by

$$\boldsymbol{u}(t) = -\sum_{i=1}^{r} h_i(\boldsymbol{z}(t))\boldsymbol{F}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)). \tag{5}$$

If $\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) = \boldsymbol{x}(t)$ and $\boldsymbol{A}_i(\boldsymbol{x}(t))$, $\boldsymbol{B}_i(\boldsymbol{x}(t))$ and $\boldsymbol{F}_j(\boldsymbol{x}(t))$ are constant matrices for all i and j, then (3) and (5) reduce to the Takagi-Sugeno fuzzy model and controller, respectively. Therefore, (3) and (5) are more general representation.

From (3) and (5), the closed-loop system can be represented as

$$\begin{aligned} \dot{\boldsymbol{x}}(t) \quad &= \quad \sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z}(t))h_j(\boldsymbol{z}(t)) \\ &\times \quad \{\boldsymbol{A}_i(\boldsymbol{x}(t)) - \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{F}_j(\boldsymbol{x}(t))\}\hat{\boldsymbol{x}}(\boldsymbol{x}(t)). \end{aligned} \tag{6}$$

A stable controller design consisting of (3) and (5) was discussed in [11].

Remark 1: As shown in [10], [11], the number of rules in polynomial fuzzy model generally becomes fewer than that in T-S fuzzy model, and our SOS approach to polynomial fuzzy models provides much more relaxed stability and stabilization results than the existing LMI approaches to T-S fuzzy model and control. These facts will be found in Section III.

B. Guaranteed Cost Control via SOS

This subsection gives a guaranteed cost control design condition whose feasibility can be checked via SOSTOOLS (not via LMI solvers). Hence the fuzzy controller design with polynomial rule consequence is numerically a feasibility problem. From now, to lighten the notation, we will drop the notation with respect to time t. For instance, we will employ $\boldsymbol{x}$, $\hat{\boldsymbol{x}}(\boldsymbol{x})$ instead of $\boldsymbol{x}(t)$, $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$, respectively. Thus, we drop the notation with respect to time t, but it should be kept in mind that $\boldsymbol{x}$ means $\boldsymbol{x}(t)$. In addition, we will employ $\hat{\boldsymbol{x}}$ instead of $\hat{\boldsymbol{x}}(\boldsymbol{x})$. It should be also kept in mind that $\hat{\boldsymbol{x}}$ means $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$. Let $\boldsymbol{A}_i^k(\boldsymbol{x})$ denotes the k-th row of $\boldsymbol{A}_i(\boldsymbol{x})$, $\boldsymbol{K} = \{k_1, k_2, \cdots k_m\}$ denote the row indicies of $\boldsymbol{B}_i(\boldsymbol{x})$ whose corresponding row is equal to zero, and define $\tilde{\boldsymbol{x}} = (x_{k_1}, x_{k_2}, \cdots, x_{k_m})$.

To obtain more relaxed stability results, we employ a polynomial Lyapunov function [10] represented by $\hat{\boldsymbol{x}}^T\boldsymbol{P}(\tilde{\boldsymbol{x}})\hat{\boldsymbol{x}}$, where $\boldsymbol{P}(\tilde{\boldsymbol{x}})$ is a polynomial matrix in $\boldsymbol{x}$. If $\hat{\boldsymbol{x}} = \boldsymbol{x}$ and $\boldsymbol{P}(\tilde{\boldsymbol{x}})$ is a constant matrix, then the polynomial Lyapunov function reduces to the quadratic Lyapunov function $\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}$. Therefore, the polynomial Lyapunov function is a more general representation.

Next, we define the outputs for the polynomial fuzzy model (3) as

$$\boldsymbol{y} = \sum_{i=1}^{r} h_i(\boldsymbol{z})\boldsymbol{C}_i(\boldsymbol{x})\hat{\boldsymbol{x}}, \tag{7}$$

where $\boldsymbol{C}_i(\boldsymbol{x})$ are also polynomial matrices. We also consider the following performance function to be optimized.

$$J = \int_0^{\infty} \hat{\boldsymbol{y}}^T \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix} \hat{\boldsymbol{y}} dt, \tag{8}$$

where

$$\hat{\boldsymbol{y}} = \sum_{i=1}^{r} h_i(\boldsymbol{z}) \begin{bmatrix} \boldsymbol{C}_i(\boldsymbol{x}) \\ -\boldsymbol{F}_i(\boldsymbol{x}) \end{bmatrix} \hat{\boldsymbol{x}}. \tag{9}$$

$\boldsymbol{Q}$ and $\boldsymbol{R}$ are positive definite matrices. Theorem 1 provides the SOS design condition that minimizes the upper bound of the given performance function (8).

Theorem 1: If there exist a symmetric polynomial matrix $\boldsymbol{X}(\tilde{\boldsymbol{x}}) \in \boldsymbol{R}^{N\times N}$ and a polynomial matrix $\boldsymbol{M}_i(\boldsymbol{x}) \in \boldsymbol{R}^{m\times N}$ such that (10), (11), (12) and (13) hold, the guaranteed cost controller that minimizes the upper bound of the given performance function (8) can be designed as $\boldsymbol{F}_i(\boldsymbol{x}) = \boldsymbol{M}_i(\boldsymbol{x})\boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}})$.

$$\underset{X(\tilde{x}),M_i(x)}{minimize} \quad \lambda$$

$$subject\ to$$

$$\boldsymbol{v}_1^T(\boldsymbol{X}(\tilde{\boldsymbol{x}}) - \epsilon_1(\boldsymbol{x})\boldsymbol{I})\boldsymbol{v}_1 \ is\ SOS \tag{10}$$

$$\boldsymbol{v}_2^T \begin{bmatrix} \lambda & \hat{\boldsymbol{x}}^T(\boldsymbol{0}) \\ \hat{\boldsymbol{x}}(\boldsymbol{0}) & \boldsymbol{X}(\tilde{\boldsymbol{x}}(\boldsymbol{0})) \end{bmatrix} \boldsymbol{v}_2 \ is\ SOS \tag{11}$$

$$-\boldsymbol{v}_3^T \begin{bmatrix} \boldsymbol{N}_{ii}(\boldsymbol{x}) + \epsilon_{2ii}(\boldsymbol{x})\boldsymbol{I} & \boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{C}_i^T(\boldsymbol{x}) & -\boldsymbol{M}_i^T(\boldsymbol{x}) \\ \boldsymbol{C}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) & -\boldsymbol{Q}^{-1} & \boldsymbol{0} \\ -\boldsymbol{M}_i(\boldsymbol{x}) & \boldsymbol{0} & -\boldsymbol{R}^{-1} \end{bmatrix} \boldsymbol{v}_3 \ is\ SOS, \tag{12}$$

$$-\boldsymbol{v}_4^T \begin{bmatrix} \boldsymbol{N}_{ij}(\boldsymbol{x}) + \boldsymbol{N}_{ji}(\boldsymbol{x}) & \begin{pmatrix} \boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{C}_i^T(\boldsymbol{x}) \\ +\boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{C}_j^T(\boldsymbol{x}) \end{pmatrix} & -\boldsymbol{M}_i^T(\boldsymbol{x}) - \boldsymbol{M}_j^T(\boldsymbol{x}) \\ \begin{pmatrix} \boldsymbol{C}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) \\ +\boldsymbol{C}_j(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) \end{pmatrix} & -2\boldsymbol{Q}^{-1} & \boldsymbol{0} \\ -\boldsymbol{M}_i(\boldsymbol{x}) - \boldsymbol{M}_j(\boldsymbol{x}) & \boldsymbol{0} & -2\boldsymbol{R}^{-1} \end{bmatrix} \boldsymbol{v}_4 \quad is\ SOS, \quad i < j, \tag{13}$$

where

$$\begin{aligned} &\boldsymbol{N}_{ij}(\boldsymbol{x}) = \\ &\boldsymbol{T}(\boldsymbol{x})\boldsymbol{A}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) - \boldsymbol{T}(\boldsymbol{x})\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{M}_j(\boldsymbol{x}) \\ &+ \boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{A}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x}) - \boldsymbol{M}_j^T(\boldsymbol{x})\boldsymbol{B}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x}) \\ &- \sum_{k\in \boldsymbol{K}} \frac{\partial \boldsymbol{X}(\tilde{\boldsymbol{x}})}{\partial x_k} \boldsymbol{A}_i^k(\boldsymbol{x})\hat{\boldsymbol{x}}. \end{aligned} \tag{14}$$

$\boldsymbol{v}_1$, $\boldsymbol{v}_2$, $\boldsymbol{v}_3$ and $\boldsymbol{v}_4$ are vectors that are independent of $\boldsymbol{x}$. $\boldsymbol{T}(\boldsymbol{x}) \in \boldsymbol{R}^{N\times n}$ is a polynomial matrix whose (i, j)-th entry is given by $T^{ij}(\boldsymbol{x}) = \frac{\partial \hat{x}_i}{\partial x_j}(\boldsymbol{x})$. $\epsilon_1(\boldsymbol{x}) > 0$ and $\epsilon_{2ii}(\boldsymbol{x}) > 0$ at $\boldsymbol{x} \neq \boldsymbol{0}$, and $\epsilon_1(\boldsymbol{x}) = 0$ and $\epsilon_{2ii}(\boldsymbol{x}) = 0$ at $\boldsymbol{x} = \boldsymbol{0}$.

(proof) If (10) is satisfied for $\epsilon_1(\boldsymbol{x}) > 0$ at $\boldsymbol{x} \neq \boldsymbol{0}$ and $\epsilon_1(\boldsymbol{x}) = 0$ at $\boldsymbol{x} = \boldsymbol{0}$, then $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ is a positive definite polynomial matrix. Next, consider a candidate of polynomial Lyapunov function $V(\boldsymbol{x}) = \hat{\boldsymbol{x}}^T\boldsymbol{P}(\tilde{\boldsymbol{x}})\hat{\boldsymbol{x}}$, where $\boldsymbol{P}(\tilde{\boldsymbol{x}}) = \boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}})$. If (10) is satisfied, then it is clear that $V(\boldsymbol{x}) > 0$ at $\boldsymbol{x} \neq 0$.

By noting that $\dot{x}_k = \sum_{i=1}^{r} h_i(\boldsymbol{z})\boldsymbol{A}_i^k(\boldsymbol{x})\hat{\boldsymbol{x}}$, the time derivative of the Lyapunov function $V(\boldsymbol{x})$ along the trajectory of (6) becomes as follows:

$$\begin{aligned} \dot{V}(\boldsymbol{x}) &= \hat{\boldsymbol{x}}^T\boldsymbol{P}(\tilde{\boldsymbol{x}})\dot{\hat{\boldsymbol{x}}} + \dot{\hat{\boldsymbol{x}}}^T\boldsymbol{P}(\tilde{\boldsymbol{x}})\hat{\boldsymbol{x}} + \hat{\boldsymbol{x}}^T\dot{\boldsymbol{P}}(\tilde{\boldsymbol{x}})\hat{\boldsymbol{x}} \\ &= \hat{\boldsymbol{x}}^T\boldsymbol{P}(\tilde{\boldsymbol{x}})\boldsymbol{T}(\boldsymbol{x})\dot{\boldsymbol{x}} + \dot{\boldsymbol{x}}^T\boldsymbol{T}^T(\boldsymbol{x})\boldsymbol{P}(\tilde{\boldsymbol{x}})\hat{\boldsymbol{x}} \\ &\quad + \hat{\boldsymbol{x}}^T\left(\sum_{k=1}^{n} \frac{\partial \boldsymbol{P}(\tilde{\boldsymbol{x}})}{\partial x_k}\dot{x}_k\right)\hat{\boldsymbol{x}} \\ &= \sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z}) \\ &\quad \times \hat{\boldsymbol{x}}^T\Bigg(\boldsymbol{P}(\tilde{\boldsymbol{x}})\boldsymbol{T}(\boldsymbol{x})\{\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\} \\ &\quad + \{\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\}^T\boldsymbol{T}^T(\boldsymbol{x})\boldsymbol{P}(\tilde{\boldsymbol{x}}) \\ &\quad + \sum_{k\in\boldsymbol{K}} \frac{\partial \boldsymbol{P}(\tilde{\boldsymbol{x}})}{\partial x_k}\boldsymbol{A}_i^k(\boldsymbol{x})\hat{\boldsymbol{x}}\Bigg)\hat{\boldsymbol{x}} \\ &= \sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\hat{\boldsymbol{x}}^T U_{ij}(\boldsymbol{x})\hat{\boldsymbol{x}} \end{aligned} \tag{15}$$

where

$$\begin{aligned} \boldsymbol{U}_{ij}(\boldsymbol{x}) &= \boldsymbol{P}(\tilde{\boldsymbol{x}})\boldsymbol{T}(\boldsymbol{x})\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{P}(\tilde{\boldsymbol{x}})\boldsymbol{T}(\boldsymbol{x})\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}) \\ &+ \boldsymbol{A}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x})\boldsymbol{P}(\tilde{\boldsymbol{x}}) - \boldsymbol{F}_j^T(\boldsymbol{x})\boldsymbol{B}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x})\boldsymbol{P}(\tilde{\boldsymbol{x}}) \\ &+ \sum_{k\in\boldsymbol{K}} \frac{\partial \boldsymbol{P}(\tilde{\boldsymbol{x}})}{\partial x_k}\boldsymbol{A}_i^k(\boldsymbol{x})\hat{\boldsymbol{x}}. \end{aligned} \tag{16}$$

Next, we assume that there exists a positive definite polynomial matrix $\boldsymbol{P}(\tilde{\boldsymbol{x}})$ satisfying (17).

$$\begin{aligned} &\sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\hat{\boldsymbol{x}}^T\boldsymbol{U}_{ij}(\boldsymbol{x})\hat{\boldsymbol{x}} \\ &+ \hat{\boldsymbol{x}}^T\left(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix} \boldsymbol{C}_i(\boldsymbol{x}) \\ -\boldsymbol{F}_i(\boldsymbol{x}) \end{bmatrix}\right)^T \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix} \\ &\quad \times \left(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix} \boldsymbol{C}_i(\boldsymbol{x}) \\ -\boldsymbol{F}_i(\boldsymbol{x}) \end{bmatrix}\right)\hat{\boldsymbol{x}} < 0 \end{aligned} \tag{17}$$

Then, $\dot{V}(\boldsymbol{x}) < 0$ at $\boldsymbol{x} \neq \boldsymbol{0}$ since

$$\begin{aligned} &\hat{\boldsymbol{x}}^T\left(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix} \boldsymbol{C}_i(\boldsymbol{x}) \\ -\boldsymbol{F}_i(\boldsymbol{x}) \end{bmatrix}\right)^T \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix} \\ &\quad \times \left(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix} \boldsymbol{C}_i(\boldsymbol{x}) \\ -\boldsymbol{F}_i(\boldsymbol{x}) \end{bmatrix}\right)\hat{\boldsymbol{x}} \geq 0. \end{aligned}$$

In other words, the closed loop system (6) is stable if (10) and (17) are satisfied. We will show that (17) holds if (12) and (13) are satisfied later.

We note that (17) is equivalent to the following condition:

$$\hat{\boldsymbol{y}}^T \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix} \hat{\boldsymbol{y}} < -\dot{V}(\boldsymbol{x}) \tag{18}$$

By integrating (18) from 0 to ∞, we have

$$\begin{aligned} J &= \int_0^\infty \hat{\boldsymbol{y}}^T \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix} \hat{\boldsymbol{y}}dt \\ &< -V(\boldsymbol{x})|_0^\infty = -\hat{\boldsymbol{x}}^T\boldsymbol{P}(\tilde{\boldsymbol{x}})\hat{\boldsymbol{x}}|_0^\infty. \end{aligned} \tag{19}$$

Since the closed loop system (6) is stable (if both (10) and (17) hold), $\hat{\boldsymbol{x}} \to \boldsymbol{0}$ at $t \to \infty$. Hence, (19) becomes

$$J < \hat{\boldsymbol{x}}^T(\boldsymbol{0})\boldsymbol{P}(\tilde{\boldsymbol{x}}(\boldsymbol{0}))\hat{\boldsymbol{x}}(\boldsymbol{0}). \tag{20}$$

Here we consider the following relation.

$$J < \hat{\boldsymbol{x}}^T(\boldsymbol{0})\boldsymbol{P}(\tilde{\boldsymbol{x}}(\boldsymbol{0}))\hat{\boldsymbol{x}}(\boldsymbol{0}) \le \lambda \tag{21}$$

From Schur complements, the above inequality can be rewri-iten as

$$\begin{bmatrix} \lambda & \hat{\boldsymbol{x}}^T(\boldsymbol{0}) \\ \hat{\boldsymbol{x}}(\boldsymbol{0}) & \boldsymbol{X}(\tilde{\boldsymbol{x}}(\boldsymbol{0})) \end{bmatrix} \ge \boldsymbol{0}. \tag{22}$$

If (11) is satisfied, then (22) holds. Hence, we can design the guaranteed cost controller (that minimizes the upper bound of J) by minimizing λ under the guarantee of (10), (11) and (17).

Next, we show that (17) holds if the SOS conditions (12) and (13) are satisfied. If (12) and (13) hold, then we have

$$-\sum_{i=1}^{r} h_i^2(\boldsymbol{z})\Big(\boldsymbol{W}_{ii}(\boldsymbol{x}) + \boldsymbol{E}_{ii}(\boldsymbol{x})\Big) > \boldsymbol{0}, \tag{23}$$

$$-\sum_{i=1}^{r}\sum_{i<j}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})(\boldsymbol{W}_{ij}(\boldsymbol{x}) + \boldsymbol{W}_{ji}(\boldsymbol{x})) \ge \boldsymbol{0},\ i<j, \tag{24}$$

where

$$\boldsymbol{W}_{ij}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{N}_{ij}(\boldsymbol{x}) & \boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{C}_i^T(\boldsymbol{x}) & -\boldsymbol{M}_i^T(\boldsymbol{x}) \\ \boldsymbol{C}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) & -\boldsymbol{Q}^{-1} & \boldsymbol{0} \\ -\boldsymbol{M}_i(\boldsymbol{x}) & \boldsymbol{0} & -\boldsymbol{R}^{-1} \end{bmatrix},$$

$$\boldsymbol{E}_{ii}(\boldsymbol{x}) = \begin{bmatrix} \epsilon_{2ii}(\boldsymbol{x})\boldsymbol{I} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \end{bmatrix}.$$

The inequalities (23) and (24) imply

$$\begin{aligned} &-\sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\boldsymbol{W}_{ii}(\boldsymbol{x}) \\ &-2\sum_{i=1}^{r}\sum_{i<j}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\left(\frac{\boldsymbol{W}_{ij}(\boldsymbol{x}) + \boldsymbol{W}_{ji}(\boldsymbol{x})}{2}\right) \\ &= -\sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\boldsymbol{W}_{ij}(\boldsymbol{x}) > \boldsymbol{0}. \end{aligned} \tag{25}$$

Using Schur complements,

$$\sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\boldsymbol{W}_{ij}(\boldsymbol{x}) < \boldsymbol{0}$$

can be converted into

$$\begin{aligned} &\sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\boldsymbol{N}_{ij}(\boldsymbol{x}) \\ &+ \Big(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix}\boldsymbol{C}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) \\ -\boldsymbol{M}_i(\boldsymbol{x})\end{bmatrix}\Big)^T \begin{bmatrix}\boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R}\end{bmatrix} \\ &\times \Big(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix}\boldsymbol{C}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) \\ -\boldsymbol{M}_i(\boldsymbol{x})\end{bmatrix}\Big) < \boldsymbol{0}. \end{aligned} \tag{26}$$

We note that $\boldsymbol{M}_i(\boldsymbol{x}) = \boldsymbol{F}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}})$.

Again, recall (14).

$$\begin{aligned} &\boldsymbol{N}_{ij}(\boldsymbol{x}) \\ =&\boldsymbol{T}(\boldsymbol{x})\boldsymbol{A}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) - \boldsymbol{T}(\boldsymbol{x})\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{M}_j(\boldsymbol{x}) \\ &+ \boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{A}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x}) - \boldsymbol{M}_j^T(\boldsymbol{x})\boldsymbol{B}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x}) \\ &- \sum_{k\in\boldsymbol{K}} \frac{\partial \boldsymbol{X}(\tilde{\boldsymbol{x}})}{\partial x_k}\boldsymbol{A}_i^k(\boldsymbol{x})\hat{\boldsymbol{x}} \end{aligned} \tag{27}$$

We rewrite the fifth term of (27). Since $\boldsymbol{P}(\tilde{\boldsymbol{x}})\boldsymbol{X}(\tilde{\boldsymbol{x}}) = \boldsymbol{I}$, we first partially differentiate it with respect to x_k.

$$\frac{\partial \boldsymbol{P}(\tilde{\boldsymbol{x}})}{\partial x_k}\boldsymbol{X}(\tilde{\boldsymbol{x}}) + \boldsymbol{P}(\tilde{\boldsymbol{x}})\frac{\partial \boldsymbol{X}(\tilde{\boldsymbol{x}})}{\partial x_k} = \boldsymbol{0} \tag{28}$$

Hence, we have the following equation.

$$-\frac{\partial \boldsymbol{X}(\tilde{\boldsymbol{x}})}{\partial x_k} = \boldsymbol{X}(\tilde{\boldsymbol{x}})\frac{\partial \boldsymbol{P}(\tilde{\boldsymbol{x}})}{\partial x_k}\boldsymbol{X}(\tilde{\boldsymbol{x}}) \tag{29}$$

Therefore, (27) can be rewritten as

$$\begin{aligned} &\boldsymbol{N}_{ij}(\boldsymbol{x}) \\ =&\boldsymbol{T}(\boldsymbol{x})\boldsymbol{A}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) - \boldsymbol{T}(\boldsymbol{x})\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{M}_j(\boldsymbol{x}) \\ &+ \boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{A}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x}) - \boldsymbol{M}_j^T(\boldsymbol{x})\boldsymbol{B}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\boldsymbol{x}) \\ &+ \sum_{k\in\boldsymbol{K}} \boldsymbol{X}(\tilde{\boldsymbol{x}})\frac{\partial \boldsymbol{P}(\tilde{\boldsymbol{x}})}{\partial x_k}\boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{A}_i^k(\boldsymbol{x})\hat{\boldsymbol{x}} \\ =&\boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{U}_{ij}(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}). \end{aligned} \tag{30}$$

Multiplying both side of (26) by $\boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}})$ gives

$$\begin{aligned} &\sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z})h_j(\boldsymbol{z})\boldsymbol{U}_{ij}(\boldsymbol{x}) \\ &+ \left(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix}\boldsymbol{C}_i(\boldsymbol{x}) \\ -\boldsymbol{F}_i(\boldsymbol{x})\end{bmatrix}\right)^T \\ &\times \begin{bmatrix}\boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R}\end{bmatrix}\left(\sum_{i=1}^{r} h_i(\boldsymbol{z})\begin{bmatrix}\boldsymbol{C}_i(\boldsymbol{x}) \\ -\boldsymbol{F}_i(\boldsymbol{x})\end{bmatrix}\right) < \boldsymbol{0}. \end{aligned} \tag{31}$$

Thus (17) holds if the SOS conditions (12) and (13) are satisfied.

(Q.E.D.)

Remark 2: Currently, sum of squares programs (SOSPs) are solved by reformulating them as semidefinite programs (SDPs). SOSTOOLS automates the conversion from SOSP to SDP and the SDP can be solved by a SDP solver [12]. At present, SOOSTOOLS uses other free MATLAB add-ons such as SeDuMi [14] or SDPT3 [15] as the SDP solver. In this paper, we numerically find $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ satisfying the SOS condition in Theorem 1 via SeDuMi in addition to SOSTOOLS. For more details of how to solve the SDPs using SeDuMi, see [12], [14].

Remark 3: Note that $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, $\boldsymbol{v}_3$ and $\boldsymbol{v}_4$ are vectors that are independent of $\boldsymbol{x}$, because $\boldsymbol{L}(\boldsymbol{x})$ is not always a positive semi-definite matrix for all $\boldsymbol{x}$ even if $\hat{\boldsymbol{x}}^T\boldsymbol{L}(\boldsymbol{x})\hat{\boldsymbol{x}}$ is an SOS, where $\boldsymbol{L}(\boldsymbol{x})$ is a symmetric polynomial matrix in $\boldsymbol{x}$. However, it is guaranteed from Proposition 2 in [11] that if $\boldsymbol{v}^T\boldsymbol{L}(\boldsymbol{x})\boldsymbol{v}$ is an SOS, then $\boldsymbol{L}(\boldsymbol{x}) \ge 0$ for all $\boldsymbol{x}$.

Remark 4: To avoid introducing non-convex condition, we assume that $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ only depends on states $\tilde{\boldsymbol{x}}$ whose dynamics is not directly affected by the control input, namely states whose corresponding rows in $\boldsymbol{B}_i(\boldsymbol{x})$ are zero. In relation to this, it may be advantageous to employ an initial state transformation to introduce as many zero rows as possible in $\boldsymbol{B}_i(\boldsymbol{x})$.

III. Design Examples

To illustrate the validity of the design approach, this section provides two design examples. The first example deals with a complicated nonlinear system. For this nonlinear system, any globally stabilizing T-S fuzzy controllers can not be designed via the existing LMI approach. The second example presents micro helicopter control from the application points of view. Even for the helicopter dynamics represented by a Takagi-Sugeno fuzzy model, we will show that the SOS control approach is better than the existing LMI approach.

A. Complicated Nonlinear System

Consider the following nonlinear system [11]:

$$\begin{aligned}\dot{x}_1 &= -x_1 + x_1^2 + x_1^3 + x_1^2 x_2 - x_1 x_2^2 + x_2 + x_1 u,\\ \dot{x}_2 &= -\sin x_1 - x_2.\end{aligned} \tag{32}$$

The nonlinear system is unstable. Based on the concept of sector nonlinearity [1], the nonlinear system can be exactly represented by a Takagi-Sugeno fuzzy model for $x_1 \in [-d_1 \ d_1]$ and $x_2 \in [-d_2 \ d_2]$, where d_1 and d_2 are constants satisfying $0 < d_1 < \infty$ and $0 < d_2 < \infty$.

The Takagi-Sugeno fuzzy model is obtained as

$$\dot{\boldsymbol{x}} = \sum_{i=1}^{8} h_i(\boldsymbol{z})\{\boldsymbol{A}_i \boldsymbol{x} + \boldsymbol{B}_i u\}, \tag{33}$$

where $\boldsymbol{x} = [x_1 \ x_2]^T$ and $\boldsymbol{z} = [x_1 \ x_2]^T$. $\boldsymbol{A}_i$, $\boldsymbol{B}_i$ matrices and the membership functions $h_i(\boldsymbol{z})$ $(i = 1, \cdots, 8)$ are given in [11].

For a large d_1, e.g., $d_1 > 0.9$, the following LMI stable design conditions [1] are unsolvable for the feedback system consisting of the Takagi-Sugeno fuzzy model (33) and the corresponding Takagi-Sugeno fuzzy controller.

$$\boldsymbol{X} > \boldsymbol{0} \tag{34}$$
$$-\boldsymbol{X}\boldsymbol{A}_i^T - \boldsymbol{A}_i\boldsymbol{X} + \boldsymbol{M}_i^T\boldsymbol{B}_i^T + \boldsymbol{B}_i\boldsymbol{M}_i > \boldsymbol{0} \tag{35}$$
$$\begin{aligned}&-\boldsymbol{X}\boldsymbol{A}_i^T - \boldsymbol{A}_i\boldsymbol{X} - \boldsymbol{X}\boldsymbol{A}_j^T - \boldsymbol{A}_j\boldsymbol{X}\\ &\quad+\boldsymbol{M}_j^T\boldsymbol{B}_i^T + \boldsymbol{B}_i\boldsymbol{M}_j + \boldsymbol{M}_i^T\boldsymbol{B}_j^T + \boldsymbol{B}_j\boldsymbol{M}_i \geq \boldsymbol{0}\\ &\qquad i < j\end{aligned} \tag{36}$$

where $\boldsymbol{M}_i = \boldsymbol{F}_i\boldsymbol{X}$. This means that LMI conditions (44) - (46) [1] for guaranteed cost control are also infeasible for the same large d_1. In addition, the Takagi-Sugeno fuzzy model has eight rules since the nonlinear system is complicated. We will see that the polynomial fuzzy system (that can exactly and globally represent the nonlinear system) has only two rules. On the other hand, we can have the following polynomial fuzzy system that can exactly represent the dynamics of the nonlinear system for $x_1 \in (-\infty \ \infty)$ and $x_2 \in (-\infty \ \infty)$.

$$\dot{\boldsymbol{x}} = \sum_{i=1}^{2} h_i(\boldsymbol{z})\{\boldsymbol{A}_i(\boldsymbol{x})\hat{\boldsymbol{x}} + \boldsymbol{B}_i(\boldsymbol{x})u\} \tag{37}$$

$$\boldsymbol{y} = \sum_{i=1}^{2} h_i(\boldsymbol{z})\boldsymbol{C}_i(\boldsymbol{x})\hat{\boldsymbol{x}} \tag{38}$$

where $\boldsymbol{x} = \hat{\boldsymbol{x}} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}$ and $z = x_1$,

$$\begin{aligned}\boldsymbol{A}_1(\boldsymbol{x}) &= \begin{bmatrix} -1 + x_1 + x_1^2 + x_1x_2 - x_2^2 & 1 \\ -1 & -1 \end{bmatrix},\\ \boldsymbol{A}_2(\boldsymbol{x}) &= \begin{bmatrix} -1 + x_1 + x_1^2 + x_1x_2 - x_2^2 & 1 \\ 0.2172 & -1 \end{bmatrix},\\ \boldsymbol{B}_1(\boldsymbol{x}) &= \begin{bmatrix} x_1 \\ 0 \end{bmatrix}, \boldsymbol{B}_2(\boldsymbol{x}) = \begin{bmatrix} x_1 \\ 0 \end{bmatrix},\\ \boldsymbol{C}_1(\boldsymbol{x}) &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \boldsymbol{C}_2(\boldsymbol{x}) = \begin{bmatrix} 1 & 0 \end{bmatrix}.\end{aligned}$$

The membership functions are given as

$$h_1(\boldsymbol{z}) = \frac{\sin x_1 + 0.2172x_1}{1.2172x_1}, \ \ h_2(\boldsymbol{z}) = \frac{x_1 - \sin x_1}{1.2172x_1}.$$

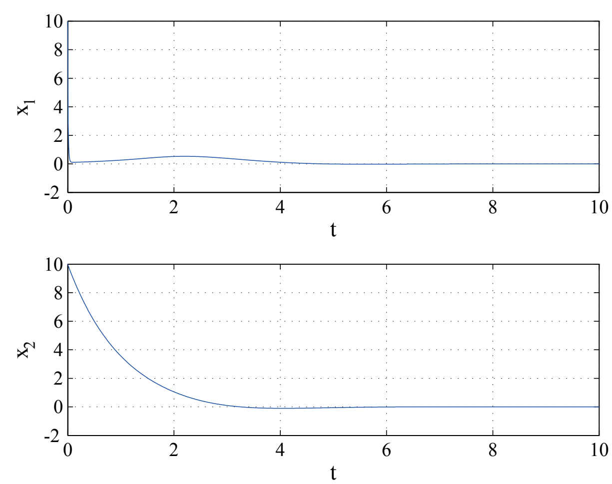

Fig. 1. Guaranteed cost control result.

The SOS design condition in Theorem 1 is feasible when the orders of both $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are not zero. Conversely, when the orders of both $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are zero, that is, when both $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are constant matrices instead of polynomial matrices in $\boldsymbol{x}$, the design condition in Theorem 1 reduces to the existing LMI design condition. In other words, when $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are constant matrices, the polynomial fuzzy controller reduces to the Takagi-Sugeno fuzzy controller. Only in this case, the SOS design condition in Theorem 1 is infeasible. This means that the polynomial fuzzy controller stabilizes globally and asymptotically the polynomial fuzzy system (37) although it may be difficult to stabilize globally and asymptotically the nonlinear system via Takagi-Sugeno fuzzy controllers.

The guaranteed cost controllers for $\boldsymbol{Q} = \boldsymbol{I}$, $\boldsymbol{R} = 1$ and $\boldsymbol{x}(0) = \begin{bmatrix} 10 & 10 \end{bmatrix}^T$ gives $J = 183.3$ when the orders of $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are 0 and 1. The stable polynomial fuzzy controller designed in [11] gives $J = 297.7$. Thus, the performance index value of the guaranteed cost control is better than that of the stable control. In addition, if the orders of $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are increased, the performance index value can be improved. For example, the guaranteed cost control gives $J = 144.0$ when the order of $\boldsymbol{M}_i(\boldsymbol{x})$ are 2. Furthermore, the guaranteed cost control gives $J = 131.0$ when the orders of $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are 2 and 3, respectively.

A main difference between the Takagi-Sugeno fuzzy model and the polynomial fuzzy model is that (37) can have x_1 and x_2 in the $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ matrices, i.e., that $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are permitted to be polynomial matrices in $\boldsymbol{x}$. Furthermore, our approach deals with a more general Lyapunov function (polynomial Lyapunov function). Thus, our approach provides more relaxed design results than the existing LMI approach. In addition, the polynomial fuzzy model (37) is an exact global model for the nonlinear system though the Takagi-Sugeno fuzzy model (33) is an (exact) semi-global model for the nonlinear system.

B. Micro Helicopter Control

A co-axial counter rotating helicopter dynamics can be written as

$$\dot{u}(t) = -\frac{a}{I_z}\psi(t)v(t) + \frac{1}{m}U_X(t), \tag{39}$$

$$\dot{v}(t) = \frac{a}{I_z}\psi(t)u(t) + \frac{1}{m}U_Y(t), \tag{40}$$

$$\dot{w}(t) = \frac{1}{m}U_Z(t), \tag{41}$$

under some assumptions [17], where $a = 1.5$, $m = 0.2$ and $I_z = 0.2857$. u, v and w denote velocities of x, y and z axis directions, respectively. ψ is angle around z axis. $U_X(t)$, $U_Y(t)$ and $U_Z(t)$ denote control input variables.

Based on the concept of sector nonlinearity [1], the nonlinear system can be exactly represented by a Takagi-Sugeno fuzzy model for $\psi(t) \in [-\pi\ \pi]$. The Takagi-Sugeno fuzzy model is obtained as

$$\dot{\boldsymbol{x}}(t) = \sum_{i=1}^{2} h_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i\boldsymbol{x}(t) + \boldsymbol{B}_i\boldsymbol{u}(t)\}, \tag{42}$$

$$\boldsymbol{y}(t) = \sum_{i=1}^{2} h_i(\boldsymbol{z}(t))\boldsymbol{C}_i\boldsymbol{x}(t), \tag{43}$$

where $\boldsymbol{z}(t) = \psi(t)$ and

$$\begin{aligned}\boldsymbol{x}(t) &= [u(t)\ v(t)\ w(t)\ e_x(t)\ e_y(t)\ e_z(t)]^T,\\ \boldsymbol{u}(t) &= [U_X(t)\ U_Y(t)\ U_Z(t)]^T.\end{aligned}$$

The elements $e_x(t)$, $e_y(t)$ and $e_z(t)$ are defined as $e_x(t) = x(t) - x_{ref}$, $e_y(t) = y(t) - y_{ref}$, $e_z(t) = z(t) - z_{ref}$, where x_{ref}, y_{ref} and z_{ref} are constant target positions. $\boldsymbol{A}_i$, $\boldsymbol{B}_i$ and $\boldsymbol{C}_i$ matrices and the membership functions are given as follows.

$$\boldsymbol{A}_1 = \begin{bmatrix} 0 & -\frac{a\pi}{I_Z} & 0 & 0 & 0 & 0 \\ \frac{a\pi}{I_Z} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix},$$

$$\boldsymbol{A}_2 = \begin{bmatrix} 0 & \frac{a\pi}{I_Z} & 0 & 0 & 0 & 0 \\ -\frac{a\pi}{I_Z} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix},$$

$$\boldsymbol{B}_1 = \boldsymbol{B}_2 = \begin{bmatrix} \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & \frac{1}{m} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\boldsymbol{C}_1 = \boldsymbol{C}_2 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$h_1(\psi(t)) = \frac{\psi(t) + \pi}{2\pi}\ ,\ h_2(\psi(t)) = \frac{\pi - \psi(t)}{2\pi}.$$

Note that the Takagi-Sugeno fuzzy model exactly represents the dynamics (39) - (41) for the range $\psi(t) \in [-\pi\ \pi]$.

Consider the performance index (8) again. We can find feedback gains that minimizes the upper bound of (8) by solving the following LMIs [1]. From the solutions $\boldsymbol{X}$ and $\boldsymbol{M}_i$, the feedback gains can be obtained as $\boldsymbol{F}_i = \boldsymbol{M}_i\boldsymbol{X}^{-1}$. Then, the controller satisfies $J < \boldsymbol{x}^T(0)\boldsymbol{X}\boldsymbol{x}(0) < \lambda$.

$$\underset{\boldsymbol{X},\boldsymbol{M}_i,}{minimize}\ \lambda$$
$$subject\ \ to$$

$$\boldsymbol{X} > \boldsymbol{0},\ \begin{bmatrix} \lambda & \boldsymbol{x}^T(0) \\ \boldsymbol{x}(0) & \boldsymbol{X} \end{bmatrix} > \boldsymbol{0}, \tag{44}$$

$$\hat{\boldsymbol{U}}_{ii} < \boldsymbol{0} \tag{45}$$

$$\hat{\boldsymbol{V}}_{ij} < \boldsymbol{0} \qquad i < j, \tag{46}$$

where

$$\hat{\boldsymbol{U}}_{ii} = \begin{bmatrix} \boldsymbol{H}_{ii} & \boldsymbol{X}\boldsymbol{C}_i^T & -\boldsymbol{M}_i^T \\ \boldsymbol{C}_i\boldsymbol{X} & -\boldsymbol{Q}^{-1} & \boldsymbol{0} \\ -\boldsymbol{M}_i & \boldsymbol{0} & -\boldsymbol{R}^{-1} \end{bmatrix},$$

$$\hat{\boldsymbol{V}}_{ij} = \begin{bmatrix} \boldsymbol{H}_{ij} + \boldsymbol{H}_{ji} & \boldsymbol{X}\boldsymbol{C}_i^T & -\boldsymbol{M}_j^T & \boldsymbol{X}\boldsymbol{C}_j^T & -\boldsymbol{M}_i^T \\ \boldsymbol{C}_i\boldsymbol{X} & -\boldsymbol{Q}^{-1} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ -\boldsymbol{M}_j & \boldsymbol{0} & -\boldsymbol{R}^{-1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{C}_j\boldsymbol{X} & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{Q}^{-1} & \boldsymbol{0} \\ -\boldsymbol{M}_i & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{R}^{-1} \end{bmatrix},$$

$$\boldsymbol{H}_{ij} = \boldsymbol{X}\boldsymbol{A}_i^T + \boldsymbol{A}_i\boldsymbol{X} - \boldsymbol{B}_i\boldsymbol{M}_j - \boldsymbol{M}_j^T\boldsymbol{B}_i^T.$$

The above LMI condition is feasible for this fuzzy model.

On the other hand, the SOS design condition in Theorem 1 is also feasible when the orders of $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are zero and two, respectively. We compare the LMI-based guaranteed cost controller (designed by solving the (44) - (46)) with the controller (designed by the SOS condition in Theorem 1), that is, with the SOS-based guaranteed cost controller. Table I shows comparison results of performance function values J for the LMI controllers and the SOS controllers, where the initial positions are $u(0) = 0$, $v(0) = 0$, $w(0) = 0$ $e_x(0) = -0.6$, $e_y(0) = -0.4$ and $e_z(0) = -1$. In Table I, Cases I, II and III denote three cases of selecting the weighting matrices $(\boldsymbol{Q}, \boldsymbol{R}) = (\boldsymbol{I}, 0.1\boldsymbol{I})$, $(\boldsymbol{Q}, \boldsymbol{R}) = (\boldsymbol{I}, \boldsymbol{I})$, and $(\boldsymbol{Q}, \boldsymbol{R}) = (\boldsymbol{I}, 10\boldsymbol{I})$, respectively.

TABLE I
COMPARISON OF PERFORMANCE FUNCTION VALUES J

	Case I	Case II	Case III
LMI controller	0.67286	1.5522	3.8873
SOS controller	0.57539	1.0388	2.3350
Reduction rate of J [%]	14.4859	33.0756	39.9326

It is found from Table I that the performance index values of the SOS based guaranteed cost control (Theorem 1) are better than those of the LMI based guaranteed cost control ((44) - (46)) in all the cases.

IV. CONCLUSIONS

This paper has presented guaranteed cost control of polynomial fuzzy systems via a sum of squares (SOS) approach. First, we have presented a polynomial fuzzy model and controller that are more general representation of the well-known Takagi-Sugeno (T-S) fuzzy model and controller, respectively. Secondly, we have derived a guaranteed cost control design condition based on polynomial Lyapunov functions. Hence, the design approach discussed in this paper is more general than the existing LMI approaches (to T-S fuzzy control system designs) based on quadratic Lyapunov functions. The design condition realizes guaranteed cost control by minimizing the upper bound of a given performance function. In addition, the design condition in the proposed approach can be represented in terms of SOS and is numerically (partially symbolically) solved via the recent developed SOSTOOLS. To illustrate the validity of the design approach, two design examples have been provided. Both the examples have shown that our approach provides more extensive design results for the existing LMI approach.

Our future works are to apply this approach to a real micro helicopter and to extend this approach to a variety of control techniques.

ACKNOWLEDGMENT

The authors would like to thank Mr. K. Yamauchi and Mr. T. Komatu, UEC, Japan, for their contribution to this research.

REFERENCES

[1] K. Tanaka and H. O. Wang: Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach, JOHN WILEY & SONS, INC, 2001.

[2] H.-N. Wu and H.-X. Li, “Finite-Dimensional Constrained Fuzzy Control for a Class of Nonlinear Distributed Process Systems”, IEEE Transactions on Systems, Man and Cybernetics Part B, Vo1. 37, No. 5, pp.1422-1430, Octber, 2007.

[3] W.-W. Lin, W.-J. Wang and S.-H. Yang, “A Novel Stabilization Criterion for Large-Scale T-S Fuzzy Systems”, IEEE Transactions on Systems, Man and Cybernetics Part B, Vo1. 37, No. 4, pp.1074-1079, August, 2007.

[4] P. Baranyi, “TP model transformation as a way to LMI based controller design”, IEEE Transaction on Industrial Electronics, vol. 51, no. 2, pp.387-400, April 2004.

[5] K. Tanaka, et al., “Switching Control of an R/C Hovercraft: Stabilization and Smooth Switching,”, IEEE Transactions on Systems, Man and Cybernetics, Part B, Vol.31, No.6, pp.853-863, Dec. 2001.

[6] C.-H. Sun and W.-J. Wang, “An improved stability criterion for T-S fuzzy discrete systems via vertex expression”, IEEE Transactions on Systems, Man and Cybernetics Part B, Vo1. 36, No. 3, pp.672-678, June, 2006.

[7] H. O. Wang, J. Li, D. Niemann and K. Tanaka, “T-S fuzzy Model with Linear Rule Consequence and PDC Controller: A Universal Framework for Nonlinear Control Systems”, 9th IEEE International Conference on Fuzzy Systems, San Antonio, May, 2000, pp.549-554.

[8] D. Tikk, P.Baranyi, R. J. Patton, “Approximation Properties of TP Model Forms and its Consequences to TPDC Design Framework”, Asian Journal of Control, Vol. 9, No. 3, pp. 221-231, Sept. 2007.

[9] G. Feng, “A Survey on Analysis and Design of Model-Based Fuzzy Control Systems”, IEEE Trans. on Fuzzy Systems, Vol.14, no.5, pp.676-697, Oct. 2006.

[10] K. Tanaka, H. Yoshida, H. Ohtake and H. O. Wang “A Sum of Squares Approach to Stability Analysis of Polynomial Fuzzy Systems”, 2007 American Control Conference, New York, July, 2007, pp.4071-4076.

[11] K. Tanaka, H. Yoshida, H. Ohtake and H. O. Wang, “ Stabilization of Polynomial Fuzzy Systems via a Sum of Squares Approach”, 2007 IEEE International Sympoisum on Intelligent Control, pp.160-165, Singapore, October 2007.

[12] S. Prajna, A. Papachristodoulou, P. Seiler and P. A. Parrilo: SOSTOOLS:Sum of Squares Optimization Toolbox for MATLAB, Version 2.00, 2004.

[13] T. Takagi and M. Sugeno, “Fuzzy Identification of Systems and Its Applications to Modeling and Control”, IEEE Trans. on SMC 15, no. 1, pp.116-132, 1985.

[14] J. F. Sturm: “Using SeDUMi 1.02, a MATLAB toolbox for optimization over symmetric cones”, Optimization Methods and Software, vol. 11 & 12, pp.625-653, August 1999.

[15] K. C. Toh, R. H. Tutuncu and M. J. Todd, “On the impliment of SDPT3 (version 3.1) -a MATLAB software package for semidefinite-quadratic-linear programming”, 2004 IEEE International Conference on Computer Aided Control System Designs, pp.290-296, Sept. 2004.

[16] S. Prajna, A. Papachristodoulou and F. Wu: “Nonlinear Control Synthesis by Sum of Squares Optimization: A Lyapunov-based Approach”, Proceedings of the Asian Control Conference (ASCC), Melbourne, Australia, Feb. 2004, pp.157-165.

[17] K. Tanaka, T. Komatsu, H. Ohtake and H. O. Wang, “Micro Helicopter Control: LMI approach vs SOS approach”, 2008 IEEE International Conference on Fuzzy Systems, pp.347-353, Hong Kong, June 2008.