2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

A Descriptor System Approach to Estimating Domain of Attraction for Non-Polynomial Systems via LMI optimizations

Hiroyuki Ichihara†

***Abstract*— A computational methodology of estimating the domain of attraction (DA) is addressed for non-polynomial systems by a descriptor system approach. For an existing technique of approximating a non-polynomial function, a further investigation is conducted on the existence of upper and lower polynomial bounds of the non-polynomial function. In formulation of the DA analysis conditions, an implicit form and a generalized Lyapunov function are utilized for dealing with the non-polynomial systems in polynomial fashion and provide two stability conditions which can be reduced to linear matrix inequality (LMI) problems. A relation between these conditions is also discussed. Numerical examples illustrate our DA analysis method.**

I. Introduction

Estimating the domain of attraction (DA) of an equilibrium point is an important problem in nonlinear dynamical systems. For the purpose of estimating the DA, level sets of Lyapunov functions (LF) are often used [1]. Recently, computational methods of an estimating the DA using the LF have been developed for polynomial systems [2], [3], [4], [5], [6]. These methods utilize solutions of a linear matrix inequality (LMI) or semidefinite programming (SDP) problem through sum of squares (SOS) [7] or square matricial representation (SMR) [8], [9]. Inconveniently, actual plants such as mechanical and biological systems are not polynomial systems but descriptor non-polynomial systems. For dealing a non-polynomial system with polynomial fashion, there have been two approaches that provide a worst-case guarantee: the first one is replacing non-polynomial terms with new variables of polynomial and adding new dynamical equations [10]; the second one is replacing non-polynomial terms with Taylor expansions and the remainders having intervals [11], [12]. In a computational method, stability and stabilization for implicit polynomial systems [13] and parameter-dependent descriptor systems [14] have been investigated. In particular, for an implicit form, it is known that a generalized LF introducing redundant variables is available for stability and performance analysis [14], [15].

In this paper, a method of estimating the DA is investigated for a descriptor non-polynomial system. The non-polynomial terms are approximated by polynomials. To add a consideration of the existing approximate technique, we confirm an existence of upper and lower polynomial bounds of the terms. For an implicit form of the system, a generalized LF [14], [15], [16], [17] is proposed to obtain a scalar type stability condition for estimating the DA. In addition to this, a matrix type stability condition is also proposed using the polynomial annihilator [18]. Both of the conditions are reduced to LMIs through SOS or SMR. Numerical examples of estimating the DA are shown.

This paper is organized as follows. Section II introduces the problem formulation. Section III describes polynomial approximation of functions. Section IV presents proposed analysis methods. Section V explains optimization technique for polynomial formulation so far. Section VI illustrates numerical examples. Lastly, VII concludes the paper with remarks.

Notation:

- vert $\mathcal{R}$: vertices of a polytope $\mathcal{R}$
- $A \succ 0$: positive definite matrix
- A': transpose of matrix A
- 0_n: zero vector of size n
- $0_{n\times n}$: zero matrix of size $n \times n$
- I_n: identity matrix of size n

II. Preliminaries

Consider the descriptor non-polynomial system

$$\begin{aligned}&\left[E_0(x(t)) + \sum_{i=1}^{r} F_i(x(t))g_i(x_{\tau_i}(t))\right]\dot{x}(t)\\ &= a_0(x(t)) + \sum_{i=1}^{r} b_i(x(t))g_i(x_{\tau_i}(t)),\ x_0 = x(0)\end{aligned} \tag{1}$$

where $x = \begin{bmatrix}x_1, \ldots, x_n\end{bmatrix}' \in \mathbb{R}^n$ is the state, $x_0 \in \mathbb{R}^n$ is the initial state, the functions $E_0, F_1, \ldots,\ F_r : \mathbb{R}^n \to \mathbb{R}^{n\times n}$, $a, b_1, \ldots, b_r\ :\ \mathbb{R}^n \to \mathbb{R}^n$ are polynomials, $\tau_1,\ \ldots, \tau_r \in \{1, \ldots, n\}$ are indexes, and the functions $g_1, \ldots, g_r : \mathbb{R} \to \mathbb{R}$ are non-polynomials. It is assumed for convenience that

$$a(0_n) + \textstyle\sum_{i=1}^{r} b_i(0_n)g_i(0) = 0_n.$$

Most of physical systems satisfy this assumption. In the case where

$$E_0(x) = I_n,\ F_i(x) = 0_{n\times n},\ i = 1, \ldots, r \tag{2}$$

holds in (1), it is the (non-descriptor) non-polynomial system.

For the system (1), the origin is assumed to be the equilibrium point of interest. Let $\phi(t; x_0) \in \mathbb{R}^n$ be the solution of the system (1) for $t \geq 0$. The DA of the origin defined by

$$\mathcal{D} = \left\{x_0 \in \mathbb{R}^n :\ \lim_{t\to\infty} \phi(t; x_0) = 0_n\right\}$$

is the set of initial states which converge to the origin asymptotically.

†School of Science and Technology, Meiji University, Kawasaki 214-8571, Japan.

Let $v : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable, positive definite and radially unbounded function. v is supposed to be an LF for the origin in (1), i.e. the time derivative of v along the trajectories of (1) is locally negative definite. Then the level set

$$\mathcal{V}(c) = \{x \in \mathbb{R}^n : \; v(x) \le c\} \setminus \{0_n\}$$

is an estimate of $\mathcal{D}$ if $\dot{v}(x) < 0$ for all $x \in \mathcal{V}(c)$.

The problem addressed in this paper is as follows: for a chosen LF, computing the largest estimate $\mathcal{V}(c^*)$ where

$$c^* = \sup_{c>0} \; c \quad \text{s.t.} \quad \dot{v}(x) < 0 \quad \forall x \in \mathcal{V}(c). \tag{3}$$

For simplicity, v is a quadratic function in this paper, i.e. $v(x) = x'Px$ where P is a positive definite matrix of compatible dimension. This does not limit our method to quadratic Lyapunov functions. For a higher degree LF, see Remark 4.7.

III. Taylor expansion of Functions

Let us define a set on $\mathcal{V}(c)$ as

$$\mathcal{V}_{\tau_i}(c) = \left\{x_{\tau_i} \in \mathbb{R} : \; x \in \mathcal{V}(c)\right\}.$$

It is assumed that the first k derivatives of non-polynomial g_i are continuous on $\mathcal{V}_{\tau_i}(c)$. Then, for each $x_{\tau_i} \in \mathcal{V}_{\tau_i}(c)$, there exists a parameter $\theta_i \in \mathbb{R}$ such that g_i is equivalent to the Taylor expansion up to degree $k-1$ and the Lagrange form of the remainder:

$$g_i(x_{\tau_i}) = h_i(x_{\tau_i}) + \theta_i \frac{x_{\tau_i}^k}{k!} \tag{4}$$

where

$$h_i(x_{\tau_i}) = \sum_{j=0}^{k-1} \left. \frac{d^j g_i(x_{\tau_i})}{dx_{\tau_i}^j} \right|_{x_{\tau_j}=0} \frac{x_{\tau_i}^j}{j!}.$$

In the form (4), θ_i exists under the interval [12]:

$$\theta_i \in [\underline{\theta}_i(c) \quad \bar{\theta}_i(c)] \tag{5}$$

where

$$\underline{\theta}_i \le \frac{d^k g_i(x_{\tau_i})}{dx_{\tau_i}^k} \le \bar{\theta}_i \quad \forall x_{\tau_i} \in \mathcal{V}_{\tau_i}(c).$$

By using the interval above, it is possible to show that $\underline{\theta}_i$ and $\bar{\theta}_i$ construct polynomial functions which provide upper and lower bounds on g_i as follows:

Lemma 3.1: If there exist $\underline{\theta}_i$ and $\bar{\theta}_i$ satisfying (5), then g_i has polynomial bounds $\underline{\rho}_i^{(k)}$ and $\bar{\rho}_i^{(k)}$ for all $x_{\tau_i} \in \mathcal{V}_{\tau_i}(c)$ such as

$$\frac{\underline{\rho}_i^{(k)}(x_{\tau_i})}{x_{\tau_i}^k} \le \frac{g_i(x_{\tau_i})}{x_{\tau_i}^k} \le \frac{\bar{\rho}_i^{(k)}(x_{\tau_i})}{x_{\tau_i}^k}, \; x_{\tau_i} \ne 0 \tag{6}$$

where

$$\begin{bmatrix} \underline{\rho}_i^{(k)}(x_{\tau_i}) \\ \bar{\rho}_i^{(k)}(x_{\tau_i}) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} h_i(x_{\tau_i}) + \begin{bmatrix} \underline{\theta}_i(c) \\ \bar{\theta}_i(c) \end{bmatrix} \frac{x_{\tau_i}^k}{k!} \tag{7}$$

and $\underline{\rho}_i^{(k)}(0) = \bar{\rho}_i^{(k)}(0) = g_i(0)$.

Proof: Assume that (5) holds. In the case where $x_{\tau_i}^k > 0$, one has

$$\underline{\theta}_i(c) x_{\tau_i}^k \le \theta_i x_{\tau_i}^k \le \bar{\theta}_i(c) x_{\tau_i}^k \quad \forall x_{\tau_i} \in \mathcal{V}_{\tau_i}(c).$$

Similarly, in the case where $x_{\tau_i}^k < 0$, one also has

$$\bar{\theta}_i(c) x_{\tau_i}^k \le \theta_i x_{\tau_i}^k \le \underline{\theta}_i(c) x_{\tau_i}^k \quad \forall x_{\tau_i} \in \mathcal{V}_{\tau_i}(c).$$

Using (4) and (7), we have

$$\begin{aligned} \underline{\rho}_i^{(k)}(x_{\tau_i}) \le g_i(x_{\tau_i}) \le \bar{\rho}_i^{(k)}(x_{\tau_i}), \; x_{\tau_i}^k > 0 \\ \bar{\rho}_i^{(k)}(x_{\tau_i}) \le g_i(x_{\tau_i}) \le \underline{\rho}_i^{(k)}(x_{\tau_i}), \; x_{\tau_i}^k < 0, \end{aligned}$$

which implies (6). From (4) and (7) again, it is obvious that $\underline{\rho}_i^{(k)}(0) = \bar{\rho}_i^{(k)}(0) = g_i(0)$. ■

For multiple non-polynomial functions, let us define the rectangle

$$\mathcal{R} = [\underline{\theta}_1, \bar{\theta}_1] \times \cdots \times [\underline{\theta}_r, \bar{\theta}_r].$$

IV. Estimate Conditions

By introducing the approximation (4), a solution for the problem of estimating DA has been investigated for the non-descriptor non-polynomial systems as follows:

Lemma 4.1 ([12]): $\mathcal{V}(c)$ is an estimate DA of the system (1) with (2) if there exist a scalar $c(>0)$ and a matrix $P(\succ 0)$ such that

$$\bar{p}(x) + \bar{q}(x)'\theta < 0 \quad \forall (x, \theta) \in \mathcal{V}(c) \times \text{vert } \mathcal{R} \tag{8}$$

where

$$\begin{aligned} &\bar{p}(x) = \frac{\partial v(x)}{\partial x} a_1(x), \; \bar{q}_i(x) = \frac{\partial v(x)}{\partial x} a_{2,i}(x) \\ &a_1(x) = a_0(x) + B(x)h(x), \; a_{2,i}(x) = b_i(x) \frac{x_{\tau_i}^k}{k!} \\ &B = \left[b_1, \ldots, b_r\right], \; h = \left[h_1, \ldots, h_r\right]' \\ &\bar{q} = \left[\bar{q}_1, \ldots, \bar{q}_r\right]', \; \theta = \left[\theta_1, \ldots, \theta_r\right]'. \end{aligned}$$

In the preceding paragraph, we will concentrate on descriptor system (1) again. The system (1) is rewritten to

$$\left[E_1(x) + \sum_{i=1}^r E_{2,i}(x)\theta_i\right]\dot{x} = a_1(x) + \sum_{i=1}^r a_{2,i}(x)\theta_i$$

by (4) where

$$E_1(x) = E_0(x) + \sum_{i=1}^r F_i(x) h_i(x_{\tau_i}), \; E_{2,i}(x) = F_i(x) \frac{x_{\tau_i}^k}{k!}.$$

Moreover the system above can be represented as

$$0 = f_1(\zeta) + \textstyle\sum_{i=1}^r f_{2,i}(\zeta)\theta_i.$$

where

$$f_1(\zeta) = a_1(x) - E_1(x)\dot{x}, \; f_{2,i}(\zeta) = a_{2,i}(x) - E_{2,i}(x)\dot{x}.$$

From this system representation, the system (1) is written to the implicit form:

$$\mathcal{E}\dot{\zeta} = f(\zeta, \theta), \quad \theta \in \mathcal{R} \tag{9}$$

where

$$\mathcal{E} = \begin{bmatrix} I_n & 0_{n\times n} \\ 0_{n\times n} & 0_{n\times n} \end{bmatrix},\ \zeta = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$
$$f(\zeta,\theta) = \begin{bmatrix} \dot{x} \\ f_1(\zeta) \end{bmatrix} + \sum_{i=1}^{r} \begin{bmatrix} 0_n \\ f_{2,i}(\zeta) \end{bmatrix} \theta_i.$$

For the system (9), consider a LF candidate:

$$v^{\#}(x) = \zeta'(P^{\#})'\mathcal{E}\zeta = \zeta'\mathcal{E}'P^{\#}\zeta \tag{10}$$

where

$$P^{\#} = \begin{bmatrix} P & 0_{n\times n} \\ S & R \end{bmatrix}.$$

It is called as the generalized LF and is chosen for stability analysis [14], [15]. From the fact that

$$v^{\#}(x) = v(x),$$

both the level sets of $v^{\#}$ and v are the same for a level c. Let us introduce polynomials

$$p(\zeta) = 2\zeta'(P^{\#})' \begin{bmatrix} \dot{x} \\ f_1(\zeta) \end{bmatrix}$$
$$q_i(\zeta) = 2\zeta'(P^{\#})' \begin{bmatrix} 0_n \\ f_{2,i}(\zeta) \end{bmatrix}$$
$$q = \begin{bmatrix} q_1, \dots, q_r \end{bmatrix}'.$$

The following theorem gives conditions on estimating the DA of (1).

Theorem 4.1: $\mathcal{V}(c)$ is an estimate DA of the system (1) if there exist a scalar $c(>0)$, matrices $P(\succ 0)$, S and R such that

$$p(\zeta) + q(\zeta)'\theta < 0 \quad \forall (x,\dot{x},\theta) \in \mathcal{V}(c) \times \mathbb{R}^n \setminus \{0_n\} \times \mathrm{vert}\ \mathcal{R}. \tag{11}$$

Proof: Assume that (11) is satisfied. One can find that p is the Taylor expansion of $\dot{v}^{\#}$ truncated at degree $k-1$ and that $q'\theta$ is the remainder of the truncation in the Lagrange form. Lemma 3.1 represents that non-polynomial function g_i has upper and lower polynomial bounds in $\mathcal{V}_{\tau_i}(c)$ and that such bounds are determined by any vectors $\theta_i \in [\underline{\theta}_i, \bar{\theta}_i]$. Thus the constraint implies that

$$p(\zeta) + q(\zeta)'\theta < 0 \ \ \forall (x,\dot{x},\theta) \in \mathcal{V}(c) \times \mathbb{R}^n \setminus \{0_n\} \times \mathcal{R}$$

because $p + q'\theta$ is affine in θ and $\mathcal{R}$ is a convex polytope. By the fact that

$$\dot{v}(x) = \dot{v}^{\#}(x) = p(\zeta) + q(\zeta)'\theta,$$

one can obtain $\dot{v}(x) < 0$ for all $x \in \mathcal{V}(c)$. ■

Remark 4.1: It is possible to obtain a numerical solution for a relaxed problem of (11) through SOS or SMR. More precisely, the relaxed problem is not a convex but a quasi-convex, which can reach to an optimal solution by using a line search method and LMI.

Lemma 4.2: If (2) holds in the system (1), the stability condition (11) is equivalent to (8).

Proof: Assume that (2) holds. Then, the system (1) with (2) is written as

$$\dot{x} = a_1(x) + a_2(x)'\theta$$

where $a_2 = \begin{bmatrix} a_{2,1} \cdots a_{2,r} \end{bmatrix}'$. Then we have

$$\begin{aligned} p(\zeta) &= 2\zeta'(P^{\#})' \begin{bmatrix} \dot{x} \\ a_1(x) - \dot{x} \end{bmatrix} \\ &= 2\zeta'(P^{\#})' \begin{bmatrix} a_1(x) + a_2(x)'\theta \\ -a_2(x)'\theta \end{bmatrix} \\ &= 2\Big(x'Pa_1(x) + x'Pa_2(x)'\theta \\ &\qquad - (x'S' + \dot{x}'R')a_2(x)'\theta\Big) \\ q_i(\zeta) &= 2\zeta'(P^{\#})' \begin{bmatrix} 0_n \\ a_{2,i}(x) \end{bmatrix} \\ &= 2(x'S' + \dot{x}'R')a_{2,i}(x). \end{aligned}$$

By taking into account of $\partial v(x)/\partial x = x'P$, we obtain

$$p(\zeta) + q(\zeta)'\theta = \bar{p}(x) + \bar{q}(x)'\theta,$$

which is the assertion. ■

Remark 4.2: An advantage of Theorem 4.1 over Lemma 4.1 is to be able to handle with the descriptor system. As Lemma 4.2 says, the condition in Theorem 4.1 for the non-descriptor system is equivalent to that in Lemma 4.1. In this sense, Theorem 4.1 gives a generalized condition for the DA analysis. In the case of the non-descriptor system, however, the computational cost is more expensive than Lemma 4.1 because the sizes of the vectors of monomials to construct SOS or SMR become larger due to $\dot{x}$ in ζ. Moreover Theorem 4.1 needs the additional decision variables S and R in $P^{\#}$. As a result, these variables may not contribute to expanding the estimate DA.

The implicit form (9) has a representation such as

$$\mathcal{E}\dot{\zeta} = \mathcal{A}(x,\theta)\zeta, \quad \theta \in \mathcal{R} \tag{12}$$

where

$$\begin{aligned} \mathcal{A}(x,\theta) &= \begin{bmatrix} 0_{n\times n} & I_n \\ \mathcal{A}_{21}(x,\theta) & \mathcal{A}_{22}(x,\theta) \end{bmatrix} \\ \mathcal{A}_{21}(x,\theta) &= A_1(x) + B(x)\Delta(\theta)C(x_\tau) \\ \mathcal{A}_{22}(x,\theta) &= -E_1(x) - E_2(x)(\theta \otimes I_n) \\ E_2 &= \begin{bmatrix} E_{2,1}, \dots, E_{2,r} \end{bmatrix},\ \Delta(\theta) = \mathrm{diag}\left(\theta_1, \dots, \theta_r\right). \end{aligned}$$

In addition, $A_1 : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ and $C : \mathbb{R} \to \mathbb{R}^{r\times n}$ satisfy

$$a_1(x) = A_1(x)x,\ \begin{bmatrix} x_{\tau_1}^k, \dots, x_{\tau_r}^k \end{bmatrix}' \frac{1}{k!} = C(x_\tau)x$$

where $x_\tau = \begin{bmatrix} x_{\tau_1}, \dots, x_{\tau_r} \end{bmatrix}'$.

Remark 4.3: The representation of A_1 is not unique in general. There has been no rule to decide a best one before performing stability analysis. Alternatively, one can use a feasible solution of the following problem:

$$\begin{aligned} &\text{find} \ \ A_1(x) \\ &\text{s.t.} \ \ a_1(x) = A_1(x)x \ \ \forall x \in \mathbb{R}^n \\ &\qquad \mathrm{He}\{PA_1(0)\} \prec 0. \end{aligned} \tag{13}$$

The first condition gives possible representation of A_1, and the second condition guarantees stability at the equilibrium point. Since the first condition is equivalent to linear simultaneous equations [18], the problem above can be reduced to a semidefinite programming (SDP) problem or LMI problem.

Theorem 4.2: $\mathcal{V}(c)$ is an estimate DA of the system (1) if there exist $c(>0)$, $P(\succ 0)$, S, R and a matrix polynomial $N : \mathbb{R}^n \to \mathbb{R}^{2n\times n}$ such that $N(x)x = 0_{2n}$ for all $x \in \mathbb{R}^n$ and

$$\begin{bmatrix} 0_{n\times n} & P \\ P & 0_{n\times n} \end{bmatrix} + \mathsf{He}\left\{ \begin{bmatrix} S' \\ R' \end{bmatrix} \begin{bmatrix} \mathcal{A}_{21}(x,\theta)' \\ \mathcal{A}_{22}(x,\theta)' \end{bmatrix}' + \begin{bmatrix} N(x) & 0_{2n\times n} \end{bmatrix} \right\} \prec 0$$
$$\forall (x,\theta) \in \mathcal{V}(c) \times \text{vert}\ \mathcal{R}. \quad (14)$$

Proof: Assume that (14) holds. Then (14) is also satisfied for all $\theta \in \mathcal{R}$ because $\mathcal{A}_{21}$ and $\mathcal{A}_{22}$ are affine in θ. From (12), one can obtain the relation

$$0 = \mathcal{A}_{21}(x,\theta)x + \mathcal{A}_{22}(x,\theta)\dot{x}.$$

Pre- and post-multiplying ζ' and its transpose to (14), and using the relation above and $N(x)x = 0$ leads to

$$0 > 2x'P\dot{x} = \dot{v}(x) \quad \forall x \in \mathcal{V}(c).$$

Thus $\mathcal{V}(c)$ is an estimate DA of (12), i.e. an estimate DA of (1). ■

Remark 4.4: N is said to be a polynomial annihilator of x [18]. As the definition implies that it annihilates when x is post-multiplied. The role of the annihilator is choosing a best representation of the matrix condition if one eliminates vector ζ or x from the corresponding scalar condition of a quadratic form. In this case, one may initially consider an annihilator $\mathcal{N} : \mathbb{R}^{2n} \to \mathbb{R}^{2n\times 2n}$ such as $\mathcal{N}(\zeta)\zeta = 0_{2n}$. Then the condition $2\zeta'(P^{\#})'A(x,\theta)\zeta < 0$ is rewritten to

$$\mathsf{He}\left\{(P^{\#})'A(x,\theta) + \mathcal{N}(\zeta)\right\} \prec 0.$$

Since $(P^{\#})'A(x,\theta)$ depends only x in ζ, $\mathcal{N}$ could be reduced to $\mathcal{N}_x : \mathbb{R}^n \to \mathbb{R}^{2n\times 2n}$ such as $\mathcal{N}_x(x)\zeta = 0_{2n}$. Suppose that $\mathcal{N}_x(x)$ has the structure $\begin{bmatrix} N(x) & N_0(x) \end{bmatrix}$. Here $N_0 : \mathbb{R}^n \to \mathbb{R}^{2n\times n}$ is $0_{2n\times n}$ because $N_0(x)\dot{x}$ must be 0_{2n}.

Remark 4.5: It is possible to obtain an annihilator by solving an LP problem. Since (14) can be reduced to an SDP problem, or an LMI problem, it is obvious that one can find the annihilator and other decision variables simultaneously. For a fixed degree of $N(x)$, solutions of the LP problem cover all possible annihilators. Thus, by using Lemma 4.2, one can choose a annihilator without realizing that it is a best one for stability analysis.

Remark 4.6: In Remark 4.3, the representation of A_1 is not unique because A_1 has a freedom of the representation by using an annihilator. Indeed, one can define $N_{A_1} : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ such as $N_{A_1}(x)x = 0_n$. In Lemma 4.2, N absorbs the role of N_{A_1}. In this sense, Lemma 4.2 solves the problem for one of the best representations of A_1. To see this, replace A_1 in (14) as $A_1 + N_{A_1}$. Then

$$\begin{bmatrix} S' \\ R' \end{bmatrix} \begin{bmatrix} \mathcal{A}_{21}' + N_{A_1}' \\ \mathcal{A}_{22}' \end{bmatrix}' = \begin{bmatrix} S' \\ R' \end{bmatrix} \begin{bmatrix} \mathcal{A}_{21}' \\ \mathcal{A}_{22}' \end{bmatrix}' + \begin{bmatrix} S' \\ R' \end{bmatrix} \begin{bmatrix} N_{A_1} & 0_{n\times n} \end{bmatrix}.$$

If one redefine

$$N + \begin{bmatrix} S' \\ R' \end{bmatrix} N_{A_1}$$

as N, then one can obtain (14) again. However it is difficult to pick up an optimal N_{A_1} from N.

Remark 4.7: The LFs have been quadratic functions so far in this paper. It does not limit to the availability of the stability conditions in Theorems 4.1 and 4.2. Indeed, we can choose a matrix polynomial $P : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ that is positive definite for all $x \in \mathcal{V}(c)$ in Theorem 4.1. Then we may add $x^T\dot{P}(x)x$ to $q(\zeta)$ in (11). On the other hand, in Theorem 4.2, for example, using a similar technique as [13], we can choose $P(x)$ as

$$\begin{bmatrix} I_n \\ \Theta(x) \end{bmatrix}' \mathcal{P} \begin{bmatrix} I_n \\ \Theta(x) \end{bmatrix}$$

where $\mathcal{P}(\succ 0) \in \mathbb{R}^{2n\times 2n}$ and $\Theta : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ is a matrix polynomial. Then, letting $\tilde{P}(x)$ be

$$\begin{bmatrix} 0_{n\times n} \\ \Phi(x) \end{bmatrix}' \mathcal{P} \begin{bmatrix} I_n \\ \Theta(x) \end{bmatrix}$$

where Φ satisfies $\dot{\Theta}(x)x = \Phi(x)\dot{x}$, we may replace the first term in the left side of (14) with

$$\begin{bmatrix} 0_{n\times n} & P(x) + \tilde{P}(x)' \\ P(x) + \tilde{P}(x) & 0_{n\times n} \end{bmatrix}.$$

Thus any information on the range of $\dot{x}$ does not required for stability analysis while other methods based on matrix conditions often need the range.

V. Polynomial Optimization

We have investigated the conditions for estimating the DA in Lemma 4.1, Theorem 4.1 and 4.2 so far. It is possible to relax such conditions through the technique of SOS or SMR into LMIs for a fixed estimate DA. In this section, we will review the SOS technique.

Let us consider a symmetric matrix polynomial

$$F(x) = \textstyle\sum_{\alpha\in\mathcal{F}} F_\alpha x^\alpha$$

where $x \in \mathbb{R}^n$, $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$, $F : \mathbb{R}^n \to \mathbb{R}^{q\times q}$ and

$$\mathcal{F} = \left\{\alpha \in \mathbb{R}^n : \textstyle\sum_{i=1}^n \alpha_i \le 2N\right\}.$$

If F has a form

$$F(x) = \textstyle\sum_{j=1}^{\ell} G_j(x)'G_j(x)$$

then F is said to be *SOS* where $G_j : \mathbb{R}^n \to \mathbb{R}^{1\times q}$. Let G be such that $G = \begin{bmatrix} G_1' \cdots G_\ell' \end{bmatrix}'$. Obviously, $F = G'G$. Here note that G has forms of

$$G(x) = Y(I_q \otimes z_N(x)) = \tilde{Y}(z_N(x) \otimes I_q)$$

where $z_N : \mathbb{R}^n \to \mathbb{R}^s$ includes monomials whose maximum degree is at most N, Y and $\tilde{Y}$ are matrices of suitable dimension. In particular, a maximum s is $(n+N)!/(N!\cdot n!)$. Taking the former form of G, F is rewritten by

$$F(x) = (I_q \otimes z_N(x))' Q (I_q \otimes z_N(x)) \tag{15}$$

where $Q = Y'Y$ is a positive semidefinite matrix. In fact, F is SOS with respect to z_N if and only if there exists a positive semidefinite matrix Q such that (15) holds for all $x \in \mathbb{R}^n$ [19], [20]. If $F(\theta)$ is SOS, then $F(\theta)$ is positive semidefinite for all $x \in \mathbb{R}^n$. To check the existence of Q, one can solve a standard LMI problem

$$\text{find } Q \succeq 0 \ \text{ s.t. } (Q, I_q \otimes A_\alpha)_q = F_\alpha \ \forall \alpha \in \mathcal{F} \tag{16}$$

where A_α satisfies $z_N z_N' = \sum_{\alpha \in \mathcal{F}} A_\alpha \theta^\alpha$,

$$\begin{aligned} (X,Y)_q &= \mathrm{tr}_q(X'Y) \\ \mathrm{tr}_q(M) &= \begin{bmatrix} \mathrm{tr}\, M_{(11)} & \cdots & \mathrm{tr}\, M_{(1q)} \\ \vdots & \ddots & \vdots \\ \mathrm{tr}\, M_{(q1)} & \cdots & \mathrm{tr}\, M_{(qq)} \end{bmatrix} \end{aligned}$$

for X, $Y \in \mathbb{R}^{nq \times nq}$, $M \in \mathbb{R}^{nq \times nq}$, M is divided into $q \times q$ blocks and each matrix is $M_{(jk)} \in \mathbb{R}^{n \times n}$ $(j, k = 1, \ldots, q)$.

VI. Numerical Examples

In this section, numerical examples are illustrated by solving SOS relaxation problems. The software environment is as follows: MATLAB 7.10.0 (R2010a), YALMIP (R20100702) [21] and SeDuMi Ver. 1.3 [22]. For the Taylor expansion up to degree k in (4) and (5), c_k denotes a maximized c.

A. Example 1

Consider a descriptor non-polynomial system

$$\begin{cases} (2 - \cos x_1)\dot{x}_1 = -x_1 + x_1^2/3 \\ \dot{x}_2 = x_1 - 3x_2 + \sin x_2, \end{cases}$$

which is rewritten in the form (1) with

$$\begin{aligned} &E_0(x) = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix},\ a_0(x) = \begin{bmatrix} -x_1 + x_1^2/3 \\ x_1 - 3x_2 \end{bmatrix} \\ &F_1(x) = \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix},\ F_2(x) = 0_{2\times 2} \\ &b_1(x) = \begin{bmatrix} 0 \\ 0 \end{bmatrix},\ b_2(x) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &g_1(x_{\tau_1}) = \cos x_{\tau_1},\ \tau_1 = 1 \\ &g_2(x_{\tau_2}) = \sin x_{\tau_2},\ \tau_2 = 2. \end{aligned}$$

For the system, we use a LF candidate

$$v(x) = x_1^2 + x_2^2, \tag{17}$$

i.e. $P = I_2$. Then $\mathcal{V}_{\tau_1}(c) = \mathcal{V}_{\tau_2}(c) = [-\sqrt{c}, \sqrt{c}]$.

In the case where $k = 1$, we have

$$\begin{aligned} h_1(x_{\tau_1}) = 1, &\quad \frac{dg_1(x_{\tau_1})}{dx_{\tau_1}} = -\sin x_{\tau_1} \\ h_2(x_{\tau_2}) = 0, &\quad \frac{dg_2(x_{\tau_2})}{dx_{\tau_2}} = \cos x_{\tau_2} \end{aligned}$$

and the interval (5) with

$$\underline{\theta}_1(c) = -y,\ \bar{\theta}_1(c) = y,\ y = \begin{cases} \sin\sqrt{c}, \text{ if } \sqrt{c} \le \pi/2 \\ 1, \text{ otherwise} \end{cases}$$

$$\underline{\theta}_2(c) = z,\ \bar{\theta}_2(c) = 1,\ z = \begin{cases} \cos\sqrt{c}, \text{ if } \sqrt{c} \le \pi \\ -1, \text{ otherwise.} \end{cases}$$

Solving the problem in Theorem 4.1 with maximizing c_1, we find the problem is feasible at $c_1 = 1.1$.

From (13), we obtain

$$A_1(x) = \begin{bmatrix} -1.0000 + 0.3333x_1 & 0.0000 \\ 1.0000 & -3.0000 \end{bmatrix}$$

from

$$a_1(x) = a_0(x) + B(x)h(x) = \begin{bmatrix} -x_1 + x_1^2/3 \\ x_1 - 3x_2 \end{bmatrix}.$$

In a similar way, we obtain $c_1 = 1.2$ by Theorem 4.2 with

$$N(x) = \begin{bmatrix} 0.1417x_2 & -0.1417x_1 \\ -0.0237x_2 & 0.0237x_1 \\ 0.1034x_2 & -0.1034x_1 \\ 0.0225x_2 & -0.0225x_1 \end{bmatrix}.$$

The results of maximizing $c_k (k = 1, \ldots, 7)$ are shown in Table I, in which the sequences of c_k by Theorems 4.1 and 4.2 converge to finite values, respectively. In this example, they give an almost same DA.

TABLE I

Maximized c_k for implicit system.

k	1	2	3	4	5	6	7
Theorem 4.1	1.1	2.4	3.1	3.5	4.3	4.5	4.5
Theorem 4.2	1.2	2.4	3.1	3.5	4.3	4.5	4.5

B. Example 2

Consider a non-polynomial system ([12])

$$\begin{cases} \dot{x}_1 = -x_1 + x_2 + 0.5(e^{x_1} - 1) \\ \dot{x}_2 = -x_1 - x_2 + x_1x_2 + x_1 \cos x_1. \end{cases}$$

In a similar way as Example 1, one can obtain the form (1) with

$$\begin{aligned} &E_0(x) = I_2,\ F_1(x) = F_2(x) = 0_{2\times 2} \\ &a_0(x) = \begin{bmatrix} -x_1 + x_2 \\ -x_1 - x_2 + x_1x_2 \end{bmatrix} \\ &b_1(x) = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix},\ b_2(x) = \begin{bmatrix} 0 \\ x_1 \end{bmatrix} \\ &g_1(x_{\tau_1}) = e^{x_{\tau_1}} - 1,\ \tau_1 = 1 \\ &g_2(x_{\tau_2}) = \cos x_{\tau_2},\ \tau_2 = 1. \end{aligned}$$

For the system, we use a LF candidate (17) again.

In the case where $k = 3$, we have

$$\begin{aligned} h_1(x_{\tau_1}) = x_{\tau_1} + 0.5000x_{\tau_1}^2, &\quad \frac{d^3 g_1(x_{\tau_1})}{dx_{\tau_1}^3} = e^{x_{\tau_1}} \\ h_2(x_{\tau_2}) = 1 - 0.5000x_{\tau_2}^2, &\quad \frac{d^3 g_2(x_{\tau_2})}{dx_{\tau_2}^3} = \sin x_{\tau_2} \end{aligned}$$

and the interval (5) with

$$\underline{\theta}_1(c) = e^{-\sqrt{c}},\ \bar{\theta}_1(c) = e^{\sqrt{c}}$$
$$\underline{\theta}_2(c) = -y,\ \bar{\theta}_2(c) = y,\ y = \begin{cases} \sin\sqrt{c},\ \text{if } \sqrt{c} \le \pi/2 \\ 1,\ \text{otherwise.} \end{cases}$$

We obtain $c_3 = 0.28$ from Theorem 4.1. On the other hand, solving (13) gives

$$A_1(x) = \begin{bmatrix} -0.5000 + 0.2500x_1 & 1.0000 \\ 0.5000x_2 - 0.5000x_1^2 & -1.0000 + 0.5000x_1 \end{bmatrix}$$

from

$$a_1(x) = a_0(x) + B(x)h(x) = \begin{bmatrix} -x_1/2 + x_2 + x_1^2/4 \\ -x_2 - x_1^3/2 + x_1x_2 \end{bmatrix}.$$

Theorem 4.2 gives $c_3 = 0.28$ with

$$N(x) = \begin{bmatrix} -0.0624x_2 - 0.0032x_1x_2 - 0.2151x_2^2 & 0.0624x_1 + 0.0032x_1^2 + 0.2151x_1x_2 \\ 0.0269x_2 + 0.0894x_1x_2 - 0.0924x_2^2 & -0.0269x_1 - 0.0894x_1^2 + 0.0924x_1x_2 \\ -0.0169x_2 + 0.0232x_1x_2 - 0.0528x_2^2 & 0.0169x_1 - 0.0232x_1^2 + 0.0528x_1x_2 \\ -0.0375x_2 - 0.0742x_1x_2 + 0.0406x_2^2 & 0.0375x_1 + 0.0742x_1^2 - 0.0406x_1x_2 \end{bmatrix}.$$

The results of maximizing $c_k (k = 1, \dots, 6)$ are shown in Table II. One can observe that the sequences of c_k converge. Theorems 4.1 ane 4.2 give almost same computational results as the conventional method by Lemma 4.1.

TABLE II

MAXIMIZED c_k FOR NON-IMPLICIT SYSTEM.

k	1	2	3	4	5	6
Lemma 4.1	0.06	0.21	0.28	0.31	0.32	0.32
Theorem 4.1	0.06	0.21	0.28	0.31	0.32	0.32
Theorem 4.2	0.06	0.21	0.28	0.31	0.32	0.32

VII. CONCLUSIONS

We have investigated the computational methods of estimating the DA for the descriptor non-polynomial system. For the implicit form of the system, the generalized LF led to the two stability conditions that can be rewritten as LMIs through SOS or SMR. The scalar type condition is a generalization of the existing result, and the matrix type condition is an equivalent condition to the scalar type. These facts have been confirmed by the numerical examples.

A furture work will investigate some synthesis problems using the matrix type condition.

Acknowledgment

The author thanks to Prof. Graziano Chesi and Prof. Masakatsu Kawata for discussion of the topic. The author also thanks to the reviewers for many helpful comments.

REFERENCES

[1] H. K. Khalil, *Nonlinear Systems, 3rd edition*. Prentice Hall, 2001.

[2] Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, "Some control applications of sum of squares programming," in *Proc. IEEE Conf. on Decision and Control*, 2003, pp. 4676–4681.

[3] G. Chesi, "Estimating the domain of attraction for uncertain polynomial systems," *Automatica*, vol. 40, pp. 1981–1986, 2004.

[4] ——, "LMI techniques for optimization over polynomials in control: a survey," *IEEE Trans. Automat. Contr.*, vol. 11, pp. 2500–2510, 2010.

[5] D. Henrion and J. B. Lasserre, "How GloptiPoly is applied to problems in robust and nonlinear control," *IEEE Control Syst. Mag.*, pp. 72–83, 2004.

[6] ——, "Convergent relaxations of polynomial marix inequalities and static output feedback," *IEEE Trans. Automat. Contr.*, vol. 51, pp. 192–202, 2006.

[7] P. A. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Institute of Technology, Pasadena, CA, may 2000.

[8] G. Chesi, A. Tesi, A. Vicino, and R. Genesio, "On convexification of some minimum distance problems," in *Proc. European Control Conf.*, 1999.

[9] G. Chesi, A. Garulli, A. Tesi, and A. Vicino, *Homogeneous Polynomial Forms for Robustness Analysis of Uncertain Systems*. Berlin Heidelberg: Springer-Verlag, 2009.

[10] A. Papachristodoulou and S. Prajna, "Analysis of non-polynomial systems using the sum of squares decomposition," in *Positive Polynomials and Control, LNCIS Vol. 312*. Springer, 2005, pp. 23–43.

[11] G. Chesi, "Domain of attraction: Estimates for non-polynomial systems via lmis," in *Proc. IFAC World Congr.*, 2005.

[12] ——, "Estimating the domain of attraction for non-polynomial systems via LMI optimizations," *Automatica*, vol. 45, pp. 1536–1541, 2009.

[13] D. F. Coutinho, C. E. de Souza, and A. Trofino, "Regional stability analysis of implict polynomial systems," in *Proc. IEEE Conf. on Decision and Control*, 2006, pp. 4224–4229.

[14] I. Masubuchi, Y. Kamitane, A. Ohara, and N. Suda, "H_∞ control for descriptor systems: A matrix inequalities approach," *Automatica*, vol. 33, pp. 669–673, 1997.

[15] M. Kawata and K. Inoue, "Nonlinear H_∞ control for an inverted pendulum system using descriptor representation," *Transaction of SICE*, vol. 36, pp. 650–659, 2000, (in Japanese).

[16] E. Fridman and U. Shaked, "A descriptor system approach to h-infinity control of linear time-delay systems," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 2081–2084, 2002.

[17] Y.-Y. Cao and Z. Lin, "A descriptor system approach to robust stability analysis and controller synthesis," *IEEE Trans. Automat. Contr.*, vol. 49, pp. 2081–2084, 2004.

[18] H. Ichihara, "Optimal control for polynomial systems using matrix sum of squares relaxations," *IEEE Trans. Automat. Contr.*, vol. 54, pp. 1050–1055, 2009.

[19] M. Kojima, "Sums of squares relaxations of polynomial semidefinite programs," Tokyo Instituite of Technology, Tech. Rep. B-397, 2003.

[20] C. W. Scherer and C. W. J. Hol, "Matrix sum-of-squares relaxations for robust semi-definite programs," *Math. Program. Ser. B*, vol. 107, pp. 189–211, 2006.

[21] J. Löfberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conf.*, Taipei, Taiwan, 2004, available from http://control.ee.ethz.ch/˜joloef/yalmip.php.

[22] J. F. Sturm, *SeDuMi: Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones (updated for version 1.05)*, 1998. [Online]. Available: http://sedumi.mcmaster.ca/