

**AN IMPROVED METHOD FOR ESTIMATING THE
DOMAIN OF ATTRACTION OF UNCERTAIN
RATIONAL NONLINEAR SYSTEMS BY
USING LMI STABILITY CONDITIONS**

Péter Polcz¹, Gábor Szederkényi^{1,2}, and Tamás Péni²

¹Faculty of Information Technology and Bionics, Pázmány Péter Catholic
University, Práter u. 50/a, H-1083 Budapest, Hungary

²Systems and Control Laboratory, Institute for Computer Science and
Control (MTA SZTAKI), Hungarian Academy of Sciences, Kende u. 13-17,
H-1111 Budapest, Hungary

JLR - 4/2015

Abstract

An optimization based method is proposed in this report for the computation of Lyapunov functions and regions of attractions for uncertain nonlinear systems containing polynomial and rational terms. The Lyapunov function is given in a special quadratic form, and the negativity of its derivative is ensured using appropriate LMI conditions. The starting point of the method is an approach taken from the literature, in which the conservatism of the solution is reduced by utilizing Finsler's lemma. The improvements and new contributions can be summarized as follows: 1) The model transformation to the required form for optimization is done automatically. At the same time, the number of monomials and rational terms in the computational problem is kept as low as possible using linear fractional transformation (LFT) and further automatic model simplification steps. This technique results in the dimension reduction of the problem compared to known solutions in the literature. 2) An algorithm was given for the generation of appropriate annihilators, which appear in Finsler's lemma. 3) An improved method was proposed for determining the possible largest invariant set for the dynamics using the computed Lyapunov function. 4) In comparison with the technique taken from the literature, a generalized and simplified formula was given. The operation of the method is illustrated on a few examples taken from the literature.

Contents

1	Introduction	2
2	Basic notions, known results	3
2.1	System class, Lyapunov functions, domain of attraction	3
2.2	Further notations, abbreviations	4
2.3	Sufficient LMI conditions for stability using the Finsler's lemma and the notion of annihilators	4
2.4	Dynamical system representation	6
2.5	Linear fractional transformations	7
3	Estimating the domain of attraction	8
3.1	Handling non-basis functions	8
3.2	Eliminating redundant terms	9
3.3	Annihilator generation	10
3.4	Finding an appropriate Lyapunov function	10
3.5	Finding the maximal level set for a given polytope	12
3.6	Finding the most appropriate outer polytope	14
4	Illustrative examples and results	15
4.1	Van der Pol dynamics	15
4.2	Uncertain Van der Pol dynamics	17
4.3	Third order rational system	18
4.4	A continuous fermentation process	19
4.4.1	Open-loop system (OLS)	21
4.4.2	Effect of linear substrate feedback	23
5	Conclusions	25

1. Introduction

Approximating the domain of attraction is often a fundamental task in model analysis and controller design/evaluation. The stability properties of dynamical systems are most often studied using Lyapunov functions, therefore, an extensive literature exists on the computational construction of Lyapunov functions [9]. A fundamental result in this field is the existence of so-called maximal Lyapunov functions for a wide class of nonlinear systems and the corresponding (computationally demanding) iterative procedure to approximate them [10]. An algorithm for generating Lyapunov functions for a special class of nonlinear systems based on the construction of polytopes is given in [11]. In [12], a linear programming based method is given for the construction of Lyapunov functions for general planar nonlinear systems. In [13], maximal Lyapunov functions were defined and computed for hybrid (piecewise nonlinear) systems.

At the same time, the use of linear matrix inequalities (LMI) and semidefinite programming (SDP) techniques for nonlinear systems has become very popular due to their advantageous properties and the availability of efficient numerical tools to solve LMI problems. These new techniques provide a powerful framework for stability analysis and robust control and filtering problems. Ghaoui et.al. [14] used quadratic Lyapunov functions and linear fractional transformations (LFT) to represent a rational nonlinear system and defined convex conditions for stability analysis and state feedback design. Using sum of squares (SOS) programming to maximize the estimate of the region of attraction can be found in [9, 15]. Stability conditions in both references are converted into LMIs using SOS relaxations and the generalization of the S-procedure. Topcu et.al. [16] utilized a further branch-and-bound type refinement in the parameter space to reduce the solution's conservatism.

A recent important result from this line of research is published in [1], where the authors use Finsler's lemma and the notion of annihilators to compute rational Lyapunov functions for a wide class of locally asymptotically stable nonlinear systems. The newly introduced sufficient conditions for the stability are affine parameter dependent LMIs, because they are characterized by affine functions of the state (x) and uncertain parameters (δ). Affine parameter dependent LMIs can be computationally handled by checking their feasibility at the corner points (vertices) of a polytopic region, on which the state and uncertain parameters are defined. In [1], it is shown that with some additional conservatism, the use of the vertices could be avoided by modifying the LMIs with the S-Procedure.

This work is based on the results of [1], and introduces an improved method to generate sufficient linear matrix inequality (LMI) conditions for local and regional asymptotic stability of polynomial and rational nonlinear systems. These LMI conditions are given through a Lyapunov function containing monomial and rational terms with the prescribed properties. In case of uncertain systems, the (possibly time-dependent) uncertain parameters will also appear in the Lyapunov function. In the last chapter, computational results are presented on illustrative examples taken from the literature.

2. Basic notions, known results

In this section, we present the basic notions, notations and known results on which our computational results are based.

2.1 System class, Lyapunov functions, domain of attraction

We consider nonlinear systems of the form

$$\dot{x} = f(x, \delta), \quad x \in \mathbb{R}^n, \quad x_0 \in \mathcal{X}, \quad \delta \in \mathcal{D}, \quad \dot{\delta} \in \check{\mathcal{D}}, \quad (2.1)$$

where x is the state vector, x_0 is the initial condition, and δ is the vector of (possibly time-dependent) uncertain parameters. Furthermore, it is assumed that $\mathcal{X}$, $\mathcal{D}$, and $\check{\mathcal{D}}$ are known polytopic regions. We assume that $x^* = 0 \in \mathbb{R}^n$ is a locally asymptotically stable equilibrium point of (2.1) for all $\delta \in \mathcal{D}$. The set of all initial conditions, from which the solutions converge to x^* is called the *domain of attraction* (DOA). We search for Lyapunov functions in the form

$$V(x, \delta) = x^T P(x, \delta) x, \quad v_l(x) \leq V(x, \delta) \leq v_u(x) \quad (2.2)$$

with the property

$$\dot{V}(x, \delta, \dot{\delta}) \leq -v_d(x), \quad \forall (x, \delta, \dot{\delta}) \in \mathcal{X} \times \mathcal{D} \times \check{\mathcal{D}}, \quad (2.3)$$

where P is a positive definite symmetric matrix function, and v_l , v_u and v_d are continuous positive functions on $\mathcal{X}$. Clearly, if (2.3) is fulfilled, then any closed level set of V completely inside $\mathcal{X}$ bounds an invariant region of the state space that is part of the domain of attraction.

The measure of *conservatism* is a property, which describes an invariant region. Let us consider two domains ε and γ bounded by two different level sets. We call ε less *conservative* than γ if ε is a better estimate of the actual region of attraction in the sense that the area/volume of ε is larger than that of γ .

Roughly speaking, our main objective is to find a Lyapunov function $V(x)$ having a level set, which bounds the least conservative invariant region. In fact, introducing higher degree monomials into $V(x)$ generally results in better estimates, although a small increment in the number of monomials generates a huge increase in the dimension of the problem. Therefore, the rapidly growing computational burden must be taken into consideration through considering the possibilities of dimension reduction.

2.2 Further notations, abbreviations

In this report, we will use the following additional notations:

- $\text{Co}(\mathcal{X})$ denotes the convex expansion of the set $\mathcal{X}$, more specifically, $\text{Co}(v_i, i = 1 \dots n)$ is the convex hull of the set of vertices $\{v_i \mid i = \overline{1, n}\}$.
- $\vartheta(\mathcal{X})$ denotes the set of all vertices (corner points) of the convex polytope $\mathcal{X}$ (i.e. $\text{Co}(\vartheta(\mathcal{X})) = \mathcal{X}$).
- $\nabla V(x)$ is the gradient of the scalar function $V(x)$.
- $0_{n \times m}$ and I_n denote the $n \times m$ zero matrix and $n \times n$ unit matrix, respectively.
- A *basis function* is an element of a particular basis for a function space. In this report, monomials $(x_1^{p_1} x_2^{p_2} \dots x_k^{p_k})$ and rational functions with monomial numerator will be considered as basis functions.
- open-loop system (OLS)
- closed-loop system (CLS)
- Kronecker tensor product: $A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1m}B \\ \vdots & \ddots & \vdots \\ a_{n1}B & \dots & a_{nm}B \end{bmatrix} = \text{kron}(A, B)$
- $[a_1, \dots, a_n]_{\text{col}} = [a_i]_{\text{col}}^{i=\overline{1, n}} = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}$
- $[a_1, \dots, a_n]_{\text{row}} = [a_i]_{\text{row}}^{i=\overline{1, n}} = \begin{bmatrix} a_1 & \dots & a_n \end{bmatrix}$
- $\text{row}_i(A)$ denotes the i th row of A
- $\text{col}_i(A)$ denotes the i th column of A

2.3 Sufficient LMI conditions for stability using the Finsler's lemma and the notion of annihilators

In the case of a linear time invariant (LTI) dynamical system $\dot{x} = Ax$, the necessary and sufficient Lyapunov conditions (2.2, 2.3) for the system's global stability can be easily converted into their *equivalent* LMI form: $P > 0$ (P is positive definite), and $PA + A^T P < 0$.

In contrary, when having an $\dot{x} = \mathcal{A}(\delta)x$ uncertain linear system with an affine $\mathcal{A}(\delta)$ state transition matrix function, the Lyapunov inequality will become an affine parameter (δ) dependent LMI. According to Proposition 5.4 in [4], if we can specify a bounded polytopic domain, in which the uncertain parameter $\delta \in \mathcal{D}$ operates then, it is enough to check the feasibility of the inequality in the corner points of the $\mathcal{D}$ polytope: $P\mathcal{A}(\delta) + \mathcal{A}(\delta)^T P < 0$, $\forall \delta \in \vartheta(\mathcal{D})$. Assuming a constant matrix P in (2.2), this system of LMIs is still *equivalent* with (2.3).

In the case of a nonlinear system of the form $\dot{x} = \mathcal{A}(x)x$ or having a Lyapunov function candidate of the form $V(x) = \pi^T P \pi$, with $\pi = \pi(x)$ column vector of rational terms in x ,

equivalent LMI conditions cannot be formulated. Therefore, the Finsler lemma is used in [1] to reduce the conservativeness of the sufficient LMI conditions.

Lemma 2.3.1 (Finsler's lemma) *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be given a polytopic set, $P : \mathcal{X} \mapsto \mathbb{R}^{p \times p}$, $N : \mathcal{X} \mapsto \mathbb{R}^{r \times p}$ be given matrix functions, with P symmetric. Let $Q(x)$ be a basis for the null space of $N(x)$. Then, the following are equivalent:*

- (i) $\pi^T P(x) \pi > 0, \quad \forall x \in \mathcal{X}, \quad \forall \pi \in \mathbb{R}^p \text{ s.t. } N(x)\pi = 0$
- (ii) $\exists L : \mathcal{X} \mapsto \mathbb{R}^{p \times r} \text{ s.t. } P(x) + L(x)N(x) + N^T(x)L^T(x) > 0, \quad \forall x \in \mathcal{X}$
- (iii) $Q^T(x)P(x)Q(x) > 0, \quad \forall x \in \mathcal{X}.$

Stating that $P(x)$ and $N(x)$ are affine functions and L is constant matrix, we obtain a special case of the Finsler's lemma. Henceforth, the conditions (i) and (ii) are no longer equivalent, but (ii) shall continue to be a sufficient condition for (i), which is a satisfactory result in our case. Furthermore, (ii) will become a polytopic LMI (i.e. an affine parameter dependent LMI) of the following form

$$\exists L \in \mathbb{R}^{p \times r} : P(x) + LN(x) + N^T(x)L^T > 0, \quad \forall x \in \vartheta(\mathcal{X}) \quad (2.4)$$

which can be transformed into a single parameter independent LMI with a higher dimension. We have to design an $N(x)$ affine matrix function in such a way that $N(x)\pi = 0$ for all π . Therefore, $N(x)$ is called an *annihilator* of π . For example:

$$\pi = \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \\ x_1 x_2 \end{bmatrix}, \quad N(x) = \begin{bmatrix} x_2 & -x_1 & 0 & 0 \\ x_1 & 0 & -1 & 0 \\ x_2 & 0 & 0 & -1 \\ 0 & x_1 & 0 & -1 \\ 0 & 0 & x_2 & -x_1 \end{bmatrix}, \quad N(x)\pi = 0, \forall x \in \mathcal{X} \quad (2.5)$$

It is obvious that (2.4) is a less conservative sufficient condition for $\pi^T P \pi > 0$, than $P > 0$. In order to demonstrate the effect of the annihilator, consider the following example:

$$\mathcal{X} = [-1, 1] \times [-1, 1], \quad P = \begin{bmatrix} 1.8 & 0 & 0 & 0 \\ 0 & 1.2 & 0 & 0 \\ 0 & 0 & -0.5 & 0 \\ 0 & 0 & 0 & 1.2 \end{bmatrix}, \quad L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -0.8 & 0 & 0 & 0 \\ 0 & 0 & -0.1 & 0 & 0 \end{bmatrix}$$

P is definitely not a positive definite matrix because it has a negative eigen value (-0.5), however, $P + LN(x) + N^T(x)L^T > 0$ for every $x \in \vartheta(\mathcal{X}) = \{(1, 1), (-1, 1), (-1, -1), (1, -1)\}$. In general, increasing the size of the annihilator introduces more degrees of freedom in the computational problem.

2.4 Dynamical system representation

As a starting point, we use the same differential-algebraic representation of nonlinear models that was introduced in [1], namely:

$$\begin{aligned} \dot{x} &= f(x, \delta) = Ax + B\pi & x_0 &\in \mathcal{X} \\ 0 &= G(x, \delta)x + F(x, \delta)\pi & \delta &\in \mathcal{D}, \dot{\delta} \in \check{\mathcal{D}} \end{aligned} \quad (2.6)$$

$$0 = \aleph_{\pi_b}(x, \delta)\pi_b \quad \text{where } \pi_b = \begin{bmatrix} x \\ \pi \end{bmatrix} \in \mathbb{R}^m, \quad m = n + p \quad (2.7)$$

1. $x \in \mathbb{R}^n$ is the state vector, $\delta \in \mathbb{R}^d$ is column vector of uncertain model parameters.
2. $f(x, \delta)$ is a rational function well defined on $\mathcal{X} \times \mathcal{D}$, and $f(0, \delta) = 0$ for all $\delta \in \mathcal{D}$.
3. $\pi = \pi(x, \delta) \in \mathbb{R}^p$ is a column vector of nonlinear monomials in (x, δ) or rational terms with monomial numerators. π_b is introduced as a composition of x and δ , and contains the all possible basis functions appearing in $f(x, \delta)$.
4. $G : \mathcal{X} \times \mathcal{D} \mapsto \mathbb{R}^{p \times n}$ and $F : \mathcal{X} \times \mathcal{D} \mapsto \mathbb{R}^{p \times p}$ are affine matrix functions of (x, δ) , with assumption that $F(x, \delta)$ is invertible for all $(x, \delta) \in \mathcal{X} \times \mathcal{D}$. The second equation of (2.6) introduces a constraint representing the relation between x and π , namely: $\pi = -F(x, \delta)^{-1}G(x, \delta)x$.
5. Although $C_b(x, \delta) = [G(x, \delta) \ F(x, \delta)]$ is an affine annihilator of π_b , this will not be maximal. In order to exploit the possible degrees of freedom, an additional $\aleph_{\pi_b}(x, \delta)$ annihilator is introduced, which will represent the extra relations between x and π .

With the intention of simplifying the further computations, we propose to merge the two annihilators into a single one:

$$\mathcal{N}_{\pi_b}(x, \delta) = \begin{bmatrix} C_b(x, \delta) \\ \aleph_{\pi_b}(x, \delta) \end{bmatrix} \in \mathbb{R}^{q \times m} \quad (2.8)$$

Therefore, the representation (2.6, 2.7) can be written as

$$\begin{aligned} \dot{x} &= Ax + B\pi & x_0 &\in \mathcal{X} \\ 0 &= \mathcal{N}_{\pi_b}(x, \delta)\pi_b & \delta &\in \mathcal{D}, \dot{\delta} \in \check{\mathcal{D}} \end{aligned} \quad (2.9)$$

This form separates the linear part of the system (x) from its nonlinear part (π). In [1], the authors propose that π contains all basic monomials of degree less than or equal to the maximal degree term in the system equation. This clearly causes a combinatorial explosion as the number of variables and their degrees increase. Therefore, we propose the application of LFT to decrease the number of monomials in π , hoping that the solutions conservatism will not increase significantly.

2.5 Linear fractional transformation (LFT)

Using the LFT, the linear and nonlinear part of any rational nonlinear uncertain system having the form $\dot{x} = \mathcal{A}(x, \delta)x$ can be separated in the following manner:

$$\begin{array}{c} \begin{array}{c} \xleftarrow{\dot{x}} \begin{array}{|c|c|} \hline M_{11} & M_{12} \\ \hline M_{21} & M_{22} \\ \hline \end{array} \xleftarrow{x} \\ \downarrow z \quad \quad \quad \downarrow w \\ \Delta(x, \delta) \end{array} \end{array} \quad \begin{array}{l} \begin{bmatrix} \dot{x} \\ z \end{bmatrix} = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \cdot \begin{bmatrix} x \\ w \end{bmatrix} \longleftrightarrow \begin{array}{l} \dot{x} = M_{11}x + M_{12}w \\ z = M_{21}x + M_{22}w \end{array} \end{array} \quad (2.10)$$

$$w = \Delta(x, \delta) z \quad (2.11)$$

where M_{ij} are constant matrices and $\Delta(x, \delta)$ is a diagonal matrix containing monomials of the state variables and uncertain parameters.

Equation (2.10) can be considered as a linear time invariant (LTI) system with a nonlinear input defined by the nonlinear uncertain dynamics (2.11). Eliminating z and w from (2.10) and (2.11) and by using an auxiliary step, we get the following:

$$\mathcal{A}(x, \delta)x = \underbrace{M_{11}x}_{\text{linear part}} + \underbrace{M_{12}(I_p - \Delta(x, \delta)M_{22})^{-1}\Delta(x, \delta)M_{21}x}_{\text{nonlinear part}}, \quad (2.12)$$

The auxiliary step constitutes the following identity (the arguments in Δ will be suppressed):

$$\Delta(I_p - M_{11}\Delta)^{-1}M_{21} = (I_p - \Delta M_{11})^{-1}\Delta M_{21} \quad (2.13)$$

This identity can be easily seen, if I_p is substituted by $\Delta^{-1}\Delta$, than the left hand side of (2.13) will be:

$$\Delta((\Delta^{-1} - M_{11})\Delta)^{-1}M_{21} = (\Delta^{-1} - M_{11})^{-1}M_{21} = (\Delta^{-1} - M_{11})^{-1}I_p M_{21} \quad (2.14)$$

The same idea is applied on the newly introduces I_p :

$$(\Delta^{-1} - M_{11})^{-1}\Delta^{-1}\Delta M_{21} = (\Delta(\Delta^{-1} - M_{11}))^{-1}\Delta M_{21} = (I_p - \Delta M_{11})^{-1}\Delta M_{21} \quad (2.15)$$

Note that the first equation of (2.10) resembles the first equation of (2.6). In the next chapter, we will see that by choosing $\pi = w$, $A = M_{11}$, $B = M_{12}$, we will get the same representation as in (2.6). Furthermore, composing an annihilator matrix for $\pi_b = [x, \pi]_{\text{col}}$ will also be a quite straightforward operation.

3. Estimating the domain of attraction

Due to the fact that the linear fractional representation (LFR) in equations (2.10, 2.11) is a special case of (2.6), it can be easily converted into that form by introducing the following notations:

$$\begin{aligned} A &= M_{11} & G(x, \delta) &= -\Delta(x, \delta)M_{21} & \in \mathbb{R}^{p \times n} \\ B &= M_{12} & F(x, \delta) &= I_p - \Delta(x, \delta)M_{22} & \in \mathbb{R}^{p \times p} \\ \pi &= -F(x, \delta)^{-1}G(x, \delta)x & & \in \mathbb{R}^p \end{aligned} \quad (3.1)$$

To repeat, the matrix $C_b(x, \delta) = [G(x, \delta) \ F(x, \delta)]$ is an annihilator of $\pi_b = [x, \pi]_{\text{col}}$.

By using the LFT, the following issues emerge that need to be handled:

1. The π in equation (3.1), may contain non-basis functions.
2. The π may contain redundant terms with (possibly) different scalar multiplier.
3. The annihilator $\aleph_{\pi_b}(x, \delta)$ in equation (2.7) shall be determined automatically.

In the next three sections, we propose a possible resolution for each issue.

3.1 Handling non-basis functions

Considering the properties of the LFR, we can assume that π contains only rational terms, therefore, the task is to split up the elements with a polynomial numerator. As a consequence, the size of π will increase, which makes it necessary to modify B and $F(x)$ appropriately to satisfy the model equations (2.6):

$$\underbrace{\begin{bmatrix} f(x, \delta) \end{bmatrix}}_{F(x, \delta)} \underbrace{\begin{bmatrix} \frac{\alpha p(x, \delta) + \beta q(x, \delta)}{r(x, \delta)} \end{bmatrix}}_{\pi} = \begin{bmatrix} \alpha f(x, \delta) & \beta f(x, \delta) \end{bmatrix} \begin{bmatrix} \frac{p(x, \delta)}{r(x, \delta)} \\ \frac{q(x, \delta)}{r(x, \delta)} \end{bmatrix}, \quad (3.2)$$

where $p(x, \delta)$ and $q(x, \delta)$ are monomials, $f(x, \delta)$ is an affine function and $r(x, \delta)$ is an arbitrary polynomial in x and δ (it can be 1 as well). The same step should be adopted on the matrix B . This transform generally increases the repetitive elements in π , but the next step will resolve this problem.

3.2 Eliminating redundant terms

As mentioned previously, the same basis function can appear several times in π , therefore, it is computationally advantageous to eliminate them. Conversely, the corresponding columns in B and $F(x, \delta)$ shall be merged in the following way:

$$\underbrace{\begin{bmatrix} a(x, \delta) & b(x, \delta) \end{bmatrix}}_{B \text{ or } F(x, \delta)} \underbrace{\begin{bmatrix} \alpha p(x, \delta) \\ \beta p(x, \delta) \end{bmatrix}}_{\pi} = \begin{bmatrix} \alpha a(x, \delta) + \beta b(x, \delta) \end{bmatrix} \begin{bmatrix} p(x, \delta) \end{bmatrix} \quad (3.3)$$

Even though [1] assumed $F(x, \delta) \in \mathbb{R}^{p \times p}$ to be an invertible square matrix (3.1), actually all multiplications by the inverse in the computations can be avoided. Accordingly, we have the possibility to ease this strict condition on $F(x, \delta)$. In the next section, we will give a more general formula for the LMIs, in which we assume that $F(x, \delta) \in \mathbb{R}^{q \times p}$ is a general rectangular matrix, with p being the number of terms in π . This relaxation does not affect the size of the LMIs, they still remain semidefinite problems. At the same time, the last step (reduction in the number of terms in π) can produce identical rows in $C_b(x, \delta) = \begin{bmatrix} G(x, \delta) & F(x, \delta) \end{bmatrix}$ (possibly multiplied by a constant). In order to decrease the number of symbolic decision parameters in the optimization problem, it is advisable to remove these redundant rows from $C_b(x, \delta)$. The following pseudocode shows the entire procedure:

```

Data:
     $\pi$ , indices( $\pi$ ) = (first)  $1 \dots p$  (last), having repetitive monomials
     $B$ ,  $F(x)$ ,  $G(x)$ : system matrices
Result:
     $\pi$  without repetitive monomials,
    modified  $B$ ,  $F(x)$ ,  $G(x)$  model matrices corresponding to the new  $\pi$ 
for  $j = p$  down to 2 do
    for  $i = 1$  to  $j-1$  do
         $\frac{\alpha p(x)}{r_i(x)} \leftarrow \pi[i]$ ,  $\frac{\beta q(x)}{r_j(x)} \leftarrow \pi[j]$ 
        if  $p(x) = q(x)$  and  $r_i(x) = r_j(x)$  then
            Generate the new column in  $B$  and  $F(x)$  (element-wise):
             $B[\text{column } i] = \alpha B[\text{column } i] + \beta B[\text{column } j]$ 
             $F(x)[\text{column } i] = \alpha F(x)[\text{column } i] + \beta F(x)[\text{column } j]$ 
             $\pi[i] = p(x)/r_i(x)$ 
            Remove column  $j$  from  $B$  and  $F(x)$ .
            Remove the  $j$ th element from  $\pi$ 
        end
    end
end
 $p \leftarrow \text{size}(\pi)$ 
 $C_b(x) \leftarrow \begin{bmatrix} G(x) & F(x) \end{bmatrix}$ 
Erase redundant rows from  $C_b(x)$ 

```

3.3 Annihilator generation

Although $C_b(x, \delta)$ is a true annihilator of π_b representing the dependence of π upon x , it shall be not maximal, therefore, finding a second $\aleph_{\pi_b}(x, \delta)$ annihilator with rows not appearing in $C_b(x, \delta)$ is essential. $C_b(x, \delta)$ and $\aleph_{\pi_b}(x, \delta)$ will form together the annihilator $\aleph_{\pi_b}(x, \delta)$ of (2.9). Using Matlab's symbolic toolbox, we have written an algorithm, which generates an affine annihilator $\aleph_{\pi_b}(x, \delta)$ having a special form.

If π contains only monomials (no rational terms), then it is enough if in each row of the matrix there appear only two nonzero items because, when an element $a(x, \delta)$ of π_b can be eliminated by two other elements $b(x, \delta), c(x, \delta)$, than $a(x, \delta)$ surely can be eliminated by using only one from $b(x, \delta)$ and $c(x, \delta)$. This statement is no longer valid when having monomials and basis rational terms too. Consider the following simple example:

$$\begin{bmatrix} -1 & x & x \end{bmatrix} \begin{bmatrix} x & \frac{1}{x^2+1} & \frac{x^2}{x^2+1} \end{bmatrix}^T = 0 \quad (3.4)$$

Fortunately, the LFT can produce some rows in $C_b(x, \delta)$, which constitutes the extra relations between basis functions having different denominators. Therefore, we assumed that it is enough if in each row of the matrix there appear only two nonzero items, which will eliminate the two corresponding elements in π_b . In our algorithm, we search for an adequate $\aleph_{\pi_b}(x, \delta)$ in the following form:

$$\aleph_{\pi_b}(x, \delta) = \begin{bmatrix} \cdots & \alpha x_i & \cdots & \beta x_j & \cdots \\ \cdots & \alpha x_i & \cdots & \beta & \cdots \\ \cdots & \alpha & \cdots & \beta x_j & \cdots \end{bmatrix} \quad (3.5)$$

We have chosen two elements from π , let them be $a(x, \delta), c(x, \delta)$. If the numerator $p(x, \delta)$ and the denominator $q(x, \delta)$ of their simplified fraction $a(x, \delta)/c(x, \delta)$ are monomials of degree 0 or 1, then there exists $b(x, \delta) = q(x, \delta)$, $d(x, \delta) = -p(x, \delta)$ affine monomials such that $a(x, \delta)b(x, \delta) + c(x, \delta)d(x, \delta) = 0$. If this new row do not appears in $C_b(x, \delta)$, it can be appended to the annihilator matrix: $\begin{bmatrix} \cdots & b(x, \delta) & \cdots & d(x, \delta) & \cdots \end{bmatrix}$. This procedure is evaluated on each pair of the elements of π .

3.4 Finding an appropriate Lyapunov function

After the previous steps we can assume a model (2.9). A Lyapunov function candidate for this system is given in the form $V(x, \delta) = x^T \mathbf{P}(x, \delta) x = \pi_b^T P \pi_b$, where $P \in \mathbb{R}^{m \times m}$ is a constant symmetric matrix. As it has been mentioned in Section 2.3, the positive definiteness of $V(x, \delta)$ can be expressed by a stricter inequality:

$$\begin{aligned} \exists L_b \in \mathbb{R}^{m \times q}, \quad \text{such that:} \\ P + L_b \aleph_{\pi_b}(x, \delta) + \aleph_{\pi_b}(x, \delta)^T L_b^T > 0, \quad \forall x \in \vartheta(\mathcal{X} \times \mathcal{D}) \end{aligned} \quad (3.6)$$

The negative definiteness of $\dot{V}(x, \delta, \dot{\delta}) = \nabla V(x, \delta) \dot{x}$ is ensured by the following sufficient LMI condition:

$$\begin{aligned} \exists L_a \in \mathbb{R}^{(n^2+n+2p+np) \times q}, \quad \text{such that:} \\ P_a + P_a^T + L_a \aleph_{\pi_a}(x, \delta, \dot{\delta}) + \aleph_{\pi_a}(x, \delta, \dot{\delta})^T L_a^T < 0, \quad \forall x \in \vartheta(\mathcal{X} \times \mathcal{D} \times \check{\mathcal{D}}), \end{aligned} \quad (3.7)$$

where the matrix P_a is defined as:

$$P_a = \begin{bmatrix} PA_a \\ 0_{(p+n^2+np) \times (n+2p+n^2+np)} \end{bmatrix} \quad A_a = \begin{bmatrix} A & B & 0_{n \times p} & 0_{n \times (n^2+np)} \\ 0_{p \times n} & 0_{p \times p} & I_p & 0_{p \times (n^2+np)} \end{bmatrix} \quad (3.8)$$

In order to define the annihilator $\mathcal{N}_{\pi_a}(x, \delta, \dot{\delta})$, some auxiliary variables are needed to be introduced:

$$\begin{aligned} E_k &= \text{row}_k(I_n) \\ E_b(x) &= [xE_k]_{\text{col}}^{k=\overline{1,n}} = I_n \otimes x \in \mathbb{R}^{n^2 \times n} \\ E_a(x) &= [\pi E_k]_{\text{col}}^{k=\overline{1,n}} = I_n \otimes \pi \in \mathbb{R}^{np \times n} \end{aligned} \quad \begin{aligned} J_0 &= \begin{bmatrix} I_n \\ 0_{p \times n} \end{bmatrix} \\ J_1 &= \begin{bmatrix} 0_{n \times p} \\ I_p \end{bmatrix} \end{aligned} \quad (3.9)$$

The $\mathcal{N}_{\pi_b}(x, \delta)$ affine annihilator of π_b is decomposed as follows:

$$\mathcal{N}_{\pi_b}(x, \delta) = N_0 + \sum_{k=1}^n \bar{N}_k x_k + \sum_{k=1}^d \hat{N}_k \delta_k \quad (3.10)$$

$$\bar{N}(x) = \sum_{k=1}^d \bar{N}_k x_k, \quad \hat{N}(\delta) = \sum_{k=1}^d \hat{N}_k \delta_k, \quad \hat{N}(\dot{\delta}) = \sum_{k=1}^d \hat{N}_k \dot{\delta}_k \quad (3.11)$$

where x_k, δ_k are elements of x and δ , respectively. By using the $N_0, \bar{N}_k, \hat{N}_k$ constant matrices, the following variables are constructed:

$$\begin{aligned} \bar{N}_b &= [\bar{N}_k J_0]_{\text{row}}^{k=\overline{1,n}} \in \mathbb{R}^{q \times n^2} & G_b(x, \delta) &= I_n \otimes \mathcal{N}_{\pi_b}(x, \delta) J_0 \in \mathbb{R}^{qn \times n^2} \\ \bar{N}_a &= [\bar{N}_k J_1]_{\text{row}}^{k=\overline{1,n}} \in \mathbb{R}^{q \times np} & F_b(x, \delta) &= I_n \otimes \mathcal{N}_{\pi_b}(x, \delta) J_1 \in \mathbb{R}^{qp \times np} \end{aligned} \quad (3.12)$$

Finally, the annihilator $\mathcal{N}_{\pi_a}(x, \delta, \dot{\delta})$ is defined as:

$$\mathcal{N}_{\pi_a}(x, \delta, \dot{\delta}) = \begin{bmatrix} \mathcal{N}_{\pi_b}(x, \delta) & 0_{q \times p} & 0_{q \times n^2} & 0_{q \times np} \\ W(x, \delta, \dot{\delta}) & \mathcal{N}_{\pi_b}(x, \delta) J_1 & \bar{N}_b & \bar{N}_a \\ E_b(x)[A \ B] & 0_{n^2 \times p} & -I_{n^2} & 0_{n^2 \times np} \\ 0_{ns_b \times m} & 0_{ns_b \times p} & \aleph_\mu(x) & 0_{ns_b \times np} \\ 0_{ns_a \times m} & 0_{ns_a \times p} & 0_{ns_a \times n^2} & \aleph_\eta(x, \delta) \\ 0_{qn \times m} & 0_{qn \times p} & G_b(x, \delta) & F_b(x, \delta) \end{bmatrix} \quad (3.13)$$

where:

$$\begin{aligned} W(x, \delta, \dot{\delta}) &= \hat{N}(\dot{\delta}) + \mathcal{N}_{\pi_b}(x, \delta) J_0 [A \ B] \\ \aleph_\mu(x) &= I_n \otimes \aleph_x \in \mathbb{R}^{ns_b \times n^2} \\ \aleph_\eta(x, \delta) &= I_n \otimes \aleph_\pi \in \mathbb{R}^{ns_a \times np} \end{aligned} \quad (3.14)$$

The matrices $P_a, E_k, E_b(x), G_b(x, \delta), F_b(x, \delta), \aleph_\mu(x), \aleph_\eta(x, \delta)$ are taken from [1]. The latter two variables are annihilators of μ and η artificial variables, respectively, which appear in the derivative of $\mathcal{N}_{\pi_b}(x, \delta)\pi_b$. They are defined as:

$$\begin{aligned} \mu_k &\stackrel{\text{def}}{=} x \dot{x}_k, \quad \mu = [\mu_k]_{\text{col}}^{k=\overline{1,n}} = E_b(x)[A \ B]\pi_b \in \mathbb{R}^{n^2} \\ \eta_k &\stackrel{\text{def}}{=} \pi \dot{x}_k, \quad \eta = [\eta_k]_{\text{col}}^{k=\overline{1,n}} = E_a(x)[A \ B]\pi_b \in \mathbb{R}^{np} \end{aligned} \quad (3.15)$$

The affine matrix function $\mathcal{N}_{\pi_a}(x, \delta, \dot{\delta})$ is constructed by merging the two annihilators $C_a(x, \delta, \dot{\delta})$ and $\aleph_{\pi_a}(x, \delta, \dot{\delta})$ defined in the same reference, and it is still an annihilator of

$\pi_a = [\pi_b, \dot{\pi}, \nu, \eta]_{\text{col}} \in \mathbb{R}^{m+p+n^2+np}$. The second and the third row of $\mathcal{N}_{\pi_a}(x, \delta, \dot{\delta})$ are consequences of the time derivative of $\mathcal{N}_{\pi_b}(x, \delta)\pi_b = 0$.

Similarly to $\mathcal{N}_{\pi_b}(x, \delta)$, the annihilator $\mathcal{N}_{\pi_a}(x, \delta, \dot{\delta})$ can also have redundant rows, these should be eliminated. In fact, redundancy does not only mean repetitive rows, but also rows filled with zeros.

3.5 Finding the maximal level set for a given polytope

Our solution for finding the maximal level set of Lyapunov functions is based on two popular approaches from the literature. The differences between the methods proposed by [15] and [1] are related to the definition of the sets, on which the conditions are stated, and the objective function, which is used to maximize the size of the invariant region.

1. Topcu prescribed the Lyapunov function to be positive definite on the whole $\mathbb{R}^n$ but the function's time derivative should be negative definite only in the inside of a level set ε , which would be the final invariant level set as it reached its maximal size (Lemma 1. in [15]). In order to maximize the size of ε , Topcu defined a variable sized region $\mathcal{P}_\beta = \{x \in \mathbb{R}^n \mid p(x) \leq \beta\}$, which should lie inside ε , while the variable β is maximized. The polynomial $p(x)$ is a design factor, which determines the shape and the orientation of the inner region.
2. On the other hand, [1] do not prescribe any constraint outside a given $\mathcal{X} \subset \mathbb{R}^n$ polytopic set, but for every $x \in \mathcal{X}$ the Lyapunov function and its derivative is required to be positive and negative definite, respectively. Furthermore, an additional constraint is introduced, more specifically, that the level set $\varepsilon = \{x \in \mathbb{R}^n \mid V(x) = 1\}$ should lie in the inside the polytope $\mathcal{X}$. In this case, the objective function to be minimized constitutes the sum of values of the Lyapunov function in some $x \in \mathcal{X}$ points, which are strategically chosen. Such an objective function can enforce that the level set $V(x) = 1$ will be as close to the boundary of $\mathcal{X}$ as possible.

In order to find the maximal invariant level set, we adopted a combined method of these two techniques mentioned above. First of all, we defined a small $\mathcal{Y}$ polytope around the locally stable origin inside $\mathcal{X}$. Similarly to the first approach, $\mathcal{Y}$ could be a variable size polytope, the size of which can be maximized during the optimization producing a maximal level set around $\mathcal{Y}$. The challenge is that the matrix inequality conditions are no longer linear using this approach, hence the need arises for further relaxations.

In comparison, we set $\mathcal{Y} \subset \mathcal{X}$ to have a small but constant size, looking for a level set $\varepsilon_\alpha = \{x \in \mathcal{X} \mid V(x) = \alpha, 1 \leq \alpha\}$ that is inside of $\mathcal{X}$, while ε_1 is outside of $\mathcal{Y}$. Finally, we have to maximize α , s.t.

1. ε_α lies in the inside of $\mathcal{X}$
2. $\mathcal{Y}$ is inside ε_1 (without this condition the $V(x)$ can be scaled arbitrarily obtaining an unbounded optimum).

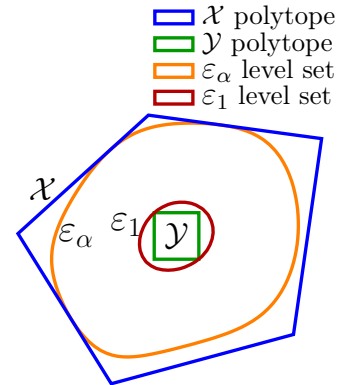


Figure 3.1

Transforming into LMI conditions

The first condition is satisfied if the value of the Lyapunov function is higher than α along the boundary of $\mathcal{X}$ (denoted by $\partial\mathcal{X}$). Formally:

$$V(x, \delta) = \pi_b^T P \pi_b \geq \alpha \quad \forall \delta \in \mathcal{D}, \quad \forall x \in \partial\mathcal{X} \quad (3.16)$$

Needless to say that $\partial\mathcal{X}$ is not a convex set, but it can be reproduced as the union of the facets of $\mathcal{X}$, which are indeed convex ($n - 1$ dimensional) manifolds (see Figure 3.2). Therefore, the same condition is tested several times for each facet of $\mathcal{X}$, which will be converted into LMI conditions.

In order to define the a_k orthogonal vector, let me introduce the following notations:

- $\mathcal{F}_k^{(\mathcal{X})}$ is the k th facet of $\mathcal{X}$, where $k = \overline{1, M_{\mathcal{X}}}$, $M_{\mathcal{X}}$ is the number of facets of $\mathcal{X}$,
- d_k is the distance of the origin to $\mathcal{F}_k^{(\mathcal{X})}$,
- n_k is a normal vector orthogonal to $\mathcal{F}_k^{(\mathcal{X})}$ pointing towards it, (Figure 3.2, red arrows)
- Finally $a_k = \frac{n_k}{d_k}$, which results that $\mathcal{F}_k^{(\mathcal{X})} \subset \{x : \langle a_k, x \rangle = 1\}$.

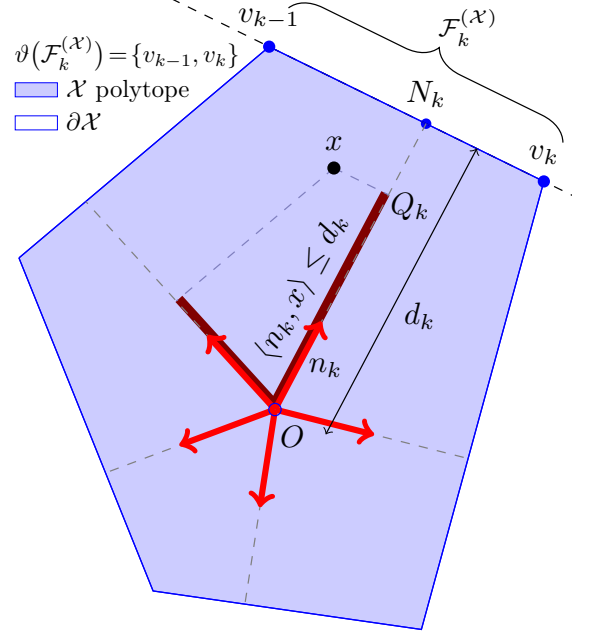


Figure 3.2

With the intention of demonstration, Figure 3.2 illustrates that an arbitrary $x \in R^n$ point is an element of $\mathcal{X}$ if $\langle n_k, x \rangle \leq d_k$, alternatively, if $\langle a_k, x \rangle \leq 1$ for all $k = \overline{1, M_{\mathcal{X}}}$.

Using the notations, the inequality (3.16) can be reformulated as:

$$\forall k = \overline{1, M_{\mathcal{X}}} : \quad V(x, \delta) = \pi_c^T \begin{bmatrix} P & 0 \\ 0 & -\alpha \end{bmatrix} \pi_c \geq 0 \quad \forall \delta \in \mathcal{D}, \quad \forall x \in \mathcal{F}_k^{(\mathcal{X})} \quad (3.17)$$

$$\text{where } \pi_c = \begin{bmatrix} \pi_b \\ 1 \end{bmatrix} \in \mathbb{R}^{m+1}$$

Similarly to [1], the following notations are introduced:

$$\begin{aligned} C_0 &= [I_n \ 0_{n \times p}] \in \mathbb{R}^{n \times m} & \mathcal{N}_{\pi_c}(x, \delta) &= [C_0 \ -x] \in \mathbb{R}^{n \times (m+1)} \\ C_k &= [a_k^T C_0 \ -1] \in \mathbb{R}^{1 \times (m+1)} & Q_k &= \begin{bmatrix} I_m \\ a_k^T C_0 \end{bmatrix} \in \mathbb{R}^{(m+1) \times m} \end{aligned} \quad (3.18)$$

Observe that $C_0 \pi_b = x$ and $\mathcal{N}_{\pi_c}(x, \delta) \pi_c = 0$ for all $x \in \mathbb{R}^n$. Furthermore, knowing that $x \in \mathcal{F}_k^{(\mathcal{X})} \subset \{x : a_k^T x = 1\}$ follows that $a_k^T C_0 \pi_b - 1 = C_k \pi_c = 0$ but only for $x \in \mathcal{F}_k^{(\mathcal{X})}$ (and not for every $x \in \mathbb{R}^n$), therefore C_k is a “weak” annihilator of π_c , which describes the extra properties of x originating from the fact that x is an element of $\mathcal{F}_k^{(\mathcal{X})}$. On the other hand, $C_k Q_k = 0_{1 \times m}$, consequently, Q_k is a null-space of C_k .

Two more variables are defined:

$$R_k(x, \delta) = \begin{bmatrix} L_{c_k} \mathcal{N}_{\pi_b}(x, \delta) + \mathcal{N}_{\pi_b}(x, \delta)^T L_{c_k}^T & 0_{m \times 1} \\ 0_{1 \times m} & 0_{1 \times 1} \end{bmatrix} + M_{c_k} \mathcal{N}_{\pi_c}(x, \delta) + \mathcal{N}_{\pi_c}(x, \delta)^T M_{c_k}^T \quad (3.19)$$

$$P_{c_k}^{(\alpha)}(x, \delta) = \begin{bmatrix} P & 0 \\ 0 & -\alpha \end{bmatrix} + R_k(x, \delta) \quad (3.20)$$

It is important to mention that $P_{c_k}^{(\alpha)}(x, \delta)$ is an affine matrix function in x and δ . The identities $\mathcal{N}_{\pi_b}(x, \delta)\pi_b = 0_{m \times 1}$ and $\mathcal{N}_{\pi_c}(x, \delta)\pi_c = 0_{(m+1) \times 1}$ for all $x \in \mathbb{R}^n$ implies that $\pi_c^T R_k(x, \delta)\pi_c = 0$ for all $x \in \mathbb{R}^n$, hence a sufficient condition for (3.17) can be formulated as:

$$\forall k = \overline{1, M_{\mathcal{X}}} : \pi_c^T P_{c_k}^{(\alpha)}(x, \delta)\pi_c \geq 0 \quad \forall \delta \in \mathcal{D}, \forall x \in \mathcal{F}_k^{(\mathcal{X})} \quad (3.21)$$

With reference to the equivalence of (i) and (iii) in the Finsler's lemma, and using that $C_k \pi_c = 0$ for all $x \in \mathcal{F}_k^{(\mathcal{X})}$, where C_k features a null-space Q_k , (3.21) is equivalent with the following LMI:

$$\begin{aligned} \forall k = \overline{1, M_{\mathcal{X}}} : \quad & \exists L_{c_k} \in \mathbb{R}^{m \times q}, \quad \exists M_{c_k} \in \mathbb{R}^{(m+1) \times n}, \quad \text{such that:} \\ & Q_k^T P_{c_k}^{(\alpha)}(x, \delta) Q_k \geq 0, \quad \forall (x, \delta) \in \vartheta(\mathcal{F}_k^{(\mathcal{X})} \times \mathcal{D}), \end{aligned} \quad (3.22)$$

Under the same consideration, the second condition ($\mathcal{Y}$ lies in the inside of ε_1) can be formulated as:

$$\begin{aligned} \forall k = \overline{1, M_{\mathcal{Y}}} : \quad & \exists \bar{L}_{c_k} \in \mathbb{R}^{m \times q}, \quad \exists \bar{M}_{c_k} \in \mathbb{R}^{(m+1) \times n}, \quad \text{such that:} \\ & \bar{Q}_k^T \bar{P}_{c_k}^{(\alpha)}(x, \delta) \bar{Q}_k \leq 0, \quad \forall (x, \delta) \in \vartheta(\mathcal{F}_k^{(\mathcal{Y})} \times \mathcal{D}), \end{aligned} \quad (3.23)$$

Finally, the optimization problem to find an appropriate Lyapunov function and its maximal invariant level set complete inside of a given polytope $\mathcal{X}$ constitutes:

$$\min_{1 \leq \alpha} (-\alpha) \quad \text{s.t. (3.6), (3.7), (3.22), (3.23)} \quad (3.24)$$

3.6 Finding the most appropriate outer polytope

Finding the most suitable $\mathcal{X}$, in which the Lyapunov conditions could be satisfied is an iterative problem. The basic concept is to choose an $\mathcal{X}^{(0)}$ initial polytope, which surely satisfies the LMI conditions, than, in each step ($\mathcal{X}^{(k)}$) find out the $\varepsilon^{(k)}$ maximal level set and give a new, larger $\mathcal{X}^{(k+1)}$ polytope considering the shape of $\varepsilon^{(k)}$. One possible solution can be choosing some uniformly distributed discrete points lying on $\varepsilon^{(k)}$ level set. These points will span a polytope, which should be enlarged with a given increment, such that its shape does not modify (practically the coordinates of every corner is multiplied by an $1 < \gamma \ll 2$ scalar factor).

In fact, there are cases when this iteration cannot be applied, especially in case of systems featuring significantly asymmetric DOA. In case of higher order systems, the same iteration can also be processed assuming a rectangular polytope.

4. Illustrative examples and results

In this chapter, we illustrate the applicability of the approach presented previously through diverse numerical examples.

The results presented in this section have been computed in the Matlab environment. For symbolic computations we used the Matlab's built-in Symbolic Math Toolbox based on MuPAD. For linear fractional transformations (LFT) we used the Enhanced LFR-toolbox [6, 5]. To model and solve semidefinite optimization (SDP) problems we used the Mosek solver with YALMIP [8].

In each example the values of the following variables will be presented:

1. the generated π column vector of nonlinearities and the A , B , $\mathcal{N}_{\pi_b}(x, \delta)$ matrices from the representation (2.6),
2. the generated matrix P from the Lyapunov function: $V(x, \delta) = \pi_b^T P \pi_b$ and the value of α . The π , P and α values determine the maximal ε_α invariant level set of Lyapunov function: $\varepsilon_\alpha = \{x \in \mathcal{X} \mid V(x) = \pi_b^T P \pi_b = \alpha\}$.
3. the area/volume (A/V) of the inner region bounded by ε_α , which will be given in square units (u^2) and cube units (u^3), respectively.

4.1 Van der Pol dynamics

The equation

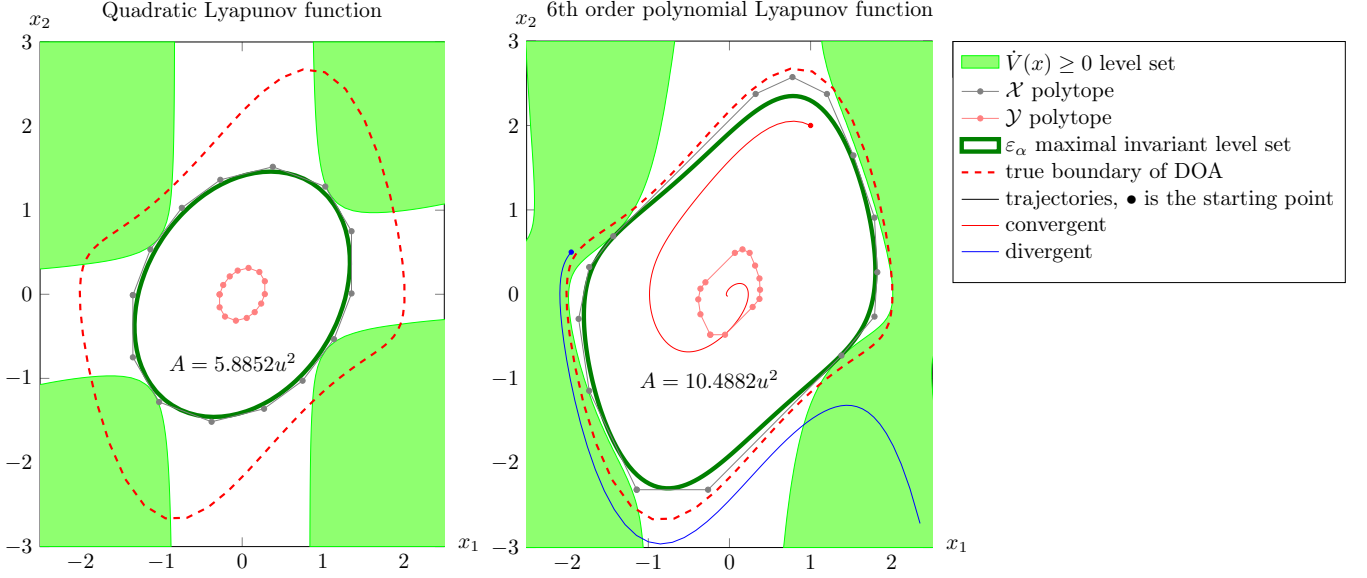
$$\begin{aligned} \dot{x}_1 &= -x_2 \\ \dot{x}_2 &= x_1 - \delta(1 - x_1^2)x_2 \end{aligned} \quad \text{with } \delta = 1 \tag{4.1}$$

describes a time-inverted oscillator introduced by the Dutch electrical engineer and physicist Balthasar van der Pol. This system has an asymptotically stable equilibrium point at the origin, and it has a limit cycle, which defines the boundary of the region of attraction (ROA). This limit cycle (dashed red line) and some trajectories (blue lines: divergent, red lines: convergent) of the system can be seen in Figure 4.1, which also illustrates the maximal invariant level set (green line, right) generated by the algorithm described in Chapter 3. The nonlinear basis functions of π used in [1] were

$$\pi^T = [x_1^2 \quad x_1 x_2 \quad x_2^2 \quad x_1^3 \quad x_1^2 x_2 \quad x_1 x_2^2 \quad x_2^3] \tag{4.2}$$

In comparison, this method generates a smaller number of monomials:

$$\pi^T = [x_1^2 x_2 \quad x_1 x_2]. \tag{4.3}$$



(a) The estimated DOA using a quadratic Lyapunov function (b) The generated DOA estimation using the algorithm presented in the previous chapters.

Figure 4.1: DOA estimation of the time inverted Van der Pol system

At the same time, the area of the estimated invariant region is close enough to that of the true DOA. The model matrices of (2.6) are:

$$A = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathcal{N}_{\pi_b}(x) = \begin{bmatrix} 0 & 0 & 1 & -x_1 \\ 0 & x_1 & 0 & -1 \\ x_2 & 0 & 0 & -1 \\ x_2 & -x_1 & 0 & 0 \end{bmatrix} \quad (4.4)$$

The P and invariant level set we obtained are the following:

$$P = \begin{bmatrix} 5.961 & -4.3596 & 1.1176 & -0.033319 \\ -4.3596 & 5.0502 & -4.5011\text{E-}6 & -0.0073781 \\ 1.1176 & -4.5011\text{E-}6 & -0.057457 & 0.019362 \\ -0.033319 & -0.0073781 & 0.019362 & 0.25254 \end{bmatrix} \quad (4.5)$$

$$\varepsilon_\alpha = \{x \in \mathcal{X} \mid V(x) = \alpha = 17.4086\} \quad (4.6)$$

The area bounded by ε_α is $A = 10.4882u^2$. The polytope $\mathcal{X}$ was generated iteratively, after that further corner points were added manually, which together comprise the final $\mathcal{X}$.

Quadratic Lyapunov function

Figure 4.1 features a smaller level set too (green line, right), which was obtained by the algorithm, but now constraining the Lyapunov function to be quadratic. This can be done if the decision variable P of the optimization process is constrained to be:

$$P = \begin{bmatrix} \bar{P}_{n \times n} & 0_{n \times p} \\ 0_{p \times n} & 0_{p \times p} \end{bmatrix}, \quad \bar{P}_{n \times n} \text{ is symmetric} \quad (4.7)$$

the generated Lyapunov function will be quadratic in x :

$$\bar{P}_{n \times n} = \begin{bmatrix} 12.8311 & -3.0838 \\ -3.0838 & 10.7177 \end{bmatrix} \Rightarrow V(x) \simeq 13x_1^2 - \frac{31x_1x_2}{5} + 11x_2^2 \quad (4.8)$$

The ε_α and its area are:

$$\varepsilon_\alpha = \{x \in \mathcal{X} \mid V(x) = \alpha = 21.2364\}, \quad A = 5.8852u^2 \quad (4.9)$$

The final $\mathcal{X}$ level set was generated iteratively in an automatic manner.

4.2 Uncertain Van der Pol dynamics

Here, the classical time-inverted Van der Pol model (4.1) is considered with an uncertain δ parameter value. The algorithm has generated the following uncertain model:

$$\pi = \begin{bmatrix} \delta x_2 \\ \delta x_1^2 x_2 \\ \delta x_1 x_2 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \end{bmatrix} \quad (4.10)$$

$$\mathcal{N}_{\pi_b}(x, \delta) = \begin{bmatrix} 0 & 0 & 0 & 1 & -x_1 \\ 0 & 0 & x_1 & 0 & -1 \\ 0 & \delta & -1 & 0 & 0 \\ x_2 & -x_1 & 0 & 0 & 0 \end{bmatrix} \quad (4.11)$$

We presumed $\delta \in [\delta_{\min}, \delta_{\max}]$ to be an unknown constant (i.e. its time derivative is considered zero). In the numerical calculations we used $\delta \in [1, 3]$. In Figure 4.2, the limit cycle of the system is illustrated for three different values of δ . The area of the obtained invariant domain (bounded by the green line) is $A = 5.7456u^2$. The generated matrix P from the $V(x, \delta)$ and the value of α are:

$$P = \begin{bmatrix} 15.2167 & -4.7641 & -0.42159 & 0.66543 & -0.0048827 \\ -4.7641 & 10.3229 & 0.054745 & 0.37397 & 0.041035 \\ -0.42159 & 0.054745 & 0.011178 & 8.015\text{E-}07 & -0.0067726 \\ 0.66543 & 0.37397 & 8.015\text{E-}07 & 0.023439 & -0.0053948 \\ -0.0048827 & 0.041035 & -0.0067726 & -0.0053948 & -0.24081 \end{bmatrix} \quad (4.12)$$

$$\alpha = 22.3339 \quad (4.13)$$

The polytope $\mathcal{X}$ is evaluated gradually in an automatic manner as described in Section 3.5. The initial polytope was chosen as $\mathcal{X}^{(0)} = [-0.3, 0.3]^2$.

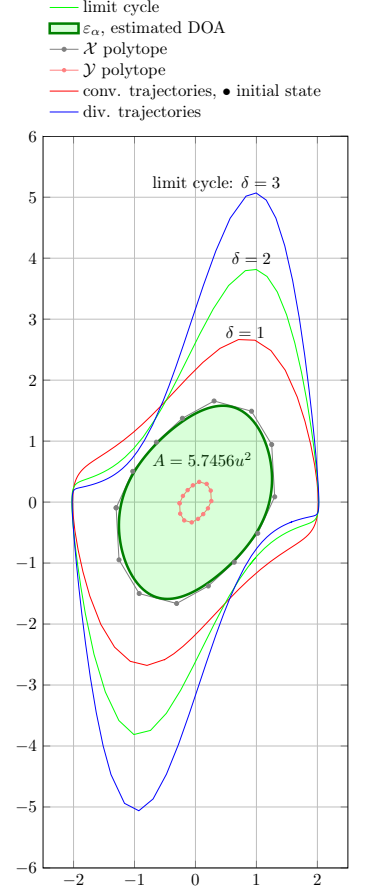


Figure 4.2: Van der Pol system with a constant unknown δ model parameter

4.3 Third order rational system

In order to illustrate the performance of the algorithm, a third order rational system is taken from [1]:

$$\begin{aligned} \dot{x}_1 &= x_2 + \varepsilon_3 x_3 + \varepsilon_1 \zeta(x) \\ \dot{x}_2 &= -x_1 - x_2 + \varepsilon_2 x_1^2 \\ \dot{x}_3 &= \varepsilon_3(-2x_1 - 2x_3 - x_1^2) \end{aligned} \quad (4.14)$$

where

$$\zeta(x) = \frac{x_1}{x_2^2 + 1}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = \frac{1}{2}$$

This system has a locally asymptotically stable equilibrium point at the origin. This fact can be seen from the negative eigenvalues of matrix A , which is in fact the Jacobian matrix of the system at the equilibrium. The model matrices obtained for this system are the following:

$$A = \begin{bmatrix} 0.5 & 1 & 0.5 \\ -1 & -1 & 0 \\ -1 & 0 & -1 \end{bmatrix} \quad (4.15)$$

$$B = \begin{bmatrix} 0 & 0 & 0 & -0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & -0.5 & 0 & 0 & 0 \end{bmatrix} \quad (4.16)$$

$$\pi^T = [x_1 x_2 \zeta(x) \quad x_1 \zeta(x) \quad x_1^2 \quad x_2^2 \zeta(x) \quad x_2 \zeta(x) \quad x_1 x_2] \quad (4.17)$$

$$\mathcal{N}_{\pi_b}(x, \delta) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & -x_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & x_2 & 0 & 0 & -x_1 \\ 0 & 0 & 0 & 0 & x_2 & 0 & 0 & 0 & -x_1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -x_1 & 0 \\ 0 & 0 & 0 & 1 & -x_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & 0 & 0 & -x_1 & 0 & 0 \\ 0 & x_1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & x_3 & -x_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_1 & 0 & 0 & 0 & -1 & 0 & -x_1 & 0 & 0 \\ x_1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ x_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ x_2 & 0 & 0 & 0 & 0 & 0 & -x_2 & -1 & 0 \\ x_2 & -x_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_3 & 0 & -x_1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.18)$$

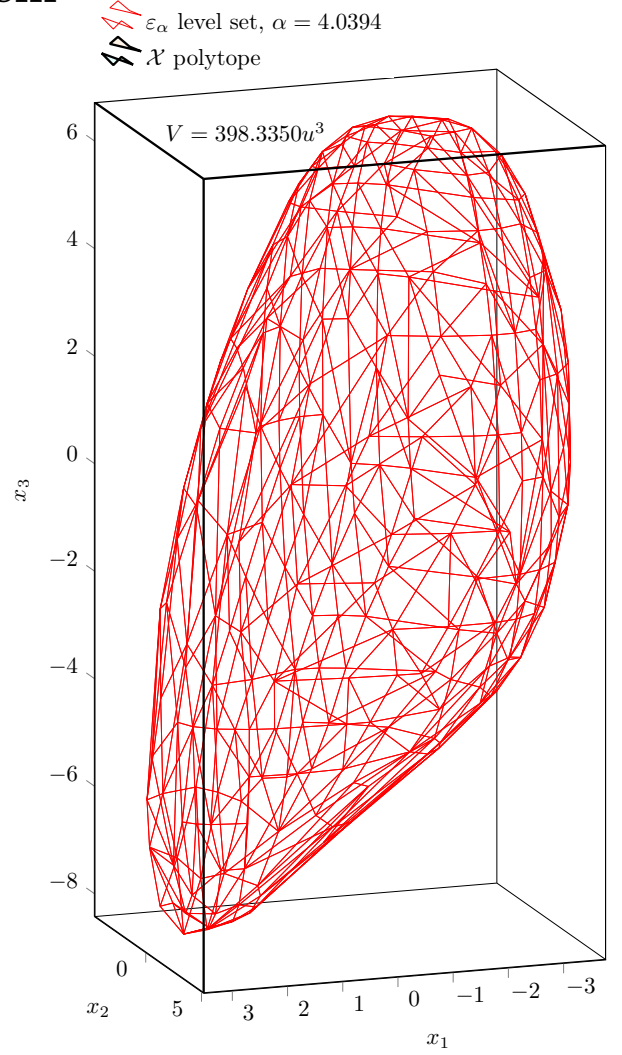


Figure 4.3: Estimated DOA of the third order rational system illustrated by the red polygon.

The maximal invariant level set of $V(x) = \pi_b^T P \pi_b$, and the generated matrix P are:

$$P = \begin{bmatrix} 0.24682 & 0.048679 & 0.11214 & 0.042366 & -14.9084 & 14.9247 & -0.050515 & 12.3269 & -12.2731 \\ 0.048679 & 0.2128 & 0.031464 & 0.021364 & 0.10189 & -0.15608 & -0.047148 & 0.060929 & 0.021878 \\ 0.11214 & 0.031464 & 0.11504 & -0.0067033 & 0.00080533 & 0.0178 & -0.0095644 & -0.0020643 & 0.0040921 \\ 0.042366 & 0.021364 & -0.0067033 & 0.01445 & -0.00141 & -0.0030255 & -3.9744 & 3.9999 & -4.0343 \\ -14.9084 & 0.10189 & 0.00080533 & -0.00141 & 0.0031407 & 0.0025979 & -4.0152 & -3.9773 & 5.268 \\ 14.9247 & -0.15608 & 0.0178 & -0.0030255 & 0.0025979 & 0.011247 & -10.8675 & -1.3232 & -0.0065652 \\ -0.050515 & -0.047148 & -0.0095644 & -3.9744 & -4.0152 & -10.8675 & 56.6889 & -0.023007 & 12.3672 \\ 12.3269 & 0.060929 & -0.0020643 & 3.9999 & -3.9773 & -1.3232 & -0.023007 & 56.4424 & -28.2011 \\ -12.2731 & 0.021878 & 0.0040921 & -4.0343 & 5.268 & -0.0065652 & 12.3672 & -28.2011 & 0.014463 \end{bmatrix} \quad (4.19)$$

$$\varepsilon_\alpha = \{x \in \mathcal{X} \mid V(x) = \alpha = 4.0394\} \quad (4.20)$$

The volume of the estimated DOA for the system in [1] is $32.022u^3$. Our modified algorithm generated an invariant level set of $398.3350u^3$. The possible cause of this difference can be the increased numerical stability obtained by the dimension reduction of the problem by considering less terms in π and eliminating the redundant rows from the annihilator matrices.

The polytope $\mathcal{X}$ was generated automatically, and it was bound to be a rectangular region, which results in a smaller number of facets being considered in the optimization task, furthermore, it's easier to approximate iteratively the most appropriate rectangular $\mathcal{X}$. The estimated DOA is shown in Figure 4.3.

4.4 A continuous fermentation process

It is known that bioreactors often show strongly nonlinear dynamical behaviour and their efficient control is a challenging task [17].

In this section, we present an isothermal nonlinear continuous fermentation process model taken from [2] with constant volume V and constant physico-chemical properties. The dynamics of the process is given by the state-space model

$$\frac{dX}{dt} = \mu(S)X - \frac{XF}{V} \quad (\text{biomass component mass balance}) \quad (4.21)$$

$$\frac{dS}{dt} = \frac{\mu(S)X}{Y} + \frac{(S_F - S)F}{V} \quad (\text{substrate component mass balance}) \quad (4.22)$$

$$\mu(S) = \mu_{max} \frac{S}{K_2 S^2 + S + K_1} \quad (\text{growth rate function}) \quad (4.23)$$

The variables and parameters of the model together with their units and values are given in Table 4.1. The above model can easily be written in standard input-affine form with the centered state vector

$$x = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T = \begin{bmatrix} S - S_0 & X - X_0 \end{bmatrix}^T \quad (4.24)$$

consisting of the centered biomass and substrate concentrations. The centered input flowrate is chosen as manipulate input variable $u = F - F_0$. The centered model is: $\dot{x} = f(x) + g(x)u$

$$f(x) = \begin{bmatrix} (x_1 + X_0) \cdot \mu(x_2 + S_0) & -\frac{(x_1 + X_0)F_0}{V} \\ -\frac{(x_1 + X_0) \cdot \mu(x_2 + S_0)}{Y} & +\frac{(S_F - (x_2 + S_0))F_0}{V} \end{bmatrix} \quad (4.25)$$

$$g(x) = -\frac{1}{V} \begin{bmatrix} x_1 + X_0 \\ x_2 + S_0 - S_F \end{bmatrix} \quad (4.26)$$

Table 4.1: Variables and parameters of the fermentation process

X	Biomass concentration		$[g/l]$
S	Substrate concentration		$[g/l]$
F	Feed flow rate		$[l/h]$
V	Volume	4	$[l]$
S_F	Substrate feed concentration	10	$[g/l]$
Y	Yield coefficient	0.5	
μ_{max}	maximal growth rate	1	$[l/h]$
K_1	Saturation parameter	0.03	$[g/l]$
K_2	Inhibition parameter	0.5	$[l/g]$
Steady-state operating point:			
X_0	equilibrium point of X	4.8907	$[g/l]$
S_0	equilibrium point of S	0.2187	$[g/l]$
F_0	Inlet feed flow rate	3.2089	$[l/h]$

The zero-dynamics of the centered system is an equilibrium point if

$$f(0) = \begin{bmatrix} X_0 \cdot \mu(S_0) - X_0 F_0 / V \\ -X_0 \cdot \mu(S_0) / Y + (S_F - S_0) F_0 / V \end{bmatrix} = 0, \quad (4.27)$$

meaning that $F_0 = V\mu(S_0)$ and $X_0 = (S_F - S_0)Y$. The S_0 coordinate of steady state operating point of the system is computed such that the biomass production $X_0 F_0$ is maximized:

$$S_0 = \arg \max_{S_0 > 0} (S_F - S_0)\mu(S_0) = \frac{-K_1 + \sqrt{K_1^2 + S_F^2 K_1 K_2 + S_F K_1}}{K_2 S_F + 1} \quad (4.28)$$

The computational steps of the previous statement can be seen in the sequel. Firstly, the expression of $X_0 S_0$ is transformed to a simplified form:

$$\begin{aligned} X_0 F_0 &= VY\mu(S_0)(S_F - S_0) = \mu_{max} VY \frac{-S_0^2 + S_F S_0}{K_2 S_0^2 + S_0 + K_1} \\ &= \frac{\mu_{max} VY}{K_2} \frac{-K_2 S_0^2 + K_2 S_F S_0}{K_2 S_0^2 + S_0 + K_1} \\ &= \frac{\mu_{max} VY}{K_2} \left(-1 + \frac{(K_2 S_F + 1)S_0 + K_1}{K_2 S_0^2 + S_0 + K_1} \right) \end{aligned} \quad (4.29)$$

In order to obtain the extremal value of the above expression, its derivative is computed with respect to S_0 .

$$\begin{aligned} \max_{S_0} : \frac{d}{dS_0} X_0(S_0)F_0(S_0) &= 0 \\ \Leftrightarrow \frac{d}{dS_0} \frac{(K_2 S_F + 1)S_0 + K_1}{K_2 S_0^2 + S_0 + K_1} &= 0 \\ \Leftrightarrow \frac{(K_2 S_F + 1)(K_2 S_0^2 + S_0 + K_1) - (2K_2 S_0 + 1)((K_2 S_F + 1)S_0 + K_1)}{(K_2 S_0^2 + S_0 + K_1)^2} &= 0 \\ \Leftrightarrow (K_2 S_F + 1)S_0^2 + 2K_1 S_0 - S_F K_1 &= 0 \end{aligned} \quad (4.30)$$

Finally, the optimal value of S_0 can be derived. The numerical operating point values are given in Table 4.1.

In the following, we present the computational result of the method. Firstly, the procedure is adopted on the open-loop system, then the effect of a negative substrate feedback is analyzed.

4.4.1 Open-loop system (OLS)

If the centered input flow rate $u = 0$, the model can be transformed into the following nonlinear state-space model:

$$\begin{aligned} \dot{x} &= f(x) = A(x)x, \quad \text{where} \\ A(x) &= \begin{bmatrix} -\frac{F_0(c_2x_2^2 + c_1x_2)}{q(x_2)V} & \frac{\mu_{max}V(x_1 + X_0) - X_0F_0(c_2x_2 + c_1)}{q(x_2)V} \\ -\frac{\mu_{max}S_0}{q(x_2)Y} & -\frac{F_0}{V} - \frac{\mu_{max}(x_1 + X_0)}{q(x_2)Y} - \frac{F_0(S_0 - S_F)(c_2x_2 + c_1)}{q(x_2)V} \end{bmatrix} \\ q(x_2) &= c_2x_2^2 + c_1x_2 + c_0 & c_1 &= 2K_2S_0 + 1 \\ c_0 &= K_2S_0^2 + S_0 + K_1 & c_2 &= K_2 \end{aligned} \quad (4.31)$$

The model matrices for this system

$$\begin{aligned} A &= \begin{bmatrix} 0 & 0.40111 \\ -1.6045 & -1.6045 \end{bmatrix}, \quad B = \begin{bmatrix} 0.082016 & -1.4716 & -0.7358 & -8.9905 \\ -0.16403 & 2.9432 & 1.4716 & 17.981 \end{bmatrix} \\ \pi &= [x_1x_2 \quad x_1x_2^2 \quad x_2^3 \quad x_2^2]^T \cdot \zeta(x_2), \quad \text{where } \zeta(x_2) \simeq (3.6688q(x_2))^{-1} \end{aligned} \quad (4.32)$$

The approximated value of $\mathcal{N}_{\pi_b}(x)$ with 4 decimal digit accuracy is:

$$\mathcal{N}_{\pi_b}(x) = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & -x_2 \\ 0 & 0 & 0 & 1 & 0 & -x_1 \\ 0 & 0 & 0 & x_2 & -x_1 & 0 \\ 0 & 0 & x_2 & -1 & 0 & 0 \\ 0 & 0 & x_2 & 0 & 0 & -x_1 \\ 0 & -0.2463x_2 & 0 & 0 & 0.4517x_2 + 0.101 & x_2 + 0.2463 \\ 0 & -0.2237x_1 & 0.2237 & 0 & 0.4103x_1 & x_1 \\ 0 & -0.2237x_2 & 0 & 0 & 0.4103x_2 & x_2 + 0.2237 \\ x_2 & -x_1 & 0 & 0 & 0 & 0 \\ -0.2463x_2 & 0 & x_2 + 0.2463 & 0.4517x_2 + 0.101 & 0 & 0 \\ -0.2237x_2 & 0 & x_2 + 0.2237 & 0.4103x_2 & 0 & 0 \end{bmatrix} \quad (4.33)$$

The generated P , α and the area of the estimate are:

$$\begin{aligned} P &= \begin{bmatrix} 971.2401 & 450.6579 & 11177.6448 & 243310.7436 & 6332.5198 & -14173.4478 \\ 450.6579 & 637.0046 & 20799.7569 & 87571.9883 & -13185.8278 & 1712.5523 \\ 11177.6448 & 20799.7569 & -265074.8729 & -488241.4065 & 43817.9314 & -38256.3894 \\ 243310.7436 & 87571.9883 & -488241.4065 & -361262.1183 & -14259.2742 & -277713.8548 \\ 6332.5198 & -13185.8278 & 43817.9314 & -14259.2742 & -96549.5592 & -134720.6252 \\ -14173.4478 & 1712.5523 & -38256.3894 & -277713.8548 & -134720.6252 & 22741.6173 \end{bmatrix} \\ \alpha &= 1.3354, \quad A = 0.010699u^2 \end{aligned} \quad (4.34)$$

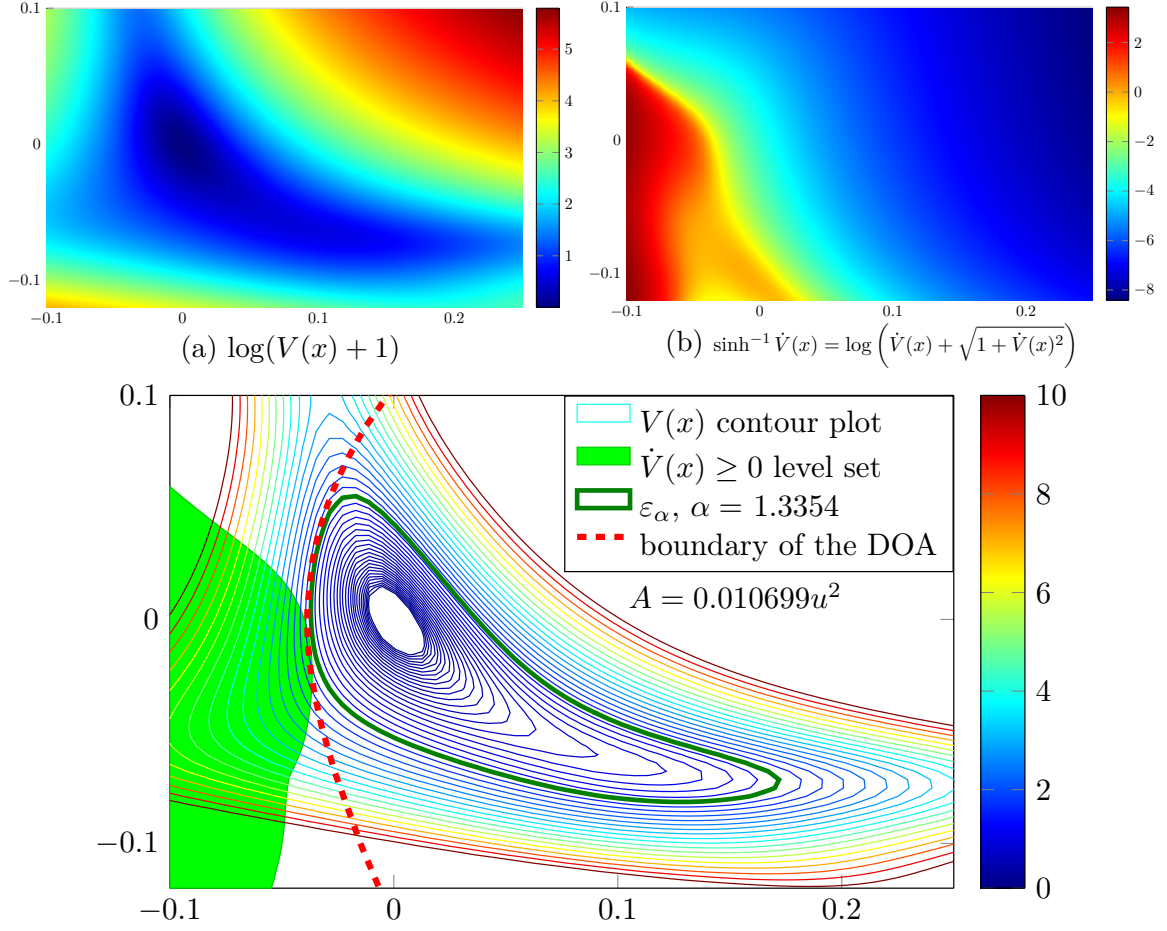


Figure 4.4: Lyapunov function and its derivative (OLS, $k = 0$)

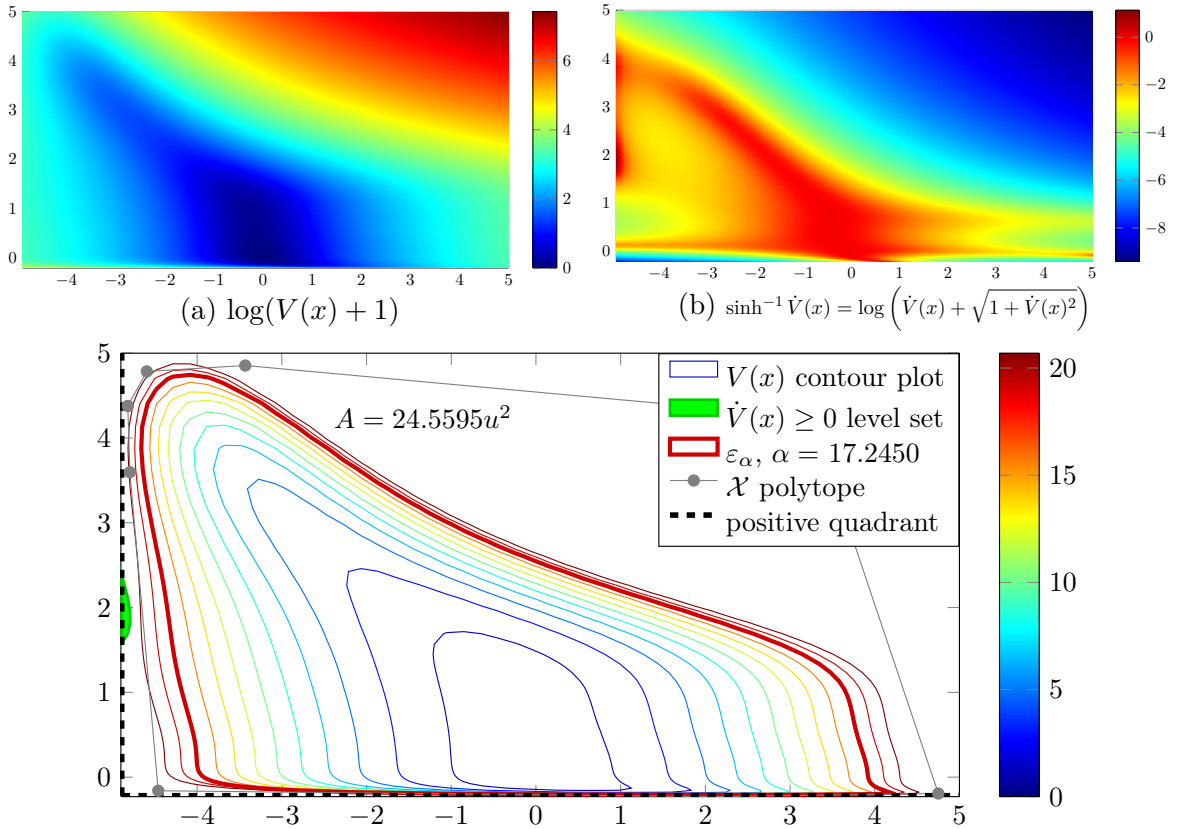


Figure 4.5: Lyapunov function and its derivative (CLS, $k = -0.8$)

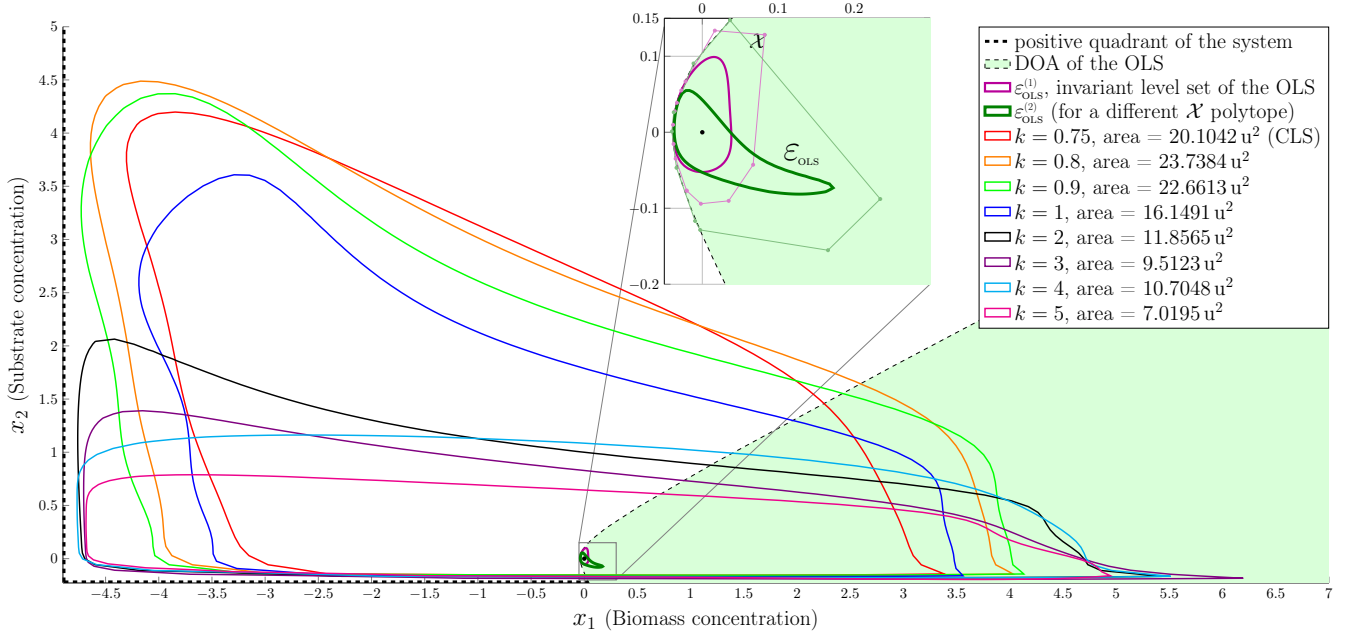


Figure 4.6: Green area: domain of attraction of the continuous fermentation process without feedback. The final invariant level set for the open-loop system (OLS) is shown in the magnified plot (violet and green, for two different polytope $\mathcal{X}$ s). The final invariant level sets for the closed loop system (CLS) are shown in different colors according to the different k feedback gains.

4.4.2 Effect of linear substrate feedback

Let us define the centered input flow rate as $u = -kx_2$, where $k > 0$ is the feedback gain. From (4.25,4.26), the equation of the closed-loop system can be transformed into the following form:

$$\begin{aligned} \dot{x} &= \mathfrak{A}(x)x, \quad \text{where} \quad \mathfrak{A}(x) = \begin{bmatrix} \mathfrak{A}_{11}(x) & \mathfrak{A}_{12}(x) \\ \mathfrak{A}_{21}(x) & \mathfrak{A}_{22}(x) \end{bmatrix} \\ \mathfrak{A}_{11}(x) &= -\frac{F_0(c_2x_2^2 + c_1x_2)}{r(x_2)V} - \frac{kx_2}{V} \\ \mathfrak{A}_{12}(x) &= \frac{\mu_{\max}V(x_1 + X_0) - X_0F_0(c_2x_2 + c_1)}{r(x_2)V} - \frac{kX_0}{V} \\ \mathfrak{A}_{21}(x) &= -\frac{\mu_{\max}S_0}{r(x_2)Y} \\ \mathfrak{A}_{22}(x) &= \frac{F_0(S_F - S_0)(c_2x_2 + c_1)}{r(x_2)V} - \frac{\mu_{\max}(x_1 + X_0)}{r(x_2)Y} - \frac{k(x_2 + S_0 - S_F) + F_0}{V} \end{aligned} \quad (4.35)$$

After the necessary transformations and using $k = 0.8$ as a feedback gain, the rounded numerical values of model matrices with 4 decimal digit accuracy are

$$\begin{aligned}
A &= \begin{bmatrix} 0 & 1.3792 \\ -1.6045 & -3.5607 \end{bmatrix}, \quad B = \begin{bmatrix} 0.082016 & 0.2 & -1.4716 & -0.7358 & -8.9905 & 0 \\ -0.16403 & 0 & 2.9432 & 1.4716 & 17.981 & 0.2 \end{bmatrix} \\
\pi^T &= \begin{bmatrix} (x_1 x_2) \zeta(x) & x_1 x_2 & (x_1 x_2^2) \zeta(x) & x_2^3 \zeta(x) & x_2^2 \zeta(x) & x_2^2 \end{bmatrix}, \quad \zeta(x) \simeq \left(3.6688 q(x_2)\right)^{-1} \\
\mathcal{N}_{\pi_b}(x, \delta) &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & -x_2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -x_1 & 0 \\ 0 & 0 & 0 & 0 & x_2 & -x_1 & 0 & 0 \\ 0 & 0 & 0 & x_2 & 0 & 0 & 0 & -x_1 \\ 0 & 0 & x_2 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & x_2 & 0 & 0 & 0 & -x_1 & 0 \\ 0 & x_1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & x_2 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & -0.2463x_2 & 0 & 0 & 0 & 0.4517x_2 + 0.101 & x_2 + 0.2463 & 0 \\ 0 & -0.2237x_1 & 0.2237 & 0 & 0 & 0.4103x_1 & x_1 & 0 \\ 0 & -0.2237x_2 & 0 & 0 & 0 & 0.4103x_2 & x_2 + 0.2237 & 0 \\ x_2 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ x_2 & -x_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.2463x_2 & 0 & x_2 + 0.2463 & 0 & 0.4517x_2 + 0.101 & 0 & 0 & 0 \\ -0.2237x_2 & 0 & x_2 + 0.2237 & 0 & 0.4103x_2 & 0 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{4.36}$$

The generated P , α and the area of the estimate are:

$$\begin{aligned}
P &= \begin{bmatrix} 1.1199 & 0.92033 & -30.625 & 30.4466 & -124.0099 & 135.835 & 17.7229 & -6.9136 \\ 0.92033 & 2.0768 & -107.4816 & 88.8882 & -484.882 & 98.0527 & -5.2247 & -12.5877 \\ -30.625 & -107.4816 & -33.3373 & 4.1097 & -67.6147 & 534.8658 & 48.412 & -21.4304 \\ 30.4466 & 88.8882 & 4.1097 & 2.1802 & -67.8471 & -387.9268 & -31.6243 & 0.78088 \\ -124.0099 & -484.882 & -67.6147 & -67.8471 & -7.4534 & -23.2976 & -348.0484 & 241.5972 \\ 135.835 & 98.0527 & 534.8658 & -387.9268 & -23.2976 & -215.8171 & -279.9563 & 23.7192 \\ 17.7229 & -5.2247 & 48.412 & -31.6243 & -348.0484 & -279.9563 & 87.3201 & 13.5614 \\ -6.9136 & -12.5877 & -21.4304 & 0.78088 & 241.5972 & 23.7192 & 13.5614 & 0.61819 \end{bmatrix} \\
\alpha &= 17.2450, \quad A = 24.5595u^2
\end{aligned} \tag{4.37}$$

The plot of the Lyapunov function and its derivative is illustrated in Figure 4.4. We have measured the area of the estimated DOA using different feedback gains. The results of the computations can be seen in Figure 4.6 alongside the estimated DOA of the open-loop system. Apparently, the closed-loop system features a much larger invariant level set than the open-loop system. Thus, it is clearly shown that a simple linear substrate feedback may ensure a practically useful guaranteed stability region.

5. Conclusions

In this work, we presented an optimization-based computational method for determining Lyapunov functions and invariant regions for nonlinear dynamical systems. The starting point of the method is the approach presented in [1]. The improvements and new contributions can be summarized as follows:

1. The model transformation to the required form for optimization is done automatically using LFT and further algebraic model transformation steps. This technique results in the dimension reduction of the problem compared to known solutions in the literature.
2. An algorithm was given for the generation of appropriate annihilators for the vector π_b .
3. An improved method was proposed for determining the possible largest invariant set for the dynamics using the computed Lyapunov function.
4. A simplified formula was given for the LMI, which ensures the negative definiteness of the Lyapunov function. The simplification was done by merging the two annihilators of π_b from the representation (2.6). The formula does not require the system matrix $F(x)$ to be quadratic in the model (2.6). Therefore, it can be considered as a generalization of the formula presented in [1].

The operation of the approach was illustrated through examples taken from the literature. The results confirmed the applicability of the proposed method, and the estimated DOAs were very similar or bigger than in [1]. It is worth mentioning again the third order rational system in Section 4.3, for which we obtained approximately 10 times larger DOA than the authors in [1].

The method was applied for the stability analysis of a simple continuous bioreactor model. The Lyapunov functions for the open and closed loop cases, respectively, were successfully computed and the corresponding guaranteed stability regions were determined for different feedback gains.

Publications

The results summarized in this institutional research report have been published in and submitted to [18] and [19], respectively.

Acknowledgements

Finally, we gratefully acknowledge the support of the National Research, Development and Innovation Office – NKFIH through grant OTKA NF104706. The research was also supported by Pázmány Péter Catholic University through the project KAP-1.1-15.

Bibliography

- [1] A. Trofino and T. J. M. Dezuo. LMI stability conditions for uncertain rational nonlinear systems. *International Journal of Robust and Nonlinear Control*, 2013. 2, 5, 6, 9, 11, 12, 13, 15, 18, 19, 25
- [2] G. Szederkényi, N. R. Kristensen, K. M. Hangos, and S. B. Jorgensen. Nonlinear analysis and control of a continuous fermentation process. *Computers and Chemical Engineering*, 26:659–670, 2002. 19
- [3] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.
- [4] Carsten Scherer and Siep Weiland. Linear matrix inequalities in control. *Lecture Notes, Dutch Institute for Systems and Control, Delft, The Netherlands*, 2000. 4
- [5] J-F. Magni. Linear fractional representation toolbox (version 2.0) for use with matlab. *Free Web publication <http://www.cert.fr/dcsd/idco/perso/Magni>*, 2006. 15
- [6] S. Hecker, András Varga, and J-F. Magni. Enhanced LFR-toolbox for MATLAB. pages 25–29, 2004. 15
- [7] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28).*, 2015.
- [8] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. 15
- [9] Graziano Chesi. *Domain of attraction: analysis and control via SOS programming*, volume 415. Springer Science & Business Media, 2011. 2
- [10] A. Vannelli and M. Vidyasagar. Maximal Lyapunov functions and domains of attraction for autonomous nonlinear systems. *Automatica*, 21:69–80, 1985. 2
- [11] Y. Ohta, H. Imanishi, L. Gong, and H. Haneda. Computer generated Lyapunov functions for a class of nonlinear systems. *IEEE Transactions on Circuits and Systems*, 40:343–354, 1993. 2
- [12] P. Giesl and S. Hafstein. Construction of Lyapunov functions for nonlinear planar systems by linear programming. *Journal of Mathematical Analysis and Applications*, 388:463–479, 2012. 2

- [13] Sz. Rozgonyi, K. M. Hangos, and G. Szederkényi. Determining the domain of attraction of hybrid non-linear systems using maximal Lyapunov functions. *Kybernetika*, 46:19–37, 2010. 2
- [14] Laurent El Ghaoui and Gérard Scorletti. Control of rational systems using linear-fractional representations and linear matrix inequalities. *Automatica*, 32(9):1273 – 1284, 1996. 2
- [15] U. Topcu, A. K. Packard, and P. Seiler. Local stability analysis using simulations and sum-of-squares programming. *Automatica*, 44(10):2669–2675, 2008. 2, 12
- [16] U. Topcu, A. Packard, P. Seiler, and G. Balas. Robust region-of-attraction estimation. *IEEE Transactions on Automatic Control*, 55(1):137–142, 2010. 2
- [17] D. Dochain, M. Perrier, and M. Guay. Extremum seeking control and its application to process and reaction systems: A survey. *Mathematics and Computers in Simulation*, 82:369–380, 2011. 19
- [18] Péter Polcz, Gábor Szederkényi, and Tamás Péni. An improved method for estimating the domain of attraction of nonlinear systems containing rational functions. *Journal of Physics: Conference Series*, 659(1):012038, 2015. 25
- [19] P. Polcz and G. Szederkényi. Computational stability analysis of a nonlinear fermentation process. In *11th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems - DYCOPS-CAB 2016, Trondheim, Norway, June 6-8*, page submitted, 2016. 25