From Standard to Structured Robust Control Design: Application to Aircraft Automatic Glide-slope Approach

P. Simplício * **D. Navarro-Tapia** * **A. Iannelli** * **A. Marcos** *

* *University of Bristol, BS8 1TR, United Kingdom (e-mail:*
pedro.simplicio/diego.navarro-tapia/andrea.iannelli/andres.marcos
@bristol.ac.uk; website: www.tasc-group.com)

Abstract: The newly-developed structured $\mathcal{H}_\infty$ paradigm provides a powerful and versatile tool for multi-variable/multi-requirement control synthesis, with successful applications growing rapidly. However, due to its versatility, insights arising naturally from the standard (i.e., unstructured) $\mathcal{H}_\infty$ framework as well as other design choices are easily overlooked. Hence, this paper shows how this type of information can be exploited to have better systematic understanding and control solutions. As application scenario, the design and analysis of a robust glide-slope approach controller is addressed. This controller is part of a complete automatic landing system developed under the scope of an aircraft landing challenge proposed by ONERA and AIRBUS.

Keywords: Structured control design, robust control analysis, aircraft landing challenge

1. INTRODUCTION

Although $\mathcal{H}_\infty$ theory provides a powerful framework for multi-variable robust control synthesis and analysis, it has a few practical limitations that need to be always carefully addressed. Most importantly, $\mathcal{H}_\infty$ controllers are monolithic (i.e., obtained as a $n \times m$ matrix) and have the same order of the generalised plant for which they are designed. Hence, they are less suitable for embedded implementation (despite the increasing computational power) unless post-design truncation methods are used (which is challenging without loss of performance). These limitations motivated the development of a non-smooth optimiser for fixed-structure controller synthesis known as structured $\mathcal{H}_\infty$ (Apkarian and Noll, 2006) and, since 2015, it is also able to account for parametric plant uncertainties (Apkarian et al., 2015). It also allows to decouple and handle multiple requirements, models and control channels.

Successful applications of structured robust control are rapidly growing. Notable developments include the design of a self-scheduled flight controller through its multi-model capability (Lhachemi et al., 2014) and the refinement of Rosetta's orbit controller, which was uploaded to the spacecraft before its final insertion manoeuvre with the target comet in May 2014 (Falcoz et al., 2015). However, due to the facilitated handling of design objectives, recent applications of structured $\mathcal{H}_\infty$ often overlook the employment of the frequency-domain shaping functions characteristic of the standard (i.e., full-order) $\mathcal{H}_\infty$ framework and their relationship with closed-loop properties is thus underrated. In addition, being a non-smooth optimiser, structured $\mathcal{H}_\infty$ is sensitive to the initialisation of the algorithm and to the choice of parameters that are free to be tuned. These issues may influence critically the solution found by the algorithm, yet little attention is typically devoted to them and their impact is often mitigated via multiple runs initialised with random conditions.

The main objective of this paper is therefore to emphasise how insights from standard $\mathcal{H}_\infty$ control and intuitive choices of initial conditions may guide the control engineer towards improved and clearer structured $\mathcal{H}_\infty$ solutions. The focus is therefore on analysing these issues and not the intrinsic sacrifice of performance between structured and full-order $\mathcal{H}_\infty$ or μ-synthesis controllers.

To do so, a sequential process will be followed for the design of a glide-slope approach controller. This controller is part of a complete autoland system developed for an aircraft landing challenge (Biannic, 2016). The design of an autoland with good flight performance and robustness against dispersions and wind disturbances is very challenging and has been extensively addressed using diverse techniques (Biannic and Apkarian, 2001; Looye and Joos, 2006; Sadat-Hoseini et al., 2013; Theis et al., 2017).

For the design process presented, the effect of parametric uncertainties is modelled via a Linear Fractional Transformation (LFT) and the robustness of successive controllers is analysed using the structured singular value μ. LFT modelling and μ analysis are linear tools directly applicable under the $\mathcal{H}_\infty$ framework. Details on these techniques can be found in Doyle et al. (1991) and references therein. The results are validated through nominal and Monte-Carlo (MC) nonlinear simulations. For further details on this type of analysis applied to a Space system, the reader is referred to Simplício et al. (2016). A secondary objective is thus to show how conclusions from nonlinear simulations are anticipated using the linear tools mentioned above and how this information is incorporated in the design process.

The paper is organised as follows: Sec. 2 summarises the aircraft landing challenge addressed, Sec. 3 specifies the objectives and model considered for glide-slope approach, Sec. 4 formulates the problem as a standard/structured

$\mathcal{H}_\infty$ optimisation and Sec. 5 proceeds with the combined design and analysis of successively refined controllers.

2. THE AIRCRAFT LANDING CHALLENGE

The aircraft landing benchmark adopted for this paper has been proposed by ONERA and AIRBUS to motivate the development of enhanced aircraft automatic landing systems. A detailed description of this open-source and freely available benchmark is provided in Biannic and Roos (2015) and Biannic (2016) and summarised in this section.

It features a model for the nonlinear six degree-of-freedom simulation of a large rigid-body aircraft in full configuration from 1000 ft above runway until touchdown. The aircraft is controlled via its engine throttle and aerodynamic actuators (elevator, ailerons and rudder), all of them with internal dynamics and saturations. As output, it provides conventional aircraft dynamics variables, together with noisy Instrument Landing System (ILS) measurements of the vertical (glide-slope) and lateral (localiser) deviations with respect to the ideal landing trajectory. The model also incorporates the ground effect on the aerodynamics and three-dimensional Dryden wind turbulence inputs. In addition, it specifies a set of requirements at touchdown, as well as the dispersions of flight parameters to be considered: longitudinal and lateral wind levels, aircraft mass, center of gravity, runway altitude, sea-level temperature, runway slope, glide-slope angle and localiser displacement.

An automatic landing is divided into two main phases: a) final approach, where the ILS glide-slope (vertical) and localiser (lateral) errors must be minimised while keeping the calibrated airspeed constant and the sideslip to zero and b) flare and decrab, activated at around 15 m to reduce vertical speed, align the aircraft with the runway axis and consequently minimise the load on the landing gears at touchdown. As this paper is focused on glide-slope approach, all the models are only simulated until flare activation. For the dedicated flare control design and analysis, the reader is referred to Navarro-Tapia et al. (2017). Equivalently, the localiser controller provided in Biannic (2016) is kept as baseline throughout the paper.

3. PROBLEM & MODEL DESCRIPTION

As mentioned in Sec. 2, the main objective of the glide-slope approach control system is to minimise the vertical ILS deviation while keeping the calibrated airspeed constant. Furthermore, this must be efficiently performed for a dispersed (off-nominal) range of flight parameters and for diverse levels of wind turbulence. The common assumption that aircraft longitudinal and lateral dynamics are sufficiently decoupled such that corresponding control channels can be tackled independently is also adopted here and proved in Iannelli et al. (2017).

In the same reference, the most dominant flight parameters of the benchmark are identified and an LFT is generated to capture their variations based on small perturbations around the trim states. These parameters include the aircraft mass δ_m, center of gravity $\delta_{x_{\mathrm{CG}}}$, runway altitude $\delta_{h_{\mathrm{rwy}}}$ and sea-level temperature δ_{T0}, which are encapsulated as the real uncertainty block:

$$\Delta = \mathrm{diag}\left(\delta_m \mathbb{I}_6, \delta_{x_{\mathrm{CG}}}\mathbb{I}_6, \delta_{h_{\mathrm{rwy}}}\mathbb{I}_6, \delta_{T0}\mathbb{I}_6\right), \; ||\Delta||_\infty \leq 1 \qquad (1)$$

As shown on the right side of Fig. 1, this uncertainty block is acting on the Linear Time-Invariant (LTI) system $G(\mathrm{s})$ with states $\mathbf{x}_{\mathrm{G}}(\mathrm{s}) = [u(\mathrm{s}); w(\mathrm{s}); q(\mathrm{s}); \theta(\mathrm{s}); \delta_{\mathrm{T}}(\mathrm{s}); \delta_{\mathrm{E}}(\mathrm{s})]$ (corresponding to longitudinal and vertical speed in body axes, pitch rate and angle, throttle and elevator state, respectively). Inputs correspond to the actuator commands $\mathbf{u}(\mathrm{s}) = [\delta_{\mathrm{Tc}}(\mathrm{s}); \delta_{\mathrm{Ec}}(\mathrm{s})]$ and longitudinal/vertical wind disturbances $\mathbf{w}_{\mathrm{d}}(\mathrm{s}) = [w_{\mathrm{x}}(\mathrm{s}); w_{\mathrm{z}}(\mathrm{s})]$. The chosen outputs correspond to the calibrated and vertical speeds to be controlled $\mathbf{v}(\mathrm{s}) = [v_{\mathrm{c}}(\mathrm{s}); v_{\mathrm{z}}(\mathrm{s})]$, together with the pitch rate and vertical load factor to be employed as feedback $\mathbf{y}_{\mathrm{f}}(\mathrm{s}) = [q(\mathrm{s}); n_{\mathrm{z}}(\mathrm{s})]$. Upper and lower case variables are used to distinguish between total and perturbed values. This system has the following state-space description:

$$\begin{bmatrix} \dot{\mathbf{x}}_G(\mathrm{s}) \\ \hline \mathbf{v}(\mathrm{s}) \\ \mathbf{y}_{\mathrm{f}}(\mathrm{s}) \end{bmatrix} = \left[\begin{array}{c|c} A_G & B_G \\ \hline C_G & D_G \end{array}\right] \begin{bmatrix} \mathbf{x}_G(\mathrm{s}) \\ \hline \mathbf{u}(\mathrm{s}) \\ \mathbf{w}_{\mathrm{d}}(\mathrm{s}) \end{bmatrix} \qquad (2)$$

Uncertainty ranges as well as closed-loop architecture are kept the same as in Biannic (2016). The latter features an outer loop controller that commands $v_{\mathrm{zc}}(\mathrm{s})$ proportional to the vertical ILS deviation and $v_{\mathrm{cc}}(\mathrm{s}) = 0$ for constant airspeed, followed by the inner loop depicted in Fig. 1.

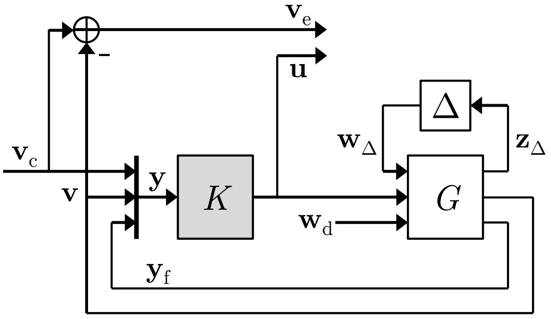

Fig. 1. Closed-loop glide-slope approach model

The glide-slope controller $K(\mathrm{s})$ must then be designed to track $\mathbf{v}_{\mathrm{c}}(\mathrm{s}) = [v_{\mathrm{cc}}(\mathrm{s}); v_{\mathrm{zc}}(\mathrm{s})]$, while ensuring flight stability and safety. It is a two degree-of-freedom tracking controller with $\mathbf{y}(\mathrm{s}) = [\mathbf{v}_{\mathrm{c}}(\mathrm{s}); \mathbf{v}(\mathrm{s}); \mathbf{y}_{\mathrm{f}}(\mathrm{s})]$ as input and internal dynamics $\dot{\mathbf{x}}_K(\mathrm{s})$ as follows:

$$\begin{bmatrix} \dot{\mathbf{x}}_K(\mathrm{s}) \\ \hline \mathbf{u}(\mathrm{s}) \end{bmatrix} = \left[\begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array}\right] \begin{bmatrix} \mathbf{x}_K(\mathrm{s}) \\ \hline \mathbf{v}_{\mathrm{c}}(\mathrm{s}) \\ \mathbf{v}(\mathrm{s}) \\ \mathbf{y}_{\mathrm{f}}(\mathrm{s}) \end{bmatrix}$$
$$C_K = \begin{bmatrix} C_{\delta_{\mathrm{T}}\,K} \\ C_{\delta_{\mathrm{E}}\,K} \end{bmatrix}, \quad D_K = \begin{bmatrix} D_{\delta_{\mathrm{T}}\,K} \\ D_{\delta_{\mathrm{E}}\,K} \end{bmatrix} \qquad (3)$$

More formally, $K(\mathrm{s})$ shall be designed in such a way that, even under dispersed flight conditions and wind levels, two driving requirements are fulfilled:

- *Tracking*, minimise the error $\mathbf{v}_{\mathrm{e}}(\mathrm{s}) = \mathbf{v}_{\mathrm{c}}(\mathrm{s}) - \mathbf{v}(\mathrm{s})$ throughout the final approach and, most importantly, at flare activation, to reduce the initial dispersions for altitude and rate conditions against which the flare controller needs to cope;
- *Actuation*, minimise the overall control effort $\mathbf{u}(\mathrm{s})$ and especially its reactivity to the disturbances $\mathbf{w}_{\mathrm{d}}(\mathrm{s})$ at high-frequencies, which are prone to induce undesirable dynamics in the system.

4. STANDARD & STRUCTURED $\mathcal{H}_\infty$ FRAMEWORK

In the $\mathcal{H}_\infty$ framework, control requirements are defined in two steps. First, the closed-loop interconnections are rearranged into a generalised plant $P(\mathrm{s})$, which gathers command and disturbance signals as exogenous inputs, and error and performance measurements as regulated outputs. Furthermore, all the uncertainties are pulled out of this plant as an upper LFT and the controller is connected as a lower LFT (Doyle et al., 1991). Secondly, the input-output channels of $P(\mathrm{s})$ are normalised by augmenting the system with dedicated frequency-dependent weights. This process is illustrated in Fig. 2.

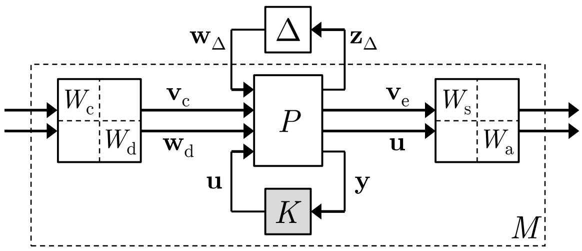

Fig. 2. Generalised interconnections in the $\mathcal{H}_\infty$ framework

The robust control problem consists in finding a stabilising controller $K^*(\mathrm{s})$ that minimises the $\mathcal{H}_\infty$-norm of $M(\mathrm{s})$:

$$\min ||M(\mathrm{s})||_\infty = \min \sup_{\omega \in \mathbb{R}} \bar{\sigma}\left(M(j\omega)\right) \tag{4}$$

which corresponds mathematically to its maximum singular value $\bar{\sigma}\left(M(j\omega)\right)$ and physically to the worst-case amplification of exogenous energy-bounded signals. Input-output weights are then employed so that, if all the control requirements are fulfilled:

$$||M(\mathrm{s})||_\infty < 1 \tag{5}$$

For the glide-slope approach problem, the design weights of Fig. 2 are chosen as follows:

- $W_\mathrm{c}(\mathrm{s})$ employs a differential scaling to $\mathbf{v}_\mathrm{c}(\mathrm{s})$ according to the maximum velocity commanded, which can be extracted from the aircraft benchmark. It can also be used to specify the frequency content of the command signals but, to keep the order of the system as small as possible, it is simply set to $W_\mathrm{c}(\mathrm{s}) = \mathrm{diag}\,(4.55, 1.95)$;
- $W_\mathrm{d}(\mathrm{s})$ is equivalent to $W_\mathrm{c}(\mathrm{s})$, but applied to scale the wind disturbances $\mathbf{w}_\mathrm{d}(\mathrm{s})$. In a similar fashion, it is defined as $W_\mathrm{d}(\mathrm{s}) = \mathrm{diag}\,(3.75, 2.50)$;
- $W_\mathrm{s}(\mathrm{s})$ imposes the *tracking* objectives mentioned in Sec. 3 since the singular values of the channel $\mathbf{v}_\mathrm{e}(\mathrm{s})$ will be bounded by $W_\mathrm{s}^{-1}(\mathrm{s})$. This inverse shall have the shape of a typical sensitivity transfer function in the diagonal terms, with a high-frequency gain around 2 dB for good stability margins, a low-frequency gain 80 dB lower for small steady-state error and a bandwidth that is suitable for the aircraft dynamics. With this, and with the input scaling $W_\mathrm{c}(\mathrm{s})$ in mind, this output weight corresponds to: $W_\mathrm{s}(\mathrm{s}) = \mathrm{diag}\left(\frac{0.10\mathrm{s}+0.01}{\mathrm{s}+1.0\times10^{-5}}, \frac{0.25\mathrm{s}+0.08}{\mathrm{s}+3.2\times10^{-5}}\right)$;
- $W_\mathrm{a}(\mathrm{s})$ enforces the *actuation* requirements in the same way, by limiting $\mathbf{u}(\mathrm{s})$ with $W_\mathrm{a}^{-1}(\mathrm{s})$. Therefore, the low-frequency gain of the latter shall correspond to the maximum actuator commands, the high-frequency gain shall be small enough to prevent intense reactivity in this zone and the roll-off frequency shall be in accordance with that of the actuator dynamics. The weight is then defined as: $W_\mathrm{a}(\mathrm{s}) = \mathrm{diag}\left(\frac{333.3\mathrm{s}+666.7}{\mathrm{s}+200.0}, \frac{12.9\mathrm{s}+295.4}{\mathrm{s}+128.9}\right)$.

Although not exploited in this paper, additional types of requirements can be directly introduced by weighting other performance signals, such as $q(\mathrm{s})$, $n_\mathrm{z}(\mathrm{s})$ or $\dot{\mathbf{u}}(\mathrm{s})$. As anticipated in Sec. 1, standard $\mathcal{H}_\infty$ synthesis with this configuration will originate controllers of order 10 (6 from $G(\mathrm{s})$, 2 from $W_\mathrm{s}(\mathrm{s})$ and 2 from $W_\mathrm{a}(\mathrm{s})$), which often result initially in fast or even unstable internal dynamics.

In the next section, the transition to non-smooth $\mathcal{H}_\infty$ optimisation (Apkarian and Noll, 2006; Gahinet and Apkarian, 2011) is addressed as a way to design fixed-order (and fixed-structure) controllers fulfilling the same performance characteristics. The structured $\mathcal{H}_\infty$ algorithm under investigation is currently part of MATLAB's *Robust Control Toolbox* through the routine *systune*.

Although *systune* allows to systematically decouple and handle additional types of control objectives, only the interconnections of Fig. 2 will be considered in order to focus on the link with the standard $\mathcal{H}_\infty$ framework. For the same reason, only little adjustments are made to the weights within different control designs, with all being analysed against the ones defined above.

5. GLIDE-SLOPE APPROACH CONTROL DESIGN

Following the model of Sec. 3 and the framework of Sec. 4, this section shows the process of designing sequentially improved structured glide-slope controllers by taking into account results from the standard $\mathcal{H}_\infty$ understanding and from other analysis tools. Hence, before proceeding with the successive control designs (five in total, see Sec. 5.1 to 5.5), the criteria against which they are analysed and compared are described:

- *Nominal* performance of the *linear* system, which is satisfied if $||M(\mathrm{s})||_\infty < 1$ with $\Delta = 0$. As mentioned in Sec. 4, it corresponds to having the multi-channel singular values of $P(\mathrm{s})$ appropriately scaled by the input weights $W_\mathrm{c}(\mathrm{s})$ and $W_\mathrm{d}(\mathrm{s})$ and bounded by the inverse weights $W_\mathrm{s}^{-1}(\mathrm{s})$ and $W_\mathrm{a}^{-1}(\mathrm{s})$. This is visualised for the different controllers in Fig. 3;
- *Robust* performance of the *linear* system, which is satisfied if the same control requirements are fulfilled for all the allowable parametric uncertainties $||\Delta||_\infty \leq 1$. This is assessed through the structured singular value $\mu(M)$, defined as:

$$\mu(M) = \frac{1}{\min_{\Delta_\mathrm{P}} \left\{\bar{\sigma}(\Delta_\mathrm{P}) : \det(\mathbb{I} - M\Delta_\mathrm{P}) = 0\right\}} \tag{6}$$

where Δ_P corresponds to the block Δ augmented with a fictitious complex uncertainty that closes the outputs and inputs of $M(\mathrm{s})$ and $\bar{\sigma}(\Delta_\mathrm{P})$ represents its maximum singular values (Doyle et al., 1991). All the requirements are fulfilled in dispersed conditions if and only if $\mu(M) < 1$ and, even when this is not the case, a controller that provides smaller values of $\mu(M)$ is necessarily associated with better Robust Performance (RP) properties.

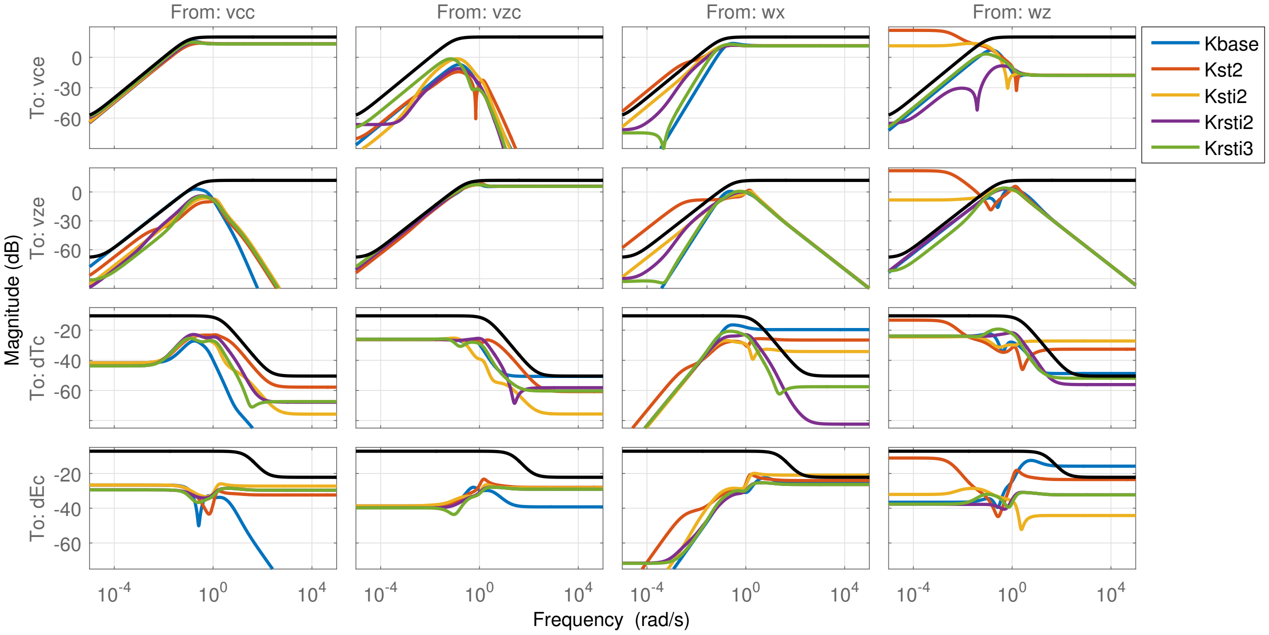

Fig. 3. Multi-channel singular values of the nominal plant ($\Delta = 0$) against the design weights W_s^{-1} and W_a^{-1} (in black)

Separate RP plots for *tracking* and *actuation* requirements are provided in Fig. 4 by computing $\mu(M)$ considering the output channels $\mathbf{v}_e(s)$ (top plot) and $\mathbf{u}(s)$ (bottom plot) individually. Although the computation of (6) is NP-hard and consequently estimated by lower and upper bounds, they coincide very well for this case and so only the latter (computed using the LMI-based algorithm (Doyle et al., 1991) available with MATLAB) is shown in Fig. 4. Robust Stability (RS) is also verified via μ analysis but, since RS is effectively met with all the controllers, it is not shown;

- *Nominal* performance via *nonlinear* simulation, which is assessed by testing the different controllers until flare activation using the nonlinear simulator of Sec. 2 with nominal flight parameters. For these tests, the localiser controller, ILS noise and wind seeds are kept the same and the headwind is set to its maximum level (30 kts). The most relevant longitudinal variables are depicted in Fig. 5;
- *Robust* performance via *nonlinear* simulation, which is assessed through MC campaigns with 100 runs of the nonlinear simulator. For these runs, all the flight parameters, ILS noise and wind seeds and levels are dispersed, but all the controllers are compared with the same conditions. The responses of the vertical velocity to be tracked and the related altitude error, together with the Gaussian fits of their distributions at flare activation, are provided in Fig. 6.

It shall be noted that there is no formal connection between the results of Fig. 6 and the requirements analysed in Fig. 4 with $\mu(M)$, although it is reasonable to argue that better *tracking* and *actuation* properties during the flight are more likely to originate narrower dispersions at flare activation and, consequently, ease the effort required from the flare controller (see Sec. 3). In addition, the set of parametric dispersions simulated is larger than the one addressed with μ analysis, which only accounts for the parameters captured in (1).

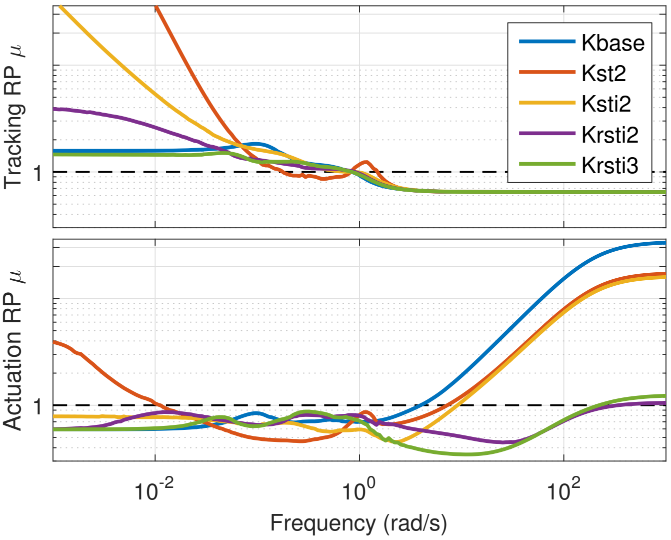

Fig. 4. Structured singular value $\mu(M)$ for RP analysis of tracking and actuation requirements separately

5.1 Baseline controller: K_{base}

The controller to be considered as design baseline is provided in Biannic (2016) and here designated K_{base}. It is a second-order system with pure integration of velocity errors which, referring to (3), particularises into:

$$A_{K_{\text{base}}} = \mathbb{O}_2, \qquad B_{K_{\text{base}}} = [\mathbb{I}_2 \; -\mathbb{I}_2 \; \mathbb{O}_2] \tag{7}$$

It shall be noticed that, according to the weights defined in Sec. 4, K_{base} provides very poor nominal and robust performance in terms of *actuation* (Figs. 3 and 4, respectively). The reason for choosing such weights was precisely to minimise control effort at high-frequencies, preventing the saturation of δ_T (bottom-left plot of Fig. 5) and reducing the amplitude of δ_E (bottom-right plot of the same figure).

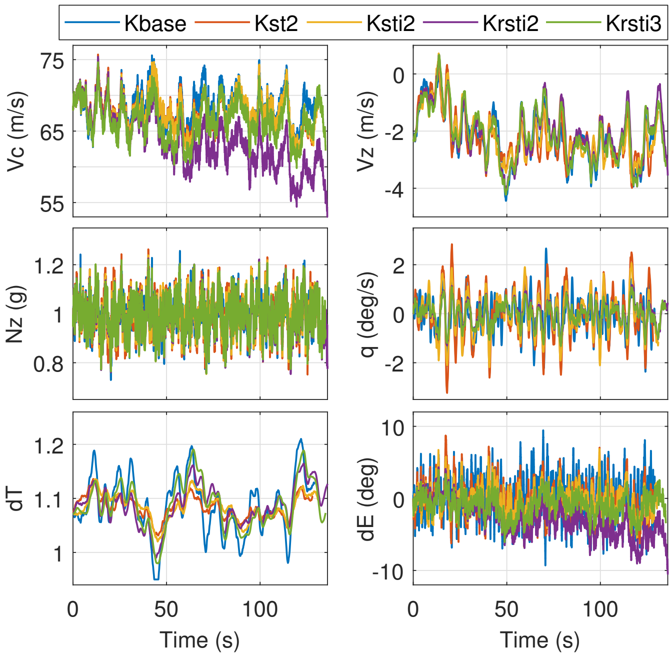

Fig. 5. Nonlinear simulation under nominal conditions and maximum headwind level until flare activation

5.2 *First structured controller:* K_{st2}

This control design, K_{st2}, is obtained applying structured $\mathcal{H}_\infty$ and the requirements of Sec. 4 to the nominal plant ($\Delta = 0$) and using a set of random second-order controllers as initial conditions. Note that this is not a structured controller *per se*; structured $\mathcal{H}_\infty$ was employed to constrain its order but not its architecture at this point.

Poor nominal and robust performance with respect to $w_{\mathrm{x}}(\mathrm{s})$ and $w_{\mathrm{z}}(\mathrm{s})$ disturbance inputs are observed in Figs. 3 and 4, respectively. This, when tested with the nonlinear simulator, leads to an unacceptable *tracking* in Fig. 6.

5.3 *Structured controller initialised with* $\mathcal{H}_\infty$*:* K_{sti2}

The objective of this control design, K_{sti2}, is precisely to obtain more meaningful singular value responses by avoiding the random initialisation employed for K_{st2}. To achieve this while keeping the design choices of Sec. 5.2 (i.e. using $\Delta = 0$ and the same $\mathcal{H}_\infty$ objectives), the following generally-applicable synthesis procedure is proposed:

(1) A standard $\mathcal{H}_\infty$ controller is computed. This controller is of order 10 (see Sec. 4) and originates $||M(\mathrm{s})||_\infty = 1.00$;
(2) A controller is generated reducing the standard $\mathcal{H}_\infty$ design to order 2 via balanced truncation;
(3) The same structured problem of Sec. 5.2 is solved, but using the reduced controller as single initial condition. The solution is K_{sti2}, which yields $||M(\mathrm{s})||_\infty = 1.07$.

Without a proper initialisation, *systune* would converge to solutions similar to K_{st2}. However, due to the manipulation proposed above, the performance achieved with K_{sti2} is substantially better (see Figs. 3 and 6).

Nonetheless, because K_{sti2} is specifically tuned for the nominal plant, it is still very sensitive to model uncertainties as indicated by the plots of $\mu(M)$ in Fig. 4.

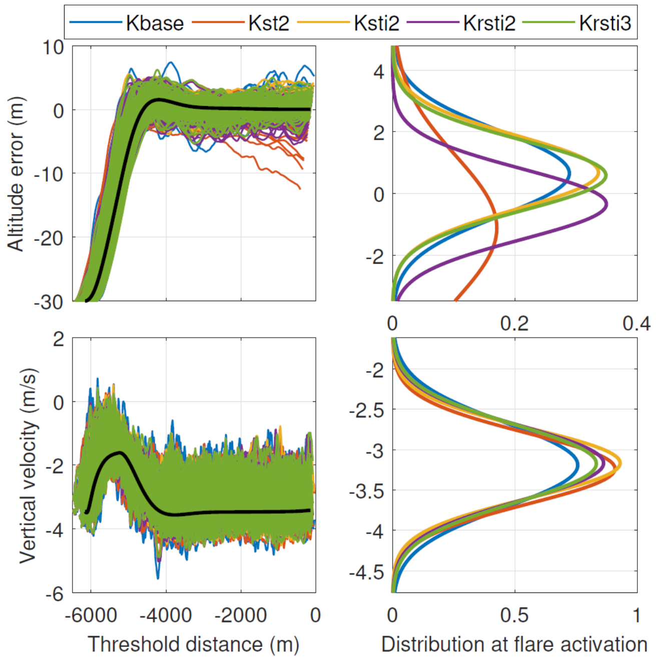

Fig. 6. Results of equal MC runs until flare activation (black lines show the nominal response without wind)

5.4 *Structured controller retuned for the LFT:* K_{rsti2}

This glide-slope controller, K_{rsti2}, is designed following the exact same steps of Sec. 5.3 (i.e., initialising with a reduced standard $\mathcal{H}_\infty$ controller), but replacing the nominal plant with the full LFT, since *systune* is currently able to account for parametric uncertainties (Apkarian et al., 2015).

As expected, using the LFT for synthesis allows to significantly improve the robust performance of the system, especially in terms of *actuation* (Fig. 4). However, if done inattentively, this may come at the expense of a loss of nominal performance, as verified by the degraded *tracking* of v_{c} in Figs. 3 and 5 (recall that the calibrated airspeed should be kept constant during the approach). This is then reflected into an erratic distribution of altitude for K_{rsti2} at flare activation, in Fig. 6.

5.5 *Added-structure controller retuned for the LFT:* K_{rsti3}

As verified in Sec. 5.4, the controller developed therein for the LFT has to be retuned for better performance. To do so, the same synthesis steps were repeated using modified design weights. However, it soon became clear that the optimiser would easily converge to solutions with inadequate singular value responses of $v_{\mathrm{cc}}(\mathrm{s}) \to \delta_{\mathrm{Tc}}(\mathrm{s})$, although very good in the channels related to $\delta_{\mathrm{Ec}}(\mathrm{s})$. This is an indication that a controller with more structure is needed to meet the desired requirements.

Therefore, the original weights of Sec. 4 are kept but one extra degree-of-freedom is added to the controller. Nonetheless, this extra state is not added randomly. Instead, the ability of structured $\mathcal{H}_\infty$ to specify which parameters or blocks are free to be optimised is exploited in order to keep the very good behaviour of $\delta_{\mathrm{Ec}}(\mathrm{s})$ while only tuning the other channel. Referring to (3), this is done by initialising K_{rsti3} based on K_{rsti2}, with:

$$
\begin{aligned}
A_{K_{\mathrm{rsti3}}} &= \begin{bmatrix} A^*_{\mathrm{red}(K_{\mathrm{rsti2}})} & 0^* & 0^* \\ 0 & & \\ 0 & A_{K_{\mathrm{rsti2}}} & \end{bmatrix}, \; B_{K_{\mathrm{rsti3}}} = \begin{bmatrix} B^*_{\mathrm{red}(K_{\mathrm{rsti2}})} \\ B_{K_{\mathrm{rsti2}}} \end{bmatrix} \\
C_{K_{\mathrm{rsti3}}} &= \begin{bmatrix} C^*_{\delta_{\mathrm{T}}\,\mathrm{red}(K_{\mathrm{rsti2}})} & 0^* & 0^* \\ 0 & C_{\delta_{\mathrm{E}}\,K_{\mathrm{rsti2}}} & \end{bmatrix} \\
D_{K_{\mathrm{rsti3}}} &= \begin{bmatrix} D^*_{\delta_{\mathrm{T}}\,K_{\mathrm{rsti2}}} \\ D_{\delta_{\mathrm{E}}\,K_{\mathrm{rsti2}}} \end{bmatrix}
\end{aligned}
$$

In these expressions, the superscript $*$ denotes the blocks and parameters that are free to be tuned and the subscript red(K_{rsti2}) represents the reduction of the $\delta_{\mathrm{Tc}}(\mathrm{s})$ channel of K_{rsti2} to order 1 via balanced truncation. It shall be noted that, although the dynamics of $\delta_{\mathrm{Ec}}(\mathrm{s})$ are unaffected, the $\delta_{\mathrm{Tc}}(\mathrm{s})$ channel is able to use its states.

Once optimised, the impact of the extra dynamics of K_{rsti3} is clearly verified in Fig. 3, with smoother responses, especially in the $v_{\mathrm{ce}}(\mathrm{s})$ channel. Consequently, an improvement of nominal performance is also observed on the other channels, which is then confirmed via nonlinear simulation (Fig. 5). Although similar to the other controllers concerning nominal *tracking*, K_{rsti3} is able to reduce significantly the high-frequency *actuation*, which results in lower pitch rate q and also load factor N_{z} oscillations. Relative to the other controllers, this comes at the cost of only one additional degree-of-freedom.

Regarding robust performance (Fig. 4), K_{rsti3} surpasses all the other controllers in terms of *actuation* and *tracking*, which is proved via the MC analysis of Fig. 6. In fact, although the controllers generate a similar vertical velocity distribution, the smallest integrated error with K_{rsti3} clearly makes it the best performing controller, with the narrowest altitude error distribution at flare activation. This then facilitates the action of the flare controller (recall Sec. 3) in the succeeding phase of the automatic landing.

6. CONCLUSIONS

This paper illustrated the application of structured $\mathcal{H}_\infty$ optimisation to the synthesis of a glide-slope approach control system with good robustness against changing aircraft conditions and strong wind gusts. The effects of varying aircraft conditions are captured via a Linear Fractional Transformation (LFT) model and the wind gusts are modelled as external disturbances.

Moreover, with this application, it was demonstrated how design choices that are easily overlooked can be decisive for the quality of the solution found. Particular emphasis was placed on the benefits of employing a standard $\mathcal{H}_\infty$ design to initialise the structured $\mathcal{H}_\infty$ optimiser and of exploiting the latter's ability to specify which parameters are free to be tuned. A generally-applicable synthesis procedure based on the former was also proposed in Sec. 5.3.

During the design process, the aforementioned choices were taken and justified based on combined linear analyses and nonlinear simulations for both nominal and dispersed flight conditions. Regarding the latter, robustness assessments in the linear domain are performed using the structured singular value μ and validated via nonlinear Monte-Carlo campaigns. All the insights provided by these tools are complementary and crucial for the development of a successful control system.

The design of well-performing and robust aircraft automatic landing systems is indeed a very complex problem and the most effective way to tackle it is by addressing its compounding control modes (glide-slope, localiser, flare) separately. The glide-slope approach controller K_{rsti3} is now integrated in a complete autoland system developed for the aircraft landing challenge.

REFERENCES

Apkarian, P., Dao, M., and Noll, D. (2015). Parametric robust structured control design. *Transactions on Automatic Control*, 60(7), 1857–1869.

Apkarian, P. and Noll, D. (2006). Nonsmooth $\mathcal{H}_\infty$ Synthesis. *Transactions on Automatic Control*, 51(1), 71–86.

Biannic, J.M. (2016). *Nonlinear Civilian Aircraft Landing Benchmark*. Technical note included in the benchmark package available with the SMAC toolbox at `http://w3.onera.fr/smac/aircraftModel`.

Biannic, J.M. and Apkarian, P. (2001). A new approach to fixed-order $\mathcal{H}_\infty$ synthesis: Application to autoland design. In *The 2001 AIAA Guidance, Navigation, and Control Conference and Exhibit*. Montreal, Canada.

Biannic, J.M. and Roos, C. (2015). Flare control law design via multi-channel $\mathcal{H}_\infty$ synthesis: Illustration on a freely available nonlinear aircraft benchmark. In *The 2015 American Control Conference*. Chicago, IL.

Doyle, J., Packard, A., and Zhou, K. (1991). Review of LFTs, LMIs and μ. In *The 30th IEEE Conference on Decision and Control*. Brighton, UK.

Falcoz, A., Pittet, C., Bennani, S., Guignard, A., Bayart, C., and Frapard, B. (2015). Systematic design methods of robust and structured controllers for satellites. *CEAS Space Journal*, 7(3), 319–334.

Gahinet, P. and Apkarian, P. (2011). Structured $\mathcal{H}_\infty$ Synthesis in MATLAB. In *The 18th IFAC World Congress*. Milan, Italy.

Iannelli, A., Simplício, P., Navarro-Tapia, D., and Marcos, A. (2017). LFT Modeling and μ Analysis of the Aircraft Landing Benchmark. In *The 20th IFAC World Congress*. Toulouse, France.

Lhachemi, H., Saussié, D., and Zhu, G. (2014). A Robust and Self-Scheduled Longitudinal Flight Control System: a Multi-Model and Structured $\mathcal{H}_\infty$ Approach. In *The AIAA SciTech 2014 Conference*. National Harbor, MD.

Looye, G. and Joos, H.D. (2006). Design of Autoland Controller Functions with Multiobjective Optimization. *J. of Guidance, Control, and Dynamics*, 29(2), 475–484.

Navarro-Tapia, D., Simplício, P., Iannelli, A., and Marcos, A. (2017). Robust Flare Control Design using Structured $\mathcal{H}_\infty$ Synthesis: a Civilian Aircraft Landing Challenge. In *The 20th IFAC World Congress*. Toulouse, France.

Sadat-Hoseini, H., Fazelzadeh, S., Rasti, A., and Marzocca, P. (2013). Final Approach and Flare Control of a Flexible Aircraft in Crosswind Landings. *Journal of Guidance, Control, and Dynamics*, 36(4), 946–957.

Simplício, P., Bennani, S., Marcos, A., Roux, C., and Lefort, X. (2016). Structured Singular-Value Analysis of the Vega Launcher in Atmospheric Flight. *Journal of Guidance, Control, and Dynamics*, 39(6), 1342–1355.

Theis, J., Ossmann, D., and Pfifer, H. (2017). Robust Autopilot Design for Crosswind Landing. In *The 20th IFAC World Congress*. Toulouse, France.