Controller Design for Polynomial Nonlinear Systems with Affine Uncertain Parameters

TONG Chang-Fei[1] ZHANG Hui[1] SUN You-Xian[1]

Abstract By means of polynomial decomposition, a control scheme for polynomial nonlinear systems with affine time-varying uncertain parameters is presented. The idea of polynomial decomposition is to convert the coefficients of polynomial into a matrix with free variables, so that the nonnegativity of polynomials with even orders can be checked by linear matrix inequality (LMI) solvers or bilinear matrix inequality (BMI) solvers. Control synthesis for polynomial nonlinear system is based on Lyapunov stability theorem in this paper. Constructing Lyapunov function and finding feedback controller are automatically finished by computer programming with algorithms given in this paper. For multidimension systems with relatively high-order controller, the controller constructed with full monomial base will be in numerous terms. To overcome this problem, the reduced-form controller with minimum monomial terms is derived by the proposed algorithm. Then a suboptimal control aiming at minimum cost performance with gain constraints is advanced. The control scheme achieves effective performance as illustrated by numerical examples.

Key words Nonlinear control, semidefinite programming relaxation, robust control

1 Introduction

The Lyapunov stability theorem has been a cornerstone for nonlinear system analysis for several decades. In principle, the theorem states that a system $\dot{\boldsymbol{x}} = f(\boldsymbol{x})$ with equilibrium at the origin is stable if there exists a positive definite function $V(\boldsymbol{x})$ such that the derivative of $V(\boldsymbol{x})$ along the system trajectories is nonpositive. In recent years, considerable attention has been devoted to the study of polynomial nonlinear systems. Significant progress has been made in the stability analysis of those systems by sum of squares decomposition approach[1~4]. Stability analysis with this methodology is mainly based on Lyapunov stability theorem. Constructing Lyapunov functions is solved by SOSTOOLS[1] which converts the problems into semidefinite programs (SDP) solved using SeDuMi[5] which is a MATLAB toolbox for optimization over symmetric cones. Though stability analysis is solved effectively to some extent, control synthesis for nonlinear systems still remains a stubborn problem since the nonlinear components of variables in the sum of squares (SOS) terms cannot be solved directly. To solve the synthesis problem, an iterative algorithm was proposed by Jarvis-Wloszek[6]. However, the controller designed in [6] is not globally optimal; furthermore, the iterative algorithm may fail to get a solution in some cases, though the system may possess a stabilized controller.

Polynomial nonlinear system with state feedback controller is generally depicted as follows.

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + g \cdot \boldsymbol{u}(\boldsymbol{x}) \tag{1}$$

where $g \in \mathbf{R}^{n\times m}$, $\boldsymbol{f}(\boldsymbol{x}) \in \mathcal{R}_n^n$, and $\boldsymbol{u}(\boldsymbol{x}) \in \mathcal{R}_n^m$ are polynomials with $\boldsymbol{f}(\boldsymbol{0}) = \boldsymbol{u}(\boldsymbol{0}) = \boldsymbol{0}$, and the equilibrium point is at origin $\boldsymbol{x} = \boldsymbol{0}$. The demerit of constructing Lyapunov function with SOSTOOLS is that there are two unknown polynomials combined together in nonlinear form when the set of $(V, \boldsymbol{u})$ satisfies

$$\nabla V \cdot (\boldsymbol{f} + g\boldsymbol{u}) < 0$$

Given the difficulties with Lyapunov-based controller synthesis, it is most striking to find that the new convergence criterion presented in [7] based on the so-called density function ρ has much better convexity properties. Then, Prajna[8] exploited this criterion to solve control synthesis problems.

Unfortunately, the convergence criterion via density function does not involve any information about convergence rate, so the controller designed by this scheme may have slow convergence in some instances. Furthermore, it is difficult to apply density function to guaranteed cost control synthesis. Hence, it is a promising task to develop a tool for control synthesis of polynomial systems for guaranteed cost control. To the best of our knowledge, how to design an optimal controller under cost performance criterion with constrained gains remains an open problem.

In this paper, we propose a novel algorithm for polynomial decomposition and exploit this methodology to study the suboptimal cost control of polynomial nonlinear systems with affine time-varying uncertain parameters. Constructing Lyapunov function and finding suboptimal feedback controller are automatically finished by our software package based on algorithms proposed.

In this paper, $\mathcal{R}_n$ denotes the set of all polynomials in n variables; $\mathcal{R}_n^r$ the set of all polynomial vectors in n variables with r dimension; $\mathcal{R}_{n,d}^r$ the set of all polynomial vectors in n variables with r dimension and the maximum degree of the elements in the vector is d; $\mathbf{Z}_+$ the nonnegative integer set, $\mathbf{Z}_+ = \mathbf{Z}^+ \cup \{0\}$; and

$$c(n,r) = \left(\begin{array}{c} r \\ n \end{array}\right) = \left\{\begin{array}{ll} \dfrac{n!}{r!\,(n-r)!} & r > 0 \\ 1 & r = 0 \\ 0 & r < 0 \end{array}\right.$$

2 Polynomial decomposition and nonnegativity validation

In this section, we set up a decomposition framework by giving several novel definitions to facilitate the expression of our decomposition algorithm. Then the nonnegativity validation problem for polynomials can be transmitted to semidefinite positivity tests of corresponding decomposed matrices.

Definition 1. A monomial $m_r(\boldsymbol{x})$ in n variables is a function defined as $m_r(\boldsymbol{x}) = \prod_{i=1}^n \boldsymbol{x}_i^{r_i}$, for $r_i \in \mathbf{Z}_+$, and the degree of monomial is defined as $\deg(m_r) = \sum_{i=1}^n r_i = r$.

Definition 2. A polynomial $f(\boldsymbol{x}) \in \mathcal{R}_n^1$ is a finite linear combination of monomials,

$$f(\boldsymbol{x}) = \sum_r c_r m_r(\boldsymbol{x}), \text{ with } \quad c_r \in \mathbf{R}$$

The degree of $f(\boldsymbol{x})$ is denoted by $\deg(f) = \max\limits_r \ \deg(m_r)$.

Received November 21, 2006; in revised form May 25, 2007
Supported by National Natural Science Foundation of China (60674028, 60674071), the Key Technologies R & D Program of Zhejiang Province (2006C11066)
1. State Key Laboratory of Industrial Control Technology, Institute of Industrial Process Control, Zhejiang University, Hangzhou 310027, P. R. China
DOI: 10.1360/aas-007-1321

Definition 3. $\boldsymbol{x}^{\{r\}}$ is called the homogeneous monomial base of degree r, for $\boldsymbol{x} \in \mathbf{R}^n$. $\boldsymbol{x}^{\{r\}}$ can be generated as follows.

$$\boldsymbol{x}^{\{r\}} = \begin{bmatrix} x_1^r[E_2(\boldsymbol{x})]^{\{0\}} \\ x_1^{r-1}[E_2(\boldsymbol{x})]^{\{1\}} \\ x_1^{r-2}[E_2(\boldsymbol{x})]^{\{2\}} \\ \vdots \\ x_1[E_2(\boldsymbol{x})]^{\{r-1\}} \\ x_1^0[E_2(\boldsymbol{x})]^{\{r\}} \end{bmatrix}, \text{with} \quad \boldsymbol{x}^{\{0\}} = 1$$

$E_2(\boldsymbol{x})$ is a shift function that deletes the first element of $\boldsymbol{x}$, *i.e.*, $E_2(\boldsymbol{x}) = [\boldsymbol{x}_2, \boldsymbol{x}_3, \cdots, \boldsymbol{x}_n]^{\mathrm{T}}$. The dimension of $\boldsymbol{x}^{\{r\}}$ is calculated using the following formula

$$\mathrm{Dim}(\boldsymbol{x}^{\{r\}}) = c(n+r-1, r) = \frac{(n+r-1)!}{r!(n-1)!}, \text{for } \boldsymbol{x} \in \mathbf{R}^n \quad (2)$$

Definition 4. $\boldsymbol{x}^{|r|}$ is called the full monomial base of degree r for $\boldsymbol{x} \in \mathbf{R}^n$. $\boldsymbol{x}^{|r|}$ is generated from homogeneous monomial bases

$$\boldsymbol{x}^{|r|} = \begin{bmatrix} \boldsymbol{x}^{\{0\}} \\ \boldsymbol{x}^{\{1\}} \\ \vdots \\ \boldsymbol{x}^{\{r\}} \end{bmatrix}$$

The dimension of $\boldsymbol{x}^{|r|}$ is calculated by

$$\mathrm{Dim}(\boldsymbol{x}^{|r|}) = c(n+r, r) = \frac{(n+r)!}{r!n!}, \text{for } \boldsymbol{x} \in \mathbf{R}^n \quad (3)$$

Any polynomial $f(\boldsymbol{x}) \in \mathcal{R}^1_{n,d}$ can be written in the linear form of full monomial base, *i.e.*,

$$f(\boldsymbol{x}) = \boldsymbol{C}x^{|d|}, \text{with } \boldsymbol{C} \in \mathbf{R}^{1\times c(n+d,d)} \quad (4)$$

Furthermore, if $f(\boldsymbol{x})$ is in even order, *i.e.*, $d = 2r$ for $r \in \mathbf{Z}_+$. It can be decomposed in quadratic form

$$f(\boldsymbol{x}) = (\boldsymbol{x}^{|r|})^{\mathrm{T}} \cdot [P_f + L(\boldsymbol{\alpha})] \cdot \boldsymbol{x}^{|r|} \quad (5)$$

where P_f and $L(\boldsymbol{\alpha}) \in \mathbf{R}^{c(n+r,r)\times c(n+r,r)}$ are symmetric matrices, and $\boldsymbol{\alpha}$ is a free variable vector with $(\boldsymbol{x}^{|r|})^{\mathrm{T}} \cdot L(\boldsymbol{\alpha}) \cdot \boldsymbol{x}^{|r|} = 0$. Though decomposition for P_f is not unique, the set of decomposed matrices

$$S(f) = \{P_f + L(\boldsymbol{\alpha}) | (\boldsymbol{x}^{|r|})^{\mathrm{T}} \cdot [P_f + L(\boldsymbol{\alpha})] \cdot \boldsymbol{x}^{|r|} = f(\boldsymbol{x}), \\ (\boldsymbol{x}^{|r|})^{\mathrm{T}} \cdot L(\boldsymbol{\alpha}) \cdot \boldsymbol{x}^{|r|} = 0,\ f(\boldsymbol{x}) \in \mathcal{R}^1_{n,2r}\} \quad (6)$$

is unique and can be taken as the test set for nonnegativity check of $f(\boldsymbol{x})$.

Definition 5. $\boldsymbol{J}(m(\boldsymbol{x})) \in \mathbf{Z}_+^{1\times n}$ is called the exponent mapping of monomial $m(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathbf{R}^n$. If $m(\boldsymbol{x}) = \prod_{i=1}^n \boldsymbol{x}_i^{r_i}$, then $\boldsymbol{J}(m) = [r_1, r_2, \cdots, r_n]$.

Definition 6. $M(f) = P_f + L(\boldsymbol{\alpha}) \in \mathbf{R}^{c(n+r,r)\times c(n+r,r)}$ is called the quadratic decomposed matrix of polynomial $f(\boldsymbol{x}) \in \mathcal{R}^1_{n,2r}$, where P_f and $L(\boldsymbol{\alpha})$ satisfy (5). There is a total of $N(\boldsymbol{\alpha})$ free variables in $L(\boldsymbol{\alpha})$ with $N(\boldsymbol{\alpha})$ calculated as

$$N(\boldsymbol{\alpha}) = \tfrac{1}{2}c(n+r,r)[c(n+r,r)+1] - c(n+2r,2r) \quad (7)$$

Definition 7. $L_{oc}(\boldsymbol{J}) : \mathbf{Z}_+^{1\times n} \to \mathbf{Z}^+$ is the index mapping of monomial $m(\boldsymbol{x})$ with respect to $\boldsymbol{J}$ in the full monomial base. $L_{oc}(\boldsymbol{J})$ is calculated by

$$L_{oc}(\boldsymbol{J}) = 1 + c(n+s-1, s-1) + \sum_{i=1}^{n-1} c(n-i+s-s_i-1, s-s_i-1) \quad (8)$$

where $s = \sum_{i=1}^n r_i$, $s_i = \sum_{j=1}^i r_j$ for $\boldsymbol{J} = [r_1, r_2, \cdots, r_n]$. So $L_{oc}(\boldsymbol{J})$ locates the index of $m(\boldsymbol{x})$ in the full monomial base $\boldsymbol{x}^{|r|}$.

Algorithm 1 (Polynomial decomposition). $f(\boldsymbol{x})$ is depicted in (4) and (5). Denote by $(\boldsymbol{x}^{|2r|})_i$ the ith element in full monomial base $\boldsymbol{x}^{|2r|}$. $\boldsymbol{C}_i$ is the ith element of $\boldsymbol{C}$, and P_{jk} and L_{jk} are the elements in jth row and kth column of P_f and $L(\boldsymbol{\alpha})$, respectively. $L(\boldsymbol{\alpha})$ is independent of the coefficient array $\boldsymbol{C}$ of $f(\boldsymbol{x})$, and can be generated by monomial base $\boldsymbol{x}^{|r|}$. Denote an indicator function by v_{ij}

$$\begin{cases} v_{ij} = 2, \text{ for } i = j \\ v_{ij} = 1, \text{ for } i \neq j \end{cases} \quad (9)$$

Step 1. Generate $L(\boldsymbol{\alpha})$. Set $X = \boldsymbol{x}^{|r|} \cdot (\boldsymbol{x}^{|r|})^{\mathrm{T}}$, $i = 0$, $q = 1$, and $L(\boldsymbol{\alpha}) = 0$ for initialization. Step 1 is accomplished in the following three substeps.

Substep 1.1 Set $i = i+1, j = i$. If $i > \mathrm{Dim}(L(\boldsymbol{\alpha}))$, go to Step 2; else, go to Substep 1.2.

Substep 1.2 If $L_{ij} \neq 0$, go to Substep 1.3; else, set $t = 0$, find out all the elements equal to X_{ij} in the upper triangular part of X and denote the elements set by $\{X_{kl}\}$. If $\{X_{kl}\}$ is not empty, set $L_{kl} = v_{kl}\boldsymbol{\alpha}_q$, $t = t + \boldsymbol{\alpha}_q$, and increase q by 1 for every X_{kl} in the set. Finally, set $L_{ij} = L_{ji} = -v_{ij}t$ and go to Substep 1.3.

Substep 1.3 Set $j = j+1$. If $j > \mathrm{Dim}(L)$, go to Substep 1.1; else, go to Substep 1.2.

Step 2. Construct P_f. Set $P_f = 0$ for initialization. For every coefficient $\boldsymbol{C}_i$ in $\boldsymbol{C}$, if $\boldsymbol{C}_i \neq 0$, set $\boldsymbol{J}_i = \boldsymbol{J}((\boldsymbol{x}^{|2r|})_i)$, $\boldsymbol{J}_i^{(1)} = \mathrm{floor}(\boldsymbol{J}_i/2)$, and $\boldsymbol{J}_i^{(2)} = \boldsymbol{J}_i - \boldsymbol{J}_i^{(1)}$ for pre-procedure. The decomposition of $\boldsymbol{J}_i$ is taken in three substeps.

Substep 2.1 Set $t = \mathrm{sum}(\boldsymbol{J}_i^{(2)}) - r$. If $t \leq 0$, go to Substep 2.3, else go to Substep 2.2.

Substep 2.2 Set $\boldsymbol{J}_e = \boldsymbol{J}_i^{(2)} - \boldsymbol{J}_i^{(1)}$. Denote the mth element of $\boldsymbol{J}_e$ by $\boldsymbol{J}_e(m)$, for m from 1 to n. If $\boldsymbol{J}_e(m) > 0$, set $\boldsymbol{J}_i^{(1)}(m) = \boldsymbol{J}_i^{(1)}(m) + 1$ and $t = t - 1$. If $t \leq 0$, reset $\boldsymbol{J}_i^{(2)} = \boldsymbol{J}_i - \boldsymbol{J}_i^{(1)}$ and go to Substep 2.3.

Substep 2.3 Set $k = L_{oc}(\boldsymbol{J}_i^{(1)})$ and $l = L_{oc}(\boldsymbol{J}_i^{(2)})$ which are indices of the two decomposed monomials in full monomial base $\boldsymbol{x}^{|r|}$, and finally set $P_{kl} = P_{lk} = v_{kl}\boldsymbol{C}_i/2$.

When all coefficients $\boldsymbol{C}_i$s are processed and P_f is constructed, go to Step 3.

Step 3. Set $M(f) = P_f + L(\boldsymbol{\alpha})$.

Lemma 1[9]**.** Given $f(\boldsymbol{x}) \in \mathcal{R}^1_{n,2r}$, the sufficient condition for $f(\boldsymbol{x}) \geq 0, \forall \boldsymbol{x} \in \mathbf{R}^n$ is that, $f(\boldsymbol{x})$ can be rewritten in the sum of squares form: $f(\boldsymbol{x}) = \sum_{i=1}^k f_i^2(\boldsymbol{x})$. The necessary condition also holds in following three cases:

1) $n = 2$;
2) $r = 1$;
3) $n = 3$, $r = 2$.

Theorem 1. Given $f(\boldsymbol{x}) \in \mathcal{R}^1_{n,2r}$, the following statements are equivalent:

1) $f(\boldsymbol{x})$ can be rewritten in the sum of squares form: $f(\boldsymbol{x}) = \sum_{i=1}^k f_i^2(\boldsymbol{x})$.

2) There exists $\boldsymbol{\alpha} \in \mathbf{R}^{N(\boldsymbol{\alpha})}$ such that $M(f) = P_f + L(\boldsymbol{\alpha}) \geq 0$.

Proof. Statement 1) to 2). Since $\deg(f(\boldsymbol{x})) = 2r$, and $\deg(f_i(\boldsymbol{x})) \leq r$ for any $f_i(\boldsymbol{x})$, it can be written in the linear form of $\boldsymbol{x}^{|r|}$, *i.e.*, $f_i(\boldsymbol{x}) = \boldsymbol{C}_i x^{|r|}$, $\boldsymbol{C}_i \in \mathbf{R}^{1\times c(n+r,r)}$. Thus, one gets

$$f(\boldsymbol{x}) = (\boldsymbol{x}^{|r|})^{\mathrm{T}}(\sum_{i=1}^{k}\boldsymbol{C}_i^{\mathrm{T}}\boldsymbol{C}_i)\boldsymbol{x}^{|r|}$$

By (6) and Definition 6, $\sum_{i=1}^{k}\boldsymbol{C}_i^{\mathrm{T}}\boldsymbol{C}_i \geq 0 \in S(f)$. Statement 2) is derived.

Statement 2) to 1). If statement 2) holds, from singular value decomposition, one gets $M(f) = U^{\mathrm{T}}\Lambda U$. Choose $f_i(\boldsymbol{x}) = (\Lambda^{\frac{1}{2}}U\boldsymbol{x}^{|r|})_i$. Then statement 1) is obtained. □

From Lemma 1 and Theorem 1, one can see that, the positivity validation of $f(\boldsymbol{x})$ can be relaxed to matrix inequality problem, which can be solved numerically. There is no relaxation gap in the three cases as mentioned in Lemma 1.

3 Suboptimal cost control

In this section, we give the design of cost control via polynomial decomposition approach, and the designed controller is aimed at minimum average cost performance with gain constraints.

Consider the polynomial nonlinear system with affine uncertainties described, namely,

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x},\boldsymbol{\delta},\boldsymbol{u}) = \boldsymbol{f}_0(\boldsymbol{x})+\sum_{i=1}^{l}\boldsymbol{\delta}_i\boldsymbol{f}_i(\boldsymbol{x})+\sum_{i=l+1}^{k}\boldsymbol{\delta}_i g_{i-l}\boldsymbol{u}(\boldsymbol{x}) \quad (10)$$

where $g_i \in \mathbf{R}^{n\times m}$, $\boldsymbol{f}_i(\boldsymbol{x}) \in \mathcal{R}_n^n$, and $\boldsymbol{u}(\boldsymbol{x}) \in \mathcal{R}_n^m$ with $\boldsymbol{f}(0,\boldsymbol{\delta},0) = 0$. The uncertain parameters are defined as $\boldsymbol{\delta}_i \in [\boldsymbol{\delta}_i^-,\boldsymbol{\delta}_i^+]$ with bounded time varying rates as $\dot{\boldsymbol{\delta}}_i \in [\dot{\boldsymbol{\delta}}_i^-,\dot{\boldsymbol{\delta}}_i^+]$. The set of uncertain parameters can be presented in polytopic form $\Delta(\boldsymbol{\delta})\times\Lambda(\dot{\boldsymbol{\delta}})$, where $\Delta(\boldsymbol{\delta})$ and $\Lambda(\dot{\boldsymbol{\delta}})$ are convex hull of $\boldsymbol{\delta}$ and $\dot{\boldsymbol{\delta}}$, respectively. Define $\Delta_0(\boldsymbol{\delta})$ and $\Lambda_0(\dot{\boldsymbol{\delta}})$ by

$$\Delta_0(\boldsymbol{\delta}) = \{\mathrm{col}(\boldsymbol{\delta}_1,\boldsymbol{\delta}_2,\cdots,\boldsymbol{\delta}_k)|\boldsymbol{\delta}_i \in \{\boldsymbol{\delta}_i^-,\boldsymbol{\delta}_i^+\}, i=1,2,\cdots,k\}$$

$$\Lambda_0(\dot{\boldsymbol{\delta}}) = \{\mathrm{col}(\dot{\boldsymbol{\delta}}_1,\dot{\boldsymbol{\delta}}_2,\cdots,\dot{\boldsymbol{\delta}}_k)|\dot{\boldsymbol{\delta}}_i \in \{\dot{\boldsymbol{\delta}}_i^-,\dot{\boldsymbol{\delta}}_i^+\}, i=1,2,\cdots,k\}$$

$\Delta(\boldsymbol{\delta})$ can be generated by $\Delta_0(\boldsymbol{\delta})$

$$\Delta(\boldsymbol{\delta}) = \{\boldsymbol{\delta} = \sum\nolimits_{j=1}^{2^k}\lambda_j\boldsymbol{\delta}^{(j)}|\lambda_j \geq 0, \boldsymbol{\delta}^{(j)} \in \Delta_0(\boldsymbol{\delta}),$$
$$j=1,2,\cdots,2^k, \sum\nolimits_{j=1}^{2^k}\lambda_j = 1\}$$

$\Lambda(\dot{\boldsymbol{\delta}})$ can also be generated by $\Lambda_0(\dot{\boldsymbol{\delta}})$ in the same way. We call $\Delta_0(\boldsymbol{\delta})$ and $\Lambda_0(\dot{\boldsymbol{\delta}})$ vertices sets of $\Delta(\boldsymbol{\delta})$ and $\Lambda(\dot{\boldsymbol{\delta}})$. One can see that they are with finite elements. Then, a problem with set $\Delta(\boldsymbol{\delta})\times\Lambda(\dot{\boldsymbol{\delta}})$ can be tested by $\Delta_0(\boldsymbol{\delta})\times\Lambda_0(\dot{\boldsymbol{\delta}})$ when it is convex with respect to $\boldsymbol{\delta}$ and $\dot{\boldsymbol{\delta}}$, which converts an infinite test problem into a finite one.

Theorem 2. Consider the nonlinear system described in (10). If there exists a state feedback controller $\boldsymbol{u}(\boldsymbol{x}) = [\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_m]^{\mathrm{T}} \in \mathcal{R}_{n,d}^m$ with $\boldsymbol{u}(0) = 0$ and $\boldsymbol{u}_i(\boldsymbol{x}) = \boldsymbol{K}_i E_2(\boldsymbol{x}^{|d|})$ for $i = 1,2,\cdots,m$, a parameter-dependent Lyapunov function $V(\boldsymbol{x},\boldsymbol{\delta}) = \boldsymbol{x}^{\mathrm{T}}P(\boldsymbol{\delta})\boldsymbol{x}$ with $P(\boldsymbol{\delta}) = P_0 + \sum_{i=1}^{k}\boldsymbol{\delta}_i P_i$, such that

$$P(\boldsymbol{\delta}^{(j)}) > 0,\ j = 1,2,\cdots,2^k \quad (11)$$

$$M[2\boldsymbol{x}^{\mathrm{T}}P(\boldsymbol{\delta}^{(j)})\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\delta}^{(j)},\boldsymbol{u}) + \boldsymbol{x}^{\mathrm{T}}R\boldsymbol{x} + \sum_{i=1}^{k}\dot{\boldsymbol{\delta}}_i^{(t)}\boldsymbol{x}^{\mathrm{T}}P_i\boldsymbol{x}] \leq 0,$$
$$j = 1,2,\cdots,2^k,\ \ t = 1,2,\cdots,2^k \quad (12)$$

$$M[\boldsymbol{x}^{\mathrm{T}}P_i\boldsymbol{f}_i(\boldsymbol{x})] \geq 0,\ i = 1,2,\cdots,l \quad (13)$$

$$M[\boldsymbol{x}^{\mathrm{T}}P_i g_{i-l}\boldsymbol{u}(\boldsymbol{x})] \geq 0,\ i = l+1,l+2,\cdots,k \quad (14)$$

where $\boldsymbol{\delta}^{(j)} \in \Delta_0(\boldsymbol{\delta})$, $\dot{\boldsymbol{\delta}}^{(t)} \in \Lambda_0(\dot{\boldsymbol{\delta}})$, and P_is are symmetric matrices for $i = 0,1,\cdots,k$, then, system (10) is globally stable and converges to the origin with guaranteed cost performance

$$\int_{t_0}^{\infty}\boldsymbol{x}(t)^{\mathrm{T}}R\boldsymbol{x}(t)\mathrm{d}t \leq \boldsymbol{x}(t_0)^{\mathrm{T}}P(\boldsymbol{\delta})\boldsymbol{x}(t_0) \quad (15)$$

Proof. $-P(\boldsymbol{\delta})$ is convex for $\boldsymbol{\delta}$ since it is affine with $\boldsymbol{\delta}$; thus, $-P(\boldsymbol{\delta}) < 0$ can be tested in vertices set $\Delta_0(\boldsymbol{\delta})$. Hence, from the definition of $V(\boldsymbol{x},\boldsymbol{\delta})$ and (11), we get, for any $\boldsymbol{\delta} \in \Delta(\boldsymbol{\delta})$, $V(\boldsymbol{x},\boldsymbol{\delta}) = 0$ only when $\boldsymbol{x} = 0$, and $V(\boldsymbol{x},\boldsymbol{\delta}) > 0$ with $\lim_{||\boldsymbol{x}||\to\infty} V(\boldsymbol{x}) \to \infty$ for any $\boldsymbol{x} \in \mathbf{R}^n/\{0\}$. Differentiating $V(\boldsymbol{x},\boldsymbol{\delta})$ along the system trajectory of (10), we have

$$\dot{V}(\boldsymbol{x},\boldsymbol{\delta}) = 2\boldsymbol{x}^{\mathrm{T}}P(\boldsymbol{\delta})\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\delta},\boldsymbol{u}) + \sum_{i=1}^{k}\dot{\boldsymbol{\delta}}_i\boldsymbol{x}^{\mathrm{T}}P_i\boldsymbol{x}$$

By (13) and (14), $\dot{V}(\boldsymbol{x},\boldsymbol{\delta})$ is convex for $\boldsymbol{\delta}$ and $\dot{\boldsymbol{\delta}}$; thus $\dot{V}(\boldsymbol{x},\boldsymbol{\delta}) + \boldsymbol{x}^{\mathrm{T}}R\boldsymbol{x} \leq 0$ can be tested by (12), which yields (15) when integrated from t_0 to infinity. □

In practice, we modify condition (11) to $P(\boldsymbol{\delta}^{(j)}) \geq \varepsilon$ with $\varepsilon \geq 0.001$ to avoid numerical problems. For example, for $P(\boldsymbol{\delta}) > 0$, if all eigenvalues of $P(\boldsymbol{\delta})$ are smaller than 10^{-9}, condition (12) may lose constraint effect by solvers under the tolerance of 10^{-9} and produce wrong results. Usually, the degree d in $\boldsymbol{u}(\boldsymbol{x})$ is chosen, where

$$d = \begin{cases} \deg(\boldsymbol{f}), & \text{for} \deg(\boldsymbol{f}) \text{ is odd} \\ \deg(\boldsymbol{f})+1, & \text{for} \deg(\boldsymbol{f}) \text{ is even} \end{cases} \quad (16)$$

The controller designed by Theorem 2 usually has numerous monomials in practice. Denote the number of monomials in $\boldsymbol{u}(\boldsymbol{x})$ by $N(\boldsymbol{u})$. Then, it is calculated as follows.

$$N(\boldsymbol{u}) = -m + m\frac{(n+d)!}{n!d!} \quad (17)$$

For example, $N(\boldsymbol{u}) = 19$ when $\boldsymbol{x} \in \mathbf{R}^3$ and $\boldsymbol{u} \in \mathcal{R}_{3,3}^1$, which means that there are 19 monomials in the single dimension controller of 3 states feedback when the degree is 3. Actually, some monomial terms in the controller contribute little to the stabilization performance and can be removed. However, how to find out the redundant terms is still a nonconvex problem as mentioned in [10]. In this paper, we minimize the norm 1 of the coefficients (gains) in $\boldsymbol{u}(\boldsymbol{x})$, which approximately tries to minimize the number of nonzero terms. The reason of norm 1 approximation is that, for amplitude distribution of the optimal residual, it tends to have more zeroes and very small residuals compared to norm 2 approximation solution. For optimal control, we aim at minimizing the cost function $\int_{t_0}^{\infty}\boldsymbol{x}(t)^{\mathrm{T}}R\boldsymbol{x}(t)\mathrm{d}t$, which is relaxed to minimization of the corresponding upper bound $\boldsymbol{x}(t_0)^{\mathrm{T}}P(\boldsymbol{\delta})\boldsymbol{x}(t_0)$ in our control scheme. (15) emphasizes the dependence of the performance criterion on $\boldsymbol{x}(t_0)$ and $P(\boldsymbol{\delta})$. In order to find an optimal control using the cost performance criterion, usually, it is necessary to eliminate the dependence on $\boldsymbol{x}(t_0)$. Mathematically, a simple way to deal with this problem is to average the performance obtained for a linearly independent set of initial states, *i.e.*, to assume the initial states to be random variables uniformly distributed on the surface of the n-dimensional unit sphere with $\mathrm{E}[\boldsymbol{x}(t_0)\boldsymbol{x}(t_0)^{\mathrm{T}}] = I_n$. Then, we have

$$\begin{aligned}\mathrm{E}[\boldsymbol{x}(t_0)^{\mathrm{T}}P(\boldsymbol{\delta})\boldsymbol{x}(t_0)] &= \mathrm{E}[\mathrm{tr}(P(\boldsymbol{\delta})\boldsymbol{x}(t_0)\boldsymbol{x}(t_0)^{\mathrm{T}})] = \\ &\mathrm{tr}(E[P(\boldsymbol{\delta})\boldsymbol{x}(t_0)\boldsymbol{x}(t_0)^{\mathrm{T}}]) = \\ &\mathrm{E}[\mathrm{tr}(P(\boldsymbol{\delta}))]\end{aligned}$$

Finally, minimizing cost performance is relaxed to minimizing $\frac{1}{2^k}\sum_{j=1}^{2^k}\mathrm{tr}(P(\boldsymbol{\delta}^{(j)}))$ with $\boldsymbol{\delta}^{(j)} \in \Delta_0(\boldsymbol{\delta})$. The detailed algorithm for control synthesis is presented in Algorithm 2.

Algorithm 2 (Control synthesis). We exploit PENBMI[11] with YALMIP[12] interface to solve the optimization problem derived from polynomial decomposition. The controller design algorithm is processed in the following two steps.

Step 1. Set appropriate values for ε, and $\tau_{\lim}$ which is a limit gate for monomial removals, and then, solve the following optimization problem:

$$\min_{P(\boldsymbol{\delta}),K,\boldsymbol{\alpha}} \sum_{i=1}^{m}\sum_{j=1}^{N_i} B_{ij},\ N_i = c(n+d,d)-1$$

s.t.

$$1)\quad \begin{cases} -B_{ij} \le K_{ij} \le B_{ij} & i = 1,2,\cdots,m \\ B_{ij\ge 0} & j = 1,2,\cdots,N_i \end{cases}$$

$$2)\quad (11) \sim (14) \quad \text{hold.}$$

where K_{ij}s are coefficients of monomials in $\boldsymbol{u}(\boldsymbol{x})$. Remove the corresponding monomial term when $|K_{ij}| \le \tau_{\lim}$ in the solved $\boldsymbol{u}(\boldsymbol{x})$. Construct a reduced controller with the remaining monomials and denoted it by $\tilde{\boldsymbol{u}}(\boldsymbol{x})$ with $\tilde{\boldsymbol{u}}_i(\boldsymbol{x}) = \sum_{j=1}^{l_i} K_{ij}m_{ij}(\boldsymbol{x})$ for $i = 1,2,\cdots,m$, where l_i is the number of monomials in $\tilde{\boldsymbol{u}}_i(\boldsymbol{x})$.

Step 2. Substitute $\tilde{\boldsymbol{u}}(\boldsymbol{x})$ for $\boldsymbol{u}(\boldsymbol{x})$ and solve the following optimization problem:

$$\min_{P(\boldsymbol{\delta}),K,\boldsymbol{\alpha}} \frac{1}{2^k}\sum_{j=1}^{2^k}\mathrm{tr}(P(\boldsymbol{\delta}^{(j)}))$$

s.t.

$$1)\quad \underline{T}_{ij} \le K_{ij} \le \bar{T}_{ij},\ i = 1,\cdots,m,\ j = 1,\cdots,l_i$$

$$2)\quad (11) \sim (14) hold.$$

where $\underline{T}_{ij}$s and $\bar{T}_{ij}$s are the lower and upper bounds constraints for K_{ij}s, respectively, which are given in specific systems to meet physical limitations or other restrictions.

The cost performance in (15) cannot be minimized directly. Our scheme is to minimize the average upper bound cost. In this sense, the solved $\tilde{\boldsymbol{u}}^*(\boldsymbol{x})$ in Algorithm 2 is a suboptimal solution.

4 Numerical examples

Example 1. Consider the polynomial described system

$$\begin{cases} \dot{x}_1 = x_2 - \delta_1 x_1^3 + \delta_2 x_1^2 \\ \dot{x}_2 = \delta_3 u \end{cases} \tag{18}$$

where $\delta_1 \in [0.5, 1.5]$ with $\dot{\delta}_1 \in [-0.5, 0.5]$, $\delta_2 \in [1.0, 2.0]$ with $\dot{\delta}_2 \in [-0.5, 0.5]$, and $\delta_3 \in [0.9, 1.1]$ with $\dot{\delta}_3 \in [-0.01, 0.01]$. We pick up $u(\boldsymbol{x}) = \boldsymbol{K}E_2(\boldsymbol{x}^{|3|})$, which has the same order as $\boldsymbol{f}(\boldsymbol{x})$. Choose $R = I_2$. Then the cost function is l_2 norm of $\boldsymbol{x}(t)$. Setting $\varepsilon = 0.01$ and $\tau_{\lim} = 0.1$, and using Algorithm 2, we get $u(\boldsymbol{x})$ and $\tilde{u}(\boldsymbol{x})$ in Step 1 as follows

$$\begin{aligned}u(\boldsymbol{x}) = &-1.451x_1 - 2.4876x_2 - 2.865x_1^2 - 0.014318x_2^2 + \\ &1.3305\times 10^{-16}x_1x_2 - 0.18563x_1^3 - 0.0539x_2^3 + \\ &6.8569\times 10^{-15}x_1x_2^2 - 1.6481\times 10^{-16}x_1^2x_2\end{aligned}$$

$$\tilde{u}(\boldsymbol{x}) = k_1x_1 + k_2x_2 + k_3x_1^2 + k_4x_1^3$$

If the control gains are constrained as $-20 \le k_i \le 20$, for $i = 1,2,3,4$, *i.e.*, $\underline{\boldsymbol{T}} = -20$ and $\bar{\boldsymbol{T}} = 20$, then, we get the results as follows

$$P_0 = \begin{bmatrix} 6.8267 & 0.3179 \\ 0.3179 & 0.0669 \end{bmatrix},\ P_1 = \begin{bmatrix} -3.0034 & -0.1156 \\ -0.1156 & 0.0156 \end{bmatrix}$$

$$P_2 = \begin{bmatrix} 0.0000 & -0.0000 \\ -0.0000 & -0.0051 \end{bmatrix},\ P_3 = \begin{bmatrix} -0.0071 & 0.0000 \\ 0.0000 & 0.0000 \end{bmatrix}$$

$$V(\boldsymbol{x},\boldsymbol{\delta}) = \boldsymbol{x}^{\mathrm{T}}(P_0 + \sum_{i=1}^{3}\delta_iP_i)\boldsymbol{x}$$

$$u^*(\boldsymbol{x}) = -20x_1 - 14.2099x_2 - 20x_1^2 + 1.9549x_1^3$$

$$\frac{1}{8}\sum_{j=1}^{8}\mathrm{tr}(P(\boldsymbol{\delta}^{(j)})) = 3.8981$$

Pick up $\delta_1 = 1+0.5\sin(t)$, $\delta_2 = 1.5+0.5\cos(t)$, and $\delta_3 = 1+0.1\sin(0.1t)$ in our numerical simulation. The phase portrait of the closed loop system with $\tilde{u}^*(\boldsymbol{x})$ is illustrated in Fig. 1. The sizes of arrow lines in Fig.1 are proportional to $||\dot{\boldsymbol{x}}(t)||_2^2$, which signify the convergence rate of $\boldsymbol{x}(t)$. All trajectories converge to the origin as shown in the figure. Choose initial states on the unit circle plane, *e.g.*, $\boldsymbol{x}(t_0) = [\sin(\frac{k}{8}\pi), \cos(\frac{k}{8}\pi)]$, for $k = 0,1,\cdots,15$. The evolutions of the closed loop system are shown in Fig. 2.

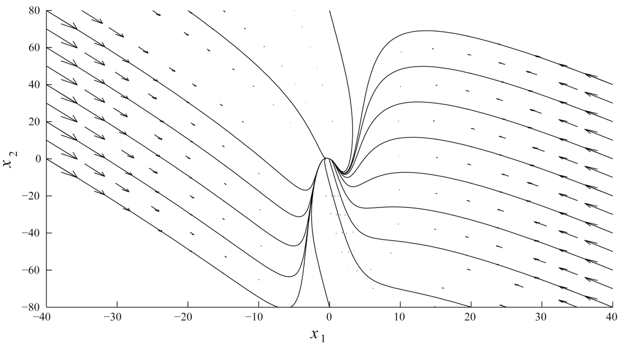

Fig. 1 Phase portrait of the closed loop system in Example 1

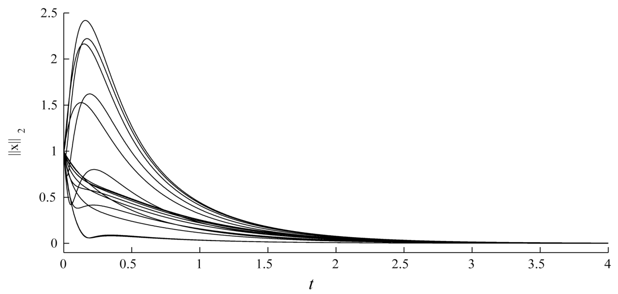

Fig. 2 Evolutions of the closed loop system with initial states on the unit circle plane in Example 1

Example 2. Consider the following nonlinear system

$$\begin{cases} \dot{x}_1 = 2\sin x_2 \\ \dot{x}_2 = x_1^2 + u \end{cases} \tag{19}$$

We get 2-order Taylor series approximation to sin x_2 at zero as follows

$$\sin x_2 = x_2 + R_2(x_2)$$

where $R_2(x_2)$ is the residual series of $\sin x_2$ with an order greater than 2. From Lagrangian mean-value theorem, $R_2(x_2) = \frac{\sin^{(3)} x_2}{3!}|_{x_2=\xi} \cdot x_2^3 = -\frac{1}{6}\sin\xi \cdot x_2^3$, with $0 < \xi < x_2$. Take $\xi = \theta x_2$, for $0 < \theta < 1$, and choose $\delta = \sin(\theta x_2)$. Then system (19) can be approximated as

$$\left\{\begin{array}{l} \dot{x}_1 = 2x_2 - \frac{1}{3}\delta x_2^3 \\ \dot{x}_2 = x_1^2 + u \end{array}\right. \tag{20}$$

with $\delta \in [-1, 1]$ and $\dot{\delta} \in (-1, 1)$. Choose $R = I_2$, $\deg(u) = 3$, $\varepsilon = 0.01$, $\tau_{\lim} = 0.001$, $\underline{\boldsymbol{T}} = -20$, and $\bar{\boldsymbol{T}} = 20$, applying Algorithm 2, then we get the suboptimal cost controller

$$u^*(\boldsymbol{x}) = -8.9918x_1 - 20x_2 - x_1^2 - 0.051532x_1^2x_2 - 20x_2^3 \tag{21}$$

The phase portrait and evolutions of closed loop system (19) are illustrated in Figs 3 and 4, respectively.

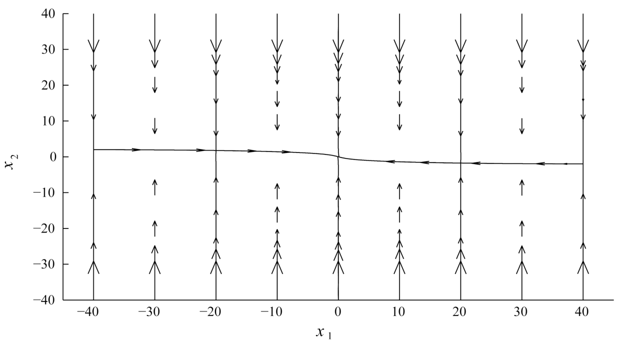

Fig. 3 Phase portrait of the closed loop system (19) in Example 2

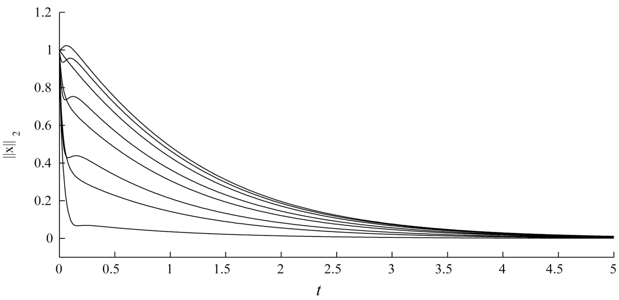

Fig. 4 Evolutions of the closed loop system (19) with initial states on the unit circle plane in Example 2

5 Conclusion

This paper presents an algorithm for polynomial decomposition, which can efficiently check the nonnegativity of polynomials with high orders. The proposed decomposition method is exploited for control synthesis of polynomial nonlinear systems with constrained controller gains. The state feedback control law underlying cost performance with minimum nonzero monomial terms is obtained via optimizing on minimum norm 1 of controller gains and subsequently optimizing on minimal average trace of $P(\boldsymbol{\delta})$ at the vertices of $\Delta(\boldsymbol{\delta})$. Numerical examples show that the proposed control scheme exhibits effective performance for polynomial nonlinear systems with affine uncertain parameters.

References

1 Parrilo P A. Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization [Ph. D. dissertation], California Institute of Technology, USA 2000

2 Parrilo P A. Nonlinear stability via sum of squares programming. In: Proceedings of the 2004 American Control Conference. Boston, USA: IEEE, 2004. 3272

3 Papachristodoulou A. Analysis of nonlinear time-delay systems using the sum of squares decomposition. In: Proceedings of the 2004 American Control Conference. Boston, USA: IEEE, 2004. 4153∼4158

4 Papachristodoulou A, Prajna S. A tutorial on sum of squares techniques for systems analysis. In: Proceedings of the 2005 American Control Conference. Portland, USA: IEEE, 2005. 2686∼2700

5 Sturm J F. A matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 1999, **11**(1-4): 625∼653

6 Jarvis-Wloszek Z W. Lyapunov Based Analysis and Controller Synthesis for Polynomial Systems Using Sum-of-Squares Optimization [Ph. D. dissertation], University of California, 2003

7 Rantzer A. A dual to Lyapunov′s stability theorem. *Systems and Control Letters*, 2001, **42**: 161∼168

8 Prajna S, Parrilo P A, Rantzer A. Nonlinear control synthesis by convex optimization. *IEEE Transactions on Automatic Control*, 2004, **49**(2): 310∼314

9 Parrilo P A. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming Series B*, 2003, **96**(2): 293∼320

10 Hassibi A, How J P, Boyd S. Low-authority controller design via convex optimization. *Journal of Guidance, Control and Dynamics*, 1999, **22**(6): 862∼872

11 Henrion D, Lofberg J, Kocvara M, Stingl M. Solving molynomial static output feedback problems with penbmi. In: Proceedings of IEEE Conference on Decision and Control. USA: IEEE, 2005. 7581∼7586

12 Lofberg J. YALMIP : a toolbox for modeling and optimization in MATLAB. In: Proceedings of International Symposium on Computer Aided Control System Design. Taiwan, China: IEEE, 2004. 284∼289

TONG Chang-Fei Ph. D. candidate at Institute of Industrial Process Control of Zhejiang University. His research interest covers robust control and nonlinear control. Corresponding author of this paper.
E-mail: cftong@iipc.zju.edu.cn

ZHANG Hui Received his Ph. D. degree from Zhejiang University in 2004. He is now an associate professor, Zhejiang University. His research interest covers stochastic systems and information theoretical approaches to control systems.
E-mail: zhanghui@iipc.zju.edu.cn

SUN You-Xian Professor in Department of Control Science and Engineering at Zhejiang University, the academician of Chinese Academy of Engineering. His research interest covers process control, robust control theory and its application, fault tolerance control theory and its application, and modelling and computer control of pulp and paper-making processes.