Adaptive Augmentation of an Optimal Baseline Controller for a Hypersonic Vehicle

Manfredo Martinino[1], Joel Bordeneuve-Guibé[1] and Vincent Morio[2]

***Abstract*— The aim of this work is to design an adaptive augmentation of an optimal baseline controller for the flight dynamics of a HSV (HyperSonic Vehicle) using model-reference adaptive control (MRAC). The baseline controller is able to track a bounded input with a desired dynamic and with zero steady state error. The adaptive augmentation is used to compensate the uncertainties, due to a poor knowledge of the physical system, that may degrade the baseline closed-loop performances. The main contribution of this paper is the combination of a MDZM and a projection operator together with two modifications, proposed by the authors, to improve the performances of the closed loop. The adaptive controller has been implemented in Simulink and integrated to a NASA X-30 model. Simulation results are provided to show the effectiveness of the augmented controller notably in presence of aerodynamic uncertainties and control degradations.**

I. INTRODUCTION

The objective of this study is to design an adaptive augmentation of an already existing LQR-PI baseline controller for a hypersonic vehicle. According to Eugene Lavretsky [1], the adaptive controller would try to perform an online estimation of the process uncertainty and then produce a control input to anticipate, overcome, or minimize undesirable deviations from the prescribed closed-loop plant behavior.

Nowadays, adaptive control is widely used for systems regulation both as the only source of control as well as a support to a baseline controller. In particular, the latter case consists in merging the adaptive contribution with the one obtained from the baseline. On the other hand, the idea of an adaptive augmentation of an existing baseline controller turned out in many works such as [2] or [3] to be the best way to ensure rejection of the disturbances, to robustly cancel the model uncertainties and to exploit an already existing optimized controller. Actually, the rational for using an augmentation approach (as opposed to all adaptive) stems from the fact that in most realistic applications, a system may already have a baseline controller, which is often designed to provide acceptable level of performance and robustness over the flight envelope.

A serious and intrinsic problem of the adaptive laws, no matter if they are used alone or mixed with pre-existing baselines, is the influence the presence of unmatched disturbances may have on gains calculation. For example, a bounded process noise may easily lead to parameters drift. The adaptive parameters are actually calculated in a chain of nonlinear integrators where a wind-up effect can arise due to saturation. In order to make the adaptive laws more robust, a certain numbers of methods have been proposed. One of the most advanced, created by combining a *Modulated Dead-Zone Modification* (MDZM) and a *projection operator*, is proposed in [1] as well as in [3]. The contribution of the MDZM is that of protecting the adaptive parameters from drifting with noise, while the projection operator bounds the overall adaptive process and prevents the nonlinear integrators from winding-up.

The main task of an adaptive law is to estimate and manage the system uncertainties. In order to do this, the idea is to build an opportune set of functions to approximate these uncertainties with a finite linear combination of known basis functions and unknown constant parameters. In this study, a Neural Network (NN) with 1 hidden layer, 30 neurons, and RBFs (Radial Basis Functions) will be used to model the system matched uncertainties.

This paper also presents simple but effective methods the authors have used to solve two typical issues arising when adaptive laws are used. The first one concerns how to obtain the best matching between the regulated system and the reference model, knowing that the connection between them is the state tracking error. In this study, the components of the error vector are weighted differently in order to achieve the best tracking performance of a certain target variable. The second issue concerns the evolution dynamics of the adaptive gains. As presented in section II, in the classic theory the adaptive gains are purely the result of the integration of their derivatives. Differently, we will propose an improvement of gain calculation using a derivative contribution. The latter will lead to a greater readiness in gains dynamics ensuring a better reference tracking.

II. THEORETICAL BASIS

A. Adaptive augmentation

Let us suppose the MIMO system we would like to control being described by the following open-loop state space representation [1]:

$$\begin{cases} \dot{x_p} = A_p x_p + B_p \Lambda (u + f(x_p)) \\ y = C_p x_p + D_p u \end{cases} \tag{1}$$

where n_p and m are the dimensions of the system state x_p and of the control u, respectively. Also, we assume that

[1]Manfredo Martinino and Joel Bordeneuve-Guibé are with ISAE-Supaéro and Université de Toulouse, Toulouse, France joel.bordeneuve@isae-supaero.fr

[2]Vincent Morio is with Direction Générale de l'Armement - Maîtrise de l'Information, Rennes, France vincent.morio@intradef.gouv.fr

$A_p \in \mathbb{R}^{n_p \times n_p}$ and $B_p \in \mathbb{R}^{n_p \times m}$ are known, while $\Lambda \in \mathbb{R}^{m \times m}$ is an unknown diagonal matrix with strictly positive diagonal elements λ_i. The pair $(A_p, B_p\Lambda)$ is assumed controllable, and the constant uncertainty Λ is introduced to model possible imperfections in the system control channels. The unknown nonlinear function $f(x) : \mathbb{R}^{n_p} \to \mathbb{R}^m$ represents the system matched uncertainty. This function can be written as $f(x) = \Theta^T \Phi(x_p)$, where $\Theta \in \mathbb{R}^{N,m}$ is the unknown constant matrix of ideal parameters, and $\Phi(x_p) \in \mathbb{R}^N$ represents the known locally Lipschitz-continuous regressor vector.

By augmenting the system state with an output error feedback term $e_{yI} = \int_0^t (y - y_{cmd}) d\tau$, we get:

$$\underbrace{\begin{pmatrix} \dot{e_{yI}} \\ \dot{x_p} \end{pmatrix}}_{\dot{x}} = \underbrace{\begin{pmatrix} 0_{m,m} & C_p \\ 0_{n_p,m} & A_p \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} e_{yI} \\ x_p \end{pmatrix}}_{x} + \underbrace{\begin{pmatrix} D_p \\ B_p \end{pmatrix}}_{B} \Lambda(u + f(x_p)) + \underbrace{\begin{pmatrix} -I_{m,m} \\ 0_{n_p,m} \end{pmatrix}}_{B_{ref}} y_{cmd} \tag{2}$$

Having defined an augmented state x, an LQR controller with integral action (LQR-PI) K_{lqr} is designed in order to track the closed-loop reference model defined by:

$$\begin{cases} \dot{x}_{ref} &= A_{ref} x_{ref} + B_{ref} y_{cmd} \\ y_{ref} &= C_{ref} x_{ref} \end{cases} \tag{3}$$

where $A_{ref} = A - BK_{lqr}^T$ and $C_{ref} = C - DK_{lqr}^T$. Then, the adaptive controller can be designed according to theoretical results presented in [1]. The basic idea is to add an adaptive control input to the one produced by the baseline controller and then to inject the result in the system (see Fig. 1). Then, one can write:

$$u = u_{bl} + u_{ad} = -K_{lqr}^T x + \left[-\hat{K}_u^T u_{bl} - \hat{\Theta}^T \Phi(x_p) \right] \tag{4}$$

where, beyond the already mentioned quantities, u_{bl} is the baseline input and $\hat{K}_u$, $\hat{\Theta}$ are the two adaptive gains calculated in order to enforce the state tracking error $e = x - x_{ref}$ to become asymptotically zero:

$$\begin{aligned} \dot{\hat{K}}_u &= \Gamma_u u_{bl} e^T P_{ref} B \\ \dot{\hat{\Theta}} &= \Gamma_\Theta \Phi(x_p) e^T P_{ref} B \end{aligned} \tag{5}$$

The gain computation includes the two so called *tuning*

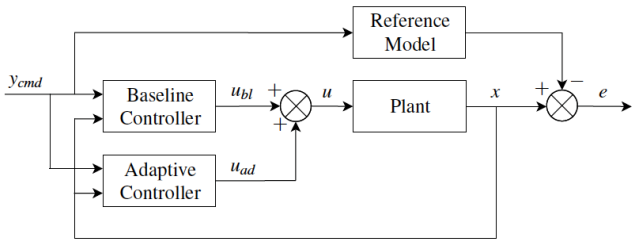

Fig. 1. Classic model-reference adaptive architecture.

knobs Γ_u and Γ_Θ, whose choice is experience-based, and the P_{ref} matrix. The latter is the unique symmetric positive-definite solution of the algebraic Lyapunov equation:

$$A_{ref}^T P_{ref} + P_{ref} A_{ref} = -Q_{ref} \tag{6}$$

with some appropriately chosen matrix $Q_{ref} = Q_{ref}^T > 0$.

B. Modifications for robustness enhancement

The adaptive process is actually extremely critical and the gains values can be perturbed and spoilt by the presence of unmatched uncertainties, as noise. In addition, it is not recommended to let the gains evolve when the state tracking error becomes very small. These considerations have led to the introduction of designing some mechanisms able to enhance the robustness of the overall adaptive process.

1) Modulated dead-zone modification: The dead-zone modification (DZM) stops the adaptation process when the norm of the tracking error becomes smaller than the prescribed value e_0. It is very powerful especially when the noise disturbs the gain adaptation or when the adaptation does not stop even with low values of state tracking error. The modulated DZM (MDZM) is a small adjustment of the DZM proposed by Slotine and Coetsee in [4] to accommodate the shut-down of the adaptation process without discontinuities that can lead to chattering (high-frequency oscillations). The equation for gain derivative calculation is then:

$$\begin{aligned} \dot{\bar{\hat{\Theta}}} &= \Gamma_{\bar{\Theta}} \bar{\Phi}(x) \mu(||e||) e^T P_{ref} B \\ \mu(||e||) &= \max\left(0, \min\left(1, \frac{||e|| - \delta e_0}{(1-\delta) e_0}\right)\right) \end{aligned} \tag{7}$$

where $\bar{\hat{\Theta}} = \begin{pmatrix} \hat{K}_u^T & \hat{\Theta}^T \end{pmatrix}^T$, $\Gamma_{\bar{\Theta}} = \begin{pmatrix} {\Gamma_u}^T & {\Gamma_\Theta}^T \end{pmatrix}^T$ and $\bar{\Phi}(x) = \begin{pmatrix} {u_{bl}}^T & \Phi(x_p)^T \end{pmatrix}^T$.

The Lipschitz-continuous modulation function $\mu(||e||)$ contains two new variables to be tuned: the error boundary e_0 and a constant, $0 < \delta < 1$, that defines the slope of the modulation function.

Note that one can use Lyapunov-based arguments to prove the bounded tracking and the uniform ultimate boundedness (UUB) of all signals.

2) Projection operator: The projection operator bounds the overall adaptive process and prevents the nonlinear integrators from winding-up. It was introduced by Kreisselmeier and Narendra in [5] and can be seen as a way to limit, through a projection process, the values the adaptive gains can assume. The idea is to associate a convex function $f_j : \mathbb{R}^n \to \mathbb{R}$ to each column ($\Theta_j \in \mathbb{R}^{n \times 1}$) of the adaptive gain matrix. The function is designed to give back, as result, $f(\Theta_j) \leq 0$ for acceptable gain values, and $0 < f(\Theta_j) < 1$ in case of gains that are exceeding the imposed limits. The adaptive gains are then calculated as:

$$\dot{\bar{\hat{\Theta}}} = Proj(\Gamma_{\bar{\Theta}} \bar{\Phi}(x) \mu(||e||) e^T P_{ref} B) \tag{8}$$

and the chosen convex function is:

$$f_j = f(\hat{\Theta}_j) \frac{(1 + \varepsilon_j^\Theta)||\hat{\Theta}||^2 - ({\Theta_j}^{max})^2}{\varepsilon_j^\Theta ({\Theta_j}^{max})^2} \tag{9}$$

where the new parameters to be tuned are ${\Theta_j}^{max}$ and ${\varepsilon_j}^{\Theta}$. The first one is the pre-specified bound and the second one represents the projection tolerance. This second parameter plays quite the same role of the modulating function in the MDZM. One can prove that the projection operator contributes to the negative semi-definiteness of the Lyapunov function thus ensuring the UUB property of all signals in the corresponding closed-loop system.

C. Neural network

In this study, a Neural Network (NN) has been used to model the system matched uncertainties. As activating function, it has been decided to use the Gaussian Radial Basis Function (RBF), expressed as:

$$\phi(x,x_c) = e^{-(x-x_c)^T W(x-x_c)} = e^{-||x-x_c||_W{}^2} \tag{10}$$

where, $x \in \mathbb{R}^n$ is the input, $x_c \in \mathbb{R}^n$ is the center, and $W = W^T > 0$ is a diagonal positive-definite symmetric matrix of weights with values:

$$W_i = \frac{1}{2\sigma_i^2} \qquad (i = 1,...,N) \tag{11}$$

where σ_i represents the width of the i^{th} Gaussian function, thus obtaining:

$$\phi_i(x) = e^{-\frac{||x-C_i||^2}{2\sigma_i^2}} \tag{12}$$

The RBFs functions have been chosen because they have been shown to be capable of approximating generic classes of functions on compact sets and within any pre-specified tolerance. This result is known as Universal Approximation Theorem [1].
Finally, the regressor vector will be composed of as many RBFs as neurons in the network, plus a constant bias. The latter, equal to 1, will model an always active neuron, which corresponds to a pure bias on the control input. Thus, the regressor vector is given by:

$$\Phi(x_P) = \Big(\phi_1(x_P),...,\phi_{N_n}(x_P),1\Big)^T \tag{13}$$

where N_n represents the number of neurons in the network ($N_n = 30$ here). This structure will make the adaptive augmentation able to approximate the system uncertainties.
The modifications for robustness (the MDZM and the projection operator), will act differently on the regressor vector produced by the NN. Following eq. (7), the MDZM will actually scale the regressor vector by a factor $\mu(||e||)$, while the projection operator acts on the whole adaptive gain $\hat{\Theta}$ as showed in (8).

III. MODEL DESCRIPTION

The model used in this paper is a generic, horizontal take-off, single-stage-to-orbit (SSTO) configuration, studied by NASA in early 80's in the frame of the National Aerospace Plane (NASP) program, also known as X-30 (see Fig. 2). Success of this project was theoretically possible through development of advanced technologies in the areas of aerodynamics, materials, structures, flight control and propulsion. In particular, two of these "enabling" technologies were related to the propulsion system, which would consist of an air-breathing supersonic combustion ramjet, or scramjet, and also to the development of active cooling systems and advanced heat-resistant materials able to maintain structural integrity at very high temperatures. Despite cancellation of the NASP program in 1995, hypersonic air-breathing vehicles are still an active field of research, as evidenced by the large number of countries interested in these technologies.

Fig. 2. Artist view of NASA X-30 concept.

From the flight control system design perspective, current studies show that it is necessary to take into account a certain number of physical phenomena specific to this type of vehicle during early design stages:

- Model uncertainties that may take important values,
- Interactions between the propulsion system and aerodynamics,
- Flexible modes of structure,
- Dispersions and measurement noises.

These various aspects, combined with drastic requirements in terms of desired performance and severe operating constraints related to the technologies employed, make the autopilot particularly sensitive to the choices made in terms of architecture and aeropropulsive design, and may call the use of conventional control methods into question. As a result, the implementation of an intelligent flight control law, such as the one proposed in this paper, becomes critical to guarantee the success of the mission.

A six degree-of-freedom Matlab-Simulink implementation of the X-30 has been considered in this study using aerodynamic, propulsion and mass data borrowed from [6]. The atmosphere model follows the US 1976 standard and the flight dynamics model considers a rotating, spherical Earth. No flexible modes are included. The system is described by the state vector $X \in \mathbb{R}^{12\times 1}$:

$$X = (\varphi\ \lambda\ r\ V_T\ \alpha\ \beta\ p\ q\ r\ \phi\ \theta\ \psi)^T \tag{14}$$

where φ is the geodetic latitude, λ the longitude, r the vector-radius, V_T the total velocity, α the angle-of-attack, β the sideslip angle, ϕ, θ, ψ Euler angles, and p, q, r are the absolute angular velocities around roll, pitch and yaw axis respectively. The system is controlled by the input vector $U \in \mathbb{R}^{4\times 1}$ such that:

$$U = (\delta_e\ \delta_a\ \delta_r\ \varphi_{st})^T \tag{15}$$

where δ_e, δ_a and δ_r represent respectively the deflection of elevator, ailerons and rudder, and φ_{st} is the fuel equivalent ratio (FER). The model also includes actuators dynamics and saturations, thought these are not considered in the linearized state space representation of the vehicle, as well as simplified sensor models. The underlying model can be expressed compactly as a nonlinear model given by:

$$\dot{X} = f(X,U) \tag{16}$$

In order to facilitate the control design, a trimming routine is used to linearize (16) at a given flight point defined in terms of altitude, Mach number and mass:

$$\dot{x}_p = A_p x_p + B_p u + \varepsilon(t) \tag{17}$$

where ε is the linearization error, which is assumed to be small, $x_p = X - X_0$ and $u = U - U_0$ are respectively the perturbations from the trim state X_0 and trim input U_0 satisfying $f(X_0,U_0) = 0$, and

$$A_p = \left.\frac{\partial f(X,U)}{\partial X}\right|_{\substack{X=X_0\\U=U_0}}, \quad B_p = \left.\frac{\partial f(X,U)}{\partial U}\right|_{\substack{X=X_0\\U=U_0}} \tag{18}$$

A modal analysis of (17) shows a strong decoupling among the longitudinal, lateral and speed dynamics, thus allowing three different controllers to be synthesized independently. In this study, only the longitudinal dynamics will be considered. Therefore, the state, integral error state and input will be chosen as:

$$x_p = (\alpha \;\; q)^T, \quad e_{y_I} = \int_0^t (\alpha - \alpha_{cmd}) d\tau, \quad u = \delta_e \tag{19}$$

IV. MODIFICATIONS FOR PERFORMANCE ENHANCEMENT

In this section, two modifications are introduced in the adaptation process in order to get a better tracking performance. The first one consists in weighting differently the components of the state tracking error, while the second one adds a derivative term, in parallel with the integrator, in the calculation of the adaptive gains.

A. Modulation of the state tracking error components

Considering only the longitudinal dynamics, the state tracking error $e \in \mathbb{R}^{3\times 1}$ can be expressed as:

$$e = x - x_{ref} = \left(e_{y_I} - e_{y_{I_{ref}}},\ \alpha - \alpha_{ref},\ q - q_{ref}\right)^T \tag{20}$$

Although every term in e has the same weight, one could be more interested in a better tracking of a certain variable rather than another. In our case, an accurate tracking of the angle-of-attack is essential: this is why we propose a modification of the error vector such that:

$$e = \left(G_I(e_{y_I} - e_{y_{I_{ref}}}) \;\; G_P(\alpha - \alpha_{ref}) \;\; G_D(q - q_{ref})\right)^T \tag{21}$$

where G_I, G_P and $G_D \in \mathbb{R}^+$ are 3 weights that allow to tune the decrease rate of every component of e. In this study, more importance has been granted to the $\alpha - \alpha_{ref}$ (proportional) component with a 1.5 gain, while less influence has been given to the $q - q_{ref}$ (derivative) component with a 0.5 gain. Finally, the integral member of the error has conserved its weight with a unitary gain.

Notice that the results obtained with this modulation of the error vector could also be achieved by an adequate tuning of the Q_{ref} matrix in (6). However, this method would have a poor physical meaning and would not allow to influence the MDZM, while the proposed method directly targets the behavior of variables of interest.

Finally, we can prove that the introduced gains do not interfere with Lyapunov proof of stability. As shown in the demonstration below, only the norm of the error vector (always positive for any choice of gains) matters for stability. In fact, considering the derivative of the Lyapunov function, one can write:

$$\dot{V}(e,\Delta\bar{\Theta}) = -e^T Q_{ref} e + 2tr(\Delta\bar{\Theta}^T[\Gamma_{\bar{\Theta}}^{-1}\dot{\bar{\Theta}} - \bar{\Phi}e^T P_{ref} B]\Lambda) \tag{22}$$

and if the adaptive laws are selected in the form:

$$\dot{\bar{\Theta}} = \Gamma_{\bar{\Theta}}\bar{\Phi}(u_{bl}, x_p)e^T P_{ref} B \tag{23}$$

then,

$$\dot{V}(e,\Delta\bar{\Theta}) = -e^T Q_{ref} e < 0 \tag{24}$$

which states the UUB of $(e,\Delta\bar{\Theta})$. With some considerations presented in [1] (not influenced by our modification), one can demonstrate that the tracking error tends to zero asymptotically as $t \to \infty$.

B. Derivative contribution in adaptive gains calculation

According to (5), adaptation gains are computed thanks to the integration of the following expression (MDZM and projection operator have been discarded for sake of simplicity):

$$C(\bar{\Phi}, e) = \Gamma_{\bar{\Theta}}\bar{\Phi}(x)e^T P_{ref} B \tag{25}$$

However, computing gains this way leads to a drawback: changes in the error vector will be transmitted to the gains updating with a certain delay. During our simulation campaign, we have observed that, specifically for periodical commands, the slowness of gains evolution can profoundly affect performances of the closed-loop system. In order to fix this issue, we propose a slight change in gains calculation, by adding new terms. This addition, similar to a derivative effect, will modify (5) to:

$$\dot{\bar{\Theta}} = \int C(\bar{\Phi}, e)\cdot dt + k_D \cdot C(\bar{\Phi}, e) \tag{26}$$

where k_D is the new derivative gain and $C(\bar{\Phi}, e)$ is given by (25). Although a formal proof of stability has not been provided so far based on Lyapunov theory, simulations have shown that this modification boosts the evolution of the adaptation process and allows a significant enhancement of closed-loop performances. .

V. SIMULATION RESULTS

A. Simulation conditions

Simulations have been performed using the complete six degree-of-freedom model of the NASA X-30 vehicle described in section III, including actuator saturations and nonlinearities. Although the adaptation has been developed only for the longitudinal dynamics, coupling effects between longitudinal and lateral axes have not been inhibited. However, the commanded input will only concern the longitudinal dynamics (angle-of-attack α_{ref}). Simulations are carried out for 140 s (long enough to exhibit the long term stability of the control) for a characteristic flight condition, i.e. $h = 20000\,m$ and $M = 6$, where the trimming values are:

$$\alpha = 0.92442\,^\circ \quad q = 0\,^\circ/s \quad \delta_e = 4.2647\,^\circ$$

The first simulations are performed without system degradations or uncertainties, i.e. in nominal conditions. Then, system uncertainties will be added to the model in order to evaluate the benefits of the adaptive augmentation.
The following results have been achieved thanks to a fine tuning of the numerous simulation parameters, e.g. Γ_u, Γ_Θ and Q_{ref} for the adaptive laws; δ and e_0 for the MDZM; ${\Theta_j}^{max}$ and ${e_j}^{\Theta}$ for the projection operator; N_{number}, C_i and σ_i for the NN; G_P, G_I and G_D for the error gains; k_D for the derivative contribution.

B. Normal functioning

The commanded input is a square signal centered around the trimmed value ($\alpha = 0.92442\,^\circ$). The results are shown in Fig. 3. The new adaptive augmented controller (purple)

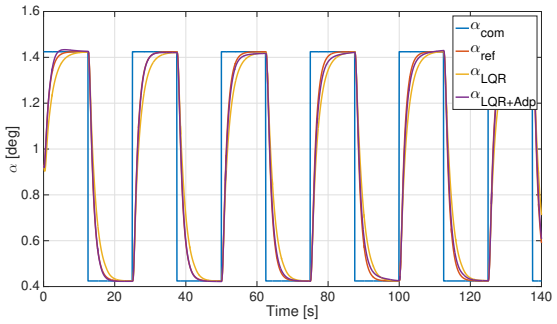

Fig. 3. Square input, $\alpha_{trim} \pm 0.5\,^\circ$, nominal conditions.

is able to track the reference model (red), resulting always faster than the baseline control (orange). To demonstrate the last sentence, let us compute the mean-square error between the adaptive and baseline controller such that:

$$\begin{aligned} \Delta e^2 &= e^2_{LQR+Adp} - e^2_{LQR} \\ &= (\alpha_{LQR+Adp} - \alpha_{cmd})^2 - (\alpha_{LQR} - \alpha_{cmd})^2 \end{aligned} \tag{27}$$

As shown in Fig. 4, the error function values are less than zero, meaning that the baseline controller produce a bigger tracking error than the augmented controller.

The evolution of the adaptation gains is shown in Fig. 5. On the top is traced the product of the adaptive gains

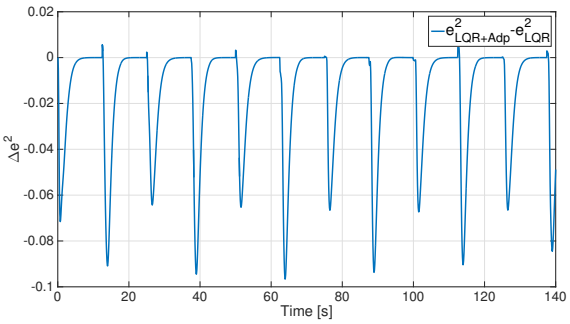

Fig. 4. Error function in nominal conditions.

with the regressor vector, i.e. $\hat{\Theta}^T \Phi(x_P)$ (see sec. II), while on the bottom is traced the product of the gains with the baseline control input $\hat{K_u}^T u_{bl}$. In particular, the colored

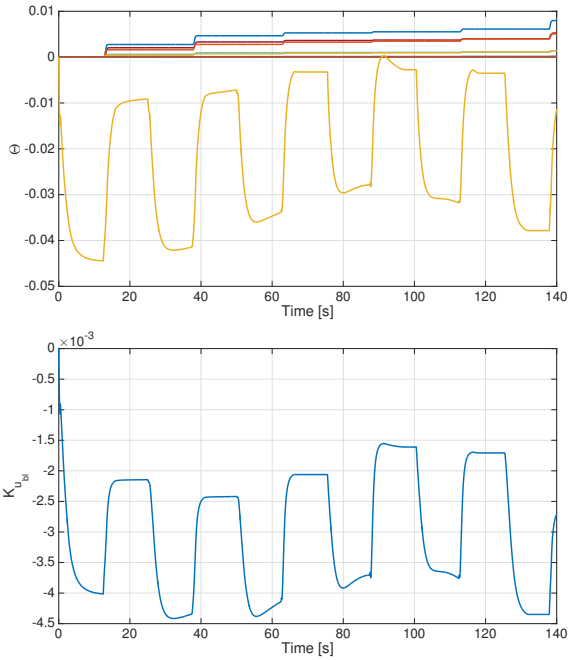

Fig. 5. Adaptive gains evolution through simulation, nominal conditions.

curves on the top figure stand for the gains of the neural nonlinear functions, while the curve with bigger variations represents the gain that multiplies the constant component of the regressor vector. Finally, observing the regressor gains and the baseline gain, we point out that they remain bounded during the simulation.

C. Degraded control condition

The benefits of the adaptive control arises when the behavior of the regulated system deviates from the predicted one. In this section, we will simulate a degradation of the actuators behaviour in order to prove the adaptive augmentation effectiveness. This degradation is simulated via the diagonal matrix $\Lambda \in \mathbb{R}^{4\times4}$ introduced in the Simulink model. It consists in an ensemble of constant gains, each

corresponding to a control variable: $\lambda_{elev_{right}}$, $\lambda_{elev_{left}}$, λ_{rud}, λ_{thrust}. In this simulation, we will impose a loss of 30% on elevators effectiveness, thus reducing the maneuverability of the aircraft. The gains are chosen as $\lambda_{elev_{right}} = 0.7$, $\lambda_{elev_{left}} = 0.7$, $\lambda_{rud} = 1$ and $\lambda_{thrust} = 1$. The advantage of

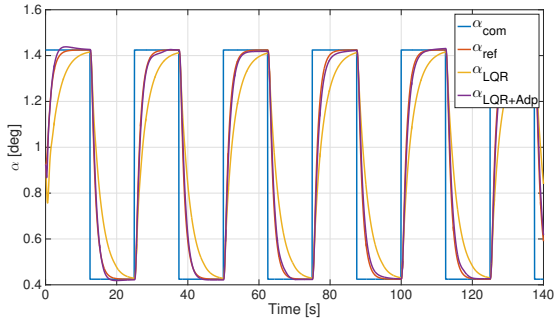

Fig. 6. Square input, $\alpha_{trim} \pm 0.5\,^{\circ}$, controls degradation.

using an adaptive augmentation is clearly demonstrated in Fig. 6, where the baseline controller is not able to reach the commanded value of the angle-of-attack while the adaptive augmented controller provides a good tracking of the square wave. It is important to point out that the capacity of the adaptive controller in tracking the reference model has also to be attributed to the modulation of the state tracking error as introduced in (21). The evolution of adaptive gains

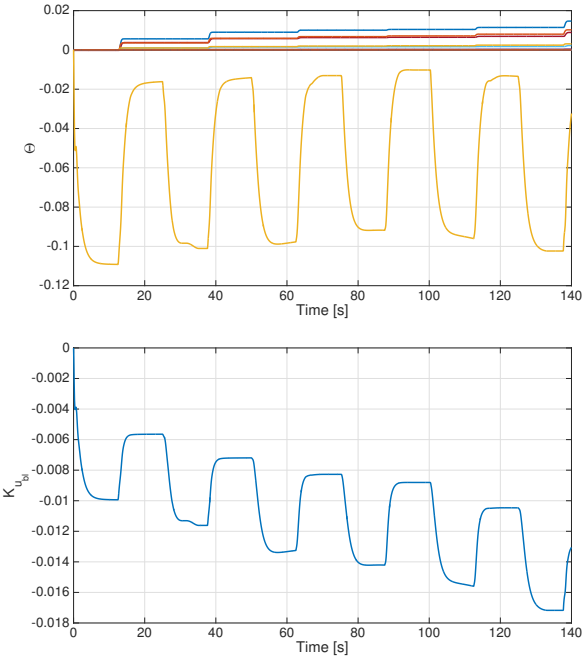

Fig. 7. Adaptive gains evolution through simulation, controls degradation.

shows that they remain bounded in time (see Fig. 7). Both the graphs clearly demonstrate the continuous work of the MDZM visible in the constant sections of the gains evolution. In fact, when the regulated system is sufficiently close to the reference model (in terms of the modulated components of the state tracking error), the adaptation process is stopped and the gains frozen until the state error rises again.

D. Aerodynamic coefficient uncertainties

The hypersonic flight is characterized by some very complex phenomena that make it very difficult to calculate with a sufficient precision the aerodynamic coefficients using CFD tools. In addition, the wind tunnel experiments, performed to measure the unknown quantities, are very expensive to set up and in any case not sufficient to reproduce the exact conditions encountered during the flight of an HSV. Thus it is crucial to design a controller able to guide the system, even when it is affected by important discrepancies with respect to the expected conditions.
We will now test both the baseline and the adaptive controllers in the presence of altered aerodynamic coefficients, by assuming that the pitching coefficients are now defined by $C_{m_\alpha} = -1.5C^{bl}_{m_\alpha}$ and $C_{m_q} = 0.2C^{bl}_{m_q}$, where the apex bl stands for the nominal quantity. The first one is a strong destabilizing coefficient, while the second one reduces the damping of the aircraft rotation around the y-axis. Simulation

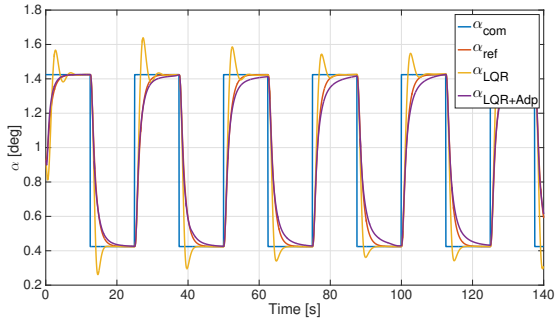

Fig. 8. Square input, $\alpha_{trim} \pm 0.5\,^{\circ}$, aerodynamic uncertainties.

results are shown in Fig. 8. The adaptive controller is able to track the command slightly losing in rapidity. On the other hand, the baseline controller produces many overshoots that are not admissible for a HSV. Concerning gains, one can note in Fig. 9 a more important activation of the neural network and the boundedness of all signals.

E. Nonlinear pitching moment

Up to now we have tested the effects of constant aerodynamic uncertainties on the baseline and on the augmented controller. In this section, we will consider that the pitching moment evolves nonlinearly, as it has been proposed by Lavretsky in [7]:

$$C_{m_\alpha}(\alpha) = -1.5 \cdot C^{bl}_{m_\alpha} + e^{-\frac{\left(\alpha - \frac{\pi}{180}\right)^2}{0.0116^2}} \cdot C^{bl}_{m_\alpha} \qquad (28)$$

where $C^{bl}_{m_\alpha}$ is the nominal aerodynamic coefficient and -1.5 is a coefficient added to produce instability. The exponential function adds a variable, nonlinear quantity centered in $\alpha_{centre} = 1^{\circ}$ to throw in the disturbance in the middle of the simulated range of angles-of-attack.

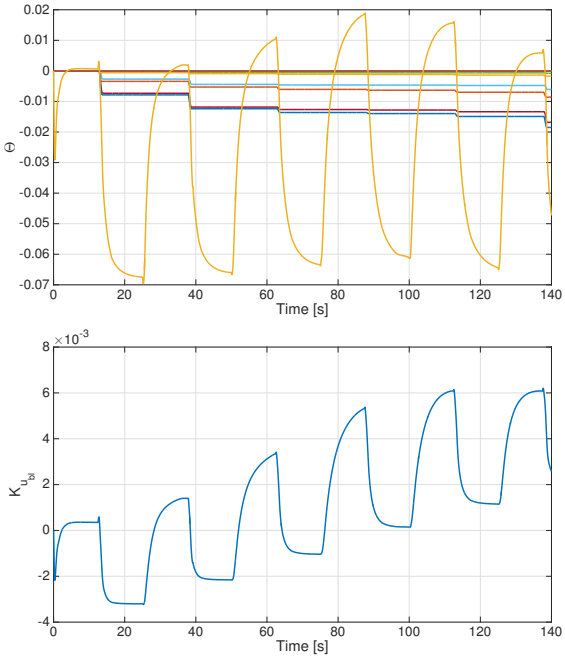

Fig. 9. Adaptive gains evolution, aerodynamic uncertainties.

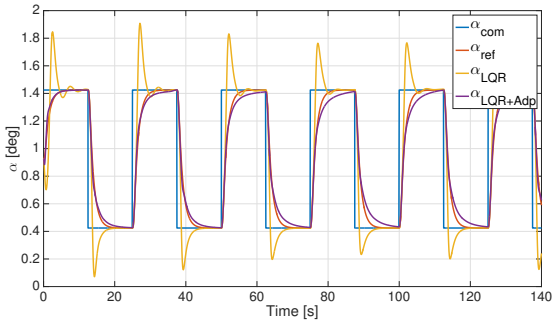

Fig. 10. Square input, $\alpha_{trim} \pm 0.5\,^{\circ}$, nonlinear pitching moment.

The results in Fig. 10 are very encouraging. The baseline controller is really stressed by the nonlinear aerodynamic coefficient and produces huge overshoots. On the contrary, the augmented controller remains close to the reference model, thus ensuring a good tracking. The evolutions of gains (Fig. 11), are very similar to those observed in the previously, when an unstable but constant C_{m_α} was proposed. This reaffirms once again the relation between the uncertainties introduced in the system and the gains trend.

VI. CONCLUSION

This work aimed at designing an adaptive augmentation of a baseline controller for the longitudinal dynamics of a hypersonic vehicle using MRAC. Due to the complexity and the vulnerability of the adaptation process, two modifications have been added to the controller: a modified dead-zone to stop the adaptation process and freeze the gains, and a projection operator allowing to bound the whole adaptation process. In order to obtain better performances in terms

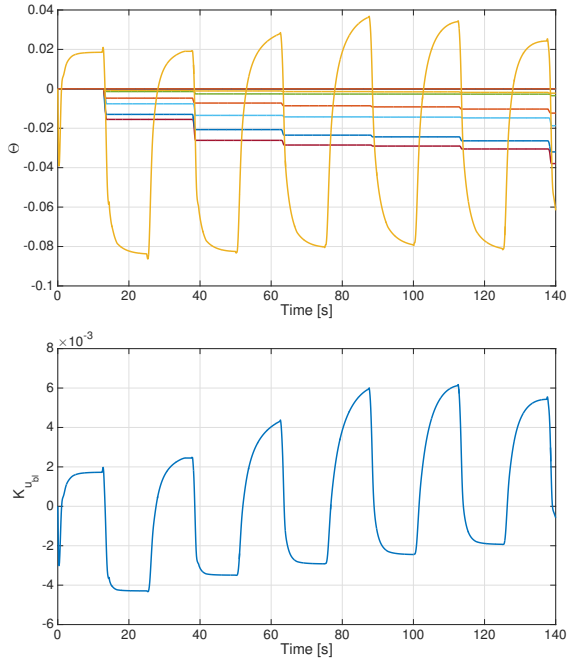

Fig. 11. Adaptive gains evolution, nonlinear pitching moment.

of reference model tracking, two additional modifications have been proposed by the authors. These modifications are dedicated to modulate the state tracking error components and to boost the dynamic of the adaptive gains. Robustness and performances of the newly-designed adaptive controller have been assessed by performing numerical simulations in nominal conditions, degraded control conditions (with a loss of 30% of elevator effectiveness), and also by considering aerodynamic uncertainties and non-linearities on pitching moment coefficients. Simulation results have shown that the augmented adaptation process provided better performances than the baseline controller for the considered cases.
In this first work, the modifications proposed by the authors have been proved effective in a qualitative way; currently, other simulations are in progress to quantify the performance enhancement deriving from them.

REFERENCES

[1] E. Lavretsky and K. Wise, Robust and Adaptive Control - With Aerospace Applications. Springer-Verlag, London, 2013.
[2] D.P. Wiese, A.M. Annaswamy, J.A. Muse and M.A. Bolender, Adaptive Control of a Generic Hypersonic Vehicle. Proc. of the AIAA Guidance, Navigation and Control Conference, Boston, MA, August 2013.
[3] A.M. Annaswamy, Adaptive Control of the Generic Hypersonic Vehicle. MIT Technical Report, 2012.
[4] J.-J.E. Slotine and J.A. Coetsee, Adaptive Sliding Controller Synthesis for Non-linear Systems. Int. J. Control, 43(4), 1986.
[5] J.G. Kreisselmeier and K. Narendra, Stable model reference adaptive control in the presence of bounded disturbances. IEEE Transactions on Automatic Control, vol. 27, n°6, pp.1169-1175, December 1982.
[6] J.D. Shaughnessy, S.Z. Pinckney, J.D. McMinn, C.I. Cruz and M.-L. Kelley, Hypersonic Vehicle Simulation Model: Winged-Cone Configuration. NASA Technical Memorandum n°102610, Langley Research Center, 1990.
[7] E. Lavretsky, Adaptive Control: Introduction, Overview and Applications. IEEE Robust and Adaptive Control Workshop, 2008.