A New Sum-of-Squares Design Framework for Robust Control of Polynomial Fuzzy Systems With Uncertainties

著者（英）	Kazuo Tanaka, Motoyasu Tanaka, Ying-Jen Chen, Hua O. Wang
journal or publication title	IEEE Transactions on Fuzzy Systems
volume	24
number	1
page range	94-110
year	2016-02
URL	http://id.nii.ac.jp/1438/00009298/

doi: 10.1109/TFUZZ.2015.2426719

A New Sum-of-Squares Design Framework for Robust Control of Polynomial Fuzzy Systems with Uncertainties

Kazuo Tanaka, *Fellow, IEEE,* Motoyasu Tanaka, *Member, IEEE,* Ying-Jen Chen, *Member, IEEE,* and Hua O. Wang, *Senior Member, IEEE*

***Abstract*—This paper presents a new sum-of-squares (SOS, for brevity) design framework for robust control of polynomial fuzzy systems with uncertainties. Two kinds of robust stabilization conditions are derived in terms of SOS. One is global SOS robust stabilization conditions that guarantee the global and asymptotical stability of polynomial fuzzy control systems. The other is semi-global SOS robust stabilization conditions. The latter is available for very complicated systems that are difficult to guarantee the global and asymptotical stability of polynomial fuzzy control systems. The main feature of all the SOS robust stabilization conditions derived in this paper are to be expressed as non-convex formulations with respect to polynomial Lyapunov function parameters and polynomial feedback gains. Since a typical transformation from non-convex SOS design conditions to convex SOS design conditions often results in some conservative issues, the new design framework presented in this paper gives key ideas to avoid the conservative issues. The first key idea is that we directly solve non-convex SOS design conditions without applying the typical transformation. The second key idea is that we bring a so-called copositivity concept. These ideas provide some advantages in addition to relaxations. To solve our SOS robust stabilization conditions efficiently, we introduce a gradient algorithm formulated as a minimizing optimization problem of the upper bound of the time derivative of an SOS polynomial that can be regarded as a candidate of polynomial Lyapunov functions. Three design examples are provided to illustrate the validity and applicability of the proposed design framework. The examples demonstrate advantages of our new SOS design framework for the existing LMI approaches and the existing convex SOS approach.**

***Index Terms*—copositivity, polynomial Lyapunov function, polynomial fuzzy system with uncertainty, robust stabilization, sum of squares.**

I. Introduction

TODAY there exists a large body of literature on Takagi-Sugeno (T-S) fuzzy model-based control [1]. Especially, linear matrix inequalities (LMIs) based designs, e.g., [2], [3], have been paid a lot of attention after LMI-based designs have been discussed in [4]-[6]. A key feature of the approach is that it renders simple, natural and effective design procedures as alternatives or supplements to other nonlinear control techniques (e.g., [7]) that require special and rather involved knowledge. The LMI-based design approaches entail obtaining numerical solutions by convex optimization methods such as the interior point method [8].

Though LMI-based approaches have enjoyed great success and popularity, there still exist a large number of design problems that either cannot be represented in terms of LMIs, or the results obtained through LMIs are sometimes conservative. Recently, as a post-LMI framework, an SOS based approach has received a great deal of attention in control of nonlinear systems using polynomial fuzzy systems and controllers, which includes the well-known Takagi-Sugeno fuzzy systems and controllers as special cases. An SOS approach to polynomial fuzzy control system designs has first presented in [9]-[13]. It can be seen that SOS approaches [9]-[22] provide more extensive and/or relaxed results for the existing LMI approaches [2], [3], [23]-[35] to T-S fuzzy model and control. However, there exists a very few literature on SOS-based robust control designs for polynomial fuzzy systems with uncertainties. To the best of our knowledge, an SOS-based robust control design for polynomial fuzzy systems with uncertainties has been discussed only in [36]. The most important point of SOS-based design conditions is that, to obtain convex SOS design conditions, the existing SOS-based design conditions [9]-[20] utilize a typical transformation from non-convex SOS design conditions to convex SOS design conditions. However, the transformation often results in some conservative issues although no such conservatism exists in LMI transformation cases. In [36], the typical transformation is employed to obtain convex SOS robust stabilization conditions. Furthermore, not only the conservative issues but also other two difficulties are found in the existing SOS approach. One is a restrictive polynomial Lyapunov function setting that leads to conservative stability results. The other is that the stability does not generally hold globally in the existing SOS approach. These will be concretely discussed in Remarks 2 and 3. This paper gives new ideas to solve the conservative issues and the difficulties in the existing SOS approach.

This paper presents a new SOS design framework for robust control of polynomial fuzzy systems with uncertainties. The framework gives key ideas to avoid the conservative issues. The first key idea is that we directly solve non-convex SOS design conditions without applying the typical transformation. The second key idea is that we bring a so-called copositivity concept. These ideas provide some advantages in addition to

Manuscript received April 20, 2007; revised November 18, 2007. This work was supported in part by a Grant-in-Aid for Scientific Research (C) 25420215 from the Ministry of Education, Science and Culture of Japan.

Kazuo Tanaka, Motoyasu Tanaka and Ying-Jen Chen 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; mtanaka@uec.ac.jp; chen@rc.mce.uec.ac.jp;).

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

relaxations. To solve our SOS robust stabilization conditions efficiently, we introduce a gradient algorithm formulated as a minimizing optimization problem of the upper bound of the time derivative of an SOS polynomial that can be regarded as a candidate of polynomial Lyapunov functions.

The rest of the paper is organized as follows. Section II recalls a polynomial fuzzy system defined in [9]-[13] and defines a polynomial fuzzy system with uncertainty. Sections III and IV give a new SOS framework for robust control, i.e., robust stabilization conditions to design a robust fuzzy controller and an algorithm to solve them, respectively. Section V entails two design examples to demonstrate the validity and applicability of the proposed design framework. The examples demonstrate advantages of our SOS robust stabilization conditions for the existing LMI approaches and the existing convex SOS approach. Sections VI and VII present semi-global robust stabilization conditions and their design example, respectively. The design example deals with a kind of unmanned aerial vehicles (UAVs) that is a very complicated system with high nonlinearity.

It is assumed that all the matrices and vectors in this paper have appropriate dimensions. $\boldsymbol{P} \succ \boldsymbol{0}$ ($\boldsymbol{P} \succeq \boldsymbol{0}$) means that $\boldsymbol{P}$ is a positive definite matrix (positive semi-definite matrix).

II. Polynomial fuzzy system with uncertainties

Consider the following nonlinear system:

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

where f is a smooth nonlinear function such that $f(\boldsymbol{0},\boldsymbol{0}) = \boldsymbol{0}$. $\boldsymbol{x}(t) = [x_1(t)\ x_2(t)\ \cdots\ x_n(t)]^T$ 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 [2], we can exactly represent (1) with the following T-S fuzzy model [37] (globally or at least semi-globally).

Model Rule i:

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

where $z_j(t)$ $(j = 1, 2, \cdots, p)$ is the premise variable. The membership function, M_{ij}, denotes the *j*th premise variable component in the *i*th $Model\ Rule$. 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 overall dynamics of the system is represented by fuzzy blending of the linear system models. That is, the defuzzification process of the T-S model (2) can be represented as

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \frac{\sum_{i=1}^{r} w_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i\boldsymbol{x}(t) + \boldsymbol{B}_i\boldsymbol{u}(t)\}}{\sum_{i=1}^{r} w_i(\boldsymbol{z}(t))} \\ &= \sum_{i=1}^{r} h_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i\boldsymbol{x}(t) + \boldsymbol{B}_i\boldsymbol{u}(t)\},\end{aligned} \tag{3}$$

where

$$\boldsymbol{z}(t) = [z_1(t) \cdots z_p(t)]$$

and

$$w_i(\boldsymbol{z}(t)) = \prod_{j=1}^{p} M_{ij}(z_j(t)).$$

Since the number of $Model\ Rule$ that fire for all t is larger than or equal to one in general, 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,

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

In [9] and [12], we proposed a new type of fuzzy model with polynomial model consequence, i.e., fuzzy model whose consequent parts are represented by polynomials. Using the sector nonlinearity concept [2], we exactly represent (1) with the following polynomial fuzzy model (4) even if the nonlinear system (1) contains polynomial elements. The main difference between the T-S fuzzy model [37] and the polynomial fuzzy model is consequent part representation. The fuzzy model (4) has a polynomial model consequence.

Model Rule i:

$$If\ z_1(t)\ is\ M_{i1}\ and \cdots\ and\ \ z_p(t)\ is\ M_{ip}$$
$$then\ \dot{\boldsymbol{x}}(t) = \boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t), \tag{4}$$

where $i = 1, 2, \cdots, r$. r denotes the number of $Model\ Rules$. $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$ 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 α_1, α_2, $\cdots$, α_n are nonnegative integers. $\boldsymbol{A}_i(\boldsymbol{x}(t)) \in \boldsymbol{R}^{n\times N}$ 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 (4) has a polynomial in each consequent part. We assume that

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

throughout this paper.

The defuzzification process of the model (4) can be represented as

$$\dot{\boldsymbol{x}}(t) = \sum_{i=1}^{r} h_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t)\}. \tag{5}$$

The polynomial fuzzy model is an extension of the T-S fuzzy model. The extension bring us some advantages [12]. One is that SOS stabilization conditions provides more relaxed results than the existing LMI stabilization conditions. Another advance is that original nonlinear systems with polynomial terms can be exactly and globally represented by polynomial fuzzy models although the T-S fuzzy models are sometimes not global models for the original nonlinear systems with polynomial terms.

Remark 1. *Stability conditions for the T-S fuzzy system have been mainly represented in terms of LMIs [2]. Hence, the LMI stability conditions can be solved numerically and efficiently by interior point algorithms, e.g., by LMI solvers. On the other hand, the convex SOS conditions in [9]-[20], [36] for polynomial fuzzy systems are represented as convex SOS problems. Clearly, the problems can not be directly solved by LMI solvers, but they can be solved via an SOS solver (SOSOPT [38], SOSTOOLS [39], etc.) and an SDP solver [40], [41].*

This paper focuses on stabilization of the polynomial fuzzy model with uncertainties. Hence, we define a polynomial fuzzy model with uncertainties as follows.

Model Rule i**:**

$$\begin{aligned} & If\ z_1(t)\ is\ M_{i1}\ and \cdots\ and\ \ z_p(t)\ is\ M_{ip} \\ & then \\ & \dot{\boldsymbol{x}}(t) = \{\boldsymbol{A}_i(\boldsymbol{x}(t)) \\ & \qquad +\boldsymbol{D}_{ai}(\boldsymbol{x}(t))\boldsymbol{\Delta}_{ai}(\boldsymbol{x}(t))\boldsymbol{E}_{ai}(\boldsymbol{x}(t))\}\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) \\ & \quad +\{\boldsymbol{B}_i(\boldsymbol{x}(t)) \\ & \qquad +\boldsymbol{D}_{bi}(\boldsymbol{x}(t))\boldsymbol{\Delta}_{bi}(\boldsymbol{x}(t))\boldsymbol{E}_{bi}(\boldsymbol{x}(t))\}\boldsymbol{u}(t), \end{aligned} \tag{6}$$

where $i = 1, 2, \cdots, r$. $\boldsymbol{D}_{ai}(\boldsymbol{x}(t))$, $\boldsymbol{D}_{bi}(\boldsymbol{x}(t))$, $\boldsymbol{E}_{ai}(\boldsymbol{x}(t))$ and $\boldsymbol{E}_{bi}(\boldsymbol{x}(t))$ are polynomial matrices in $\boldsymbol{x}(\boldsymbol{x}(t))$. $\boldsymbol{\Delta}_{ai}(\boldsymbol{x}(t))$ and $\boldsymbol{\Delta}_{bi}(\boldsymbol{x}(t))$ denote uncertain matices in $\boldsymbol{x}(t)$ and satisfy

$$\parallel \boldsymbol{\Delta}_{ai}(\boldsymbol{x}(t)) \parallel \leq \beta_{ai}(\boldsymbol{x}(t)), \tag{7}$$

$$\parallel \boldsymbol{\Delta}_{bi}(\boldsymbol{x}(t)) \parallel \leq \beta_{bi}(\boldsymbol{x}(t)), \tag{8}$$

where $\beta_{ai}(\boldsymbol{x}(t))$ and $\beta_{bi}(\boldsymbol{x}(t))$ denote the upper bound of the norm of the uncertainties.

The defuzzification process of the model (6) can be represented as

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \sum_{i=1}^{r} h_i(\boldsymbol{z}(t))\{\boldsymbol{A}_i(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{B}_i(\boldsymbol{x}(t))\boldsymbol{u}(t) \\ & \quad +\boldsymbol{D}_{ai}(\boldsymbol{x}(t))\boldsymbol{\Delta}_{ai}(\boldsymbol{x}(t))\boldsymbol{E}_{ai}(\boldsymbol{x}(t))\hat{\boldsymbol{x}}(\boldsymbol{x}(t)) \\ & \quad +\boldsymbol{D}_{bi}(\boldsymbol{x}(t))\boldsymbol{\Delta}_{bi}(\boldsymbol{x}(t))\boldsymbol{E}_{bi}(\boldsymbol{x}(t)))\boldsymbol{u}(t)\}. \end{aligned} \tag{9}$$

From now, to lighten the notation, we will drop the notation with respect to time t. For instance, we will employ $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}(\boldsymbol{x})$ instead of $\boldsymbol{x}(t)$ and $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$, respectively. Thus, we drop the notation with respect to time t, but it should be kept in mind that $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}(\boldsymbol{x})$ means $\boldsymbol{x}(t)$ and $\hat{\boldsymbol{x}}(\boldsymbol{x}(t))$, respectively.

For the model (9), we design the following fuzzy controller.

$$\boldsymbol{u} = -\sum_{i=1}^{r} h_i(\boldsymbol{z})\boldsymbol{F}_i(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \tag{10}$$

A convex SOS robust design condition for the control system consisting of (9) and (10) was presented in [36]. However, as will be mentioned in Remarks 2 and 3, some disadvantages exist in the existing SOS approaches [9]-[20] [36].

Remark 2. *In [9]-[20] and [36], the Lyapunov function candidate (11) is used.*

$$V(\boldsymbol{x}) = \hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}})\hat{\boldsymbol{x}}(\boldsymbol{x}), \tag{11}$$

where $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ is a polynomial matrix in $\tilde{\boldsymbol{x}}$. If $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x}$ and $\boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}})$ is a constant matrix, then (11) reduces to the quadratic Lyapunov function. The zero equilibrium is asymptotically stable when the Lyapunov function exists. However, the globality is not guaranteed. The stability holds globally only if $\boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}})$ is a constant matrix. The important point is that, to avoid introducing non-convex condition, $\tilde{\boldsymbol{x}}$ in the polynomial matrix $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ is defined as follows. Let $\boldsymbol{K} = \{k_1, k_2, \cdots, k_m\}$ denote the row indices of $\boldsymbol{B}_i(\boldsymbol{x})$ whose corresponding row is equal to zero, and define $\tilde{\boldsymbol{x}} = (x_{k_1}, x_{k_2}, \cdots, x_{k_m})$ using the $\boldsymbol{K}$. In other words, to avoid introducing non-convex condition, it is assumed in the literature 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. The restriction caused by $\tilde{\boldsymbol{x}}$ depends on the $\boldsymbol{B}_i(\boldsymbol{x})$ matrices and it leads to some conservative stability results. A new SOS framework that will be presented in Section III permits a non-restrictive polynomial Lyapunov function setting.

Remark 3. *As mentioned in Remark 2, (11) is employed as a candidate Lyapunov function. The transformation from non-convex conditions to convex conditions is carried out as follows. The time derivative of $V(\boldsymbol{x})$ along the feedback system trajectory, that consists of (5) and (10), can be represented by the general form.*

$$\dot{V}(\boldsymbol{x}) = \hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{S}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) < 0, \tag{12}$$

where $\boldsymbol{S}(\boldsymbol{x})$ is a non-convex polynomial matrix since it has cross terms with respect to $\boldsymbol{X}^{-1}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{F}_i(\boldsymbol{x})$. The transformation is carried out by dropping $\hat{\boldsymbol{x}}(\boldsymbol{x})$ off from both side of the inequality and by multiplying the dropped inequality on the left and right by $\boldsymbol{X}(\tilde{\boldsymbol{x}})$. As a result of the transformation, we have the following convex condition with respect to $\boldsymbol{X}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{M}_i(\boldsymbol{x})$, where $\boldsymbol{M}_i(\boldsymbol{x}) = \boldsymbol{F}_i(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}})$.

$$-\boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{S}(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) \succ \boldsymbol{0}.$$

Finally, we arrive at the convex SOS condition,

$$-\boldsymbol{v}^T\{\boldsymbol{X}(\tilde{\boldsymbol{x}})\boldsymbol{S}(\boldsymbol{x})\boldsymbol{X}(\tilde{\boldsymbol{x}}) + \epsilon(\boldsymbol{x})\boldsymbol{I}\}\boldsymbol{v} \quad is \quad SOS,$$

where $\epsilon(\boldsymbol{x})$ is a slack variable (a radially unbounded positive definite polynomial) to keep the positivity of the SOS condition. In the transformation, we utilize the fact that $-\boldsymbol{S}(\boldsymbol{x}) \succ \boldsymbol{0} \Rightarrow -\hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{S}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) > 0$. However, it should be emphasized that this is a sufficient condition, i.e., in general, it is not always satisfied that $-\hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{S}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) > 0 \Rightarrow -\boldsymbol{S}(\boldsymbol{x}) \succ \boldsymbol{0}$. It becomes a necessary and sufficient condition only if $\boldsymbol{S}(\boldsymbol{x})$ is a constant matrix and $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x}$. Only in the case, no conservatism exists. In the LMI case [2], this path is always equivalent since $\boldsymbol{S}(\boldsymbol{x})$ is a constant matrix and $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x}$. Thus, this conservative path in the convex SOS transformation often causes conservative results although this path is always equivalent in the LMI case. In [36], the same transformation is employed to obtain convex SOS robust stabilization conditions. A new SOS framework that will be presented in Section III can avoid this main problem.

A new SOS framework that will be presented in Section III can completely avoid the two problems mentioned in Remarks

2 and 3. The utility of the new SOS design framework will be demonstrated in design examples.

III. SOS Stabilization Conditions

Section III presents SOS stabilization conditions based on copositivity concept.

If (13) holds, the matrix $\boldsymbol{J} = [J_{ij}] \in \boldsymbol{R}^{\ell\times\ell}$ is copositive.

$$\boldsymbol{y}^T\boldsymbol{J}\boldsymbol{y} = \sum_{i=1}^{\ell}\sum_{j=1}^{\ell} y_i y_j J_{ij} \geq 0, \quad (13)$$

where $\boldsymbol{y} = [y_1, y_2, \ldots, y_\ell]^T \in \boldsymbol{R}^\ell$ and $y_i \geq 0$. Since checking copositivity of a matrix is a co-NP complete problem, we take a technique for copositivity checking relaxation [39].

Corollary 1. *[39]*
A relaxation is to introduce $y_i = \hat{y}_i^2$ and to check whether (14) is satisfied or not.

$$\boldsymbol{Q}^s(\hat{y}) = (\sum_{k=1}^{\ell}\hat{y}_k^2)^s \sum_{i=1}^{\ell}\sum_{j=1}^{\ell}\hat{y}_i^2\hat{y}_j^2 J_{ij} \quad is \quad SOS, \quad (14)$$

where $\hat{y} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_\ell]^T$ and s is a nonnegative integer.

By using the copositivity checking relaxation, we derive SOS robust stabilization conditions that are different from the SOS robust stabilization conditions in [36]. Theorem 1 presents the SOS robust stabilization conditions.

Theorem 1. *If there exist a polynomial $V(\boldsymbol{x})$, polynomial matrices $\boldsymbol{F}_j(\boldsymbol{x})$ and polynomials $\bar{g}_{ij}(\boldsymbol{x})$ such that (15) $\sim$ (17) are satisfied with $\alpha < 0$ and $\lambda > 0$, the polynomial fuzzy controller (10) stabilizes the system (9), and $V(\boldsymbol{x})$ becomes a Lyapunov function.*

$$V(\boldsymbol{x}) - \epsilon(\boldsymbol{x}) \quad is \quad SOS, \quad (15)$$

$$(\sum_{k=1}^{r}\hat{h}_k^2)^s \sum_{i=1}^{r}\sum_{j=1}^{r}\hat{h}_i^2\hat{h}_j^2\{-\bar{\Lambda}_{ij}(\boldsymbol{x}) + \alpha V(\boldsymbol{x})\} \quad is \quad SOS, \quad (16)$$

$$\boldsymbol{v}_1^T\boldsymbol{L}_{ij}(\lambda,\boldsymbol{x})\boldsymbol{v}_1 \quad is \quad SOS, \quad (17)$$

where $\boldsymbol{v}_1$ denotes vector that is independent of $\boldsymbol{x}$. $\epsilon(\boldsymbol{x})$ is a radially unbounded positive definite polynomial and s is a non-negative integer.

$$\bar{\Lambda}_{ij}(\boldsymbol{x}) = \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\{\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\}\hat{\boldsymbol{x}}(\boldsymbol{x}) + \bar{g}_{ij}(\boldsymbol{x}), \quad (18)$$

$$\boldsymbol{L}_{ij}(\lambda,\boldsymbol{x}) = \begin{bmatrix} \lambda\bar{g}_{ij}(\boldsymbol{x}) & * & * & * & * \\ \lambda\boldsymbol{D}_{ai}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \lambda\boldsymbol{D}_{bi}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \beta_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} \\ \beta_{bi}(\boldsymbol{x})\boldsymbol{E}_{bi}(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I} \end{bmatrix}. \quad (19)$$

The asterisk $*$ denotes the transposed elements (matrices) for symmetric positions.

Proof:
Consider a candidate of Lyapunov functions $V(\boldsymbol{x})$. The time derivative of $V(\boldsymbol{x})$ is given as

$$\dot{V}(\boldsymbol{x}) = \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\dot{\boldsymbol{x}}. \quad (20)$$

By subsitituting the closed loop dynamics consisting of (9) and (10) into (20), the time derivative of $V(\boldsymbol{x})$ along the trajectory becomes

$$\begin{aligned}\dot{V}(\boldsymbol{x}) =& \sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\{\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}) \\ &+\boldsymbol{D}_{ai}(\boldsymbol{x})\boldsymbol{\Delta}_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x}) \\ &-\boldsymbol{D}_{bi}(\boldsymbol{x})\boldsymbol{\Delta}_{bi}(\boldsymbol{x})\boldsymbol{E}_{bi}(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\}\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ =& \sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j\{\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}(\boldsymbol{A}_i(\boldsymbol{x}) \\ &-\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) + \boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x})\},\end{aligned}$$

where

$$\boldsymbol{\zeta}_i(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{D}_{ai}(\boldsymbol{x}) & \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{D}_{bi}(\boldsymbol{x}) \end{bmatrix},$$

$$\boldsymbol{\eta}_{ij}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{\Delta}_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ -\boldsymbol{\Delta}_{bi}(\boldsymbol{x})\boldsymbol{E}_{bi}(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \end{bmatrix}.$$

Note that

$$\begin{aligned}\lambda\boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\zeta}_i^T(\boldsymbol{x}) &+ \frac{1}{\lambda}\boldsymbol{\eta}_{ij}^T(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x}) \\ &\geq \boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x}) + \boldsymbol{\eta}_{ij}^T(\boldsymbol{x})\boldsymbol{\zeta}_i^T(\boldsymbol{x})\end{aligned}$$

for any $\lambda > 0$. In addition, since $\boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x}) = \boldsymbol{\eta}_{ij}^T(\boldsymbol{x})\boldsymbol{\zeta}_i^T(\boldsymbol{x})$, we have the following relation.

$$\lambda\boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\zeta}_i^T(\boldsymbol{x}) + \frac{1}{\lambda}\boldsymbol{\eta}_{ij}^T(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x}) \geq 2\boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x}).$$

Hence

$$\begin{aligned}&\boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x}) \\ &\leq \frac{\lambda}{2}\boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\zeta}_i^T(\boldsymbol{x}) + \frac{1}{2\lambda}\boldsymbol{\eta}_{ij}^T(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x}) \\ &= \frac{\lambda}{2}\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{D}_{ai}(\boldsymbol{x})\boldsymbol{D}_{ai}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T \\ &+ \frac{\lambda}{2}\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{D}_{bi}(\boldsymbol{x})\boldsymbol{D}_{bi}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T \\ &+ \frac{1}{2\lambda}\hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{E}_{ai}^T(\boldsymbol{x})\boldsymbol{\Delta}_{ai}^T(\boldsymbol{x})\boldsymbol{\Delta}_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ &+ \frac{1}{2\lambda}\hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{F}_j^T(\boldsymbol{x})\boldsymbol{E}_{bi}^T(\boldsymbol{x})\boldsymbol{\Delta}_{bi}^T(\boldsymbol{x}) \\ &\quad\times\boldsymbol{\Delta}_{bi}(\boldsymbol{x})\boldsymbol{E}_{bi}(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ &\leq \boldsymbol{\Pi}_{ij}(\lambda,\boldsymbol{x}),\end{aligned}$$

where

$$\begin{aligned}\boldsymbol{\Pi}_{ij}(\lambda,\boldsymbol{x}) =& \frac{\lambda}{2}\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{D}_{ai}(\boldsymbol{x})\boldsymbol{D}_{ai}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T \\ &+ \frac{\lambda}{2}\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{D}_{bi}(\boldsymbol{x})\boldsymbol{D}_{bi}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T \\ &+ \frac{1}{2\lambda}\boldsymbol{\beta}_{ai}^2(\boldsymbol{x})\hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{E}_{ai}^T(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ &+ \frac{1}{2\lambda}\boldsymbol{\beta}_{bi}^2(\boldsymbol{x})\hat{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{F}_j^T(\boldsymbol{x})\boldsymbol{E}_{bi}^T(\boldsymbol{x})\boldsymbol{E}_{bi}(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}).\end{aligned}$$

From the above inequality, we have

$$\begin{aligned}\dot{V}(\boldsymbol{x}) &= \sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \{ \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}} (\boldsymbol{A}_i(\boldsymbol{x}) \\ &\quad -\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) + \boldsymbol{\zeta}_i(\boldsymbol{x})\boldsymbol{\eta}_{ij}(\boldsymbol{x})\} \\ &\le \sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \{ \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}} (\boldsymbol{A}_i(\boldsymbol{x}) \\ &\quad -\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) + \boldsymbol{\Pi}_{ij}(\lambda, \boldsymbol{x})\}.\end{aligned}$$

We introduce a polynomial $\bar{g}_{ij}(\boldsymbol{x})$ satisfying

$$\sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \boldsymbol{\Pi}_{ij}(\lambda, \boldsymbol{x}) \le \sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \bar{g}_{ij}(\boldsymbol{x}). \tag{21}$$

Then, we have

$$\begin{aligned}\dot{V}(\boldsymbol{x}) &\le \sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \{ \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}} (\boldsymbol{A}_i(\boldsymbol{x}) \\ &\quad -\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) + \bar{g}_{ij}(\boldsymbol{x})\}. \end{aligned} \tag{22}$$

To show that $\dot{V}(\boldsymbol{x}) < 0$ at $\boldsymbol{x} \neq 0$, we consider the condition satisfying $\dot{V}(\boldsymbol{x}) \le \alpha V(\boldsymbol{x})$, where $\alpha < 0$. That is,

$$\begin{aligned}&\sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \{ \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}} (\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ &+\bar{g}_{ij}(\boldsymbol{x})\} - \alpha V(\boldsymbol{x}) \le 0.\end{aligned}$$

By applying the copositivity presented in Lemma 1, we obtain (16).

On the other hand, from the inequality (21) and $\lambda > 0$, we obtain

$$\sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \{\lambda \bar{g}_{ij}(\boldsymbol{x}) - \lambda \boldsymbol{\Pi}_{ij}(\lambda, \boldsymbol{x})\} \ge 0. \tag{23}$$

Using schur complement, (23) can be converted to

$$\sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \boldsymbol{L}_{ij}(\lambda, \boldsymbol{x}) \ge \boldsymbol{0}, \tag{24}$$

where

$$\boldsymbol{L}_{ij}(\lambda, \boldsymbol{x}) = \begin{bmatrix} \lambda \bar{g}_{ij}(\boldsymbol{x}) & * & * & * & * \\ \lambda \boldsymbol{D}_{ai}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \lambda \boldsymbol{D}_{bi}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \beta_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} \\ \beta_{bi}(\boldsymbol{x})\boldsymbol{E}_{bi}(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I} \end{bmatrix}.$$

The condition (24) holds if (17) is satisfied. ■

Theorem 2. *Assume that $\boldsymbol{\Delta}_{bi}(\boldsymbol{x}) = \boldsymbol{0}$ for all i, i.e., there are no uncertainties with respect to the input terms. Then, the SOS robust stabilization conditions in Theorem 1 become simple. If there exist a polynomial function $V(\boldsymbol{x})$, polynomial matrices $\boldsymbol{F}_j(\boldsymbol{x})$ and polynomials $\bar{g}_i(\boldsymbol{x})$ such that (25) $\sim$ (27) are satisfied with $\alpha < 0$ and $\lambda > 0$, the polynomial fuzzy controller (10) stabilizes the system (9).*

$$V(\boldsymbol{x}) - \epsilon(\boldsymbol{x}) \quad is \quad SOS, \tag{25}$$

$$(\sum_{k=1}^{r}\hat{h}_k^2)^s \sum_{i=1}^{r}\sum_{j=1}^{r} \hat{h}_i^2 \hat{h}_j^2 \{-\bar{\Lambda}_{ij}(\boldsymbol{x}) + \alpha V(\boldsymbol{x})\} \quad is \quad SOS, \tag{26}$$

$$\boldsymbol{v}_1^T \begin{bmatrix} \lambda \bar{g}_i(\boldsymbol{x}) & * & * \\ \lambda \boldsymbol{D}_{ai}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}})^T & 2\boldsymbol{I} & \boldsymbol{0} \\ \beta_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & 2\boldsymbol{I} \end{bmatrix} \boldsymbol{v}_1 \quad is \quad SOS, \tag{27}$$

where $\epsilon(\boldsymbol{x})$ is a radially unbounded positive definite polynomial, s is a non-negative integer, and

$$\begin{aligned}\bar{\Lambda}_{ij}(\boldsymbol{x}) = &\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}} \{\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\}\hat{x}(\boldsymbol{x}) \\ &+\bar{g}_i(\boldsymbol{x}).\end{aligned} \tag{28}$$

Proof: The proof is omitted since it is directly obtained from Theorem 1. In this case, (17) is reduced to (27). ■

IV. Algorithm to Solve SOS Conditions

Section IV presents an algorithm to solve the SOS robust stabilization conditions given in Section III. The algorithm is based on a gradient algorithm formulated as a minimizing optimization problem of the upper bound of the time derivative of an SOS polynomial that can be regarded as a candidate of polynomial Lyapunov functions.

We first explain the outline of its key idea below.

A. Key Idea

Consider the non-convex condition

$$\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) \prec 0, \tag{29}$$

where $\phi_g(\boldsymbol{x})$ and $\phi_h(\boldsymbol{x})$ are polynomial matrices in $\boldsymbol{x}$ and both of them are decision variables (matrices). The problem is to find a solution satisfying (29). With a positive definite polynomial matrix $\psi(\boldsymbol{x})$ in $\boldsymbol{x}$, the problem (29) may be converted as

$$-\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) + \alpha\psi(\boldsymbol{x}) \succeq 0. \tag{30}$$

If we get a solution of (30) with $\alpha < 0$, the problem (29) is feasible. Regularly, (30) can be converted as

'$-\boldsymbol{v}^T\{\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) - \alpha\psi(\boldsymbol{x})\}\boldsymbol{v}$ is SOS',

where $\boldsymbol{v}$ denotes a vector that is independent of $\boldsymbol{x}$. Note that the SOS condition is bilinear (not convex) with respect to decision variables since there exists the term $\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x})$. Now consider very small perturbations $\delta\phi_g(\boldsymbol{x})$, $\delta\phi_h(\boldsymbol{x})$ and $\delta\psi(\boldsymbol{x})$ as in [42], [43]. Since $\delta\phi_g(\boldsymbol{x})$ and $\delta\phi_h(\boldsymbol{x})$ are very small perturbations, it can be noted with a reasonable approximation that

$$\begin{aligned}\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) &\simeq (\phi_g(\boldsymbol{x}) + \delta\phi_g(\boldsymbol{x}))(\phi_h(\boldsymbol{x}) + \delta\phi_h(\boldsymbol{x})) \\ &= \phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) + \delta\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) \\ &\quad +\phi_g(\boldsymbol{x})\delta\phi_h(\boldsymbol{x}) + \delta\phi_g(\boldsymbol{x})\delta\phi_h(\boldsymbol{x}).\end{aligned}$$

Note that the term, $\delta\phi_g(\boldsymbol{x})\delta\phi_h(\boldsymbol{x})$, is an extremely small in comparison with other terms since it is the product term of these small perturbations. Then, $(\phi_g(\boldsymbol{x}) + \delta\phi_g(\boldsymbol{x}))(\phi_h(\boldsymbol{x}) + \delta\phi_h(\boldsymbol{x}))$ can be represented as $\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) + \delta\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x}) + \phi_g(\boldsymbol{x})\delta\phi_h(\boldsymbol{x})$. From this fact, we transform

'$-\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x})+\alpha\psi(\boldsymbol{x})\succeq 0$'
to

$$\text{'}-\boldsymbol{v}^T\{\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x})+\delta\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x})+\phi_g(\boldsymbol{x})\delta\phi_h(\boldsymbol{x}) \\ -\alpha\psi(\boldsymbol{x})-\alpha\delta\psi(\boldsymbol{x})\}\boldsymbol{v}\ is\ SOS'. \quad (31)$$

Now we can formulate (31) as a minimizing optimization problem based on convex SOS with respect to $\delta\phi_g(\boldsymbol{x})$, $\delta\phi_h(\boldsymbol{x})$ and $\delta\psi(\boldsymbol{x})$.

$$\begin{aligned}
&\min_{\delta\phi_g(\boldsymbol{x}),\delta\phi_h(\boldsymbol{x}),\delta\psi(\boldsymbol{x})} \alpha \\
&\quad subject\ to \\
&\boldsymbol{v}_1^T\{\psi(\boldsymbol{x})+\delta\psi(\boldsymbol{x})-\epsilon(\boldsymbol{x})\}\boldsymbol{v}_1 \quad is \quad SOS, && (32)\\
&-\boldsymbol{v}_2^T\{\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x})+\delta\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x})+\phi_g(\boldsymbol{x})\delta\phi_h(\boldsymbol{x}) \\
&\quad -\alpha\psi(\boldsymbol{x})-\alpha\delta\psi(\boldsymbol{x})\}\boldsymbol{v}_2 \quad is \quad SOS, && (33)\\
&\boldsymbol{v}_3^T\begin{bmatrix} \epsilon_G\phi_g^T(\boldsymbol{x})\phi_g(\boldsymbol{x}) & \delta\phi_g(\boldsymbol{x}) \\ \delta\phi_g(\boldsymbol{x}) & \boldsymbol{I}\end{bmatrix}\boldsymbol{v}_3 \quad is \quad SOS, && (34)\\
&\boldsymbol{v}_4^T\begin{bmatrix} \epsilon_H\phi_h^T(\boldsymbol{x})\phi_h(\boldsymbol{x}) & \delta\phi_h(\boldsymbol{x}) \\ \delta\phi_h(\boldsymbol{x}) & \boldsymbol{I}\end{bmatrix}\boldsymbol{v}_4 \quad is \quad SOS, && (35)\\
&\boldsymbol{v}_5^T\begin{bmatrix} \epsilon_\psi\psi^T(\boldsymbol{x})\psi(\boldsymbol{x}) & \delta\psi(\boldsymbol{x}) \\ \delta\psi(\boldsymbol{x}) & \boldsymbol{I}\end{bmatrix}\boldsymbol{v}_5 \quad is \quad SOS, && (36)
\end{aligned}$$

where $\boldsymbol{v}_1$ - $\boldsymbol{v}_5$ denote vectors that are independent of $\boldsymbol{x}$. ϵ_G, ϵ_H and ϵ_ψ are very small positive values. $\epsilon(\boldsymbol{x})$ is a radially unbounded positive definite polynomial. (34), (35) and (36) guarantee to keep the assumption that $\delta\phi_g(\boldsymbol{x})$, $\delta\phi_h(\boldsymbol{x})$ and $\delta\psi(\boldsymbol{x})$ are very small perturbations, respectively.

Note that the decision variables are $\delta\phi_g(\boldsymbol{x})$, $\delta\phi_h(\boldsymbol{x})$ and $\delta\psi(\boldsymbol{x})$ in the minimizing optimization. The minimizing optimization is iteratively performed by substituting the solutions
$\delta\phi_g(\boldsymbol{x})$, $\delta\phi_h(\boldsymbol{x})$ and $\delta\psi(\boldsymbol{x})$
obtained at the Nth iteration into the iteration law

$$\begin{aligned}
\phi_g^{N+1}(\boldsymbol{x}) &= \phi_g^N(\boldsymbol{x})+\delta\phi_g(\boldsymbol{x}),\\
\phi_h^{N+1}(\boldsymbol{x}) &= \phi_h^N(\boldsymbol{x})+\delta\phi_h(\boldsymbol{x}),\\
\psi^{N+1}(\boldsymbol{x}) &= \psi^N(\boldsymbol{x})+\delta\psi(\boldsymbol{x}).
\end{aligned}$$

Thus, the decision variables are updated so as to minimize the minimizing parameter α. As a result, $\phi_g(\boldsymbol{x})$, $\phi_h(\boldsymbol{x})$ and $\psi(\boldsymbol{x})$ are iteratively updated from the initial setting ($\phi_g^0(\boldsymbol{x})$, $\phi_h^0(\boldsymbol{x})$ and $\psi^0(\boldsymbol{x})$) so as to minimize the minimizing parameter α. The initial setting of $\phi_g^0(\boldsymbol{x})$ $\phi_h^0(\boldsymbol{x})$ and $\psi^0(\boldsymbol{x})$ should be sometimes carefully selected. So the grid search will be employed to select the initial setting. If the minimizing optimization problem is feasible with $\alpha<0$, it is a solution of (29), i.e., $\phi_g(\boldsymbol{x})\phi_h(\boldsymbol{x})\prec 0$.

B. Algorithm

We can consider

$$V(\boldsymbol{x})=\bar{\boldsymbol{x}}^T(\boldsymbol{x})\boldsymbol{P}\bar{\boldsymbol{x}}(\boldsymbol{x}),$$

where $\boldsymbol{P}\in\boldsymbol{R}^{\rho\times\rho}$ is a positive definite matrix and $\bar{\boldsymbol{x}}(\boldsymbol{x})\in\boldsymbol{R}^\rho$ is a column vecrtor whose entries are all monomials in $\boldsymbol{x}$ such that $\bar{\boldsymbol{x}}(\boldsymbol{x})=\boldsymbol{0}$ iff $\boldsymbol{x}=\boldsymbol{0}$ and $||\bar{\boldsymbol{x}}(\boldsymbol{x})||\to\infty$ for $||\boldsymbol{x}(\boldsymbol{x})||\to\infty$. For example, if we choose the vector $\bar{\boldsymbol{x}}(\boldsymbol{x})=[x_1\ x_2]$ in the case of $\boldsymbol{x}=[x_1\ x_2]$, $V(\boldsymbol{x})$ becomes a quadratic Lyapunov function. If $\bar{\boldsymbol{x}}(\boldsymbol{x})=[x_1^2\ \ x_1x_2\ \ x_2^2]$ is chosen, $V(\boldsymbol{x})$ becomes a 4th-order polynomial Lyapunov function.

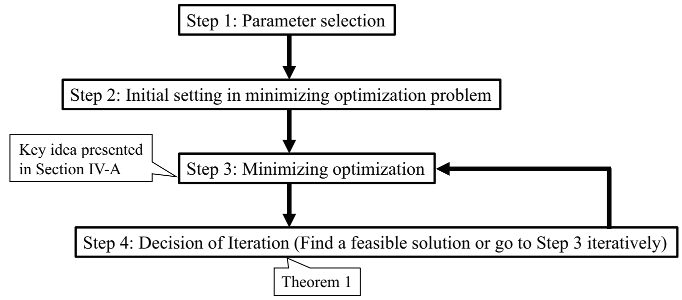

Fig. 1. Outline of algorithm.

The algorithm to solve the SOS conditions consists of four steps. Fig. 1 shows the outline of the algorithm. The key idea mentioned in Section IV-A will be used in Step 3. We check whether the SOS conditions given in Theorem 1 are strictly and exactly feasible or not in Step 4. This algorithm can be regarded as a gradient algorithm formulated as a minimizing optimization problem of the upper bound of the time derivative of the polynomial $V(\boldsymbol{x})$. Table I summarizes main variables and parameters in the minimizing optimization algorithm, where p_i is the i-th diagonal element of the positive definite matrix $\boldsymbol{P}$. For simplicity, all the non-diagonal elements of the positive definite matrix $\boldsymbol{P}$ are set to zero in the initial setting. However, note that, after performing the algorithm, the nondiagonal elements of the matrix $\boldsymbol{P}$ can become non-zero. In fact, the complicated example in Section VII obtains the matrix $\boldsymbol{P}$ whose nondiagonal elements are non-zero although the nondiagonal elements of the matrix $\boldsymbol{P}$ are set to zero in the initial setting.

TABLE I
LIST OF MAIN VARIABLES AND PARAMETERS IN MINIMIZING OPTIMIZATION ALGORITHM.

N	number of iteration
λ^{min}, λ^{max}	lower and upper bounds of λ satisfying $0<\lambda^{min}\le\lambda\le\lambda^{max}$
p_i^{min}, p_i^{max}	lower and upper bounds of p_i satisfying $0<p_i^{min}\le p_i\le p_i^{max}$
q_λ, $\Delta\lambda$	number of divided segments and interval such that $q_\lambda\Delta\lambda=\lambda^{max}-\lambda^{min}$
q_{pi}, Δp_i	number of divided segments and intervals such that $q_{pi}\Delta p_i=p_i^{max}-p_i^{min}$

Step 1: Set $N=0$. Select positive scalars λ^{min}, λ^{max}, $\Delta\lambda$ and Δp_i $(i=1,2,\cdots\rho)$ satisfying the relations defined in Table I.

Step 2: For all the combinations $(\lambda,p_1,p_2,\cdots,p_\rho)$ on all the grid points $[\lambda^{min}\ \lambda^{max}]\times[p_1^{min}\ p_2^{max}]\times\cdots\times[p_\rho^{min}\ p_\rho^{max}]$ with the intervals $\Delta\lambda$, Δp_1, Δp_2, $\cdots$ Δp_ρ, solve

$$\min_{\boldsymbol{F}_j(\boldsymbol{x}),\bar{g}_{ij}(\boldsymbol{x})} \alpha \quad subject\ to\ (15),(16)\ and\ (17) \qquad (37)$$

and find the grid point with the minimum α. If a grid point with $\alpha<0$ is found, it is a strict solution of Theorem 1. If any feasible solutions with $\alpha<0$ are not obtained, then substitute $\boldsymbol{F}_j(\boldsymbol{x})$, $\bar{g}_{ij}(\boldsymbol{x})$, $V(\boldsymbol{x})$ and λ obtained at the minimum grid point into $\boldsymbol{F}_j^N(\boldsymbol{x})$, $\bar{g}_{ij}^N(\boldsymbol{x})$, $V^N(\boldsymbol{x})$ and λ^N, respectively, and go to Step 3.

Step 3: Set $\boldsymbol{F}_j(\boldsymbol{x}) = \boldsymbol{F}_j^N(\boldsymbol{x})$, $\bar{g}_{ij}(\boldsymbol{x}) = \bar{g}_{ij}^N(\boldsymbol{x})$, $V(\boldsymbol{x}) = V^N(\boldsymbol{x})$ and $\lambda = \lambda^N$. For the given $\boldsymbol{F}_j(\boldsymbol{x})$, $\bar{g}_{ij}(\boldsymbol{x})$, $V(\boldsymbol{x})$, and λ, solve the following SOS optimization problem.

$$\min_{\delta \boldsymbol{F}_j(\boldsymbol{x}),\delta V(\boldsymbol{x}),\delta \bar{g}_{ij}(\boldsymbol{x}),\delta\lambda} \alpha \qquad \textit{subject to } (38) \sim (44)$$

The SOS conditions (38) $\sim$ (44) are derived by applying the key idea to Theorem 1.

$$V(\boldsymbol{x}) + \delta V(\boldsymbol{x}) - \epsilon(\boldsymbol{x}) \;\; is \;\; SOS, \tag{38}$$

$$\begin{aligned}&(\sum_{k=1}^{r} \hat{h}_k^2)^s \sum_{i=1}^{r}\sum_{j=1}^{r} \hat{h}_i^2 \hat{h}_j^2 \{-(\bar{\Lambda}_{ij}(\boldsymbol{x}) + \delta\bar{\Lambda}_{ij}(\boldsymbol{x}))\\ &\qquad +\alpha(\delta V(\boldsymbol{x}) + V(\boldsymbol{x}))\} \;\; is \;\; SOS,\end{aligned} \tag{39}$$

$$\begin{aligned}&(\sum_{k=1}^{r} \hat{h}_k^2)^s \sum_{i=1}^{r}\sum_{j=1}^{r} \hat{h}_i^2 \hat{h}_j^2 \times\\ &\boldsymbol{v}_1^T \begin{bmatrix} \lambda\bar{g}_{ij}(\boldsymbol{x}) + \delta\lambda\bar{g}_{ij}(\boldsymbol{x}) + \lambda\delta\bar{g}_{ij}(\boldsymbol{x}) & * & * & * & * \\ \boldsymbol{D}_{ai}^T(\boldsymbol{x})\boldsymbol{\mu}^T(\boldsymbol{x}) & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{D}_{bi}^T(\boldsymbol{x})\boldsymbol{\mu}^T(\boldsymbol{x}) & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \beta_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} \\ \beta_{bi}(\boldsymbol{x})\boldsymbol{E}_{bi}(\boldsymbol{x})(\boldsymbol{F}_j(\boldsymbol{x}) + \delta\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I} \end{bmatrix} \boldsymbol{v}_1 \;\; is \;\; SOS,\end{aligned} \tag{40}$$

$$\boldsymbol{v}_2^T \begin{bmatrix} \epsilon_V V^2(\boldsymbol{x}) & \delta V(\boldsymbol{x}) \\ \delta V(\boldsymbol{x}) & I \end{bmatrix} \boldsymbol{v}_2 \;\; is \;\; SOS, \tag{41}$$

$$\boldsymbol{v}_3^T \begin{bmatrix} \epsilon_F \boldsymbol{F}_j^T(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}) & \delta\boldsymbol{F}_j(\boldsymbol{x}) \\ \delta\boldsymbol{F}_j^T(\boldsymbol{x}) & \boldsymbol{I} \end{bmatrix} \boldsymbol{v}_3 \;\; is \;\; SOS, \tag{42}$$

$$\boldsymbol{v}_4^T \begin{bmatrix} \epsilon_g \bar{g}_{ij}^2(\boldsymbol{x}) & \delta\bar{g}_{ij}(\boldsymbol{x}) \\ \delta\bar{g}_{ij}(\boldsymbol{x}) & I \end{bmatrix} \boldsymbol{v}_4 \;\; is \;\; SOS, \tag{43}$$

$$\boldsymbol{v}_5^T \begin{bmatrix} \epsilon_L \lambda^2 & \delta\lambda \\ \delta\lambda & I \end{bmatrix} \boldsymbol{v}_5 \;\; is \;\; SOS, \tag{44}$$

where

$$\begin{aligned}\delta\bar{\Lambda}_{ij}(\boldsymbol{x}) &= \frac{\partial \delta V(\boldsymbol{x})}{\partial \boldsymbol{x}}\{\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x})\}\hat{\boldsymbol{x}}(\boldsymbol{x})\\ &\quad -\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{B}_i(\boldsymbol{x})\delta\boldsymbol{F}_j(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) + \delta\bar{g}_{ij}(\boldsymbol{x}),\\ \boldsymbol{\mu}(\boldsymbol{x}) &= \lambda(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}) + \delta\lambda(\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}) + \lambda(\frac{\partial \delta V(\boldsymbol{x})}{\partial \boldsymbol{x}}).\end{aligned}$$

$\boldsymbol{v}_1$ - $\boldsymbol{v}_5$ denote vectors that are independent of $\boldsymbol{x}$. ϵ_V, ϵ_F, ϵ_g and ϵ_L are very small positive values and s is a non-negative integer.

Step 4: For $\delta V(\boldsymbol{x})$ and $\delta\lambda$ obtained by solving the SOS optimization problem in Step 3, let $V^{N+1}(\boldsymbol{x}) = V^N(\boldsymbol{x}) + \delta V(\boldsymbol{x})$ and $\lambda^{N+1} = \lambda^N + \delta\lambda$, respectively. Then set $N = N + 1$. Next, set $V(\boldsymbol{x}) = V^N(\boldsymbol{x})$ and $\lambda = \lambda^N$. For the given $V(\boldsymbol{x})$ and λ, solving the minimizing SOS problem (45).

$$\min_{\boldsymbol{F}_j(\boldsymbol{x}),\bar{g}_{ij}(\boldsymbol{x})} \alpha \;\; \textit{subject to } (15), (16) \textit{ and } (17) \tag{45}$$

If a feasible solution with $\alpha < 0$ is obtained, it is a strict solution of Theorem 1. If any feasible solutions with $\alpha < 0$ are not obtained, then substitute $\boldsymbol{F}_j(\boldsymbol{x})$ and $\bar{g}_{ij}(\boldsymbol{x})$ obtained by solving (45) into $\boldsymbol{F}_j^N(\boldsymbol{x})$ and $\bar{g}_{ij}^N(\boldsymbol{x})$, respectively, and go to Step 3.

Remark 4. *Assume that $\boldsymbol{\Delta}_{bi}(\boldsymbol{x}) = \boldsymbol{0}$ for all i, i.e., there are no uncertainties with respect to the input terms. Then, (40) and (43) can be simplified as (46) and (47), respectively.*

$$\begin{aligned}&(\sum_{k=1}^{r} \hat{h}_k^2)^s \sum_{i=1}^{r}\sum_{j=1}^{r} \hat{h}_i^2 \hat{h}_j^2 \times\\ &\boldsymbol{v}_1^T \begin{bmatrix} \lambda\bar{g}_i(\boldsymbol{x}) + \delta\lambda\bar{g}_i(\boldsymbol{x}) + \lambda\delta\bar{g}_i(\boldsymbol{x}) & * & * \\ \boldsymbol{D}_{ai}^T(\boldsymbol{x})\boldsymbol{\mu}^T(\boldsymbol{x}) & 2\boldsymbol{I} & \boldsymbol{0} \\ \beta_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & 2\boldsymbol{I} \end{bmatrix} \boldsymbol{v}_1 \;\; is \;\; SOS,\end{aligned} \tag{46}$$

$$\boldsymbol{v}_4^T \begin{bmatrix} \epsilon_g \bar{g}_i^2(\boldsymbol{x}) & \delta\bar{g}_i(\boldsymbol{x}) \\ \delta\bar{g}_i(\boldsymbol{x}) & I \end{bmatrix} \boldsymbol{v}_4 \;\; is \;\; SOS. \tag{47}$$

Remark 5. *We need to carefully deal with SOS solutions since some numerical reliability options exist in the SOS solvers and their feasible results might be changed very slightly according to the options, particularly, for complicated systems. In other words, feasible area plots (, e.g., such as Figs. 4 and 5) might change very slightly according to the options. To obtain more reliable solutions for SOS conditions, we perform the following double checking throughout this paper. After getting a feasible solution in the algorithm, we carefully perform the so-called SOS test (, e.g., 'issos' command in SOSOPT) for the polynomials calculated by substituting the feasible solution into the considered SOS conditions. That is, with one of most reliable options, we check whether the polynomials (calculated by substituting the feasible solution into the considered SOS conditions) are judged as SOS polynomials or not. If the check returns an infeasible result, we strictly judge 'infeasible'. This double checking is important to have reliable solutions in the use of SOSOPT[38] or SOSTOOLS [39] and an SDP solver [40], [41]*

V. DESIGN EXAMPLES

A. Design Example I

Consider the following nonlinear system with an uncertainty.

$$\begin{aligned}\dot{x}_1 &= (-1 + \Delta(t) + x_1 + x_1^2 + x_1x_2 - x_2^2)x_1\\ &\quad +x_2 + x_1 u,\\ \dot{x}_2 &= -2\sin(x_1) - 6x_2 + 7u,\end{aligned} \tag{48}$$

where $\Delta(t)$ is the uncertainty satisfying $|\Delta(t)| \leq c$ for all t. All the simulation results given in this design example are carried out for $\Delta(t) = c\sin(200\pi t)$. However, it should be noted that $\Delta(t)$ is the uncertain term and only its upper bound, i.e. c, is known as well as the standard robust control setting.

Using the sector nonlinearity technique [2], the nonlinear system with the uncertainty is exactly converted into the following two-rule polynomial fuzzy system with uncertainties:

$$\begin{aligned}\dot{\boldsymbol{x}} = \sum_{i=1}^{r} h_i(\boldsymbol{z})\{&(\boldsymbol{A}_i(\boldsymbol{x}) + \boldsymbol{D}_{ai}(\boldsymbol{x})\boldsymbol{\Delta}_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x}))\boldsymbol{x}\\ &+\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{u})\},\end{aligned}$$

where r=2, $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x} = [x_1,\ x_2]$, $\boldsymbol{z} = x_1$, and

$$\boldsymbol{A}_1(\boldsymbol{x}) = \begin{bmatrix} -1 + x_1 + x_1^2 + x_1x_2 - x_2^2 & 1 \\ -2 & -6 \end{bmatrix},$$
$$\boldsymbol{A}_2(\boldsymbol{x}) = \begin{bmatrix} -1 + x_1 + x_1^2 + x_1x_2 - x_2^2 & 1 \\ 0.4344 & -6 \end{bmatrix},$$
$$\boldsymbol{B}_1(\boldsymbol{x}) = \boldsymbol{B}_2(\boldsymbol{x}) = \begin{bmatrix} x_1 \\ 7 \end{bmatrix},$$
$$\boldsymbol{D}_{a1}(\boldsymbol{x}) = \boldsymbol{D}_{a2}(\boldsymbol{x}) = \begin{bmatrix} c \\ 0 \end{bmatrix},$$
$$\boldsymbol{\Delta}_{a1}(\boldsymbol{x}) = \boldsymbol{\Delta}_{a2}(\boldsymbol{x}) = \Delta(t)/c,$$
$$\boldsymbol{E}_{a1}(\boldsymbol{x}) = \boldsymbol{E}_{a2}(\boldsymbol{x}) = \begin{bmatrix} 1 & 0 \end{bmatrix},$$
$$h_1(\boldsymbol{z}) = \frac{\sin(x_1) + 0.2172x_1}{1.2172x_1},\ h_2(\boldsymbol{z}) = \frac{x_1 - \sin(x_1)}{1.2172x_1}.$$

Since $||\boldsymbol{\Delta}_{a1}(\boldsymbol{x})|| = ||\boldsymbol{\Delta}_{a2}(\boldsymbol{x})|| = ||\Delta(t)/c|| \leq 1$, we have $\beta_{a1} = \beta_{a2} = 1$. Moreover, the algorithm presented in Section IV is carried out with the initial setting $s = 0$, $\epsilon_g = 0.001$, $\epsilon_F = 0.001$, $\epsilon_V = 0.001$, $\epsilon_L = 0.001$, $\lambda^{min} = 0.2$, $\lambda^{max} = 5$, $\Delta\lambda = 0.8$, $p_i^{min} = 0.2$, $p_i^{max} = 1$, $\Delta p_i = 0.2$ for $i = 1, 2$. To show the validity of derived conditions, we compare the feasible values of c for the proposed robust control design method and the SOS-based design method of [36]. The proposed robust control design method is feasible for $c \leq 0.76$, and the method of [36] is feasible for $c \leq 0.39$. It shows that the proposed robust design method provides more relaxed results than the method of [36].

TABLE II
FEASIBLE AREAS FOR c.

Convex SOS robust [36]	$c \leq 0.39$
Our SOS robust	$c \leq 0.76$

For $c = 0.76$, Fig. 2 shows the behavior of the nonlinear system (48) with $u = 0$. Thus, the system is unstable when $u = 0$. By solving the conditions in Theorem 2, a feasible solution for $c = 0.76$ can be obtained as

$$\lambda = 0.6811,$$
$$V(\boldsymbol{x}) = 1.0524x_1^2 + 0.1361x_2^2$$
$$F_1(\boldsymbol{x}) = \begin{bmatrix} 1.6566x_1 + 0.2669x_2 + 0.8952 \\ 0.2669x_1 - 0.1902 \end{bmatrix}^T,$$
$$F_2(\boldsymbol{x}) = \begin{bmatrix} 1.6314x_1 + 0.2696x_2 + 1.0418 \\ 0.2696x_1 - 0.2061 \end{bmatrix}^T,$$
$$\begin{aligned} \bar{g}_1(\boldsymbol{x}) &= 0.6850x_1^4 - 0.5481x_1^3x_2 + 1.1416x_1^2x_2^2 \\ &- 0.0932x_1^3 + 1.4081x_1^2x_2 + 0.5751x_1x_2^2 \\ &+ 1.8407x_1^2 + 0.0615x_1x_2 + 0.6297x_2^2, \\ \bar{g}_2(\boldsymbol{x}) &= 0.6797x_1^4 - 0.5397x_1^3x_2 + 1.1377x_1^2x_2^2 \\ &+ 0.0217x_1^3 + 1.3704x_1^2x_2 + 0.5679x_1x_2^2 \\ &+ 1.8395x_1^2 - 0.0914x_1x_2 + 0.6296x_2^2. \end{aligned}$$

Fig. 3 shows the controlled behavior for six different initial conditions. It can be seen from Fig. 3 that the design fuzzy controller stabilizes the system from all the initial conditions although the system has uncertainties.

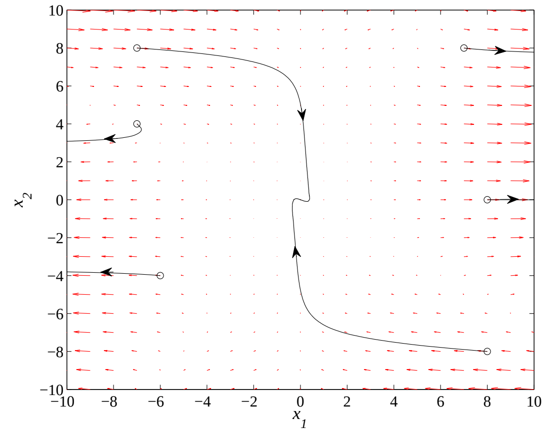

Fig. 2. Behavior of the nonlinear system (48) with $u = 0$.

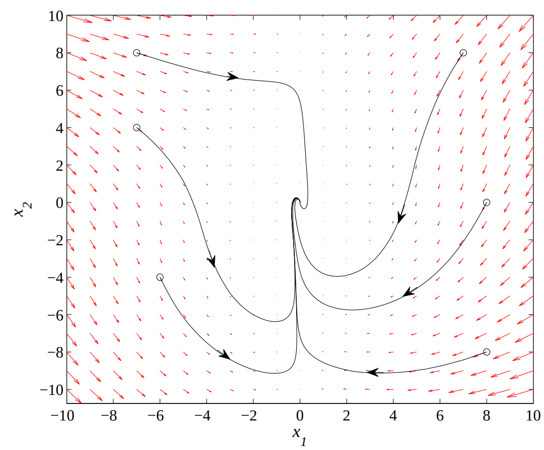

Fig. 3. Controlled behavior of the nonlinear system (48).

Based on the sector nonlinearity technique [2], the nonlinear system (48) can be exactly represented by a T-S 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 constant satisfying $0 < d_1 < \infty$ and $0 < d_2 < \infty$. The T-S fuzzy model is obtained as

$$\sum_{i=1}^{8} h_i(\boldsymbol{z})\{(\boldsymbol{A}_i + \boldsymbol{D}_{ai}\boldsymbol{\Delta}_{ai}(t)\boldsymbol{E}_{ai})\boldsymbol{x} + \boldsymbol{B}_i\boldsymbol{u}\}, \quad (49)$$

where

$$\boldsymbol{A}_1 = \begin{bmatrix} k_{max} & 1 \\ -2 & -6 \end{bmatrix},\ \boldsymbol{A}_2 = \begin{bmatrix} k_{max} & 1 \\ -2 & -6 \end{bmatrix},$$
$$\boldsymbol{A}_3 = \begin{bmatrix} k_{max} & 1 \\ \frac{-2\sin(d_1)}{d_1} & -6 \end{bmatrix},\ \boldsymbol{A}_4 = \begin{bmatrix} k_{max} & 1 \\ \frac{-2\sin(d_1)}{d_1} & -6 \end{bmatrix},$$
$$\boldsymbol{A}_5 = \begin{bmatrix} k_{min} & 1 \\ -2 & -6 \end{bmatrix},\ \boldsymbol{A}_6 = \begin{bmatrix} k_{min} & 1 \\ -2 & -6 \end{bmatrix},$$
$$\boldsymbol{A}_7 = \begin{bmatrix} k_{min} & 1 \\ \frac{-2\sin(d_1)}{d_1} & -6 \end{bmatrix},\ \boldsymbol{A}_8 = \begin{bmatrix} k_{min} & 1 \\ \frac{-2\sin(d_1)}{d_1} & -6 \end{bmatrix},$$

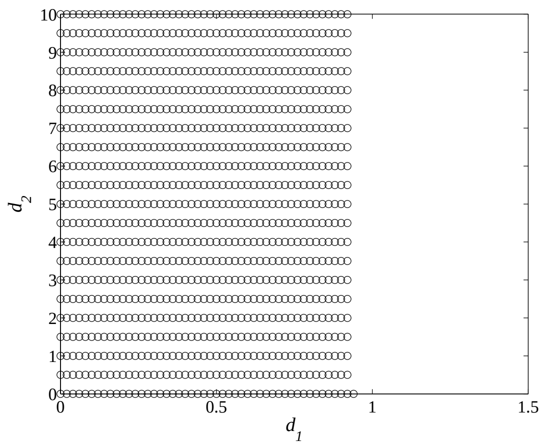

Fig. 4. Feasible area of the LMI-based robust control design conditions proposed in [2] for the T-S fuzzy model (49) with $c = 0.76$.

$$
\begin{aligned}
&\boldsymbol{B}_1 = \boldsymbol{B}_3 = \boldsymbol{B}_5 = \boldsymbol{B}_7 = \begin{bmatrix} d_1 \\ 7 \end{bmatrix}, \\
&\boldsymbol{B}_2 = \boldsymbol{B}_4 = \boldsymbol{B}_6 = \boldsymbol{B}_8 = \begin{bmatrix} -d_1 \\ 7 \end{bmatrix}, \\
&\boldsymbol{D}_{ai} = \begin{bmatrix} c \\ 0 \end{bmatrix}, \ i = 1, \ \cdots, 8, \\
&\boldsymbol{\Delta}_{ai} = \Delta(t)/c, \ i = 1, \ \cdots, 8, \\
&\boldsymbol{E}_{ai} = \begin{bmatrix} 1 & 0 \end{bmatrix}, \ i = 1, \ \cdots, 8, \\
&k_{max} = \max_{|x_1|<d_1, |x_2|<d_2} (-1 + x_1 + x_1^2 + x_1 x_2 - x_2^2), \\
&k_{min} = \min_{|x_1|<d_1, |x_2|<d_2} (-1 + x_1 + x_1^2 + x_1 x_2 - x_2^2).
\end{aligned}
$$

The membership functions are given as follows.

$$
\begin{aligned}
h_1(\boldsymbol{z}) &= \frac{k - k_{min}}{k_{max} - k_{min}} \cdot \frac{\sin x_1 - (\sin d_1/d_1) x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{x_1 + d_1}{2d_1} \\
h_2(\boldsymbol{z}) &= \frac{k - k_{min}}{k_{max} - k_{min}} \cdot \frac{\sin x_1 - (\sin d_1/d_1) x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{d_1 - x_1}{2d_1} \\
h_3(\boldsymbol{z}) &= \frac{k - k_{min}}{k_{max} - k_{min}} \cdot \frac{x_1 - \sin x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{x_1 + d_1}{2d_1} \\
h_4(\boldsymbol{z}) &= \frac{k - k_{min}}{k_{max} - k_{min}} \cdot \frac{x_1 - \sin x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{d_1 - x_1}{2d_1} \\
h_5(\boldsymbol{z}) &= \frac{k_{max} - k}{k_{max} - k_{min}} \cdot \frac{\sin x_1 - (\sin d_1/d_1) x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{x_1 + d_1}{2d_1} \\
h_6(\boldsymbol{z}) &= \frac{k_{max} - k}{k_{max} - k_{min}} \cdot \frac{\sin x_1 - (\sin d_1/d_1) x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{d_1 - x_1}{2d_1} \\
h_7(\boldsymbol{z}) &= \frac{k_{max} - k}{k_{max} - k_{min}} \cdot \frac{x_1 - \sin x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{x_1 + d_1}{2d_1} \\
h_8(\boldsymbol{z}) &= \frac{k_{max} - k}{k_{max} - k_{min}} \cdot \frac{x_1 - \sin x_1}{(1 - (\sin d_1/d_1)) x_1} \cdot \frac{d_1 - x_1}{2d_1}
\end{aligned}
$$

Fig. 4 shows the feasible area of the LMI-based robust control design conditions proposed in [2] for the T-S fuzzy model (49) with $c = 0.76$.

Remark 6. *For nonlinear systems with polynomial terms, it is impossible to exactly construct a global T-S fuzzy model. In this example, a local T-S fuzzy model with 8 rules can be constructed by assuming the ranges of x_1 and x_2, e.g., $|x_1| < d_1$ and $|x_2| < d_2$, where d_1 and d_2 are nonnegative values. If we select huge values for d_1 and d_2, the local T-S fuzzy model could be a global model, however, it becomes much harder to guarantee the stability for larger values of d_1 and d_2. In other words, smaller values of d_1 and d_2 becomes easier to guarantee the stability of the local T-S fuzzy model. However, the LMI robust conditions for T-S fuzzy models are infeasible even for very small values, e.g., $d_1 > 0.96$ when $c = 0.76$.*

B. Design example II

Consider the following nonlinear system with uncertainties.

$$
\begin{aligned}
\dot{x}_1 &= (-1 + \Delta_a(t) + x_1 + x_1^2 + x_1 x_2 - x_2^2) x_1 \\
&\quad + x_2 + x_1 u, \\
\dot{x}_2 &= -2\sin(x_1) x_1 - 6x_2 - 4\sin(x_1)(1 + \Delta_b(t)) u,
\end{aligned} \tag{50}
$$

where $\Delta_a(t)$ and $\Delta_b(t)$ are the uncertainties satisfying $|\Delta_a(t)| \leq c_a$ and $|\Delta_b(t)| \leq c_b$ for all t. All the simulation results given in this design example are carried out for $\Delta_a(t) = c_a \sin(200\pi t)$ and $\Delta_b(t) = c_b \sin(200\pi t)$. However, it should be noted that $\Delta_a(t)$ and $\Delta_b(t)$ are the uncertain terms and only their upper bounds, i.e. c_a and c_b, are known as well as the standard robust control setting.

Using the sector nonlinearity technique [2], the nonlinear system with the uncertainties is exactly converted into the polynomial fuzzy system (9) with r=2, $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x} = [x_1, \ x_2]$, $\boldsymbol{z} = x_1$ and

$$
\begin{aligned}
&\boldsymbol{A}_1(\boldsymbol{x}) = \begin{bmatrix} -1 + x_1 + x_1^2 + x_1 x_2 - x_2^2 & 1 \\ -2 & -6 \end{bmatrix}, \\
&\boldsymbol{A}_2(\boldsymbol{x}) = \begin{bmatrix} -1 + x_1 + x_1^2 + x_1 x_2 - x_2^2 & 1 \\ 2 & -6 \end{bmatrix}, \\
&\boldsymbol{B}_1(\boldsymbol{x}) = \begin{bmatrix} x_1 \\ -4 \end{bmatrix}, \ \boldsymbol{B}_2(\boldsymbol{x}) = \begin{bmatrix} x_1 \\ 4 \end{bmatrix}, \\
&\boldsymbol{D}_{a1}(\boldsymbol{x}) = \boldsymbol{D}_{a2}(\boldsymbol{x}) = \begin{bmatrix} c_a \\ 0 \end{bmatrix}, \\
&\boldsymbol{\Delta}_{a1}(\boldsymbol{x}) = \boldsymbol{\Delta}_{a2}(\boldsymbol{x}) = \Delta_a(t)/c_a, \\
&\boldsymbol{E}_{a1}(\boldsymbol{x}) = \boldsymbol{E}_{a2}(\boldsymbol{x}) = \begin{bmatrix} 1 & 0 \end{bmatrix} \\
&\boldsymbol{D}_{b1}(\boldsymbol{x}) = \begin{bmatrix} 0 \\ -4c_b \end{bmatrix}, \ \boldsymbol{D}_{b2}(\boldsymbol{x}) = \begin{bmatrix} 0 \\ 4c_b \end{bmatrix}, \\
&\boldsymbol{\Delta}_{b1}(\boldsymbol{x}) = \boldsymbol{\Delta}_{b2}(\boldsymbol{x}) = \Delta_b(t)/c_b, \\
&\boldsymbol{E}_{b1}(\boldsymbol{x}) = \boldsymbol{E}_{b2}(\boldsymbol{x}) = 1, \\
&h_1(\boldsymbol{z}) = \frac{\sin(x_1) + 1}{2}, \ h_2(\boldsymbol{z}) = \frac{1 - \sin(x_1)}{2}.
\end{aligned}
$$

Since $||\boldsymbol{\Delta}_{a1}(\boldsymbol{x})|| = ||\boldsymbol{\Delta}_{a2}(\boldsymbol{x})|| = ||\Delta_a(t)/c_a|| \leq 1$ and $||\boldsymbol{\Delta}_{b1}(\boldsymbol{x})|| = ||\boldsymbol{\Delta}_{b2}(\boldsymbol{x})|| = ||\Delta_b(t)/c_b|| \leq 1$, we have $\beta_{a1} = \beta_{a2} = \beta_{b1} = \beta_{b2} = 1$. Moreover, the algorithm presented in Section IV is carried out with the initial setting $s = 0$, $\epsilon_g = 0.001$, $\epsilon_F = 0.001$, $\epsilon_V = 0.001$, $\epsilon_L = 0.001$, $\lambda^{min} = 0.2$, $\lambda^{max} = 5$, $\Delta\lambda = 0.8$, $p_i^{min} = 0.2$, $p_i^{max} = 1$, $\Delta p_i = 0.2$ for $i = 1, 2$. To show the validity of derived conditions, we compare the feasible areas in the region ($0.01 \leq c_a \leq 0.12$ and $0.01 \leq c_b \leq 0.26$) for the proposed robust control design method and the SOS-based design method of [36] as shown in Fig. 5. It shows that the proposed robust design method provides more relaxed results than the method of [36].

For $c_a = c_b = 0.1$, Fig. 6 shows the behavior of the nonlinear system (50) with $u = 0$. By solving the conditions in Theorem 1, a feasible solution for $c_a = c_b = 0.1$ can be obtained as

$$\begin{aligned}
&\lambda = 6.806,\\
&V(\boldsymbol{x}) = 1.3643x_1^2 + 0.1451x_2^2\\
&F_1(\boldsymbol{x}) = \begin{bmatrix} 1.6015x_1 + 0.3447x_2 + 0.7279 \\ 0.3447x_1 + 0.0207 \end{bmatrix}^T,\\
&F_2(\boldsymbol{x}) = \begin{bmatrix} 1.6672x_1 + 0.3119x_2 + 0.7391 \\ 0.3119x_1 - 0.0983 \end{bmatrix}^T,\\
&\bar{g}_{1,1}(\boldsymbol{x}) = 0.4911x_1^4 + 0.0476x_1^3x_2 + 0.3544x_1^2x_2^2\\
&\qquad + 0.0276x_1^3 - 0.1997x_1^2x_2 - 0.1470x_1x_2^2\\
&\qquad + 0.6121x_1^2 - 0.2999x_1x_2 + 0.2630x_2^2,\\
&\bar{g}_{1,2}(\boldsymbol{x}) = 0.5928x_1^4 + 0.1080x_1^3x_2 + 0.4294x_1^2x_2^2\\
&\qquad + 0.1331x_1^3 - 0.0122x_1^2x_2 - 0.0341x_1x_2^2\\
&\qquad + 0.7168x_1^2 - 0.1211x_1x_2 + 0.4064x_2^2,\\
&\bar{g}_{2,1}(\boldsymbol{x}) = 0.5621x_1^4 + 0.1221x_1^3x_2 + 0.4267x_1^2x_2^2\\
&\qquad + 0.1256x_1^3 + 0.0282x_1^2x_2 - 0.0193x_1x_2^2\\
&\qquad + 0.7251x_1^2 - 0.1052x_1x_2 + 0.3879x_2^2,\\
&\bar{g}_{2,2}(\boldsymbol{x}) = 0.5316x_1^4 + 0.0840x_1^3x_2 + 0.3902x_1^2x_2^2\\
&\qquad + 0.1579x_1^3 + 0.1456x_1^2x_2 + 0.0557x_1x_2^2\\
&\qquad + 0.6790x_1^2 - 0.1412x_1x_2 + 0.3752x_2^2.
\end{aligned}$$

Fig. 7 shows the controlled behavior for six different initial conditions. It can be seen from Fig. 7 that the design fuzzy controller stabilizes the system from all the initial conditions although the system has uncertainties.

Remark 7. *Design Examples I and II show that our approach provides more relaxed results than the existing LMI approach and the existing SOS approach. In addition, as mentioned in Remark 6, the LMI-based approach for the T-S fuzzy model does not guarantte the global stability of the nonlinear system.*

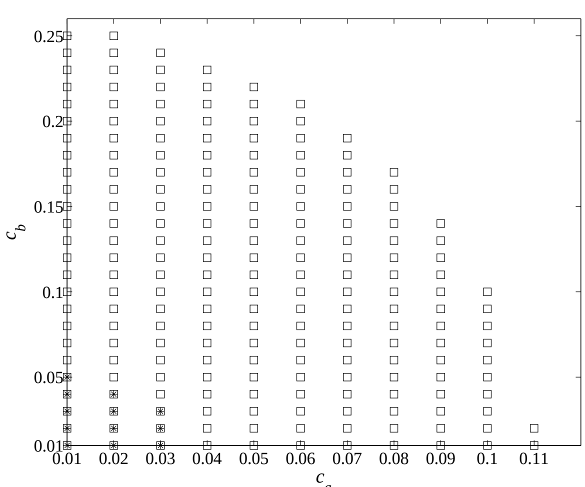

Fig. 5. Feasible areas for proposed robust control design method (□) and the SOS-based design method of [36] (*).

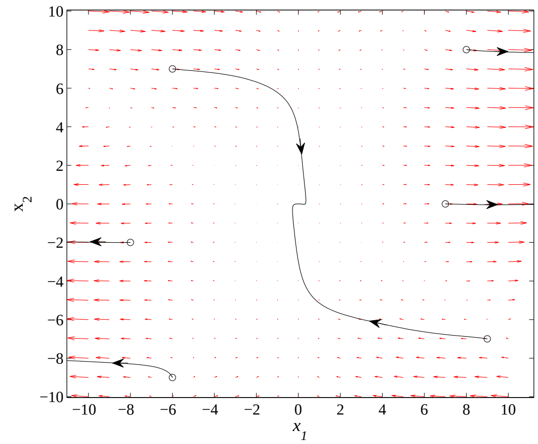

Fig. 6. Behavior of the nonlinear system (50) with $u = 0$.

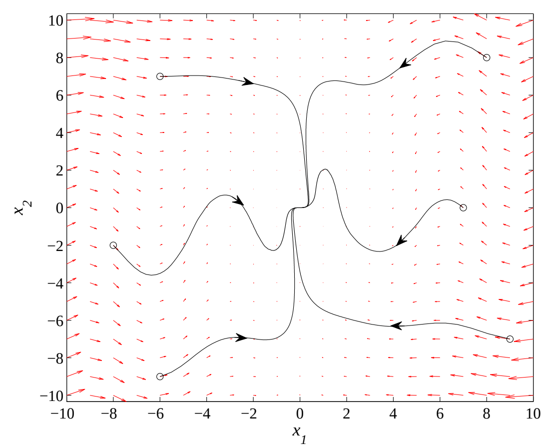

Fig. 7. Controlled behavior of the nonlinear system (50).

VI. Semi-Global Robust Stabilization Conditions with Considering Input Constraints

Sections III and V gave global robust stabilization conditions and their design examples. It is known that the global stabilization is sometimes difficult to be achieved for complicated systems, e.g., unmanned aerial vehicles (UAVs), in practical. Moreover, it is usually the case that the input constraints exist in practical systems. Therefore, Section VI proposes a semi-global robust control design method with considering the input constraints. Section VII will show altitude control of a paraglider-type UAV as a design example of the semi-global robust stabilization with considering the input constraints.

Consider the operation domain

$$D_o = \{\boldsymbol{x} : x_\beta^{min} \le x_\beta \le x_\beta^{max},\ \beta = 1, \cdots, n\} \tag{51}$$

and input constraints

$$u_\ell^{min} \le u_\ell \le u_\ell^{max},\ \ell = 1, \cdots, m. \tag{52}$$

For the operation domain (51), the semi-global robust control satisfying the input constrains (52) can be designed by the following theorem.

Theorem 3. *The polynomial controller (10) satisfying the input constraints (52) stabilizes the system (9) and the outmost Lyapunov function level set $\Omega_{V,\gamma} = \{\boldsymbol{x} : V(\boldsymbol{x}) \leq \gamma\}$ contained in the operation domain (51) is a contractively invariant set if there exist a polynomial function $V(\boldsymbol{x})$, polynomial matrices $\boldsymbol{F}_j(\boldsymbol{x})$, polynomials $\bar{g}_{ij}(\boldsymbol{x})$, $Q_\beta(\boldsymbol{x})$, $\tau_\beta(\boldsymbol{x})$, $\varphi_\beta(\boldsymbol{x})$ and a scalar $\alpha < 0$ such that (17) and the following conditions hold:*

$$V(\boldsymbol{x}) - \epsilon(\boldsymbol{x}) \quad \textit{is SOS}, \tag{53}$$

$$(\sum_{k=1}^{r} \hat{h}_k^2)^s \sum_{i=1}^{r}\sum_{j=1}^{r} \hat{h}_i^2\hat{h}_j^2\{-\Upsilon_{ij}(\boldsymbol{x}) + \alpha V(\boldsymbol{x})\} \quad \textit{is SOS}, \tag{54}$$

$$Q_\beta(\boldsymbol{x}) \textit{ is SOS}, \quad \beta = 1, \cdots, n, \tag{55}$$

$$\begin{aligned}\sum_{\beta=1}^{n} \varphi_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}) - \boldsymbol{d}_\ell \boldsymbol{F}_i(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ - u_\ell^{min} \quad \textit{is SOS}, \quad i = 1, \cdots, r, \ \ell = 1, \cdots, m,\end{aligned} \tag{56}$$

$$\begin{aligned}\sum_{\beta=1}^{n} \tau_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}) + \boldsymbol{d}_\ell \boldsymbol{F}_i(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ + u_\ell^{max} \quad \textit{is SOS}, \quad i = 1, \cdots, r, \ \ell = 1, \cdots, m,\end{aligned} \tag{57}$$

$$\varphi_\beta(\boldsymbol{x}) \textit{ is SOS}, \quad \beta = 1, \cdots, n, \tag{58}$$

$$\tau_\beta(\boldsymbol{x}) \textit{ is SOS}, \quad \beta = 1, \cdots, n, \tag{59}$$

where $\epsilon(\boldsymbol{x})$ is a radially unbounded positive definite polynomial. $\boldsymbol{d}_\ell = [d_1^\ell \ d_2^\ell \ \cdots \ d_m^\ell]$ with $d_\ell^\ell = 1$ and $d_j^\ell = 0 \ \forall j \neq \ell$. s is a nonnegative integer, and

$$\begin{aligned}\Upsilon_{ij}(\boldsymbol{x}) =& \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}(\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) + \bar{g}_{ij}(\boldsymbol{x}) \\ &- \sum_{\beta=1}^{n} Q_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}).\end{aligned}$$

If a solution satisfying the conditions (17), (53)∼(59) is found, the outmost Lyapunov function level set $\Omega_{V,\gamma} = \{\boldsymbol{x} : V(\boldsymbol{x}) \leq \gamma\}$ contained in the operation domain (51), i.e. the contractively invariant set, can be obtained by solving the following optimization problem.

$$\max_{\phi_\beta(\boldsymbol{x})} \ \gamma \quad \textit{subject to}$$

$$\begin{aligned}\phi_\beta(\boldsymbol{x})(V(\boldsymbol{x}) - \gamma) - (x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}) \\ \textit{is SOS}, \quad \beta = 1, \ \cdots, \ n,\end{aligned} \tag{60}$$

$$\phi_\beta(\boldsymbol{x}) \textit{ is SOS}, \quad \beta = 1, \ \cdots, \ n. \tag{61}$$

Proof: In the proof, we need to show that

1) If the conditions (17), (53)∼(55) hold, then the outmost Lyapunov function level set $\Omega_{V,\gamma} = \{\boldsymbol{x} : V(\boldsymbol{x}) \leq \gamma\}$ contained in the operation domain (51) is a contractively invariant set;

2) If the conditions (56)∼(59) hold, then the input constraints (52) are satisfied in the operation domain (51).

1) For the operation domain (51), the following condition holds:

$$\sum_{\beta=1}^{n} Q_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}) \leq 0 \tag{62}$$

where $Q_\beta(\boldsymbol{x}) \geq 0$ that is guaranteed by (55). From (21) and (22), if (17) holds, then

$$\begin{aligned}\dot{V}(\boldsymbol{x}) \leq \sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \{ &\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}(\boldsymbol{A}_i(\boldsymbol{x}) \\ &- \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) + \bar{g}_{ij}(\boldsymbol{x})\}.\end{aligned} \tag{63}$$

It can be obtained from (62) and (63) that $\dot{V}(\boldsymbol{x}) \leq \alpha V(\boldsymbol{x}) < 0$ for $D_o - \{0\}$ if there exist $\alpha < 0$ such that

$$\begin{aligned}\sum_{i=1}^{r}\sum_{j=1}^{r} h_i h_j \{ &\frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}}(\boldsymbol{A}_i(\boldsymbol{x}) - \boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x}) \\ &+ \bar{g}_{ij}(\boldsymbol{x}) - \alpha V(\boldsymbol{x}) \\ &- \sum_{\beta=1}^{n} Q_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max})\} \leq 0.\end{aligned} \tag{64}$$

By applying the copositivity presented in Corollary 1, the condition (64) holds if (54) is satisfied. Furthermore, if (53) holds, then $V(\boldsymbol{x})$ is a positive definite and radially unbounded function which means that the level set $\Omega_{V,\gamma}$ is bounded for any value of $\gamma > 0$. Consequently, if the conditions (17), (53)∼(55) hold, then the outmost Lyapunov function level set $\Omega_{V,\gamma}$ contained in the operation domain (51) is a contractively invariant set. Moreover, by applying polynomial S-procedure, $\Omega_v \subseteq D_o$ is carried out by (60) and (61).

2) For the operation domain (51), the following two conditions hold:

$$-\sum_{\beta=1}^{n} \varphi_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}) \geq 0, \tag{65}$$

$$-\sum_{\beta=1}^{n} \tau_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}) \geq 0, \tag{66}$$

where $\varphi_\beta(\boldsymbol{x}) \geq 0$ and $\tau_\beta(\boldsymbol{x}) \geq 0$ that are guaranteed by (58) and (59) respectively. By applying the vector $\boldsymbol{d}_\ell$, the l-th input can be represented as

$$u_\ell = \boldsymbol{d}_\ell \boldsymbol{u} = -\sum_{i=1}^{r} h_i \boldsymbol{d}_\ell \boldsymbol{F}_i(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}). \tag{67}$$

It can be obtained from (65) and (67) that $u_\ell - u_\ell^{min} \geq 0$ for the operation domain (51) if the following condition holds:

$$\begin{aligned}-\sum_{i=1}^{r} h_i \boldsymbol{d}_\ell \boldsymbol{F}_i(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) - u_\ell^{min} \\ \geq -\sum_{\beta=1}^{n} \varphi_\beta(\boldsymbol{x})(x_\beta - x_\beta^{min})(x_\beta - x_\beta^{max}).\end{aligned} \tag{68}$$

It is obviously that (68) holds if (56) is satisfied. On the other hand, it can be obtained from (66) and (67) that $u_\ell^{max} - u_\ell \geq 0$

for the operation domain (51) if the following condition holds:

$$u_\ell^{max}+\sum_{i=1}^{r}h_i\boldsymbol{d}_\ell\boldsymbol{F_i}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) \geq -\sum_{\beta=1}^{n}\tau_\beta(\boldsymbol{x})(x_\beta-x_\beta^{min})(x_\beta-x_\beta^{max}). \quad (69)$$

It is obviously that (69) holds if (57) is satisfied. ■

Theorem 4. *Assume that $\Delta_{bi}(\boldsymbol{x})=0\ \forall i$, i.e. there are no uncertainties with respect to input terms. The polynomial controller (10) satisfying the input constraints (52) stabilizes the system (9) and the outmost Lyapunov function level set $\Omega_{V,\gamma}=\{\boldsymbol{x}:V(\boldsymbol{x})\leq\gamma\}$ contained in the operation domain (51) is a contractively invariant set if there exist a polynomial function $V(\boldsymbol{x})$, polynomial matrices $\boldsymbol{F}_j(\boldsymbol{x})$, polynomials $\bar{g}_i(\boldsymbol{x})$, $Q_\beta(\boldsymbol{x})$, $\tau_\beta(\boldsymbol{x})$, $\varphi_\beta(\boldsymbol{x})$ and a scalar $\alpha<0$ such that (55)∼(59) and the following conditions hold:*

$$V(\boldsymbol{x})-\epsilon(\boldsymbol{x}) \quad \textit{is SOS}, \quad (70)$$

$$(\sum_{k=1}^{r}\hat{h}_k^2)^s\sum_{i=1}^{r}\sum_{j=1}^{r}\hat{h}_i^2\hat{h}_j^2\{-\bar{\Upsilon}_{ij}(\boldsymbol{x})+\alpha V(\boldsymbol{x})\} \quad \textit{is SOS}, \quad (71)$$

$$\boldsymbol{v}_1^T\begin{bmatrix}\lambda\bar{g}_i(\boldsymbol{x}) & * & * \\ \lambda\boldsymbol{D}_{ai}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial\boldsymbol{x}})^T & 2\boldsymbol{I} & \boldsymbol{0} \\ \beta_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & 2\boldsymbol{I}\end{bmatrix}\boldsymbol{v}_1 \quad \textit{is SOS}, \quad (72)$$

where $\epsilon(\boldsymbol{x})$ is a radially unbounded positive definite polynomial. s is a nonnegative integer, and

$$\bar{\Upsilon}_{ij}(\boldsymbol{x})=\frac{\partial V(\boldsymbol{x})}{\partial\boldsymbol{x}}(\boldsymbol{A}_i(\boldsymbol{x})-\boldsymbol{B}_i(\boldsymbol{x})\boldsymbol{F}_j(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x})+\bar{g}_i(\boldsymbol{x}) -\sum_{\beta=1}^{n}Q_\beta(\boldsymbol{x})(x_\beta-x_\beta^{min})(x_\beta-x_\beta^{max}).$$

Proof: The proof is omitted since it is directly obtained from Theorem 3. In this case, (17) is reduced to (72). ■

Theorem 5. *Assume that $\boldsymbol{B}_i(\boldsymbol{x})=\boldsymbol{B}(\boldsymbol{x})$, $\boldsymbol{D}_{bi}(\boldsymbol{x})=\boldsymbol{D}_b(\boldsymbol{x})$, $\Delta_{bi}(\boldsymbol{x})=\Delta_b(\boldsymbol{x})$ and $\boldsymbol{E}_{bi}(\boldsymbol{x})=\boldsymbol{E}_b(\boldsymbol{x})$ for all i. The polynomial controller (10) satisfying the input constraints (52) stabilizes the system (9) and the outmost Lyapunov function level set $\Omega_{V,\gamma}=\{\boldsymbol{x}:V(\boldsymbol{x})\leq\gamma\}$ contained in the operation domain (51) is a contractively invariant set if there exist a polynomial function $V(\boldsymbol{x})$, polynomial matrices $\boldsymbol{F}_i(\boldsymbol{x})$, polynomials $\bar{g}_i(\boldsymbol{x})$, $Q_\beta(\boldsymbol{x})$, $\tau_\beta(\boldsymbol{x})$, $\varphi_\beta(\boldsymbol{x})$ and a scalar $\alpha<0$ such that (55)∼(59) and the following conditions hold:*

$$V(\boldsymbol{x})-\epsilon(\boldsymbol{x}) \quad \textit{is SOS}, \quad (73)$$

$$\sum_{i=1}^{r}\hat{h}_i^2\{-\bar{\Upsilon}_i(\boldsymbol{x})+\alpha V(\boldsymbol{x})\} \quad \textit{is SOS}, \quad (74)$$

$$\boldsymbol{v}_1^T\boldsymbol{L}_i(\lambda,\boldsymbol{x})\boldsymbol{v}_1 \quad \textit{is SOS}, \quad (75)$$

where $\epsilon(\boldsymbol{x})$ is a radially unbounded positive definite polynomial. s is a nonnegative integer, and

$$\bar{\Upsilon}_i(\boldsymbol{x})=\frac{\partial V(\boldsymbol{x})}{\partial\boldsymbol{x}}(\boldsymbol{A}_i(\boldsymbol{x})-\boldsymbol{B}(\boldsymbol{x})\boldsymbol{F}_i(\boldsymbol{x}))\hat{\boldsymbol{x}}(\boldsymbol{x})+\bar{g}_i(\boldsymbol{x}), -\sum_{\beta=1}^{n}Q_\beta(\boldsymbol{x})(x_\beta-x_\beta^{min})(x_\beta-x_\beta^{max}),$$

$$\boldsymbol{L}_i(\lambda,\boldsymbol{x})= \begin{bmatrix}\lambda\bar{g}_i(\boldsymbol{x}) & * & * & * & * \\ \lambda\boldsymbol{D}_{ai}^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial\boldsymbol{x}})^T & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \lambda\boldsymbol{D}_b^T(\boldsymbol{x})(\frac{\partial V(\boldsymbol{x})}{\partial\boldsymbol{x}})^T & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \beta_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0} \\ \beta_b(\boldsymbol{x})\boldsymbol{E}_b(\boldsymbol{x})\boldsymbol{F}_i(\boldsymbol{x})\hat{\boldsymbol{x}}(\boldsymbol{x}) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & 2\boldsymbol{I}\end{bmatrix}.$$

Proof: The proof is omitted since it is directly obtained by the same fashion as in the proof of Theorem 3. ■

VII. Design Example of Semi-global Robust Stabilization with Considering Input Constraints

Section VII gives a design example of the semi-global robust stabilization for a paraglider-type UAV. Under some assumptions, the altitude error dynamics of a paraglider-type UAV around the trimmed equilibrium are given as

$$\begin{aligned}\ddot{e}(t) &= 0.1336(1+\Delta_\theta(t))u(t) \\ &+ p_1(\dot{e}(t))(1+\Delta_\alpha(t))\cos(0.2\dot{e}(t))\dot{e}(t) \\ &+ p_2(\dot{e}(t))(1+\Delta_\alpha(t))\sin(0.2\dot{e}(t))\dot{e}(t), \quad (76)\end{aligned}$$

where $e(t)$ denotes the altitude error between the altitude of the UAV and a constant desired altitude, and $u(t)$ is the throttle input difference from the trimmed equilibrium throttle input. $p_1(\dot{e}(t))$ and $p_2(\dot{e}(t))$ are polynomial elements including the aerodynamics generated by the canopy of the UAV and are described as

$$\begin{aligned}p_1(\dot{e}(t)) &= 6.270\cdot10^{-4}\cdot\dot{e}^2(t)+7.271\cdot10^{-2}\dot{e}(t), \\ p_2(\dot{e}(t)) &= 1.188\cdot10^{-4}\cdot\dot{e}^2(t)-7.358\cdot10^{-3}\dot{e}(t).\end{aligned}$$

We consider two kinds of uncertainties. The first uncertainty is aerodynamics uncertainty, i.e., $\Delta_\alpha(t)$, since it is very difficult to exactly obtain the real aerodynamics of the canopy. The second uncertainty is input uncertainty, i.e., $\Delta_\theta(t)$, since the thrust force generated by a motor is influenced by battery condition, wind conditions, and so on.

Using the sector nonlinearity technique [2], the nonlinear system (76) with the uncertainties is exactly converted into the following four-rule polynomial fuzzy system with uncertainties:

$$\dot{\boldsymbol{x}}=\sum_{i=1}^{r}h_i(\boldsymbol{z})\{(\boldsymbol{A}_i(\boldsymbol{x})+\boldsymbol{D}_{ai}(\boldsymbol{x})\boldsymbol{\Delta}_{ai}(\boldsymbol{x})\boldsymbol{E}_{ai}(\boldsymbol{x}))\boldsymbol{x} +(\boldsymbol{B}(\boldsymbol{x})+\boldsymbol{D}_b(\boldsymbol{x})\boldsymbol{\Delta}_b(\boldsymbol{x})\boldsymbol{E}_b(\boldsymbol{x}))\boldsymbol{u}\},$$

where r=4, $\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{x} = [\dot{e},\ e]$, $\boldsymbol{z} = \dot{e}$ and

$$\boldsymbol{A}_1(\boldsymbol{x}) = \begin{bmatrix} p_1(\dot{e}(t)) + p_2(\dot{e}(t)) & 0 \\ 1 & 0 \end{bmatrix},$$
$$\boldsymbol{A}_2(\boldsymbol{x}) = \begin{bmatrix} p_1(\dot{e}(t)) - p_2(\dot{e}(t)) & 0 \\ 1 & 0 \end{bmatrix},$$
$$\boldsymbol{A}_3(\boldsymbol{x}) = \begin{bmatrix} -p_1(\dot{e}(t)) + p_2(\dot{e}(t)) & 0 \\ 1 & 0 \end{bmatrix},$$
$$\boldsymbol{A}_4(\boldsymbol{x}) = \begin{bmatrix} -p_1(\dot{e}(t)) - p_2(\dot{e}(t)) & 0 \\ 1 & 0 \end{bmatrix},$$
$$\boldsymbol{B}(\boldsymbol{x}) = \begin{bmatrix} 0.1336 \\ 0 \end{bmatrix},$$
$$\boldsymbol{D}_{a1}(\boldsymbol{x}) = \begin{bmatrix} c_2 p_1(\dot{e}(t)) & c_2 p_2(\dot{e}(t)) \\ 0 & 0 \end{bmatrix},$$
$$\boldsymbol{D}_{a2}(\boldsymbol{x}) = \begin{bmatrix} c_2 p_1(\dot{e}(t)) & -c_2 p_2(\dot{e}(t)) \\ 0 & 0 \end{bmatrix},$$
$$\boldsymbol{D}_{a3}(\boldsymbol{x}) = \begin{bmatrix} -c_2 p_1(\dot{e}(t)) & c_2 p_2(\dot{e}(t)) \\ 0 & 0 \end{bmatrix},$$
$$\boldsymbol{D}_{a4}(\boldsymbol{x}) = \begin{bmatrix} -c_2 p_1(\dot{e}(t)) & -c_2 p_2(\dot{e}(t)) \\ 0 & 0 \end{bmatrix},$$
$$\boldsymbol{\Delta}_{a1}(\boldsymbol{x}) = \boldsymbol{\Delta}_{a2}(\boldsymbol{x}) = \boldsymbol{\Delta}_{a3}(\boldsymbol{x}) = \boldsymbol{\Delta}_{a4}(\boldsymbol{x}) = \frac{1}{c_2}\begin{bmatrix} \Delta_\alpha(t) & 0 \\ 0 & \Delta_\alpha(t) \end{bmatrix},$$
$$\boldsymbol{E}_{a1}(\boldsymbol{x}) = \boldsymbol{E}_{a2}(\boldsymbol{x}) = \boldsymbol{E}_{a3}(\boldsymbol{x}) = \boldsymbol{E}_{a4}(\boldsymbol{x}) = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix},$$
$$\boldsymbol{D}_b(\boldsymbol{x}) = \begin{bmatrix} 0.1336c_1 \\ 0 \end{bmatrix},$$
$$\boldsymbol{\Delta}_b(\boldsymbol{x}) = \Delta_\theta(t)/c_1,$$
$$\boldsymbol{E}_b(\boldsymbol{x}) = 1,$$
$$h_1(\boldsymbol{z}) = \frac{(\cos(0.2\dot{e}) + 1)(\sin(0.2\dot{e}) + 1)}{4},$$
$$h_2(\boldsymbol{z}) = \frac{(\cos(0.2\dot{e}) + 1)(1 - \sin(0.2\dot{e}))}{4},$$
$$h_3(\boldsymbol{z}) = \frac{(1 - \cos(0.2\dot{e}))(\sin(0.2\dot{e}) + 1)}{4},$$
$$h_4(\boldsymbol{z}) = \frac{(1 - \cos(0.2\dot{e}))(1 - \sin(0.2\dot{e}))}{4}.$$

Since $||\boldsymbol{\Delta}_{ai}(\boldsymbol{x})|| \leq 1$ for $i = 1, \cdots, 4$ and $||\boldsymbol{\Delta}_b(\boldsymbol{x})|| \leq 1$ as well as in the previous two examples, we have $\beta_{ai} = 1$ for $i = 1, \cdots, 4$ and $\beta_b = 1$. For any situation satisfying $c_1 > 0$ and $c_2 > 0$, no solutions can be obtained by applying the globally robust control design proposed in [36]. Also, no solutions can be found by applying the globally robust control design of the proposed Theorem 1.

Assume that the operation domain for the UAV system is $D_o = \{\dot{e} : -1.5 \leq \dot{e} \leq 1.5$ and $e : -10 \leq e \leq 10\}$, the input constraint is $-5 \leq u \leq 5$, and $c_1 = c_2 = 0.1$. Moreover, the algorithm for solving Theorem 5 is carried out with the initial setting $\epsilon_g = 0.005$, $\epsilon_F = 0.005$, $\epsilon_V = 0.005$, $\epsilon_L = 0.005$, $\lambda^{min} = 0.5$, $\lambda^{max} = 2$, $\Delta\lambda = 0.5$, $p_i^{min} = 0.2$, $p_i^{max} = 1$, $\Delta p_i = 0.4$ for $i = 1, 2$. By solving the conditions in Theorem 5, a feasible solution is found as

$$\alpha = -0.0027,\ \lambda = 5.8979,$$
$$V(\boldsymbol{x}) = 2.8789x_1^2 + 0.1894x_1x_2 + 0.0683x_2^2$$
$$\boldsymbol{F}_1(\boldsymbol{x}) = [F_1^{11}(\boldsymbol{x})\ F_1^{12}(\boldsymbol{x})],$$
$$F_1^{11}(\boldsymbol{x}) = 0.2919x_1^2 - 0.02140x_1x_2 + 0.0016x_2^2 + 0.2850x_1 + 0.0073x_2 + 1.2042,$$
$$F_1^{12}(\boldsymbol{x}) = -0.0214x_1^2 + 0.0016x_1x_2 - 0.0002x_2^2 + 0.0073x_1 + 1.3281 \times 10^{-5}x_2 + 0.18254,$$
$$\boldsymbol{F}_2(\boldsymbol{x}) = [F_2^{11}(\boldsymbol{x})\ F_2^{12}(\boldsymbol{x})],$$
$$F_2^{11}(\boldsymbol{x}) = 0.2677x_1^2 - 0.0231x_1x_2 + 0.0012x_2^2 + 0.3501x_1 + 0.0125x_2 + 1.3162,$$
$$F_2^{12}(\boldsymbol{x}) = -0.0231x_1^2 + 0.0012x_1x_2 - 0.0002x_2^2 + 0.0125x_1 + 6.3453 \times 10^{-5}x_2 + 0.1877,$$
$$\boldsymbol{F}_3(\boldsymbol{x}) = [F_3^{11}(\boldsymbol{x})\ F_3^{12}(\boldsymbol{x})],$$
$$F_3^{11}(\boldsymbol{x}) = 0.266x_1^2 - 0.02187x_1x_2 + 0.0012x_2^2 - 0.3234x_1 - 0.0116x_2 + 1.3234,$$
$$F_3^{12}(\boldsymbol{x}) = -0.0219x_1^2 + 0.0012x_1x_2 - 0.0002x_2^2 - 0.0116x_1 - 0.0002x_2 + 0.1868,$$
$$\boldsymbol{F}_4(\boldsymbol{x}) = [F_4^{11}(\boldsymbol{x})\ F_4^{12}(\boldsymbol{x})],$$
$$F_4^{11}(\boldsymbol{x}) = 0.2813x_1^2 - 0.0212x_1x_2 + 0.0016x_2^2 - 0.2650x_1 - 0.0078x_2 + 1.2181,$$
$$F_4^{12}(\boldsymbol{x}) = -0.0212x_1^2 + 0.0016x_1x_2 - 0.0002x_2^2 - 0.0078x_1 - 0.0002x_2 + 0.18,$$

etc. Moreover, by solving the optimization problem of (60) and (61), the outmost Lyapunov function level set contained in the operation domain is obtained as $\Omega_{V,6.1822} = \{\boldsymbol{x} : V(\boldsymbol{x}) \leq 6.1822\}$, and the SOS multipliers are obtained as $\phi_1(\boldsymbol{x}) = 0.3640$ and $\phi_2(\boldsymbol{x}) = 15.676$. Fig. 8 shows the outmost Lyapunov function level set (the contractively invariant set) and the controlled results for six cases of initial states. From Fig. 8, the control system is asymptotically stable and the $\Omega_{V,6.1822} = \{\boldsymbol{x} : V(\boldsymbol{x}) \leq 6.1822\}$ is a contractively invariant set. Fig. 9 shows the control inputs for the six cases. It can be seen from Fig. 9 that all control inputs satisfy the input constraint $-5 \leq u(t) \leq 5$. Fig. 10 shows the time response for Case 1. Fig. 11 shows the control input for Case 1.

VIII. Conclusions

This paper has presented a new SOS design framework for robust control of polynomial fuzzy systems with uncertainties. Two kinds of robust stabilization conditions, i.e., global SOS robust stabilization conditions and semi-global SOS robust stabilization conditions, are derived in terms of SOS. The new design framework has given key ideas to avoid conservative issues. The first key idea is that we directly solve non-convex SOS design conditions without applying the typical transformation. The second key idea is that we bring a so-called copositivity concept. These ideas provide some advantages in addition to relaxations. To solve our SOS robust stabilization conditions efficiently, we have introduced a gradient algorithm formulated as a minimizing optimization problem of the upper bound of the time derivative of polynomial Lyapunov functions. Three design examples have been provided to illustrate the validity and applicability of the proposed design framework. The examples have demonstrated advantages of the new SOS design framework for the existing LMI approaches and the existing convex SOS approach. Our next subjects are to apply the advanced SOS robust stabilization conditions to more complex systems, e.g., [44], [45], [46].

Acknowledgment

The authors would like to thank Mr. Takahiro Endo, The University of Electro-Communications, Tokyo, Japan, for his support of this research.

References

[1] K. Tanaka and M. Sugeno, “Stability Analysis and Design of Fuzzy Control Systems,” FUZZY SETS AND SYSTEMS 45, no. 2, pp. 135-156, Jan. 1992.

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

[3] 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.

[4] H. O. Wang, K. Tanaka and M. F. Griffin, “An Analytical Framework of Fuzzy Modeling and Control of Nonlinear Systems,” 1995 American Control Conference, Seattle, June 1995, pp.2272 - 2276.

[5] H. O. Wang, K. Tanaka and M. F. Griffin, “An Approach to Fuzzy Control of Nonlinear Systems”, IEEE Transactions on Fuzzy Systems, Vol.4, No.1, pp.14-23, Feb. 1996.

[6] K. Tanaka, T. Ikeda and H. O. Wang, “Robust Stabilization of a Class of Uncertain Nonlinear Systems via Fuzzy Control”, IEEE Transactions on Fuzzy Systems, Vol.4, No.1, pp.1-13, Feb. 1996.

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

[8] Y. Nesterov and A. Nemirovskii, Interior-Point Polynomial Algorithms in Convex Programming, Society for Industrial and Applied Mathematics, Philadelphia, 1994.

[9] 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.

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

[11] 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 International Conference on Decision and Control, New Orleans, Dec. 2007, pp. 5954-5959.

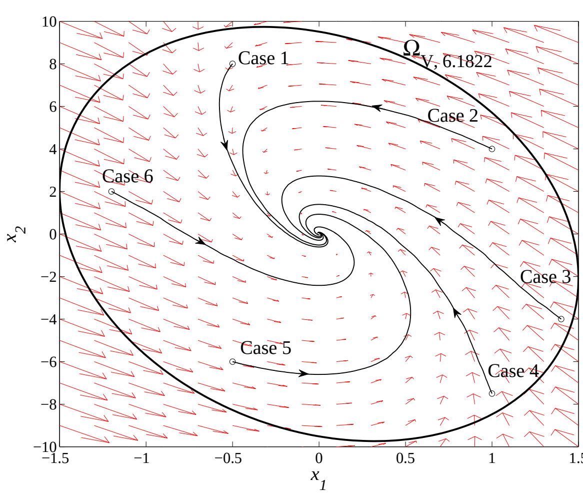

Fig. 8. The controlled results for six cases of initial states.

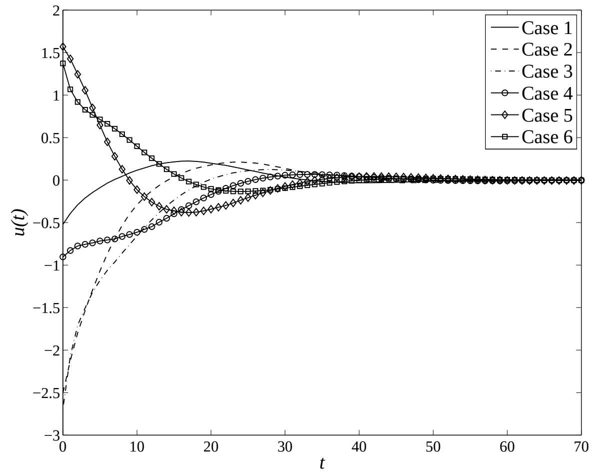

Fig. 9. The control inputs for the six cases.

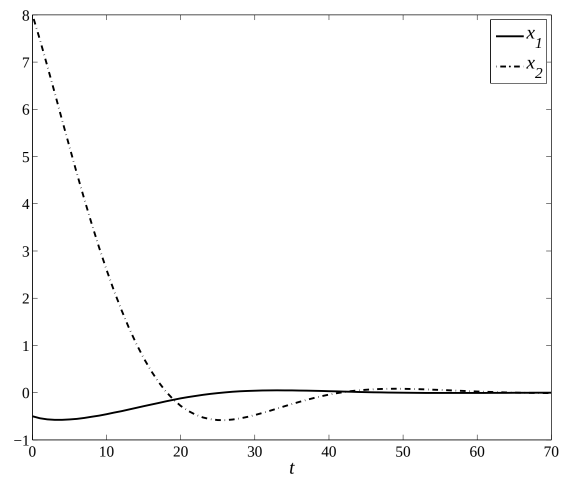

Fig. 10. Time response for Case 1.

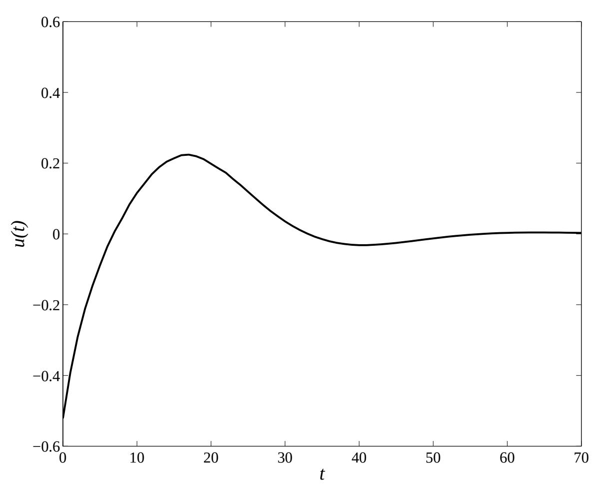

Fig. 11. The control input for Case 1.

[12] K. Tanaka, H. Yoshida, H. Ohtake and H. O. Wang, "A Sum of Squares Approach to Modeling and Control of Nonlinear Dynamical Systems with Polynomial Fuzzy Systems", IEEE Transactions on Fuzzy Systems, vol.17, no.4, pp.911-922, August 2009.

[13] K. Tanaka, H. Ohtake and H. O. Wang, "Guaranteed Cost Control of Polynomial Fuzzy Systems via a Sum of Squares Approach," IEEE Transactions on Systems, Man and Cybernetics Part B, Vol.39, No.2, pp.561-567, April 2009.

[14] K. Tanaka, H. Ohtake, T. Seo, M. Tanaka and H. O. Wang, "Polynomial Fuzzy Observer Designs:A Sum of Squares Approach," IEEE Transactions on Systems, Man, and Cybernetics, Part B, Vol.42, No.5, pp.1330-1342, Oct. 2012.

[15] 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, Hong Kong, June 2008, pp. 347-353.

[16] M. Narimani, and H. K. Lam, "SOS-Based Stability Analysis of Polynomial Fuzzy-Model-Based Control Systems Via Polynomial Membership Functions," IEEE Transactions on Fuzzy Systems, vol.18, no.5, pp.862-871, Oct. 2010.

[17] H. K. Lam, "Polynomial fuzzy-model-based control systems: stability analysis via piecewise-linear membership functions," IEEE Trans. Fuzzy Syst., vol. 19, no. 3, pp. 588-593, Jun. 2011.

[18] J. -C. Lo, Y. -T. Lin, W. -S. Chang and F. -Y. Lin, "SOS-based Fuzzy Stability Analysis via Homogeneous Lyapunov Functions," 2014 IEEE International Conference on Fuzzy Systems, Beijing, July 2014, pp.2300- 2305.

[19] G. -R. Yu and H. -T. Huang "A Sum-of-Squares Approach to Synchronization of Chaotic Systems with Polynomial Fuzzy Systems," Proceedings of 2012 International Conference on Fuzzy Theory and Its Applications, Taichung, Nov. 2012, pp.175-180.

[20] H. K. Lam, M. Narimani, H. Li, and H. Liu, "Stability analysis of polynomial-fuzzy-model-based control systems using switching polynomial Lyapunov function," IEEE Trans. Fuzzy Syst., vol. 21, no. 5, pp. 800-813, Oct. 2013.

[21] A. Sala and C. Arino, "Polynomial fuzzy models for nonlinear control: A Taylor series approach," IEEE Trans. Fuzzy Syst., vol. 17, no. 6, pp. 1284-1295, Dec. 2009.

[22] A. Schwung, T. Guβner, and J. Adamy, "Stability Analysis of Recurrent Fuzzy Systems: A Hybrid System and SOS Approach," IEEE Trans. Fuzzy Syst., vol. 19, no. 3, pp.423-431, June 2011.

[23] K. Tanaka, T. Ikeda, H. O. Wang, "Fuzzy Regulators and Fuzzy Observers: Relaxed Stability Conditions and LMI-Based Designs," IEEE Transactions on fuzzy systems, vol. 6, no. 2, pp.250-265, May 1998.

[24] S. Hong and R. Langari, "Synthesis of an LMI-based Fuzzy Control System with guaranteed Optimal H^{∞} Performance," *Proc. of FUZZ-IEEE'98*, Anchorage, AK, May 1998, pp. 422-427.

[25] M. Sugeno, "On Stability of Fuzzy Systems Expressed by Fuzzy Rules with Singleton Consequents," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 2, pp. 201-224 April, 1999.

[26] R. -J. Wang, W. -W. Lin and W. -J. Wang, "Stabilizability of linear quadratic state feedback for uncertain fuzzy time-delay systems," IEEE Transactions on Systems, Man and Cybernetics, Part B, vol. 34, no.2, pp.1288-1292, April 2004.

[27] W. -J. Wang and C, -H. Sun, "A relaxed stability criterion for T-S fuzzy discrete systems," IEEE Transactions on Systems, Man and Cybernetics, Part B, vol. 34, no.5, pp.2155-2158, Oct. 2004.

[28] S. -S. Chen, Y. -C. Chang, S. -F. Su, S. -L. Chung and T. -T. Lee, "Robust static output-feedback stabilization for nonlinear discrete-time systems with time delay via fuzzy control approach," IEEE Transactions on Fuzzy Systems, vol. 13, no.2, pp.263-272, April 2005.

[29] C. -C. Hsiao, S. -F. Su, T. -T. Lee and C. -C. Chuang, "Hybrid compensation control for affine TSK fuzzy control systems," IEEE Transactions on Systems, Man and Cybernetics, Part B, vol. 34, no.4, pp.1865-1873 August 2004.

[30] P. Baranyi, et al., "SVD-based complexity reduction to TS fuzzy models," IEEE Transaction on Industrial Electronics, vol. 49, no. 2, pp.433-443, April 2002.

[31] 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.

[32] P. Baranyi, et al., ,"Numerical Reconstruction of the HOSVD-based Canonical Form of Polytopic Dynamic Models," Proc. of 10th Int. Conf. on Intelligent Engineering Systems (INES 2006), London, United Kingdom, June 2006, pp.196-201.

[33] K. Tanaka, H. Ohtake and H. O. Wang: "A Descriptor System Approach to Fuzzy Control System Design via Fuzzy Lyapunov Functions," IEEE Transactions on Fuzzy Systems, vol.15, no. 3, pp.333 - 341, June 2007.

[34] H.-N Wu and H.-X. Li: "New Approach to Delay-Dependent Stability Analysis and Stabilization for Continuous-Time Fuzzy Systems With Time-Varying Delay," IEEE Transactions on Fuzzy Systems, vol. 15, no. 3, pp.482 - 493, June 2007.

[35] J. -C. Lo and M. -L. Lin: "Existence of Similarity Transformation Converting BMIs to LMIs," IEEE Transactions on Fuzzy Systems, vol. 15, no.5, pp.840 - 851, Oct 2007.

[36] K. Cao, X. Z. Gao, T. Vasilakos, W. Pedrycz, "Analysis of stability and robust stability of polynomial fuzzy model-based control systems using a sum-of-squares approach," Journal of soft comput, vol. 18, issue 3, pp.433-442, June 2013.

[37] 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, Jan.-Feb. 1985.

[38] G. Balas, A. Packard, P. Seiler, U. Topcu, "Robustness analysis of nonlinear systems," 2009, Available at: http://www.aem.umn.edu/∼AerospaceControl/, accessed July 2013.

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

[40] 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.

[41] K. C. Toh, R. H. Tutuncu and M. J. Todd, "On the implement of SDPT3 (version 3.1) - A MATLAB software package for semidefinite-quadratic-linear programming," 2004 IEEE International Conference on Computer Aided Control System Designs, Taipei, Sept. 2001, pp.290-296.

[42] Y.-J. Chen, H. Ohtake, K. Tanaka, and H. O. Wang, "Relaxed Stabilization Criterion for T-S Fuzzy Systems by Minimum-Type Piecewise-Lyapunov-Function-Based Switching Fuzzy Controller," IEEE Transactions on Fuzzy Systems, Vol.20, No.6, pp.1166-1173, Dec. 2012.

[43] T. Hu, "Nonlinear control design for linear differential inclusions via convex hull of quadratics," Automatica, vol. 43, no. 4, pp. 685-692, 2007.

[44] R. Sakthivel, S. Santra, K. Mathiyalagan, S. Anthoni, "Robust reliable sampled-data control for offshore steel jacket platforms with nonlinear perturbations," Nonlinear Dynamics, vol. 78, issue 2, pp.1109-1123, Oct. 2014.

[45] R. Sakthivel, P. Vadivel, K. Mathiyalagan and A. Arunkumar "Fault-Distribution Dependent Reliable $\mathcal{H}_\infty$ Control for Takagi-Sugeno Fuzzy Systems," Journal of Dynamic Systems, Measurement, and Control, Vol.136, Issue 2, 021021, Jan. 2014.

[46] K. Mathiyalagan, H. Su, P. Shi, R. Sakthivel, "Exponential $\mathcal{H}_\infty$ Filtering for Discrete-Time Switched Neural Networks With Random Delays," IEEE Transactions on Cybenetics, Accepted.

Kazuo Tanaka (S'87 - M'91 - SM'09 - F'14) received the B.S. and M.S. degrees in Electrical Engineering from Hosei University, Tokyo, Japan, in 1985 and 1987, and Ph.D. degree, in Systems Science from Tokyo Institute of Technology, in 1990, respectively.

He is currently a Professor in Department of Mechanical Engineering and Intelligent Systems at The University of Electro-Communications. He was a Visiting Scientist in Computer Science at the University of North Carolina at Chapel Hill in 1992 and 1993. He received the Best Young Researchers Award from the Japan Society for Fuzzy Theory and Systems in 1990, the Outstanding Papers Award at the 1990 Annual NAFIPS Meeting in Toronto, Canada, in 1990, the Outstanding Papers Award at the Joint Hungarian-Japanese Symposium on Fuzzy Systems and Applications in Budapest, Hungary, in 1991, the Best Young Researchers Award from the Japan Society for Mechanical Engineers in 1994, the Outstanding Book Awards from the Japan Society for Fuzzy Theory and Systems in 1995, 1999 IFAC World Congress Best Poster Paper Prize in 1999, 2000 IEEE Transactions on Fuzzy Systems Outstanding Paper Award in 2000, the Best Paper Selection at 2005 American Control Conference in Portland, USA, in 2005, the Best Paper Award at 2013 IEEE International Conference on Control System, Computing and Engineering in Penang, Malaysia, in 2013, the Best Paper Finalist at 2013 International Conference on Fuzzy Theory and Its Applications, Taipei, Taiwan in 2013. His research interests include intelligent systems and control, nonlinear systems control, robotics, brain-machine interface and their applications. He co-authored (with Hua O. Wang) the book Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach (Wiley-Interscience, 2001).

He has served as an Associate Editor for Automatica and for the IEEE Transactions on Fuzzy Systems, and is on the IEEE Control Systems Society Conference Editorial Board. He is a fellow of IEEE and IFSA.

Motoyasu Tanaka (S'05 - M'12) received his B.E., M.S., and Ph.D. degrees in Engineering from the Department of Mechanical Engineering and Intelligent Systems at the University of Electro- Communications, Japan in 2005, 2007, and 2009, respectively. From 2009 to 2012, he worked at Canon, Inc., Tokyo, Japan. He is currently an Assistant Professor in the Department of Mechanical Engineering and Intelligent Systems at the University of Electro-Communications. His research interests include biologically inspired robotics and dynamic based nonlinear control. He received the IEEE Robotics and Automation Society Japan Chapter Young Award from the IEEE Robotics and Automation Society Japan Chapter in 2006. He is a member of the IEEE, SICE, and RSJ.

Ying-Jen Chen (M'12) received the B.S. degree in electrical engineering from the National Taiwan Ocean University, Keelung, Taiwan, in 2002, the M.S. degree in electrical engineering from the Lunghwa University of Science and Technology, Taoyuan, Taiwan, in 2004 and the Ph.D. degree in electrical engineering from the National Central University, Jhongli, Taiwan, in 2011. He is currently a postdoctoral researcher with the Department of Mechanical Engineering and Intelligent Systems, the University of Electro-Communications, Tokyo, Japan. His current research interests are in the areas of fuzzy control system, neural networks, and pattern recognition.

Hua O. Wang (M'94-SM'01) received the B.S. degree from the University of Science and Technology of China (USTC), Hefei, China, in 1987, the M.S. degree from the University of Kentucky, Lexington, KY, in 1989, and the Ph.D. degree from the University of Maryland, College Park, MD, in 1993, all in Electrical Engineering.

He has been with Boston University where he is currently an Associate Professor of Aerospace and Mechanical Engineering since September 2002. He was with the United Technologies Research Center, East Hartford, CT, from 1993 to 1996, and was a faculty member in the Department of Electrical and Computer Engineering at Duke University, Durham, NC, from 1996 to 2002. Dr. Wang served as the Program Manager (IPA) for Systems and Control with the U.S. Army Research Office (ARO) from August 2000 to August 2002. During 2000 - 2005, he also held the position of Cheung Kong Chair Professor and Director with the Center for Nonlinear and Complex Systems at Huazhong University of Science and Technology, Wuhan, China.

Dr. Wang is a recipient of the 1994 O. Hugo Schuck Best Paper Award of the American Automatic Control Council, the 14th IFAC World Congress Poster Paper Prize, the 2000 IEEE Transactions on Fuzzy Systems Outstanding Paper Award. His research interests include control of nonlinear dynamics, intelligent systems and control, networked control systems, robotics, cooperative control, and applications. He co-authored (with Kazuo Tanaka) the book Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach (Wiley-Interscience, 2001). Dr. Wang has served as an Associate Editor for the IEEE Transactions on Automatic Control and was on the IEEE Control Systems Society Conference Editorial Board. He is an Editor for the Journal of Systems Science and Complexity. He is an appointed member of the 2006 Board of Governors of the IEEE Control Systems Society and a senior member of IEEE.