



**HAL**  
open science

# Estimating the robust domain of attraction for non-smooth systems using an interval Lyapunov equation

Alexandre Goldsztejn, Gilles Chabert

► **To cite this version:**

Alexandre Goldsztejn, Gilles Chabert. Estimating the robust domain of attraction for non-smooth systems using an interval Lyapunov equation. *Automatica*, 2019, 100, pp.371-377. 10.1016/j.automatica.2018.03.036 . hal-01333216v2

**HAL Id: hal-01333216**

**<https://hal.science/hal-01333216v2>**

Submitted on 27 Jun 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Estimating the robust domain of attraction for non-smooth systems using an interval Lyapunov equation

Alexandre Goldsztejn <sup>a,1</sup> Gilles Chabert <sup>b</sup>

<sup>a</sup> *CNRS, Ecole Centrale de Nantes, LS2N*

<sup>b</sup> *Institut Mines Télécom Atlantique, LS2N*

---

## Abstract

The Lyapunov equation allows finding a quadratic Lyapunov function for an asymptotically stable fixed point of a linear system. Applying this equation to the linearization of a nonlinear system can also prove the exponential stability of its fixed points. This paper proposes an interval version of the Lyapunov equation, which allows investigating a given Lyapunov candidate function for non-smooth nonlinear systems inside an explicitly given neighborhood, leading to rigorous estimates of the domain of attraction (EDA) of exponentially stable fixed points. These results are developed in the context of uncertain systems. Experiments are presented, which show the interest of the approach including with respect to usual approaches based on sum-of-squares for the computation of EDA.

*Key words:* Nonlinear systems; Exponentially stable fixed point; Domain of attraction; Lyapunov function; Interval analysis.

---

## 1 Introduction

This paper addresses the computation of estimates of the domain of attraction (EDA), i.e., subsets of the domain of attraction (DA), of exponentially stable fixed points for nonsmooth uncertain systems. It is well known that the size of the DA is not related to the strength of attraction of the fixed point, as shown by the following simple example.

**Example 1** Consider the system  $x'(t) = \theta^3 x(t)^3 - \theta x(t)$ , for which the origin is an attracting fixed point. Since  $f'(0) = -\theta$ , the greater  $\theta$  the stronger the attraction of the origin. However, the DA of 0 is the open interval  $(-\theta^{-1}, \theta^{-1})$ , which decreases as  $\theta$  increases.

The computation of EDA hence brings complementary information to the study of the system linearization at the fixed point, and is of practical importance.

Lyapunov functions have turned out to be a central tool in the study of the stability of fixed points. They can

also be used to estimate their DA since, under mild assumptions, level sets of Lyapunov functions are included inside the DA. Proving that a function  $v$  is a Lyapunov function involves proving that a related function, namely the Lie derivative  $\dot{v}(x) = \nabla v(x)^T f(x)$  of the Lyapunov function with respect to the flow, is negative definite (ND), i.e., negative inside a given neighborhood of the fixed point except at the fixed point where it is zero. This can be done formally for simple systems:

**Example 2** Continuing Example 1, we choose a quadratic Lyapunov function  $v(x) = x^2$ . The Lie derivative of  $v$  with respect to the flow is  $\dot{v}(x) = v'(x)f(x) = 2x(\theta^3 x^3 - \theta x)$ . Its sign can be formally investigated as follows. It has 4 roots: 0 is a double root, with negative second derivative, therefore  $\dot{v}$  is locally negative definite at 0. The two other roots are  $\pm\theta^{-1}$ . Therefore,  $\dot{v}$  is negative inside the open interval  $(-\theta^{-1}, \theta^{-1})$ , except for  $\dot{v}(0) = 0$ . From Lyapunov theory,  $(-\theta^{-1}, \theta^{-1})$  is inside the DA of 0 (it is actually the RA of 0).

However, numerical algorithms, which are required for more realistic problems, are faced to a specific difficulty for proving this property: Indeed,  $\dot{v}(x)$  gets arbitrarily close to zero as  $x$  approaches the fixed point, hence preventing any attempt from proving directly numerically that it is positive except at the fixed point. This issue has been resolved in the framework of two different the-

---

*Email addresses:* alexandre.goldsztejn@gmail.com (Alexandre Goldsztejn), gilles.chabert@mines-nantes.fr (Gilles Chabert).

<sup>1</sup> Corresponding author. Tel: + 33 2 40 37 69 37. Fax: +33 2 40 37 69 30. Address: ECN, 1 rue de la No, 44300 Nantes, France.

ories by checking indirectly that  $\dot{v}$  is ND.

First, in the context of polynomial dynamical systems, proving that the Lie derivative is ND inside a region can be expressed as finding positive definite (PD) polynomials satisfying some constraints:

**Example 3** *Proving that  $\dot{v}$  is negative inside a set  $\mathcal{V}(c) = \{v(x) \leq c\}$  can be done by finding polynomials  $p(x)$ ,  $q_1(x)$  and  $q_2(x)$  that are positive for  $x \neq 0$  and furthermore satisfying*

$$p(x) + q_1(x)\dot{v}(x) = q_2(x)(v(x) - c). \quad (1)$$

*Indeed, the right hand side is negative if and only if  $x \in \mathcal{V}(c)$ , while the left hand side is positive if  $\dot{v}(x)$  is positive. Therefore  $\dot{v}(x)$  is proved to be negative for  $x \in \mathcal{V}(c)$  (see Theorem 4.1 from [1]). Continuing Example 1, for  $c = \theta^{-2}$ , we have the degenerated solution  $p(x) = 0$ ,  $q_1(x) = 1$  and  $q_2(x) = 2\theta^3 x^2$ , which proves  $\dot{v}(x) \leq 0$  inside  $\mathcal{V}(c) = [-\theta^{-1}, \theta^{-1}]$ . In fact, non degenerated solutions can be found for  $c < \theta^{-2}$ , which prove that  $\dot{v}(x) < 0$  inside  $\mathcal{V}(c) \setminus \{0\}$  for  $c < \theta^{-2}$ .*

Finding such PD polynomials can be relaxed to finding polynomials that are sum of squares (SOS). Finding such SOS polynomials amounts to solve linear matrix inequalities (LMIs), LMIs being convex and solved by efficient solvers today. This way, SOS polynomials have been applied to prove that functions are Lyapunov functions for polynomial systems, see, e.g., [2,3,1], and further extended to non-polynomial systems [4,5,6] and to hybrid systems [7]. In spite of its powerfulness, this approach presents several drawbacks: Its computational complexity turns out to be sensitive to both the dimension of the system and the degree of the polynomials involved in the description of the system (see Subsection 5.7). Furthermore, the current developments of this approach do not allow to handle fixed points that depend on the uncertainties. Finally, this approach happens to lack rigor in the computed results: LMI solvers are not certified, e.g., a relatively small uncertainty on the fixed point can cause the warning "SeDuMi numerical problems warning" of the LMI solver SEDUMI used in SMRSOFT [8], which is difficult to interpret. Verified automatic procedures for LMI error bounds [9] could be used, but were not investigated in this context up to our knowledge.

Second, interval analysis basically allows to check the sign of a given function, either by simple interval evaluation or using branch and bound algorithms. However, this approach fails because of the issue mentioned above. A more elaborated application of interval analysis to this end was proposed in [10]: The Lie derivative is formally computed and its Hessian is evaluated over a box centered on the fixed point. The negative definiteness of this interval matrix, which can be checked automatically, then implies under mild hypothesis that the Lie

derivative is concave and therefore negative, hence that the Lyapunov candidate function is actually a Lyapunov function. This test is called the interval Hessian test in the rest of the paper.

**Example 4** *Continuing Example 1, the second order derivative of  $\dot{v}(x)$  is  $\dot{v}''(x) = -4\theta + 24\theta^3 x^2$ . A simple interval evaluation of this expression on any interval  $[x]$  strictly included inside  $[-\frac{1}{\theta\sqrt{6}}, \frac{1}{\theta\sqrt{6}}]$  has a negative upper bound, hence proving that  $\dot{v}$  is concave inside  $[x]$ . Since  $\dot{v}(0) = 0$  and  $\dot{v}'(0) = 0$  (by construction),  $\dot{v}$  is proved to be negative inside  $[x]$ .*

**Remark 1** *Although not directly related to the present work, [11] proposed an interval method that builds a Lyapunov-like function to prove some capture property in a more general setting than fixed point stability. However, it requires an initial EDA.*

In the present paper, we propose another approach based on interval analysis: So-called slope matrices are used to rigorously linearize the system and the Lyapunov equation dedicated to linear systems is applied to this linearization, leading to a sufficient condition that requires only the interval evaluation of the first derivatives of the flow. This approach is carefully developed so as to handle uncertain dynamical systems and non-quadratic Lyapunov functions. Finally, we define a nonlinear optimization problem, which solution provides the largest estimate of the robust domain of attraction (LERDA) that can be computed given a Lyapunov function. We stress that the present paper considers a given Lyapunov candidate function and addresses the problem of proving that it is actually a Lyapunov function and building ERDA from it. The problem of finding good Lyapunov function candidates is not in the scope of this paper.

The paper is organized as follows: Section 2 introduces the basics of interval analysis that will be used. Section 3 presents the Interval Lyapunov Equation (8), which provides a verifiable sufficient condition for a Lyapunov candidate function to actually be Lyapunov function. Section 4 presents theoretical developments that use this sufficient condition to build an ERDA, as well as the nonlinear optimization problem that can extend this ERDA to the LERDA. Finally, Section 5 presents several academic systems to evaluate the proposed approach and compares it to the state of the art.

## 2 Interval analysis

### 2.1 Interval evaluation of an expression

Interval analysis is a branch of numerical analysis that was born in the 1960's. It consists in computing with intervals of reals instead of reals, providing a framework for handling uncertainties and verified computations (see e.g. [12,13]).

An interval is a closed connected subset of  $\mathbb{R}$ , the set of intervals being denoted by  $\mathbb{IR}$  (which includes the empty set as well as unbounded intervals). An interval vector  $[x] \in \mathbb{IR}^n$ , also called a box, is the cartesian product of its interval components, i.e.,  $[x] = [x]_1 \times [x]_2 \times \dots \times [x]_n$ . Interval matrices are defined similarly:  $[A] \in \mathbb{IR}^{n \times m}$  and  $A \in [A] \iff \forall (i, j) \in \{1, \dots, n\}^2, A_{ij} \in [A]_{ij}$ . A real number  $x \in \mathbb{R}$  (respectively a real vector  $x \in \mathbb{R}^n$  or a real matrix  $A \in \mathbb{R}^{n \times m}$ ) will be identified with the degenerated interval (respectively a degenerated interval vector or a degenerated interval matrix).

The image of a set  $X$  by a function is the set of all images of points in this set, i.e.,  $f(X) = \{f(x) : x \in X\}$ , called the range of  $f$  over  $X$ . So-called interval extensions of functions intend to compute over-approximations of this range. They are first defined for elementary functions. Operations  $\circ \in \{+, \times, -, \div\}$  are extended to intervals in the following way:  $[x] \circ [y] := \{x \circ y : x \in [x], y \in [y]\}$  (which is an interval since these functions are continuous). The division is defined for intervals  $[y]$  that do not contain zero. Unary elementary functions  $f(x)$  like  $\exp(x)$ ,  $\ln(x)$ ,  $\sin(x)$ , etc., are also extended to intervals similarly:  $f([x]) = \{f(x) : x \in [x]\}$ . All these elementary interval extensions form the interval arithmetic (IA). As real numbers are identified to degenerated intervals, the IA actually generalizes the real arithmetic, and mixed operations like  $1 + [1, 2] = [2, 3]$  are interpreted as interval operations, e.g. in this case  $[1, 1] + [1, 2] = [2, 3]$ . An interval function  $[f] : \mathbb{IR}^n \rightarrow \mathbb{IR}^m$  is an interval extension of the real function  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  if for all  $[x] \in \mathbb{IR}^n$  we have  $[f]([x]) \supseteq f([x])$ . Thus interval extensions allow computing enclosures of real functions range over boxes. So-called natural interval extensions of a function are obtained by evaluating an expression of this function for interval arguments using the IA. In particular when every variable has one unique occurrence in the function's expression the natural interval extension is optimal, i.e. it computes the exact function range. However, when a function expression has several occurrences of some variables, its interval evaluation may overestimate the range.

**Example 5** Consider the function  $\dot{v}(x) = 2x(x^3 - x)$  with  $\theta = 1$ . The range of this function over  $[x] = [-1, 1]$  is  $[-0.5, 0]$ . The interval evaluation of this expression for  $[x]$  is  $[-4, 4]$ , which overestimates the range of the function. The interval evaluation of the factorized expression  $\dot{v} = 2x^2(x - 1)(x + 1)$  is  $[-8, 0]$ , which also overestimates the range but proves that  $\dot{v}(x) \leq 0$  inside  $[x]$ .

The overestimation of the range illustrated in the previous example is one of the critical issue to be tackled when applying interval analysis. This example shows the main issue to be resolved in this paper: The function  $\dot{v}(x)$  is positive definite, but the simple interval evaluation of its expression will fail proving this fact. In general for higher dimensional non trivial systems, no factorization

will be able to prove that a function is ND by a simple interval evaluation.

In practice, the interval arithmetic is implemented using machine representable numbers and outer rounding. Among numerous implementations of IA, we can cite the C/C++ libraries GAOL [14] and the MATLAB toolbox INTLAB [15]. The developments presented in the rest of the paper use the ideal real IA. The algorithms are finally implemented using outwardly rounded floating point IA.

## 2.2 Slope matrices

Given a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ , a box  $[x] \in \mathbb{IR}^n$  and a vector  $\tilde{x} \in [x]$ , a  $(f, [x], \tilde{x})$ -slope matrix enclosure is an interval matrix  $[S]$  such that

$$\forall x \in [x], \exists S(x) \in [S], f(x) = f(\tilde{x}) + S(x)(x - \tilde{x}). \quad (2)$$

The matrix  $S(x)$  is called a  $(f, [x], \tilde{x})$ -slope matrix and  $[S]$  its enclosure. For differentiable functions  $f$ , a slope matrix enclosure can be computed using interval evaluations of the derivatives. For example,  $[S] = [Df]([x])$  is a  $(f, [x], \tilde{x})$ -slope matrix enclosure, which actually holds for any  $\tilde{x} \in [x]$ . Hansen proposed in [16] to evaluate the derivatives in a more clever way on sub-boxes of  $[x]$  to obtain a tighter  $(f, [x], \tilde{x})$ -slope matrix enclosure:

$$[S]_{ij} = \left[ \frac{\partial f_i}{\partial x_j} \right]([x_1], \dots, [x_j], \tilde{x}_{j+1}, \dots, \tilde{x}_n). \quad (3)$$

Hansen's matrix is sharper than the simple interval evaluation of the derivatives, but is valid only for a fixed  $\tilde{x}$ .

Both slope matrix enclosures can be generalized to non-differentiable Lipschitz functions  $f$  using generalized gradients [17], and the corresponding mean-value theorem, or the inclusion algebra described in Section 2.3 of [12]. Although not detailed here, slope matrices can also be computed using a slope arithmetic [18]. This approach can lead to tighter slope matrix enclosure, in particular when max functions are involved, e.g., for systems with saturations (see Appendix A for an example of INTLAB [15] code that computes a slope matrix enclosure using the slope arithmetic).

## 3 Verification of Lyapunov functions

In this section, we consider a Lipschitz continuous vector field  $f : [u] \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ , where  $[u]$  is a bounded neighborhood of the origin, such that  $f(0) = 0$ . We also consider a twice differentiable Lyapunov candidate function  $v : [u] \rightarrow \mathbb{R}$ , i.e.,  $v(0) = 0$  and  $v$  is positive definite (PD) inside  $[u]$ . We say that this Lyapunov candidate function is a  $[u]$ -Lyapunov function if furthermore  $\forall u \in [u] \setminus \{0\}, \dot{v}(u) < 0$ , where  $\dot{v}(u) = \nabla v(u)^T f(u)$ . From Lyapunov theory, the existence of such a Lyapunov

function entails that the origin is asymptotically stable, and that any level set  $\mathcal{V}(c) = \{u \in \mathbb{R}^n : v(u) \leq c\} \subseteq [u]$  is included inside the DA for the origin.

We consider a  $(f, [u], 0)$ -slope matrix  $A(u) \in [A]$ :

$$\forall u \in [u], \exists A(u) \in [A], f(u) = A(u)u. \quad (4)$$

We also consider a  $(\nabla v, [u], 0)$ -slope matrix  $P(u) \in [P]$ :

$$\forall u \in [u], \exists P(u) \in [P], \nabla v(u) = P(u)u. \quad (5)$$

Note that neither  $P(u)$  nor  $[P]$  are symmetric in general.

As seen in (4) and (5), these slope matrices allow rewriting the vector field and the Lyapunov candidate function gradient under the form of linear functions. As a consequence, the Lie derivative of  $v$  has the following form:

$$\dot{v}(u) = \nabla v(u)^T f(u) = u^T P(u)^T A(u)u \quad (6)$$

The following proposition uses these rigorous linearizations to extend the Lyapunov equation for linear systems to the nonlinear system in the neighborhood  $[u]$ .

**Proposition 1** *Given a Lyapunov candidate function  $v$ , define*

$$Q(u) = P(u)^T A(u) + A(u)^T P(u), \quad (7)$$

*and suppose that  $Q([u])$  contains only SND matrices. Then  $v$  is a  $[u]$ -Lyapunov function for  $f$ . In particular,  $v$  is a  $[u]$ -Lyapunov function if the interval matrix*

$$[Q] = [P]^T [A] + [A]^T [P] \quad (8)$$

*is SND, i.e., all symmetric matrices inside  $[Q]$  are SND.*

**Proof.** The same quadratic form as (6) is obtained by symmetrizing its matrix, i.e.,  $\dot{v}(u) = \frac{1}{2}u^T (P(u)^T A(u) + A(u)^T P(u))u$ , which is equal to  $\frac{1}{2}u^T Q(u)u$  by definition of  $Q(u)$ . Since  $Q(u)$  is assumed to be SND,  $\dot{v}(u) < 0$  for  $u \in [u] \setminus \{0\}$ , and  $v$  is proved to be a  $[u]$ -Lyapunov function for  $f$ . Finally, since  $P(u) \in [P]$  and  $A(u) \in [A]$ , we have  $Q(u) \in [Q]$ . Therefore,  $[Q]$  SND implies  $Q(u)$  SND for all  $u \in [u]$ , and the first statement implies that  $v$  is a  $[u]$ -Lyapunov function.  $\square$

**Remark 2** *In fact, Proposition 1 also proves that 0 is exponentially stable. Therefore, it can succeed only if the fixed point is actually exponentially stable.*

**Remark 3** *Slope matrices have been used similarly in the more general context of hyperbolic fixed points in [19] (see Lemma 2.13 in that paper). The scope of Proposition 1 is smaller since it applies only to exponentially stable fixed points, but is stronger for this smaller class of fixed points.*

Checking if  $[Q]$  is SND is NP-hard [20]. However, easily implementable polynomial sufficient conditions are available, e.g., checking the Sylvester criterion using the interval LU-decomposition for computing the determinant enclosures.

In the special case of quadratic Lyapunov functions,  $P(u) = P = D^2 v(0)$  is symmetric, therefore (8) becomes  $[Q] = P[A] + [A]^T P$ .

## 4 Certified ERDA and LERDA

In this section, we consider an uncertain vector field  $f(x, \theta)$ ,  $\theta$  being an uncertain parameter, whose domain is  $\Theta = \{\theta \in [\theta] : g(\theta) \leq 0\}$ , where the box  $[\theta]$  is bounded. We suppose that for all  $\theta \in \Theta$  the system has a fixed point  $x_\theta \in [x_\theta]$ , an uncertain fixed point box enclosure  $[x_\theta]$  being given. Let  $\mathcal{D}_\theta$  be the DA of  $x_\theta$  for the system  $\dot{x} = f(x, \theta)$ . The robust domain of attraction (RDA) is then defined by

$$\mathcal{D}_\Theta = \bigcap_{\theta \in \Theta} \mathcal{D}_\theta. \quad (9)$$

In [1,6] only fixed points that do not depend on the uncertainty are considered. Notice that in the case where the fixed point actually depends on the uncertainty, the RDA may be empty although each uncertain system has a nonempty DA.

For a fixed uncertainty  $\theta \in \Theta$ , let  $f_\theta(u) = f(x_\theta + u, \theta)$ , whose fixed point is the origin. Note that  $\mathcal{D}_\theta - x_\theta = \{x - x_\theta : x \in \mathcal{D}_\theta\}$  is the domain of attraction of 0. Given a box  $[u] \ni 0$ , we denote by  $[A_\theta]$  a  $(f_\theta, [u], 0)$ -slope matrix enclosure and  $A_\theta(u)$  its corresponding slope matrix. Since translations affect neither derivatives nor slope matrices, e.g.,  $Df_\theta(u) = D_x f(x_\theta + u, \theta)$ , one easily verifies that for any fixed value  $\theta \in \Theta$  both  $[A_\theta] = [D_x f]([x], \theta)$  with  $[x] = [u] + x_\theta$ , and its Hansen improvement, whose  $(i, j)$  entry is

$$[A_\theta]_{ij} = \left[ \frac{\partial f_i}{\partial x_j} \right]([x]_1, \dots, [x]_j, (x_\theta)_{j+1}, \dots, (x_\theta)_n, \theta), \quad (10)$$

are  $(f_\theta, [u], 0)$ -slope matrix enclosures. A slope matrix enclosure  $[A_\Theta]$  valid for all values  $\theta \in \Theta$  is then  $[A_\Theta] = [D_x f]([x], [\theta])$ , with  $[x] = [x_\theta] + [u]$ , or its Hansen improvement

$$[A_\Theta]_{ij} = \left[ \frac{\partial f_i}{\partial x_j} \right]([x]_1, \dots, [x]_j, [x_\theta]_{j+1}, \dots, [x_\theta]_n, [\theta]). \quad (11)$$

Note that these slope matrices are valid for all  $\theta \in [\theta]$ , which is a superset of  $\Theta$ . This can sensibly overapproximate the slope matrix in the case where  $\Theta$  is actually smaller than  $[\theta]$ , this issue being addressed below by splitting  $[\theta]$ .

We also consider a given Lyapunov candidate function  $v : \mathbb{R}^n \rightarrow \mathbb{R}$ , together with a PD quadratic minorant denoted by

$$w(u) = u^T W u \leq v(u), \quad (12)$$

with  $W \in \mathbb{R}^{n \times n}$  being SPD. Such a minorant implies in particular that  $v$  is radially unbounded and therefore has bounded level sets. If  $v$  is quadratic PD then  $w = v$ . We also consider a  $(\nabla v, [u], 0)$ -slope matrix enclosure denoted by  $[P]$ , which can be the interval evaluation  $[P] = [D^2 v]([u])$  of the second derivatives of  $v$ , or its Hansen improvement whose  $(i, j)$  entry is

$$[P]_{ij} = \left[ \frac{\partial^2 v}{\partial u_i \partial u_j} \right]([u]_1, \dots, [u]_j, 0, \dots, 0). \quad (13)$$

The corresponding  $(\nabla v, [u], 0)$ -slope matrix is denoted by  $P(u)$ .

Given an approximation of the fixed point(s)  $\hat{x} \in \mathbb{R}^n$ , the aim of this section is to compute  $c > 0$  such that

$$\mathcal{V}(\hat{x}, c) = \hat{x} + \{u \in \mathbb{R}^n : v(u) \leq c\} \subseteq \mathcal{D}_\Theta. \quad (14)$$

Subsection 4.1 provides a test to decide, whether  $\mathcal{V}(\hat{x}, c)$  is inside the RDA. Subsection 4.2 provides a nonlinear optimization problem that allows computing the LERDA, i.e., given  $v$  and  $\hat{x}$  to compute the maximal  $c^*$  such that  $\mathcal{V}(\hat{x}, c)$  is inside the RDA if and only if  $c < c^*$ . This value of  $c^*$  is called the LERDA radius, although it is actually not a distance but a squared distance.

#### 4.1 Certifying an ERDA

The following proposition requires the computation of a box enclosure  $[u]$  of a level set  $\mathcal{V}(0, c)$ . Since  $w$  is a minorant of  $v$ , we have  $\mathcal{V}(0, c) \subseteq \mathcal{W}(0, c)$ , and  $w$  being quadratic PD,

$$[u] = c^{\frac{1}{2}} [-1, 1] \text{diag}^{\frac{1}{2}}(P^{-1}) \supseteq \mathcal{W}(0, c), \quad (15)$$

where  $\text{diag}^{\frac{1}{2}}$  is the square root of the diagonal elements (the interval hull of an ellipsoid is given, e.g., in [21]).

Furthermore, for absorbing the uncertainty of the fixed point inside a level set of the Lyapunov function, we will need a Lipschitz constant for this Lyapunov candidate function. It is given by the following lemma.

**Lemma 1** *The Lyapunov candidate function  $v$  is  $L$ -Lipschitz continuous inside  $[u]$  (for the Euclidean distance) with  $L = \|[P]([u])[u]\|_2$ .*

**Proof.** Consider arbitrary  $u, u' \in [u]$ . Then using the mean-value theorem for several variables, there exists  $u'' \in [u]$  such that  $v(u) - v(u') = \nabla v(u'')^T (u - u')$ .

The proof is concluded noting that  $|\nabla v(u'')^T (u - u')| \leq \|\nabla v(u'')\|_2 \|u - u'\|_2$  by the Cauchy-Schwartz inequality, and that  $\nabla v(u'') = P(u'')u'' \in [P]([u])[u]$ .  $\square$

**Proposition 2** *Given a Lyapunov candidate function  $v$ ,  $\hat{x} \in \mathbb{R}^n$ ,  $c > 0$  and  $[u] \supseteq \mathcal{V}(0, c)$ , suppose that the interval matrix*

$$[Q] = [P]^T [A_\Theta] + [A_\Theta]^T [P] \quad (16)$$

*is SND and that  $c > \Delta := L \|[x_\theta] - \hat{x}\|_2$  with  $L = \|[P]([u])[u]\|_2$ . Then  $\mathcal{V}(\hat{x}, c - \Delta) \subseteq \mathcal{D}_\Theta$ .*

**Proof.** We first prove that

$$\forall x_\theta \in [x_\theta], \mathcal{V}(\hat{x} - x_\theta, c - \Delta) \subseteq \mathcal{V}(0, c). \quad (17)$$

For arbitrary  $\theta \in \Theta$  and  $u' \in \mathcal{V}(\hat{x} - x_\theta, c - \Delta)$ , we have  $u' = \hat{x} - x_\theta + u$  with  $v(u) \leq c - \Delta$ . Therefore,  $|v(u) - v(u')| \leq L \|u - u'\|_2 \leq \Delta$ , which proves that  $v(u') \leq v(u) + \Delta \leq c$ , i.e.,  $u' \in \mathcal{V}(0, c)$ , from which (17) follows.

For an arbitrary  $\theta \in \Theta$ , (16) and Proposition 1 prove that  $v$  is a Lyapunov function for  $f_\theta$  inside  $[u]$ . Furthermore, since  $\mathcal{V}(0, c) \subseteq [u]$  by hypothesis,  $\mathcal{V}(0, c) \subseteq \mathcal{D}_\theta - x_\theta$ . As a consequence of (17),  $\mathcal{V}(\hat{x} - x_\theta, c - \Delta) \subseteq \mathcal{D}_\Theta - x_\theta$  or equivalently  $\mathcal{V}(\hat{x}, c - \Delta) \subseteq \mathcal{D}_\Theta$ .  $\square$

Given  $\hat{x}$  (usually chosen as  $\hat{x} \approx \text{mid}[x_\theta]$ ) and  $c$ , Proposition 2 can fail for the following reasons:

- The interval matrix (16) is not SND. This can be the consequence of a too large  $[u]$ , which can be reduced by decreasing  $c$ , or a too large  $\Theta$ , which can be reduced by splitting  $[\theta]$  and applying Proposition 2 to all resulting subdomains  $\Theta_i = \{\theta \in [\theta_i] : g(\theta) \leq 0\}$  (empty subdomains being discarded). Then, the smallest ERDA obtained applying Proposition 2 to all subdomains  $\Theta_i$ , i.e.,  $c - \Delta = \min_i c_i - \Delta_i$ , gives rise to the ERDA  $\mathcal{V}(\hat{x}, c - \Delta)$  valid for all uncertainties inside  $\Theta$ .
- The condition  $c \leq \Delta$  is not satisfied. This does not happen if  $[x_\theta]$  is small enough, in particular if the fixed point does not depend on the uncertainty. A too large uncertainty on the fixed point may prevent Proposition 2 from building any ERDA.

In the case where the fixed point does not depend on the uncertainty, i.e.,  $x_\theta = x_0$  for all  $\theta \in \Theta$ , and the vector field is smooth in a neighborhood of the fixed point Proposition 2 will succeed in building an ERDA provided that the Lyapunov candidate function is chosen correctly: Indeed, since (15) converges to zero as  $c$  does and the interval extension of the derivatives are convergent, decreasing  $c$  and splitting  $[\theta]$  makes the slope matrix enclosure converge to thin matrices, and therefore Proposition 2 will eventually succeed provided that

$$D^2 v(0) Df(x_0) + Df(x_0)^T D^2 v(0) \quad (18)$$

is SND. This is the case when  $D^2v(0)$  is chosen as the solution of the Lyapunov equation for the linearized system. Note that the worst-case complexity of splitting  $[\theta]$  to obtain an ERDA is exponential with respect to the number of uncertain parameters.

However, if the vector field is non-smooth at the fixed point then generalized derivatives may not converge to thin intervals and the approach may fail to compute any ERDA.

#### 4.2 LERDA computation

The following proposition shows how to increase a given ERDA to a LERDA by solving a nonlinear optimization problem.

**Proposition 3** *Suppose that the Lyapunov function  $v$  is radially unbounded, which is entailed by  $w(u) \leq v(u)$ , and consider a given ERDA  $\mathcal{V}(\hat{x}, \underline{c})$ . Let  $c^*$  be the (global) solution of the following nonlinear optimization problem:*

$$\begin{aligned} \min \quad & v(x - \hat{x}) \\ \text{subject to} \quad & v(x - \hat{x}) \geq \underline{c} \\ & \nabla v(x - \hat{x})^T f(x, \theta) \geq 0 \\ & \theta \in \Theta. \end{aligned} \quad (19)$$

Then for all positive lower bound  $c^+ < c^*$ , we have  $\mathcal{V}(\hat{x}, c^+) \subseteq \mathcal{D}_\Theta$ .

**Proof.** Since  $c^*$  is the global minimum, there is no feasible  $(x, \theta)$  such that  $v(x) \leq c^+ < c^*$ , i.e.,  $\forall (x, \theta) \in \mathbb{R}^n \times \Theta, \underline{c} \leq v(x - \hat{x}) \leq c^+$  implies  $\nabla v(x - \hat{x})^T f(x, \theta) < 0$ . That is, for any fixed  $\theta \in \Theta$  and any trajectory  $x(t)$  starting in  $\mathcal{V}(\hat{x}, c^+) \setminus \mathcal{V}(\hat{x}, \underline{c})$ , we have  $\frac{d}{dt}v(x(t) - \hat{x}) \leq \epsilon$  with

$$\epsilon = \max_{\substack{\underline{c} \leq v(x - \hat{x}) \leq c^+ \\ \theta \in \Theta}} \nabla v(x - \hat{x})^T f(x, \theta). \quad (20)$$

Since  $v$  is radially unbounded, the feasible set of this optimization problem is compact and  $\epsilon < 0$ . This trajectory therefore reaches  $\mathcal{V}(\hat{x}, \underline{c})$  in a finite time, and then converges to  $x_\theta$  because  $\mathcal{V}(\hat{x}, \underline{c}) \subseteq \mathcal{D}_\Theta$ .  $\square$

The initial ERDA  $\mathcal{V}(\hat{x}, \underline{c})$  can be computed using Proposition 2. The nonlinear optimization problem (19) can be solved using branch and bound algorithms, which compute a certified lower bound  $c^+ < c^*$  arbitrarily close to  $c^*$ .

It is crucial to note that the resolution has to be performed inside  $\mathbb{R}^n \times [\theta]$ . Generally, branch and bound algorithms are not able to handle such unbounded domains. Some specific techniques were recently proposed

	DER <sub>10-3</sub>	HAN <sub>10-3</sub>	HES <sub>10-3</sub>	OPT <sub>10-3</sub>	OPT <sub>10-4</sub>
$c^{(1)}$	0.00198	0.00198	0.000862	0.28 <sub>9</sub> <sup>8</sup>	0.28 <sub>90</sub> <sup>89</sup>
t.	4	< 1	< 1	376	464
$c^{(2)}$	0.0345	0.0345	0.0121	1.7 <sub>802</sub> <sup>99</sup>	1.80 <sub>11</sub> <sup>08</sup>
time	< 1	4	4	300	386

Table 1

Example 4.6 from [1]: The largest EDA that Proposition 2 can validate, using either the interval evaluation of the DERivative or its HANsen improvement and the interval HESSian test [10], as well as the LEDA radius enclosure computed by solving (19) (decimals in exponent show the lower bound, decimals in subscript the upper bound). The computation time is given in ms.

in [22] to handle unbounded domains using quadratic minorants. In the same line, the quadratic PD minorant  $w(u)$  gives rise to a specific contractor using (15), which allows handling unbounded domains.

## 5 Experiments

In all experiments, an approximate fixed point  $\hat{x}$  was provided and used as the center of the EDA. An interval Newton operator was used in order to compute a certified enclosure of the fixed point(s) starting at  $\hat{x}$ , including in Subsection 5.5 where the fixed point actually depends on the uncertainties. The Sylvester criterion was used to test whether an interval matrix is SPD or SND, the determinant being computed using an interval LU-decomposition.

A dichotomic search is used for finding the greatest value of  $c$  that can be validated using Proposition 2. The nonlinear problem (19) is solved using IBEX [23,24,25] with standard settings with an initial upper bound for the objective  $v(x - \hat{x})$  of  $10^8$ , which does not impact the rigorously since the domain of the variables are kept unbounded, but sensibly speeds up the solving process<sup>2</sup>. Experiments are performed on a standard laptop equipped with an Intel i7 3 GHz and 8 Gb of memory. The C++ code used to perform these experiments is available at <http://ibex-lib.org/papers>.

The interval Hessian test proposed in [10] has also been used in the framework proposed here, which handles more carefully EDA and tackles uncertain systems and non-quadratic Lyapunov functions. This allows a detailed comparison with our approach. A detailed comparison with [1] is also presented.

	DER <sub>10-3</sub>	HAN <sub>10-3</sub>	HES <sub>10-3</sub>	OPT <sub>10-3</sub>	OPT <sub>10-4</sub>
$c$	0.0247	0.0565	0.00987	$0.3212^{09}$	$0.3210_9^5$
time	< 1	< 1	< 1	52	52

Table 2

Example 1 from [5]: Same data as in Table 1.

### 5.1 Example 4.6 from [1]

The following 3D polynomial system was considered in [1]:

$$\begin{aligned}\dot{x}_1 &= x_1 - x_1^3 + \frac{1}{2}x_3 + x_2^2 \\ \dot{x}_2 &= -x_2 - x_2^3 + \frac{1}{2}x_3^2 - x_1^2 \\ \dot{x}_3 &= x_1 + 2x_2 - x_3^3 + x_1^2 - x_2^2\end{aligned}\quad (21)$$

This system has two attractive fixed points, which approximations are  $x^{(1)} = (1.362, -0.829, 0.956)$  and  $x^{(2)} = (-1.110, -0.524, -1.063)$ . The Lyapunov candidate function  $v(x) = x_1^2 + x_2^2 + x_3^2$  was used for both fixed points, and the following LEDA were found in [1]:  $c^{(1)} = 0.291$  and  $c^{(2)} = 1.803$ .

The results obtained using the proposed approach are given in Table 1, with the same conclusion as in the previous subsection. However, the LEDA provided by [1] are not compatible with the enclosures found here. The following feasible points are provided by IBEX using a relative precision of  $10^{-4}$ :  $x_f^{(1)} = (1.23994309, -1.03446048, 0.47379879)$  and  $x_f^{(2)} = (0.22508293, -0.49578243, -0.92706316)$ . They have positive Lie derivatives and belong to the LEDA provided in [1], which are therefore slightly larger than the LEDA. However, we ran SMRSOFT 1.2 [8] for this problem, which computed a slightly smaller LEDA compatible with the enclosures provided here. Timing for the LEDA computations by SMRSOFT 1.2 [8] are 10 seconds for the first fixed point, and 13 seconds for the second.

### 5.2 Example 1 from [5]

This system is a non-polynomial system defined by

$$\begin{aligned}\dot{x}_1 &= -x_1 + x_2 + \frac{1}{2}(\exp(x_1) - 1) \\ \dot{x}_2 &= -x_1 - x_2 + x_1x_2 + x_1 \cos(x_1).\end{aligned}\quad (22)$$

The origin is an attractive fixed point, which was studied using the quadratic Lyapunov function  $v(x) = x_1^2 + x_2^2$  in [5,6]. Using sixth order Taylor expansions of the non-polynomial operations, the LEDA  $c = 0.3210$  was found in [1], which was improved to  $c = 0.3216$  in [6].

<sup>2</sup> This also has the advantage of letting the algorithm halt in case the feasible set is empty, e.g., when the fixed point is globally attractive. In this case,  $\mathcal{V}(\hat{x}, 10^8)$  is proved to be inside the domain of attraction.

	DER <sub>10-3</sub>	HAN <sub>10-3</sub>	HES <sub>10-3</sub>	OPT <sub>10-3</sub>	OPT <sub>10-4</sub>
$c$	0.0288	0.0288	0.00995	$0.699_6^{89}$	$0.699_4^2$
time	< 1	4	< 1	215	248

Table 3

Whirling Pendulum. Same data as in Table 1.

	DER <sub>10-3</sub>	HAN <sub>10-3</sub>	HES <sub>10-3</sub>	OPT <sub>10-3</sub>
$c$	0.00858	0.0120	0.00385	$0.30_3^2$
time	< 1	4	< 1	1992

Table 4

Example 5.6 from [1]. Same data as in Table 1.

The results obtained using the proposed approach are given in Table 2, with the same conclusion as in the previous subsection. In particular, the LEDA found in [1] is compatible with the enclosure found here. However, this enclosure is not compatible with the LEDA found in [6]. The following feasible point has been found by IBEX using a relative precision of  $10^{-4}$ :  $x_f = (0.460239373, 0.330548554)$ . This point has a positive Lie derivative and belongs to the LEDA computed in [6], which is therefore slightly larger than the LEDA.

### 5.3 Whirling pendulum [5,6]

This model is a non-polynomial system defined by

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{c}{m}x_2 + \omega^2 \sin(x_1) \cos(x_1) - \frac{g}{l} \sin(x_1),\end{aligned}\quad (23)$$

with  $c = \frac{2}{10}$ ,  $m = 1$ ,  $\omega = \frac{9}{10}$  and  $g = l = 10$ . The origin is an attractive fixed point, which is used as the center of the computed EDA. The following quadratic Lyapunov function is used in [5,6]:  $v(x) = x_1^2 + x_1x_2 + 4x_2^2$ . Using seventh order Taylor expansions of the non-polynomial operations, the LEDA  $c = 0.6990$  was found in [1], which was improved to  $c = 0.6998$  in [6].

The results obtained using the proposed approach are given in Table 3, with the same conclusion as in the previous subsection. The LEDA enclosure is compatible with the one given in [5]. However, it is not compatible with one given in [6]: The feasible point  $x_f = (-0.74135892, 0.30717105)$ , computed by IBEX using a relative precision of  $10^{-4}$ , has a positive Lie derivative and belongs to  $\mathcal{V}(0, 0.69931)$ , which contradicts [6].

### 5.4 Example 5.6 from [1]

This system is an uncertain polynomial system defined by

$$\begin{aligned}\dot{x}_1 &= -2x_1 - x_2 - (1 - \theta_2)x_1^2 + \theta_1x_1x_2 \\ \dot{x}_2 &= -2x_1 - 3x_2 - \theta_2x_1x_2 + (2 + \theta_1)x_2^2.\end{aligned}\quad (24)$$

The domain for the uncertain parameters is  $\Theta = \{\theta \in \mathbb{R}^2 : \theta_1^2 + \theta_2^2 \leq 1\}$ . The origin is an attractive fixed point,



	DER <sub>10-3</sub>	HAN <sub>10-3</sub>	HES <sub>10-3</sub>	OPT <sub>10-3</sub>
$c$	0	0.0968	0	$4 \cdot \frac{59}{60}$
time	< 1	< 1	4	86
$c'$	0.00010	0.171	0	NA
time	34	33	132	NA

Table 5

A simple microbial growth process. Same data as in Table 1. The EDA  $c'$  are computed by a dichotomic search using Proposition 2 and splitting regularly the parameters domain into 25 subdomains (in this case,  $c'$  has no meaning).

which does not depend on the uncertainties and is used as a center for the computed EDA. The following quadratic Lyapunov function is used in [5]:  $v(x) = 3x_1^2 + x_1x_2 + x_2^2$ . The LERDA  $c = 0.303$  was found in [1].

The results obtained using the proposed approach are given in Table 4, with the same conclusion as in the previous subsection. The LERDA found in [1] is compatible with the enclosure found here.

### 5.5 A simple microbial growth process

This system is an **uncertain non-polynomial** system:

$$\begin{aligned} \dot{x} &= x \left( \frac{s\mu_m}{k_s + s} - d\alpha \right) \\ \dot{s} &= d(s_i - s) - \frac{\mu_m k s x}{k_s + s}, \end{aligned} \quad (25)$$

where  $\alpha = 0.5$ ,  $k = 10.53$ ,  $d = 0.36$ ,  $s_i = 5.7$ ,  $\mu_m \in [1.19, 1.21]$  and  $k_s \in [7.09, 7.11]$ . There are therefore two uncertain parameters. It was studied in [26] where an initial condition were rigorously simulated and seemed to converge to a fixed point close to  $\hat{x} = (0.845, 1.253)$ . This fixed point actually depends on the uncertainties, therefore the ERDA computed here actually proves the convergence of trajectories towards an uncertain fixed point.

The results obtained using our approach are given in Table 5. Proposition 2 fails using the derivative-based slope matrix enclosure as well as using the interval Hessian test [10] because the uncertain domain is too large. The ERDA  $\mathcal{V}(\hat{x}, c')$  is computed by splitting  $\Theta$  into 25 regular sub-intervals, the results being shown in Table 5. The derivative-based slope enclosure now succeeds computing an EDA, but [10] still fails. In fact, [10] fails computing any EAD for this system: For arbitrarily small uncertainty subdomains we have  $\Delta > c$  for some of them, which prevents any application of Proposition 2.

### 5.6 A linear system with saturation [27]

We consider the following **linear system with saturated input** [27]:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_1 - \max\{-5, \min\{5, 13x_1 + 7x_2\}\}. \end{aligned} \quad (26)$$

	DER <sub>10-3</sub>	HAN <sub>10-3</sub>	HES <sub>10-3</sub>	OPT <sub>10-3</sub>
$c_2$	0.0179	0.0179	0.0179	$0.20 \frac{7}{8}$
time	< 1	4	12	60
$c_4$	0.00510	0.00510	0.00139	$25 \cdot \frac{8}{9}$
time	< 1	4	4	184

Table 6

Linear system with saturations. Same data as in Table 1.

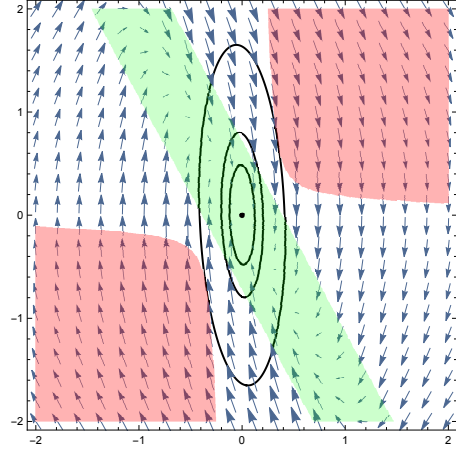


Fig. 1. The vector field for the linear system with saturated input. In green, the area where there is no saturation. In red, the area where the Lie derivative of the quadratic Lyapunov function is positive. The smaller ellipsoid correspond to the EDA computed using Proposition 2 using either the derivative enclosure or its Hansen improvement, as well as the interval Hessian test [10]. The medium-size ellipsoid corresponds to the EDA computed using Proposition 2 with the slope arithmetic (see Appendix A). The largest ellipsoid corresponds to the LEDA computed solving (19).

Linearizing the system at the stable fixed point 0 and solving the corresponding Lyapunov equation, we select the Lyapunov candidate function  $v_2(u) = 1.22x_1^2 + 0.083x_1x_2 + 0.077x_2^2$ . In order to extend the EDA computed with this Lyapunov function, we also consider the degree 4 Lyapunov candidate function  $v_4(x) = v_2(x) + (x_1 + x_2)^4$ .

The results obtained are summarized in Table 6, Figure 1 and Figure 2, which are in line with the previous results. We see that the dichotomic search stops at the same value either using the derivative slope enclosure, its Hansen improvement or the interval Hessian test [10]. This value corresponds exactly to the interval hull of the ellipsoid that reaches the saturations, where the slope enclosures become wider because of the null generalized derivative of the max function. This can be improved using the slope arithmetic [18]. Indeed, the short MATLAB code presented in Appendix A uses the slope arithmetic implemented in INTLAB [15] and proves using Proposition 2 that  $\mathcal{V}(0, 0.0484)$  is an EDA, which include saturated inputs, as seen on Figure 1.

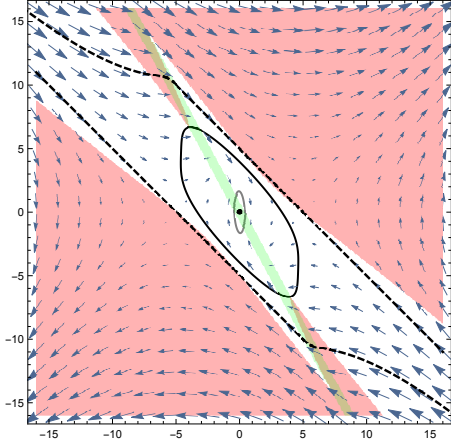


Fig. 2. Same as Figure 1, but for the degree 4 Lyapunov function. The DA is delimited by dashed lines. The EDA computed using Proposition 2 is too small to be visible. However, it supplies a lower bound necessary for solving (19), which solution gives rise to the non-ellipsoid EDA depicted in black full line. The gray ellipsoid corresponds to the LEDA computed using the quadratic Lyapunov function.

### 5.7 A scalable academic system

#### The scalable polynomial system

$$\dot{x} = -(1 - \prod x_i) x \quad (27)$$

has a stable fixed point at the origin. We consider the Lyapunov function  $v(x) = x^T x$ , whose LEDA can be proved to be  $\mathcal{V}(0, n)$ .

The EDA found by applying Proposition 2 cannot be better than  $\mathcal{V}(0, 1)$ : Indeed, the slope matrix enclosure for this EDA is computed over its interval hull  $[-1, 1]^n$ , which contains the fixed point  $(1, \dots, 1)$ . The values of  $c$  actually computed by dichotomy using Proposition 2 using the different slope matrix enclosures are shown in Figure 3. We see that they are smaller than 1, but converge to 1 as the dimension increases. Timings are shown in Figure 4. The resolution of the NLP (19) was run with a relative precision of  $10^{-3}$  and a timeout of 15 minutes. Both the system and the Lyapunov function enjoy the same symmetry: All variables can be exchanged. Following [28], we break this symmetry by adding the symmetry breaking constraints (SBCs)  $x_1 \leq x_i$  for all  $i \in \{2, \dots, n\}$ . The resolution of the NLP (19) allows computing a sharp enclosure of the LEDA without reaching the timeout until  $n = 5$  ( $n = 6$  using SBCs). For dimensions between  $n = 6$  and  $n = 11$  ( $n = 7$  and  $n = 15$  using SBCs), the branch and bound still allows enlarging the EDA computed using Proposition 3 but does not provide a sharp enclosure of the LEDA anymore. No improvement over the initial EDA is provided for higher dimensions.

We use SMRSOFT [8] to compute EDA for this system.

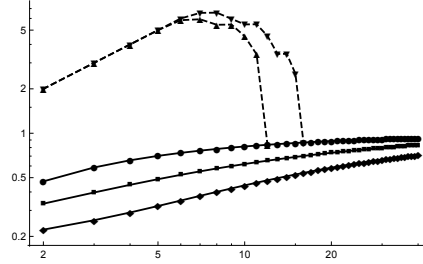


Fig. 3. Scalable academic problem: Radii for the different methods. Full line with diamond, square and disk: Dichotomy with Proposition 2 using respectively [10], derivative enclosure and Hansen improvement. Dashed line with upward and downward triangles: Solution of (19) respectively without and with symmetry breaking.

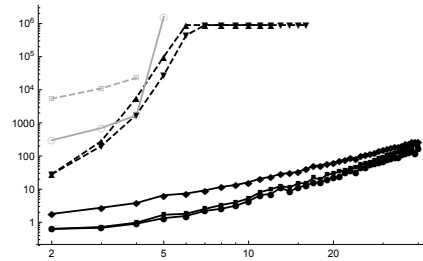


Fig. 4. Scalable academic problem: Timings for the different methods. Black: Same as Figure 3. Gray, full line circle and dashed line empty square: SMRSOFT for respectively checking the EDA  $\mathcal{V}(0, 0.5)$  and computing the LEDA.

It is used to check the EDA  $\mathcal{V}(0, 0.5)$ , and to compute the LEDA of this system. Timings, shown in Figure 4, are successfully computed up to  $n = 5$  for the EDA and  $n = 4$  for the LEDA, higher dimensions requiring too long computation and making our version of MATLAB crashing.

## 6 Conclusion

The results presented above show the interest of the proposed approach: Given a Lyapunov candidate function, it successfully builds the LERDA on academic state-of-the-art examples tackled by the SOS approach. On the one hand, it has interesting advantages with respect to the SOS approach: It handles directly non-polynomial systems, while the SOS approach requires Taylor expansions with bounded remainders, as well as non-smooth systems. Also, its application to the scalable benchmark has shown its superiority in terms of computational effort, although the branch and bound algorithm also remains restricted to small dimensions. On the other hand, the SOS-based approach has a wider scope: Some other decision variables can be included inside the resulting LMI problem (e.g. controller parameters or coefficients of the Lyapunov function), and in some cases, it can prove the asymptotic stability of non exponentially stable fixed point using non quadratic functions (see, e.g., Example 1 in [3]).

Finally, the interval Lyapunov equation test proposed here presents several improvements with respect to the interval Hessian test proposed in [10]: It requires only first-order derivatives interval evaluation, hence applicable to non-smooth systems, and actually computes larger ERDA.

## References

- [1] Chesi G. *Domain of Attraction: Analysis and Control via SOS Programming, Lecture Notes in Control and Information Sciences*, vol. 415. Springer, 2011.
- [2] Tibken B. Estimation of the domain of attraction for polynomial systems via LMIs. *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, vol. 4, 2000; 3860–3864 vol.4.
- [3] Papachristodoulou A, Prajna S. On the construction of Lyapunov functions using the sum of squares decomposition. *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, vol. 3, IEEE, 2002; 3482–3487.
- [4] Papachristodoulou A, Prajna S. *Analysis of Non-polynomial Systems Using the Sum of Squares Decomposition*. Springer Berlin Heidelberg, 2005; 23–43.
- [5] Chesi G. Estimating the domain of attraction for non-polynomial systems via LMI optimizations. *Automatica* 2009; **45**(6):1536 – 1541.
- [6] Wu M, Yang Z, Lin W. Domain-of-attraction estimation for uncertain non-polynomial systems. *Communications in Nonlinear Science and Numerical Simulation* 2014; **19**(9):3044 – 3052.
- [7] Luk CK, Chesi G. Guaranteed estimates of the domain of attraction for a class of hybrid systems. *International Journal of Robust and Nonlinear Control* 2015; **25**(17):3270–3285.
- [8] Chesi G. SMRSOFT: a matlab toolbox for optimization over polynomials and dynamical systems study via sos programming. <http://www.eee.hku.hk/~chesi> 2011.
- [9] Jansson C, Chaykin D, Keil C. Rigorous error bounds for the optimal value in semidefinite programming. *SIAM Journal on Numerical Analysis* 2008; **46**(1):180–200.
- [10] Delanoue N, Jaulin L, Cottencau B. An algorithm for computing a neighborhood included in the attraction domain of an asymptotically stable point. *Communications in Nonlinear Science and Numerical Simulation* 2015; **21**(1):181–189.
- [11] Ratschan S, She Z. Providing a basin of attraction to a target region of polynomial systems by computation of lyapunov-like functions. *SIAM Journal on Control and Optimization*, *48*(7) 2010; :4377–4394.
- [12] Neumaier A. *Interval methods for systems of equations*. Cambridge university press, 1990.
- [13] Jaulin L, Kieffer M, Didrit O, Walter E. *Applied Interval Analysis*. Springer, 2001.
- [14] Goualard F. *GAOL 3.1.1: Not Just Another Interval Arithmetic Library*. Laboratoire d'Informatique de Nantes-Atlantique, 4.0 edn. Oct 2006.
- [15] Rump SM. INTLAB - INTerval LABoratory. *Developments in Reliable Computing*, Csentesi T (ed.). Kluwer Academic Publishers: Dordrecht, 1999; 77–104.
- [16] Hansen E. *Topics in Interval Analysis*. Oxford University Press, 1969.
- [17] Clarke FH. *Optimization and Nonsmooth Analysis*. Society for Industrial and Applied Mathematics: Philadelphia, PA, 1990.
- [18] Rump SM. Expansion and estimation of the range of nonlinear functions. *Mathematics of Computation* 1996; **65**(216):1503–1512.
- [19] Wilczak D, Zgliczyski P. Connecting orbits for a singular nonautonomous real ginzburg–landau type equation. *SIAM Journal on Applied Dynamical Systems* 2016; **15**(1):495–525.
- [20] Rohn J. Positive definiteness and stability of interval matrices. *SIAM Journal on Matrix Analysis and Applications* 1994; **15**(1):175–184.
- [21] Domes F, Neumaier A. Rigorous enclosures of ellipsoids and directed cholesky factorizations. *SIAM Journal on Matrix Analysis and Applications* 2011; **32**(1):262–285.
- [22] Domes F, Neumaier A. Linear and parabolic relaxations for quadratic constraints. *Journal of Global Optimization* 2016; **65**(3):457–486.
- [23] Chabert G, Jaulin L. Contractor Programming. *Artif. Intell.* 2009; **173**(11):1079–1100.
- [24] Araya I, Trombettoni G, Neveu B, Chabert G. Upper Bounding in Inner Regions for Global Optimization Under Inequality Constraints. *J. Global Optimization* 2014; **60**(2):145–164.
- [25] Chabert G. IBEX: a C++ library for constraint processing over real numbers. <http://www.ibex-lib.org/> 2011.
- [26] Lin Y, Stadtherr MA. Validated solution of ODEs with parametric uncertainties. *ESCAPE 16 and PSE 9, Computer Aided Chemical Engineering*, vol. 21, Marquardt W, Pantelides C (eds.). Elsevier, 2006; 167–172.
- [27] Kailath T. *Linear systems*. Prentice-Hall, 1980.
- [28] Goldsztejn A, Jermann C, de Angulo VR, Torras C. Variable Symmetry Breaking in Numeric Constraint Problems. *Artificial Intelligence* 2015; **229**:105–125.

## A Slope matrix enclosure using intlab slope arithmetic

Note: INTLAB [15] does provide slopes neither for min nor max function, which have therefore been implemented using the function abs in the following MATLAB code. This is non-optimal, specific slope expressions for these functions would certainly improve the EDA.

```

Min=@(a,b)0.5*(-abs(a-b)+a+b);
Max=@(a,b)0.5*(abs(a-b)+a+b);
f=@(x)[x(2);x(1)-Max(-5,Min(5,13*x(1)
+7*x(2)))]];

P=[1.22 0.0417 ; 0.0417 0.0774];
c=0.22^2;
xi=midrad([0;0],sup(sqrt(intval(c))*sqrt(
diag(inv(intval(P))))));
xs=slopeinit(mid(xi),xi);
fs=f(xs);
J=fs.s;
Q=P*J+transpose(J)*P;
mQ=-Q;
display(mQ(1,1));
display(mQ(1,1)*mQ(2,2)-mQ(1,2)*mQ(2,1));

```

The two displayed determinants are positive, therefore Sylvester criterion proves that the interval matrix  $-Q$  is definite positive. The test fails for  $c \geq 0.23^2$ .