A NON-CONSERVATIVE APPROACH FOR THE ESTIMATION OF THE REGION OF OPERATION OF UNCERTAIN ADAPTIVE CONTROL SYSTEMS

Mario Luca Fravolini  
University of Perugia  
Perugia, Italy

Tansel Yucelen  
Missouri University of Science and Technology  
Rolla, MO, USA

Antonio Moschitta  
University of Perugia  
Perugia, Italy

Benjamin Gruenwald  
Missouri University of Science and Technology  
Rolla, MO, USA

ABSTRACT

A challenging problem for Model Reference Adaptive Control Systems is the accurate characterization of the transient response in the presence of large uncertainties. Early prior research by the authors has demonstrated that using a projection mechanism for parameters adaptation the tracking error dynamics behaves as a linear system perturbed by bounded uncertainties. This brings the benefit that the stability analysis can be cast in terms of a convex optimization problem with LMI constraints so that efficient numerical tools can be used for the adaptive controller design. A possible limitation of the approach is that the design is restricted to quadratic control Lyapunov functions that could produce a conservative estimation of the regions of operation for the actual uncertain adaptive system. In this paper this approach is extended to arbitrary high degree polynomial Lyapunov functions by translating the design and performance requirements in terms of Sum of Square (SOS) inequalities and then using SOS optimization tools for the design. In this effort the new SOS approach is introduced and compared with the previous one. A numerical example based on the short period longitudinal dynamics of the F16 aircraft is used to demonstrate the efficacy of the novel method.

INTRODUCTION

It is a well-known fact that adaptive control systems are particularly effective in compensating for large modeling uncertainties, faults and time varying disturbances in linear and nonlinear plants [1]. Despite these desirable features adaptive controllers are not widely employed in practice especially for safety critical systems [2]. One of the main reasons is due to the lack of systematic analysis and design tools for predicting, a-priori, the evolution of the error and of the adaptation gains during the transients in conjunction to the problem of relating controller parameters to time domain specifications [3]. The difficulty in predicting the transient response originates from the inherent nonlinearity of the parameter’s adaptation law. For instance fast error tracking in Model Reference Adaptive Control (MRAC) can be usually achieved by increasing the adaptation rate, but it is well known that this could induce chattering and peaking in the control signal, actuator saturation and the excitation of unmodelled dynamics. Clearly, these problems are of primary relevance for the verification and validation standpoint of critical applications such as aerospace systems [4].

In the last 30 years a large number of adaptation laws and robust modifications have been proposed to guarantee the asymptotic convergence of the tracking error or the boundedness within specified domains. For instance adaptive $H_2$, $H_\infty$ and $L_1$ approaches have been proposed in [5,6,7,8] with the purpose of minimizing specific performance metrics with the effect of improving, in general terms, the response of the system but without an immediate relation with the transitory behavior of the tracking error. Recently, for example the authors in [9] and [10] proposed methodologies that address, specifically, the convergence of the tracking error within specified domains. For these schemes the design and verification is carried out through the Lyapunov direct method that is typically restricted to the employment of quadratic Lyapunov functions. In these frameworks the system operative regions are quantitatively overestimated as a function of the norm of the error vector but available bounds turn out to be
SOS optimization is a powerful numeric method used to solve generic optimization problems characterized by a linear cost function with constraints having a polynomial structure. An important feature of SOS constraints is that they can be transformed into equivalent linear matrix inequality constraints, therefore, SOS programs can be converted to semidefinite programs and solved efficiently with convex optimization tools. SOS optimization is thus particularly suited for the automated design of high degree polynomial Lyapunov functions for nonlinear systems characterized by a polynomial vector field. In this case the requirement of a positive-definite Lyapunov function ($V > 0$) whose time derivative is negative semidefinite ($\dot{V} < 0$) can be immediately cast as SOS constraints.

Assuming, as in previous research [11,12,13], a polynomial vector field for the uncertain system and further assuming a polynomial vector field also for the weight adaptation law (to be designed), one can conclude that the overall vector field of the MRAC system is polynomial and therefore can be analyzed exploiting SOS tools. This allows the adoption of high order polynomial Lyapunov functions that in turn allow the automated design of invariant regions that are not limited to be ellipsoidal (quadratic forms), but can have, theoretically, an arbitrary degree. Some related works dealing with the application of SOS optimization for the analysis and the design of adaptive control systems can be found in [17,18].

In the present paper we introduce the novel SOS formulation for MRAC and compare it with the previously introduced Linear Matrix Inequality (LMI) approach in [11,12,13].

A detailed comparative study between the two techniques, focused on the estimation of minimal size invariant regions for the tracking error is illustrated and applied to the short period longitudinal dynamics of an F16 aircraft model under MRAC control.

**MATHEMATICAL FORMULATION OF THE ADAPTIVE CONTROL SCHEME**

Consider the nonlinear uncertain dynamical system [11]

$$\dot{x}_p(t) = A_p x(t) + B_p \Lambda \left( \Delta(x_p(t)) + u(t) + \varepsilon_p(t) \right) \quad (1)$$

where $x_p(t) \in \mathbb{R}^{n_p}$ is the state (accessible) vector, $u(t) \in \mathbb{R}$ is the control input, $A_p \in \mathbb{R}^{n_p \times n_p}$ and $B_p \in \mathbb{R}^{n_p}$ are nominal system matrices, $\Delta(x_p) \in \mathbb{R}^{n_p} \rightarrow \mathbb{R}$ is a matched system uncertainty; $\Lambda$ is a positive unknown diagonal control effectiveness gain such that $\Lambda_{\text{min}} \leq \Lambda \leq \Lambda_{\text{max}}$ and $\varepsilon_p(t)$ is a contribution that takes into account the uncertainty that cannot be captured by an $\Delta(x_p)$. We assume that $\varepsilon_p(t)$ is unknown but bounded such that

$$|\varepsilon_p(t)| \leq \varepsilon_M \quad (2)$$

where $\varepsilon_M$ is a known upper bound; define also the set $\Pi_\varepsilon = \{ (\varepsilon_p(t) \in \mathbb{R} / |\varepsilon_p(t)| \leq \varepsilon_M \}$. Furthermore we assume that the couple $(A_p, B_p)$ is controllable and that uncertainty in (1) is linearly parameterized as

$$\Delta(x_p) = W_p^T x_p \quad (3)$$

where $W_p \in \mathbb{R}^{n_p}$ is an unknown constant weight vector. To address command following let $c(t) \in \mathbb{R}^{n_c}$ ($n_c = 1$ in this study) be a given smooth command signal and $x_c(t) \in \mathbb{R}^{n_c}$ the integral of the tracking error satisfying

$$\dot{x}_c(t) = E_p x_p(t) - c(t) \quad (4)$$

where $E_p$ is a binary matrix used to select the subset of the $x_p(t)$ states to be followed by $c(t)$. Now, (1) can be augmented with (4) obtaining

$$\dot{x}(t) = Ax(t) + B \Lambda (W_p^T x_p(t) + u(t) + \varepsilon_p(t)) + B_c c(t), \quad (5)$$

where $x = [x_p^T, \dot{x}_c^T]^T \in \mathbb{R}^n$ ($n = n_p + n_c$) is the augmented state vector and

$$A = \begin{bmatrix} A_p & 0 \\ E_p & 0 \end{bmatrix} \in \mathbb{R}^{n \times n}, \quad B = \begin{bmatrix} B_p^T, 0 \end{bmatrix}^T \in \mathbb{R}^{n \times m}, \quad B_m = [0, -1]^T \in \mathbb{R}^{n \times n_c} \quad (6)$$

Next, consider the feedback control law

$$u(t) = u_n(t) + u_{al}(t) \quad (7)$$

where $u_n(t)$ and $u_{al}(t)$ are the nominal and adaptive controller parts, respectively.
where the nominal control law \( u_n(t) \) is given by

\[
u_n(t) = -K_0 x(t) \quad (8)
\]

where \( K_0 \in \mathbb{R}^n \) is chosen such that \( A_m = A - BK_0 \) is Hurwitz. The reference model is

\[
x_m(t) = A_m x_m(t) + B_m e(t) \quad (9)
\]

Using (8) and (3) in (1), yields for the system

\[
\dot{x}(t) = A_n x(t) + B \left( W^T x(t) + u_u(t) + e(t) \right) + B_m e(t) \quad (10)
\]

where \( W^T = A \left( W^T B + (A^{-1} - 1)K_0 \right) \) is an unknown, aggregated matrix where \( M \) is a binary state selection matrix such that \( x_p = M x \) and \( e(t) = A \xi e_p(t) \). Motivated by the structure of the uncertain term in (10), we adopted the following adaptive control law

\[
u_u(t) = -\dot{W}^T(t) x(t) \quad (11)
\]

where \( \dot{W}(t) \in \mathbb{R}^{n} \) is the estimate of \( W \) satisfying a standard projection based update law

\[
\dot{W}(t) = \Gamma \cdot \text{Proj}[\dot{W}(t), x(t) e^T(t) P \cdot B] \quad (12)
\]

where matrix \( P \) is positive definite, the scalar \( \Gamma > 0 \) is the learning rate and \( e(t) = x(t) - x_m(t) \) is the tracking error between the system and the reference model (9). Substituting the adaptive law (11) in (10), the resulting error dynamics is

\[
\dot{e}(t) = A_n e(t) - B \left( \dot{W}^T(t) x(t) + e(t) \right) \quad (13)
\]

where \( \dot{W}(t) = \dot{W}(t) - W \) is the weight estimation error. By constructions the projection operator guarantees the confinement of the weights \( \dot{W}(t) \) within a domain \( \Pi_w \) such that \( ||\dot{W}(t)||_F^2 \leq W_M^2 \), where \( W_M \) is a user defined radius for the hyper sphere that defines \( \Pi_w \). The parameters adaptation domain is defined as \( \Pi_w = \{ \dot{W}(t) \in \mathbb{R}^n| \quad \dot{W}(t)^T \dot{W}(t) \leq W_M^2 \} \). The value of \( W_M \) is selected large enough so that the unknown vector \( W \) in internal to \( \Pi_w \). This implies that the weight estimation error \( \dot{W}(t) \) is bounded such that

\[
||\dot{W}(t)||_F^2 \leq 4W_M^2 = W_{M_0}^2 \quad (14)
\]

Define also the set \( \Pi_{w_0} = \{ \dot{W}(t) \in \mathbb{R}^n| \quad \dot{W}(t)^T \dot{W}(t) \leq W_{M_0}^2 \} \).

### Stability Analysis via a Quadratic Lyapunov Function

In the literature, the stability and boundedness properties of the error dynamics (13) are usually investigated using a quadratic Lyapunov function in the following form

\[
V(e, \dot{W}) = e^T P e + \Gamma^{-1} \dot{W}^T \dot{W} \quad (15)
\]

where \( P \) is positive definite matrix and \( \Gamma \) is a positive scalar. The time derivative of (15) is

\[
\dot{V}(e, \dot{W}) = e^T P e + \Gamma^{-1} \dot{W}^T \dot{W} \quad (16)
\]

Based on Lyapunov stability arguments, using (12) and exploiting the property of the projection operator \( \{ \text{Proj}[\dot{W}(t), x(t) e^T(t) P B] - x(t)e^T(t) P B \} \leq 0 \), it is not difficult to show that that:

\[
\dot{V}(e, \dot{W}) \leq e^T \left( \Lambda_m^T P + P \Lambda_m \right) + 2e^T P B e + 2\dot{W}^T \Gamma \dot{W} \quad (17)
\]

Inequality (17) is the basis for the stability and boundedness analysis that will be carried out in the next section.

### LMI Formulation of the Performance Requirements

Transient bounds, based on inequalities similar to (17), are proposed by many authors in literature. Unfortunately, these bounds turn out to be overly conservative resulting in scarce utility for practical design purposes. Further, the bounding region, being a function of the norm of the state vector, does not allow a selective analysis of the response along the single error components. To face this problem the authors in [11,12,13] have introduces a LMI based analysis and design framework for adaptive control systems that allows the computation of componentwise bounds as the results of an optimized design with LMI constraints. We now provide a brief characterization of the approach for the problem under investigation.

#### LMI Formulation

In order to carry out the boundedness analysis we define the ellipsoidal set that originates from the Lyapunov function level set \( V(e, \dot{W}) = 1 \), that is

\[
\Omega = \{ e, \dot{W} \in \mathbb{R}^{n_x} | \quad e^T P e + \dot{W}^T \Gamma^{-1} \dot{W} \leq 1 \} \quad (18)
\]

It is now introduced a robust positive invariance definition for a compact set (in this study \( \Omega \)).

**Definition-1:** A set \( \Omega \) is said to be robustly positively invariant for the error dynamics (13) if, for every initial state \( e(0) \in \Omega \) and for any allowed uncertainty the trajectories of the system remains in \( \Omega \) for any \( t \geq 0 \).

At this point we are ready to introduce a set of LMIs requirements that provide sufficient conditions for the robust positive invariance of the sets \( \Omega \) and for the fulfillment of tracking error performance.

#### Requirements on the Lyapunov Function

A necessary condition for the positive invariance of \( \Omega \) is that the Lyapunov function must be positive definite, that is: \( V(e, \dot{W}) > 0 \) for all \( (e, \dot{W}) \neq (0,0) \). Considering (15) this is equivalent to the requirements:

\[
P > 0, \Gamma > 0 \quad (19)
\]
Requirements of Robust Positive Invariance for Ω

A sufficient condition for Ω to be robustly positively invariant for the closed loop adaptive system is given by the following theorem.

**Theorem 1:** Consider the closed loop error dynamics (13), the adaptation law (12) in the presence of bounded disturbance \( W \in \Pi_W \) and \( e \in \Pi_e \). If there exist a positive scalar \( \alpha \) and a positive definite symmetric matrix \( P \) satisfying the constraint:

\[
\begin{bmatrix}
A^T e + A_m P + \alpha P + P B e_c \\
(PE_c^T)^T - \alpha + \alpha^{-1} W^2_{M_0}
\end{bmatrix} < 0
\]

(20)

then, \( \Omega \) is robustly positively invariant.

**Proof:** It is sufficient to show that the time derivative of the Lyapunov function (16) is \( \leq 0 \) along the boundary \( \partial \Omega \) of \( \Omega \) for any admissible \( \hat{W} \) and \( e(t) \). In fact, since by definition \( V(t) \) is constant along \( \partial \Omega \), if the derivative \( \dot{V}(t) \) is \( \leq 0 \) along \( \partial \Omega \), for any admissible uncertainty, (assuming also that the initial state is internal to \( \Omega \) then, the system trajectories cannot escape \( \Omega \) for any \( t \geq 0 \), thus proving that \( \Omega \) is a robust positive invariant set. It is not difficult to show that the robust invariance requirement for \( \Omega \) is satisfied if the following condition

\[
\dot{e}^T (A_{m}^T P + P A_{m}) + 2 e^T P B e_c \leq 0
\]

(21)

holds for every admissible uncertainty such that:

\[
\begin{align*}
\dot{e}^T P e + &\Gamma^{-1} \hat{W}^T \hat{W} \geq 1 \\
\hat{W}^T \hat{W} &\leq W_{M_0}^2 \\
e^T e &\leq e_M^2
\end{align*}
\]

(22)

(23)

(24)

Specifically, (21) states that \( V(t) \leq 0 \), which derives immediately from (17). Condition (22) defines the region external to the boundary \( \partial \Omega \) and descends from (18); conditions (23) define the admissible values for the adaptive weights and descend from (14). Finally, (24) expresses the componentwise bounds on the unmodelled uncertainty and descends from (2).

Condition (21) with the constraints (22-24) can be transformed into condition (20) using the S-procedure [13]. In fact, applying the S-procedure, results that the above conditions are satisfied if there exist positive coefficients \( \alpha \) and \( \beta \) such that

\[
\begin{align*}
\dot{e}^T (A_{m}^T P + PA_{m}) e + 2 e^T P B e_c + &\alpha (e^T P e + \hat{W}^T \Gamma^{-1} \hat{W} - 1) \\
+ &\beta (4 e_M^2 W^2_{M} - \hat{W}^T \hat{W}) \leq 0
\end{align*}
\]

(25)

Expressing (25) in matrix form, and selecting \( \beta = \alpha^{-1} \), it easy to prove that condition (25) is equivalent to condition (20).

**Requirements on the tracking error**

Componentwise tracking error requirements are specified as follows:

\[
|e_i(t)| \leq \rho_i e_M \quad t \geq 0 \quad i = 1,...,n
\]

(26)

where the \( e_M \) are desired reference bounds and the \( \rho_i \) are scalar (scaling) optimization parameters that will be determined through the optimized design. Define also the (scaled) error performance polyhedron \( \Pi_{ep} = \{ e \mid |e_i(t)| \leq \rho_i e_M, i = 1,...,n \} \).

In the optimized design, that will be introduced shortly, the objective will be to reduce, as much as possible, the size of the set \( \Pi_{ep} \), by minimizing the values of coefficients \( \rho_i \). To analyze the trajectories of the error \( e(t) \) it is defined the subset \( \Omega_e = \{ e(t) \mid e^T P e(t) \leq 1 \} \) that is the projection of the sets \( \Omega \) in the error variables subspace. To satisfy constraints (26), we require the containment of the ellipsoid \( \Omega_e \) within the polyhedron \( \Pi_{ep} \). Exploiting geometric considerations it can be shown that condition \( \Omega_e \subseteq \Pi_{ep} \) holds if and only if the following LMIs are satisfied

\[
\begin{bmatrix}
P & g_i \\
g_i^T & \rho_i^2 e_M^2
\end{bmatrix} > 0 \quad i = 1,...,n
\]

(27)

where \( g_i = [0,...,1,...,0]^T \) is the \( i \)th base vector of \( R^n \).

**LMI BASED DESIGN AND PERFORMANCE VERIFICATION FRAMEWORK**

In this section it is briefly introduced a procedure that provides sufficient conditions to verify the invariance of the set \( \Omega \) and to verify the transitory performance requirements for the error variables. The method is based on the following results previously proven in [13].

**Theorem 2:** Consider the adaptive system (12) and assume nominal reduction coefficients \( \rho_i = 1 \) in (26). If there exist a positive matrices, \( P > 0 \) and an adaptation rate \( \Gamma > 0 \) such that the LMIs (19), (20), and (27) are satisfied, then the set \( \Omega \) is positively invariant and the system satisfies the nominal componentwise error bounds (26).

**A Convex optimization approach for the design and verification of the performance requirements**

Considering Theorem 2, in case feasible solutions exist, then it is of interest to verify if, it is possible to improve the nominal tracking error requirements given in (26). In [13] it was shown that this problem can be formulated as a convex optimization relying on a linear cost function (to be defined) whose optimization variables are the free parameters and matrices that are involved in the LMIs (19), (20), and (27). In this effort it was decided to choose as optimization objective the minimization of the “size” of the scaled error performance polyhedron \( \Pi_{ep} \). This objective is quantified by the following linear cost function:

\[
J = \sum_{i=1}^n c_i \rho_i
\]

(28)
where the weight coefficients \( v_i \) are used to emphasize the importance of specific error variables in the optimization and the \( p_i \) are defined in (26). The problem is thus formalized as the following minimization problem

\[
\text{minimize} \quad J \quad \text{subject to MI}s \quad (29)
\]

It should be observed that problem (29) is not linear due to the presence of the product between some of the optimization variables in the constraints. The engineering approach used to face the nonlinear optimization problem (29) was to fix a subset of the optimization variables \( P, \alpha, \rho_1, \ldots, \rho_n \), so that the reduced optimization problem turns out to be linear in the remaining variables. Then, the resulting linear optimization problem is repetitively solved by performing a discretized grid search on the subspace of the fixed variables. Finally, the solution leading to the minimum value of \( J \) on the grid is considered as the suboptimal solution. Further details on the optimization procedure can be found in [13].

**Note-1.** Infeasibility of the optimization problem implies that at least one of the constraints in (29) cannot be satisfied for the current set of design parameters. Since the proposed method provides sufficient conditions for the satisfaction of the requirements then infeasibility does not means, in general, that the performance cannot be fulfilled by the adaptive system, but simply that the method is not able to guarantee the performance for the current values of the design parameters.

**SUM OF SQUARE OPTIMIZATION**

Consider a set of variables \( e = [e_1, e_2, \ldots, e_n]^T \), the set \( \mathbb{R}[e] \) denotes the set of polynomials in \( e \) with real coefficients. Denote with \( \mathbb{N} \) the finite set of positive integers \( \{a_1, a_2, \ldots, a_n\} \) such that \( a_{\min} \leq a_i \leq a_{\max} \). A monomial is defined as \( z_i = e_1^{a_1} e_2^{a_2} \cdots e_n^{a_n} \). The sum \( a_i = a_1 + a_2 + \ldots + a_n \) indicates the degree the \( i \)-th monomial. A polynomial is a linear combination of monomials, and is denoted as \( p = \sum_{\text{order}} c_i z_i \), where \( c = [c_1, c_2, \ldots, c_n] \) is a vector of "base" monomials of length \( q \). A polynomial \( p \) is a Sum of Squares (SOS) if there exist polynomials \( f_i \) such that \( p = \sum_{i=1}^m f_i^2 \). The set of SOS polynomials is a subset of \( \mathbb{R}[e] \) and is denoted as \( \text{SoS}[e] \). We observe that if \( p \) is a sum of square polynomial, then \( p(e) \geq 0 \) \( \forall e \in \mathbb{R}^n \). Therefore \( p(e) \in \text{SoS} \) is a sufficient condition for a polynomial to be globally non-negative.

SOS programs are optimization problems involving SOS polynomial constraints. Further details on SOS optimization can be found in [15] and [16]. As highlighted in the introductory section the MRAC problem under investigation can be posed within the SOS optimization framework. This aspect will be discussed in the next section.

**ADAPTIVE CONTROL BASED ON HIGH ORDER LYAPUNOV FUNCTIONS**

Consider the new Lyapunov function

\[
V_t(e, \hat{W}) = V_e(e) + \hat{W}^T \Gamma^{-1} \hat{W} \quad (30)
\]

where \( V_e(e) \) is a generic high degree polynomial function to be computed by the SOS optimized design. Note that in this study, for simplicity, the part depending on \( W(t) \) has retained the standard quadratic structure. The analysis of stabilities follows the same line used in the previous quadratic LMI based approach with the important difference that in this case \( V_e(e) \) is an unknown high order function. The time derivative of \( V_t(t) \) is:

\[
\dot{V}_t = \frac{\partial V_e}{\partial e} \dot{e} + 2 \hat{W}^T \Gamma^{-1} \dot{\hat{W}} \quad (31)
\]

Substituting (13) in (31) results

\[
\dot{V}_t = \frac{\partial V_e}{\partial e} (A_m e + B e) - \frac{\partial V_e}{\partial e} B \hat{W}^T x + 2 \hat{W}^T \Gamma^{-1} \dot{\hat{W}} \quad (32)
\]

The adaptation law for the weights is based on parameter projection, and has the form

\[
\dot{w}(t) = \frac{\Gamma}{2} \cdot \text{Proj} \left( \hat{W}(t), x(t), \frac{\partial V_e(t)}{\partial e} B \right) \quad (33)
\]

It is immediate to recognize that (33) is a generalization of (12) applied to a generic \( V_e(e) \). Substituting (33) in (32) results

\[
\dot{V}_t = \frac{\partial V_e}{\partial e} (A_m e + B e) + \hat{W}^T \left[ -x \frac{\partial V_e}{\partial e} B + \text{proj} \left( \hat{W}, x \frac{\partial V_e}{\partial e} B \right) \right] \quad (34)
\]

Using now the projection operator property:

\[
\hat{W}^T \left[ -x \frac{\partial V_e}{\partial e} B + \text{proj} \left( \hat{W}, x \frac{\partial V_e}{\partial e} B \right) \right] \leq 0, \text{ results immediately the following condition}
\]

\[
\dot{V}_t(t) \leq \frac{\partial V_e}{\partial e} (A_m e + B e) \quad (35)
\]

**SUM OF SQUARE (SOS) DESIGN AND PERFORMANCE VERIFICATION FRAMEWORK**

In this section we recast the problem of robust positive invariance and performance requirements based on the high order Lyapunov function \( V_t(t) \). For this purpose define the level set generated by \( V_t(t) \) as

\[
\Omega_t = \left\{ e, \hat{W} \in R^{n+1} \mid V_t(e) + \hat{W}^T \Gamma^{-1} \hat{W} \leq 1 \right\} \quad (36)
\]

The definition of robust positive invariance for the set \( \Omega_t \) is identical to the definition-1 given or the set \( \Omega \).
Requirements on the Lyapunov function
A necessary condition for the positive invariance of $\Omega_i$ is that the Lyapunov function must be positive definite, that is: $V_i(e, \dot{W}) > 0$ for all $(e, \dot{W}) \neq (0, 0)$. This is achieved by requiring that $V_i(e) \geq e_s^T e$ $\forall e$, where $e_s > 0$ is an arbitrary small number. According to the SOS constraint this inequality is transformed in the following SOS constraint
\[ V_i(e) - e_s^T e^T \in \text{SoS polynomials} \] (37)

Requirements of Robust Positive Invariance for $\Omega_i$
A sufficient robust positive invariance condition for $\Omega_i$ is given by the following theorem.

Theorem 3: Consider the closed loop error dynamics (13), the adaptation laws (33) in the presence of bounded disturbance $W \in \Pi_W$ and $e \in \Pi_e$. If there exist positive scalars $s_1$ and $s_2$ and a Lyapunov function $V_i(t)$, satisfying the condition
\[ \frac{\partial V_i}{\partial e} (A_me + Be_M) + s_1 V_i(t) + \dot{W}^T \Gamma^{-1} \dot{W} - 1) + s_2 (W_m^2 - \dot{W}^T \dot{W}) \leq 0 \]
for $e_s = +e_M$ and $e_s = -e_M$ (38)
then, $\Omega_i$ is robustly positively invariant.

Proof: Following the same approach used in the demonstration of theorem 1 results that 1) the requirement $V_i(t) \leq 0$ is verified if $\frac{\partial V_i}{\partial e} (A_me + Be_M) \leq 0$, which derives immediately from (35). 2) Condition $V_i(t) + \dot{W}^T \Gamma^{-1} \dot{W} - 1$ defines the region external to the boundary $e \Omega_i$ and descends from (36); 3) conditions $W_m^2 - \dot{W}^T \dot{W} \geq 0$ define the admissible region for the weights estimation error and descend from the boundedness of $\dot{W}(t)$ induced by the projection operator (33). 4) Finally, noticing that the scalar uncertain $e$ is polytopic, results that the validity for each value of $e$ in $\Pi_e$ is equivalent to simply testing the condition on the 2 vertices $[\pm e_M, +e_M]$ of the uncertainty domain $\Pi_e$.

Using the extended S procedure, conditions 1), 2), 3) and 4) can be transformed immediately into the inequality (38)\]

The SOS optimization formulation the two inequality conditions (38) are transformed into the following SOS constraints
\[ -\left[ \frac{\partial V_i}{\partial e} (A_me + Be_M) + s_1 V_i(t) + \dot{W}^T \Gamma^{-1} \dot{W} - 1) + s_2 (W_m^2 - \dot{W}^T \dot{W}) \right] \in \text{SoS polynomials} \]
for $e_s = +e_M$ and $e_s = -e_M$ (39)

Requirements on the tracking error
Componentwise tracking error requirements are identical to those specified in (26). Define the level set $\Omega_{e_i} = \{ e_r(e) \leq 1 \}$ and the polyhedron $\Pi_{e_i} = \{ e \mid e_r(t) \leq \rho_i e_M \quad i = 1, ..., n \}$. Applying the extended S procedure, the containment $\Omega_{e_i} \subseteq \Pi_{e_i}$ is verified if the following constraints are satisfied
\[ (\rho_i^2 e_M^2 - e_s^2) - s_{i+2} (1 - V_i) > 0 \quad i = 1, ..., n. \] (40)
where $s_{i+2}$ are arbitrary positive scalars. As before, these requirements are transformed is SOS constraints, that is
\[ (\rho_i^2 e_M^2 - e_s^2) - s_{i+2} (1 - V_i) \in \text{SoS polynomials} \quad i = 1, ..., n \] (41)

SOS Optimization procedure
To set up the SOS optimization we are requested to specify the structure of the polynomial function $V_i(e)$. In this study it was assumed that $V_i(e) \in \text{SoS}$, that is it can be written in the form
\[ V_i(e) = z^T M_z(e) \in \text{SoS polynomials} \] (42)
where $M>0$ is an unknown positive definite matrix to be computed in the SOS optimization and $z(e)$ is a defined base of monomials. For instance if $e = [e_1, e_2]$ and $\alpha_{max} = 2$, then we have the following base of monomials $z = [z_1, z_2, z_1^2, z_2^2]$. The resulting high order Lyapunov function can be rewritten as follows
\[ V_1 = z(e)^T M_z(e) + \dot{W}^T \Gamma^{-1} \dot{W} \] (43)

Considering the Lyapunov function (43) the purpose of the SOS optimization is to select a $V_1$ such that the size of the set $\Omega_{e_i} = \{ e \mid V_i(e) \leq 1 \}$ is minimized. As before, we selected the following linear cost function
\[ J = \sum_{i=1}^{n} V_i(e) \] (44)
The design problem is fully formalized as the following SOS minimization problem
\[ v_1, v_2, ..., v_n, a_1, a_2, ..., a_n \text{subject to SOSs } (39, 41, 42) \] (45)

Note that problem (45) is not linear due to the presence of the product between some of the optimization variables. The problem was solved by applying a two-steps iterative procedure that works as follows:
\[ \text{-step-a) Fix the parameters } s_1, s_2, ..., s_n \text{ and then optimize for } J \text{ assuming as optimization variables the elements of the } M \text{ matrix that define Lyapunov function } V_1(e). \text{ Then go to step-b).} \]
\[ \text{-step-b) Fix } V_1 = z^T M_z \text{ achieved in a), then optimize } J \text{ assuming the } s_1, s_2, ..., s_n \text{ as free optimization variables. Then go to step-b).} \]

The above procedure is iterated until J does converge to a minimum.

Initialization of the iterative procedure
The iterative SOS optimization requires a feasible initial solution. In the first iteration we fixed the parameters $s_i$ to the (corresponding) values provided by the LMI optimization (29) that were derived using a quadratic Lyapunov function. In fact, since in the SOS approach
\( V_e(e) \) has a degree (typically \( \geq 2 \)), then, the solution obtained in the particular case of a quadratic Lyapunov function will provide, surely, a feasible solution also for the SOS problem that is based on quadratic or higher order Lyapunov functions.

**ILLUSTRATIVE AEROSPACE APPLICATION**

The performance of the two design methods were compared using the short period longitudinal dynamics of the F16 aircraft model reported in [20]. The dynamical model is

\[
\begin{bmatrix}
\dot{x}_1(t) \\
\dot{x}_2(t) \\
\dot{x}_3(t)
\end{bmatrix} =
\begin{bmatrix}
-1.01 & 0.90 & 0 \\
0.82 & -1.07 & 0 \\
1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
x_1(t) \\
x_2(t) \\
x_3(t)
\end{bmatrix} +
\begin{bmatrix}
-0.0022 \\
-0.1756 \\
0
\end{bmatrix} (\Delta(x)+u(t))+
\begin{bmatrix}
0 \\
0 \\
-1
\end{bmatrix} r(t)
\]

where \( x_1(t) \) is the angle of attack \( \alpha(t) \) in radians, \( x_2(t) \) is the pitch rate \( q(t) \) in rad/sec, and \( x_3(t) \) is the augmented state associated with the integral of the angle of attack tracking error \( \alpha_0(t) \). The \( u(t) \) is the elevator deflection in degrees and \( r(t) \) is the reference command for the angle of attack that was fixed in this study as a square wave with amplitude 5 degs. The input uncertainty \( \Lambda \) was assumed time varying and equal to \( \Lambda_{\text{max}} = 1.25 \) and \( \Lambda_{\text{min}} = 0.75 \). The upper bound for the approximation error in (2) was fixed at \( x_M = 0.5 \). The bound for the norm of the vector \( \|W\| \) was fixed at \( W_M = 7.5 \). In the simulation studies the matched uncertainty \( \Delta(x_p) = W_1 x_1 + W_2 x_2 \) was fixed at \( W_p = [-4; 6] \), while the bounded unmodelled error \( \varepsilon(t) \) was assumed time varying and equal to \( 0.1 \sin(\pi t) \).

The reference model was derived from (46) in uncertain free case assuming \( \Lambda = 0; \Delta(x) = 0; \varepsilon = 0 \). The baseline linear controller \( K_0 \) was designed (in nominal conditions without considering any sort of uncertainty) using the LQR technique with weight matrices equal to \( Q = \text{diag}[0.1; 100] \) and \( R = 1 \), that is \( K_0 \) was designed, essentially, to be a tracking controller for the angle of attack. The LQR design produced a control matrix \( K_0 = [10.87; 6.05; 10.0] \). Considering the overall uncertainty vector \( W(t) \) in set (10), this was upper bounded with: \( W_M = [\Lambda_{\text{max}} |W_{M_1}^T M + (\Lambda_{\text{min}} - 1) K_0]| = 15.66 \). The adaptation rate was fixed at \( \Gamma = 1000 \).

As for the performance requirements for the error components we fixed \( |e_1(t)| \leq e_{M1} = 1 \) rad and \( |e_2(t)| \leq e_{M2} = 1 \) rad/sec. The cost function used both in the LMI and in the SOS optimization was \( J = 1 \cdot \rho_1 + 1 \cdot \rho_2 \).

**LMI-based estimation of ellipsoidal domains of operation.**

Based on the above uncertainties and settings we set up the optimization procedure for the design of ellipsoidal invariant based on the solution of the optimization problem (29) where the set of free optimization variables are \( P, \alpha, \beta, \rho_1, \rho_2 \). Considering \( \alpha \) as a grid optimization parameter, the problem (29) reduces to the following

\[
\text{minimize } J \quad \text{subject to MIs (47)}
\]

**FIGURE 1.** The optimized ellipsoidal invariant set \( \Omega_e \) and the corresponding bounding set \( \Pi_{e\rho} \) generated solving the LMI optimization (47) (projection in the \( e_1-e_2 \) plane). The set \( \Pi_{e\rho} \) represents the optimized bounding set generated with a forth order Lyapunov function via SOS optimization (45) and \( \Omega_{e1} \) is the corresponding invariant set.

Figure 2 shows some closed loop tracking error trajectories in the \( e_1-e_2 \) plane with initial conditions taken on the boundary of the set \( \Omega_e \). As expected, being \( \Omega_e \) invariant, the closed loop trajectories are completely contained in the invariant set \( \Omega_e \). Figure 3 shows the evolution of the reference model angle attack \( x_{\text{ref}}(t) \) as long as the actual angle of attack \( x_1(t) \) provided by the uncertainty adaptive systems for the considered sample trajectories. It can be observed that, following the initial transitory, the adaptation process is able to compensate most of the uncertainty. The compensation cannot be perfect due to the bounded persistent error \( \varepsilon(t) \). Figure 4 shows the corresponding overall control signal (7) that is the elevator deflection.
The iterative optimization procedure for the SOS optimization is the definition of the structure for the polynomial function. The first step of the SOS optimization resulted equal to 31.

The resulting optimized fourth order Lyapunov function was

\[ V(e) = 3505e_1^2 - 3101.0e_1e_2 + 1319.0e_1e_3 + 8.661e_2e_3 + 2213.0e_1e_2^2 + 761.5e_1e_2e_3 - 3.406e_2e_2^2 + 2457.0e_1e_3^2 - 1.83e_2e_3^2 + 0.0157e_3^3 - 911.6e_1e_3^3 + 2142.0e_2e_3^3 - 3.735e_2e_3^2 + 2068.0e_2e_3^2 + 4.252e_3e_3^2 - 0.0193e_1e_3^2 + 1530.0e_2e_3^2 + 1.07e_3e_3^2 - 0.003018e_2e_3 + 1396.0e_2e_3^2 + 2358.0e_3e_3^2 - 1.148e_2e_3^2 + 2071.0e_3e_3^2 - 2.002e_3e_3^2 + 0.006953e_3^2 + 859.4e_3e_3^2 - 1.107e_3e_3^2 + 0.004814e_3^2 + 722.8e_3^2 - 1.669e_3^2 + 0.003301e_3^2. \]

The above \( V(e) \) was then used to compute the invariant set \( \Omega_{V} \) and to build the polynomial adaptation law (33). In Figure 1 it is also shown the optimized bounding set \( \Pi_{SV1} \) and the corresponding forth order \( \Omega_{SV1} \) invariant set \( (V(e)=1) \). It is immediately observed that the set \( \Omega_{SV1} \) is much smaller than \( \Omega_{V} \). This fact proves the effectiveness of employing high order Lyapunov functions and SOS optimization instead of the standard quadratic Lyapunov functions and LMI optimization for the computation of not conservative bounding regions of operation for the tracking error. Figure 5 shows some error trajectories in the \( e_1e_2 \) plane with initial conditions taken on the boundary of the projected set \( \Omega_{SV1} \). As expected, also in this case, the trajectories are completely contained in the invariant set \( \Omega_{SV1} \). Figure 6 shows the reference command \( x_{r1}(t) \) and the actual \( x_1(t) \) for the sample trajectories, while Figure (7) shows the corresponding elevator deflections \( u(t) \).

**Comparison with pure linear LQR control**

To have an idea of the effectiveness of the proposed adaptive control for the tracking of the angle of attack for the considered uncertain model of the F16, an experiment was carried by disabling the adaptive control so that the system is controlled only by the baseline linear controller \( K_o \). Figure 8 shows the tracking error responses for initial condition taken on the boundary of the set \( \Omega_{SV1} \) while Figure 9 shows the corresponding control effort. It is observed that the baseline linear controller \( K_o \) is unable to compensate satisfactory the modelling uncertainties. The
difference between adaptive (both LMI-based and SOS-based) and non-adaptive control is evident for large values of $t$ ($t > 100$) when the adaptation mechanism has almost completely compensated the uncertainty. It is also observed that the remarkable performance improvement provided by the adaptive controllers is achieved without requiring more aggressive control contributions. In fact, the control authority of the adaptive controllers shown in Figure 4 and 6 is comparable to the control authority of the linear controller $K_0$ shown in Figure 9.

![Graph](image1)

**Figure 5.** Closed loop trajectories starting from the boundary of $\Omega_{e_1}$ under SOS-based adaptive control.

![Graph](image2)

**Figure 6.** Angle of attack tracking performance under SOS-based adaptive control (initial conditions on the boundary of $\Omega_{e_1}$).

CONCLUSIONS

In this work it was proposed a new method for the estimation of non-conservative operative regions for the tracking error of uncertain MRAC systems. The approach is based on the optimized design of minimal size invariant sets that originate from high order polynomial control Lyapunov functions. The optimized design of the invariant set is carried out using SOS optimization tools. The SOS design was then compared with an LMI-based design of ellipsoidal invariant regions based on standard quadratic Lyapunov functions. A detailed simulation study carried out on an F16 aircraft model has shown that the SOS based design is able to estimate invariant operation regions that are much smaller that the regions produce with standard quadratic Lyapunov functions. A final study has also shown that either SOS or LMI based adaptive controllers perform much better than the baseline LQR controller in the presence of significant uncertainties.

![Graph](image3)

**Figure 7.** Actuator deflection $u(t)$ under the SOS-based adaptive control (initial conditions on the boundary of $\Omega_{e_1}$).

![Graph](image4)

**Figure 8.** Angle of attack tracking performance under pure Linear control $K_0$ (initial conditions on the boundary of $\Omega_{e_1}$).
FIGURE 9. Actuator deflection $u(t)$ under the linear control $K_0$ (initial conditions on the boundary of $\Omega_e$).

REFERENCES


