CEAS-GNC-2024-067

EuroGNC Navigation Guidance Control CEAS Bristol, UK June 11th-13th 2024
University of BRISTOL ROYAL AERONAUTICAL SOCIETY AIAA

Enhanced Estimation of Region of Attraction Using Union Theorem in Sum-of-Squares Optimization for Longitudinal Stability Analysis

Bhaskar Biswas P.h.D Student, Cranfield University, School of Aerospace, Transport and Manufacturing, MK43 0AL, Cranfield, U.K. bhaskar.biswas@cranfield.ac.uk

Dmitry Ignatyev Senior Research Fellow, Cranfield University, School of Aerospace, Transport and Manufacturing, MK43 0AL, Cranfield, U.K. D.Ignatyev@cranfield.ac.uk

Argyrios Zolotas Professor, Cranfield University, School of Aerospace, Transport and Manufacturing, MK43 0AL, Cranfield, U.K. A.Zolotas@cranfield.ac.uk

Antonios Tsourdos Professor, Cranfield University, School of Aerospace, Transport and Manufacturing, MK43 0AL, Cranfield, U.K. a.tsourdos@cranfield.ac.uk

ABSTRACT

The Region-of-Attraction (ROA) is vital in determining a safe flight envelope and issuing flight control system safety certificates. Sum-of-Squares (SOS) optimization is a numerical method for estimating ROA in polynomial dynamical systems. However, it suffers from conservative estimations. This paper presents an innovative application of the Union Theorem within SOS optimization to significantly enhance ROA estimation. It focuses on ROA estimation for the polynomial dynamics governing aircraft short-period mode, evidencing enhanced safety margins. Extensive simulations performed for the Generic Transport Model (GTM) illustrate the efficacy of the proposed approach as a strong candidate for improving ROA analysis and facilitating future flight control clearance.

Keywords: Region of Attraction; Sum-of-Squares Optimization; Union Theorem; Generic Transport Model Aircraft; Short Period

Nomenclature

V	=	Air Speed
α	=	Angle of Attack
θ	=	Pitch Angle
q	=	Pitch Rate
δ_{elev}	=	Elevator Deflection Angle
δ_{th}	=	Engine Throttle
$\mathbb{R}^n$	=	Set of Real Number
$\mathcal{R}_n$	=	Set of Polynomial Function
Σ_n	=	Set of Sum-of-Squares Function
$V(x)$	=	Lyapunov Function

γ, β	=	Scalar
R_A	=	Region of Attraction
Ω_γ	=	Subset of Region of Attraction
P_β	=	Set of Shape Function
$p(x)$	=	Shape Function
l_1, l_2, s_i	=	Sum-of-Squares Polynomial Function
ψ	=	Inclination Angle of Straight Line

1 Introduction

In the realm of aerospace engineering, the precise understanding and characterization of the dynamic behavior of aircraft are paramount for ensuring flight safety, control, and stability, ([1] and [2]). The Region of Attraction (ROA) for a given aircraft mode represents the set of initial conditions from which the system will asymptotically stabilize to a desirable equilibrium point, [3], crucial for assessing the aircraft's stability and performance. Enlarging the ROA is of paramount importance as it allows for greater operational flexibility, improved safety margins, and enhanced system performance. Estimating ROA, especially for complex nonlinear dynamical systems, remains a formidable challenge in the field. This challenge has motivated the development of advanced mathematical and optimization techniques, among which Sum-of-Squares (SOS) optimization has proven to be a powerful tool.

The SOS optimization technique centers on SOS polynomials, which are non-negative polynomials expressed as the sums of squares of other polynomials. This property, shared by Lyapunov functions (LFs) and SOS polynomials, enables the representation of LFs as SOS polynomials. This methodology significantly extends the utility of linear matrix inequality (LMI) methods. Detailed insights into LMIs and their role in approximating the ROA are provided in [4] and [5]. In [6], an algorithm is presented to determine if a real polynomial qualifies as an SOS polynomial and, if so, to reveal its representation. As a relaxation technique, SOS explores the essential conditions for expressing non-negative polynomials as relaxed SOS polynomials, as elaborated in [7] and [8]. Instructional resources on the SOS optimization approach, including MATLAB code for ROA computation, are available in [9], [10], and [11]. In [12], the authors proposed an algorithm designed to effectively handle nonconvex SOS problems. Dedicated software tools like SOSOPT [13], BiSOS [12], and SOSTOOLS [14] are tailor-made to address challenges specific to SOS optimization. For further in-depth insights into the SOS optimization technique, readers can consult references [15], [16], and [17].

The works of authors [18], [19], [20], and [21] showcase the application of the SOS optimization technique in both controller design and the estimation of the ROA. In [22], [23], and [24], the authors employed the SOS optimization method to calculate the ROA for aircraft systems. These investigations integrated a Shape Function (SF) approach to expand the ROA's size. However, when dealing with an ROA of a real-world system, which has a complicated shape and might lack symmetry, the method still provides conservative results. A possible solution, in this case, is to increase the degree of polynomials used, however, it leads to a polynomial increase in the computational burden. To address these limitations, in [25], we introduced the Union Theorem as a solution for estimating the ROA in polynomial dynamical systems with non-symmetric or constrained actual ROAs. The Union Theorem facilitates the use of multiple SFs to enhance the ROA estimation. However, it should be noted that this paper focuses on presenting the effectiveness of the proposed method through straightforward examples.

The Generic Transport Model (GTM) aircraft [22] developed by NASA is a representative model encompassing the typical behavior of transport aircraft and therefore was used to demonstrate the capabilities of the proposed method. The short-period mode of the GTM aircraft is particularly critical, as it involves

rapid oscillations in pitch and requires precise control to ensure flight safety. Proper estimation of the ROA for this mode is essential for enhancing flight stability and control. In this article, we demonstrate the advantages of the SOS optimization technique in estimating an enhanced ROA for a short-period mode of the GTM aircraft.

The central objective of this paper is to utilize the Union Theorem within the framework of Sum-of-Squares (SOS) optimization to enhance the estimation of the ROA for the intricate polynomial dynamics exhibited by the real-world GTM aircraft model, with a particular emphasis on its short-period mode. By harnessing the power of the Union Theorem, it becomes feasible to integrate multiple Shape Functions (SFs) into the SOS optimization framework, leading to an improved estimation of the ROA while keeping the computational burden under control. This improved estimation seeks to make a meaningful contribution to the advancement of methods for evaluating and broadening the safe flight envelope.

The structure of this paper is outlined as follows: In Section 2, we provide an overview of the polynomial dynamics of the aircraft, background information on SOS optimization, and the formulation of the problem. Section 3 delves into the methodology used to estimate the improved ROA by incorporating the Union Theorem into the SOS optimization framework. Section 4 offers a comprehensive explanation of the numerical resolution procedure. The selection criteria for the required SFs are discussed in Section 5. Section 6 presents the results of our simulations. Lastly, in Section 7, we conclude with final remarks.

2 Preliminaries and Problem Formulation

2.1 The GTM Aircraft Polynomial Dynamics Model

NASA's GTM represents a 5.5 percent scale commercial aircraft. In the work cited as [22], the authors developed a polynomial model for the longitudinal dynamics of the GTM using data from a look-up table. The authors presented two sets of dynamics, one incorporating linear pitch rate damping functions and the other incorporating nonlinear pitch rate damping functions. In this context, we are focusing on the dynamics associated with nonlinear pitch rate damping functions, which better represent the real aircraft behaviour. The dynamics are given in the Appendix. Let's examine flight dynamics during level flight with a speed of $V = 45$ m/s. Under these circumstances, the trim value for the dynamics is as follows (Precision in variables prevents numerical errors):

$$\begin{bmatrix} V_t \\ \alpha_t \\ q_t \\ \theta_t \end{bmatrix} = \begin{bmatrix} 45\,\text{m/s} \\ 0.04927\,\text{rad} \\ 0\,\text{rad/s} \\ 0.04927\,\text{rad} \end{bmatrix} = \begin{bmatrix} 45\,\text{m/s} \\ 2.823\,\text{deg} \\ 0\,\text{deg/s} \\ 2.823\,\text{deg} \end{bmatrix}; \begin{bmatrix} \delta_{elev,t} \\ \delta_{th,t} \end{bmatrix} = \begin{bmatrix} 0.04892\,\text{rad} \\ 14.33901\% \end{bmatrix} = \begin{bmatrix} 2.8029\,\text{deg} \\ 14.33901\% \end{bmatrix} \tag{1}$$

In this research, we have specifically focused on the short period mode. To derive an open-loop polynomial model for the short period dynamics, we have extracted it from the 4-state polynomial longitudinal model described in appendix. This extraction is achieved by maintaining the values of V, θ, δ_{elev}, and δ_{th} at their respective trim values. The resulting model is provided below:

$$\begin{aligned}\dot{\alpha} = &-1.492063704793301\alpha^3 + 4.458632857384574\alpha^2 + 0.003062392312500003\alpha q \\ &+ 0.006226214343749994q^2 - 3.663222991982888\alpha + 0.92265866875q + 0.1698297338734228\end{aligned} \tag{2a}$$

$$\dot{q} = -7.227225000000001\alpha^3 + 1.101013509374996q^3 + 19.4319\alpha^2 - 47.19807492674931\alpha - 4.372278750000001q + 2.278970400659191 \quad (2b)$$

In order to adapt the model for computational algorithms, we shifted the equilibrium point to the origin. This transformation is accomplished by introducing a new set of variables denoted as $[\alpha_0, q_0] = [\alpha - \alpha_t,\ q - q_t]$, where the subscribe "0" corresponds to the origin. The adjusted model is presented below,

$$\dot{\alpha_0} = -1.492063704793301\alpha_0^3 + 4.238107862260349\alpha_0^2 + 0.003062392312500003\alpha_0 q_0 + 0.006226214343749994q_0^2 - 3.234767491359781\alpha_0 + 0.9228095412289729q_0 \quad (3a)$$

$$\dot{q_0} = -7.227225000000001\alpha_0^3 + 1.101013509374996q_0^3 + 18.36372593158282\alpha_0^2 - 45.33602748254988\alpha_0 - 4.372278750000001q_0 \quad (3b)$$

The Eq. (3) can be represented as autonomous system

$$\dot{x} = f(x). \quad (4)$$

Here, the trim point x = 0 of the shifted dynamics is an equilibrium point (EP). The goal is to estimate an enlarged ROA for this system, ensuring that the states, including the angle of attack (α), pitch angle (θ), and pitch rate (q), return to the EP even if they deviate as a result of significant disturbances over a wide range.

2.2 SOS Optimization

Some important definitions related to SOS optimization from,

Definition 1 (Region of Attraction) *Let us consider $\phi(x)$ as the solution of the system expressed by Eq. (4). The ROA of origin, $x = 0$ (EP) is, [3],*

$$R_A = \{x \in \mathbb{R}^n : \phi(x)\ is\ \forall\ t \geq 0\ and \lim_{t\to\infty} \phi(x) \to 0\} \quad (5)$$

Lemma 1 *If there exist a continuously scalar function $V(x) : \mathbb{R}^n \to \mathbb{R}$ and a positive scalar $\gamma \in \mathbb{R}^+$, such that,*

$$V(x) > 0\ \forall\ x \neq 0\ and\ V(0) = 0 \quad (6a)$$

$$\Omega_\gamma = \{x : V(x) \leq \gamma\}\ is\ bounded \quad (6b)$$

$$\Omega_\gamma \subseteq \{x : \frac{\partial V(x)}{\partial x}\dot{x} < 0\} \cup \{0\} \quad (6c)$$

Then the origin is asymptotically stable and $\Omega_\gamma \subseteq R_A$, [26].

The objective is to estimate an enlarged ROA, denoted as Ω_γ, for the system defined by Eq. (2). This can be achieved by optimizing the value of γ. One approach to optimize γ is through the use of the SOS optimization method. This method allows for the simultaneous determination of the LF, $V(x)$, expressed as a SOS function, and the optimized value of γ. However, directly estimating the ROA only using Lemma 1 in SOS optimization leads to conservative results, as elaborated upon and shown in [25]. To enhance the estimation of the ROA, a novel function known as the Shape Function (SF) has been introduced in [18] alongside Lemma 1, which is discussed below.

Definition 2 (Shape Function) *The Shape Function (SF) is a positive function that encompasses an area enclosed by the bounded LF. It induces the bounded LF to expand incrementally in each iteration by enlarging the SF itself until a part of the LF touches the unstable region. In simpler terms, the SF determines the final shape and size of the bounded LF. The variable size region under it is defined as [18],*

$$P_\beta = \{x : p(x) \leq \beta\} \subseteq \Omega_\gamma \tag{7}$$

2.2.1 SOS Optimization form of the ROA

If we incorporate the condition represented by Eq. (7) of the Shape Function (SF) with Lemma 1, the method for estimating the maximized Region of Attraction (ROA) can be formulated using the following Sum-of-Squares (SOS) optimization framework, as outlined in [18] and [25].

$$\begin{aligned}
&\beta^* = \max_{s_0,\, s_1 \in \sum_n,\, V(x) \in \mathcal{R}_n,\, \gamma \in \mathbb{R}^+} \beta \\
&\text{Subject to} : V(x) - l_1 \in \sum\nolimits_n && \text{(8a)} \\
&-\left[\frac{\partial V(x)}{\partial x} f(x) + l_2\right] + (V(x) - \gamma)s_0 \in \sum\nolimits_n && \text{(8b)} \\
&-(V(x) - \gamma) + (p(x) - \beta)s_1 \in \sum\nolimits_n && \text{(8c)}
\end{aligned}$$

Here, l_1, l_2, s_0 and s_1 are SOS polynomials, while $\sum_n$ is the set of SOS polynomial. Equation (8) outlines the process of maximizing the variable β, which leads to the determination of optimal values for both V and γ. Combining a single SF with Lemma 1 produces better results compared to using Lemma 1 alone to estimate the ROA, as discussed in [18] and [25]. However, it's important to acknowledge that Equation (8) provides an estimation for only a portion of the actual ROA, especially for systems with non-symmetric or unbounded actual ROAs. This issue is comprehensively discussed and demonstrated in [25]. To address this concern, we proposed the Union Theorem in [25], the details of which are explained in the subsequent section.

3 Enhanced ROA Estimation Using Union Theorem

Equation (8c) clearly states that the area encompassed by the SF should be confined within the bounds set by the LF and its corresponding γ value. SOS optimization tools, like SOSOPT, systematically augment the parameter β in each iteration, thereby encompassing additional regions within the SF. This expansion ensures the creation of a new LF, effectively covering the entirety of the SF's domain, all the while maintaining a constant γ. This iterative process, recognized as the $V - s$ iteration, will be thoroughly expounded upon in a subsequent section.

Estimating the ROA using a single SF within SOS optimization, as depicted in Eq. (8), involves the $V - s$ iteration algorithm. This method involves an augmented number of iterations to expand the ROA estimate. However, it's important to note that this approach isn't a universal panacea. It may not prove effective in certain scenarios, such as instances of early optimization convergence or numerical infeasibility. This is particularly true when dealing with systems exhibiting unbounded or irregular ROAs.

Another aspect to consider is that the geometric center of the SF, whether fixed or adaptive, is invariably positioned at the origin. This characteristic imposes limitations on the estimation process, especially when dealing with non-symmetric or unbounded ROAs. This limitation arises because, over the course of several iterations, the SF gradually approaches the unstable region along specific directions

within the non-symmetric ROA. Subsequent increases in β do not yield any new LF that encircles the SF, as it's on the verge of touching the unstable region. Essentially, the LF captures the ultimate region exclusively along this particular direction. However, a substantial unexplored territory remains between the estimated ROA derived from Eq. (8) and the actual ROA. This gap is particularly pronounced when the actual ROA exhibits non-symmetric characteristics, making it challenging to estimate it through the single SF-based approach. To address this issue, we proposed Union Theorem in [25]. The Theorem is,

Theorem 1 (Union Theorem) *Let us consider the given polynomials $V(x)$, $p_1(x)$, $p_2(x)$,...,$p_n(x)$ define sets A_V, A_1, A_2,...,A_n such that,*

$$\begin{aligned} A_V &= \{x \in \mathbb{R}^n : V(x) - \gamma \leq 0\} \\ A_1 &= \{x^1 \in \mathbb{R}^n : p_1(x) - \beta_1 \leq 0\} \\ A_2 &= \{x^2 \in \mathbb{R}^n : p_2(x) - \beta_2 \leq 0\} \\ &\vdots \\ A_n &= \{x^n \in \mathbb{R}^n : p_n(x) - \beta_n \leq 0\} \end{aligned}$$

Here, $\left\{\gamma, \beta_1, \beta_2, ...\beta_n\right\} \in \mathbb{R}^+$. Now if there exit polynomials $s_1 \in \sum_n$, $s_2 \in \sum_n$,...,$s_n \in \sum_n$, such that,

$$\begin{aligned} &-(V(x) - \gamma) + (p_1(x) - \beta_1)s_1 \in \sum\nolimits_n \\ &-(V(x) - \gamma) + (p_2(x) - \beta_2)s_2 \in \sum\nolimits_n \\ &\quad\vdots \\ &-(V(x) - \gamma) + (p_n(x) - \beta_n)s_n \in \sum\nolimits_n \end{aligned}$$

Then, $A_1 \cup A_2 \cup ... \cup A_n \subseteq A_V$.

The Union Theorem tells us that if we introduce multiple SFs ($p_{1,..,n}$) in different places then we can get an LF (V), which will encircle all of them. If we incorporate the Union Theorem with the Eq. (8), then the modified form of the Eq. (8) can be expressed in the following way,

$$\left[\beta_1^*, \beta_2^*, ..., \beta_n^*\right] = \max_{s_0, s_1, s_2, ..., s_n \in \sum_n,\, V(x) \in \mathcal{R}_n,\, \gamma \in \mathbb{R}^+} \beta_1, \beta_2, ..., \beta_n$$

$$\text{Subject to:} \quad V(x) - l_1 \in \sum\nolimits_n \tag{9a}$$

$$-\left[\frac{\partial V(x)}{\partial x} f(x) + l_2\right] + (V - \gamma)s_0 \in \sum\nolimits_n \tag{9b}$$

$$-(V(x) - \gamma) + (p_1(x) - \beta)s_1 \in \sum\nolimits_n \tag{9c}$$

$$-(V(x) - \gamma) + (p_2(x) - \beta_2)s_2 \in \sum\nolimits_n \tag{9d}$$

$$\vdots$$

$$-(V(x) - \gamma) + (p_n(x) - \beta_n)s_n \in \sum\nolimits_n \tag{9n}$$

Remark 1. Eq. (9a) ensures the positive definiteness of $V(x)$, which is the 1st condition for a LF.

Remark 2. Eq. (9b) tells us that the region enclosed by $V(x) \leq \gamma$ should be always inside the stable region as $\dot{V}(x) \leq -l_2$.

Remark 3. Eq. (9c)-(9n) make sure that the regions enclosed by $p_1(x) \leq \beta_1, ..., p_n(x) \leq \beta_1$ should be always inside the region enclosed by $V(x) \leq \gamma$.

The $V - s$ iteration algorithm is applied to solve Eq. (9). During each iteration, this algorithm incrementally raises the values of β for each SF, generating an associated LF that encloses additional regions by encompassing all SFs. While it's possible for the algorithm employing multiple SFs to halt if a

particular SF approaches the unstable region in one direction, it has already achieved an expanded ROA. This expansion arises from the simultaneous increase of all SFs in various directions, thereby enlarging the ROA in multiple dimensions.

4 Numerical Algorithm

The iterative procedure, known as the $V-s$ iteration algorithm, addresses Eq. (9) through a sequence of three distinct steps: the γ-step, the β-step, and the V-step, as detailed in references [18] and [25]. The first two steps named γ-step and β-step involve optimizing γ and β_i, respectively. In contrast, the third step, known as the V-step, involves solving a feasibility problem to find a new LF ($V(x)$) that fulfills all associated constraints under it. The algorithm's framework is as follows,

Algorithm :

1. γ-step: Solve for s_0 and γ^* for a given $V(x)$ and fixed l_2 :

$$\gamma^* = \max_{s_0 \in \sum_n} \gamma \;\; \text{s.t.} : -\left[\frac{\partial V(x)}{\partial x} f(x) + l_2\right] + (V(x) - \gamma) s_0 \in \sum\nolimits_n \tag{10}$$

2. β-step: Solve for $s_1, s_2, ..., s_n$ and $\beta_1^*, \beta_2^*, ..., \beta_n^*$ for a given $V(x)$, $p_i(x)$ and using obtained γ^* from γ-step:

$$\left[\beta_1^*, \beta_2^*, ..., \beta_n^*\right] = \max_{s_1, s_2, ..., s_n \in \sum_n} \beta_1, \beta_2 ..., \beta_n \;\; \text{s.t.} :$$

$$-(V(x) - \gamma^*) + (p_1(x) - \beta_1) s_1 \in \sum\nolimits_n \tag{11a}$$

$$-(V(x) - \gamma^*) + (p_2(x) - \beta_2) s_2 \in \sum\nolimits_n \tag{11b}$$

$$\vdots$$

$$-(V(x) - \gamma^*) + (p_n(x) - \beta_n) s_n \in \sum\nolimits_n \tag{12n}$$

3. V-step: Using the obtained γ^*,s_0, s_1, s_2, ..., s_n and $\beta_1^*, \beta_2^*, ..., \beta_n^*$ from previous two steps, solve for a new V which satisfies the following :

$$V(x) - l_1 \in \sum\nolimits_n \tag{12a}$$

$$-\left[\frac{\partial V(x)}{\partial x} f(x) + l_2\right] + (V(x) - \gamma^*) s_0 \in \sum\nolimits_n \tag{12b}$$

$$-(V(x) - \gamma^*) + (p_1(x) - \beta_1^*) s_1 \in \sum\nolimits_n \tag{12c}$$

$$-(V(x) - \gamma^*) + (p_2(x) - \beta_2^*) s_2 \in \sum\nolimits_n \tag{12d}$$

$$\vdots$$

$$-(V(x) - \gamma^*) + (p_n(x) - \beta_n^*) s_n \in \sum\nolimits_n \tag{13n}$$

4. Scale V: Replace $V(x)$ with $V(x)/\gamma^*$ after each V-step. This scaling process roughly normalizes $V(x)$ and tends to keep the γ^* computed in the next step (γ-step) close to unity.

5. Repeat all the steps from $1-4$ using the scaled $V(x)$ from step 4 to the γ-step as an input. Continue the process till the algorithm provides a feasible $V(x)$ or $(\beta_i(j) - \beta_i(j-1)) \geq \delta$ (Predefined Tolerance).

Remark 1. To implement the $V-s$ algorithm during the γ-step, an initial LF denoted as V_0 is essential. There are various methods available for obtaining V_0, including the use of the Lyapunov Equation (LE) based on the linearized dynamics (LD) of the system, as used in [25] and [27]. The authors of [27] proposed that conducting multiple iterations can enhance the estimation of the ROA. After completing the algorithm, a final LF, denoted as V_{1R}, and increased β values for all Shape Functions (SFs) are

obtained. However, even with the incorporation of multiple SFs, V_{1R} may not entirely cover the entire potential ROA. To extend the coverage, the algorithm is restarted with V_{1R} as the initial LF for the subsequent iteration round, potentially resulting in an enlarged ROA. Different types of SFs may be used in each round, and employing diverse types can sometimes yield improved results. Nevertheless, it's essential to recognize that the LF may become saturated after a certain number of iteration rounds, reaching its maximum coverage and unable to capture additional areas, even with additional rounds. Furthermore, there may be instances where no new LF satisfying constraint Eq. (12) can be found, despite the presence of a significant stable region beyond the already captured region of $V(x)$.

Remark 2. The following degrees are required to satisfy for a feasible solution of (10)-(12), ([18], [25], [27]),

$$\deg(V(x)) \geq \deg(l_1)$$
$$\deg(p_i(x)) + \deg(s_i) \geq \deg(V(x)), \ \ [i = 1, ..., n]$$
$$\deg(V(x)) + \deg(s_0) \geq \max\{\deg(\frac{\partial V(x)}{\partial x} f), \ \deg(l_2)\}$$

5 Selection of Shape Function

Utilizing the Shape Function (SF) is a critical factor in enhancing the estimation of the ROA. The selection of the SF also impacts the overall duration of the iteration process. Different types of SFs lead to the generation of unique LFs with varying ROAs and computational time requirements. According to the definition, any positive definite function can serve as an SF. The below SF with a shifting center x^* is proposed in [27],

$$p(x) = (x - x^*)^T N (x - x^*) \tag{13}$$

The positive definite shape matrix N plays a crucial role in defining the SF, determining the size of the ROA, and calculating the value of V. It's worth noting that relying on a single fixed value for N may not yield optimal results. Therefore, it is advisable, during each iteration round, to explore various N values and compare the estimated ROAs to choose the most suitable one. To determine the shifting center, denoted as $x*$, for the SF, we utilize a method introduced in [27]. This method involves calculating $x*$ by solving the equation of a straight line that passes through the center and intersects the bounded LF. While we've provided the mathematical expression for choosing the shifting center in the 2-D case, it's essential to highlight that this method can be applied to the 3-D case as well. Let's consider the equations for the bounded LF and the 2-D straight line,

$$V(x_1, x_2) = \gamma^* \tag{14a}$$
$$x_2 = \tan\psi x_1 \tag{14b}$$

Here, ψ is the pre-defined inclination angle of the line, and for every shifting center, there is a fixed ψ. Consider the solution point or intersection point of the above equations to be (x^*_{1I}, x^*_{2I}). The distance from the center to the point (x^*_{1I}, x^*_{2I}) is given by $\rho_\alpha = \sqrt{x^*_{1I} + x^*_{2I}}$. So, the shifting center is,

$$x^*_1 = \sigma\rho_\alpha \cos\psi \tag{15a}$$
$$x^*_2 = \sigma\rho_\alpha \sin\psi \tag{15b}$$

Where, $\sigma \in (0, 1)$. More discussion about the selection of the value of σ can be found in [25] and [27].

6 Simulation Results

Extensive simulations have been conducted to evaluate the effectiveness of the proposed approach on the short period mode behavior of the GTM aircraft, represented by Eq. (3). These simulations were

carried out using the SOSOPT Toolbox [13]. It's important to highlight that each iteration round requires a new initial LF. In the first round, the LF is obtained by solving the Lyapunov equation (LE) for the linearized dynamics (LD). In subsequent rounds, the LF from the previous iteration is utilized as the initial LF for the next round. The center of the SF, as represented by Eq. (15), is necessary to initiate the simulation for each new round of iteration, as it changes with each round. To determine the center, x^*, Eqs. (14a) and (14b) are solved with a pre-defined angle ψ. The angles and the SF matrix, N, are selected based on error and trial methods. More discussion about the selection of the SF can be found in [22]. For these simulations, specific parameter settings were chosen: $deg(V(x) \rightarrow [min(2), max(6)]$, $deg(s_0) \rightarrow [min(2), max(4)]$, $deg(s_{1,\ldots,n}) \rightarrow [min(0), max(4)]$, $l_1 = l_2 = 10^{-6}(x'x)$, and $\sigma = 0.8$ to initiate the simulation. Additionally, we compared the results generated by the proposed method in this paper to those generated by a single SF-based approach in paper [22].

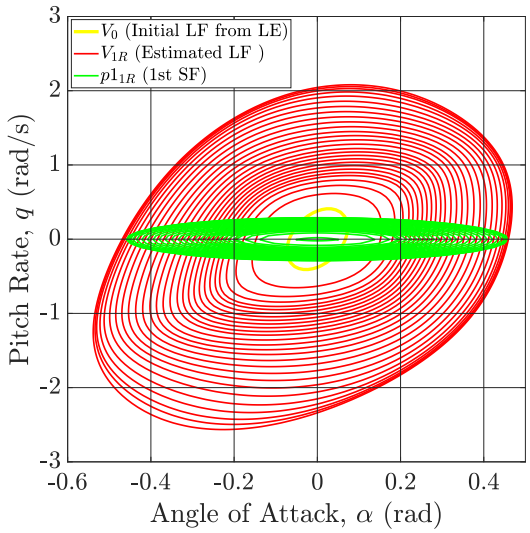
Here, we present the results using the Union Theorem in the SOS optimization method. To illustrate the simulation outcomes, we executed three successive rounds of iterations. Initially, we employed a single SF centered at the origin for the first round. Subsequently, for the second and third rounds, we employed a total of four SFs. The parameters used for this process are given in Table 1.

Table 1 Simulation Parameters

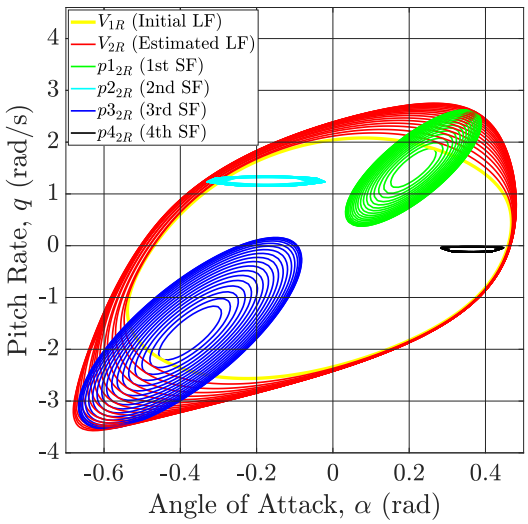
Round	Initial LF (V_0)	Angle (ψ°)	Shape Matrix (N)
1st (1R)	V_0 =From LE	$\psi 1_{1R}$ =Not Applicable	$N1_{1R} = [0.3491\ \ 0; 0\ \ 0.8727]$
2nd (2R)	$V_0 = V_{1R}$	$\psi 1_{2R}$=82	$N1_{2R} = [98.529\ \ -11.975; -11.975\ \ 2.471]$
		$\psi 2_{2R}$=98	$N2_{2R} = [0.3491\ \ 0; 0\ \ 0.8727]$
		$\psi 3_{2R}$=257.5	$N3_{2R} = [98.529\ \ -11.975; -11.975\ \ 2.471]$
		$\psi 4_{2R}$=350	$N4_{2R} = [0.3491\ \ 0; 0\ \ 0.8727]$
3rd (3R)	$V_0 = V_{2R}$	$\psi 1_{3R}$=82	$N1_{3R} = [221.6731\ \ -27.0953; -27.0953\ \ 4.3269]$
		$\psi 2_{3R}$=108	$N2_{3R} = [0.3491\ \ 0; 0\ \ 0.8727]$
		$\psi 3_{3R}$=257	$N3_{3R} = [219.5183\ \ -34.6099; -34.6099\ \ 6.4817]$
		$\psi 4_{3R}$=350	$N4_{3R} = [0.3491\ \ 0; 0\ \ 0.8727]$

The findings are depicted in Fig. 1, illustrating the utilization of various shape functions (SFs) due to the system's irregular ROA. Consequently, the shape matrix is strategically selected to expand in diverse directions. Figs. 1(a), 1(b), and 1(c) display the evolution of LFs and their associated SFs for each iteration, with a new LF emerging when SFs are updated. In Fig. 1(a), it is evident that the estimated region expands with each iteration, but the process halts when the LF approaches the unstable region in the second and fourth quadrants. Nonetheless, gaps persist in the first and third quadrants. In the subsequent two iterations, SFs and their centers are carefully chosen to allow the estimated LF to encompass more area without relinquishing previously acquired territory. This iterative approach culminates in Fig. 1(c), where the third round vastly outperforms the first two in terms of region coverage.

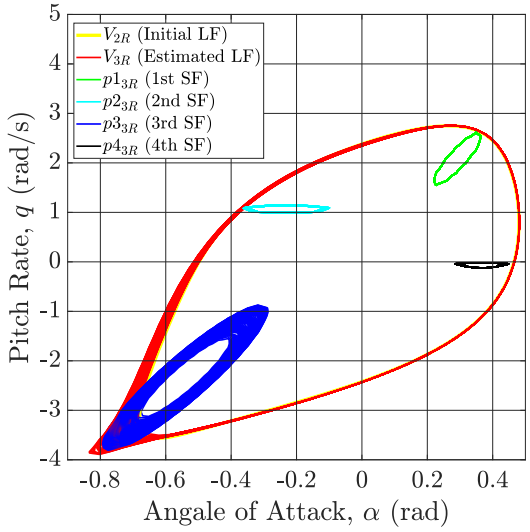
Fig. 1(d) compares the ROAs estimated from all rounds with those obtained using linearized dynamics. Notably, the ROAs in Figs. 1(a), 1(b), and 1(c) are initially estimated around the origin (0,0), but they have been shifted to the trim point and presented in Figs. 1(e) and 1(f). Additionally, through Monte Carlo simulation, we have identified initial points for stable and unstable trajectories, as depicted in Fig. 1(e). The estimated ROA consistently remains within the stable region. Fig. 1(f) provides a degree-based representation of the final estimated ROA. In Fig. 1(e), we compare the results obtained using a single SF [results from [22]] and the Union Theorem, where the latter outperforms existing methods. From the



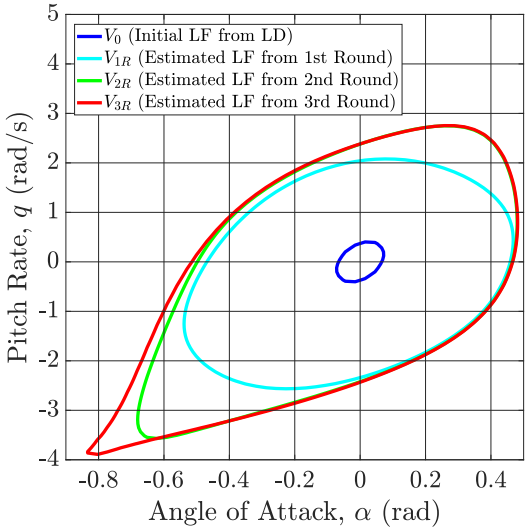
(a) Estimated ROA from 1st Round



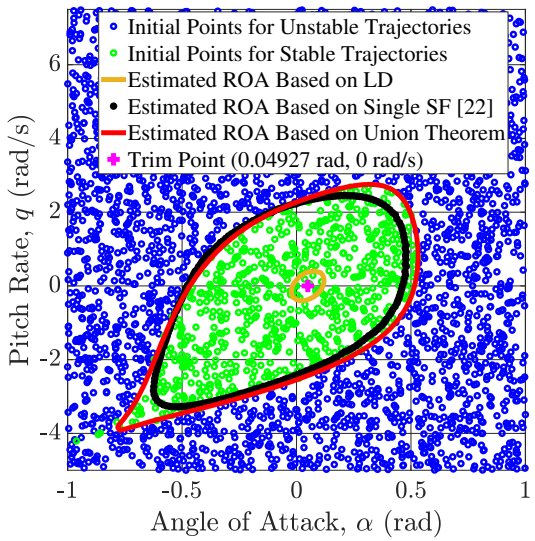
(b) Estimated ROA from 2nd Round



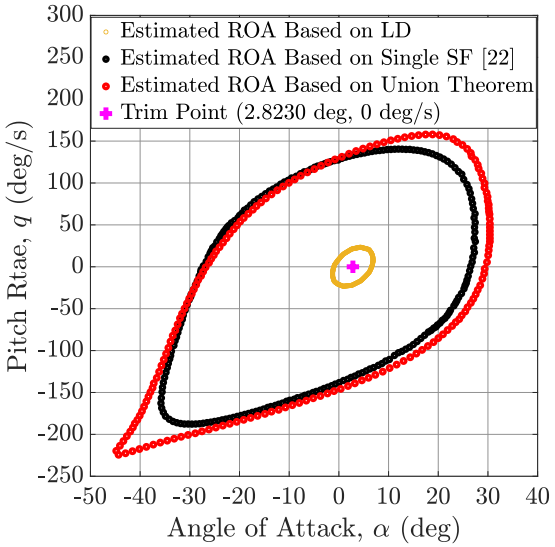
(c) Estimated ROA from 3rd Round



(d) Comparison Among Results from all the Rounds



(e) Monte Carlo Simulation with Estimated ROAs



(f) Comparison Among Estimated ROAs based on LD, Single SF and Union Theorem

Fig. 1 Estimated ROA Using Union Theorem

figure one can see that our method covers the whole stable region, thus providing a realistic estimation of the ROA. Such a level of accuracy is important for the certification of a safe flight envelope.

To generate results based on our proposed method [represented by Figs. 1(a), 1(b), and 1(c)], the entire computational simulation process required 16 minutes, encompassing 28, 19, and 73 iterations in the first, second, and third rounds of iteration, respectively. On the other hand, to identify the stable and unstable areas of the system, a total of 3550 initial points have been generated for the Monte Carlo simulation [represented by Fig. 1(e)], with the total computational time being 2 hours and 22 minutes. Comparing the computational time of our proposed method with the Monte Carlo method, it is clear that our proposed method takes significantly less time than the Monte Carlo simulation to identify the stable areas, i.e., the ROA of the system.

In Ref. [22], the authors utilized a single SF, and we also employed a single SF in the first round of iteration. However, there is a discrepancy in the estimated ROAs between the two cases, despite the consistent use of a single SF. This difference arises from the need to define certain variables like SOS polynomials s_0 and s_1. The authors of [22] did not specify these variables, so we introduced our own definitions, leading to variations in the outcomes. It's worth noting that utilizing a single SF proves challenging in encompassing the entire area of an irregular ROA. This challenge arises because the simulation halts when the estimated LF approaches the unstable region in specific directions. Consequently, both the results from [22] and our initial iteration failed to cover the entire area, as the estimated LF came close to the unstable region in the second and fourth (near to - q axis) quadrants.

For the GTM model, the valid ranges for angle of attack and pitch rate are $-5\,\text{deg} \leq \alpha \leq 50\,\text{deg}$ and $-70\,\text{deg/s} \leq q \leq 70\,\text{deg/s}$, respectively. Upon examining Fig. 1(f), it becomes evident that our estimated ROA goes beyond lower angles of attack limit and pitch rate limits. However, we can still see that our predicted expansion of the stable ROA into the higher angles-of-attack region for up to 50 deg is within model validity.

We intentionally did not impose any limits on these variables during simulations to make a fair benchmarking against [22], where the authors also did not constrain these variables in their study. Nonetheless, it is crucial to emphasize that this observation does not undermine the applicability of the Union Theorem for estimating a larger ROA in complex real-world problems. Furthermore, these limitations can be naturally included in the optimization problem.

The analyses are performed on Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz, 16.00 RAM.

7 Conclusion

This paper proposed to use Union Theorem within the SOS optimization framework for flight dynamics analysis. The proposed method stands out from the other considered methods. In particular, it provides almost the same ROA estimation as the Monte Carlo, however, as opposed to it, it provides a safety certificate in the whole identified ROA in the form of LF. The other SOS optimisation tools provide the same LF certificates, however, the estimated ROAs are more conservative, thus limiting the application of these methods to the real aircraft flight control clearance. However, it is important to acknowledge some future directions for improving the proposed method. The outcomes are influenced by factors such as the type (N) and quantity of SFs utilized, as well as the selection of SF centers, which can significantly impact the results. In this study, the SFs and their corresponding angles for center positioning were manually chosen. Adopting a more structured approach in selecting these components has the potential to enhance the obtained results.

Acknowledgments

The first author acknowledges the Government of India, Ministry of Social Justice and Empowerment for supporting the studies under the National Overseas Scholarship for Ph.D. with scholarship number 11015/33/2019 SCD-V NOS.

Appendix

The polynomial Longitudinal Dynamics of the GTM Aircraft, [22],

$$
\begin{aligned}
\dot{V} =& 1.233\times10^{-8}V^4q^2 + 4.85310^{-9}\alpha^3\delta_{th}^3 + 3.705\times10^{-5}V^3\alpha q - 2.184\times10^{-6}V^3q^2 + 2.203\times10^{-2}V^2\alpha^3 \\
&- 2.836\times10^{-6}\alpha^3\delta_{th}^2 + 3.885\times10^{-7}\alpha^2\delta_{th}^3 - 1.069\times10^{-6}V^3q - 4.517\times10^{-2}V^2\alpha^2 \\
&- 2.140\times10^{-3}V^2\alpha\delta_{elev} - 3.282\times10^{-3}V^2\alpha q - 8.901\times10^{-4}V^2\delta_{elev}^2 + 9.677\times10^{-5}V^2q^2 \\
&- 2.037\times10^{-4}\alpha^3\delta_{th} - 2.270\times10^{-4}\alpha^2\delta_{th}^2 - 2.912\times10^{-8}\alpha\delta_{th}^3 + 1.591\times10^{-3}V^2\alpha \\
&- 4.077\times10^{-4}V^2\delta_{elev} + 9.475\times10^{-5}V^2q - 1.637\alpha^3 - 1.631\times10^{-2}\alpha^2\delta_{th} + 4.903\alpha^2\theta \\
&- 4.903\alpha\theta^2 + 1.702\times10^{-5}\alpha\delta_{th}^2 - 7.771\times10^{-7}\delta_{th}^3 + 1.634\theta^3 - 4.319\times10^{-4}V^2 - 2.142\times10^{-1}\alpha^2 \\
&+ 1.222\times10^{-3}\alpha\delta_{th} + 4.541\times10^{-4}\delta_{th}^2 + 9.823\alpha + 3.261\times10^{-2}\delta_{th} - 9.807\theta + 4.284\times10^{-1}
\end{aligned}
$$

$$
\begin{aligned}
\dot{\alpha} =& -3.709\times10^{-11}V^5q^2 + 6.869\times10^{-11}V\alpha^3\delta_{th}^3 + 7.957\times10^{-10}V^4\alpha q + 9.860\times10^{-9}V^4q^2 \\
&+ 1.694\times10^{-5}V^3\alpha^3 - 4.015\times10^{-8}V\alpha^3\delta_{th}^2 - 7.722\times10^{-12}V\alpha^2\delta_{th}^3 - 6.086\times10^{-9}\alpha^3\delta_{th}^3 \\
&- 2.013\times10^{-8}V^4q - 5.180\times10^{-5}V^3\alpha^2 - 2.720\times10^{-6}V^3\alpha\delta_{elev} - 1.410\times10^{-7}V^3\alpha q \\
&+ 7.352\times10^{-7}V^3\delta_{elev}^2 - 8.736\times10^{-7}V^3q^2 - 1.501\times10^{-3}V^2\alpha^3 - 2.883\times10^{-6}V\alpha^3\delta_{th} \\
&+ 4.513\times10^{-9}V\alpha^2\delta_{th}^2 - 4.121\times10^{-10}V\alpha\delta_{th}^3 + 3.557\times10^{-6}\alpha^3\delta_{th}^2 \\
&+ 6.841\times10^{-10}\alpha^2\delta_{th}^3 + 4.151\times10^{-5}V^3\alpha + 3.648\times10^{-6}V^3\delta_{elev} + 3.566\times10^{-6}V^3q \\
&+ 6.246\times10^{-6}V^2\alpha q + 4.589\times10^{-3}V^2\alpha^2 + 2.410\times10^{-4}V^2\alpha\delta_{elev} - 6.514\times10^{-5}V^2\delta_{elev}^2 \\
&+ 2.580\times10^{-5}V^2q^2 - 3.787\times10^{-5}V\alpha^3 + 3.241\times10^{-7}V\alpha^2\delta_{th} + 2.409\times10^{-7}V\alpha\delta_{th}^2 \\
&+ 1.544\times10^{-11}V\delta_{th}^3 + 2.554\times10^{-4}\alpha^3\delta_{th} - 3.998\times10^{-7}\alpha^2\delta_{th}^2 + 3.651\times10^{-8}\alpha\delta_{th}^3 \\
&+ 4.716\times10^{-7}V^3 - 3.677\times10^{-3}V^2\alpha - 3.231\times10^{-4}V^2\delta_{elev} - 1.579\times10^{-4}V^2q \\
&+ 2.605\times10^{-3}V\alpha^2 + 1.730\times10^{-5}V\alpha\delta_{th} - 5.201\times10^{-3}V\alpha\theta - 9.026\times10^{-9}V\delta_{th}^2 \\
&+ 2.601\times10^{-3}V\theta^2 + 3.355\times10^{-3}\alpha^3 - 2.872\times10^{-5}\alpha^2\delta_{th} - 2.134\times10^{-5}\alpha\delta_{th}^2 \\
&- 1.368\times10^{-9}\delta_{th}^3 - 4.178\times10^{-5}V^2 + 2.272\times10^{-4}V\alpha - 6.483\times10^{-7}V\delta_{th} \\
&- 2.308\times10^{-1}\alpha^2 - 1.532\times10^{-3}\alpha\delta_{th} + 4.608\times10^{-1}\alpha\theta - 2.304\times10^{-1}\theta^2 \\
&+ 7.997\times10^{-7}\delta_{th}^2 - 5.210\times10^{-3}V - 2.013\times10^{-2}\alpha + 5.744\times10^{-5}\delta_{th} + q + 4.616\times10^{-1} \\
\dot{q} =& -6.573\times10^{-9}V^5q^3 + 1.747\times10^{-6}V^4q^3 - 1.548\times10^{-4}V^3q^3 - 3.569\times10^{-3}V^2\alpha^3 \\
&+ 4.571\times10^{-3}V^2q^3 + 4.9530\times10^{-5}V^3q + 9.596\times10^{-3}V^2\alpha^2 + 2.049\times10^{-2}V^2\alpha\delta_{elev} \\
&- 2.431\times10^{-2}V^2\alpha - 3.063\times10^{-2}V^2\delta_{elev} - 4.388\times10^{-3}V^2q - 2.594\times10^{-7}\delta_{th}^3 \\
&+ 2.461\times10^{-3}V^2 + 1.516\times10^{-4}\delta_{th}^2 + 1.089\times10^{-2}\delta_{th} + 1.430\times10^{-1} \\
\dot{\theta} =& q
\end{aligned}
$$

The dynamics is valid for 30 m/s $\leq V \leq$ 60 m/s, -5 deg $\leq \alpha \leq$ 50 deg, -50 deg $\leq \theta \leq$ 50 deg and -70 deg/s $\leq q \leq$ 70 deg/s

References

[1] M. G. Goman, A. V. Khramtsovsky, and E. N. Kolesnikov. Evaluation of aircraft performance and maneuverability by computation of attainable equilibrium sets. *Journal of Guidance, Control and Dynanmics*, 31(2), March 2008. DOI: 10.2514/1.29336.

[2] S. J. Gill, M. H. Lowenberg, S. A. Neild, B. Krauskopf, G. Puyou, and E. Coetzee. Upset dynamics of an airliner model: A nonlinear bifurcation analysis. *Journal of Aircraft*, 50(6), November 2013. DOI: 10.2514/1.C032221.

[3] K. H. Khalil. *Nonlinear Systems*. Prentice Hall, 3rd edition, 2002. ISBN: 978-0131227408.

[4] J. G. VanAntwerp and R. D. Braatz. A tutorial on linear and bilinear matrix inequalities. *Journal of Process Control*, 10(4), 2000. DOI: 10.1016/S0959-1524(99)00056-6.

[5] B. Tibken. Estimation of the domain of attraction for polynomial systems via LMIs. *Proceedings of the 39th IEEE Conference on Decision and Control*, 4, 2000. DOI: 10.1109/CDC.2000.912314.

[6] V. Powers and T. Wormann. An algorithm for sums of squares of real polynomials. *Journal of Pure and Applied Algebra*, 127(1), 2000. DOI: 10.1016/S0022-4049(97)83827-3.

[7] P. A. Parrilo. *Structured Semidefinite Programs and Semi-algebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, California, USA, 2000. DOI: 10.7907/2K6Y-CH43.

[8] P. A. Parrilo. Semidefinite programming relaxations for semi-algebraic problems. *Mathematical Programming*, 96(2), 2003. DOI: 10.1007/s10107-003-0387-5.

[9] G. Balas, A. Packard, P. Seiler, and U. Topcu. Robustness analysis of nonlinear systems. NASA Workshop, NASA Langley Research Center, Hampton, Virginia, USA, 2009. `http://users.cms.caltech.edu/~utopcu/saveMaterial/LangleyWorkshop.html`.

[10] M. M. Peet. LMI methods in optimal and robust control. Lecture, Arizona State University, Phoenix, Arizona, USA, 2020. `https://control.asu.edu/MAE598_frame.htm`.

[11] A. Papachristodoulou and S. Prajna. A tutorial on sum of squares techniques for systems analysis. *Proceedings of the 2005, American Control Conference*, 4(2), 2005. DOI: 10.1109/ACC.2005.1470374.

[12] T. Cunis and B. Legat. Sequential sum-of-squares programming for analysis of nonlinear systems. *American Control Conference*, 2023. DOI: 10.23919/ACC55779.2023.10156153.

[13] P. Seiler. Sosopt: A toolbox for polynomial optimization. *ArXiv*, 2013. DOI: 10.48550/arXiv.1308.1889.

[14] A. Papachristodoulou, J. Anderson, G. Valmorbida, S. Prajna, P. Seiler, P. Parrilo, M. M. Peet, and D. Jagt. SOSTOOLS version 4.00 sum of squares optimization toolbox for MATLAB. *ArXiv*, 2013. DOI: 10.48550/arXiv.1310.4716.

[15] A. A. Ahmadi. Sum of squares (SOS) techniques : An introduction. Lecture, Princeton University, Princeton, New Jersey, USA, 2016. `https://www.princeton.edu/~aaa/Public/Teaching/ORF523/ORF523_Lec15.pdf`.

[16] S. Lall. Sums of squares. Lecture, Stanford University, Stanford, California, USA, 2011. `https://web.stanford.edu/class/ee364b/lectures/sos_slides.pdf`.

[17] D. Kunisky. Lecture notes on sum-of-squares optimization. Lecture notes, 2022. `http://www.kunisky.com/static/teaching/2022spring-sos/sos-notes.pdf`.

[18] Z. J. Wloszek, R. Feeley, T. Weehong, K. Sun, and A. Packard. Some controls applications of sum of squares programming. *42nd IEEE International Conference on Decision and Control*, 5, 2003. DOI: 10.1109/CDC.2003.1272309.

[19] W. Tan and A. Packard. Searching for control lyapunov functions using sums of squares programming. *sibi*, 1(1), 2004.

[20] Z. Jarvis-Wloszek. *Lyapunov Based Analysis and Controller Synthesis for Polynomial Systems using Sum-of-Squares Optimization*. PhD thesis, University of California, Berkeley, California, USA, 2003. `https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=6de7b336f6fc12cf158c4b6e0c6786fcd0dcd4cd`.

[21] W. Tan. *Nonlinear Control Analysis and Synthesis using Sum-of-Squares Programming*. PhD thesis, University of California, Berkeley, California, USA, 2006. `https://bcci.me.berkeley.edu/downloads/papers/weehong_4.pdf`.

[22] A. Chakraborty, P. Seilera, and G. Balas. Nonlinear region of attraction analysis for flight control verification and validation. *Control Engineering Practice*, 19(4), 2011. DOI: 10.1016/j.conengprac.2010.12.001.

[23] A. Khrabrov, M. Sidoryuk, and D. Ignatyev. Estimation of regions of attraction of spin modes. *7th European Conference for Aerospace Sciences*, 2017. DOI: 10.13009/EUCASS2017-312.

[24] D. Li, A. Tsourdos, Z. Wang, and D. Ignatyev. Nonlinear analysis for wing-rock system with adaptive control. *Journal of Guidance, Control, and Dynamics*, 45(11), 2022. DOI: 10.2514/1.G006775.

[25] B. Biswas, D. Ignatyev, A. Zolotas, and A. Tsourdos. Region of attraction estimation using union theorem in sum-of-squares optimization. *ArXiv*, 2023. DOI: 10.48550/arXiv.2305.11655.

[26] M. Vidyasagar. *Nonlinear Systems Analysis*. SIAM, 2nd edition, 2002. ISBN: 0898719186.

[27] D. Li, D. Ignatyev, A. Tsourdos, and Z. Wang. Estimation of non-symmetric and unbounded region of attraction using shifted shape function and r-composition. *ISA Transactions*, 136, 2022. ISSN: 0019-0578. DOI: 10.1016/j.isatra.2022.11.015.