Local stability analysis for large polynomial spline systems

Torbjørn Cunis, Jean-Philippe Condomines, Laurent Burlion

Local stability analysis for large polynomial spline systems ⋆

Torbjørn Cunis [a,b], Jean-Philippe Condomines [b], Laurent Burlion [c]

[a] *ONERA – The French Aerospace Lab, 2 avenue Edouard Belin, 31055 Toulouse, France*

[b] *ENAC, Université de Toulouse, 7 avenue Edouard Belin, 31055 Toulouse, France*

[c] *Rutgers, The State University of New Jersey, 98 Brett Road, Piscataway, NJ 08854, USA*

Abstract

Polynomial switching systems such as multivariate splines provide accurate fitting while retaining an algebraic representation and offering arbitrary degrees of smoothness; yet, application of sum-of-squares techniques for local stability analysis is computationally demanding for a large number of subdomains. This communiqué presents an algorithm for region of attraction estimation that is confined to those subdomains actually covered by the estimate, thereby significantly reducing computation time. Correctness of the results is subsequently proven and the run time is approximated in terms of the number of total and covered subdomains. Application to longitudinal aircraft motion concludes the study.

Key words: Nonlinear analysis; stability analysis; switching functions; polynomial methods; Lyapunov function.

1 Introduction

Recently, several works on polynomial fitting have been led and provide a constructive method for determining models based on analytical computation due to their continuous and differentiable nature. Computing the exact region of attraction for this kind of nonlinear dynamics is very hard if not impossible. Therefore, researchers have focused on determining polynomial Lyapunov functions for polynomial systems building upon sum-of-squares [1–3] including extensions to rational and composite Lyapunov-functions [4–6]. However, when polynomials are unsuitable to represent system dynamics, piecewise defined polynomials such as splines [7] provide more tractable models, requiring extended effort when determining local stability: it is well known for example, that stability of the subsystems does not guarantee stability of the entire system [8]. Therefore, approximation techniques have been developed for the estimation of the region of attraction of piecewise systems. Early work was limited to a priori given, multiple quadratic Lyapunov functions [9] and could only provide very rough estimates of the region of attraction. In [10] though, analysis of polynomial fuzzy models is considered employing again composite Lyapunov functions; taking each the point-wise extremum, this approach provides directly a continuous function. It is worth noting that this might come at the cost of a large number of decision variables and that there are no relaxations with respect to the respectively active subdomains. [11] proposes a further approach using multiple Lyapunov functions for switching hybrid systems with polyhedral subdomains which share a boundary in the origin. Sum-of-squares complexity was discussed [12]; where subdomains are considered for stability, the complexity increases with the number of bounding constraints.

The present paper focuses on a new formulation of the region of attraction estimation for large piecewise systems of local polynomial dynamics and polynomial domains, such as switching systems and multivariate splines, within the sum-of-squares framework. We present preliminary results and extend previous work to switching systems (in Sec. 2). The main result, an algorithm for splines, is discussed in Sec. 3, and is applied to an engineering example in Sec. 4. The appendix illustrates an extension to multiple Lyapunov functions. For the implementation of the constraints, in particular the polynomial containment problem (Lemma 1), we rely on the semidefinite programming techniques of [1, 13]. A concise discussion of the $\mathcal{V}$-s-iteration is given in [14].

2 Preliminaries

Consider the autonomous system $\dot{x} = f(x)$ given by the $k \in \mathbb{N}$ ordinary differential equations

$$\dot{x} = f_i(x)\,, \qquad \text{if } x \in \Phi_i \tag{1}$$

⋆ This paper was not presented at any IFAC meeting. Corresponding author T. Cunis.

Email addresses: tcunis@umich.edu (Torbjørn Cunis), jean-philippe.condomines@enac.fr (Jean-Philippe Condomines), laurent.burlion@rutgers.edu (Laurent Burlion).

for $1 \le i \le k$, where $x \in \mathbb{R}^n$, $f_i \in \mathbb{R}[x]^n$, and $\Phi_1, \ldots, \Phi_k$ are intersections of polynomial inequalities $\Omega_{\varphi \le x_0} =_{\text{def}} \{x \in \mathbb{R}^n \mid \varphi(x) \le x_0\}$ with $\varphi \in \mathbb{R}[x]$, $x_0 \in \mathbb{R}^n$, forming a set partition of $\mathbb{R}^m$. [1]

Notation The interior, boundary, and closure of $\mathcal{A} \subseteq \mathbb{R}^m$ are notated by int$\mathcal{A}$, $\partial\mathcal{A}$, and cl$\mathcal{A}$, respectively. $\mathcal{A}^* =_{\text{def}} \mathcal{A} - \{0\}$. The set of sum-of-squares polynomials is notated by $\Sigma[x]$.

Assumptions $f(0) = 0$, i.e., the origin is a stationary point of f.

Lemma 1 *Let $p, q_1, \ldots, q_k \in \mathbb{R}[x]$; we have*

$$\Omega_{q_1 \le 0} \cap \cdots \cap \Omega_{q_k \le 0} \subseteq \Omega_{p \le 0}$$

if there exist $s_1, \ldots, s_k \in \Sigma[x]$ such that $\sum_{i=1}^k s_i q_i - p \in \Sigma[x]$.

Lemma 1 is an explicit formulation of [1, Lemma 2]. We then write with $\mathbf{s} = (s_1, \ldots, s_k)$,

$$\mathbf{s} \vdash \bigcap_i \Omega_{q_i \le 0} \subseteq_\Sigma \Omega_{p \le 0} \tag{2}$$

and say "$\mathbf{s}$ proves" the inclusion. We also say that "q_i, p solve" the inclusion.

Problem 2 *Let $p \in \Sigma[x]$; solve the optimisation*

$$\max_{\substack{\beta^\diamond, \gamma^\diamond \in \mathbb{R} \\ \mathcal{V} \in \mathbb{R}[x]}} \beta^\diamond \quad \text{such that}$$

$$\Omega_{p \le \beta^\diamond} \subseteq \Omega_{\mathcal{V} \le \gamma^\diamond} \tag{3}$$

$$\Omega^*_{\mathcal{V} \le \gamma^\diamond} \subseteq \{x \mid \nabla\mathcal{V}(x)\, f(x) < 0\} \tag{4}$$

with $\mathcal{V}(\cdot)$ positive definite [2] *and $\mathcal{V}(x) = 0$.*

If $(\mathcal{V}, \beta^\diamond, \gamma^\diamond)$ solve Problem 2 for p, then $\Omega_{\mathcal{V} \le \gamma^\diamond}$ is an invariant subset of the region of attraction [1, Lemma 1] and admits the largest inscribing region $\Omega_{p \le \beta^\diamond}$. Simultaneously searching for an optimal function $\mathcal{V}(\cdot)$ while proving invariance of $\Omega_{\mathcal{V} \le \gamma^\diamond}$ involves bilinear terms. If the degree of $\mathcal{V} \in \mathbb{R}[x]$ is restricted and $f \in \mathbb{R}[x]$, the $\mathcal{V}$-s-iteration [1, 14] solves Problem 2 by alternating-iteratively searching for $\mathcal{V}$, $\gamma^\diamond$, and $\beta^\diamond$ maximal such that s_1, s_2 prove (3) and (4).

Remark 3 *The sum-of-squares formulation is limited to nonnegativity; we thus make use of that $p(x) < 0$ if $p(x) \le -\epsilon\, |x|_2^2$ for $p \in \mathbb{R}[x]$, $x \ne 0$, and $\epsilon > 0$.*

[1] $\mathcal{A}_1, \ldots, \mathcal{A}_k$ form a *set partition* of a body $\mathbb{K}$ if and only if they are pairwise interior-disjunct and $\bigcup_i \mathcal{A}_i = \mathbb{K}$.

[2] A continuous function φ is said to be *positive definite* (p.d.) if $\varphi(\cdot) > 0$ everywhere except the origin and $\varphi(0) = 0$.

We write $\Omega_{\mathcal{V},f,\epsilon} =_{\text{def}} \{x \mid \nabla\mathcal{V} f(x) \le -\epsilon x^T x\}$ for $\mathcal{V} \in \mathbb{R}[x]$, $f \in \mathbb{R}[x]^m$, and $\epsilon > 0$.

If there exists a single function $\mathcal{V}: \mathbb{R}^m \to \mathbb{R}$ p.d. such that $\nabla\mathcal{V} f_i(x) < 0$ for all $x \in \mathcal{A}^*$ and all $1 \le i \le k$, then $\mathcal{A}$ is also invariant for the piecewise system and $\mathcal{V}$ is a Lyapunov function of the switching dynamics. However, this requirement is unnecessary strict when it comes to the subsystems that are *not* active [8]; indeed, the following suffices:

Corollary 4 *Let $\mathcal{V}: \mathbb{R}^m \to \mathbb{R}$ be continuous p.d. with $\mathcal{V}(0) = 0$ and $\mathcal{A} = \Omega_{\mathcal{V} \le \alpha}$ for some $\alpha \in \mathbb{R}$; if*

$$\forall x \in \mathcal{A}^*. \quad (x \in \Phi_i \Rightarrow \nabla\mathcal{V} f_i(x) < 0) \tag{5}$$

for all $1 \le i < j \le k$, then $\mathcal{A} = \bigcup_i (\mathcal{A} \cap \Phi_i)$ is invariant.

Lemma 1 encodes (5) into a polynomial sum-of-squares problem, recalling Φ_i is an intersection of polynomial inequalities. Now, $\bigcup_i (\Omega_{\mathcal{V} \le \gamma^\diamond} \cap \Phi_i)$ is invariant if

$$\mathbf{s}_{2,i} \vdash \Omega^*_{\mathcal{V} \le \gamma_i} \cap \Phi_i \subseteq_\Sigma \Omega_{\mathcal{V}, f_i, \epsilon}, \tag{6}$$

$\mathbf{s}_{2,i} \subset \Sigma[x]$, for all $1 \le i \le k$ and $\gamma^\diamond = \min\{\gamma_1, \ldots, \gamma_k\}$.

Remark 5 *The idea of Corollary 4, and of the paper, can be extended to piecewise defined functions*

$$\mathcal{V}(x) = \mathcal{V}_i(x), \qquad \text{if } x \in \Phi_i,$$

for $1 \le i \le k$. We further illustrate this in the appendix.

3 Spline systems

While we have not taken further assumptions on the Φ_i, in the present literature the invariant set is commonly assumed to cover all domains. [3] For large spline systems with bounded domains and only local stability, each subdomain taken into account whilst not part of the invariant set adds an inactive boundary to the computational load. We therefore present an adapted algorithm to efficiently compute a region of attraction estimation for spline systems.

Definition 6 *A* spline system *is a triple $\mathcal{SP} = (\mathcal{I}, f_{(\cdot)}, \mathcal{E})$, where $\mathcal{I} \subset \mathbb{N}$ are the domains, $f: \mathcal{I} \times \mathbb{R}^m$ are the piecewise nonlinear dynamics, and $\mathcal{E}: \mathcal{I} \times \mathcal{I} \to \{\mathbb{R}^m \to \mathbb{R}\}$ is the weighted switching relation, where the dynamics switch from f_i to f_j with $i, j \in \mathcal{I}$ for $x \in \mathbb{R}^m$ if and only if $h_{ij} = \mathcal{E}(i,j)$ is defined and $h_{ij}(x) > 0$.* [4]

[3] Either by looking for global stability [15] or choosing boundaries crossing the origin [9, 11].

[4] For a spline structure to behave like a spline, we tacitly understand that $\mathcal{E}(\cdot,\cdot)$ is irreflexive as well as symmetric with $h_{ij} = \mathcal{E}(i,j) = -h_{ji}$ if defined for $i, j \in \mathcal{I}$.

We define some further notation: for $i \in \mathcal{I}$, let $\mathrm{adj}[i] =_{\mathrm{def}} \{ j \,|\, h_{ij} = \mathcal{E}(i,j) \text{ is defined} \}$ be the set of adjacent domains of Φ_i, that is, $\Phi_i = \bigcap_{j \in \mathrm{adj}[i]} \{ x \,|\, h_{ij}(x) \leq 0 \}$; $\mathcal{V}_K(\cdot)$ denotes the function-candidate in the K-th iteration and $I_K \subset \mathcal{I}$ will denote the set of domains covered by the invariant set $\Omega_{\mathcal{V}_K \leq \gamma^\diamond}$; for $i' \in \mathcal{I}$, $\mathrm{dist}_K[i']$ is the distance of $\Phi_{i'}$ with respect to $\mathcal{V}_K$, defined as

$$\mathrm{dist}_K[i'] = \sup\{ \gamma \,|\, \Omega_{\mathcal{V}_K \leq \gamma} \cap \Phi_{i'} = \varnothing \} \,; \tag{7}$$

at last, we extend $\mathrm{adj}[\cdot]$ to $2^{\mathcal{I}}$ with $\mathrm{adj}[I] =_{\mathrm{def}} \bigcup_{i \in I} \mathrm{adj}\,[i] - I$ for $I \subset \mathcal{I}$.

With this notation, we can state Algorithm 1 computing the optimal estimate $\Omega_{\mathcal{V} \leq \gamma^\diamond}$ for a spline structure: If, for any iteration K, the invariant set $\Omega_{\mathcal{V}_K \leq \gamma^\diamond}$ is contained in the subdomains I_K, it suffices to check invariance only of these subdomains and with respect to the boundaries in between, instead of proving Eq. (6) for *all* $i \in \mathcal{I}$. Now, in order to examine whether an $i' \in \mathcal{I}$ is covered by the optimal invariant set of $\mathcal{V}_K$, we preliminary compute the invariant set for some $I' \subset \mathcal{I} - \{i'\}$ and evaluate the distance of i'; only if i' is “closer” than the boundary of the preliminary invariant set, $i' \in I_K$.

The algorithm consists of three cascaded loops; the outer **for** loop over K of the basic $\mathcal{V}$-s-iteration (“K-iteration”), an inner **repeat-until** loop determining I_K (“I_K-loop”), and inner-most **for** loops over the elements of I_K. While the restriction to I_K in line 8 reduces the problem size, the I_K-loop itself adds to the run time. On the other hand, if $\mathcal{V}_K$ and $\mathrm{adj}[i] \cap I_K$ remain unchanged, Eq. $(*_i)$ yields the same value γ_i; that is, after each i_{next} added to I_K, it suffices to re-execute line 8 for any $i \in I_K \cap \mathrm{adj}[i_{\mathrm{next}}]$. We refer to the thus modified algorithm as Algorithm 1b.

Proposition 7 *After each repetition of the K-iteration in Algorithm 1(b), the following hold:*

(1) $\Omega_{\mathcal{V}_K \leq \gamma^\diamond} \subset \bigcup_{i \in I_K} \Phi_i$;
(2) $\Omega_{\mathcal{V}_K \leq \gamma^\diamond}$ is invariant.

PROOF. After each iteration of the I_K-loop, we have that

$$\forall i \in I_K \quad \Omega^*_{\mathcal{V}_K \leq \gamma_{\mathrm{pre}}} \cap \Phi_i \subseteq \{ x \,|\, \nabla \mathcal{V} f_i(x) < 0 \} \,; \tag{8}$$

as $\Phi_i \subseteq \bigcap_{j \in I} \Omega_{h_{ij} \leq 0}$ for any $I \subset \mathrm{adj}[i]$, $\mathrm{adj}[i] \cap I_K \subset \mathrm{adj}[i]$, $\gamma_{\mathrm{pre}} \leq \gamma_i$, and $\mathbf{s}_{2,i}$ proves $(*_i)$, [5] for all $i \in I_K$; and

$$\Omega_{\mathcal{V}_K \leq \gamma_{\min}} \subset \bigcup_{i \in I_K} \Phi_i ; \tag{9}$$

as $\gamma_{\min} \leq \mathrm{dist}_K[i']$ for $i' \notin I_K$ and $(\ddagger_{i'})$ implies (7). Since $\gamma_{\mathrm{pre}} < \gamma_{\min}$ for termination of the I_K-loop, (9) implies $\Omega_{\mathcal{V}_K \leq \gamma^\diamond} \subset \bigcup_{i \in I_K} \Phi_i$ with $\gamma^\diamond = \gamma_{\mathrm{pre}}$. Thus, $\Omega_{\mathcal{V}_K \leq \gamma^\diamond} \cap \Phi_i = \varnothing$ for all $i \in \mathcal{I} - I_K$ [6] and therefore, $\Omega_{\mathcal{V}_K \leq \gamma^\diamond} = \bigcup_{i \in \mathcal{I}} (\Omega_{\mathcal{V}_K \leq \gamma^\diamond} \cap \Phi_i)$ is invariant by Corollary 4 for (8) holds. $\square$

Algorithm 1 Estimate invariant set $\Omega_{\mathcal{V} \leq \gamma^\diamond} \subset \bigcup_{i \in I} \Phi_i$ with $I = I_{K_{\max}}$ and $\mathcal{V} = \mathcal{V}_{K_{\max}}$.

1: **for** $K = 1$ to $K_{\max}$
2:   **if** $K > 1$ **then**
3:     find $\mathcal{V}_K$ p.d. solving

$$s_1 \vdash \Omega_{p \leq \beta^\diamond} \subseteq_\Sigma \Omega_{\mathcal{V}_K \leq \gamma^\diamond} \tag{††}$$

$$\forall i \in I_K.\ \mathbf{s}_{2,i} \vdash \Omega_{\mathcal{V}_K \leq \gamma_i} \cap \bigcap_{j \in \mathrm{adj}[i] \cap I_K} \Omega_{h_{ij} \leq 0} \subseteq_\Sigma \Omega_{\mathcal{V}_K, f_i, \epsilon} \tag{$**$}$$

4:   **end**
5:   $I_K := I_{K-1}$
6:   **repeat**
7:     **for** $i \in I_K$
8:       find $\gamma_i := \max_{\gamma \geq 0} \gamma$ s.t. $\mathbf{s}_{2,i} \subset \Sigma\,[x]$ solves

$$\mathbf{s}_{2,i} \vdash \Omega_{\mathcal{V}_K \leq \gamma} \cap \bigcap_{j \in \mathrm{adj}[i] \cap I_K} \Omega_{h_{ij} \leq 0} \subseteq_\Sigma \Omega_{\mathcal{V}_K, f_i, \epsilon} \tag{$*_i$}$$

9:     **end**
10:     $\gamma_{\mathrm{pre}} := \min\{ \gamma_i \,|\, i \in I_K \}$
11:     **for** $i' \in \mathrm{adj}[I_K]$
12:       compute $\mathrm{dist}_K[i']$ as

$$\min_{\substack{\gamma \geq 0, x \\ \mathcal{V}_K(x) = \gamma}} \gamma \text{ s.t. } \bigwedge_{j \in \mathrm{adj}[i']} h_{i'j}(x) \leq 0 \tag{$\ddagger_{i'}$}$$

13:     **end**
14:     $\gamma_{\min} := \min_{i' \in \mathrm{adj}[I_K]} \mathrm{dist}_K[i']$
15:     **if** $\gamma_{\mathrm{pre}} \geq \gamma_{\min}$ **then**
16:       $i_{\mathrm{next}} := \arg\min_{i' \in \mathrm{adj}[I_K]} \mathrm{dist}_K[i']$
17:       $I_K := I_K \cup \{ i_{\mathrm{next}} \}$
18:     **end**
19:   **until** $I_K = \mathcal{I}$ or $\gamma_{\mathrm{pre}} < \gamma_{\min}$
20:   $\gamma^\diamond := \gamma_{\mathrm{pre}}$
21:   **for** $i \in I_K$
22:     find $\beta^\diamond := \max_{\beta \geq 0}$ s.t. $s_1 \in \Sigma\,[x]$ solves

$$s_1 \vdash \Omega_{p \leq \beta} \subseteq_\Sigma \Omega_{\mathcal{V}_K \leq \gamma^\diamond} \tag{†}$$

23:   **end**
24: **end**

[5] Using $\mathcal{A}_1 \cap \mathcal{A}_2 \subseteq \mathcal{A}'_1 \cap \mathcal{A}'_2$ for $\mathcal{A}_{1,2} \subseteq \mathcal{A}'_{1,2}$.

[6] Assuming that $\mathrm{int}\Phi_{i,j}$ are disjunct if $i \neq j$.

As all loops are limited by either the number of elements in $\mathcal{I}$ or $K_{\max}$, Algorithm 1b terminates and $\Omega_{\mathcal{V} \leq \gamma^\diamond}$ is an invariant set of the given spline system. Initially, we have $I_0 = \{i \in \mathcal{I} \,|\, 0 \in \Phi_i\}$, and, assuming a singleton $I_0 = \{i_0\}$, $\mathcal{V}_1 = x^T P x$ as solution to the polynomial Lyapunov equation for f_{i_0}.[7]

Asymptotic run time estimation In order to compare the run time of the proposed approach for splines to the basic approach of the previous section, we count the total number of executions of line 8 and the number of decision variables $\mathbf{s}_{2,i}$ involved each time.[8] Here, we assume that the spline structure $\mathcal{SP}$ has k subdomains; every domain has (in average) M adjacent cells; the resulting invariant set covers R subdomains; and the number of iterations is chosen as $K_{\max} = R$. Consider now the following, distinct cases: in the *worst case*, the initial invariant set $\Omega_{\mathcal{V}_1 \leq \gamma^\diamond}$ covers all R subdomains; whereas in the *average case*, $\Omega_{\mathcal{V}_K \leq \gamma^\diamond}$ grows in each repetition of the K-iteration into one further subdomain i_{next}. In both cases, I_0 is taken as singleton.

Clearly, the asymptotic run time of the basic approach in both cases is equivalent to $T_M^{\text{basic}}(k) = R\,k(M+1)$. Algorithm 1b, in the worst case, repeats line 8 in the first iteration for R times, afterwards once each iteration: proving $(*_i)$ requires a single decision variable the first time ($\operatorname{adj}[i_0] \cap \{i_0\} = \varnothing$) and $m^* + 1$ decision variables from there on, where m^* is the number of adjacent domains of i in I_1 and each repetition of the I_K-loop adds one i_{next}; in the following $R-1$ iterations, line 8 is executed R times each with $M+1$ decision variables; that is,

$$T_M^{\text{worst}}(R) = 1 + R \sum_{m^*=1}^{M} (m^* + 1) + (R-1)\,R\,(M+1)\,.$$

In the average case, Algorithm 1b executes line 8 once in every iteration for each of the r^* domains in I_K with $m^* \leq M$ decision variables; additionally, since $I_{K+1} = I_K \cup \{i_{\text{next}}\}$, line 8 is executed for every adjacent subdomain of i_{next} in I_K (i.e., at most M times) with $m^* + 1$ decision variables; that is,

$$T_M^{\text{avg}}(R) = 1 + \sum_{r^*=2}^{R} r^*\,(M+1) + R \sum_{m^*=1}^{M} (m^* + 1)\,.$$

If $M = 3$ ($M = 4$),[9] the worst-case asymptotic run time is less than T_M^{basic} for $R < 0.987\,k$ ($R < 0.981\,k$) and the average-case is less than $\frac{1}{2}T_M^{\text{basic}}$ for $R < 0.944\,k$ ($R < 0.933\,k$).

4 Application example

The short-period motion of a transport aircraft might be given as autonomous system

$$\dot{x}_1 = x_2 \tag{10}$$
$$\dot{x}_2 = f_{\text{M}}(x_1, x_2)\,, \tag{11}$$

where x_1 is the angle of attack, x_2 is the pitch rate, and $f_{\text{M}}(\cdot)$ is a 3rd-order piecewise polynomial model of the aerodynamic pitch coefficient defined as 5-by-5 rectangular spline (boundaries depicted in Fig. 1) with $f_{\text{M}}(0,0) = 0$.[10] After scaling the system to $\tilde{x} = \mathbf{D}x$ with $\mathbf{D} \in \mathbb{R}^{2\times 2}$, we compute the invariant set of the origin using Algorithm 1b. Such a problem has been discussed in [14] for a polynomial model and in [17] for a once-piecewise polynomial model.

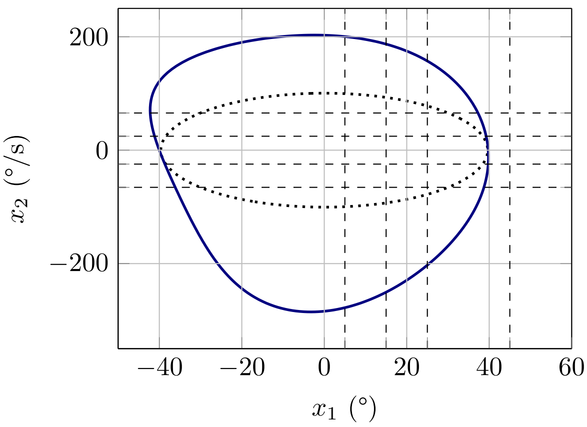

Fig. 1. Invariant set after $K_{\text{fin}} = 124$ iterations for aircraft short-period motion (solid: invariant set $\Omega_{\mathcal{V}_K \leq \gamma^\diamond}$; dotted: inscribing ellipsoid $\Omega_{p \leq \beta^\diamond}$; dashed: subdomain boundaries).

Algorithm 1b finds the optimal invariant set shown in Fig. 1, covering 20 of 25 domains, after 124 repetitions of the K-iteration and a run time equivalent of $T = 10\,098$. The basic approach runs the same number of repetitions with a run time equivalent of $T^{\text{basic}} = 13\,020$. In each repetition of the K-iteration, line 8 is executed multiple times subject to both the elements already in I_{K-1} and those added to I_K during the inner I_K-loop. The number of linear matrix inequality (LMI) variables to solve the sum-of-squares problems $(*_i)$ then varies with the number of elements in $\operatorname{adj}[i] \cap I_K$. Table 1 gives details of the computations, including the number of

[7] If I_0 is *not* a singleton, P can be found as solution to

$$A_i^T P + P A_i < \sum_{j \in \operatorname{adj}[i] \cap I_0} h_{ij}(x)\ \forall i \in I_0$$

with $A_i = \partial f_i / \partial x$.

[8] Line 8, being inside of all three loops, is the major difficulty if line 12 is efficiently computed using MATLAB's `fmincon` or a similar numerical method.

[9] $M = 3$ and $M = 4$ are tantamount to planar systems with triangular and rectangular domains, respectively.

[10] See [16] for details.

LMI variables in $(*_i)$ *averaged* for each repetition K, comparing Algorithm 1b and the basic approach. [11]

Table 1
Details of the application example: number of elements in $I_K \subseteq \mathcal{I}$; run time equivalent; number of executions of line 8; and average number of LMI variables.

	Alg. 1b				basic			
K	$\#I_K$	T_K	#l8	LMI	$\#I_K$	T_K	#l8	LMI
1	12	86	28	208	25	105	25	250
2	12	44	12	356	25	105	25	378
3	13	56	15	359	25	105	25	378
4	14	60	16	360	25	105	25	378
5	15	65	17	363	25	105	25	378
6	20	111	29	365	25	105	25	378
7	20	86	20	374	25	105	25	378
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
K_{fin}	20	86	20	374	25	105	25	378

5 Conclusion

Extensions of sum-of-squares techniques such as the $\mathcal{V}$-s-iteration for piecewise polynomial systems quickly grow infeasible for large systems, which spline models embody. In this article, we therefore presented an adapted algorithm for splines, relaxing the problem to the subdomains that are actually covered by the region of attraction estimate. We have proven correctness of our approach and demonstrated a *worst case* run time superior to the basic approach for regions of attractions spanning more than 90 % of the subdomains of a regular planar spline system. While this ratio will shrink for higher dimensions, so does the number of subdomains increase.

Acknowledgements

The implementation of Algorithm 1 utilises the `SOSOPT` / `SOSAnalysis` toolboxes of the University of Minnesota.

References

[1] Ufuk Topcu, Andrew Packard, and Peter Seiler. Local stability analysis using simulations and sum-of-squares programming. *Automatica*, 44(10):2669–2675, 2008. doi: 10.1016/j.automatica.2008.03.010.

[2] Ufuk Topcu, Andrew K. Packard, Peter Seiler, and Gary J. Balas. Robust Region-of-Attraction Estimation. *IEEE Transactions on Automatic Control*, 55(1):137–142, jan 2010. doi: 10.1109/TAC.2009.2033751.

[3] Larissa Khodadadi, Behzad Samadi, and Hamid Khaloozadeh. Estimation of region of attraction for polynomial nonlinear systems: A numerical method. *ISA Transactions*, 53(1):25–32, 2014. doi: 10.1016/j.isatra.2013.08.005.

[4] Giorgio Valmorbida and James Anderson. Region of attraction estimation using invariant sets and rational Lyapunov functions. *Automatica*, 75:37–45, 2017. doi: 10.1016/j.automatica.2016.09.003.

[5] Weehong Tan and Andrew Packard. Stability region analysis using polynomial and composite polynomial Lyapunov functions and sum-of-squares programming. *IEEE Transactions on Automatic Control*, 53(2):565–571, 2008. doi: 10.1109/TAC.2007.914221.

[6] Graziano Chesi. Rational Lyapunov functions for estimating and controlling the robust domain of attraction. *Automatica*, 49(4):1051–1057, 2013. doi: 10.1016/j.automatica.2013.01.032.

[7] Coen C. de Visser, Q. P. Chu, and J. A. Mulder. A new approach to linear regression with multivariate splines. *Automatica*, 45(12):2903–2909, 2009. doi: 10.1016/j.automatica.2009.09.017.

[8] Mikael Johansson and Anders Rantzer. Computation of piecewise quadratic Lyapunov functions for hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):555–559, 1998. doi: 10.1109/9.664157.

[9] F. Amato, R. Ambrosino, G. De Tommasi, and A. Merola. Estimation of the domain of attraction for a class of hybrid systems. *Nonlinear Analysis: Hybrid Systems*, 5(3):573–582, 2011. doi: 10.1016/j.nahs.2010.12.005.

[10] Ying-Jen Chen, Motoyasu Tanaka, Kazuo Tanaka, and Hua O. Wang. Stability Analysis and Region-of-Attraction Estimation Using Piecewise Polynomial Lyapunov Functions: Polynomial Fuzzy Model Approach. *IEEE Transactions on Fuzzy Systems*, 23(4):1314–1322, 2015. doi: 10.1109/TFUZZ.2014.2347993.

[11] Xiuliang Zheng, Zhikun She, Junjie Lu, and Meilun Li. Computing multiple Lyapunov-like functions for inner estimates of domains of attraction of switched hybrid systems. *International Journal of Robust and Nonlinear Control*, 2018. doi: 10.1002/rnc.4280.

[12] Graziano Chesi. On the Complexity of SOS Programming and Applications in Control Systems. *Asian Journal of Control*, 20(5):2005–2013, 2018. doi: 10.1002/asjc.1684.

[13] Pablo A. Parillo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming, Series B*, 96(2):293–320, 2003. doi: 10.1007/s10107-003-0387-5.

[14] Abhijit Chakraborty, Peter Seiler, and Gary J. Balas. Nonlinear region of attraction analysis for flight control verification and validation. *Control Engineering Practice*, 19 (4):335–345, 2011. doi: 10.1016/j.conengprac.2010.12.001.

[15] Antonis Papachristodoulou and Stephen Prajna. Robust Stability Analysis of Nonlinear Hybrid Systems. *IEEE Transactions on Automatic Control*, 54(5):1035–1041, 2009. doi: 10.1109/TAC.2009.2017155.

[16] Torbjørn Cunis, Laurent Burlion, and Jean-Philippe Condomines. Piecewise Polynomial Model of the Aerodynamic Coefficients of the Generic Transport Model and its Equations of Motion. Technical Report `hal-01808649`, ONERA – The French Aerospace Lab; French Civil Aviation School, Toulouse, FR, 2018. URL https://hal.archives-ouvertes.fr/hal-01808649. Appendix A with Eq. (104): $f_{\mathrm{M}}(\cdot) = \dot{q}$ for the trim condition $V_{\mathrm{A}} = 77.17\,\mathrm{m\,s^{-1}}$, $\gamma_{\mathrm{A}} = 0$, $q = 0$, $\alpha = 3.65°$, and $\eta = 2.14°$, $F = 29.81\,\mathrm{N}$.

[17] Torbjørn Cunis, Laurent Burlion, and Jean-Philippe Condomines. Piecewise Polynomial Modeling for Control and Analysis of Aircraft Dynamics beyond Stall. *Journal of Guidance, Control, and Dynamics*, 42(4):949–957, 2019. doi: 10.2514/1.G003618.

[11] The basic approach corresponds to Alg. 1 with $I_K \equiv \mathcal{I}$.

A Appendix

The proposed algorithm can be modified for multiple Lyapunov functions:

Corollary 8 *Let* $\mathcal{V}_i : \mathbb{R}^m \to \mathbb{R}$ *be continuous p.d. with* $\mathcal{V}_i(0) = 0$ *and* $\mathcal{A}_i = \Omega_{\mathcal{V}_i \le \alpha}$ *for all* $1 \le i \le k$ *and some* $\alpha \in \mathbb{R}$*; if*

$$\forall x \in \mathcal{A}_i^*. \quad (x \in \Phi_i \Rightarrow \nabla \mathcal{V}_i f_i(x) < 0) \tag{A.1}$$
$$\forall x \in \partial\Phi_i \cap \partial\Phi_j. \quad \mathcal{V}_i(x) = \mathcal{V}_j(x) \tag{A.2}$$

for all $1 \le i < j \le k$*, then* $\mathcal{A} = \bigcup_i (\mathcal{A}_i \cap \Phi_i)$ *is invariant.*

Then, Eq. (3) holds if

$$s_{1,i} \vdash \Omega_{p \le \beta_i} \subseteq_\Sigma \Omega_{\mathcal{V}_i \le \gamma^\diamond}, \tag{A.3}$$

$s_{1,i} \in \Sigma[x]$, for all $1 \le i \le k$ and $\beta^\diamond = \min\{\beta_1, \dots, \beta_k\}$.

The continuity condition (A.2) cannot be represented by sum-of-squares immediately. Papachristodoulou and Prajna [15] suggested to add an equality constraint for each polynomial boundary, however leading to increased conservativeness for large systems.

Proposition 9 *Eq.* (A.2) *holds if and only if*

$$\forall x \in \mathrm{cl}\Phi_i \cap \mathrm{cl}\Phi_j. \quad \mathcal{V}_i(x) \le \mathcal{V}_j(x) \tag{A.4}$$

for Φ_i, Φ_j *pairwise disjunct.*

PROOF. Follows directly from $\mathrm{cl}\Phi_i \cap \mathrm{cl}\Phi_j = (\partial\Phi_i \cap \partial\Phi_j) \cup (\mathrm{int}\Phi_i \cap \mathrm{int}\Phi_j)$ with $\mathrm{int}\Phi_i \cap \mathrm{int}\Phi_j = \varnothing$ and $(v_i \le v_j) \wedge (v_j \le v_i) \Leftrightarrow v_i = v_j$. $\square$

In line 3 of Algorithm 1(b), Eq. (A.4) holds for $\mathcal{V}_i, \mathcal{V}_j \in \mathbb{R}[x]$ p.d. if

$$\mathbf{r}_{ij} \vdash \bigcap_{a \in \mathrm{adj}[i] \cap I_K} \Omega_{h_{ia} \le 0} \cap \bigcap_{b \in \mathrm{adj}[j] \cap I_K} \Omega_{h_{jb} \le 0} \subseteq_\Sigma \Omega_{\mathcal{V}_i \le \mathcal{V}_j} \tag{A.5}$$

with $\mathbf{r}_{ij} \subset \Sigma[x]$ and $i \in I_K$, $j \in \mathrm{adj}[i] \cap I_K$.

Now, when adding a subdomain i_{next} to the current I_K (line 17), we search for $\mathcal{V}_{i_{\text{next}}} \in \mathbb{R}[x]$ p.d. such that (A.5) holds for $i = i_{\text{next}}$ and all $j \in \mathrm{adj}[i_{\text{next}}] \cap I_K$.

Remark 10 *Instead of* (A.3)*, we might require*

$$s_{1,i} \vdash \Omega_{p \le \beta^\diamond} \subseteq_\Sigma \Omega_{\mathcal{V}_i \le \gamma^\diamond} \tag{A.6}$$

for all $1 \le i \le k$ *when searching for* $\mathcal{V}_1, \dots, \mathcal{V}_k$.

Eq. (A.6) imposes a less strict constraint on those $\mathcal{V}_i$ with $\beta_i > \beta^\diamond$.