An SOS-based Stable Control of Polynomial Discrete Fuzzy Systems

Kazuo Tanaka, Hiroshi Ohtake and Hua O. Wang

Abstract— This paper presents stability and stabilization conditions of polynomial discrete fuzzy systems. First, based on polynomial Lyapunov functions, we derive a stability condition of polynomial discrete fuzzy systems without inputs. Secondly, we derive a stabilization condition to design a stable polynomial fuzzy controller. Both of them are represented in terms of SOS and are numerically (partially symbolically) solved via the recent developed SOSTOOLS. In addition, polynomial discrete fuzzy systems and controllers contain discrete Takagi-Sugeno (T-S) fuzzy systems and controllers as a special case. Hence, the approach discussed in this paper is more general than that based on the existing LMI approaches to discrete T-S fuzzy control system designs. To illustrate the validity of the design approach, a design example is provided. The example shows the utility of our approach.

I. INTRODUCTION

The Takagi-Sugeno (T-S) fuzzy model-based control methodology [1] provides a natural, simple and effective design approach to complement other nonlinear control techniques (e.g., [2]) that require special and rather involved knowledge. Moreover, 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) [3]. Within the general framework of T-S fuzzy model-based control systems, there has been, in particular, a flurry of research activities in the analysis and design of fuzzy control systems based on linear matrix inequalities (LMIs) (e.g., [1]). Hence, this approach have received a great deal of attention over the last decade.

Recently, we presented a sum of squares (SOS) approach to stability analysis [4] and stable controller design [5] for polynomial continuous fuzzy systems. In [6], we dealt with guaranteed cost control of polynomial continuous fuzzy systems. These are completely different approaches from the existing LMI approaches. To the best of our knowledge, the paper [4] presented the first attempt at applying an SOS to fuzzy systems. Our SOS approaches provide more extensive results for the existing LMI approaches to T-S fuzzy systems. However, SOS-based design approaches of polynomial *discrete* fuzzy systems have not been reported in the literature.

This paper presents stability and stabilization conditions of polynomial *discrete* fuzzy systems. First, based on poly-

nomial Lyapunov functions, we derive a stability condition of polynomial discrete fuzzy systems without inputs. Secondly, we derive a stabilization condition to design a stable polynomial fuzzy controller. Both of them are represented in terms of SOS and are numerically (partially symbolically) solved via the recent developed SOSTOOLS [7]. In addition, polynomial discrete fuzzy systems and controllers contain discrete Takagi-Sugeno (T-S) fuzzy systems and controllers as a special case. Hence, the approach discussed in this paper is more general than that based on the existing LMI approaches to discrete T-S fuzzy control system designs. To illustrate the validity of the design approach, a design example is provided. The example shows the utility of our approach.

II. DISCRETE TAKAGI-SUGENO FUZZY CONTROL SYSTEM AND STABILITY CONDITION BASED ON QUADRATIC LYAPUNOV FUNCTIONS

Section II summarizes discrete Takagi-Sugeno fuzzy control system and stability condition based on quadratic Lyapunov functions. The discrete Takagi-Sugeno fuzzy model [8] is described by fuzzy IF-THEN rules which represent local linear input-output relations of a nonlinear system. The main feature of this model is to express the local dynamics of each fuzzy implication (rule) by a linear system model. The overall fuzzy model of the system is achieved by fuzzy blending of the linear system models.

Consider the following nonlinear system:

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

where f is a nonlinear function. $\boldsymbol{x}(t) = [x_1(t) \ x_2(t) \ \cdots \ x_n(t)]^T$ is the state vector and $\boldsymbol{u}(t) = [u_1(t) \ u_2(t) \ \cdots \ u_m(t)]^T$ is the input vector. Based on the sector nonlinearity concept [1], we can exactly represent (1) with the discrete Takagi-Sugeno fuzzy model (2) (globally or at least semi-globally).

Model Rule i:

$$If \ z_1(t) \ is \ M_{i1} \ and \cdots \ and \ \ z_p(t) \ is \ M_{ip}$$
$$then \ \boldsymbol{x}(t+1) = \boldsymbol{A}_i\boldsymbol{x}(t) + \boldsymbol{B}_i\boldsymbol{u}(t) \quad i = 1, 2, \cdots, r, \tag{2}$$

where $z_j(t) \ (j = 1, 2, \cdots, p)$ is the premise variable. The membership function associated with the *i*th *Model Rule* and *j*th 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. The defuzzification process

of the model (2) can be represented as

$$\boldsymbol{x}(t+1) = \frac{\displaystyle\sum_{i=1}^{r} w_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i\boldsymbol{x}(t) + \boldsymbol{B}_i\boldsymbol{u}(t)\}}{\displaystyle\sum_{i=1}^{r} w_i(\boldsymbol{z}(t))}$$

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

where $\boldsymbol{z}(t) = [z_1(t) \cdots z_p(t)]$ and $w_i(\boldsymbol{z}(t)) = \prod_{j=1}^{p} M_{ij}(z_j(t))$. It should be noted from the properties of membership functions that the following relations hold.

$$\sum_{i=1}^{r} w_i(\boldsymbol{z}(t)) > 0, \quad w_i(\boldsymbol{z}(t)) \geq 0 \quad i = 1, 2, \cdots, r$$

Hence,

$$h_i(\boldsymbol{z}(t)) = \frac{w_i(\boldsymbol{z}(t))}{\displaystyle\sum_{i=1}^{r} w_i(\boldsymbol{z}(t))} \geq 0,$$

and

$$\sum_{i=1}^{r} h_i(\boldsymbol{z}(t)) = 1.$$

The parallel distributed compensation (PDC) offers a procedure to design a fuzzy controller from the given Takagi-Sugeno fuzzy model (2):

Control Rule i:

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

The overall fuzzy controller can be calculated by

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

By employing the quadratic Lyapunov function $\boldsymbol{x}^T(t)\boldsymbol{X}^{-1}\boldsymbol{x}(t)$, the well-known LMI condition [1] for the stability of the feedback system consisting of (3) and (5) is given as follows;

$$\boldsymbol{X} > \boldsymbol{0} \quad (6)$$

$$\begin{bmatrix} \boldsymbol{X} & \boldsymbol{X}\boldsymbol{A}_i^T - \boldsymbol{M}_i^T\boldsymbol{B}_i^T \\ \boldsymbol{A}_i\boldsymbol{X} - \boldsymbol{B}_i\boldsymbol{M}_i & \boldsymbol{X} \end{bmatrix} > \boldsymbol{0}, \quad (7)$$

$$\begin{bmatrix} \boldsymbol{X} & * \\ \left\{\dfrac{\boldsymbol{A}_i\boldsymbol{X} + \boldsymbol{A}_j\boldsymbol{X} - \boldsymbol{B}_i\boldsymbol{M}_j - \boldsymbol{B}_j\boldsymbol{M}_i}{2}\right\} & \boldsymbol{X} \end{bmatrix} \geq \boldsymbol{0},$$
$$i < j \quad (8)$$

where $\boldsymbol{M}_i = \boldsymbol{F}_i\boldsymbol{X}$. The conditions (6)-(8) are represented in terms of LMIs. Thus, the stability conditions can be solved numerically and efficiently by interior point algorithms.

III. POLYNOMIAL DISCRETE FUZZY MODEL AND POLYNOMIAL LYAPUNOV FUNCTION

Section II summarized discrete Takagi-Sugeno fuzzy model and stability analysis based on quadratic Lyapunov functions. In this section, we propose a new type of discrete fuzzy model with polynomial model consequence, i.e., fuzzy model whose consequent parts are represented by polynomials.

As shown in Section II, the stability conditions (6) - (8) for the T-S fuzzy system and the quadratic Lyapunov function reduce to LMIs. Hence, the stability conditions can be solved numerically and efficiently by interior point algorithms such as the LMI toolbox of MATLAB[1]. On the other hand, stability conditions for polynomial fuzzy systems and polynomial Lyapunov functions reduce to an SOS problem. Clearly, the problem is never solved by the LMI toolbox and can be solved via SOSTOOLS [7].

SOSTOOLS [7] is a free, third party MATLAB toolbox for solving sum of squares problems. The techniques behind it are based on the sum of squares decomposition for multivariate polynomials, which can be efficiently computed using semidefinite programming. SOSTOOLS is developed as a consequence of the recent interest in sum of squares polynomials, partly due to the fact that these techniques provide convex relaxations for many hard problems such as global, constrained, and boolean optimization. For more details, see the manual of SOSTOOLS [7].

A. Polynomial discrete fuzzy model

A polynomial continuous fuzzy model has been proposed in [4]. In this section, we propose a polynomial discrete fuzzy model. Using the sector nonlinearity concept, we exactly represent (1) with the following polynomial fuzzy model (9). The main difference between (2) and (9) is consequent part representation. The fuzzy models of (2) and (9) have linear model consequence and polynomial model consequence, respectively.

Model Rule i:

$$If \ z_1(t) \ is \ M_{i1} \ and \cdots \ and \ z_p(t) \ is \ M_{ip}$$
$$then \ \boldsymbol{x}(t+1) = \boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t), \quad (9)$$

where $i = 1, 2, \cdots, r$. $\boldsymbol{A}_i(\boldsymbol{x}(t))$ and $\boldsymbol{B}_i(\boldsymbol{x}(t))$ are polynomial matrices in $\boldsymbol{x}(t)$. $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$ 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 $\boldsymbol{x}(t)$. A monomial in $\boldsymbol{x}(t)$ is a function of the form $x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$, where $\alpha_1, \alpha_2, \cdots, \alpha_n$ are nonnegative integers. 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 (9) has a polynomial in each consequent part. In Sections IV and V, we will consider $\boldsymbol{T}(\tilde{\boldsymbol{x}}(t)) \in R^{N \times m}$ that is a polynomial matrix defined by $\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) = \boldsymbol{T}(\tilde{\boldsymbol{x}}(t))\boldsymbol{x}(t)$. The details of $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$ will be given in Proposition 1. The definition of $\tilde{\boldsymbol{x}}(t)$ will be also presented later.

We assume that

---

[1]A registered trademark of MathWorks, Inc.

$$\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) = 0 \text{ iff } \boldsymbol{x}(t) = 0$$

throughout this paper.

The defuzzification process of the model (9) can be represented as

$$\boldsymbol{x}(t+1) = \sum_{i=1}^{r} h_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$$
$$+ \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t)\}. \qquad (10)$$

Thus, the overall fuzzy model is achieved by fuzzy blending of the polynomial system models.

If $\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) = \boldsymbol{x}(t)$ and $\boldsymbol{A}_i(\boldsymbol{x}(t))$ and $\boldsymbol{B}_i(\boldsymbol{x}(t))$ are constant matrices for all i, then $\boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t)$ reduces to $\boldsymbol{A}_i\boldsymbol{x}(t) + \boldsymbol{B}_i\boldsymbol{u}(t)$, that is, then (10) reduces to (3). Therefore, (10) is a more general representation.

Remark 1: As we can see in Section III-C, 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 results than the existing LMI approaches to T-S fuzzy models.

B. Polynomial Lyapunov function

To obtain more relaxed stability results, we employ a polynomial Lyapunov function [4] represented by

$$\hat{\boldsymbol{x}}^T(\boldsymbol{x}(t))\boldsymbol{P}(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)), \qquad (11)$$

where $\boldsymbol{P}(\boldsymbol{x}(t))$ is a polynomial matrix in $\boldsymbol{x}(t)$. If $\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t)$ and $\boldsymbol{P}(\boldsymbol{x}(t))$ is a constant matrix, then (11) reduces to the quadratic Lyapunov function $\boldsymbol{x}^T(t)\boldsymbol{P}\boldsymbol{x}(t)$. Therefore, (11) is a more general representation.

C. Fuzzy Modeling Example

Consider the following nonlinear system:

$$\begin{cases} x_1(t+1) = x_1(t) + x_2^2(t) + u(t), \\ x_2(t+1) = -\tan x_1(t) + 2x_2(t), \end{cases} \qquad (12)$$

where we assume that $-\pi/4 \le x_1(t) \le \pi/4$.

To obtain a T-S fuzzy model, we also assume that $-d \le x_2(t) \le d$. Using the concept of sector nonlinearity, we have the T-S fuzzy model (13) that can exactly represent the dynamics under $-\pi/4 \le x_1(t) \le \pi/4$ and $-d \le x_2(t) \le d$.

$$\boldsymbol{x}(t+1) = \sum_{i=1}^{4} h_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i\boldsymbol{x}(t) + \boldsymbol{B}_i u(t)\} \qquad (13)$$

where $\boldsymbol{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T$, $\boldsymbol{z}(t) = \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T$, and

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & d \\ -\frac{4}{\pi} & 2 \end{bmatrix},$$

$$\boldsymbol{A}_2 = \begin{bmatrix} 1 & d \\ -1 & 2 \end{bmatrix},$$

$$\boldsymbol{A}_3 = \begin{bmatrix} 1 & -d \\ -\frac{4}{\pi} & 2 \end{bmatrix},$$

$$\boldsymbol{A}_4 = \begin{bmatrix} 1 & -d \\ -1 & 2 \end{bmatrix},$$

$$\boldsymbol{B}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

$$\boldsymbol{B}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

The membership functions are obtained as

$$h_1(\boldsymbol{z}(t)) = \frac{x_2(t)+d}{2d} \frac{\tan x_1(t) - x_1(t)}{(\frac{4}{\pi}-1)x_1(t)},$$

$$h_2(\boldsymbol{z}(t)) = \frac{x_2(t)+d}{2d} \frac{\frac{4}{\pi}x_1(t) - \tan x_1(t)}{(\frac{4}{\pi}-1)x_1(t)},$$

$$h_3(\boldsymbol{z}(t)) = \frac{d-x_2(t)}{2d} \frac{\tan x_1(t) - x_1(t)}{(\frac{4}{\pi}-1)x_1(t)},$$

$$h_4(\boldsymbol{z}(t)) = \frac{d-x_2(t)}{2d} \frac{\frac{4}{\pi}x_1(t) - \tan x_1(t)}{(\frac{4}{\pi}-1)x_1(t)}.$$

We can see that four rules are needed to represent the dynamics.

On the other hand, we have the following polynomial fuzzy model that can exactly represent the dynamics under $-\pi/4 \le x_1(t) \le \pi/4$.

$$\boldsymbol{x}(t+1) = \sum_{i=1}^{2} h_i(z(t))\{\boldsymbol{A}_i(\boldsymbol{x}(t))\boldsymbol{x}(t) + \boldsymbol{B}_i(\boldsymbol{x}(t))u(t)\} \qquad (14)$$

where $\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T$, $z(t) = x_1(t)$ and

$$\boldsymbol{A}_1(\boldsymbol{x}(t)) = \begin{bmatrix} 1 & x_2(t) \\ -\frac{4}{\pi} & 2 \end{bmatrix},$$

$$\boldsymbol{A}_2(\boldsymbol{x}(t)) = \begin{bmatrix} 1 & x_2(t) \\ -1 & 2 \end{bmatrix},$$

$$\boldsymbol{B}_1(\boldsymbol{x}(t)) = \boldsymbol{B}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

$$\boldsymbol{B}_2(\boldsymbol{x}(t)) = \boldsymbol{B}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

The membership functions are obtained as

$$h_1(z(t)) = \frac{\tan x_1(t) - x_1(t)}{(\frac{4}{\pi}-1)x_1(t)},$$

$$h_2(z(t)) = \frac{\frac{4}{\pi}x_1(t) - \tan x_1(t)}{(\frac{4}{\pi}-1)x_1(t)}.$$

Note that the assumption that $-d \le x_2(t) \le d$ is not needed in the construction of the polynomial fuzzy models. In addition, the number of rules in the polynomial fuzzy model is fewer than that in the T-S fuzzy model.

IV. SOS STABILITY CONDITION

A. Sum of Squares

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 $f_1(\boldsymbol{x}(t)), \cdots, f_m(\boldsymbol{x}(t))$ such that $f(\boldsymbol{x}(t)) = \sum_{i=1}^{m} f_i^2(\boldsymbol{x}(t))$. It is clear that $f(\boldsymbol{x}(t))$ being an SOS naturally implies $f(\boldsymbol{x}(t)) > 0$ for all $\boldsymbol{x}(t) \in R^n$. This can be shown equivalent to the existence of a special quadric form stated in the following proposition [10].

Proposition 1: [11] Let $f(\boldsymbol{x}(t))$ be a polynomial in $\boldsymbol{x}(t) \in R^n$ of degree $2d$. In addition, let $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$ be a column vector whose entries are all monomials in $\boldsymbol{x}(t)$ with degree no greater than d. Then $f(\boldsymbol{x}(t))$ is a sum of squares iff there exists a positive semidefinite matrix $\boldsymbol{P}$ such that

$$f(\boldsymbol{x}(t)) = \hat{\boldsymbol{x}}^T(\boldsymbol{x}(t))\boldsymbol{P}\hat{\boldsymbol{x}}(\boldsymbol{x}(t)). \qquad (15)$$

Expressing an SOS polynomial using a quadratic form as in (15) has also been referred to as the Gram matrix method.

As mentioned before, a monomial in $\boldsymbol{x}(t)$ is a function of the form $x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$, where $\alpha_1, \alpha_2, \cdots, \alpha_n$ are nonnegative integers. In this case, the degree of the monomial is given by $\alpha_1 + \alpha_2 + \cdots + \alpha_n$.

A sum of squares decomposition for $f(\boldsymbol{x}(t))$ can be computed using semidefinite programming, since it amounts to searching for an element $\boldsymbol{P}$ in the intersection of the cone of positive semidefinite matrices and a set defined by some affine constraints that arise from (15). Note in particular that the polynomial $f(\boldsymbol{x}(t))$ is globally nonnegative if it can be decomposed as a sum of squares. Hence the sum of squares decomposition in conjunction with semidefinite programming provides a polynomial-time computational relaxation for proving global nonnegativity of multivariate polynomials [11], [12], which belongs to the class of NP-hard problems. Even though the sum of squares condition is not necessary for nonnegativity, numerical experiments seem to indicate that the gap between sum of squares and nonnegativity is small [10].

B. SOS Stability Condition

To lighten the notation, only this subsection will drop the notation with respect to time t. For instance, we will employ $\boldsymbol{x}$, $\hat{\boldsymbol{x}}$ instead of $\boldsymbol{x}(t)$, $\hat{\boldsymbol{x}}(t)$, respectively. Thus, we drop the notation with respect to time t, but it should be kept in mind that $\boldsymbol{x}$ means $\boldsymbol{x}(t)$. However, to distinguish between $\boldsymbol{x}(t)$ and $\boldsymbol{x}(t+1)$, we will remain the notation with respect to time $t+1$. Hence, we will employ $\boldsymbol{x}(t+1)$, $\tilde{\boldsymbol{x}}(t+1)$, etc.

$\boldsymbol{K} = \{k_1, k_2, \cdots, k_m\}$ denote the row indices of $\boldsymbol{B}_i(\boldsymbol{x})$ whose corresponding row is equal to zero and the row indices of $\boldsymbol{A}_i(\boldsymbol{x})$ whose corresponding row does not contain non-polynomial nonlinear terms (e.g. trigonometric functions). Using $k_1, k_2, \cdots, k_{m-1}$ and k_m, we define $\tilde{\boldsymbol{x}} =$ $(x_{k_1}, x_{k_2}, \cdots x_{k_m})$. From the definition, a partial system of (10) can be represented as

$$\tilde{\boldsymbol{x}}(t+1) = \tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x}, \qquad (16)$$

where $\tilde{\boldsymbol{A}}(\boldsymbol{x})$ is a polynomial matrix. The equation (16) will play an important role in Theorems 1 and 2.

Theorem 1: The zero equilibrium of the system (10) with $\boldsymbol{u}(t) = 0$ is stable if there exists a symmetric polynomial matrix $\boldsymbol{P}(\tilde{\boldsymbol{x}}) \in \boldsymbol{R}^{N \times N}$ such that (17) and (18) are satisfied, where $\epsilon_1(\boldsymbol{x})$ and $\epsilon_{2i}(\boldsymbol{x})$ are non negative polynomials such that $\epsilon_1(\boldsymbol{x}) > 0$ ($\boldsymbol{x} \neq 0$) and $\epsilon_{2i}(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$.

$$\hat{\boldsymbol{x}}^T(\boldsymbol{x})(\boldsymbol{P}(\tilde{\boldsymbol{x}}) - \epsilon_1(\boldsymbol{x})\boldsymbol{I})\hat{\boldsymbol{x}}(\boldsymbol{x}) \; is \; SOS \qquad (17)$$

$$-\hat{\boldsymbol{x}}^T(\boldsymbol{x}) \Big(\boldsymbol{A}_i^T(\boldsymbol{x})\boldsymbol{T}^T(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x})\boldsymbol{P}(\tilde{\boldsymbol{x}})\boldsymbol{T}(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x})\boldsymbol{A}_i(\boldsymbol{x})$$
$$-\boldsymbol{P}(\tilde{\boldsymbol{x}}) + \epsilon_{2i}(\boldsymbol{x})\boldsymbol{I} \Big) \hat{\boldsymbol{x}}(\boldsymbol{x}) is \; SOS \quad \forall i, \; (18)$$

where $\boldsymbol{T}(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x}) \in \boldsymbol{R}^{N \times n}$ is a polynomial matrix given by

$$\boldsymbol{T}(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x}) = \boldsymbol{T}(\tilde{\boldsymbol{x}}(t+1)). \qquad (19)$$

In addition, if (18) holds with $\epsilon_{2i}(\boldsymbol{x}) > 0$ for $\boldsymbol{x} \neq 0$, then the zero equilibrium is asymptotically stable. If $\boldsymbol{P}(\tilde{\boldsymbol{x}})$ is a constant matrix, then the stability holds globally.

Proof: The proof is omitted due to lack of space. ∎

Remark 2: Selection of the non-negative polynomials $\epsilon_1(\boldsymbol{x})$ and $\epsilon_{2i}(\boldsymbol{x})$ such that $\epsilon_1(\boldsymbol{x}) > 0$ ($\boldsymbol{x} \neq 0$) and $\epsilon_{2i}(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$ influence the feasibility of the SOS problem. Hence, the polynomial structure of $\epsilon_1(\boldsymbol{x})$ and $\epsilon_{2i}(\boldsymbol{x})$ is needed to select carefully.

Remark 3: When $\boldsymbol{A}_i(\boldsymbol{x})$ and $\boldsymbol{P}(\tilde{\boldsymbol{x}})$ are constant matrices and $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x}$, the system representation is the same as the Takagi-Sugeno fuzzy model and stability analysis based on quadratic Lyapunov function used in many of the references, e.g., [1]. Thus, our SOS approach to polynomial fuzzy models contains the existing LMI approaches to Takagi-Sugeno fuzzy models as a special case. Therefore, our SOS approach to polynomial fuzzy models provides much more relaxed stability results than the existing LMI approaches to Takagi-Sugeno fuzzy models.

V. STABLE CONTROLLER DESIGN VIA SOS

A. Polynomial Fuzzy Controller

Since the PDC mirrors the structure of the fuzzy model of a system, a fuzzy controller with polynomial rule consequence can be constructed from the given fuzzy model (9).

Control Rule i:

$$If \; z_1(t) \; is \; M_{i1} \; and \cdots and \; z_p(t) \; is \; M_{ip}$$
$$then \; \boldsymbol{u}(t) = -\boldsymbol{F}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) \quad i = 1, 2, \cdots, r \quad (20)$$

The overall fuzzy controller can be calculated by

$$\boldsymbol{u}(t) = -\sum_{i=1}^{r} h_i(\boldsymbol{z}(t))\boldsymbol{F}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)). \qquad (21)$$

From (10) and (21), the closed-loop system can be represented as

$$\boldsymbol{x}(t+1) = \sum_{i=1}^{r}\sum_{j=1}^{r} h_i(\boldsymbol{z}(t))h_j(\boldsymbol{z}(t))$$
$$\times \{\boldsymbol{A}_i(\boldsymbol{x}(t)) - \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{F}_j(\boldsymbol{x}(t))\}\hat{\boldsymbol{x}}(\boldsymbol{x}(t)). \qquad (22)$$

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, then (10) and (21) reduce to (3) and (5), respectively. Therefore, (10) and (21) are a more general representation.

We provide another important proposition with respect to the relaxation.

Proposition 2: [10] Let $\boldsymbol{L}(\boldsymbol{x}(t))$ be an $N \times N$ symmetric polynomial matrix of degree $2d$ in $\boldsymbol{x}(t) \in R^n$. Furthermore, let $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$ be a column vector whose entries are all monomials in $\boldsymbol{x}(t)$ with degree no greater than d, and consider the following conditions.

(1) $\boldsymbol{L}(\boldsymbol{x}(t)) \geq 0$ for all $\boldsymbol{x}(t) \in R^n$.
(2) $\boldsymbol{v}^T(t)\boldsymbol{L}(\boldsymbol{x}(t))\boldsymbol{v}(t)$ is a sum of squares, where $\boldsymbol{v}(t) \in R^N$.
(3) There exists a positive semidefinite matrix $\boldsymbol{Q}$ such that $\boldsymbol{v}^T(t)\boldsymbol{L}(\boldsymbol{x}(t))\boldsymbol{v}(t) = (\boldsymbol{v}(t) \otimes \hat{\boldsymbol{x}}(\boldsymbol{x}(t)))^T \boldsymbol{Q}(\boldsymbol{v} \otimes \hat{\boldsymbol{x}}(\boldsymbol{x}(t)))$, where $\otimes$ denotes the Kronecker product.

Then (1) $\Leftarrow$ (2) and (2) $\Longleftrightarrow$ (3).

B. SOS Stabilization Condition

This subsection gives a stable control design condition represented in terms of SOS. Hence the stable fuzzy controller design with polynomial rule consequence is numerically a feasibility problem via SOSTOOLS.

To lighten the notation, this subsection will also 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 $\boldsymbol{x}(t)$.

Theorem 2: The control system consisting of (10) and (21) is stable 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 (23), (24) and (25) are satisfied, where $\epsilon_1(\boldsymbol{x})$ and $\epsilon_{2ij}(\boldsymbol{x})$ are non negative polynomials such that $\epsilon_1(\boldsymbol{x}) > 0$ ($\boldsymbol{x} \neq 0$) and $\epsilon_{2ij}(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$.

$$\boldsymbol{v}_1^T(\boldsymbol{X}(\tilde{\boldsymbol{x}}) - \epsilon_1(\boldsymbol{x})\boldsymbol{I})\boldsymbol{v}_1 \; is \; SOS \qquad (23)$$

$$\boldsymbol{v}_2^T \begin{bmatrix} \boldsymbol{X}(\tilde{\boldsymbol{x}}) - \epsilon_{2ii}(\boldsymbol{x})\boldsymbol{I} & * \\ \boldsymbol{T}(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x})\Omega_{ii}(\boldsymbol{x}) & \boldsymbol{X}(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x}) \end{bmatrix} \boldsymbol{v}_2$$
$$is \; SOS, \qquad (24)$$

$$\boldsymbol{v}_3^T \begin{bmatrix} \boldsymbol{X}(\tilde{\boldsymbol{x}}) - \epsilon_{2ij}(\boldsymbol{x})\boldsymbol{I} & \\ \frac{1}{2}\boldsymbol{T}(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x})\left(\Omega_{ij}(\boldsymbol{x}) + \Omega_{ji}(\boldsymbol{x})\right) & \end{bmatrix}$$
$$\begin{matrix} * \\ \boldsymbol{X}(\tilde{\boldsymbol{A}}(\boldsymbol{x})\boldsymbol{x}) \end{matrix} \; \boldsymbol{v}_3 \; is \; SOS, \quad i < j, \quad (25)$$

where $\Omega_{ij}(\boldsymbol{x}) = \boldsymbol{A}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{M}_j(\boldsymbol{x})$. * denotes the transposed elements (matrices) for symmetric positions. $\boldsymbol{v}_1 \in R^N$, $\boldsymbol{v}_2, \boldsymbol{v}_3 \in R^{2N}$ are vectors that are independent of $\boldsymbol{x}$. In addition, if (24) and (25) hold with $\epsilon_{2ij}(\boldsymbol{x}) > 0$ for $\boldsymbol{x} \neq 0$, then the zero equilibrium is asymptotically stable. If $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ is a constant matrix, then the stability holds globally. A stabilizing feedback gain $\boldsymbol{F}_i(\boldsymbol{x})$ can be obtained from $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ as

$$\boldsymbol{F}_i(\boldsymbol{x}) = \boldsymbol{M}_i(\boldsymbol{x})\boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}}). \qquad (26)$$

Proof: The proof is omitted due to lack of space. ∎

Remark 4: Note that $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ and $\boldsymbol{v}_3$ 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 $\hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{L}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x})$ is an SOS, where $\boldsymbol{L}(\boldsymbol{x})$ is a symmetric polynomial matrix in $\boldsymbol{x}(t)$. However, it is guaranteed from Proposition 2 that if $\boldsymbol{v}^T\boldsymbol{L}(\boldsymbol{x})\boldsymbol{v}$ is an SOS, then $\boldsymbol{L}(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$.

Remark 5: Selection of the non-negative polynomials $\epsilon_1(\boldsymbol{x})$ and $\epsilon_{2ij}(\boldsymbol{x})$ such that $\epsilon_1(\boldsymbol{x}) > 0$ ($\boldsymbol{x} \neq 0$) and $\epsilon_{2ij}(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$ influence the feasibility of the SOS problem. Hence, the polynomial structure of $\epsilon_1(\boldsymbol{x})$ and $\epsilon_{2ij}(\boldsymbol{x})$ is needed to select carefully.

Remark 6: To avoid introducing non-convex condition, we assume that $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ only depends on states $\tilde{\boldsymbol{x}}$ whose corresponding row in $\boldsymbol{A}_i(\boldsymbol{x})$ does not contain non-polynomial nonlinear terms, and states whose dynamics is not directly affected by the control input, namely states whose corresponding rows in $\boldsymbol{B}_i(\boldsymbol{x})$ are zero.

Remark 7: When $\boldsymbol{A}_i(\boldsymbol{x})$, $\boldsymbol{B}_i(\boldsymbol{x})$, $\boldsymbol{F}_i(\boldsymbol{x})$ and $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ are constant matrices and $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x}$, the system representation is basically the same as the Takagi-Sugeno fuzzy model and control used in many of the references, e.g., [1]. Thus, our SOS approach to fuzzy model and control with polynomial rule consequence contains the existing LMI approaches to Takagi-Sugeno fuzzy model and control as a special case. Therefore, our SOS approach provides much more relaxed results than the existing approaches to Takagi-Sugeno fuzzy model and control.

VI. DESIGN EXAMPLE

To illustrate the validity of the design approach, this section provides a design example.

Recall the following nonlinear system as shown in Section III-C:

$$\begin{cases} x_1(t+1) = x_1(t) + x_2^2(t) + u(t) \\ x_2(t+1) = -\tan x_1(t) + 2x_2(t), \end{cases} \quad (27)$$

where we assume that $-\pi/4 \le x_1(t) \le \pi/4$.

As mentioned in Sectiob III-C, we have the following polynomial fuzzy model that can exactly represent the dynamics under $-\pi/4 \le x_1(t) \le \pi/4$.

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

The SOS conditions are infeasible if $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$ are constant marices in Theorem 2. However, the SOS conditions are feasible if we select $\boldsymbol{M}_i(\boldsymbol{x})$ as a polynomial matrix. Thus, the polynomial fuzzy controller (21) with the polynomial feedback vectors $\boldsymbol{F}_i(\boldsymbol{x})$ is more useful than the T-S fuzzy controller (5) with the constant feedback vectors $\boldsymbol{F}_i$. This shows the utility of our SOS-based design approach.

Fig. 1 shows the control result of the polynomial fuzzy controller (21) for the initial states $[0.5 \quad 0]^T$. Fig. 2 shows time transient of Lyapunov function during the control. The designed polynomial fuzzy controller stabilizes the system.
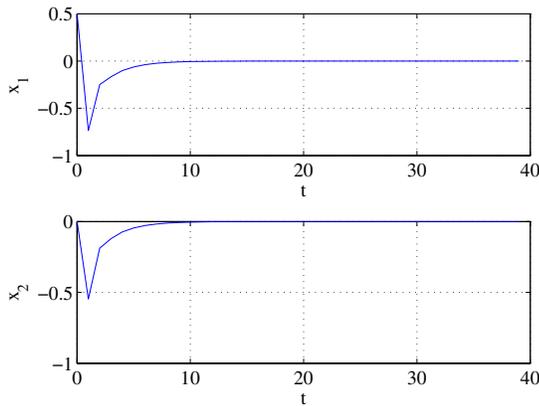
Fig. 1.   Control result.

VII. Conclusions

This paper has presented stability and stabilization conditions of polynomial discrete fuzzy systems. First, based on polynomial Lyapunov functions, we have derived a stability condition of polynomial discrete fuzzy systems without inputs. Secondly, we have derived a stabilization condition to design a stable polynomial fuzzy controller. Both of them are represented in terms of SOS and are numerically (partially symbolically) solved via the recent developed SOSTOOLS. The approach discussed in this paper is more general than that based on the existing LMI approaches to discrete T-S fuzzy control system designs. To illustrate the validity of the design approach, a design example has been provided. The example has shown the utility of our approach.
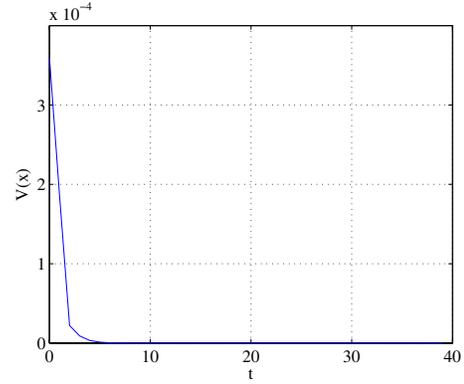
Fig. 2.   Time Transient of Lyapunov function.

Guaranteed cost control for polynomial discrete fuzzy systems will be presented in [13].

VIII. Acknowledgments

References

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

[2] R. Sepulcher, M. Jankovic and P. Kokotovic: Constructive Nonlinear Control, Springer, 1997

[3] 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.

[4] 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, pp.4071-4076, New York, July, 2007.

[5] 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.

[6] K. Tanaka, K. Yamauchi, H. Ohtake and H. O. Wang, " Guaranteed Cost Control of Polynomial Fuzzy Systems via a Sum of Squares Approach, 2007 IEEE Conference on Decision and Control, pp.5954-5959, New Orleans, Dec., 2007.

[7] S. Prajna, A. Papachristodoulou, P. Seiler and P. A. Parrilo: SOS-TOOLS:Sum of Squares Optimization Toolbox for MATLAB, Version 2.00, 2004.

[8] 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.

[9] L. Xie, S. Shishkin and M. Fu, "Piecewise Lyapunov Functions for Robust Stability of Linear Time-Varying Systems", Systems & Control Letters 31 pp.165-171, 1997.

[10] 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.

[11] P. A. Parrilo: "Structured Semidefinite Programs and Semialgebraic Geometly Methods in Robustness and Optimization", PhD thesis, California Institute of Technology, Pasadena, CA, 2000.

[12] N. Z. Skor, "Class of global minimum bounds of polynomial functions", Cybernetics, No.23, Vol.6, pp.731-734, 1987.

[13] K. Tanaka, H. Ohtake and H. O. Wang: "A Sum of Squares Approach to Guaranteed Cost Control of Polynomial Discrete Fuzzy Systems", 17th IFAC World Congress, accepted, 2008.