Assessment of aircraft flight controllers using nonlinear robustness analysis techniques

Peter Seiler[1], Gary J. Balas[2], and Andrew K. Packard[3]

[1] Aerospace Engineering & Mechanics, University of Minnesota, Minneapolis, MN, 55455, USA, seiler@aem.umn.edu
[2] Aerospace Engineering & Mechanics, University of Minnesota, Minneapolis, MN, 55455, USA, balas@aem.umn.edu
[3] Mechanical Engineering, University of California, Berkeley, CA, 94720, USA, apackard@berkeley.edu

Summary. The current practice to validate flight control laws relies on applying linear analysis tools to assess the closed loop stability and performance characteristics about many trim conditions. Nonlinear simulations are used to provide further confidence in the linear analyses and also to uncover dynamic characteristics, e.g. limit cycles, which are not revealed by the linear analysis. This chapter reviews nonlinear analysis techniques which can be applied to systems described by polynomial dynamic equations. The proposed approach is to approximate the aircraft dynamics using polynomial models. Nonlinear analyses can then be solved using sum-of-squares optimization techniques. The applicability of these methods is demonstrated with nonlinear analyses of an F/A-18 aircraft and NASA's Generic Transport Model aircraft. These nonlinear analysis techniques can fill the gap between linear analysis and nonlinear simulations and hence used to provide additional confidence in the flight control law performance.

1 Introduction

The current practice to validate flight control laws relies on applying linear analysis tools to assess the closed loop stability and performance characteristics about many trim conditions. Nonlinear simulations are then used to provide further confidence in the linear analyses and also to uncover dynamic characteristics, e.g. limit cycles, which are not revealed by the linear analysis. This approach is well-suited for validation of current commercial and military aircraft. However, there are drawbacks of this approach. First, the process is rather time-consuming and requires many well-trained control and simulation engineers. Second, most adaptive control laws lead to nonlinear, time-varying closed loop dynamics. Thus the current practice is not applicable to validating systems with adaptive control laws. There is a need for analytical tools to assess the performance of nonlinear feedback systems.

This chapter reviews an approach to reformulate nonlinear analysis problems into a form which can be solved using available software tools. The approach is applicable to nonlinear systems described by polynomial dynamics and it relies on connections between sums of squares (SOS) polynomials and positive semidefinite matrices. A polynomial p is a sum of squares if it can be expressed as $p = \sum_{i=1}^{m} f_i^2$. This connection was made in the work by Parrilo [1,2] and has led to research on computational tools for estimating regions of attraction, reachability sets, input-output gains, and robustness with respect to uncertainty. The reader is referred to [3–20] and the references contained therein. There are two key ideas in this approach. First, sufficient conditions for many nonlinear analysis problems can be formulated as set containment conditions involving either a Lyapunov function or a storage function. Second, the set containment conditions can be reformulated as polynomial non-negativity conditions using a generalized version of the S-procedure [21]. This approach will be described in more detail in the remainder of the chapter.

These nonlinear analysis techniques can fill the gap between linear analysis and nonlinear simulations. Linearized analysis is only valid over an infinitesimally small neighborhood of the equilibrium point/null input. The proposed approach provides an improvement over linearized analysis in that the results are valid over a provable region of the state/input space [20]. Moreover, the nonlinear analysis tools can complement the linear analysis tools and nonlinear simulations to provide additional confidence in the flight control law performance.

The remainder of the chapter has the following outline. The next section provides a brief review of background material including SOS polynomials, their connections to positive semidefinite matrices, and SOS programming problems. Section 3 describes the formulation of several nonlinear analysis problems in terms of optimizations with SOS constraints. This section also provides a discussion of the computational approaches to solve these problems. In Section 4 the proposed approach is applied to compare the performance of two F/A-18 control laws in their ability to suppress a loss-of-control motion known as the Falling Leaf Mode. The tools are also used to compute reachable set estimates for NASA's Generic Transport Model. Finally, conclusions are given in Section 5.

2 Sum of Squares Optimization

This section provides a brief review of computational methods for sum-of-squares polynomial optimizations. A polynomial p is a *sum of squares* (SOS) if there exist polynomials $\{f_i\}_{i=1}^{m}$ such that $p = \sum_{i=1}^{m} f_i^2$. As a simple example, $p = x^2 - 4xy + 7y^2$ is a sum of squares since $p = f_1^2 + f_2^2$ where $f_1 = (x - 2y)^2$ and $f_2 = 3y^2$. This section first presents the notation and background material. Next connections between semidefinite matrices and SOS polynomials are

described. Finally the software available to solve SOS optimization problems is discussed.

2.1 Background

Polynomial Notation

$\mathbb{R}[x]$ denotes the set of all polynomials in variables $\{x_1, \ldots, x_n\}$ with real coefficients. $\mathbb{N}$ denotes the set of nonnegative integers, $\{0, 1, \ldots\}$, and $\mathbb{N}^n$ is the set of n-dimensional vectors with entries in $\mathbb{N}$. For $\alpha \in \mathbb{N}^n$, a monomial in variables $\{x_1, \ldots, x_n\}$ is given by $x^\alpha \doteq x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$. The degree of a monomial is defined as $\deg x^\alpha \doteq \sum_{i=1}^n \alpha_i$. In this notation a polynomial in $\mathbb{R}[x]$ is simply a finite linear combination of monomials:

$$p \doteq \sum_{\alpha \in \mathcal{A}} c_\alpha x^\alpha = \sum_{\alpha \in \mathcal{A}} c_\alpha x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$$

where $c_\alpha \in \mathbb{R}$ and $\mathcal{A}$ is a finite collection of vectors in $\mathbb{N}^n$. Using the definition of deg for a monomial, the degree of p is defined as $\deg p \doteq \max_{\alpha \in \mathcal{A}} [\deg x^\alpha]$.

A polynomial p is a *sum of squares* (SOS) if there exist polynomials $\{f_i\}_{i=1}^m$ such that $p = \sum_{i=1}^m f_i^2$. The set of SOS polynomials is a subset of $\mathbb{R}[x]$ and is denoted as $\Sigma[x]$. Note that if p is a sum of squares then $p(x) \geq 0 \ \forall x \in \mathbb{R}^n$. Thus $p \in \Sigma[x]$ is a sufficient condition for a polynomial to be globally non-negative. The converse is not true, i.e. non-negative polynomials are not necessarily SOS polynomials. This is related to one of the problems posed by Hilbert in 1900 [22].

Semidefinite Programming

This brief review of semidefinite programming (SDP) is based on a survey by Vandenberghe and Boyd [23] and a monograph by Boyd, et al. [21]. A symmetric matrix $F \in \mathbb{R}^{n \times n}$ is positive semidefinite if $x^T F x \geq 0$ for all $x \in \mathbb{R}^n$. Positive semidefinite matrices are denoted by $F \succeq 0$. A semidefinite program is an optimization problem of the following form:

$$\begin{array}{ll} \min_\lambda & c^T \lambda \\ \text{subject to:} & F_0 + \sum_{k=1}^r \lambda_k F_k \succeq 0 \end{array} \tag{1}$$

The symmetric matrices $F_0, \ldots, F_r \in \mathbb{R}^{n \times n}$ and the vector $c \in \mathbb{R}^r$ are given data. The vector $\lambda \in \mathbb{R}^r$ is the decision variable and the constraint, $F_0 + \sum_{k=1}^r \lambda_k F_k \succeq 0$, is called a linear matrix inequality. Equation (1) is referred to as the primal problem. The dual associated with this primal problem is:

$$\begin{array}{ll} \max_Z & -\mathbf{Tr}\,[F_0 Z] \\ \text{subject to:} & \mathbf{Tr}\,[F_k Z] = c_k \ k = 1, \ldots, r \\ & Z \succeq 0 \end{array} \tag{2}$$

where $Z = Z^T \in \mathbb{R}^{n\times n}$ is the decision variable for the dual problem. $\mathbf{Tr}\,[\cdot]$ denotes the trace of a matrix. This dual problem can be recast in the form of Equation (1) and thus it is also a semidefinite program. While the primal and dual forms may look restrictive, these formulations are quite versatile and SDPs find applications in many problems of interest. Moreover, SDPs are convex and quality software exists to solve these problems. In particular, SeDuMi [24, 25] is a freely available MATLAB toolbox that simultaneously solves the primal and/or dual forms of a semidefinite program.

In some cases, the only goal is to find a decision variable that satisfies the linear matrix inequality constraint. These are semidefinite programming feasibility problems. The following is an example:

$$\text{Find } \lambda_1, \ldots, \lambda_r \in \mathbb{R} \text{ such that } F_0 + \sum_{k=1}^{r} \lambda_k F_k \succeq 0 \tag{3}$$

2.2 Connections Between SOS Polynomials and Semidefinite Matrices

Theorem 1 below gives a concrete statement of the connection between sums of squares and positive semidefinite matrices. Two facts that follow from [26] (refer to Theorem 1 and its preceding Lemma) are required:

1. If p is a sum of squares then p must have even degree.
2. If p is degree $2d$ ($d \in \mathbb{N}$) and $p = \sum_{i=1}^{m} f_i^2$ then $\deg f_i \leq d\ \forall i$.

Next, define z as the column vector of all monomials in variables $\{x_1, \ldots, x_n\}$ of degree $\leq d$: [4]

$$z \doteq \left[1,\ x_1,\ x_2,\ \ldots,\ x_n,\ x_1^2,\ x_1x_2,\ \ldots,\ x_n^2,\ \ldots,\ x_n^d\right]^T \tag{4}$$

There are $\binom{k+n-1}{k}$ monomials in n variables of degree k. Thus z is a column vector of length $l_z \doteq \sum_{k=0}^{d}\binom{k+n-1}{k} = \binom{n+d}{d}$. If f is a polynomial in n variables with degree $\leq d$, then f is a finite linear combination of monomials of degree $\leq d$. Consequently, there exists $a \in \mathbb{R}^{l_z}$ such that $f = a^T z$. The proof of the following theorem, introduced as the "Gram Matrix" method by Choi, Lam, and Reznick [27], is included for completeness. This result can be found more recently in [28].

Theorem 1. *Suppose $p \in \mathbb{R}[x]$ is a polynomial of degree $2d$ and z is the $l_z \times 1$ vector of monomials defined in Equation (4). Then $p \in \Sigma[x]$ if and only if there exists a symmetric matrix $Q \in \mathbb{R}^{l_z \times l_z}$ such that $Q \succeq 0$ and $p = z^T Q z$.*

[4] Any ordering of the monomials can be used to form z. In Equation (4), x^α precedes x^β in the definition of z if:

$\deg x^\alpha < \deg x^\beta$ or $\deg x^\alpha = \deg x^\beta$ and the first nonzero entry of $\alpha - \beta$ is > 0

Proof:

($\Rightarrow$) If p is a SOS, then there exists polynomials $\{f_i\}_{i=1}^m$ such that $p = \sum_{i=1}^m f_i^2$. As noted above, $\deg f_i \leq d$ for all i. For each f_i there exists a vector $a_i \in \mathbb{R}^{l_z}$ such that $f_i = a_i^T z$. Define the matrix $A \in \mathbb{R}^{l_z \times m}$ whose i^{th} column is a_i and define $Q \doteq AA^T \succeq 0$. Then $p = z^T Q z$.

($\Leftarrow$) Assume there exists $Q = Q^T \in \mathbb{R}^{l_z \times l_z}$ such that $Q \succeq 0$ and $p = z^T Q z$. Define $m \doteq rank(Q)$. There exists a matrix $A \in \mathbb{R}^{l_z \times m}$ such that $Q = AA^T$. Let a_i denote the i^{th} column of A and define the polynomials $f_i \doteq z^T a_i$. Then $p = z^T(AA^T)z = \sum_{i=1}^m f_i^2$.

■

2.3 Software for SOS Optimizations

A sum-of-squares program is an optimization problem with a linear cost and SOS constraints on the decision variables [29]:

$$\min_{u \in \mathbb{R}^n} c_1 u_1 + \cdots + c_n u_n \tag{5}$$

subject to:

$$a_{k,0}(x) + a_{k,1}(x)u_1 + \cdots + a_{k,n}(x)u_n \in \Sigma[x] \; k = 1, \ldots N_s$$

The polynomials $\{a_{k,j}\}$ are given as part of the optimization data and $u \in \mathbb{R}^n$ are decision variables. In Section 3 it will be shown that many nonlinear analysis problems can be posed within this optimization framework.

Theorem 1 provides the link to convert an SOS program into a semidefinite-programming problem. For example, the constraint $a_{k,0}(x) + a_{k,1}(x)u_1 + \cdots + a_{k,n}(x)u_n \in \Sigma[x]$ can be equivalently written as:

$$a_{k,0}(x) + a_{k,1}(x)u_1 + \cdots + a_{k,n}(x)u_n = z^T Q z \tag{6}$$

$$Q \succeq 0 \tag{7}$$

Q is a new matrix of decision variables that is introduced when an SOS constraint is converted to an LMI constraint. Equating the coefficients of $z^T Q z$ and $a_{k,0}(x) + a_{k,1}(x)u_1 + \cdots + a_{k,n}(x)u_n$ imposes linear equality constraints on the decision variables u and Q. Thus, Equation (6) can be rewritten as a set of linear equality constraints on the decision variables. All SOS constraints in Equation (5) can be replaced in this fashion with linear equality constraints and LMI constraints. As a result, the SOS program in Equation (5) can be written in the SDP dual form (Equation (2)).

While this may appear cumbersome, there is software available to perform the conversion. For example, SOSTOOLS [29], Yalmip [30], and SOSOPT [31] are freely available MATLAB toolboxes for solving SOS optimizations. These packages allow the user to specify the polynomial constraints using a symbolic toolbox. Then they convert the SOS optimization into an SDP which is solved with SeDuMi [24, 25] or another freely available SDP solver. Finally these toolboxes convert the solution of the SDP back to a polynomial solution.

A drawback is that the size of the resulting SDP grows rapidly if the SOS optimization involves polynomials with many variables and/or high degree. For a generic degree $2d$ polynomial p in n variables, the Gram matrix representation involves $l_z := \binom{n+d}{d}$ monomials. An SOS constraint on p is enforced via a positive semidefinite constraint on the $l_z \times l_z$ Gram matrix Q. For example, for a generic degree $2d = 8$ polynomial in $n = 8$ variables, the Gram matrix has dimension $l_z = 495$. The size of this positive semidefinite constraint is at or near the limits of current semidefinite programming solvers. While various techniques can be used to exploit the problem structure [32], this computational growth is a generic trend in SOS optimizations. Some methods which use simulation to ease this computational growth have been developed [16–18].

3 Nonlinear Analysis Tools

Many nonlinear analysis problems can be formulated as sum of squares programming problems. This connection was made in the work by Parrilo [1, 2] and has led to research on computational tools for estimating regions of attraction, reachability sets, input-output gains, and robustness with respect to uncertainty. The reader is referred to [3–20] and the references contained therein. The key idea is that sufficient conditions for these nonlinear analysis problems can typically be formulated as set containment conditions involving either a Lyapunov function or a storage function. The set containment conditions can be reformulated as polynomial non-negativity conditions using a generalized version of the S-procedure [21]. These problems can then be solved as SOS programs since SOS polynomials are globally non-negative. In this section this approach is described in more detail for region of attraction estimation, $L_2 - L_2$ input-output gain calculation, and estimation of reachability sets with L_2 bounded inputs. Analysis problems with different signal norms and/or with model uncertainty are described in the references given above. Software to perform all analyses described in this section is available at [31].

3.1 Region of Attraction Estimation

This section describes the computational method to estimate a region of attraction (ROA). Consider an autonomous nonlinear dynamical system of the form:

$$\dot{x} = f(x),\ x(0) = x_0 \tag{8}$$

where $x \in \mathbb{R}^n$ is the state vector and $f : \mathbb{R}^n \to \mathbb{R}^n$ is a multivariable polynomial. Assume that $x = 0$ is a locally asymptotically stable equilibrium point. Formally, the ROA is defined as:

$$\mathcal{R} = \left\{ x_0 \in \mathbb{R}^n :\ \text{If } x(0) = x_0 \text{ then } \lim_{t \to \infty} x(t) = 0 \right\} \tag{9}$$

Computing the exact ROA for nonlinear dynamical systems is difficult. There has been significant research devoted to estimating invariant subsets of the ROA [1, 3, 4, 15, 33–37]. The approach taken here is to restrict the search to ellipsoidal approximations of the ROA. Given an $n \times n$ matrix $N = N^T > 0$, define the shape function $p(x) := x^T N x$ and level set $\mathcal{E}_\beta := \{x \in \mathbb{R}^n \, : \, p(x) \leq \beta\}$. $p(x)$ defines the shape of the ellipsoid and β determines the size of the ellipsoid $\mathcal{E}_\beta$. The choice of p is problem dependent and reflects dimensional scaling information as well as the importance of certain directions in the state space. Given the shape function p, the problem is to find the largest ellipsoid $\mathcal{E}_\beta$ contained in the ROA:

$$\begin{aligned} \beta^* = & \max \beta \\ & \text{subject to: } \mathcal{E}_\beta \subset \mathcal{R} \end{aligned} \tag{10}$$

Determining the best ellipsoidal approximation to the ROA is still a challenging computational problem. Instead, lower and upper bounds for β^* satisfying $\underline{\beta} \leq \beta^* \leq \bar{\beta}$ are computed. If the lower and upper bounds are close then the largest ellipsoid level set, defined by Equation (10), has been approximately computed.

The upper bounds are computed via a search for initial conditions leading to divergent trajectories. If $\lim_{t\to\infty} x(t) = +\infty$ when starting from $x(0) = x_{0,div}$ then $x_{0,div} \notin \mathcal{R}$. If $\bar{\beta}_{div} := p(x_{0,div})$ then $\mathcal{E}_{\bar{\beta}_{div}} \not\subset \mathcal{R}$ which implies $\beta^* \leq \bar{\beta}_{div}$ and $\mathcal{E}_{\beta^*} \subseteq \mathcal{E}_{\bar{\beta}_{div}}$. An exhaustive Monte Carlo search is used to find the tightest possible upper bound on β^*. Specifically, random initial conditions are chosen starting on the boundary of a large ellipsoid: Choose x_0 satisfying $p(x_0) = \beta_{try}$ where β_{try} is sufficiently large that $\beta_{try} \gg \beta^*$. If a divergent trajectory is found, the initial condition is stored and an upper bound on β^* is computed. β_{try} is then decreased by a factor of 0.995 and the search continues until a maximum number of simulations is reached. There is a tradeoff involved in choosing the factor 0.995. A smaller factor results in a larger reduction of the upper bound for each divergent trajectory but it typically limits the accuracy of the upper bound. No divergent trajectories can be found when $\beta_{try} < \beta^*$ and this roughly limits the upper bound accuracy to β^*/(factor). The value of 0.995 is very close to one and was chosen to obtain an accurate upper bound on β^*. $\bar{\beta}_{MC}$ will denote the smallest upper bound computed with this Monte Carlo search.

The lower bounds are computed using Lyapunov functions and recent results connecting sums-of-squares polynomials to semidefinite programming. Computing these bounds requires the vector field $f(x)$ in Equation (8) to be a polynomial function. The computational algorithm is briefly described here and full algorithmic details are provided elsewhere [5–7, 10, 14, 16, 18, 19]. Lemma 1 is the main Lyapunov theorem used to compute lower bounds on β^*. This specific lemma is proved by [14] but very similar results are given in textbooks, e.g. by [38].

Lemma 1. *If there exists $\gamma > 0$ and a polynomial $V : \mathbb{R}^n \to \mathbb{R}$ such that:*

$$V(0) = 0 \text{ and } V(x) > 0 \ \forall x \neq 0 \tag{11}$$

$$\Omega_\gamma := \{x \in \mathbb{R}^n \ : \ V(x) \leq \gamma\} \text{ is bounded.} \tag{12}$$

$$\Omega_\gamma \subset \{x \in \mathbb{R}^n \ : \ \nabla V(x) f(x) < 0\} \cup \{0\} \tag{13}$$

then for all $x \in \Omega_\gamma$, the solution of Equation (8) exists, satisfies $x(t) \in \Omega_\gamma$ for all $t \geq 0$, and $\Omega_\gamma \subset \mathcal{R}$.

A function V, satisfying the conditions in Lemma 1 is a Lyapunov function and Ω_γ provides an estimate of the region of attraction. If $x = 0$ is asymptotically stable, a linearization can be used to compute a Lyapunov function. Let $A := \frac{\partial f}{\partial x}\Big|_{x=0}$ be the linearization of the dynamics about the origin and compute $P > 0$ that solves the Lyapunov equation $A^T P + PA = -I$. $V_{LIN}(x) := x^T P x$ is a quadratic Lyapunov function that satisfies the conditions of Lemma 1 for sufficiently small $\gamma > 0$. V_{LIN} can be used to compute a lower bound on β^* by solving two maximizations:

$$\begin{aligned} \gamma^* := & \max \gamma \\ & \text{subject to: } \Omega_\gamma \subset \{x \in \mathbb{R}^n \ : \ \nabla V_{LIN}(x) f(x) < 0\} \end{aligned} \tag{14}$$

$$\begin{aligned} \underline{\beta} := & \max \beta \\ & \text{subject to: } \mathcal{E}_\beta \subset \Omega_{\gamma^*} \end{aligned} \tag{15}$$

The first maximization finds the largest level set of V_{LIN}, Ω_{γ^*}, such that Lemma 1 can be used to verify $\Omega_{\gamma^*} \subseteq \mathcal{R}$. The second maximization finds the largest ellipsoid $\mathcal{E}_{\underline{\beta}}$ contain within Ω_{γ^*}.

The set containment constraints can be replaced with a sufficient condition involving non-negative functions [14]. The next Lemma provides this sufficient condition. This lemma is a generalization of the S-procedure which has been frequently applied in control theory [21]. The function s appearing in the Lemma is called a multiplier.

Lemma 2. *Define two sets $A := \{x \in R^n \ : \ f_A(x) \geq 0\}$ and $B := \{x \in R^n \ : \ f_B(x) \geq 0\}$. If there exists a function $s(x) \geq 0 \ \forall x$ such that $f_B(x) - f_A(x)s(x) \geq 0 \ \forall x$ then $A \subseteq B$.*

Proof:

Assume there exists a function $s(x) \geq 0 \, \forall x$ such that $f_B(x) - f_A(x)s(x) \geq 0$ $\forall x$. Take any $x \in A$. Then $f_B(x) \geq f_A(x)s(x) \geq 0$. Thus x is also in B. ■

The Positivstellensatz is a result from algebraic geometry that can be used to construct necessary and sufficient algebraic conditions for this set

containment condition (See [1, 2] and the references contained therein). In addition, more general set containment constraints, e.g. sets with nonstrict inequalities, can be handled. The condition in Lemma 2 can be interpreted as a simplification of the most general Positivstellensatz conditions. This simple condition will be used in the remainder of the paper since it typically requires much less computation than the most general Positivstellensatz conditions.

Applying this Lemma to $\mathcal{E}_\beta \subset \Omega_{\gamma^*}$ in Optimization ((15)) leads to:

$$\begin{aligned} \underline{\beta} := \max_{\beta,\ s(x)} \beta & \qquad (16) \\ \text{subject to: } & s(x) \geq 0 \ \forall x \\ & -(\beta - p(x))\, s(x) + (\gamma^* - V_{LIN}(x)) \geq 0 \ \forall x \end{aligned}$$

The function $s(x)$ is a decision variable of the optimization, i.e. its coefficients are decision variables that are computed as part of the optimization. It is straight-forward to show that the two non-negativity conditions in Optimization ((16)) are a sufficient condition for the set containment condition in Optimization ((15)). If $s(x)$ is restricted to be a polynomial then both constraints involve the non-negativity of polynomial functions. Restricting a polynomial to be SOS is a sufficient condition for the polynomial to be non-negative. Replacing the non-negativity conditions in Optimization ((16)) with SOS constraints leads to an SOS optimization problem:

$$\begin{aligned} \underline{\beta} := \max \beta & \qquad (17) \\ \text{subject to: } & s(x) \in \Sigma[x] \\ & -(\beta - p(x))s(x) + (\gamma^* - V_{LIN}(x)) \in \Sigma[x] \end{aligned}$$

As described in Section 2 there is freely available software to solve such SOS optimizations. $\underline{\beta}_{LIN}$ will denote the lower bound obtained from Optimization ((17)) using the quadratic Lyapunov function obtained from linearized analysis.

Unfortunately, $\underline{\beta}_{LIN}$ is usually orders of magnitude smaller than the upper bound $\bar{\beta}_{MC}$. Better lower bounds $\underline{\beta}$ can be computed by also optimizing the choice of the Lyapunov function. This leads to an optimization problem that is bilinear in the Lyapunov function and a multiplier function. Specifically, the SOS constraint that arises due to the set containment condition in Equation (14) is:

$$-(\gamma - V)s_2 - (\nabla V \cdot f + l_2) \in \Sigma[x] \qquad (18)$$

where s_2 is another SOS multiplier function and and $l_2(x) = -\epsilon_2 x^T x$ where ϵ_2 is a small positive constant on the order of 10^{-6}. If both V and s_2 are allowed to vary then the term Vs_2 in this constraint will be bilinear in the unknown coefficients of V and s_2. Thus optimizing over V and the set containment multipliers is not a convex problem due to this bilinearity and hence heuristic

solution methods are required. Several methods to compute better Lyapunov functions exist, including V-s iterations [5–7, 10], bilinear optimization [14], and the use of simulation data [16, 18]. The V-s iteration is now described in more detail. The Lyapunov function $V(x)$ in the iteration is initialized with the linearized Lyapunov function V_{LIN}. The iteration also uses functions $l_1(x) = -\epsilon_1 x^T x$ and $l_2(x) = -\epsilon_2 x^T x$ where ϵ_1 and ϵ_2 are small positive constants on the order of 10^{-6}. The V-s iteration algorithm steps are provided below.

1. γ **Step**: Hold V fixed and solve for s_2 and γ^*

$$\gamma^* := \max_{s_2 \in \Sigma[x], \gamma} \gamma \quad \text{s.t. } -(\gamma - V)s_2 - (\nabla V \cdot f + l_2) \in \Sigma[x]$$

2. $\underline{\beta}$ **Step**: Hold V, γ^* fixed and solve for s_1 and $\underline{\beta}$

$$\underline{\beta} := \max_{s_1 \in \Sigma[x], \beta} \beta \quad \text{s.t. } -(\beta - p)s_1 + (\gamma^* - V) \in \Sigma[x]$$

3. V **step**: Hold s_1, s_2, $\underline{\beta}$, γ^* fixed and solve for V satisfying:

$$\begin{aligned} &-(\gamma^* - V)s_2 - (\nabla V \cdot f + l_2) \in \Sigma[x] \\ &-(\underline{\beta} - p)s_1 + (\gamma^* - V) \in \Sigma[x] \\ &V - l_1 \in \Sigma[x], V(0) = 0 \end{aligned}$$

4. Repeat as long as the lower bound $\underline{\beta}$ continues to increase.

Software and additional documentation on the V-s iteration is provided at [31]. The basic idea of the iteration is to avoid the bilinearity in Vs_2 by holding either s_2 or V fixed. Each step of this iteration is a linear SOS optimization that can be solved with available software. In the V-s iteration, the Lyapunov functions are allowed to have polynomial degree greater than two. Increasing the degree of the Lyapunov function will improve the lower bound at the expense of computational complexity. The computational time grows rapidly with the degree of the Lyapunov function. Simulation data can also be used to construct a good initial candidate V for this iteration [17].

The V step requires additional discussion. An interior-point solver is used to find a feasible solution to the LMI feasibility problem in the V step. The Lyapunov function V that is used in the γ and β steps will be feasible for the constraints in the V step. Thus it possible for the solver to simply return the same Lyapunov function that was used in the γ and β steps. While this is possible, it typically happens that the solver returns a different V that allows both γ and β to be increased at the next iteration. An informal justification for this behavior is now given. The constraint $-(\gamma^* - V)s_2 - (\nabla V \cdot f + l_2) \in \Sigma[x]$ is active after the γ step. In the V step an interior point method obtains a new feasible V by computing decision variables that are at the analytic center of the set specified by the LMI constraints. The V step typically returns a feasible V that is "pushed away" from the constraints. Loosely, the new feasible V satisfies $-(\gamma^* - V)s_2 - \left(\nabla V \cdot f + \tilde{l}_2\right) \in \Sigma[x]$. where $\tilde{l}_2 \geq l_2$. $\tilde{l}_2 \geq l_2$ means

the next γ step has freedom to increase γ while still satisfying the constraint with l_2. A more formal theory for the behavior of this feasibility step is still an open question.

3.2 Input-Output Gains

This section describes a computational method to compute L_2 input-output gains. More details on this problem as well as computing gains with other signal norms can be found in [13,19,20]. Consider nonlinear dynamical systems of the form:

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x) \end{aligned} \tag{19}$$

where $x \in \mathbb{R}^{n_x}$ is the state vector, $u \in \mathbb{R}^{n_u}$ is the input, and $y \in \mathbb{R}^{n_y}$ is the output. Assume f is a $n_x \times 1$ polynomial function of x and u such that $f(0,0) = 0$. Also assume that h is an $n_y \times 1$ polynomial function of x such that $h(0) = 0$. Denote this system by $\mathcal{S}$.

Define the L_2 norm of a signal as $\|u\|_2 := \sqrt{\int_0^\infty u^T(t)u(t)dt}$. u is an L_2 signal if this integral is finite. The L_2-L_2 input-output gain of the system is defined as $\|\mathcal{S}\| := \sup_{u \in L_2, \|u\|_2 \neq 0} \frac{\|y\|_2}{\|u\|_2}$. A "local" input-output gain of the system can also be defined as $\|\mathcal{S}\|_R := \sup_{u \in L_2, 0 < \|u\|_2 \leq R} \frac{\|y\|_2}{\|u\|_2}$. For linear systems the magnitude of the output scales proportionally with the magnitude of the input and hence the ratio $\frac{\|y\|_2}{\|u\|_2}$ does not depend on $\|u\|_2$. Thus $\|\mathcal{S}\|_R = \|\mathcal{S}\|$ for all $R > 0$. For a nonlinear system, the local gain depends on the magnitude of the input and hence $\|\mathcal{S}\|_R$ and $\|\mathcal{S}\|$ need not be equal. The class of possible inputs increases with increasing values of R and so $\|\mathcal{S}\|_R$ is a monotonically increasing function of R and $\|\mathcal{S}\|_R \geq \|\mathcal{S}\|$ for all $R > 0$.

Lemma 3 provides a sufficient condition for the local L_2-L_2 input-output gain to be less than γ. This specific lemma can be found in [13, 19, 20] but similar results are given in textbooks [38, 39].

Lemma 3. *If there exists a $\gamma > 0$ and a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ such that:*

- *$V(0) = 0$ and $V(x) \geq 0 \quad \forall x \in \mathbb{R}^{n_x}$*
- *$\{(x,u) \in \mathbb{R}^{n_x+n_u} : V(x) \leq R^2\} \subseteq \{(x,u) \in \mathbb{R}^{n_x+n_u} : \frac{\partial V}{\partial x} f(x,u) \leq u^T u - \gamma^{-2} y^T y\}$*

then $x(0) = 0$ and $\|u\|_2 \leq R$ implies $\|y\|_2 \leq \gamma \|u\|_2$.

Proof:

A sketch of the proof is provided. Assume that $\frac{\partial V}{\partial x} f(x,u) \leq u^T u - \gamma^{-2} y^T y$ holds along the trajectories of the system $\mathcal{S}$ from time 0 to T. Integrating with respect to time yields:

$$V(x(T)) - V(x(0)) \leq \int_0^T \left(u^T u - \gamma^{-2} y^T y\right) dt \tag{20}$$

If $x(0) = 0$ and $\|u\|_2 \leq R$ then Equation (20) implies that $V(x(T)) \leq \|u\|_2^2 \leq R^2$. Thus the state trajectories satisfy $V(x(T)) \leq R^2 \; \forall T \geq 0$ and it is valid to assume $\frac{\partial V}{\partial x} f(x,u) \leq u^T u - \gamma^{-2} y^T y$ holds along the system trajectories. Moreover, Equation (20) implies that $\int_0^T \left(y^T y\right) dt \leq \gamma^2 \int_0^T \left(u^T u\right) dt$ since $V(0) = 0$ and $V(x) \geq 0 \; \forall x$. $\|y\|_2 \leq \gamma \|u\|_2$ follows by letting $T \to \infty$. ■

Lemma 3 provides a sufficient condition to prove $\|S\|_R \leq \gamma$ in terms of a storage function, V. This lemma involves one non-negativity condition on the storage function and one set containment condition. The generalized S-procedure (Lemma 2) can again be used to convert the set containment condition into a function non-negativity constraint. This leads to the following optimization for computing upper bounds on the local L_2-L_2 gain:

$$\gamma^* := \min_{V,s,\gamma} \gamma \tag{21}$$

subject to:

$$s(x,u) \in \Sigma[x,u], V(x) \in \Sigma[x], V(0) = 0 \tag{22}$$

$$u^T u - \gamma^{-2} h(x)^T h(x) - \nabla V \cdot f(x,u) - s(x,u)\left(R^2 - V(x)\right) \in \Sigma[x,u] \tag{23}$$

The constraint in Equation (23), if satisfied, ensures that $\{(x,u) \in \mathbb{R}^{n_x+n_u} : V(x) \leq R^2\} \subseteq \{(x,u) \in \mathbb{R}^{n_x+n_u} : \frac{\partial V}{\partial x} f(x,u) \leq u^T u - \gamma^{-2} y^T y\}$. Since SOS polynomials are non-negative everywhere this follows by applying the generalized S-procedure in Lemma 2. By Lemma 3, $\|S\|_R \leq \gamma$ for any γ for which the constraints are valid. γ^* is the smallest upper bound on $\|S\|_R$ which can be found with this sufficient condition.

This optimization problem involves SOS constraints on $s(x,u)$ and $V(x)$ (Equation (22)). The coefficients of the polynomials $s(x,u)$ and $V(x)$ are decision variables in the optimization. The constraint in Equation (23) is an SOS constraint on a polynomial of x and u. Unfortunately this constraint is bilinear in the decision variables since it involves a term of the form $s(x,u) \cdot V(x)$. This problem is nonconvex in the decision variables and again heuristics are required. This can be solved directly using bilinear matrix inequality solvers [17]. Alternatively, a V-s iteration can be formulated for this input-output gain problem. This iteration can be initialized with V as the quadratic storage function obtained from linear analysis [6]. Simulation data can also be used to construct a good initial candidate V for this iteration [17].

3.3 Reachable Sets

Again consider nonlinear dynamical systems, $\mathcal{S}$, in the form of Equation (19). The reachable set $\mathcal{G}(\gamma)$ is the set of states that can be reached from $x(0) = 0$ with an input satisfying $\|u\|_2^2 \leq \gamma$. Formally,

$$\mathcal{G}(\gamma) := \{x_f \in \mathbb{R}^n \ : \ \exists T, \ u(t) \text{ defined on } [0,T] \text{ s.t. } ||u||_2^2 \leq \gamma \text{ and } x(T) = x_f\} \tag{24}$$

The reachable set depends on the input energy to the system. Lemma 4 provides a sufficient condition for computing an outer bound on $\mathcal{G}(\gamma)$ [14, 19]. Similar reachable set results for linear systems can be found in [21].

Lemma 4. *If there exists a $\gamma > 0$ and a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ such that:*

- $V(0) = 0$ *and* $V(x) \geq 0 \ \ \forall x \in \mathbb{R}^{n_x}$
- $\{(x,u) \in \mathbb{R}^{n_x+n_u} \ : \ V(x) \leq \gamma\} \subseteq \{(x,u) \in \mathbb{R}^{n_x+n_u} \ : \ \nabla V \cdot f(x,u) \leq u^T u\}$

then $\mathcal{G}(\gamma) \subseteq \{x \ : \ V \leq \gamma\} := \Omega_\gamma$

As in the previous sections, the generalized S-procedure can be used to convert the set containment constraint into an SOS constraint. Also, an ellipsoidal approximation of $Reach(\gamma)$ is easier to visualize and understand. As in the ROA estimation problem, a shape function $p(x) := x^T N x$ and level set $\mathcal{E}_\beta := \{x \in \mathbb{R}^n \ : \ p(x) \leq \beta\}$ can be introduced. The problem is then to find the smallest ellipsoid $\mathcal{E}_\beta$ that contains $\mathcal{G}(\gamma)$. $\mathcal{E}_\beta$ then provides an outer approximation for the reachable set. This leads to the following optimization problem:

$$\begin{aligned} \beta^* := \min_{V,s,\gamma,\beta} \ & \beta \\ & \text{subject to:} \\ & s(x,u) \in \Sigma[x,u], \ V(x) \in \Sigma[x], \ V(0) = 0 \\ & (\beta - p) - (\gamma - V)s_2 \in \Sigma[x] \\ & -((\nabla V \cdot f - u^T u) + (\gamma - V)s_1) \in \Sigma[x,u] \end{aligned} \tag{25}$$

Again this problem is bilinear due to the Vs_1 term in the last constraint. A V-s iteration can be used to compute an ellipsoidal bound $\mathcal{E}_\beta$ on $\mathcal{G}(\gamma)$ for any value of γ.

3.4 Summary of Approach

The computational tools for other nonlinear problems (estimating regions of attraction, reachability sets, input-output gains with other signal norms, and robustness with respect to uncertainty) all essentially follow the same steps as used in the previous sections. Specifically, a Lyapunov or storage function type theorem is used to derive a sufficient condition for the nonlinear system to have a particular performance/stability property. Lyapunov and storage functions are naturally restricted to be positive definite and this can

be enforced using SOS constraints. Additional conditions can typically be formulated as set containment conditions. These set containment conditions can then be converted into function non-negativity constraints using the generalized S-procedure. Since SOS polynomials are non-negative everywhere, the non-negativity constraints can be relaxed and written as SOS constraints. In many cases this sequence of constraint reformulations leads to either a linear or bilinear SOS programming problem which yields a bound on a particular systems property (e.g. inner approximations to regions of attraction or upper bounds on system gains). Bilinear problems can be solved using one of the methods described above. Simulations or gradient searches can be used to compute dual bounds (e.g. outer approximations to regions of attraction or lower bounds on system gains). For example, lower bounds on the local gain can be computed using a power method derived for a finite horizon optimal control problem [40]. This approach provides an improvement over linearized analysis in that the results are valid over a provable region of the state/input space rather than for an infinitesimally small neighborhood of the equilibrium point/null input. Further details on this statement can be found in [20].

4 Examples

This section performs nonlinear analyzes for an F/A-18 and NASA's Generic Transport Model. The software used to perform these analyses is available at [31].

4.1 ROA Estimation for an F/A-18

The US Navy F/A-18 A/B/C/D Hornet aircraft with the original baseline flight control law experienced a number of out-of-control flight departures since the early 1980's. Many of these incidents have been described as a falling leaf motion of the aircraft [41]. The falling leaf motion has been studied extensively to investigate the conditions that lead to this behavior. The complex dynamics of the falling leaf motion and lack of flight data from the departure events pose a challenge in studying this motion. An extensive revision of the baseline control law was performed by NAVAIR and Boeing in 2001 to suppress departure phenomenon, improve maneuvering performance and to expand the flight envelope [41]. The revised control law was implemented on the F/A-18 E/F Super Hornet aircraft after successful flight tests. These flight tests included aggressive maneuvers that demonstrated successful suppression of the falling leaf motion by the revised control law. This section uses nonlinear region of attraction estimation to compare the closed-loop stability properties of the baseline and revised control laws. Additional details on this analysis can be found in [42].

The falling leaf motion of an aircraft can be characterized as large, coupled out-of-control oscillations in the roll (p) and yaw (r) direction combined

with large fluctuations in angle-of-attack (α) and sideslip (β) [41, 43]. Figure 1 shows the main characteristics of the falling leaf motion [41, 43]. This out-of-control mode exhibits periodic in-phase roll and yaw rates with large amplitude fluctuations about small or zero mean. The roll and yaw rate generation is mainly due to the large sideslip oscillation. During large sideslip and angle-of-attack motion, the dihedral effect (roll caused by sideslip) of the aircraft wings becomes extremely large and the directional stability becomes unstable. The like-signs of these two values are responsible for the in-phase motion. The roll rate motion can easily reach up to $\pm 120°/s$, while the yaw rate motion can fluctuate around $\pm 50°/s$. During this motion, the value of angle-of-attack can reach up to $\pm 70°$ with sideslip oscillations between $\pm 40°$ [43]. The required aerodynamic nose-down pitching moment is exceeded by the pitch rate generation due to the inertial coupling of the in-phase roll and yaw rates. The reduction in pitching moment is followed by a reduction in normal force, eventually causing a loss of lift in the aircraft. A distinguishing feature of the falling leaf motion is that α vs. β plot produces a mushroom shape curve as seen in Figure 1. For more details on the falling leaf motion, readers are encouraged to refer to the papers by Jaramillo & Ralston [43] and Heller, David & Holmberg [41].

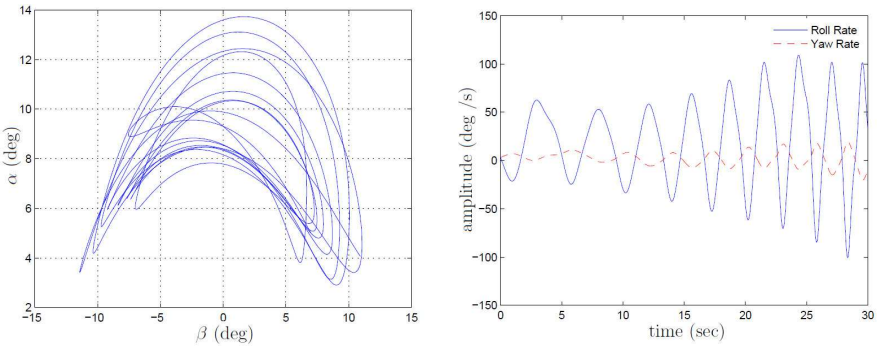

Fig. 1. Characteristic Behavior of Falling Leaf Motion

The F/A-18 Hornet is a high performance, twin engine fighter aircraft built by the McDonnell Douglas (currently known as the 'Boeing') Corporation. Each engine is a General Electric, F404-GE-400 rated at 16,100-lbf of static thrust at sea level. The aircraft features a low sweep trapezoidal wing planform with 400 ft^2 area and twin vertical tails [44]. Table 1 lists the aerodynamic reference and physical parameters of the aircraft. The conventional F/A-18 Hornet has five pairs of control surfaces: stabilators, rudders, ailerons, leading edge flaps, and trailing edge flaps. However, only the symmetric stabilator, differential aileron and differential rudder are considered as control effectors for the analysis performed in this section. Longitudinal con-

trol or pitch control is provided by the symmetric deflection of the stabilators. Deflection of differential ailerons is used to control the roll or lateral direction, while differential deflection of rudders provide directional or yaw control. There is a coupling between roll and yaw dynamics.

Table 1. Aircraft Parameters

Wing Area, S_{ref}	400 ft^2
Mean Aerodynamic Chord (c)	11.52 ft
Wing Span, b_{ref}	37.42 ft
Weight	33310 lbs
I_{xx}	23000 slug-ft^2
I_{yy}	151293 slug-ft^2
I_{zz}	169945 slug-ft^2
I_{xz}	-2971 slug-ft^2

The conventional 6DOF aircraft equations of motion are described in Stengel [45], Cook [46], and Napolitano and Spagnuolo [47] are primarily driven by the aerodynamic forces and moments acting on the aircraft. Many flight experiments have been performed to estimate the stability and control derivatives of the F/A-18 High Alpha Research Vehicle (HARV) [48–51]. The F/A-18 HARV has similar aerodynamic characteristics as the F/A-18 Hornet [52] with the exception of the F/A-18 HARV having thrust vectoring control. Hence, the F/A-18 HARV aerodynamic data are used to construct the aerodynamic coefficient data.

The nonlinear region of attraction analysis requires the aircraft dynamics to be described via a polynomial model. The computational burden of SOS optimization also restricts the model to cubic degree polynomials. Hence, a six state cubic degree polynomial model of the F/A-18 aircraft for roll-coupled maneuvers [53] was constructed for the region of attraction estimation. The polynomial model captures the key characteristics of the full 6 DOF model. This polynomial model is derived based on the characteristics of the falling leaf motion. During the falling-leaf motion, the velocity is usually on the order of 250 ft/s [43]. Hence velocity is assumed to be constant and equal to 250 ft/s in the construction of the 6-state polynomial model. Aggressive maneuvers, like bank turns, are more likely to put the aircraft in the falling leaf motion compared to straight and level flight. Hence, steady bank turn maneuvers with zero climb rate ($\dot{\theta} = 0$) are considered. As a result two other states, pitch angle (θ) and yaw angle (ψ), can be assumed constant in the six state model. Thrust effects in the sideslip direction are also neglected. Small angle approximations are used for the trigonometric terms in the full 6 DOF model to derive a polynomial representation of the aircraft dynamics. Finally, a polynomial least squares fit of the aerodynamic data over a gridded α - β space of $-20^\circ \leq \beta \leq 20^\circ$, and $-10^\circ \leq \alpha \leq 40^\circ$ is performed to

obtain the cubic polynomial model. Further details of this polynomial model approximation are provided in [42].

The baseline controller structure for the F/A-18 aircraft closely follows the *Control Augmentation System (CAS)* presented in the report by Buttrill, Arbuckle, and Hoffler [44]. The revised F/A-18 flight control law is described in the papers by Heller, David, & Holmberg [41] and Heller, Niewoehner, & Lawson [54]. The objective of the revised flight control law was to improve the departure resistance characteristics and full recoverability of the F/A-18 aircraft without sacrificing the maneuverability of the aircraft [41]. The significant change in the revised control law was the additional sideslip (β in rad) and sideslip rate ($\dot{\beta}$ in rad/s) feedback to the aileron actuators. The sideslip feedback plays a key role in increasing the lateral stability in the $30-35^{\circ}$ range of angle-of-attack. The sideslip rate feedback improves the lateral-directional damping. Hence, sideslip motion is damped even at high angles-of-attack. This feature is key to eliminating the falling leaf mode, which is an aggressive form of in-phase Dutch-roll motion. There are no direct measurements of sideslip and sideslip rate. Therefore, these signals are estimated for feedback. The sideslip and the sideslip rate feedback signals are computed based on already available signals from the sensors and using the kinematics of the aircraft.

The remainder of this section compares the regions of attraction for the baseline and revised control laws. The V-s iteration described in Section 3.1 is used to compute these ROA estimates. The analysis is performed for the F/A-18 aircraft operating at a steady ($\beta = 0$) bank turn of $\phi = 60^{o}$. This ROA analysis uses the cubic polynomial models for 60^{o} steady bank turn maneuver. The ordering of the state vector is $x^{T} := [\beta,\ p,\ r,\ \phi,\ \alpha,\ q,\ x_{c}]$. The shape matrix for the ellipsoid is chosen to be $N := (5)^{2} \cdot diag(5^{o}, 20^{o}/s, 5^{o}/s, 45^{o}, 25^{o}, 25^{o}/s, 25^{o})^{-2}$. This roughly scales each state by the maximum magnitude observed during flight conditions. The factor of $(5)^{2}$ normalizes the largest entry of the matrix N to be equal to one. The ellipsoid, $x^{T}Nx = \underline{\beta}$, defines the set of initial conditions for which the control law will bring the aircraft back to its trim point. This provides valuable information about the closed-loop stability characteristics. If the aircraft is perturbed due to a wind gust or other upset condition but remains in the ellipsoid then the control law will recover the aircraft and bring it back to trim. In other words the ellipsoid defines a safe flight envelope for the F/A-18. Hence, the ROA provides a measure of how much perturbation the aircraft can tolerate before it becomes unstable. The value of the $\underline{\beta}$ can be thought of as 'nonlinear stability margin'.

As previously mentioned, increasing the degree of the Lyapunov function will improve the lower bound estimate of the ROA. ROA bounds were first computed using the quadratic Lyapunov function from linearized analysis. This method has been proposed for validation of flight control laws [55]. The bound $\underline{\beta}_{LIN} = 8.05 \times 10^{-5}$ was computed for the baseline control law and $\underline{\beta}_{LIN} = 1.91 \times 10^{-4}$ for the revised control. Unfortunately these lower bounds are not particularly useful since they are two to three orders of mag-

nitude smaller than the corresponding upper bounds computed via Monte Carlo search. Next, lower bounds were computed with the V-s iteration using quadratic (degree 2) and quartic (degree 4) Lyapunov functions. The V-s iteration with quadratic Lyapunov functions gives $\underline{\beta}_2 = 3.45 \times 10^{-3}$ for the baseline control law and $\underline{\beta}_2 = 9.43 \times 10^{-3}$ for the revised control law. These bounds took several minutes to compute. The V-s iteration with quartic Lyapunov functions is $\underline{\beta}_4 = 1.24 \times 10^{-2}$ for the baseline control law and $\underline{\beta}_4 = 2.53 \times 10^{-2}$ for the revised control law. The bounds for the baseline and revised controllers took seven and five hours, respectively, to compute. These bounds are significantly larger than the bounds obtained for the linearized Lyapunov function. A sixth order Lyapunov function would lead to improved lower bounds but with a significant increase in computation time.

The Monte Carlo search, described in Section 3.1, was used to compute an upper bound on the ROA estimate. A search was performed with 2 million simulations each for the baseline and revised control laws. The baseline control law provides an upper bound of $\bar{\beta}_{MC} = 1.56 \times 10^{-2}$ whereas the revised control law provides an upper bound of $\bar{\beta}_{MC} = 2.95 \times 10^{-2}$. The search also returns an initial condition x_0 on the boundary of the ellipsoid, i.e. $p(x_0) = x_0^T N x_0 = \bar{\beta}_{MC}$, that causes the system to go unstable. Hence, the value of the $\bar{\beta}_{MC}$ provides an upper bound of the ROA for the F/A-18 aircraft. This is complementary information to that provided by the Lyapunov-based lower bounds. The Monte Carlo search returned the following initial condition for the closed system with the baseline control law:

$$x_0 = [-1.1206^o,\ -12.3353^o/s,\ 1.5461^o/s,\ -5.8150^o,\ 28.9786^o,\ 9.9211^o/s,\ 0]^T$$

This initial condition satisfies $p(x_0) = 1.56 \times 10^{-2}$ and the closed-loop system with the baseline control law diverges from this initial condition. Decreasing the initial condition slightly leads to a stable response. For the revised control law the Monte Carlo search returned the following initial condition:

$$x_0 = [0.3276^o,\ -8.0852^o/s,\ 2.8876^o/s,\ -2.1386^o,\ 44.8282^o,\ 9.9829^o/s,\ 0]^T$$

This initial condition satisfies $p(x_0) = 2.95 \times 10^{-2}$ an the closed-loop system with the revised control law diverges from this initial condition. Decreasing the initial condition slightly leads to a stable response.

The lower and upper bounds on β^* can be visualized by plotting slices of the ellipsoidal approximation. Figures 2 and 3 show slices of the inner/outer approximations of the best ellipsoidal ROA approximation for both the baseline and revised control laws. The slices are in the α-β (Figure 2) and p-r (Figure 3) planes. The solid lines show the slices of the inner bounds obtained from quartic Lyapunov analysis. Every initial condition within the solid ellipses will return to the trim condition (marked as a '+'). The dashed lines show the slices of the outer bounds obtained from Monte Carlo analysis. There is at least one initial condition on the outer ellipsoid which leads to a divergent trajectory. The initial condition leading to a divergent trajectory does not

necessarily lie on the slice of the ellipsoid shown in the figure. The closeness of the inner and outer ellipsoids means that we have solved, for engineering purposes, the best ROA ellipsoid problem. Recall the aerodynamic coefficients were fitted over a gridded α - β space of $-20^\circ \leq \beta \leq 20^\circ$, and $-10^\circ \leq \alpha \leq 40^\circ$. Hence, the model is not valid over the entire region shown in this figure. The lower bounds for the revised controller are larger than the upper bounds for the baseline controller. Thus the ROA for the revised controller is provably larger than the ROA for the baseline controller.

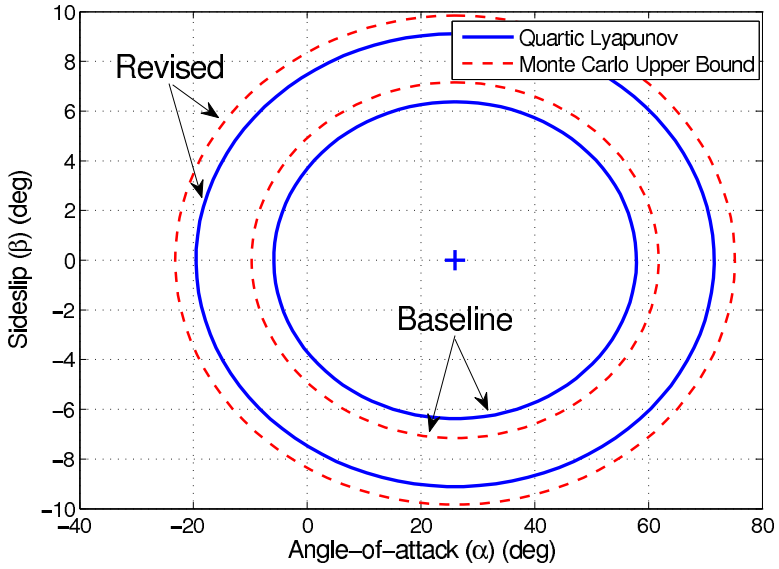

Fig. 2. ROA Estimates in α-β plane for Baseline and Revised Flight Control Law around Steady 60^o Bank Turn

4.2 Reachable Set Estimation for NASA's GTM

NASA's Generic Transport Model (GTM) is a remote-controlled 5.5 percent scale commercial aircraft [56, 57]. The main GTM aircraft parameters are provided in Table 2. NASA constructed a high fidelity 6 degree-of-freedom Simulink model of the GTM with the aerodynamic coefficients described as look-up tables. This section describes the construction of a polynomial model for the longitudinal dynamics of the GTM. This polynomial model is then used to estimate the reachable set for the open-loop longitudinal dynamics. Details on the polynomial modeling are provided in [58]. [59] provides additional motivation for using reachable sets for estimating the safe flight envelope for an aircraft.

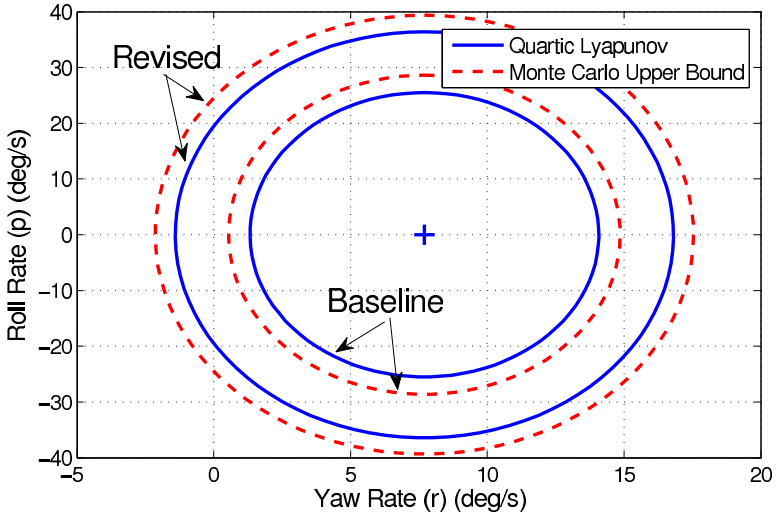

Fig. 3. ROA Estimates in p-r for Baseline and Revised Flight Control Law around Steady 60^o Bank Turn

Table 2. Aircraft and Environment Parameters

Wing Area, S	5.902 ft^2
Mean Aerodynamic Chord, $\bar{c}$	0.9153 ft
Mass, m	1.542 slugs
Pitch Axis Moment of Inertia, I_{yy}	4.254 slugs-ft^2
Air Density, ρ	0.002375 slugs/ft^3
Gravity Constant, g	32.17 ft/s^2

The longitudinal dynamics of the GTM are described by a standard four-state longitudinal model [60]:

$$\dot{V} = \frac{1}{m}\left(-D - mg\sin\left(\theta - \alpha\right) + T_x\cos\alpha + T_z\sin\alpha\right) \tag{26}$$

$$\dot{\alpha} = \frac{1}{mV}\left(-L + mg\cos\left(\theta - \alpha\right) - T_x\sin\alpha + T_z\cos\alpha\right) + q \tag{27}$$

$$\dot{q} = \frac{\left(M + T_m\right)}{I_{yy}} \tag{28}$$

$$\dot{\theta} = q \tag{29}$$

where V is the air speed (ft/s), α is the angle of attack (rad), q is the pitch rate (rad/s) and θ is the pitch angle (rad). The control inputs are the elevator deflection δ_{elev} (deg) and engine throttle δ_{th} (percent).

The drag force D (lbs), lift force L (lbs), and aerodynamic pitching moment M (lb-ft) are given by:

$$D = \bar{q}SC_D(\alpha, \delta_{elev}, \hat{q}) \tag{30}$$

$$L = \bar{q}SC_L(\alpha, \delta_{elev}, \hat{q}) \tag{31}$$

$$M = \bar{q}S\bar{c}C_m(\alpha, \delta_{elev}, \hat{q}) \tag{32}$$

where $\bar{q} := \frac{1}{2}\rho V^2$ is the dynamic pressure (lbs/ft^2) and $\hat{q} := \frac{\bar{c}}{2V}q$ is the normalized pitch rate (unitless). C_D, C_L, and C_m are unitless aerodynamic coefficients computed from look-up tables provided by NASA.

The GTM has one engine on the port side and one on the starboard side of the airframe. Equal thrust settings for both engines is assumed. The thrust from a single engine T (lbs) is a function of the throttle setting δ_{th} (percent). $T(\delta_{th})$ is specified as a ninth-order polynomial in NASA's high fidelity GTM simulation model. T_x (lbs) and T_z (lbs) denote the projection of the total engine thrust along the body x-axis and body-z axis, respectively. T_m (lbs-ft) denotes the pitching moment due to both engines. T_x, T_z and T_m are given by:

$$T_x(\delta_{th}) = n_{ENG}T(\delta_{th})\cos(\epsilon_2)\cos(\epsilon_3) \tag{33}$$

$$T_z(\delta_{th}) = n_{ENG}T(\delta_{th})\sin(\epsilon_2)\cos(\epsilon_3) \tag{34}$$

$$T_m(\delta_{th}) = r_z T_x(\delta_{th}) - r_x T_z(\delta_{th}) \tag{35}$$

$n_{ENG} = 2$ is the number of engines. $\epsilon_2 = 0.0375$ rad and $\epsilon_3 = -0.0294$ rad are angles that specify the rotation from engine axes to the airplane body axes. $r_x = 0.4498$ ft and $r_z = 0.2976$ ft specify the moment arm of the thrust.

The following terms of the longitudinal are approximated by low-order polynomials:

1. Trigonometric functions: $\sin(\alpha)$, $\cos(\alpha)$, $\sin(\theta - \alpha)$, $\cos(\theta - \alpha)$
2. Engine model: $T(\delta_{th})$
3. Rational dependence on speed: $\dfrac{1}{V}$
4. Aerodynamic coefficients: C_D, C_L, C_m

Constructing polynomial approximations for the trigonometric functions, engine model, and rational dependence on speed is relatively straight-forward. The trigonometric functions are approximated by Taylor series expansions: $\sin z \approx z - \frac{1}{6}z^3$ and $\cos z \approx 1 - \frac{1}{2}z^2$ for z in units of radians. For the engine model, a least squares technique is used to approximate the ninth order polynomial function $T(\delta_{th})$ by a third order polynomial. The least squares technique is also used to compute a linear fit to $\dfrac{1}{V}$ over the desired range of interest from 100 ft/s to 200 ft/s. Finally, polynomial least squares fits are computed for the aerodynamic coefficient look-up table provided by NASA. A degree seven polynomial model is obtained after replacing all non-polynomial terms with their polynomial approximations. The polynomial model takes the form:

$$\dot{x} = f(x, u) \tag{36}$$

where $x := [V(\text{ft/s}),\ \alpha(\text{rad}),\ q(\text{rad/s}),\ \theta(\text{rad})]$, and $u := [\delta_{elev}(\text{deg}),\ \delta_{th}(\%)]$. The degree seven polynomial model $f(x,u)$ is provided in [58]. The quality of the polynomial approximation was assessed by comparing the trim conditions and simulation responses of the polynomial model and the original model with look-up tables.

The remainder of the section describes the estimation of the reachable set for the open-loop longitudinal dynamics of the GTM. The trim condition for the analysis is:

$$x_{trim} := [150 \text{ ft/s}, 0.047 \text{ rad}, 0 \text{ rad/s}, 0.047 \text{ rad}] \tag{37}$$

The throttle input is held at its trim value and the reachable set is computed for elevator inputs around the trim elevator input. The shape function is $p(x) := x^T N x$ where:

$$N := diag(50 \text{ ft/s}, 0.35 \text{ rad}, 0.87 \text{ rad/s}, 0.35 \text{ rad})^{-2} \tag{38}$$

Upper bounds $\bar{\beta}$ were computed such that $\mathcal{G}\gamma \subseteq \mathcal{E}_{\bar{\beta}}$. These bounds were computed for many values of γ using the method described in Section 3.3. A nonlinear optimal control problem can be approximately solved to compute "worst-case" inputs for this reachable set problem [40]. The inputs are worst-case in the sense of maximizing $p(x_f)$ subject to the constraint $\|u\|_2^2 \leq \gamma^2$. This worst-case algorithm provides lower bounds $\underline{\beta}$ for the reachable set problem. These lower bounds provide complementary information to the upper bounds computed using SOS methods. Specifically the lower bounds prove that there is an $x_f \in \mathcal{G}\gamma$ such that $p(x_f) = \underline{\beta}$

Figure 4 shows the lower and upper bounds computed for the GTM. The upper bounds computed using quadratic and quartic (degree 4) storage functions are shown in squares and diamonds, respectively. Each point on the upper bound curve with quadratic storage functions took several minutes to compute. Only three points were computed for quartic storage functions since each data point on this curve took several hours to compute. Degree six storage functions would lead to improved bounds but with a significant increase in computation time. The lower bound computed using the method in [40] is shown with circles. This curve is quite far from the quartic upper bound. The reachable set for the linearized plant is drawn as a black dashed curve. This is a straight line because scaling the norm of the input scales the distance that can be reached. The optimal input for the linear plant can be computed via an optimal control problem. This worst-case input computed from the linear plant is shown in Figure 5. Simulating the nonlinear system with this worst-case input (scaled to achieve $\|u\|_2^2 = \gamma$) achieved the lower bound shown in squares in Figure 4. This lower bound is very close to the quartic upper bound. Thus the worst-case input computed from the linear plant is also a bad input for the nonlinear plant. It appears that the GTM dynamics are well-approximated by the linearization at this flight condition. This statement was further investigated by simulating the full 6DOF GTM

model with the worst-case input computed from the linear plant (scaled to have $\|u\|_2^2 = 1$). The 6DOF GTM model includes many additional modeling details including lateral dynamics, actuator dynamics, etc. Surprisingly the response of the full 6DOF GTM model with this input was very similar to the response of the polynomial model with this input. A comparison of the simulation responses is shown in Figure 6. The states (α,θ,q) have units of (rad,rad,rad/s) but are plotted in units of (deg,deg,deg/s) for ease of interpretation. For this input the polynomial model achieved $\underline{\beta} = 0.4803$ and the full GTM achieved $\underline{\beta} = 0.3478$. This nonlinear reachable analysis provides confidence that the linear model can be used as a good approximation at this flight condition.

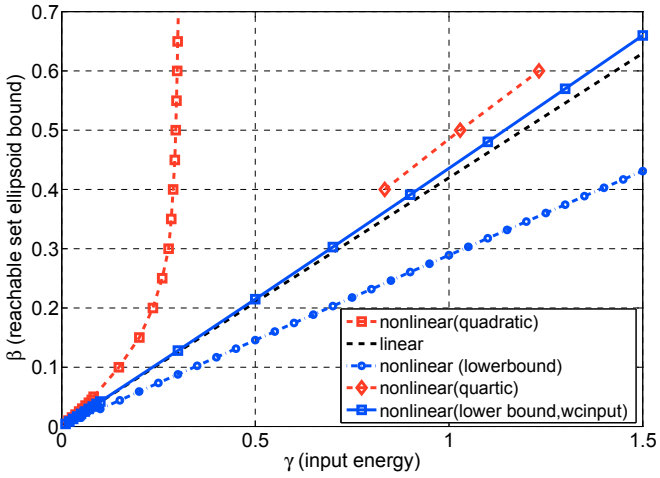

Fig. 4. Reachable Set Bounds

5 Conclusion

This chapter described the use of sum-of-squares optimizations for analyzing nonlinear polynomial systems. In particular, optimizations with SOS constraints were formulated for computing region of attraction estimates, bounds on L_2 to L_2 gain, and reachable sets. Many other nonlinear analysis problems can be formulated within this optimization framework. The approach was applied to compare the performance of two F/A-18 control laws in suppressing a loss-of-control motion known as the falling leaf mode. The reachable set for NASA's Generic Transport Model was also estimated. These nonlinear analysis tools can fill the gap between linear analyses, which are valid only for infinitesimally small neighborhoods about an equilibrium, and nonlinear

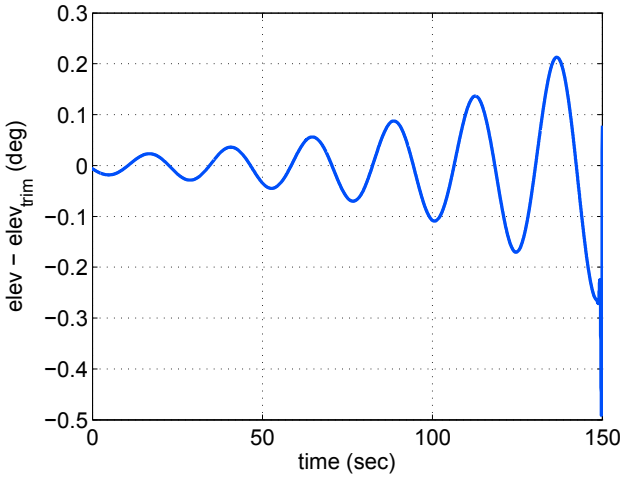

Fig. 5. Worst-case input computed from the linear plant

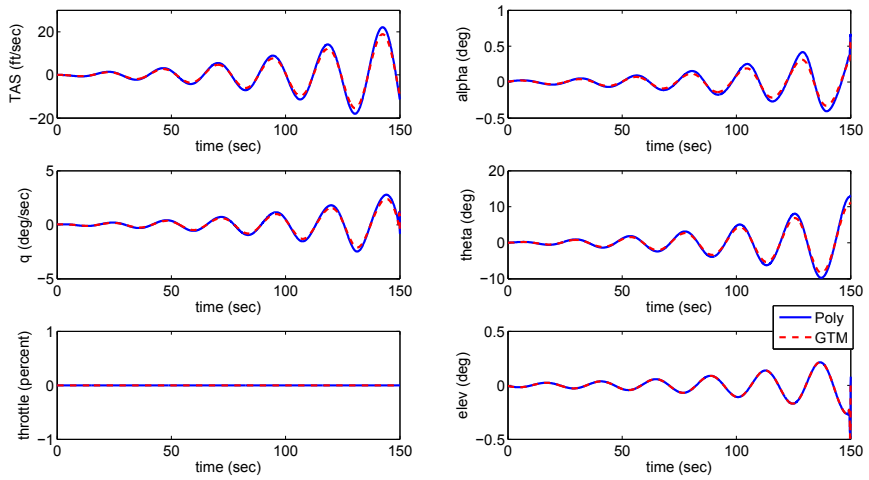

Fig. 6. Simulation of Polynomial and 6DOF GTM Models with Worst-case input

simulations. These tools can be used to provide additional confidence when validating the performance of a flight control law. Significant work remains to be done to reduce the computational cost and enable these techniques to be applied to moderate-sized systems (systems with more than ≈ 8 states).

Acknowledgments

This research was partially supported under the NASA Langley NRA contract NNH077ZEA001N entitled "Analytical Validation Tools for Safety Critical Systems" and the NASA Langley NNX08AC65A contract entitled 'Fault Diagnosis, Prognosis and Reliable Flight Envelope Assessment." The technical contract monitors are Dr. Christine Belcastro and Dr. Suresh Joshi respectively.

References

1. P. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Institute of Technology, 2000.
2. ——, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical Programming Ser. B*, vol. 96, no. 2, pp. 293–320, 2003.
3. B. Tibken, "Estimation of the domain of attraction for polynomial systems via LMIs," in *Proceedings of the IEEE Conference on Decision and Control*, 2000, pp. 3860–3864.
4. O. Hachicho and B. Tibken, "Estimating domains of attraction of a class of nonlinear dynamical systems with LMI methods based on the theory of moments," in *Proceedings of the IEEE Conference on Decision and Control*, 2002, pp. 3150–3155.
5. Z. Jarvis-Wloszek, "Lyapunov based analysis and controller synthesis for polynomial systems using sum-of-squares optimization," Ph.D. dissertation, University of California, Berkeley, 2003.
6. Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, "Some controls applications of sum of squares programming," in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 5, 2003, pp. 4676–4681.
7. W. Tan and A. Packard, "Searching for control Lyapunov functions using sums of squares programming," in *42nd Annual Allerton Conference on Communications, Control and Computing*, 2004, pp. 210–219.
8. G. Chesi, "On the estimation of the domain of attraction for uncertain polynomial systems via LMIs," in *Proceedings of the IEEE Conference on Decision and Control*, 2004, pp. 881–886.
9. A. Papachristodoulou, "Scalable analysis of nonlinear systems using convex optimization," Ph.D. dissertation, California Institute of Technology, 2005.
10. Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, *Positive Polynomials in Control*, ser. Lecture Notes in Control and Information Sciences. Springer-Verlag, 2005, vol. 312, ch. Controls Applications of Sum of Squares Programming, pp. 3–22.
11. G. Chesi, A. Garulli, A. Tesi, and A. Vicino, "Lmi-based computation of optimal quadratic lyapunov functions for odd polynomial systems," *International Journal of Robust and Nonlinear Control*, vol. 15, pp. 35–49, 2005.
12. S. Prajna, "Optimization-based methods for nonlinear and hybrid systems verification," Ph.D. dissertation, California Institute of Technology, 2005.

13. W. Tan, A. Packard, and T. Wheeler, "Local gain analysis of nonlinear systems," in *Proceedings of the American Control Conference*, 2006, pp. 92–96.
14. W. Tan, "Nonlinear control analysis and synthesis using sum-of-squares programming," Ph.D. dissertation, University of California, Berkeley, 2006.
15. B. Tibken and Y. Fan, "Computing the domain of attraction for polynomial systems via BMI optimization methods," in *Proceedings of the American Control Conference*, 2006, pp. 117–122.
16. U. Topcu, A. Packard, P. Seiler, and T. Wheeler, "Stability region analysis using simulations and sum-of-squares programming," in *Proceedings of the American Control Conference*, 2007, pp. 6009–6014.
17. W. Tan, U. Topcu, P. Seiler, G. Balas, and A. Packard, "Simulation-aided reachability and local gain analysis for nonlinear dynamical systems," in *Proceedings of the IEEE Conference on Decision and Control*, 2008, pp. 4097–4102.
18. U. Topcu, A. Packard, and P. Seiler, "Local stability analysis using simulations and sum-of-squares programming," *Automatica*, vol. 44, no. 10, pp. 2669–2675, 2008.
19. U. Topcu, "Quantitative local analysis of nonlinear systems," Ph.D. dissertation, University of California, Berkeley, 2008.
20. U. Topcu and A. Packard, "Linearized analysis versus optimization-based nonlinear analysis for nonlinear systems," in *submitted to the 2009 American Control Conference*, 2009.
21. S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, ser. Studies in Applied Mathematics. SIAM, 1994, vol. 15.
22. B. Reznick, "Some concrete aspects of Hilberts 17th problem," *Contemporary Mathematics*, vol. 253, no. 251-272, 2000.
23. L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, no. 1, pp. 49–95, 1996.
24. J. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimization Methods and Software*, pp. 625–653, 1999.
25. ——, "SeDuMi version 1.05," http://fewcal.kub.nl/sturm/software/sedumi.html, 2001.
26. B. Reznick, "Extremal PSD forms with few terms," *Duke Mathematical Journal*, vol. 45, no. 2, pp. 363–374, 1978.
27. M. Choi, T. Lam, and B. Reznick, "Sums of squares of real polynomials," *Proceedings of Symposia in Pure Mathematics*, vol. 58, no. 2, pp. 103–126, 1995.
28. V. Powers and T. Wörmann, "An algorithm for sums of squares of real polynomials," *Journal of Pure and Applied Algebra*, vol. 127, pp. 99–104, 1998.
29. S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo, *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, Available from `http://www.cds.caltech.edu/sostools` and `http://www.mit.edu/~parrilo/sostools`, 2004.
30. J. Lofberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. [Online]. Available: http://control.ee.ethz.ch/ joloef/yalmip.php
31. G. Balas, A. Packard, P. Seiler, and U. Topcu, "Robustness analysis of nonlinear systems," 2009, http://www.aem.umn.edu/ AerospaceControl/.
32. K. Gatermann and P. Parrilo, "Symmetry groups, semidefinite programs, and sums of squares," *Journal of Pure and Applied Algebra*, vol. 192, pp. 95–128, 2004.

33. A. Vannelli and M. Vidyasagar, "Maximal Lyapunov functions and domains of attraction for autonomous nonlinear systems," *Automatica*, vol. 21, no. 1, pp. 69–80, 1985.
34. J. Hauser and M. Lai, "Estimating quadratic stability domains by nonsmooth optimization," in *Proceedings of the American Control Conference*, 1992, pp. 571–576.
35. R. Genesio, M. Tartaglia, and A. Vicino, "On the estimation of asymptotic stability regions: State of the art and new proposals," *IEEE Transactions on Automatic Control*, vol. 30, no. 8, pp. 747–755, 1985.
36. E. Davison and E. Kurak, "A computational method for determining quadratic Lyapunov functions for nonlinear systems," *Automatica*, vol. 7, pp. 627–636, 1971.
37. H.-D. Chiang and J. Thorp, "Stability regions of nonlinear dynamical systems: A constructive methodology," *IEEE Transactions on Automatic Control*, vol. 34, no. 12, pp. 1229–1241, 1989.
38. M. Vidyasagar, *Nonlinear Systems Analysis*, 2nd ed. Prentice Hall, 1993.
39. H. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.
40. J. Tierno, R. Murray, J. Doyle, and I. Gregory, "Numerically efficient robustness analysis of trajectory tracking for nonlinear systems," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 20, pp. 640–647, 1997.
41. M. Heller, R. David, and J. Holmberg, "Falling leaf motion suppression in the F/A-18 Hornet with revised flight control software," in *AIAA Aerospace Sciences Meeting*, no. AIAA-2004-542, 2004.
42. A. Chakraborty, P. Seiler, and G. Balas, "Applications of linear and nonlinear robustness analysis techniques to the f/a- 18 flight control laws," in *AIAA Guidance, Navigation, and Control Conference*, no. AIAA-2009-5670, 2009.
43. P. T. Jaramillo and J. N. Ralston, "Simulation of the F/A-18D falling leaf," in *AIAA Atmospheric Flight Mechanics Conference*, 1996, pp. 756–766.
44. S. B. Buttrill, P. D. Arbuckle, and K. D. Hoffler, "Simulation model of a twintail, high performance airplane," NASA, Tech. Rep. NASA TM-107601, 1992.
45. R. Stengel, *Flight Dynamics*. Princeton University Press, 2004.
46. M. Cook, *Flight Dynamics Principles*. Wiley, 1997.
47. M. R. Napolitano and J. M. Spagnuolo, "Determination of the stability and control derivatives of the NASA F/A-18 HARV using flight data," NASA, Tech. Rep. NASA CR-194838, 1993.
48. M. R. Napolitano, A. C. Paris, and B. A. Seanor, "Estimation of the lateral-directional aerodynamic parameters from flight data for the NASA F/A-18 HARV," in *AIAA Atmospheric Flight Mechanics Conference*, no. AIAA-96-3420-CP, 1996, pp. 479–489.
49. C. D. Lluch, "Analysis of the out-of-control falling leaf motion using a rotational axis coordinate system," Master's thesis, Virginia Polytechnic Institue and State Unniversity, 1998.
50. M. R. Napolitano, A. C. Paris, and B. A. Seanor, "Estimation of the longitudinal aerodynamic parameters from flight data for the NASA F/A-18 HARV," in *AIAA Atmospheric Flight Mechanics Conference*, no. AIAA-96-3419-CP, pp. 469–478.
51. K. W. Iliff and K.-S. C. Wang, "Extraction of lateral-directional stability and control derivatives for the basic F-18 aircraft at high angles of attack," *NASA TM-4786*, 1997.

52. K. W. Illif and K.-S. C. Wang, "Retrospective and recent examples of aircraft parameter identification at NASA dryden flight research center," *Journal of Aircraft*, vol. 41, no. 4, 2004.
53. A. Schy and M. E. Hannah, "Prediction of jump phenomena in roll-coupled maneuvers of airplanes," *Journal of Aircraft*, vol. 14, no. 4, pp. 375– 382, 1977.
54. M. Heller, R. Niewoehner, and P. K. Lawson, "High angle of attack control law development and testing for the F/A-18E/F Super Hornet," in *AIAA Guidance, Navigation, and Control Conference*, no. AIAA-1999-4051, 1999, pp. 541–551.
55. ——, "On the validation of safety critical aircraft systems, part i: An overview of analytical & simulation methods," in *AIAA Guidance, Navigation, and Control Conference*, no. AIAA 2003-5559, 2003.
56. D. Cox, *The GTM DesignSim v0905*, 2009.
57. A. Murch and J. Foster, "Recent NASA research on aerodynamic modeling of post-stall and spin dynamics of large transport airplanes," in *45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada*, 2007.
58. A. Chakraborty, P. Seiler, and G. Balas, "Nonlinear region of attraction analysis for flight control verification and validation," *Control Engineering Practice*, 2010, submitted for publication.
59. R. Pandita, P. Seiler, and G. Balas, "Reachability and region of attraction analysis applied to gtm dynamic flight envelope assessment," in *AIAA Guidance, Navigation, and Control Conference*, no. AIAA-2009-6258, 2009.
60. B. Stevens and F. Lewis, *Aircraft Control and Simulaion*. John Wiley & Sons, 1992.