HAL open science

Adaptive LFT control of a civil aircraft with online frequency-domain parameter estimation

Gilles Ferreres, Georges Hardier

▶ To cite this version:

Gilles Ferreres, Georges Hardier. Adaptive LFT control of a civil aircraft with online frequency-domain parameter estimation. International Journal of Robust and Nonlinear Control, In press, 10.1002/rnc.3993 . hal-01849031

HAL Id: hal-01849031
https://hal.science/hal-01849031

Submitted on 25 Jul 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adaptive LFT control of a civil aircraft with on-line frequency domain parameter estimation

G. Ferreres and G. Hardier
ONERA/DTIS, Toulouse, France
gilles.ferreres@onera.fr, georges.hardier@onera.fr

Abstract: this paper describes the application of an indirect LFT-based state-space adaptive control scheme to a transport aircraft, within the context of the European project RECONFIGURE. The principle of the scheme is to off-line design and validate a gain-scheduled controller, depending on the plant parameters to be estimated, and to on-line combine it with a model estimator, so as to minimize the on-board computational time and complexity. A modal approach, very classical for the design of a flight control law, is used to directly synthesize the static output feedback LFT controller depending on the control and stability derivatives, i.e. the parameters of the linearized aerodynamic state-space model to be estimated. Since the gain-scheduled LFT controller online depends on the parameter estimates instead of the true values, its robustness to transient and asymptotic estimation errors needs to be assessed using μ and IQC analysis techniques. A primary concern being on-line implementation, a fully recursive frequency domain estimation technique is proposed, with a low on-line computational burden and the capability to track time-varying parameters. Full non-linear simulations along a trajectory validate the good performance properties of the combined estimator and gain-scheduled flight controller. To some extent, minimal guaranteed stability and performance properties of the adaptive scheme can be ensured, by switching to a robust controller when the parameter estimates are not reliable enough, thus bypassing the Certainty Equivalence Principle.

1. INTRODUCTION

Flight control has long been a privileged applicative field for robust and gain-scheduling control, see e.g. [1,6,10,39,42,45], and more recently for fault-tolerant and adaptive control [3,5,8,13,25,31,40,41,44,47,49-54]. Noting that strong links exist between all these fields. Indeed, robust and adaptive control can be seen as competing/complementary techniques for solving the same problem of controlling an uncertain plant: if a robust controller aims to stabilize the whole set of possible plant instances, an adaptive controller uses on-line measurements of the plant Inputs/Outputs (I/O) to stabilize the plant itself. On the other hand, gain-scheduling can be seen as a special case of indirect adaptive control, whose principle is to estimate the plant parameters and to adjust the controller as a function of these on-line estimates: in the same way, a linear gain-scheduled controller adapts to the nonlinear plant using measured scheduling parameters.

Gain-scheduling is classically used to control an airplane along a trajectory, where the controller gains depend on measured or estimated flight parameters, such as airspeed, Mach, mass or Center of Gravity (CoG) position. However, situations may occur where the performance of the gain-scheduled controller is no more satisfactory, e.g. in the presence of icing, actuator faults, or when a scheduling parameter becomes unavailable due to a sensor fault, especially airspeed [15]. Adaptive control is an attractive solution in this context to recover closed loop performance.

Most of the adaptive control literature focuses on the use of Input/Output (i.e. black box) models to control the system [27]. Nevertheless, a physical aerodynamic state-space model is available for an aircraft (A/C), and state-space methods are typically used to design robust or gain-scheduled flight control laws, so that adaptive flight controllers are usually designed using state-space methods, see e.g. [5,13,25,31,40,41,44,47,50-54].

Generally speaking, two main problems need to be solved in adaptive control, namely to obtain a priori guaranteed stability and performance properties of the adaptive closed loop, and to decrease the on-line computational time and complexity. In the context of indirect adaptive control, following [11,12], a solution based on robust and gain-scheduled control tools is to off-line design a gain-scheduled controller depending on the plant parameters to be estimated, to avoid the complexity of implementing a control design algorithm on-line.

On the other hand, LFT (Linear Fractional Transformation) representations of the plant (depending on the parameters to be estimated) and controller (depending on the parameter estimates) are connected to obtain an LFT closed loop depending on the true values of the plant parameters and on the estimation error. Using μ analysis with constant and frequency-dependent D, G scalings [9,14], and more generally Integral Quadratic Constraints (IQCs) [23,35], it is possible to study the stability and H_∞/L_2 performance properties of the adaptive closed loop, considered as a gain-scheduled closed loop with an error on the scheduling parameters. As a specificity of this validation technique, the estimator is not included in the analysis which just provides specifications, i.e. a maximal amount of the allowable estimation error, and possibly a maximal rate of variation of the estimates. This estimation error can be considered as time-invariant or time-varying, thus studying the robustness of the adaptive scheme in the face of an asymptotic or transient estimation error.

Several gain-scheduling state-space control design techniques can be used, which assume time-varying or time-invariant scheduling parameters, see [42] for a survey. In the sequel, we will focus on the modal technique of [33] because this one appears especially suitable for aeronautical applications: a static output feedback controller placing the main closed loop poles is synthesized under an LFT form. It only depends on the main A/C parameters to minimize on-line complexity, and to avoid an unsatisfactory transient response of the adaptive closed loop. Indeed, the quality of the estimated secondary A/C parameters is generally poor, so that a bad transient estimation error would give a bad transient closed loop response. This static LFT controller can be implemented either using the efficient recursive technique of [34], which avoids any matrix inversion, or using a surrogate model, see section 6.

Another main issue is to develop a method for estimating the A/C model parameters on-line, with a low computational burden and the capability to track time-varying parameters despite measurement noises, poor data information contents (e.g. in cruise condition when variables are likely to be almost constant during long periods), and external disturbances (turbulence). Whatever method used, residual errors can sometimes be large and a measure of the parameter accuracy should also be provided, so that some logic can be introduced: if the estimated scheduling parameters are not reliable enough, one can switch to a robust back-up controller ensuring minimal performance properties.

Algorithms for recursive parameter estimation have been developed for real-time applications in various fields such as industrial processes, robotics and aerospace [18,29]. Most theoretical aspects were covered by early reference publications [2,30]. In aeronautics, the early attempts to benefit from on-line parameter identification date back to the 80s, but they were limited by the available computational power of onboard processors. The common Time Domain (TD) methods are usually based on recursive or sequential Least-Squares (LS), or Extended Kalman Filters to cope with nonlinearities [8,17]. More recently, this challenge has also been addressed by recursive subspace identification of linear or LPV (Linear Parameter-Varying) models. However, the relevance of such advanced multi-model or nonlinear schemes for on-board implementations is dubious due to the present limitations of on-board computers, especially for A/C because of certification issues and code verification. That is why we look in this paper for a simple enough approach, more likely to satisfy those real time constraints, and that could constitute a viable alternative for updating the models required by the most advanced FDD (Fault Detection and Diagnosis) and FTC (Fault Tolerant Control) schemes.

The paper is organized as follows. §2 presents the aircraft problem. The general principles of the estimation process are described in §3, see [20] for a more complete presentation. An LFT flight controller is synthesized in §5 using the tools briefly described in §4, and some issues related to implementation aspects are covered by §6. Adaptive control results using the realistic non linear simulator of the RECONFIGURE project are presented in §7. Concluding remarks end the paper.

2. THE AIRCRAFT PROBLEM

The indirect adaptive control strategy presented in this paper has been applied in the framework of the EU-FP7 funded project RECONFIGURE (REconfiguration of CONtrol in Flight for Integral Global Upset REcovery). The goal of this project is to investigate and to evaluate aircraft guidance and control technologies that facilitate the automated handling of off-nominal/abnormal events, alleviate the pilot workload, and that optimize the aircraft status by automatically reconfiguring the aircraft to an optimal flight condition (see http://reconfigure.deimos-space.com/). A detailed description of the benchmark model and the fault/failure scenarios is given in [15,24]. The V&V process involves a nonlinear highly representative model of a generic Airbus civil aircraft. The benchmark contains a baseline gain scheduled PI controller, actuator/sensor models, measurement filters, as well as control law protections. The different control designs developed during the project had to demonstrate their performance and robustness thanks to the industrial benchmarking proposed by Airbus, by using an extensive set of simulations including a wide range of operational conditions (realistic flight scenarios, external disturbances, pilot maneuvers). The computational burden and complexity of the schemes had also to be compliant with stringent computational constraints in order to be possibly implemented on embedded computers. Robustness issues were tackled by using tools like a Functional Engineering Simulator, implemented during the project to achieve a traditional Monte Carlo analysis as a preliminary step of the industrial V&V process [24]. It is also noteworthy that avenues do exist for improving the V&V of more complex controllers like the proposed adaptive scheme; for instance, a practical approach for the worst-case analysis of a similar adaptive controller was already investigated by Airbus in [41].

The highly realistic nonlinear simulator of the RECONFIGURE project, which is very close to a real civil aircraft, includes a nonlinear aerodynamic model with 5 longitudinal states: the angle of attack α, the pitch rate q, the pitch angle θ, the true airspeed V and the altitude z. The acceleration N_z is also to be accounted for as an additional output[2]. Two control loops are considered: the inner loop is used to control the fast longitudinal states α and q, as well as the output N_z, using the flight controller. The outer loop is used to control the other longitudinal states using an autopilot and an auto-throttle. In the context of the RECONFIGURE project the autopilot and auto-throttle are given, only the flight controller is allowed to be modified to recover the performance of the inner loop, e.g. in the presence of an actuator/sensor failure. An adaptive flight controller is synthesized in this paper.

In the context of our indirect adaptive control scheme, a linearized longitudinal aerodynamic model with 4 states α, q, θ, V and the output N_z should be a priori estimated:

[2] In the following, the variables α, q and so on of the linearized models should be understood as departures from trim variables, whereas they correspond to global variables when using the non-linear simulator.

$$\begin{cases} \dot{\alpha} = Z_\alpha \alpha + Z_q q + Z_\delta \delta_m + Z_V V \\ \dot{q} = M_\alpha \alpha + M_q q + M_\delta \delta_m + M_V V \\ \dot{\theta} = q \\ \dot{V} = X_\alpha \alpha + X_q q + X_\delta \delta_m + X_V V \\ N_z = N_\alpha \alpha + N_q q + N_\delta \delta_m + N_V V \end{cases} \tag{1}$$

δ_m is the elevator input. The effect of altitude, which is the slowest longitudinal state, is neglected. The aircraft is supposed to move about a given flight condition and a constant engine thrust is assumed.

Nevertheless, as explained in the following, only the first two lines and the last line of model (1) need to be estimated in practice, i.e. those corresponding to $\dot{\alpha}$, $\dot{q}$ and N_z. More precisely, the issue is to estimate the control and stability derivatives, i.e. the parameters Z_α, Z_q ... of the linearized aerodynamic model. A linear in the parameters model is obtained, i.e.:

$$[\dot{\alpha} \quad \dot{q} \quad Nz]^T = \Theta [\alpha \quad q \quad \delta_m \quad V]^T \tag{2}$$

The Θ matrix includes the 12 parameters to be estimated. In the context of the adaptive scheme, the main goal of the on-line estimator is to update the values of the 5 main control and stability derivatives $Z_\alpha, M_\alpha, M_q, M_\delta, N_\alpha$ used to schedule the controller (§5.1), although all the 12 coefficients have to be estimated.

Only the first two lines and the last line of model (1) need to be estimated because the linearized model used to design the flight controller only contains the fast longitudinal states α and q, as well as the output N_z:

$$\begin{cases} \dot{\alpha} = Z_\alpha \alpha + Z_q q + Z_\delta \delta_m \\ \dot{q} = M_\alpha \alpha + M_q q + M_\delta \delta_m \\ N_z = N_\alpha \alpha + N_q q + N_\delta \delta_m \end{cases} \tag{3}$$

The flight controller, designed using the 2 states linearized model above, is a posteriori validated on the 4 states linearized model (1). Its architecture is:

$$\delta_m = K_{Nz} Nz + K_q q + K_I \int (Nz - Nz_c) dt + L\, Nz_c \tag{4}$$

where Nz_c is the reference acceleration input.

3. A FREQUENCY DOMAIN APPROACH FOR ON-LINE PARAMETER ESTIMATION

Adaptive control strategies are rather uncommon in the aircraft field, even if they begin to develop for UAVs, and this has to do with stringent certification issues. Otherwise, practical implementation of indirect adaptive schemes mostly rely on TD algorithms for estimating time-varying or uncertain model parameters [48]. Actually, TD algorithms could seem preferable at first sight, as the basic ones only involve very simple operations (e.g., by replacing a matrix inversion with a scalar division). However, advanced strategies are usually required for regularizing the estimation process, and those advantages can thus be wasted (see [18] for an overview of the pros and cons of TD methods). By contrast, FD techniques have many desirable features for on-line applications [16,19,26]. The computation time can be reduced by processing only a limited amount of frequencies within the bandwidth of interest, and the resulting indirect filtering of the wide-band disturbances (e.g., Low Frequency mismatch and High Frequency noise) improves the estimation accuracy. The standard deviation of the estimation errors can also be evaluated without any additional cost.

Furthermore, the availability of efficient tools making the transition from TD to FD possible, such as the recursive Fourier Transform (FT), greatly facilitates their use, and this was stressed in many publications during the last 20 years [4,16,26,36-38,46]. Fig. 1 gives a schematic representation of the way the FD information is linked to the estimation process. Moreover, many issues specific to on-line estimation are mitigated when using FD approaches instead of TD ones: estimates are nearly unbiased even in the presence of noisy I/O data in case of collinearity in the regressors, the estimation errors do not require to be improved afterwards due to possible colored residuals, and so on (see [37] for all these practical aspects).

Otherwise, TD/FD estimation schemes rely either on recursive algorithms (making use of measurements as soon as they are available), or alternatively on sequential procedures, processing moving data windows with a lower rate to get a succession of piecewise constant values [7,18]. In [19], a sequential process was proposed for monitoring a civil aircraft, that permits pre and post-processing stages to be included in the procedure to prevent from and to filter out inaccurate estimations. Even if both approaches have their own pros and cons [22], recursive schemes should be favored for on-line implementation, and FD is more suitable to define hybrid recursive/sequential procedures because the information naturally accumulates in the signals' FT, which behave like storing memories and are managed with more flexibility (truncation, forgetting, resetting).

Finally, one of the major reasons for which recursive TD algorithms are more popular than FD ones comes from some additional computation complexity, which has prevented from developing a fully recursive formulation of the FD approach, unlike the TD case. This paper improves the state-of-the-art existing approaches and proposes a recursive procedure from start to finish, permitting the practical implementation constraints to be satisfied, and the algorithm to be possibly embedded in aircraft computers. Other minor issues in the FT expressions of the signal derivatives are also properly settled, which are often disregarded for sequential processing thanks to the detrending achieved prior to FT computations, but which cannot as far as recursive algorithms are concerned. To sum up, the point is that FD approaches can also be simplified in order to reduce their computational complexity, similarly to what is done for TD formulations, and they can also include similar advanced mechanisms like data forgetting [18,30]. Before developing the proposed algorithm in §3.2, section 3.1 recalls the basics of the usual sequential approach in state-space form.

3.1. Standard formulation of a sequential algorithm

The transition to the FD is classically realized by means of the standard FT. As the signals are only available over a limited period of time $[0,T]$, the finite FT is used instead, which permits the FT of any signal $x(t)$, time derivative $\dot{x}(t)$, or constant value to be easily computed in the FD (see [20] for details). Practically, from the sampled values of $x(t)$, the finite FT can be computed via numerical integration:

$$X(\omega) \approx \Delta t \sum_{n=0}^{N-1} x(n\Delta t)\, e^{-j\omega n\Delta t} = \Delta t\, \tilde{X}(\omega,T) \tag{5}$$

using N values equally spaced over $[0,T]$, with a sampling period Δt (see for instance [26]). The term $\tilde{X}(\omega)$ represents the Discrete Fourier Transform (DFT) of the samples $\{\, x(n\Delta t), n = 0,\ldots, N-1 \,\}$. Efficient tools are available for computing this quantity, namely the Fast Fourier Transform (FFT) and the Chirp z-transform (CZT). As the aircraft model we are interested in is expressed in state-space form (see §2), the stability and control derivatives Θ to be estimated are included in the usual state matrices A, B, C, D; they are assumed to be constant or at least to vary slowly during the flight with respect to (w.r.t.) the estimation process and updating rate. However, when applying estimation techniques, it is generally advisable to estimate also state/output biases b_x / b_y in order to cope with I/O measurement offsets or model structure uncertainties [21]. By applying the FT to these state-space equations, if ω is chosen to be a multiple of the frequency step $\Delta\omega$ (which happens for instance when using the FFT), the FD model can be written as [19]:

$$\begin{cases} j\omega X = AX + BU + b_x\, T\, d(\omega) + x(0) - x(T) \\ \quad Y = CX + DU + b_y\, T\, d(\omega) \end{cases} \tag{6}$$

by denoting $X \equiv X(\omega), U \equiv U(\omega), Y \equiv Y(\omega)$, and where $d(\omega)$ is a Dirac function in the FD, such that $d(\omega) = 1$ for $\omega = 0$, and $d(\omega) = 0$ for $\omega = 2k\pi / T \neq 0$ [20]. As a result, the discrepancy $\delta = x(0) - x(T)$ between the state initial and final conditions is translated into a bias that impacts on all frequencies: localized effects in the TD are translated into broadband effects in the FD, and vice versa. A contrario, the biases act as broadband inputs in the TD and just modify the zero frequency. To get the most out of these specificities, it is generally worthwhile to discard the zero frequency during the estimation, which avoids the state and output biases having to be estimated. Thus, (6) is further simplified and reduces to:

$$\begin{cases} j\omega X = AX + BU + x(0) - x(T) \\ \quad Y = CX + DU \end{cases} \tag{7}$$

Hence, the discrepancies δ should be estimated in addition to the other parameters Θ, if not zero. However, this issue has not been paid much attention and is often ignored in practice [19,28], which could be only justified in a pinch if a suitable preprocessing is inserted in a sequential algorithm to filter out these effects. If we assume that all the state components are measured, an Equation Error approach can be used, that consists in minimizing a set of cost functions [36-38] for all the state and output equations in which a subset Θ_i of stability and control derivatives needs to be estimated ($1 \le i \le \dim(X) + \dim(Y)$), the costs summing on M frequency samples available from the FT ($M \le N$). By collecting these M terms in vector and matrix forms, and by denoting Θ_i the extended vector of unknown parameters in A, B, δ (C, D resp.), the cost functions lead to classical LS criteria [38]:

$$\begin{aligned} & J_i(\Theta_i) = (\mathcal{Y} - \mathcal{X}\Theta_i)^{\dagger}(\mathcal{Y} - \mathcal{X}\Theta_i) + (\Theta_i - \Theta_{i0})^T \Sigma_0^{-1}(\Theta_i - \Theta_{i0}) \\ & \text{with}: \begin{cases} \mathcal{X}^T = [X(\omega_1) \dots X(\omega_M); U(\omega_1) \dots U(\omega_M); 1 \dots 1] \\ \mathcal{Y}^T = [j\omega_1 x_i(\omega_1) \dots j\omega_M x_i(\omega_M)] \quad (\text{or } [y_i(\omega_1) \dots y_i(\omega_M)] \text{ resp.}) \end{cases} \end{aligned} \tag{8}$$

where $\dagger$ is the complex conjugate transpose operator. In (8), the conditioning of the resulting optimization has been possibly improved by introducing some a priori knowledge about the expected value of the parameters. This form of regularization can be especially useful for weakly identifiable parameters, by softening their variations and improving the convergence. The principle of this Bayes-like estimation consists in adding a penalty term to the LS criterion to weight the Θ_i increments with respect to prior values Θ_{i0} through a corresponding covariance Σ_0 (usually a diagonal matrix). (8) is nothing but a standard LS regression with complex data instead of real ones. The well-known solution [26,38] to this problem is then computed as $\hat{\Theta}_i = \left[\mathcal{R}(\mathcal{X}^{\dagger}\mathcal{X}) + \Sigma_0^{-1}\right]^{-1}\left[\mathcal{R}(\mathcal{X}^{\dagger}\mathcal{Y}) + \Sigma_0^{-1}\Theta_{i0}\right]$, where $\mathcal{R}()$ is the real part of a complex value.

3.2. Towards a fully recursive algorithm

The previous LS formulation is well known, but was mostly implemented in sequential estimation processes, ignoring the issues of the extra discrepancies δ as well as the need to invert the information matrix. Here, we are in a different prospect and wish to consider the limitations of on-board computers, as well as the certification and code verification issues. A practical implementation on civil aircraft should preclude the use of any iterative algorithm which would result in some unbounded computation loop through the data. Hence complex operations like non analytical matrix inversions should be banned. That is why we look for a simple enough approach, likely to satisfy those real time constraints, and based on a fully recursive algorithm. At first, it should be noticed that the summation (5) makes very simple operations possible (see [36] for instance):

$$\begin{cases} \tilde{X}_n(\omega) = \tilde{X}_{n-1}(\omega) + x(n\Delta t)\, e^{-j\omega n\Delta t} \\ e^{-j\omega n\Delta t} = e^{-j\omega\Delta t}\, e^{-j\omega(n-1)\Delta t} \end{cases} \tag{9}$$

to evaluate the DFT at time $n\Delta t$ from its previous value at time $(n-1)\Delta t$. For a given frequency ω, the quantity $\exp(-j\omega\Delta t)$ is constant, so that the updating of the DFT just requires two multiplications and one addition, resulting in a very low computational effort. As the estimation process also involves state derivatives, the measurements of which are usually not available, a recursive formulation similar to (9) can be worthwhile for signals $\dot{x}(t)$ to avoid a pseudo-derivation from the available signal $x(t)$, or the extra δ

parameters to be estimated if we had to use (7). The idea is to derive LS costs similar to (8) but involving $\dot{x}_i(\omega_k)$ directly in the computation of the error terms as this is the case for a measurement $y_i(\omega_k)$. This idea is new and differs from the two usual ways consisting either in computing an estimate of $\dot{x}(t)$ in the TD, or in replacing $\dot{X}(\omega)$ by $j\omega X(\omega)$ in the FD, which is incorrect whenever $\delta \neq 0$. Hence, the proposed FD algorithm permits extra parameters, like δ, to be eliminated from the Θ vector.

Otherwise, when using recursive updates, older information can be overweighted regarding to recent ones, which can result in much delay in the adaptation process. That is why data forgetting can be used in conjunction with (9), to remove the effect of oldest data by working on a limited time window [38]. In that case, the choice of the window width results from a trade-off between the amount of information available from the data and the sensitivity to A/C parameter variations. This remark also applies to a pure sequential approach, where a non recursive form of FT is applied to the data [19], resulting in a succession of piecewise constant values, updated only after some seconds and hence delayed with respect to the varying parameters. To develop a recursive algorithm and to avoid the storage of a time history of the past data, it is still possible to benefit from the FT linearity w.r.t. the signals to implement a forgetting process similar to the one of TD exponential forgetting [18]; this simply involves multiplying the previous FT by a forgetting factor λ $(\lambda < 1)$ [38]. Combining all these ideas with (9) yields the following recursive expressions for any signal $x(t)$, denoting $X_n \equiv X_n(\omega), \dot{X}_n \equiv \dot{X}_n(\omega), x_n \equiv x(n\Delta t)$ and $x_n^* = x_n\, e^{-j\omega\Delta t}$ (see [20] for details):

$$\begin{cases} X_n \approx \lambda\, X_{n-1} + \Delta t\; x_n^*\, e^{-j\omega(n\text{-}1)\Delta t} \\ \dot{X}_n \approx \lambda\, \dot{X}_{n-1} + [(j\omega\Delta t + 1)\, x_n^* - x_{n-1}]\, e^{-j\omega(n\text{-}1)\Delta t} \end{cases} \tag{10}$$

We are now going to address a key feature of the proposed estimation algorithm. With TD methods, the arrival of a new sample only results in the addition of a new row in the regression matrix [18]. So, the information matrix (or its inverse) can be updated via simple (and light) computations thanks to the Woodbury formula. With FD methods, a new sample modifies all the M rows of the matrix $\mathcal{X}$ through the FT updates of (10). That is why the usual algorithms generally rely on a sequential estimation of the parameters, due to the lack of a recursive update of the matrix $R = \mathcal{R}(\mathcal{X}^{\dagger}\mathcal{X})$ or of its inverse. However, it was established in [20] that a recursive version is achievable anyway, almost similar to the TD ones, thanks to the 3 following lemma.

Lemma 3.1 In case of $m = \dim(\Theta_i)$ regressors, the basis of a time-recursion for the matrix $\mathcal{X} \in \mathcal{R}^{M\times m}$ can be set as:

$$\mathcal{X}_n^{\dagger} = \lambda \mathcal{X}_{n-1}^{\dagger} + \Delta t\, \chi_n\, \mathcal{E}_n^{\dagger} \tag{11}$$

for $\chi \in \mathcal{R}^{m\times 1}$ defined as $\chi(n\Delta t) \equiv \chi_n = [x(n\Delta t) \;\; u(n\Delta t)]^T$ and $\mathcal{E}_n^{\dagger} = [\, e^{j\omega_1 n\Delta t} \;\; \cdots \;\; e^{j\omega_M n\Delta t}\,]$.

Lemma 3.2 As a result, the information matrix $M_n = \mathcal{X}_n^{\dagger}\mathcal{X}_n$ can be broken down into 3 parts as:

$$M_n = \lambda^2 M_{n-1} + \Delta t\, [M\Delta t \chi_n\, \chi_n^T + \lambda \chi_n\, \mathcal{E}_n^{\dagger}\mathcal{X}_{n-1} + \lambda \mathcal{X}_{n-1}^{\dagger}\mathcal{E}_n\, \chi_n^T] = \lambda^2 M_{n-1} + \sum_{i=1}^{3} B_i\, C_i\, D_i \tag{12}$$

where $B_1 = \sqrt{M}\,\Delta t\, \chi_n$, $D_1 = B_1^T$, $B_2 = \sqrt{M}\, B_1$, $D_2 = \lambda \mathcal{E}_n^{\dagger}\mathcal{X}_{n-1}$, $B_3 = D_2^{\dagger}$, $D_3 = B_2^T$, and $C_i = 1$.

Lemma 3.3 Hence, M_n^{-1} can be computed recursively by applying the matrix inversion lemma 3 times to (12), as ($\forall k = 1$ to 3):

$$A_{k+1}^{-1} = [A_k + B_k\, C_k\, D_k]^{-1} = A_k^{-1}(I - s_k^{-1} B_k\, D_k A_k^{-1}) \quad \text{with} \quad A_1 = \lambda^2 M_{n-1},\, A_4 = M_n,\, s_k = D_k A_k^{-1} B_k + 1 \tag{13}$$

Finally, (13) permits the direct updating of M_n^{-1} to be computed without requiring a tricky matrix inversion, by using only a series of matrix additions and products. For the sake of simplicity, Σ_0^{-1} was assumed to be 0 in the previous expression of M_n, but dealing with a priori knowledge is however possible by achieving at first a series of m rank 1 corrections to A_1^{-1} with the diagonal values of Σ_0^{-1}. See [20] for the proofs and for a comparison of the computational cost of this recursive algorithm w.r.t. an iterative blockwise inversion.

4. GAIN-SCHEDULING AND ROBUSTNESS TOOLS

A modal design technique of a static output feedback controller under an LFT form is presented in the first subsection. The second one presents a robustness analysis technique in the presence of time invariant or time varying parameters.

In both cases, a preliminary step is to put the open loop plant model $G(s,\theta)$, where θ is a vector of parameters, under the standard LFT form $y = G(s,\theta)u = F_l[H(s),\Delta]u$, i.e.:

$$\begin{bmatrix} y \\ z \end{bmatrix} = H(s)\begin{bmatrix} u \\ w \end{bmatrix} \tag{14}$$

$$w = \Delta\, z \tag{15}$$

The transfer matrix $H(s)$, with additional I/O w and z, is fixed while $\Delta = diag(\theta_i\, I_{q_i})$ gathers the parametric uncertainties, i.e. each parameter θ_i is repeated q_i times on the diagonal of Δ. The fictitious feedback (15) is used to introduce the parametric uncertainties in the model. Generally speaking, using the LFR Toolbox [32] for instance, any state-space representation with a polynomial or rational dependence on uncertain or varying parameters can be put under this form.

Parameters θ_i are supposed to be normalized in this section, namely the vector θ is supposed to belong to the unit hypercube H (i.e. $-1 \le \theta_i \le +1$ for all i). Associated parameters, before normalization, typically belong to an hyper-rectangle.

4.1. *Modal design of an LFT gain*

The first issue is to design a gain-scheduled LFT controller, i.e. a static LFT gain $K(\theta)$ placing closed loop poles for the open loop plant model $G(s,\theta)$, using the technique of [33]. All θ-dependent quantities in the following Lemma are supposed to be under an LFT form.

Lemma 4.1 [33] Let ($A(\theta), B(\theta), C(\theta), D(\theta)$) be a state-space representation of $G(s,\theta)$. Assume that m closed loop eigenvalues $\lambda_i(\theta)$ are to be placed. Let a static output feedback gain $K(\theta)$ satisfy the equality:

$$K(\theta)\begin{bmatrix} C(\theta) & D(\theta) \end{bmatrix}\begin{bmatrix} V(\theta) \\ W(\theta) \end{bmatrix} = W(\theta) \tag{16}$$

where $V(\theta) = [v_1(\theta) \quad \ldots \quad v_m(\theta)]$ and $W(\theta) = [w_1(\theta) \quad \ldots \quad w_m(\theta)]$. Each pair of vectors $v_i(\theta)$ and $w_i(\theta)$ must satisfy:

$$\begin{bmatrix} A(\theta) - \lambda_i(\theta)I & B(\theta) \end{bmatrix}\begin{bmatrix} v_i(\theta) \\ w_i(\theta) \end{bmatrix} = 0 \tag{17}$$

Then, the interconnection of $G(s,\theta)$ with $K(\theta)$ has m placed eigenvalues $\lambda_i(\theta)$.

In practice, closed loop eigenvalues $\lambda_i(\theta)$ are chosen first. Then, eigenvectors $v_i(\theta)$ and associated vectors $w_i(\theta)$ are computed so as to satisfy (17). The feedback gain $K(\theta)$ is computed using (16).

Remark 1: Equalities (16,17) must be exactly satisfied, so that the number of poles to be placed must be less or equal to the number of plant outputs. Thus, only the dominant plant poles are placed in practice, and the position of the closed loop secondary poles is checked a posteriori.

Remark 2: this modal method not only places the closed loop eigenvalues (i.e. the closed loop poles), but also the associated eigenvectors, i.e. this is an eigenstructure assignment method. The robustness of the controller strongly depends on the choice of the placed eigenstructure. Robust controllers can be expected when respecting as much as possible the open loop dynamics, for instance by projecting the open loop eigenvectors, i.e. by choosing closed loop eigenvectors which are as close as possible to the open loop ones.

In the same way, in §5.2, the closed loop frequency of the short period mode, which is the main mode of the aerodynamic model, will be chosen to be the same as the open loop one, just the damping ratio will be improved. This enables to design flight controllers which reveal (very) robust to parametric uncertainties.

Using the routine *fb_sched.m* of the LFR Toolbox [32], the technique provides a static feedback gain:

$$K(\theta) = F_l(\mathcal{K}, \overline{\Delta}) = K_{11} + K_{12}\,\overline{\Delta}\,(I - K_{22}\overline{\Delta})^{-1} K_{21} \tag{18}$$

where $\overline{\Delta} = \text{diag}(\theta_i\, I_{k_i})$ and $\mathcal{K} = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}$.

After the design, a key point is to check the well-posedness radius of the LFT gain, i.e. the maximal size k for which the matrix $I - K_{22}\overline{\Delta}$ is invertible for all $\theta \in k\,H$ (i.e. $-k \le \theta_i \le +k$ for all i). Using standard Matlab routines, a guaranteed value of this radius, i.e. a lower bound, can easily be obtained as the inverse of a μ upper bound of K_{22}, where μ is the structured singular value [9]. Since the vector θ of plant parameters is assumed to belong to the unit hypercube H, the well-posedness radius must be greater than 1.

4.2. Robustness analysis

Consider a closed loop LFT model $F_l[M(s),\Delta]$, where Δ contains normalized uncertain parameters θ_i. The aim of the following Lemma is to compute a guaranteed value of the worst-case H_∞ norm / L_2 gain of $F_l[M(s),\Delta]$, i.e. of its maximal value over $\theta \in H$.

Lemma 4.2 [10] The issue is to compute a guaranteed robust performance level γ, for which:

$$\| F_l[M(s),\Delta] \|_{iL_2} \le \gamma \quad \forall \theta \in H \tag{19}$$

Let $D = D^* > 0$ and $G = G^*$ be scaling matrices satisfying $D\Delta = \Delta D$ and $G\Delta = \Delta^* G$ for all structured Δ. Let $P(s) = diag(I_{n_1}/\gamma, I_{n_2})M(s)$, where n_1 (resp. n_2) is the size of the I/O of $F_l[M(s),\Delta]$ (resp. of Δ).

A sufficient condition for (19) to hold is that there exist scaling matrices D and G satisfying $\forall \omega$:

$$P^*(j\omega) D P(j\omega) + j[G P(j\omega) - P^*(j\omega) G] \le D \tag{20}$$

When parameters θ_i are supposed to be time invariant, complex frequency-dependent scaling matrices $D(\omega)$ and $G(\omega)$ are handled, and inequality (20) can be independently solved at each frequency. This is a skew μ analysis problem [10] for which efficient computational tools exist, and especially the routine *mu_margin.m* of the System Modeling, Analysis and Control (SMAC) Toolbox developed by Onera in Matlab-Simulink (available at http://w3.onera.fr/smac). Conversely, when parameters θ_i are supposed to be time varying, without bound on the rate of variation, real frequency-independent scaling matrices D and G are handled, and inequality (20) must be simultaneously solved on the frequency continuum using IQC tools, see e.g. [14,23] and the SMAC Toolbox. Last note that a bound on the rate of variation of the parameters can be introduced, using a generalization of the D,G scalings, in the framework of IQC analysis [35].

5. DESIGN AND VALIDATION OF A GAIN-SCHEDULED FLIGHT CONTROLLER

To reconfigure the inner control loop of the A/C in faulty or unexpected situations, a gain-scheduled controller depending on the control and stability derivatives is designed in §5.1 and §5.2, and validated on the nonlinear simulator in §5.3. Its robustness to an estimation error is checked in §5.4. In §5.5, a robust controller is deduced from the gain scheduled one, by introducing mean or worst-case values of the scheduling parameters.

5.1. State space design model

For the RECONFIGURE project, a set of 214 longitudinal linearized models has been provided by Airbus, covering the flight domain with respect to speed, altitude, mass and center of gravity position variations. This set is used to design and to validate the gain-scheduled controller. Owing to the range of variation of the stability derivatives ($Z_\alpha, Z_q, \dots$) over the set of 214 models, the aerodynamic model (3) can be simplified by considering only the main stability derivatives:

$$\begin{cases} \dot\alpha = Z_\alpha \alpha + q \\ \dot q = M_\alpha \alpha + M_q q + M_\delta \delta_m \\ N_z = N_\alpha \alpha \end{cases} \tag{21}$$

i.e. $Z_q \approx 1$ while $Z_\delta \approx 0, N_q \approx 0$ and $N_\delta \approx 0$. Indeed, in practice, when validating the flight controller, designed using model (21), on the full 2 state aerodynamic model (3), closed loop performance appears essentially to be the same using the simplified model (21) and the full model (3).
By normalizing M_δ and N_α to unity, the design model can be further simplified to become:

$$\begin{cases} \dot\alpha = Z_\alpha \alpha + q \\ \dot q = M_\alpha \alpha + M_q q + u \\ z = \alpha \end{cases} \tag{22}$$

After designing the gain-scheduled controller:

$$u = K(M_\alpha, M_q, Z_\alpha)[z \quad \int z dt \quad q]^T \tag{23}$$

The true feedback controller will be:

$$\delta_m = \frac{K(M_\alpha, M_q, Z_\alpha)}{M_\delta} \begin{bmatrix} 1/N_\alpha & 0 & 0 \\ 0 & 1/N_\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} N_z - N_{Zc} \\ \int (N_z - N_{Zc}) dt \\ q \end{bmatrix} \tag{24}$$

More precisely, when considering the architecture (4) of the flight controller, the feedback gains K_{Nz}, K_q, K_I can be deduced from (24). The feedforward term L will be retuned in §5.3.

5.2. Controller design

Following [33] an open loop LFT model under the form (22) is built using the LFR Toolbox [32], with input u, outputs z and q, states α and q. The parameters of the LFT model are Z_α, M_α, M_q, whose ranges of variation correspond to the set of 214 linearized models. Actuator and sensor models are added to this LFT model, as well as a 2nd order Padé model (used to represent a time-delay at the plant input) and an integrator on the z output, so that the LFT design model has 3 outputs $z, q, \int z dt$. As a result, the 3 main closed loop poles can be placed by the LFT feedback controller $K(M_\alpha, M_q, Z_\alpha)$, namely:

- The integrator pole at a specified location λ, the same for all models, i.e. for all values of M_α, M_q, Z_α.
- The short period (SP) mode as a complex mode with damping ratio 0.8 and frequency ω, where ω is chosen as the frequency of the open loop SP mode, i.e. the poles of the 2 states linearized model (22).

These closed loop poles are chosen to satisfy classical design specifications on the step response to a reference acceleration N_{Zc}, namely maximal settling time on N_z and maximal overshoots on N_z (10%) and q (30%). The criteria chosen for evaluation rely on good tracking performance of the acceleration with homogeneous responses.

Once the design LFT model is computed, depending on the free parameters M_α, M_q, Z_α, and after choosing the closed loop poles with an additional free parameter ω, the LFT feedback controller $K(M_\alpha, M_q, Z_\alpha, \omega)$ is designed using Lemma 4.1. When implementing the controller on-line, the estimated values of M_α, M_q, Z_α will be used, not only as scheduling parameters, but also to compute ω as the SP mode frequency of the estimated 2 states model (22). A minimal value is introduced for ω, to speed up the closed loop SP mode when it is too slow with respect to the closed loop integrator pole.

This LFT controller, modified following (24), is validated over the 214 linearized models under the form (3). Although the value of the LFT controller at a flight point is computed using the true values of the stability derivatives, small changes of the closed loop integrator and SP modes are expected, since the secondary stability derivatives are no more neglected. Moreover, the closed loop actuator, sensor and Padé model poles need to be checked since they were not explicitly placed in the design procedure. Fig. 2 presents the closed loop poles for all 214 models, when zooming on the integrator pole and SP mode. The result is quite satisfactory, as well as the step responses in Fig. 3. Note that the q output is normalized by its steady state value at each flight point.

5.3. Nonlinear validation

The LFT controller is validated on the nonlinear simulator for the 214 flight points. The testing maneuver usually selected for industrial validation involves joystick doublet-type signals. Both the autopilot and the auto-throttle are switched off. Time domain responses differ from those obtained with the linearized models on Fig. 3, primarily because of the additional slow longitudinal states V, θ, z in the nonlinear aerodynamic model, so that the feedforward term L of the flight controller (4) needs to be retuned.

Remember K_I is the feedback gain corresponding to the integrator on the N_z output. The linearized responses of Fig. 3 were obtained with the classical choice $L = -K_I$. Nevertheless, to obtain a satisfactory performance on the nonlinear simulator, $L = -A(mass, CoG)\, K_I$ needs to be chosen, where the multiplicative factor A only depends on the mass and center of gravity position to ease the implementation. Choosing A results from a trade-off between the fastness of the closed loop and its overshoots: increasing A increases the fastness, at the price of higher overshoots on outputs N_z and q.

Nonlinear responses, which are satisfactory, are displayed in red color in Fig. 4 for a given value of mass and center of gravity: this "C1 case" includes 25 flight points corresponding to variations of airspeed and altitude. The airplane needs to be trimmed, before applying the first filtered step on the reference acceleration input (in blue color).

5.4. Linear validation via μ and IQC analysis

The issue is to check the robustness of the LFT controller in the presence of estimation errors on the 5 main derivatives $Z_\alpha, M_\alpha, M_q, M_\delta, N_\alpha$ at each of the 214 available flight points. To this aim, an error is introduced in each stability derivative at a given flight point, e.g.:

$$Z_\alpha = Z_\alpha^0(1 + 0.1\delta_1) \quad (25)$$

where Z_α^0 is the value of Z_α at this flight point. When the normalized uncertainty δ_1 varies between -1 and 1, the corresponding variation of Z_α is $\pm 10\%$. When introducing an uncertainty on each main stability derivative in the LFT controller, a new LFT controller is deduced using the LFR Toolbox [32], which depends on $(\delta_i)_{i\in[1,5]}$. More precisely, in the structured model perturbation, the normalized estimation errors on Z_α, M_α, M_q are repeated 3 times on the diagonal, 1 time for M_δ, and 2 times for N_α. Two approximations are made to get this rather reasonable complexity: ■ the frequency ω in the initial LFT controller is fixed to its nominal value, i.e. it is computed using the true values of the stability derivatives at the flight point; ■ in the feedforward term $L = -AK_I$, K_I is also fixed to its nominal value.

At a given flight point, the LFT closed loop is obtained by connecting the new LFT controller with the linearized aerodynamic model (3), augmented with actuator, sensor and Padé models, and with an integrator on $N_z - N_{zc}$. The standard interconnection structure $M(s) - \Delta$ is obtained either for robust stability analysis in the Left Half Plane, or for robust performance analysis of the closed loop transfer function between N_{zc} and the tracking error $N_z - N_{zc}$. A first order frequency domain template is applied to $N_z - N_{zc}$ to ensure a minimal steady-state error (1%), a maximal frequency domain overshoot (6 dB), and a minimal closed loop bandwidth (0.5 rad/s): see the plot in Fig. 5, where the template is in green color and the nominal (without uncertainty) transfer functions between N_{zc} and $N_z - N_{zc}$ are in red color.

A time invariant estimation error is supposed at first. μ analysis gives a range of variation between 34.2% and 49.7% over the set of 214 flight points for the LTI (Linear Time Invariant) robust stability margin in the Left Half Plane, with an average value of 38.8%. For instance, a margin of 38.8% means that closed loop stability is guaranteed despite a simultaneous uncertainty of ±38.8% on all five main stability derivatives, which appears very satisfactory. Note that the result does not depend on the weight of 0.1, which was just introduced to normalize the uncertainty in (25).

The results obtained with a time-varying estimation error reveal not so different with a range of variation between 23.0% and 35.8% for the LTV (Linear Time Varying) robustness margin (average value of 31.5%). These results are obtained with IQC analysis and constant scaling, i.e. without bounding the variation rate of the uncertainties. If the gap between the LTI and LTV robustness margins was larger, it would be worth introducing a bounded rate of variation for the estimation error, to explore the intermediate cases, but this appears useless here.

The robust performance margins are computed in the same way, leading to a range of variation between 5.8% and 21.0% for the LTI margin (average value of 14.2%), and a range between 4.9% and 18.2% for the LTV margin (average value of 12.3%). Again, the gap between LTI and LTV robustness margins is reasonable and the results are satisfactory, when considering that performance is much more constraining than stability.

5.5. Design of a robust controller

To apply the proposed adaptive strategy, a robust controller is needed which only depends on mass and center of gravity position, i.e. it should not depend on airspeed, altitude nor on stability derivatives. For the sake of simplicity, only the C1 case is considered here, corresponding to 25 flight points. This robust controller can be deduced from the gain-scheduled one of §5.1 and §5.2 by introducing mean or worst-case values of the 5 scheduling parameters.

More precisely, 3 possible values are considered for each of the 5 parameters, namely the minimum, mean or maximum values of the derivative over the 25 linearized aerodynamic models: 3^5=243 combinations are involved. The criterion is the minimal value of the degree of stability over the 25 linearized closed loops. Remember that the degree of stability is defined as the opposite of the maximal value of the real part of the closed loop poles. The maximal value of the criterion over the 243 combinations is selected, leading to choose the mean value for M_α, the minimum value for N_α, and the maximum values for M_q, M_δ, Z_α. Fig. 6 presents the closed loop step responses for the gain-scheduled and robust controllers. As expected, the performance of the latter is degraded, see the overshoots on N_z and q. Nevertheless, this controller ensures minimal performance properties.

Remark: in the same way, it would be necessary to design a robust fixed controller for each value of mass and center of gravity position, so that when considering all the 214 available flight points, the back up controller would be scheduled/interpolated as a function of mass and center of gravity position. The back up controller was chosen not to depend on airspeed, because the airspeed measurement, which may be missing due to a sensor fault, is less reliable than the measurement/estimation of mass and center of gravity position. Conversely, we have chosen not to design a robust fixed controller over all 214 available flight points because its robust performance properties would be too bad: when designing the robust fixed controller, it is worth reducing the range of variation of the model parameters as much as possible, to obtain robust performance properties which are as satisfactory as possible.

6. PRACTICAL IMPLEMENTATION ASPECTS

To be possibly implemented onboard, the frequency domain estimator and the gain-scheduled controller should comply with the stringent constraints resulting from the present industrial coding process applicable to Airbus aircraft. Accordingly, the algorithms have been firstly encoded in full Simulink®, i.e. without any call to Mex nor Matlab® functions. In a second step, The Simulink® schemes were encoded by using the existing symbols of the Flight Control Computer (FCC) library provided by Airbus to be compliant with the SCADE process (Safety Critical Application Development Environment). After encoding the algorithm with FCC operators, the resulting computational burden evaluated by Airbus during the RECONFIGURE project is about 1.5ms, which is quite acceptable and could be further improved by updating the estimates at a lower rate. As this library only includes basic operators (logical and mathematical type), the second step has required higher level modules to be developed for coding more complex functionalities, e.g. required by matrix calculus. This was especially true for the FD estimation algorithm, which involves matrix operations with complex numbers.

As regards the gain-scheduled controller, it is noteworthy that the usual way of computing (off-line) the value of the LFT controller is not suitable for on board implementations, as the computational complexity of this procedure is generally high, and would especially require a matrix inversion of dimension 27 in the present case. To bypass this difficulty several alternatives do exist [34], for instance by resorting to a recursive method which avoids this inversion by using a fixed point technique (possibly with a relaxed sampling rate). However, the complexity of the LFT controller designed in §5.2 is rather high: even if each control or stability derivative is repeated 3 times, which is reasonable, the frequency of the SP mode is repeated 16 times, so that the on-line computational burden of the technique proposed by [34] would be too high.

That is why we have preferred another option, consisting in developing off-line a surrogate model of the LFT gains, which could be used afterwards for on-line computations, and which could also be easily coded in terms of the basic FCC library operators by using only basic scalar operations available for onboard computers. The characteristics of these surrogate models and the principles of the techniques permitting rational-type representations to be obtained are detailed in [43]. To build this surrogate model, the tool named *koala* was used, which is part of the APRICOT library (**A**pproximation of **P**olynomial and **R**ational-type for **I**ndeterminate **C**oefficients via **O**ptimization **T**ools), included into the SMAC Toolbox (available at http://w3.onera.fr/smac/). The tools included in APRICOT permit sparse rational models to be generated, which lead to simple yet accurate Linear Fractional Representations. The computational complexity of the (rational) modeling is thus consistent with the FCC library coding requirements, as it only involves a set of scalar additions and multiplications.

In the present case, the 3 control gains $K_{Nz}, K_q,\ K_I$ need to be modeled in terms of the 3 explanatory variables M_α, M_q, Z_α, the SP frequency being directly inferred from the open loop eigenvalue of the simplified state-space model (22). To get a set of reference data from which the approximated rational function can be derived, a 3D gridding is firstly achieved by computing a set of LFT exact values from regularly spaced samples in the input space $M_\alpha \times M_q \times Z_\alpha$. Overestimated bounds of the aerodynamic coefficients $\{M_\alpha \in [M_\alpha^{\min}, M_\alpha^{\max}] \times M_q \in [M_q^{\min}, M_q^{\max}] \times Z_\alpha \in [Z_\alpha^{\min}, Z_\alpha^{\max}]\}$ were selected to define the expected cubic domain. Fig. 7 illustrates the result of this process for the 3D coefficient $K_q(M_\alpha, M_q, Z_\alpha)$. The surrogate model used to approximate the 3 control gains simultaneously is computationally efficient and achieves a very low acceptable error in the whole selected domain.

7. TIME-DOMAIN RESULTS OF THE RECONFIGURE BENCHMARK

7.1. *Adaptive control strategy along a trajectory*

The issue is to deal with the time-varying aspect of the parameters to be estimated, despite an unsatisfactory closed loop excitation. This concerns the estimator but also the gain-scheduled controller, since the point is also to decide when the values of the scheduling parameters should be updated, knowing that a measure of their reliability is available as well. At the start, when controlling the A/C at a trim point, "robust" values of the stability derivatives are chosen as scheduling parameters, i.e. those corresponding to the robust controller of §5.5, to ensure minimal stability and performance closed loop properties. As early as the estimator provides (reliable) estimates, these are injected in the gain-scheduled controller. When moving from a flight point to another, time-varying parameters can be tracked either by benefiting from the forgetting factors, whose values may be adjusted along the trajectory, or by resetting the estimator whenever it is necessary [20]. Estimates may also be frozen if the I/O data content is too poor and hence the estimation errors too high.

Regarding the adaptive control strategy, a first solution is to apply the Certainty Equivalence Principle (CEP), i.e. the estimates are continuously introduced in the gain-scheduled controller whatever their reliability is. Another solution is to switch to the robust controller whenever the estimates are not reliable enough. Now, the CEP is bypassed and the adjustment of the scheduling parameters is discontinuous, using a low pass filter to avoid jumps in the scheduling values. This 2nd solution permits minimal stability and performance properties to be guaranteed for the adaptive closed loop, a crucial point if the flight controller was to be certified.

Conversely, the transient response of the adaptive closed loop may be shorter with the 1st solution, as it may be useless to wait for fully reliable estimates thanks to the robustness of the controller with respect to estimation errors: a proper time domain response may be obtained despite less reliable estimates. The two solutions are illustrated in the following, and essentially produce the same type of results. Indeed, estimates are typically unreliable when closed loop excitation is poor, so that the difference between the closed loop responses of the robust and gain-scheduled controllers is not significant.

7.2. *Flight scenario*

To evaluate the proposed adaptive scheme in the framework of RECONFIGURE, the simulated scenario is a 380 seconds realistic flight profile including 4 successive stages:

1/ A first steady flight at $(z = 12500\,\text{ft}, V_c = 240\,\text{kts})$.

2/ A climb up to $z = 20000\,\text{ft}$ for $t \in [65, 215]$, with $V_c = 240\,\text{kts}$.

3/ An acceleration up to $V_c = 335\,\text{kts}$ for $t \in [215, 300]$, with $z = 20000\,\text{ft}$.

4/ A final steady condition at $(z = 20000\,\text{ft}, V_c = 335\,\text{kts})$.

The autopilot is used to regulate the altitude z and conventional airspeed V_c at their reference values. See Fig. 8 for the time history of altitude and true airspeed (which differs from the conventional airspeed).

The testing maneuvers selected by Airbus for the industrial validation of the controllers involve joystick doublet-type signals. So, to check the performance of the adaptive flight controller, these usual stick input signals are applied to produce a reference input signal N_{zc} during the 1st and 4th parts of the flight, with autopilot and auto-throttle switched off (which explains on Fig. 8 why the altitude is no more regulated at the beginning and at the end of the flight). See Fig. 9 for the time plot of the acceleration (more precisely the load factor in g) N_z, the rotational rate q and the elevator (i.e. the A/C) input δ_m. The mass and center of gravity correspond to the C1 case.

7.3. Adaptive control results

The performance of the adaptive controller can essentially be seen on the plots of the load factors in Figs. 9 and 10 (N_z vs N_{zc}). Fig. 10 focuses on the responses at the beginning and at the end of the flight. The blue line corresponds to the reference input N_{zc}, the black and red ones to the N_z signals. As soon as reliable estimates are available for scheduling, adaptation enables to recover a much better performance, i.e. satisfactory overshoot and settling time.

Fig. 9 presents the results along the whole trajectory. As expected, closed loop excitation is globally poor during the $2^{nd}/3^{rd}$ stages corresponding to altitude/airspeed changes: see the reference input signal N_{zc} in blue color on the N_z plot. As a result, during these 2 stages, either the robust controller, or the gain-scheduled one are used depending on the reliability of the estimates and on the adaptive control strategy chosen (CEP or switch). The black curves of Figs. 9 and 10 correspond to the 1^{st} solution (CEP, i.e. the parameter estimates are used as scheduling parameters whatever their reliability), whereas the red ones correspond to the 2^{nd} solution (switch for $t \in [194, 343]$ to the robust controller, also active for $t \in [0,18]$ in both solutions, because the parameter estimates are unreliable or unavailable inside these time intervals). Regardless of the flight controller, the outer loop responses of the autopilot are satisfactory, see the plots of true airspeed and altitude in Fig. 8. Due to the poor excitation, it is difficult to judge the quality of the inner loop responses on Fig. 9, but they seem suitable.

Fig. 10 focuses on the steady flights at the beginning and at the end of the simulation. In the upper plot, the 2 solutions (black/red lines) match exactly because the robust controller is used at the beginning in both cases: the estimator is initialized so that some time is needed before an estimate is available. In the bottom plot, the transient response of the 1^{st} CEP solution (in black) appears better, noting that the overshoot and the settling time for the 2^{nd} solution (in red) are those of the robust controller for $t < 343$: once reliable estimates are available, the overshoot/settling time become satisfactory. *Hence, the 2^{nd} solution guarantees minimal stability and performance properties for the adaptive closed loop at the price of a potentially less proper transient response*.

7.4. Results of the recursive estimation algorithm

The reader is referred to [20] for a comprehensive description of the estimation results. Fig. 11 briefly illustrates the behavior of the recursive FD scheme (tuned with $M = 43$ frequency samples equally spaced in the range 0.1-0.52 Hz), through the M_δ estimate got during the simulation. The parameter uncertainty (3σ) is plotted in blue dashed lines and results in freezing the estimate (solid blue line) whenever it is too high. The green curve gives a rough idea of the reference value drawn from the closest LTI model (just one of 25 for the C1 case). Due to a loose mesh of the linearized models in the flight domain, these values are only plotted to give a trend (the true linearized values being unavailable). Owing to the simulated scenario, the variations that are tracked result either from a change in the flight condition or, at the start of the run, from a misknowledge of the initial parameter values. Both of them demonstrate the interest of the adaptation to cope with modeling uncertainties. It is worth noting that the continuous variation of z/V is monitored by the estimation algorithm during level changes, and that the parameters accuracy can also be used to trigger a freeze of the estimates whenever the uncertainty becomes too high (not enough excitation or rapid changes in the flight condition). In addition, the forgetting mechanism of (10) is only triggered in Fig. 11 when significant changes are detected in z/V, providing a way to adapt more rapidly to varying conditions.

It should also be pointed out that with such a poor excitation (during the $2^{nd}/3^{rd}$ parts of the scenario), some parameters can't be estimated properly. Hence to prevent from correlation issues or ill-conditioning, a tuned-down modeling or some regularization is required. Here, a priori knowledge via the Bayesian formulation of (8) is introduced; hence, the trends indicated by the (green) reference values cannot be instantly mimicked by the estimates, and instead these ones are drawn towards the prior valid estimates or towards average reference values used as a priori knowledge by the estimation algorithm (e.g., mean values for a given altitude). As soon as sufficient excitation is available, and the accuracy is good enough (some time is needed to store information through the recursive updates of the FT), the estimates are freed again and converge rapidly to suitable values. Thanks to the noise filtering outside the selected bandwidth, the FD algorithm does not require much time to accumulate enough information and to recover proper estimates. In case of continuous estimation (CEP), the convergence delay is even shorter as the estimates may be reliable enough although the errors are not fully satisfying.

8. CONCLUSION

The contribution of this paper is twofold. First, an adaptive scheme is presented, which deeply differs from classical indirect adaptive control schemes: these use a time domain (Least Squares or gradient) estimator, as well as a (SISO or square MIMO) transfer function to describe the plant. The controller is designed on-line. On the contrary, in our scheme, a non-square MIMO plant model can be used under a state-space form. Moreover, to minimize the computational time, one uses a frequency domain plant model estimator and a gain-scheduled controller, designed off-line.

As a second contribution, the practical interest of our scheme is illustrated by the design of an adaptive flight controller, with a computational time and a complexity that comply with the requirements of embedded A/C implementation. The indirect adaptive control strategy combines an efficient recursive frequency domain estimator with an LFT controller scheduled with respect to the parameters to be estimated. Robustness to transient and asymptotic estimation errors was proved using μ and IQC analysis techniques, and demonstrated via the industrial benchmark. Nonlinear simulations also show the capability to control the airplane along a trajectory, a more difficult problem than controlling about a steady flight point since the time-varying parameters have to be estimated despite poor closed loop excitation.

These results are very promising as regards a possible application to civil aircraft. The next step would be to further validate the adaptive flight control law. Such a validation seems very difficult using analytical methods, due to the complexity of the frequency domain model estimator and gain-scheduled LFT controller. For example, finding a Lyapunov function for the adaptive closed loop, as classically done for simpler adaptive control schemes, appears very unlikely. In this context, different complementary solutions have been or can be proposed:

1/ Checking the robustness of the LFT controller to an estimation error on the scheduling parameters (§5.4): the obtained robustness margins appear especially relevant, even if the model estimator is not accounted for in the analysis. Stability and performance properties of the adaptive closed loop can be guaranteed despite a maximal allowable amount of the estimation error, provided by the estimator.

2/ Using the available measure of the confidence on the parameter estimates, to switch to a robust controller whenever the estimates are not reliable enough (§7): this enables to guarantee minimal stability and performance properties for the adaptive closed loop, even when excitation is poor.

3/ Using extensive sets of nonlinear simulations to check the robust stability and performance properties of the adaptive flight controller: it would be worth exploring this perspective of an industrial validation, noting that a practical approach for the worst-case validation of a similar indirect adaptive control scheme was already investigated by Airbus and Onera in [41].

9. REFERENCES

[1] Aouf N., Bates D.G, Postlethwaite I., Boulet B., "Scheduling schemes for an integrated flight and propulsion control system", *Control Engineering Practice*, 10:685-696, 2002.

[2] Astrom K.J., Wittenmark B., *Adaptive Control*, 2nd edition, Addison-Wesley, Reading, 1995.

[3] Barkana I., "Classical and simple adaptive control for non-minimum phase autopilot design", *Journal of Guidance, Control and Dynamics*, 28(4):631-638, 2005.

[4] Basappa, Jategaonkar R.V., "Evaluation of recursive methods for aircraft parameter estimation", *AIAA AFM*, paper 2004-5063, Providence, USA, 2004.

[5] Beard R.W., Knoebel N., Cao C., Hovakimyan N., Matthews J., "An L1 adaptive pitch controller for miniature air vehicles", *AIAA GNC*, 2006.

[6] Biannic J.M., Roos C., Knauf A., "Design and robustness analysis of fighter aircraft control laws", *European Journal of Control - special issue on LMI techniques in Control*, 12(1):71-85, January 2006.

[7] Chandler P.R., Pachter M., Mears M., "System identification for adaptive and reconfigurable control", *Journal of Guidance, Control and Dynamics*, 18(3):516–524, 1995.

[8] Ducard G., Geering H.P., "A reconfigurable flight control system based on the EMMAE method", *ACC*, Minneapolis, USA, 2006.

[9] Fan M.K.H., Tits A.L., Doyle J.C., "Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics", *IEEE Transactions on Automatic Control*, 36(1):25–38, 1991.

[10] Ferreres G., *A practical approach to robustness analysis with aeronautical applications*, Springer Verlag, 1999.

[11] Ferreres G., Antoinette P., "Robust gain scheduling techniques for adaptive control", *ECC*, Budapest, Hungary, 2009.

[12] Ferreres G., Antoinette P., "Adaptive control using robustness and gain-scheduling", *Journal Européen des Systèmes Automatisés*, 45(4-6):279–298, 2011.

[13] Ferreres G., Hardier G., Seren C., "Adaptive control of a civil aircraft through on-line parameter estimation", *Systol'16*, Barcelona, Spain, 2016.

[14] Ferreres G., Roos C., "Robust feedforward design in the presence of LTI/LTV uncertainties", *International Journal of Robust and Nonlinear Control*, 17(14):1278–1293, 2007.

[15] Goupil P. *et al.*, "AIRBUS efforts towards advanced real-time fault diagnosis and fault tolerant control", *19th IFAC World Congress*, Cape Town, South Africa, 2014.

[16] Han Y., Kim Y., Kim H., "Frequency and time domain online parameter estimation for reconfigurable flight control system", *AIAA Infotech Aerospace*, Seattle, USA, 2009.

[17] Hanlon P.D., Maybeck P.S., "Multiple-model adaptive estimation using a residual correlation KF bank", *IEEE Transactions on aerospace and electronic systems*, 36(2):393–406, 2000.

[18] Hardier G., "An extended U-D algorithm with multiple forgetting factors for RLS estimation of model parameters", *Safeprocess*, Paris, France, 2015.

[19] Hardier G., Bucharles A., "On-line parameter identification for in-flight aircraft monitoring", *27th ICAS Congress*, Nice, France, 2010.

[20] Hardier G., Ferreres G., Seren C., "A recursive estimation algorithm to track aircraft model parameters", *Systol'16*, Barcelona, Spain, 2016.

[21] Jategaonkar R.V., *Flight vehicle system identification: a time domain methodology*, AIAA editions, Vol. 216, Reston, USA, 2006.

[22] Jiang J., Zhang Y., “A revisit to block and recursive least squares for parameter estimation”, *Computers & Electrical Engineering*, 30, 2004.

[23] Kao C.Y. *et al.*, “A Matlab toolbox for robustness analysis”, *CACSD*, 2004.

[24] Kerr M.L. et al., “RECONFIGURE FP7 project: preliminary results and contributions”, *Systol'16*, Barcelona, Spain, 2016.

[25] Kim S.H., Kim Y.S., Song C., "A robust adaptive nonlinear control approach to missile autopilot design", *Control Engineering Practice*, 12:149-154, 2004.

[26] Klein V., Morelli E., *Aircraft system identification: theory and practice*, AIAA Education Series, Reston, USA, 2006.

[27] Landau I.D. *et al.*, *Adaptive control: algorithms, analysis and applications*, Springer Verlag, 2011.

[28] Larsson R., Enqvist M., “Sequential aerodynamic model parameter identification”. *16th IFAC SYSID*, Brussels, Belgium, 2012.

[29] Ljung L., Gunnarsson S., “Adaptation and tracking in system identification: a survey”, *Automatica*, 26(1):7–21, 1990.

[30] Ljung L., Soderstrom T., *Theory and practice of recursive identification*, MIT Press, Cambridge, 1983.

[31] Lombaerts T.J.J., Huisman H.O., Chu Q.P., Mulder J.A., Joosten D.A., "Nonlinear reconfiguring flight control based on online physical model identification", *Journal of Guidance, Control and Dynamics*, 32(3):727-748, May-June 2009.

[32] Magni J.F., “Linear Fractional Representation Toolbox”, available at http://w3.onera.fr/smac/lfrt, 2004-2008.

[33] Magni J.F., “An LFT approach to robust gain scheduling”, *44th IEEE CDC*, Seville, Spain, 2005.

[34] Magni J.F., Knauf A., Döll C., “Practical implementation of LFT-scheduled feedback gains”, *6th Asian Control Conference*, Bali, Indonesia, 2006.

[35] Megretski A., Rantzer A., “System analysis via Integral Quadratic Constraints”, *IEEE Transactions on Automatic Control*, 42(6):819–830, 1997.

[36] Morelli E., “Real-time parameter estimation in the frequency domain”, *Journal of Guidance, Control and Dynamics*, 23(5), 2000.

[37] Morelli E., “Practical aspects of the EE method for aircraft parameter estimation”, *AIAA AFM*, no 2006-6144, Keystone, USA, 2006.

[38] Morelli E., Smith M.S., "Real-time dynamic modeling – Data information requirements and flight test results", *Journal of Aircraft*, 46(6):1894-1905, 2009.

[39] Nichols R.A., Reichert R.T., Rugh W.J., "Gain scheduling for Hinfinity controllers: a flight control example", *IEEE Transactions on Control Systems Technology*, 1(2):69-79, June 1993.

[40] Oudin S. *et al.*, “Adaptive LFT control of a transport aircraft on the lateral axis”, *IEEE MSC*, Dubrovnik, Croatia, 2012.

[41] Oudin S. *et al.*, “Design and worst-case validation of a longitudinal adaptive flight control law: a practical approach”, *AIAA GNC*, Boston, USA, 2013.

[42] Rugh W.J., Schamma J.S., “Research on gain scheduling”, *Automatica*, 36:1401-1425, 2000.

[43] Seren C., Hardier G., Ezerzere P., “On-line Estimation of Longitudinal Flight Parameters”, *SAE AeroTech Congress and Exhibition*, Toulouse, France, 2011.

[44] Shin J.Y., Wu N.E., Belcastro C., “Adaptive Linear Parameter Varying control synthesis for actuator failure”, *Journal of Guidance, Control and Dynamics*, 27(5):787-794, 2004.

[45] Siwakosit W., Snell S.A., Hess R.A., "Robust flight control design with handling qualities constraints using scheduled linear dynamic inversion and loop-shaping", *IEEE Transactions on Control Systems Technology*, 8(3):483-494, May 2000.

[46] Smith M.S. *et al.*, "Real-Time Stability and Control Derivative Extraction from F-15 Flight Data", NASA TM-2003-212027, 2003.

[47] Song Y., Campa G., Napolitano M., Seanor B., Perhinschi M.G., "Online parameter estimation techniques comparison within a fault tolerant flight control system", *Journal of Guidance, Control and Dynamics*, 25(3):528-537, May-June 2002.

[48] Tao G., *Adaptive control design and analysis*, John Wiley & Sons, Hoboken, USA, 2003.

[49] Wahi P., Raina R., Chowdhury F.N., "A survey of recent work in adaptive flight control", *33rd Southeastern Symposium on System Theory*, pages 7-11, March 2001.

[50] Wise K.A., Lavretsky E., "Robust and adaptive control of X-45A J-UCAS: a design trade study", *IFAC World Congress*, pp. 6555-6560, 2011.

[51] Wise K.A., Lavretsky E., Hovakimyan N., "Adaptive control of flight: theory, applications, and open problems", *ACC*, pp. 5966-5971, June 2006.

[52] Ye D., Yang G.H., "Adaptive fault-tolerant tracking control against actuator faults with application to flight control", *IEEE Transactions on Control Systems Technology*, 14(6):1088-1096, 2006.

[53] Yu X., Zhang Y., Liu Z. "Fault-Tolerant Flight Control Design with Explicit Consideration of Reconfiguration Transients," *Journal of Guidance, Control, and Dynamics*, 39 (3):556-563, March 2016.

[54] Yu X., Liu Z., Zhang Y. "Fault-Tolerant Flight Control with Finite-Time Adaptation under Actuator Stuck Failures," *IEEE Transactions on Control Systems Technology*, 25 (4):1431-1440, July 2017.

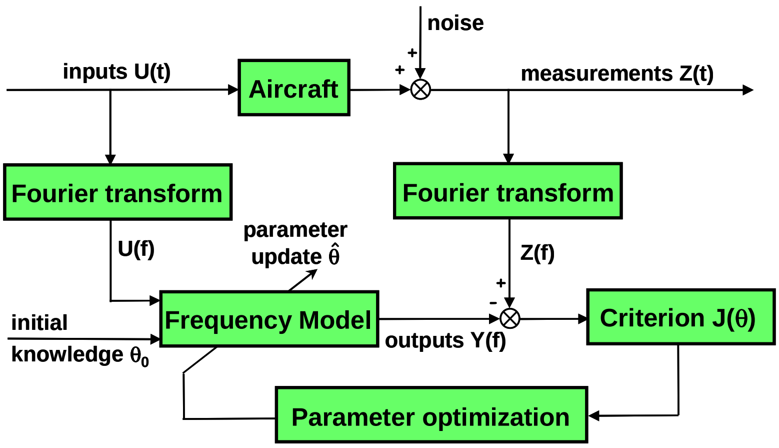

Figure 1: The FD estimation process

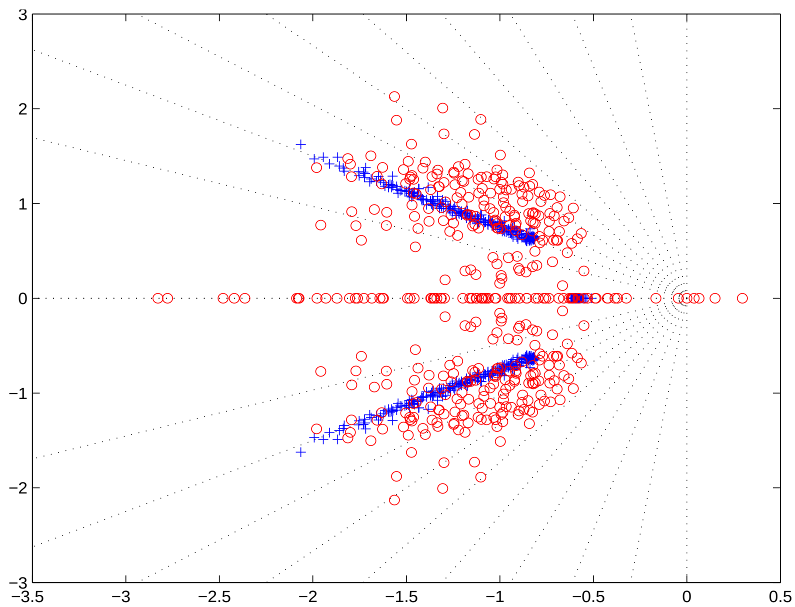

Figure 2: Open (red) and closed (blue) loop poles at the 214 flight points

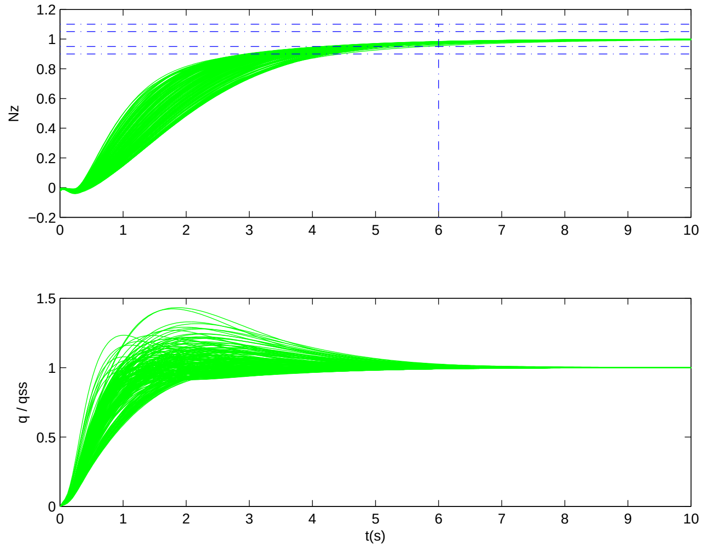

Figure 3: linear step responses at the 214 flight points

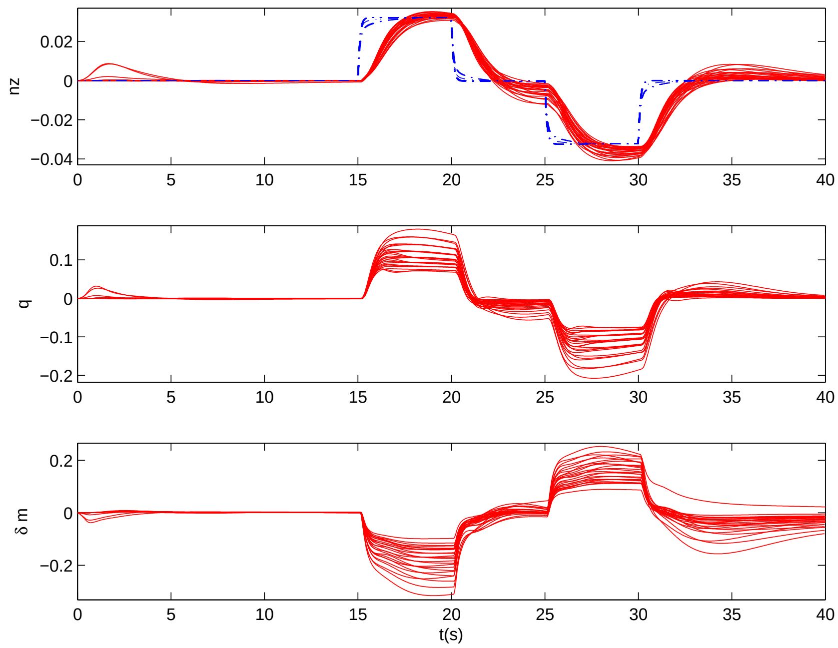

Figure 4: Nonlinear responses for the C1 case

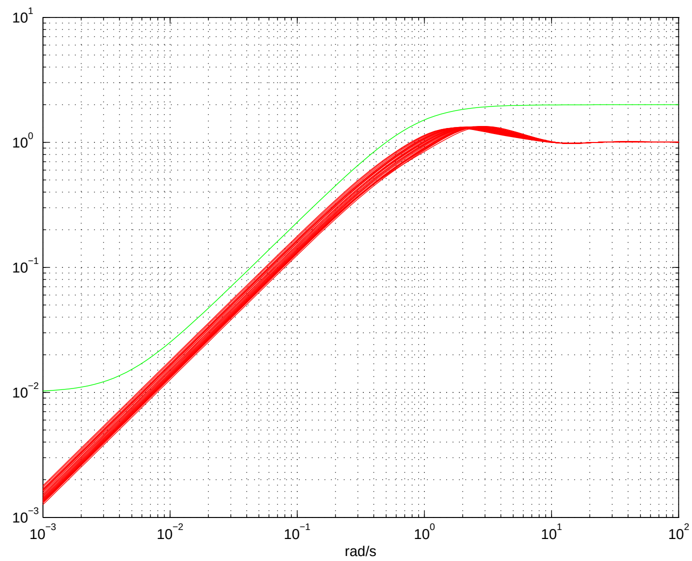

Figure 5: Nominal frequency-domain closed loop responses for the C1 case

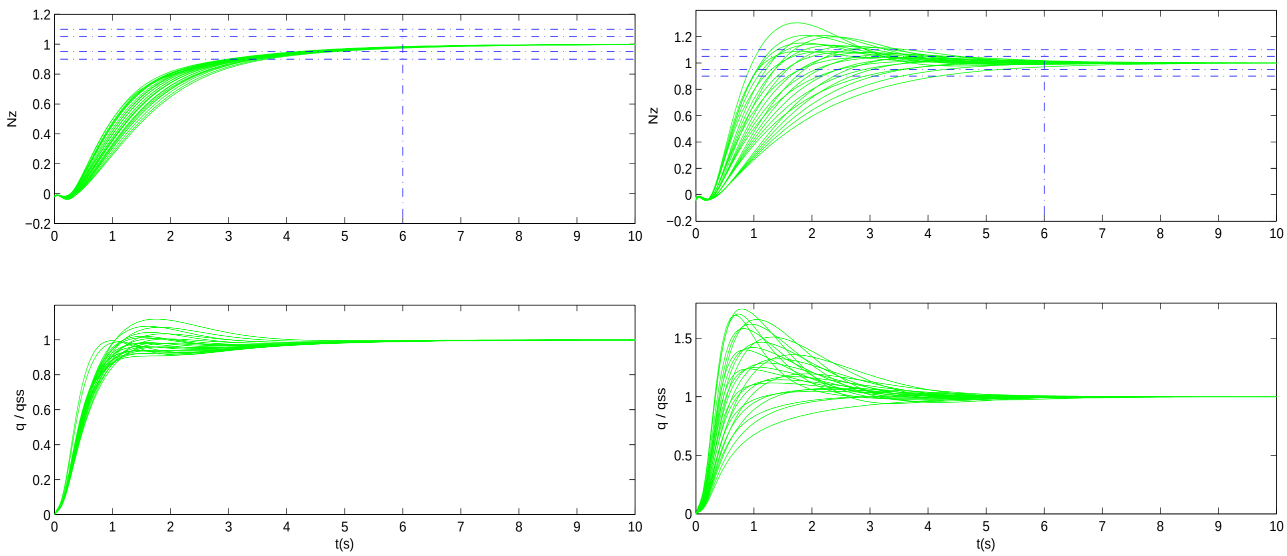

Figure 6: Closed loop step responses of the controllers for the C1 case
(left = gain-scheduled, right = robust)

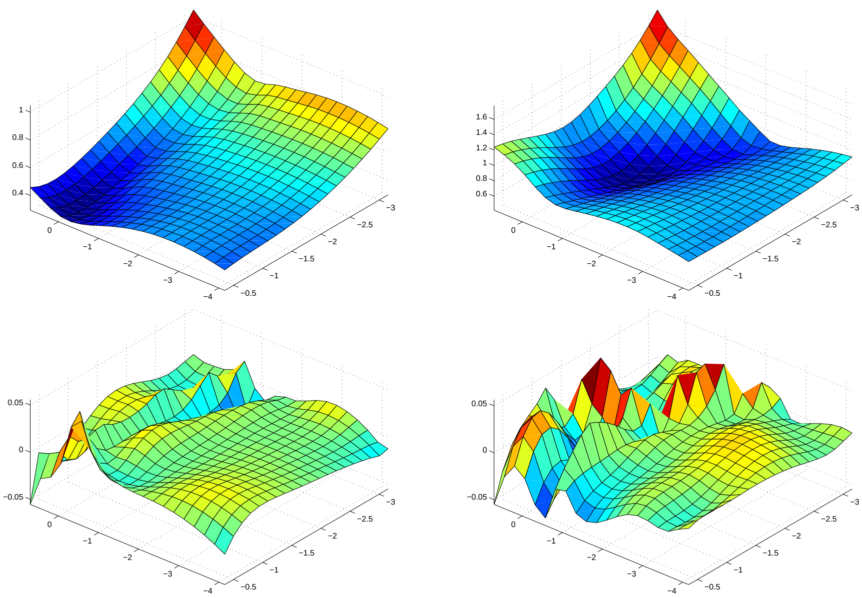

Figure 7: Approximated values (top) and approximation errors (bottom) of the gain K_q for $[M_\alpha^{\min}, M_\alpha^{\max}] \times [M_q^{\min}, M_q^{\max}]$ and 2 specific cross sections $Z_\alpha = Z_\alpha^{\min}$ (left), $Z_\alpha = Z_\alpha^{\max}$ (right)

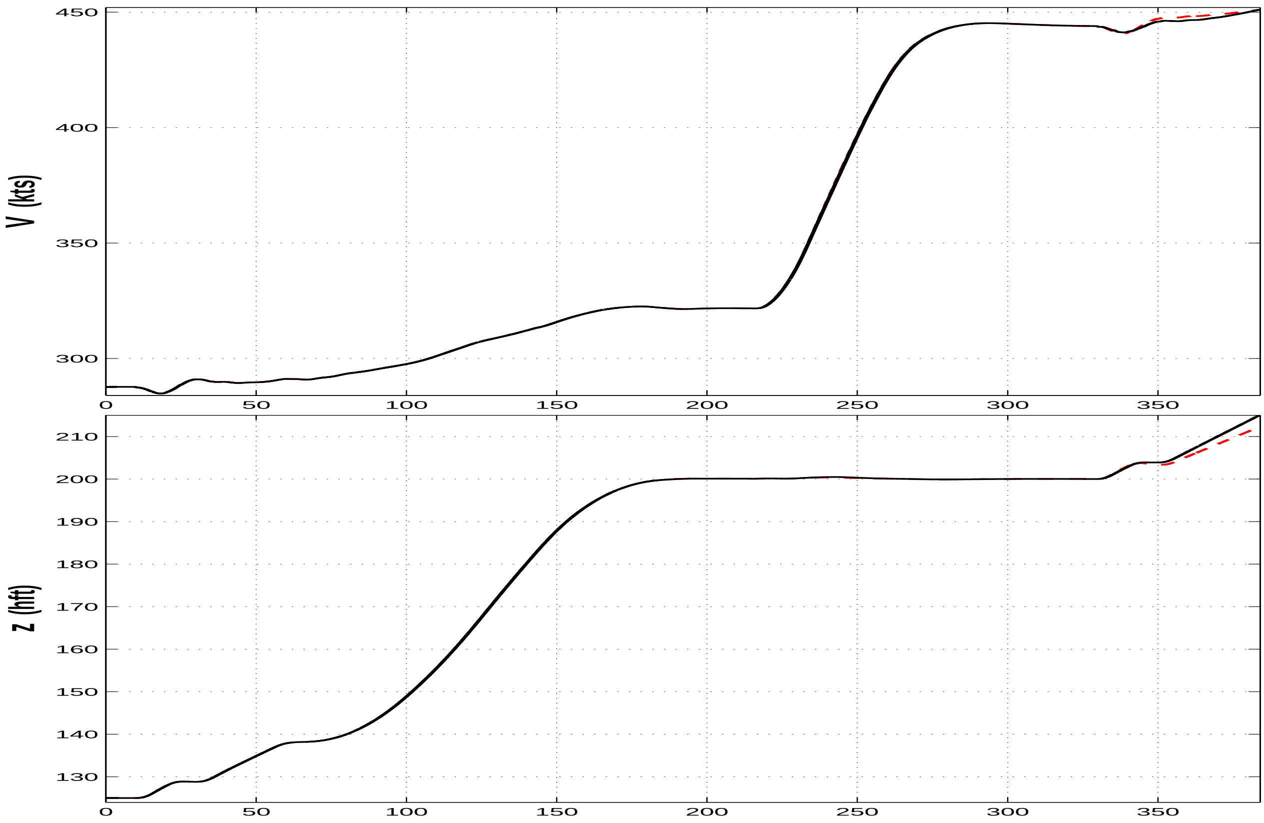

Figure 8: Time history of the slow longitudinal states (true airspeed in kts, altitude in hft)

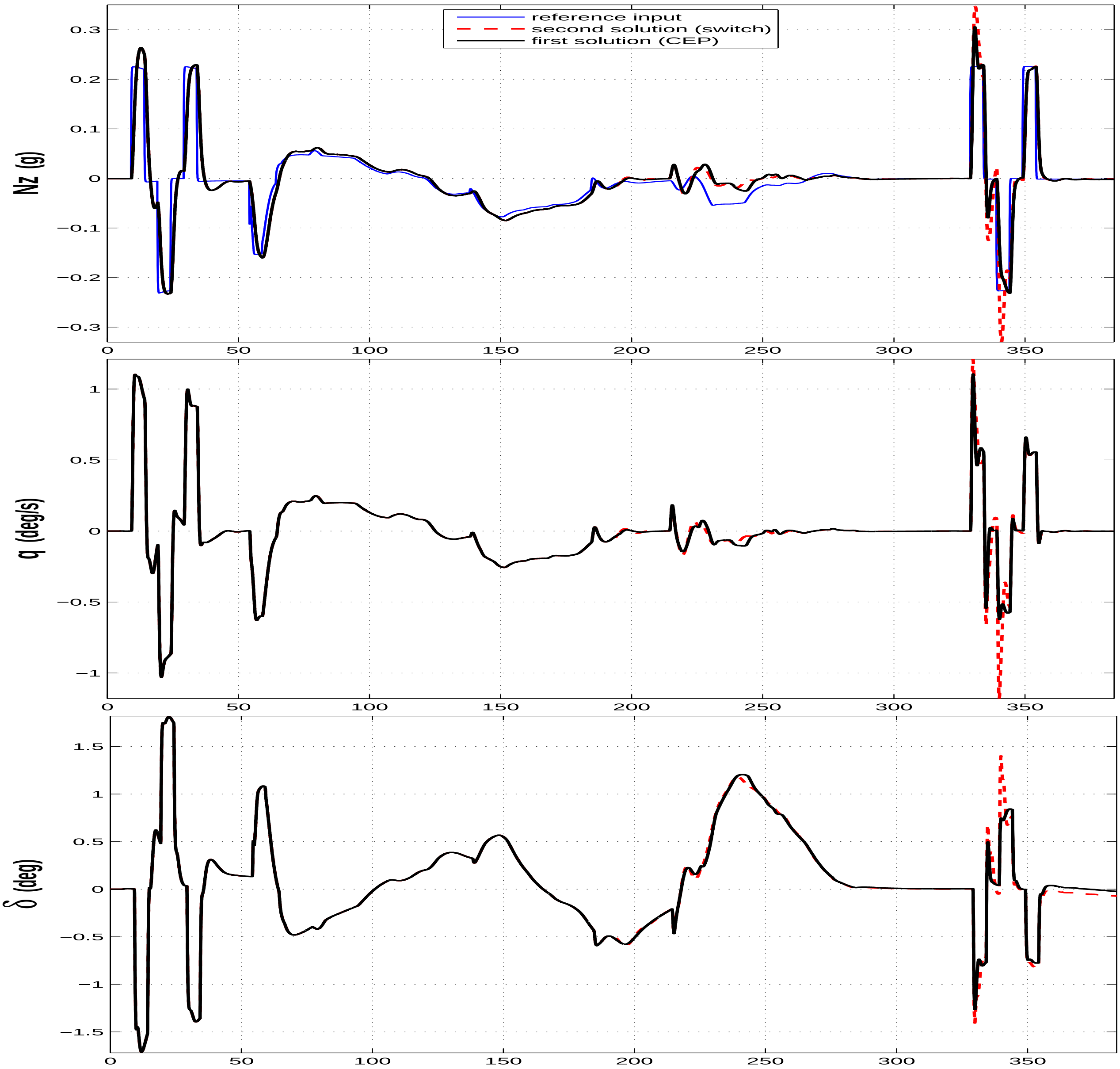

Figure 9: Time history of the load factor (g), rotational rate (deg/s) and elevator input (deg)

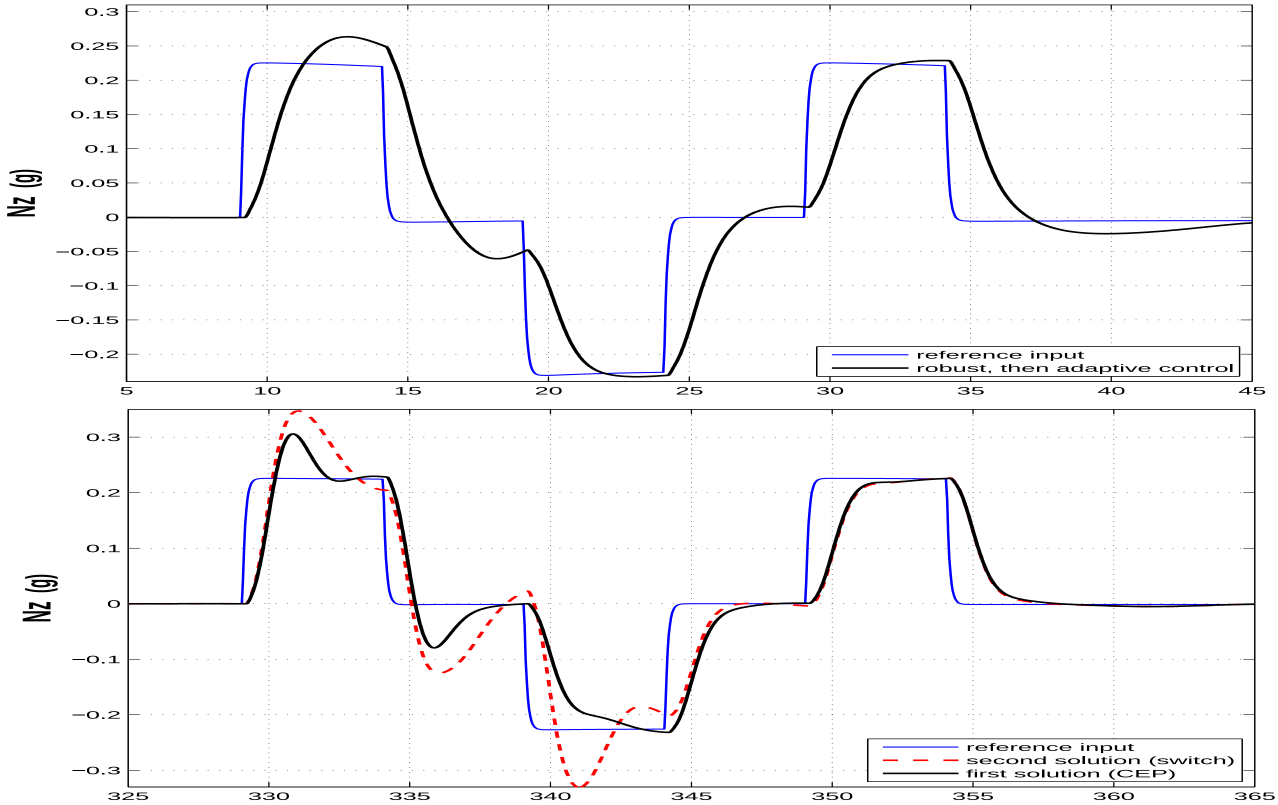

Figure 10: Zooms in on the load factor responses, at the beginning and at the end of the flight

Figure 11: Time history of the estimated M_δ parameter