Semi-automated observer-based controller design

F. O. Ramos[1,2], **fausto@iae.cta.br**
D. Alazard[1], **daniel.alazard@isae.fr**
[1] **Institut Supérieur de l'Aéronautique et de l'Espace (ISAE)**, 10 av. Édouard Belin, 31055 Toulouse CEDEX, France
[2] **Instituto Nacional de Pesquisas Espaciais (INPE)**, Av. dos astronautas, 1758, 12227-010 São José dos Campos SP Brazil

Abstract. *Controller synthesis for aerospace applications is typically complex, also said multi-objective. It is not unusual to combine different design techniques, where each one must be tuned appropriately in order to comply with the required specifications. Furthermore, if one considers gain scheduling, then the same set of procedures must be repeated for each single operating point, with additional constraints on controller interpolation. The resulting complexity can even increase once that robustness and fault tolerance are mostly necessary, which confirms the challenging nature of the entire conception and development of such systems. By the other side, the combination of observer-based approaches with intelligent computation provides some answers to this difficult task, as it will be shown in this work. Firstly, a robust controller is automatically designed by computational intelligence, with a genetic algorithm that searches a parameter space of considerable dimensions, according to the ratings given by a fuzzy system, where the specifications are stored. Subsequently, the controller found undergoes an internal reorganization following its observer-based form, where the state variables are meaningful (= physical) estimates. A faulty operation in a launcher attitude control system demonstrates the appeal of the proposed techniques.*

Keywords: *Observer-based, controller, computational intelligence, design*

1. INTRODUCTION

For aerospace applications, control system design is not only a matter of satisfying important requirements such as stability, performance, robustness to parameter variations and external disturbances, and so on, but there are also practical issues (as on-board implementation) which draw the attention of the control engineer; it would not be surprising if complexity, flexibility, and memory storage could influence and even decide the choice between rivalling structures. In such aspect, linear quadratic or PID controllers, which rely on single sets of scalar gains, would be preferable to H_∞ controllers, the latter ones typically possessing the same order than the plant model used for design (normally a simplified version of a even more complex validation model). By the other side, one may argue if and what additional features and benefits can be uncovered when using these larger realizations. Fortunately, it can be shown that any controller has an observer-based realization, as demonstrated by the deterministic separation principle (Schumacher, 1980). By that principle, controller states can be made to correspond to the plant states; in other words, the controller is also an observer and provides the estimates of the plant state vector, and maybe other desired estimates as external disturbances and faults. A new technique (Alazard and Apkarian, 1999) allows to redesign the controller into its observer-based form, but at first one must have one. However, the primary synthesis of a controller for aerospace applications is typically complex, also said multi-objective. Despite the success of deterministic mathematical reasoning behind the conventional control theory which guarantees stability and performance bounds, Computational Intelligence (CI) has been presented as an important complement in modern control systems (Bars et al., 2006).

In this work, one adapts a previous implementation on linear-quadratic controller CI-based design (Ramos and Araujo, 2008) also to H_∞ techniques. At that time, a combination of pre-existent conventional procedures for launcher attitude control system design with genetic algorithms provided optimality for all linearisation points, as well as gain vector interpolation, which was validated through hardware-in-the-loop simulations. Posteriorly, a fuzzy system replaced the original cost function mapping comprising the control specifications and the gain vector interpolation indexes. Now, one intends to produce observer-based structures from H_∞ CI-designed controllers, yielding estimates of plant states, faults and disturbances. This work is organized as follows: (i) section 2 presents briefly the new technique associated with the observer-based form (OBF); (ii) section 3 also briefly introduces and describes the general H_∞ standard problem adopted for the VLS launcher attitude control system and the CI mechanism used to tune the various weights during the design; (iii) the last section considers fault scenarios and respective OBF implementation.

2. OBSERVER-BASED REDESIGN

"Almost any system is an observer", wrote Luenberger in its introductory article (Luenberger, 1971) about these special structures, used during decades not only for full state control but also in disturbance estimation and fault tolerance; in fact, even controllers can be viewed as observers. In this section, one presents the resumed procedure (Alazard and Apkarian, 1999) to compute the observer-based realization (that is: the state feedback gain $\mathbf{K_c}$, the state estimator gain $\mathbf{K_f}$ and the YOULA parameter $\mathbf{Q}$). The general block diagram of the closed-loop system involving an observer-based controller is

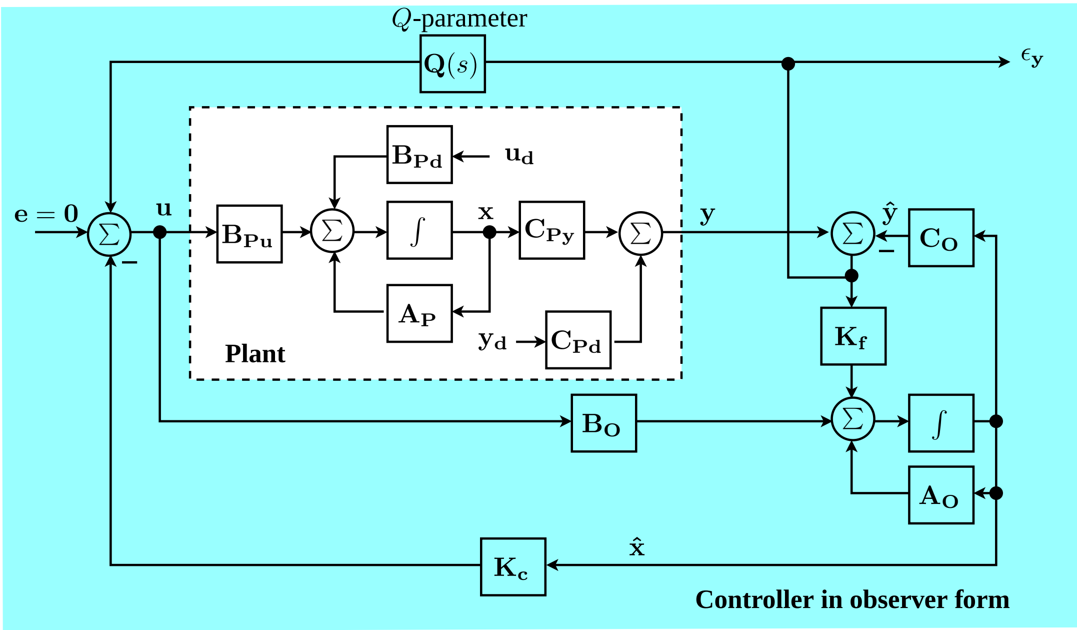

Figure 1. Observer-based structure using YOULA parametrization.

shown in the Fig. 1.

Remark : at first, the inputs $\mathbf{u_d}$ (representing external disturbances and actuator misalignments or faults) and $\mathbf{y_d}$ (representing sensor bias or faults) seen in the Fig. 1 are not considered, so that $\mathbf{B_P} = \mathbf{B_{Pu}}$ and $\mathbf{C_P} = \mathbf{C_{Py}}$. Consider the stabilizable and detectable n^{th} order plant model $\mathbf{G}(s)$ (m inputs and p outputs) with state-space realization (1(a)) and the respective stabilizing n_K^{th} order controller $\mathbf{K}(s)$ with minimal state-space realization (1(b)) :

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A_P} & \mathbf{B_P} \\ \mathbf{C_P} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix} (a), \quad \begin{bmatrix} \dot{\mathbf{x}}_{\mathbf{K}} \\ \mathbf{u} \end{bmatrix} \begin{bmatrix} \mathbf{A_K} & \mathbf{B_K} \\ \mathbf{C_K} & \mathbf{D_K} \end{bmatrix} \begin{bmatrix} \mathbf{x_K} \\ \mathbf{y} \end{bmatrix} (b). \tag{1}$$

The key idea is to express the controller as an LUENBERGER observer with a state vector $\mathbf{z} = \mathbf{T}\mathbf{x}$ and thus, we will denote $\mathbf{x_K} = \hat{\mathbf{z}} = \widehat{\mathbf{T}\mathbf{x}} = \mathbf{T}\hat{\mathbf{x}}$. It can be shown (Alazard and Apkarian, 1999) that $\mathbf{T}$ is the solution of a generalized non-symmetric RICCATI Eq. (2):

$$[-\mathbf{T}\ \mathbf{I}] \overbrace{\begin{bmatrix} \mathbf{A_P} + \mathbf{B_P}\mathbf{D_K}\mathbf{C_P} & \mathbf{B_P}\mathbf{C_K} \\ \mathbf{B_K}\mathbf{C_P} & \mathbf{A_K} \end{bmatrix}}^{\mathbf{A_{cl}}} \begin{bmatrix} \mathbf{I} \\ \mathbf{T} \end{bmatrix} = \mathbf{0}. \tag{2}$$

The characteristic matrix $\mathbf{A_{cl}}$ associated with the RICCATI Eq. (2) is nothing else than the closed-loop (c.-l.) dynamic matrix built on the state vector $[\mathbf{x}^T\ \mathbf{x_K}^T]^T$. Such a RICCATI equation can then be solved in $\mathbf{T} \in \mathbb{R}^{n_k \times n}$ by standard subspace decomposition techniques, that is :

- compute an invariant subspace associated with the set of n eigenvalues spec($\mathbf{\Gamma_n}$), chosen among $n + n_K$ eigenvalues in spec($\mathbf{A_{cl}}$), that is, $\mathbf{A_{cl}} \left[\mathbf{U_1}^T\ \mathbf{U_2}^T\right]^T = \left[\mathbf{U_1}^T\ \mathbf{U_2}^T\right]^T \mathbf{\Gamma_n}$, where $\mathbf{U_1} \in \mathbb{R}^{n \times n}$ and $\mathbf{U_2} \in \mathbb{R}^{n_K \times n}$. Such subspaces are easily computed using SCHUR decompositions of $\mathbf{A_{cl}}$.
- compute the solution $\mathbf{T} = \mathbf{U_2}\ \mathbf{U_1}^{-1}$.

Then, 3 cases can be encountered :

- *Full-order controller ($n_K = n$) :* one can compute a state feedback gain $\mathbf{K_c} = -\mathbf{C_K}\ \mathbf{T} - \mathbf{D_K}\ \mathbf{C_P}$, a state estimation gain $\mathbf{K_f} = \mathbf{T}^{-1}\mathbf{B_K} - \mathbf{B_P}\ \mathbf{D_K}$ and a static YOULA parameter $\mathbf{Q}(\mathrm{s}) = \mathbf{D_K}$ such that the observer-based structure fitted with the YOULA parameter (depicted in the Fig. 1) is equivalent to the initial controller form according its input-output behaviour.
- *Augmented-order controller ($n_K > n$) :* the YOULA parameter becomes a dynamic transfer of order $n - n_K$.
- *Reduced-order controller ($n_K < n$) :* in this case, the LQG structure shown in the Fig. 1 is no longer valid. However, if $n_K \geq n - p$ (p stands for the number of plant measurements), one can built a reduced-order estimator with a static YOULA parameter, involving an estimate $\hat{\mathbf{x}} = \mathbf{H_1}\hat{\mathbf{z}} + \mathbf{H_2}\ \mathbf{y}$ by a linear function of the controller state $\hat{\mathbf{z}}$ and the plant output $\mathbf{y}$, with the constraint $\mathbf{H_1}\ \mathbf{T} + \mathbf{H_2}\ \mathbf{C_P} = \mathbf{I_n}$. Otherwise, if $n_K < n - p$, a model reduction is required to built a (partial) state-observer realization.

The separation principle of the observer based realization allows to state that :

- the c.-l. eigenvalues can be separated into c.-l. state-feedback poles (spec($\mathbf{A}-\mathbf{BK_c}$)), c.-l. state-estimator poles (spec($\mathbf{A_P}-\mathbf{K_f C_P}$)) and the YOULA parameter poles (spec($\mathbf{A_Q}$)),
- the c.-l. state-estimator poles and the YOULA parameter poles are uncontrollable by $\mathbf{e}$,
- the c.-l. state-feedback poles and the YOULA parameter poles are unobservable from $\varepsilon_\mathbf{y}$. The transfer function from $\mathbf{e}$ to $\varepsilon_\mathbf{y}$ always vanishes.

Note that there is a combinatoric set of solutions according to the choice of n auto-conjugate eigenvalues among $n+n_K$ c.-l. eigenvalues. The range of solutions can be reduced according to the following considerations :

- a set of auto-conjugated eigenvalues must be chosen in order to find a real parametrization,
- an uncontrollable (resp. unobservable) eigenvalue in the system must be selected in the state-feedback dynamics (resp. state-estimation dynamics),
- lastly, the state-estimation dynamics (spec($\mathbf{A_P}-\mathbf{K_f C_P}$)) is usually chosen faster than the state-feedback dynamics (spec($\mathbf{A_P}-\mathbf{B_P K_c}$)).

Finally, the observer matrices $\mathbf{A_O}, \mathbf{B_O}$ and $\mathbf{C_O}$ correspond to the state-space matrices of the on-board plant model $\mathbf{G_O}$ when $n_K \geq n$. The on-board model is required in the computation of the observer form and is built upon the original plant model where other variables of interest (such as disturbances and faults, represented by the variables $\mathbf{u_d}$ and $\mathbf{y_d}$ in the Fig. 1) may also be included as states to be estimated. For example, on considering the original plant model (1(a)) and a bias term b associated with a single output, the correspondent on-board model could be given by the Eq. (3).

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{b} \\ \hline y \end{bmatrix} = \left[\begin{array}{cc|c} \mathbf{A_P} & \mathbf{0} & \mathbf{B_P} \\ \mathbf{0} & 0 & \mathbf{0} \\ \hline \mathbf{C_P} & 1 & \mathbf{0} \end{array}\right] \begin{bmatrix} \mathbf{x} \\ b \\ \hline \mathbf{u} \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{A_O} & \mathbf{B_O} \\ \hline \mathbf{C_O} & \mathbf{0} \end{array}\right] \begin{bmatrix} \mathbf{x} \\ b \\ \hline \mathbf{u} \end{bmatrix} \tag{3}$$

3. H_∞ CI-BASED DESIGN

Launcher models. The full pitch plane decoupled model $\mathbf{G_L}$ of the Brazilian launcher VLS (Filho and Carrijo, 1999) will be chosen to illustrate the design procedure. The generalized model used for the H_∞ technique is depicted in the Fig. 2. The following transfer functions are considered :

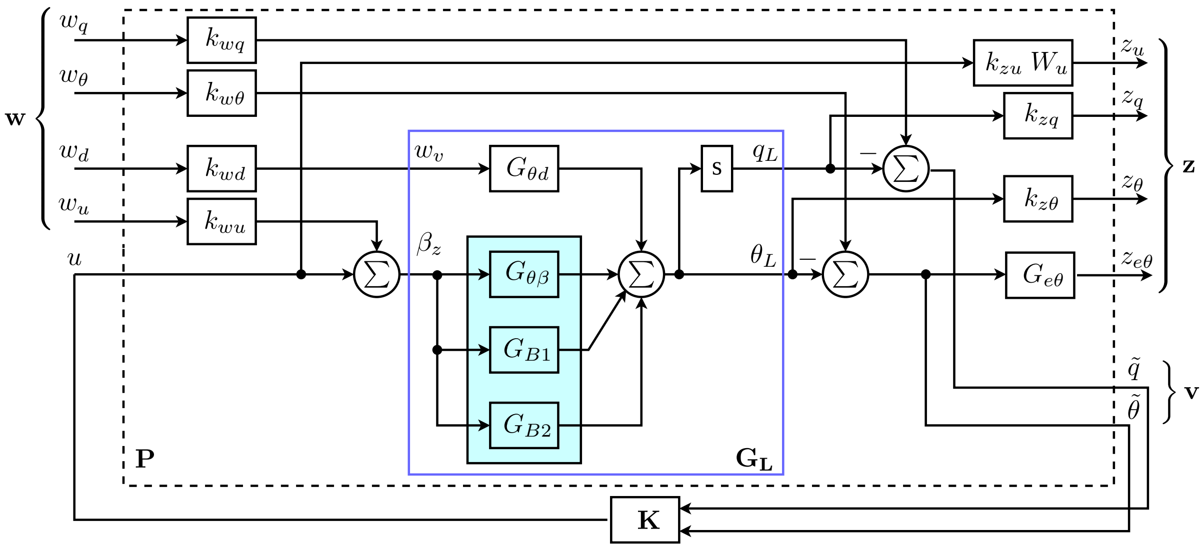

Figure 2. Generalized standard control problem for the VLS launcher.

- $G_{\theta\beta}$ and $G_{\theta d}$ are the transfer functions of the linear rigid body decoupled model from control inputs β_z (control action) and w_v (wind disturbance) to the output θ_L (attitude angle).
- G_{B1} and G_{B2} are the transfer functions of the 1^{st} and 2^{nd} bending modes.
- $G_{e\theta}$ is the transfer function representing the (approximated) integral of the error signal $k_{w\theta}\, w_\theta - \theta$. This transfer function is required to reduce the steady-state error to a step function at input w_θ (or otherwise reference input θ_{ref}). The parameter $\epsilon_{e\theta}$ is necessary to comply with the properties of the standard control problem (H_∞ design).
- W_u is the weight on the control signal u.

The design procedure. The combination of the H_∞ technique and computational intelligence is illustrated by the Fig. 3, where the genetic algorithm (GA) is the sole responsible by the generation, combination, mutation and selection of the candidates[1] used in the controller design, according to the engineering requirements stored in a fuzzy system. Some of the main characteristics of the GA employed in the CI-based design mechanism are :

- Each gene is a binary number in the form 2^n, where n is the number of bits.
- Each weight $k_{\bullet\bullet}$ used in the H_∞ standard problem depicted in the Fig. 2 is composed of two genes in the form g_1/g_2 producing a numeric interval from $1/2^n$ to $2^n/1$. An entire set of weightings is called an individual, which in this case is composed of 14 genes, resulting in a huge search space order of 10^{33} for $n = 8$ bits.
- The roulette wheel is used for the reproduction of the individuals.
- Each run is finished by a stop criterion (standard deviation of the last n ratings).
- A record of every individual is kept in order to avoid wasted time in repeated evaluations.
- The fitness function is a fuzzy system.

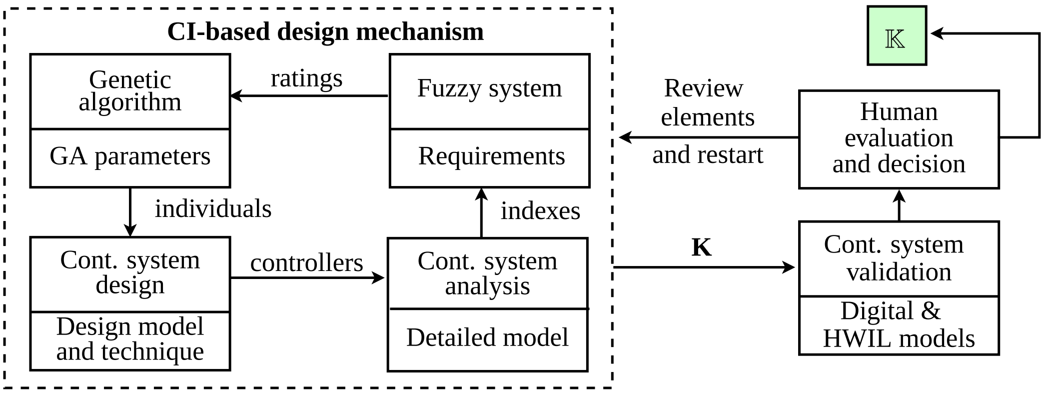

Figure 3. Block diagram of the CI-based design mechanism.

The fuzzy system is composed of linguistic variables, fuzzy sentences and fuzzy rules. The fuzzy sentences adopted in this work are based on mathematical expressions such as Gaussian or polynomial functions, with engineering specifications as linguistic input variables (rise time - t_r, settling time - t_s, overshoot - M_p, maximum amplitude of the control signal - u_{max}, gain margin - m_g, phase margin - m_p and dynamics of the closed-loop poles - p_{cl}). The linguistic output variable is "Rating" (the global rating). Each linguistic variable comprises the respective fuzzy sentences and an universe of discourse. An hypothetical example according to the specification "gain margin" would be : "The linguistic variable m_g is associated with the control system gain margin, where its universe of discourse is [0, 20] [dB]. The fuzzy sentence {*Unsatisfactory* m_g} is defined by a z-polynomial function (Eq. (4)) and the pair $\langle a, b\rangle$, with $a = 0$ and $b = 6$."

$$f(x) = \begin{cases} 1, x \le a \\ 1 - 2\left[(x-a)/(b-a)\right]^2, a < x \le (a+b)/2 \\ 2\left[b - x/(b-a)\right]^2, (a+b)/2 < x \le b \\ 0, x > b \end{cases} \tag{4}$$

4. COMPLETE CONTROLLER DESIGN

4.1 CI-based controller design

The fuzzy sentences adopted in this work represent the following engineering specifications: rise time - t_r, settling time - t_s, overshoot - M_p, maximum amplitude of the control signal - u_{max}, gain margin - m_g, phase margin - m_p and dynamics of the closed-loop poles - p_{cl}. The fuzzy system rules are given by the Eq. (5).
Remark : a further implicit specification is given by the initial upper bound on the cost γ used in the H_∞ design, associated with system robustness.

$E \triangleq$ ("t_r is Satisfactory") and ("t_s is not Large") and ("u_{max} is Satisfactory")
and ("m_g is not Unsatisfactory") and ("m_p is not Unsatisfactory") and ("p_{cl} is Slow")
R_1 : If E and ("M_p is Satisfactory") then ("Rating is Good") (5)
R_2 : if E and ("M_p is not Satisfactory") then ("Rating is Regular")
R_3 : If not E then ("Rating is Bad")

[1] Due to the text limitations, the reader is asked to refer to the existing literature (e.g., (Fleming and Purshouse, 2002)) on the definition of each term used in this section.

4.2 Observer-based redesign

In this work, the on-board model $\mathbf{G_O}$ (Eq. (6)) is built from the balanced realization $\mathbf{G_L}$ (Fig. 2) added to the estimates $\hat{b}_q$ (output bias on q_L) and $\hat{w}_v$ (formerly disturbance input w_v). It follows that $\mathbf{G_O}$ has one state more than $\mathbf{K}$, the condition “reduced-order controller ($n_K > n - p$)” stated at the section 2 is applied, and two matrices $\mathbf{H_1}$ and $\mathbf{H_2}$ must be calculated (see Alazard and Apkarian (1999)).

$$\left[\begin{array}{c} \dot{\hat{\mathbf{x}}}_{\mathbf{L}} \\ \dot{\hat{w}}_v \\ \dot{\hat{b}}_q \\ \hline \tilde{q} \\ \tilde{\theta} \end{array}\right] = \left[\begin{array}{cccc|c} \mathbf{A_P} & \mathbf{B_{Pd}} & \mathbf{0} & \mathbf{B_{Pu}} \\ \mathbf{0} & \lambda_\alpha & 0 & 0 \\ \mathbf{0} & 0 & \lambda_q & 0 \\ \hline \mathbf{C_{Pq}} & D_{Pdq} & 1 & D_{Puq} \\ \mathbf{C_{P\theta}} & D_{Pd\theta} & 0 & D_{Pu\theta} \end{array}\right] \left[\begin{array}{c} \hat{x_L} \\ \hat{w}_v \\ \hat{b}_q \\ \hline \beta_z \end{array}\right] \tag{6}$$

Remark: The steady state of the variable w_v cannot be observed, according to the transfer function $G_{\theta d}$ (there is a zero at $s = 0$), but it can be replaced by the estimation of the attack angle α as given by the expression $(\hat{w} - \hat{w}_v)\ \bar{U}^{-1}$, where $\bar{U}$ is the velocity component in the vehicle body axis X_b (longitudinal axis).

Choice of the closed-loop poles. Once that $\mathbf{K}$ is a reduced-order controller, the YOULA parameter is static, and no pole is assigned to it. Therefore, only the controller and the observer share the poles, and the two uncontrollable ones (n. 16 and 17 in the table 1) are allocated to the state-feedback dynamics, and also the 7 slowest poles of the remaining set, forming the option “A” in the table 1; option “B” results from the exchange of one of the slowest poles (no. 15) with a faster one (pole n. 11). The reason for defining these two options will be clarified later. A further point related to the closed-loop poles is associated with their natural frequencies : sets with faster poles most probably imply noisier estimates; that was the reason to add the design specification p_{cl} to the fuzzy system (see the section 3), which gives better ratings to candidates with more compressed sets of poles near the origin of the complex plane.

Table 1. Closed-loop distribution, options “A” and “B”. *(UC = uncontrollable.)*

Closed-loop poles no.	Value	Option “A”	Option “B”
1,2	$-1.3887 \pm 80.4553\ i$	$\mathbf{K_c}$	$\mathbf{K_c}$
3,4	$-3.2979 \pm 80.5979\ i$	K_f	K_f
5,6	$-2.3452 \pm 29.6751\ i$	$\mathbf{K_c}$	$\mathbf{K_c}$
7,8	$-4.1625 \pm 29.6290\ i$	K_f	K_f
9,10	$-5.3201 \pm 4.3009\ i$	K_f	K_f
11	-4.6420	K_f	$\mathbf{K_c}$
12	-3.4123	K_f	K_f
13	-0.0062	$\mathbf{K_c}$	$\mathbf{K_c}$
14	-0.0919	$\mathbf{K_c}$	$\mathbf{K_c}$
15	-0.8594	$\mathbf{K_c}$	K_f
16 (UC)	-1.0000	$\mathbf{K_c}$	$\mathbf{K_c}$
17 (UC)	0.0000	$\mathbf{K_c}$	$\mathbf{K_c}$

4.3 Evaluation of the complete design

The validation model used in the simulations includes the actuator dynamics, a realistic wind profile, noise sources and a bias profile applied to one of the plant outputs. The estimates were produced with the expressions $\hat{\alpha} = \mathbf{H_{\alpha 1}}\ \hat{z} + \mathbf{H_{\alpha 2}}\ y$ and $\hat{b}_q = \mathbf{H_{q1}}\ \hat{z} + \mathbf{H_{q2}}\ y$. There is a reason for using independent matrices $\mathbf{H_{\alpha i}}$ and $\mathbf{H_{qi}}$: during the simulations, it was noted that the option “A” is beneficial to the estimate $\hat{b}_q$ but not to $\hat{\alpha}$ regarding noise levels. By the other side, the effect of option “B” is opposite. However, on doing the redesign for each option and then composing the matrices $\mathbf{H_1}$ and $\mathbf{H_2}$ respectively for each estimate, it was possible to profit better noise levels as shown in the Fig. 4, where a disturbance signal (wind gust profile) and a bias level on the q_L output (combined with noise sources added to both outputs) were applied simultaneously into the system. The estimate $\hat{b}_q$ could be used in fault detection and isolation (bias fault). The abrupt variation of the bias b_q at 10 seconds yields a small and temporary deterioration of the estimate $\hat{\alpha}$. Finally, the estimate $\hat{\theta}$ is not only insensitive to that variation, but is also very close to the real attitude angle θ.

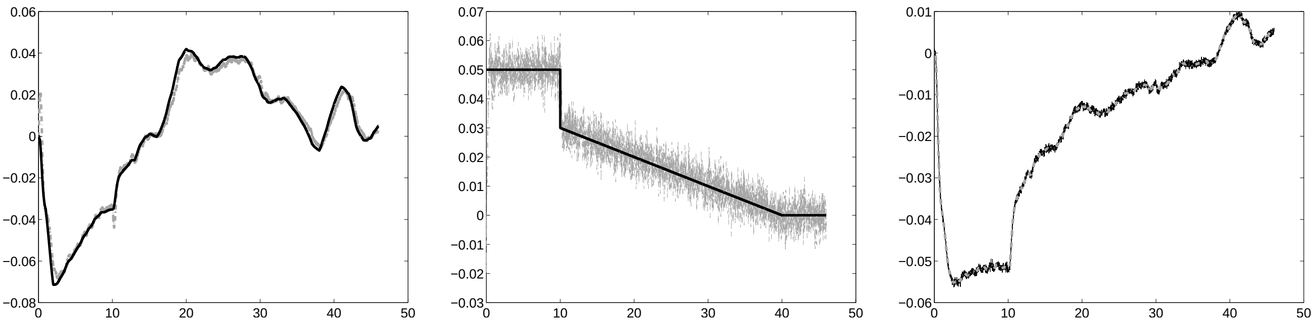

Figure 4. *(Left)* Estimation of the attack angle $\hat{\alpha}$. *(Middle)* Estimation of the bias $\hat{b}_q$ at the q_L output. *(Right)* Estimation of the output $\hat{\theta}$. *All sub-figures: grey lines - estimates, black lines - real variables; noise sources added to both plant outputs, with simultaneous occurrences of external disturbance and output bias (abrupt variation at 10 seconds).*

5. CONCLUSION

As it was shown in this work the controller structure can be employed not only in the control action but also to provide estimates of the plant state variables and other relevant signals, as faults acting on the system. The procedure demonstrated here relies on a CI-based mechanism combined with an H_∞ design technique with further observer-based redesign, and one intends to expand that mechanism to find the best combinatoric of the c.l.-poles as well. Non-linear and hardware-in-the-loop simulations are also previewed in the future work, and the same strategy Ramos and Filho (2007) that provided linear-quadratic gain scheduled controllers will be employed, that is, to include a specification in the fuzzy system taking into account the smoothing of a particular characteristic of the controller (for instance: gains $\mathbf{K_c}$ and $\mathbf{K_f}$).

6. REFERENCES

Alazard, D. and Apkarian, P. (1999). Exact observer-based structures for arbitrary compensators. *International Journal of Robust and Non-linear control*, (9):101–118.

Bars, R., Colaneri, P., de Souza, C. E., Dugard, L., Allgöwer, F., Kleimenov, A., and Scherer, C. (2006). Theory, algorithms and technology in the design of control systems. *Annual Reviews in Control*, 30:19–30.

Filho, W. C. L. and Carrijo, D. S. (1999). Control system of Brazilian launcher. In *Proc.* 4^{th} *ESA International Conference on Spacecraft Guidance, Navigation and Control Systems*.

Fleming, P. J. and Purshouse, R. C. (2002). Evolutionary algorithms in control systems engineering: a survey. *Control Engineering Practice*, 10:1223–1241.

Luenberger, D. G. (1971). An introduction to observers. *IEEE Transactions on Automatic Control*, AC-16(6):596–602.

Ramos, F. O. and Araujo, E. (2008). Fuzzy-scored genetically designed controller for the VLS-1 launcher. In *Proc. IEEE World Congress on Computational Intelligence*.

Ramos, F. O. and Filho, W. C. L. (2007). Design upgrade and validation of a launcher vehicle attitude controller combining genetic and linear quadratic optimization. In *Proc.* 19^{th} *International Congress of Mechanical Engineering*.

Schumacher, J. (1980). Compensator synthesis using (c,a,b)-pairs. *Automatic Control, IEEE Transactions on*, 25(6):1133–1138.

7. RESPONSIBILITY NOTICE

The author(s) is (are) the only responsible for the printed material included in this paper