Robust Gain Scheduled Longitudinal Control of the ADMIRE Using Parametric Robust H∞ Control

Rhenman, F.J.T.; Theodoulis, S.T.

DOI
10.2514/6.2025-2245

Publication date
2025

Document Version
Final published version

Published in
Proceedings of the AIAA SCITECH 2025 Forum

AIAA SciTech Forum
6-10 January 2025, Orlando, FL
AIAA SCITECH 2025 Forum

10.2514/6.2025-2245

Robust Gain Scheduled Longitudinal Control of the ADMIRE Using Parametric Robust $\mathcal{H}_\infty$ -control

F.J.T. Rhenman* and S. Theodoulis†
Delft University of Technology, Delft, The Netherlands

To evaluate the benefits of structured parametric robust H_∞ control a synthesis using this method is completed on a non-linear highly agile aircraft, called the ADMIRE. A gain scheduled two stage feedback controller with a feed-forward was used to design a pitch rate control augmentation system (CAS). The first stage focused on disturbance rejection while providing sufficient margins, while the second stage focused on the performance of the system. To compare the effects of parametric robust H_∞ control three sets of controllers where synthesized, one nominal with no parametric synthesis, one with only the feedback controller synthesized with it and finally one system with both. The results show a clear trade-off between robustness and performance. The controllers synthesized with parametric synthesis greatly decreases the effects of uncertainty in disturbance rejection and tracking allowing for a more robust design.

I. Nomenclature

Latin	**Description**	**Unit**	**Latin**	**Description**	**Unit**
A	state matrix		T	closed loop transfer function	
a	zero location		V	Airspeed	$[\mathrm{m\,s^{-1}}]$
B	input matrix		W	weighting filter	
C_m	pitch coefficient		w	exogenous input vector	
C_x	longitudinal coefficient		X_{cg}	center of gravity location	[m]
C_z	normal coefficient		x	state vector	
f	soft requirement function		y	measured output vector	
G_{ref}	reference model		z	evaluated output vector	
g	hard requirement function		**Greek**	**Description**	**Unit**
K	gain matrix		δ_c	canard angle	[rad]
K_{fb}	feedback controller		δ_e	elevon angle	[rad]
K_{ff}	feed-forward controller		δ_{tvc}	thrust vector angle	[rad]
M	plant and controller		θ	pitch angle	[rad]
m	mass	[kg]	σ	singular value	[]
P	Generalized plant		τ	time constant	$[\mathrm{s^{-1}}]$
q	pitch rate	[rad/sec]	μ	structured singular value	[]

II. Introduction

Control engineers have to balance many requirements to generate a good design. These requirements can range from reference tracking, disturbance rejection, stability margin and robustness [1]. A classical way to generate controllers to meet these requirements is to use well established $\mathcal{H}_\infty$ control synthesis. However, real world problems such as implementation, validation and on-site re-tuning of controller call for basic controller like PIDs, structured or limited-complexity [2]. In recent year multi-model multi-objective problem has been established to use frequency based

*M.Sc. student, Control & Simulation division, Faculty of Aerospace Engineering, P.O. Box 5058, 2600GB Delft
†Associate Professor, Control & Simulation division, Faculty of Aerospace Engineering, P.O. Box 5058, 2600GB Delft, The Netherlands; S.Theodoulis@tudelft.nl. AIAA Associate Fellow.

tuning objectives on a structured controller [3] [4] [5]. In the synthesis process a designer must select the parameters of the weighting filter for this optimization, this can be tricky and currently is mostly done by the experience of the designer [6].

The most famous technique of $\mathcal{H}_\infty$ synthesis was developed by Doyle in his seminal work in the late 1980s [7]. This allowed a controller to be optimized such that the $\mathcal{H}_\infty$ norm was minimized throughout the system, with the use of algebraic Riccatic equations. Including weighting filter, designers could tailor the system to desired needs allowing for robust controllers with adequate performance. This worked well, however the synthesis method resulted in full order controller, the order of the controller was the same as the order of the plant including the weighting filters. When dealing with complex systems and many weighting filters this can lead to high order controllers which are unpractical when applied. To solve this in the mid 2000s Apkarian and Noll developed a non-smooth optimization routine to generate controllers on a predefined structure [8]. This was further refined in the mid 2010s to allow for separating the output channels of the synthesis into individual objectives, allowing for better tuning by the designer [9].

We can synthesize a controller to achieve the required performance of a model. However a model is only a model of the real world system, and we can never be certain that it represents the system fully. The two types of uncertainties are unmodeled dynamics and parametric uncertainty[10]. Typical examples of the two in aircraft, are high frequency dynamics of the wing called flutter and not fully known aerodynamic coefficients. To solve these problems one could generate a high fidelity model with all dynamics included, however this time consuming and can be expensive. Therefore, a different approach must be taken. In recent years a new method of using structured $\mathcal{H}_\infty$ control has been developed which incorporates a numerical representation of the uncertain in a Linear Fractional Transformation (LFT) model [4]. This has seen promising results in both [4] [11] applying this synthesis technique on a tail finned control missile and showing good robustness margins on a challenging problem. Different methods for solving the controller exists but the most prevalent and easily accessible one is a min max optimization.

One of the challenging parts when synthesising a $\mathcal{H}_\infty$ controller is the weighting filter selection. Usually the synthesis is a highly manual and iterative one since a designer would constantly modify the weighting filter parameters to achieve suitable performance and robustness [12]. One of these techniques is called Co-Design which this paper tends to use to minimize the work load by the designer, by parameterizing the weighting filters and optimizing for those values.

The purpose of this paper is to investigate the use combine the two methods of automatic weighting filter selection and parametric synthesis to develop a gain-scheduled pitch rate controller for the The Aero-Data Model In a Research Environment (ADMIRE) aircraft. This aircraft was chosen as it provides a challenging system with unstable dynamics at subsonic speeds. Robust controllers have already already been designed for this aircraft however they have derived full state-space controllers with multiple states [6]. To design the uncertain model the aircraft was ported to fully Simulink version.

The paper is organized as follows. First, in section III, an overview of the $\mathcal{H}_\infty$ multi-objective problem is given as well as the the optimization routine used to solve for parametric $\mathcal{H}_\infty$ structured controllers. The following section, section IV, outlines the use of automatic weighting filters. It provides a mathematical representation of them, the benefits and how they are applied to the synthesis.

The aircraft model is presented in section V. The aircraft model is modified from the original C implementation to the new fully Simulink version, a discussion of how this is done while keeping the original behaviour is presented. The non-linear longitudinal dynamics are stated. The model is trimmed at a variate of trim points and linearized to achieve the flight points for which the controller is to be synthesised. The model is reduced to the short period dynamics to allow for a pitch augmentation system and the parameters of the short period are presented. A part of the uncertainty modeling is also included to show the benefits of there use.

In the penultimate section, section VI, the controllers to compare the parametric synthesis are developed. A set of requirements on the fundamental transfer function and on the stability of the system are presented. This is followed by a discussion the normalization of the model. The method of the two step approach, synthesising the feedback loop followed by the feed forward part, is discussed with the parameters for the weighting functions used in the automatic tuning routine. The parameters for the uncertainty of the model are included, to allow for the construction of the uncertain models to synthesis with. The results of the three controllers are presented showing a distinct trade-off in nominal and robust performance.

The final section, section VII, includes an analysis of the three CAS controllers. A frequency based analysis is conducting showing differences in mostly the control sensitivity function. Linear and non-linear simulations are included with both nominal and robust performance using Monte-Carlo simulations. Stability margins are included to show robustness at the nominal conditions. Finally the structured singular values of the systems where calculated to show the differences in robustness.

III. Multi-Objective Problem

The first concept to introduce is the LFT. A standard closed-loop LFT can be seen in Figure 1. The signals are defined as: $u \in \mathbb{R}^{n_u}$ are the control input, $y \in \mathbb{R}^{n_y}$ the measured output, $w \in \mathbb{R}^{n_w}$ the exogenous input, and $z \in \mathbb{R}^{n_z}$ the regulated output.

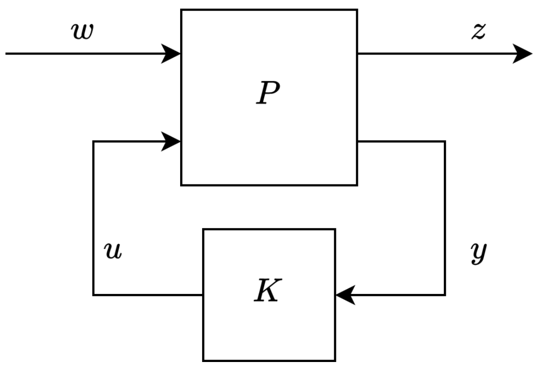


Figure 1 Standard closed-loop LFT model

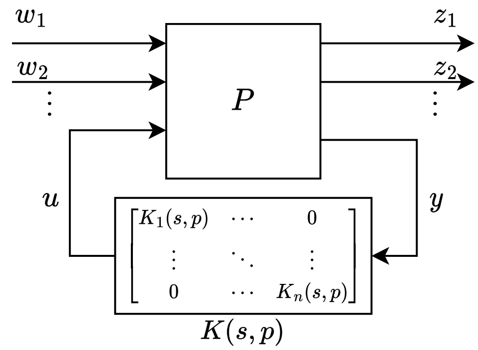


Figure 2 Standard closed-loop LFT with multi-objective signals and tunable controller

This LFT helps us outline the fundamental $\mathcal{H}_\infty$ problem, which tries to compute an optimal solution $K* \in \mathcal{K}$ with the following optimization [5].

$$\begin{aligned} \text{minimize} \quad & ||T_{w\to z}(PK)||_\infty \\ \text{subject to} \quad & K \text{ stabilizes } P \text{ internally} \\ & K \in \mathcal{K} \end{aligned} \tag{1}$$

Over the past four decades this problem has been solved in multiple ways using algebraic Riccati equation (AREs)[7], linear matrix inequalities (LMI) [13] and using a non-linear non-smooth optimizer [8]. This last way of solving for this problem is monumental as it allows for the controller structure to be defined by the designer.

The non-linear non-smooth optimization problem is formulated as Equation 2 and is visualized in Figure 2 [9]. The problem is formatted slightly differently from the standard form seen in Equation 1. There are four main differences between the two formulations, the controller is of a fixed structure, each input signal is mapped to an output, the problem can encompasses multiple models and constraints on the synthesis are split into hard and soft requirements.

$$\begin{aligned} \text{minimize} \quad & \max_{i,k} \left\{ \left\| T^k_{w_i \to z_i}(K(s,p)) \right\| \right\} \\ \text{subject to} \quad & \max_{j,k} \left\{ \left\| T^k_{w_j \to z_j}(K(s,p)) \right\| \right\} \le 1 \end{aligned} \tag{2}$$

In traditional $\mathcal{H}_\infty$ control design [14] a controller is synthesised along one signal vector where the $\mathcal{H}_\infty$ norm of this vector is taken. Apkarian's method allows the designer to specify more than one channel to be used in the optimization. This allows for different closed-loop transfer function and different feedback configurations in the optimization.

The norm $||\cdot||$ can either be an $\mathcal{H}_\infty$ norm, the peak value over frequency of the largest singular value, or the H_2 norm, the integral of the sum of squared element magnitudes of the system frequency response [15].

One important benefit for this new type of synthesis is that it allows the designer to choose the controller structure. To define a structure for the controller an LFT of the problem is generated with $K(s,p)$ being a diagonal matrix containing the tunable elements [3]. To generate these tunable LFT basic building blocks called "real parameters" are created. This allows for an LFT to be defined as a well-posed rational function $R(a)$ containing a fixed matrix M and a real parameter a, Equation 3.

$$R(a) = F_l(M, a \otimes I) \tag{3}$$

Using the properties that any interconnection of LFT models is a LFT model multiple real parameters can be used together in a single posed optimization problem [3]. Now a designer can define a structured controller such as a PID or lead-lag filter in an optimization problem seen in Figure 2. One issue that might arise is that multiple copies of the same parameter may be present in the LFT however little overhead is incurred in the optimization by the extra copies [3].

One more benefit to the optimization structure is to allow for a single controller to be optimized on k models. This means that plant uncertainty can be factored into the design, allowing the controller to be synthesised such that a certain criteria of robust stability is meet during the synthesis. Robust performance can also be achieved as well, however a designer should be always aware that there is a trade-off between robust stability and performance of any system.

The final benefit that this optimization problem contains is the ability to split between hard and soft requirements. Hard requirements are must have requirements, $g(x)$ and soft requirements, $f(x)$ are "nice-to-have" [9] requirements. Splitting requirements as such can allow for requirements on stability margins to never be violated but requirements such as on disturbance rejection to be minimized as much as possible. The format of this structure can be seen in Equation 4 and Equation 5.

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && g(x) \leq 1 \end{aligned} \tag{4}$$

$$\begin{aligned} f(x) &:= \max_{i=1,\ldots,n_f} f_i(x) \\ g(x) &:= \max_{i=1,\ldots,n_f} g_i(x) \end{aligned} \tag{5}$$

As stated before to incorporate uncertainty in the synthesis multiple models of the plant could be used, however this might lead to questions about which set of models to incorporate into the optimization to allow for a complete coverage of the uncertainty domain. Uncertainty modeling can be achieved using three types of uncertainty additive, multiplicative or parametric uncertainty [15].

In recent years methods for synthesising using an uncertain model have been presented [11]. This differs from the multi-model approach as the full spectrum of plants are synthesised across not just a collection chosen by the designer. In order to with an uncertain plant the plant and controller are structured in an LFT Figure 3.

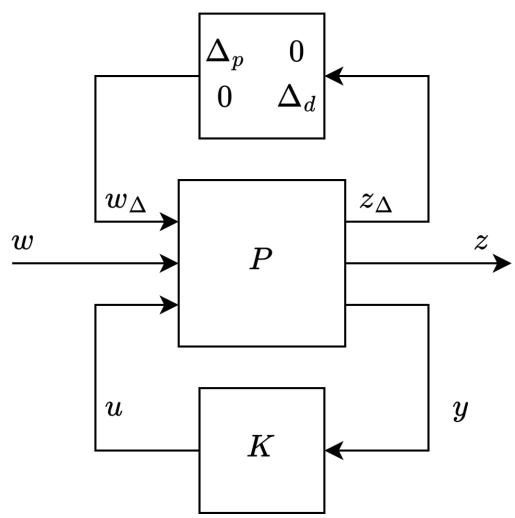


Figure 3 General LFT with controller and structured uncertainty

This type of problem is called a mixed parametric control problem which is cast as a semi-infinite min-max optimization problem, this is similar to the multi-model plant synthesis however now an infinity of plants are used, $P(\Delta)$ [5]. Where the uncertainty comes in two parts, the parametric uncertainty Δ_p and the complex-linear time invariant uncertainty Δ_d.

$$\min_{\mathcal{K} \in \mathbb{R}^n} \max_{\Delta \in \boldsymbol{\Delta}} ||T_{w \to z}(\Delta, \mathcal{K})|| \tag{6}$$

The software used to solve this optimization problem is MATLAB with the robust control toolbox which allows the use of `systune` and `slTuner` [3, 4] interface to establish and solve the problem.

Overall this new non-linear non-smooth optimizer and it's changing of the optimizer problem allows for substantial benefits to the designer. This allows for optimizing the frequency content of different transfer functions in a system. However if the designer wants to limit certain frequency content a signal weighting filters are needed.

IV. Co-Design

This section will give an overview on how co-design can be established using the problem of this type of synthesis.

In traditional multi-objective, multi-model synthesis static weighting filters W would be selected such that a closed-loop transfer function J of interest would become $T = WJ$. One problem with this is that W needs to be selected by the designer and can be arbitrary. A way to solve this issue is to frame the problem into optimizing parameters of the weighting filters, this is called co-design. Co-design was presented by in a Ph.D thesis by Gonzalez [16] and further used by Perez [17].

The most common weighting filters used in multi-objective tuning is a low-pass filter, Equation 7, and high-pass filter, Equation 8. Where ω represents the cross over frequency of the filter, M defines the minimum value and, ϵ represents the maximum value of the filter. One has to keep in mind that for a the transfer functions which these filters are applied to will become the inverse of this filters.

$$W_{lp}(s) = \frac{s/M + \omega}{s + \omega\epsilon} \qquad (7)$$

$$W_{hp}(s) = \frac{s + \omega/M}{\epsilon s + \omega} \qquad (8)$$

As established in multi-objective multi-model optimization soft requirements are sought to be minimized while hard requirements are kept less than or equal to 1. This allows the introduction of a soft requirement on the parameters of a weighting filter, δ_i. This means that the weighting filter parameters can be incorporated into the lower LFT of the optimization problem, with either a soft requirement to minimize the parameter Equation 10 or to maximize the parameter Equation 9.

$$f_k = \frac{1}{1 + \delta_i} \qquad (9)$$

$$f_k = \delta_i \qquad (10)$$

The reasoning for incorporating this in to the design of a controller is that it allows for a selection of requirements which are more tangible to a designer than a traditional weighting filter.

V. ADMIRE Aircraft

This section describes the Aero-Data Model in a Research Environment model, the modifications done to it and, the reduction to a short period model to be used in the synthesis of a flight controller.

The main reason for developing the ADMIRE was to build a more realistic and complex model that reflects the models used in industry. Since, during the The Group for Aeronautical Research and Technology in Europe (GARTEUR) project the available model, High Incidence Research Model (HIRM), was "a generic model and [was] not based on an actual aircraft" [18].

The ADMIRE started its development from the Generic Aerodata Model (GAM) in 1997 [19] and is an extension of this model. GAM itself originated from a national research project between Royal Institute of Technology (KTH) and Saab AB to investigate new control law designs on coupled and nonlinear aircraft. The GAM had a couple downsides to it, one of these was the ease of implementation in research. The model defined the aircraft aerodynamics however it required substantial effort on the control engineer side to utilize it. This lead Svensk Försvarets Forskningsinstitut [Swedish Defence Research Institute] (FOI) to build a MATLAB/Simulink environment around it [6].

The model was last updated in 2006, but even with its age it is still used as a verification tool for novel control designs such as control allocation techniques, anti-wind up designs and non-linear techniques. Its has also been used as an education device with text books using them to show examples of established theory.

A. Non-Linear Flight Dynamics

The original ADMIRE aircraft is a full six degree of freedom model. The non-linear dynamics of this full version can be seen in Equation 11, being the translational equations and Equation 12, being the rotational equations [20], these originates from the first principles. These equations are written in the body frame since the coefficients of the model are calculated in this frame so only the gravitational vector needs a transformation, $[T]^{BL}$. $f_{a,p}$ and m_B represents the external forces and external moments on the system and are effected by the thrust of the aircraft and the control surface deflections. m and I_B^B represents the mass of the system and it's moment of inertia. v_B^E and ω^{BE} are the velocity of the system w.r.t the inertial frame and the angular velocity of the vehicle.

$$m\left[\frac{dv_B^E}{dt}\right]^B + m\left[\Omega^{BE}\right]^B\left[v_B^E\right]^B = \left[f_{a,p}\right]^B + m\left[T\right]^{BL}\left[g\right]^L \tag{11}$$

$$\left[\frac{d\omega^{BE}}{dt}\right]^B = \left(\left[I_B^B\right]^B\right)^{-1}\left(-\left[\Omega^{BE}\right]^B\left[I_B^B\right]^B\left[\omega^{BE}\right]^B + \left[m_B\right]^B\right) \tag{12}$$

In the case of a longitudinal control the model can be reduced into a three degree of freedom, with the longitudinal, normal and pitch motions free, while the yaw, roll and motion in the horizontal plane is kept constant. The model is also formulated in the body plane, so that the u and w are kept as states of the model. The non-linear longitudinal equations of motion (EOM) are presented in Equation 13.

$$\dot{u} = -\frac{1}{2}\rho V^2 C_x \frac{1}{m} - g\sin\theta - qw + \frac{T}{m} \tag{13}$$

$$\dot{w} = -\frac{1}{2}\rho V^2 c_{ref} C_z \frac{1}{m} + g\sin\theta + qu \tag{14}$$

$$\dot{q} = \frac{1}{2}\rho V^2 C_m \frac{1}{I_{yy}} \tag{15}$$

$$\dot{\theta} = q \tag{16}$$

$$\tag{17}$$

The implementations of the EOM in Simulink can be seen in Figure 4. A block interconnection was used in Simulink to describe the EOM since this allowed the use of `uss` blocks to model uncertainty in the moment of inertia I_{yy} and the mass of the aircraft m, more detail on the uncertainty modeling can be seen in subsection V.C. One nice feature, using the Simulink block interconnection was the possibility to label the states, this helped in the workflow when trimming and linearizing the model.

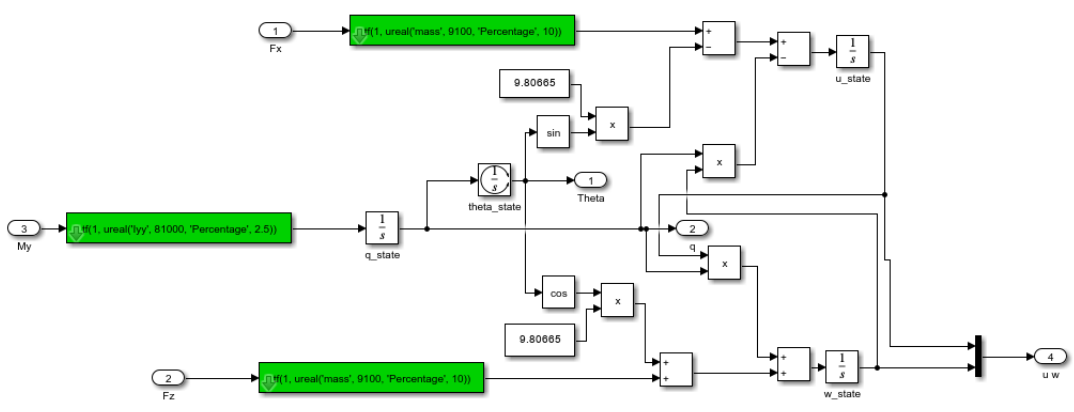


Figure 4 Simulink implementations of the EOM for the longitudinal dynamics

The coefficients use in the EOM are calculated by summation of many different lookup tables, Equation 18 Equation 19 Equation 20. The model uses 24, 31, and 37 lookup tables for the longitudinal, normal and pitch coefficients respectively. In the original model these where constructed in a non-uniform grid and linear interpolated between each points. The lookup tables where ported to Simulink's `Simulink.LookupTable` classes allowing for better access to the lookup tables if modifications where needed. Since the class only handles uniformly gridded lookup tables [21], the lookup tables where converted to uniformly gridded ones. To ensure constancy between the original version to the old version a verification routine of where each lookup tables output was compared to the original with selected input values, as well as a routine where each coefficient calculations where compared to the original with selected input values. There are multiple inputs to the model, including the control surfaces elevon (δ_e) and the canard (δ_c) other inputs to the model include leading edge surfaces (δ_{le}) and the deployment of the landing gear (δ_{ldg}).

$$C_x = f(h, M, \alpha, \delta_c, \delta_e, \delta_{le}, \delta_{ldg}) \tag{18}$$

$$C_z = f(M, \alpha, \dot{\alpha}, q, \dot{q}, V_t, \rho, n_z, \delta_c, \delta_e, \delta_{le}, \delta_{ldg}) \tag{19}$$

$$C_m = f(M, \alpha, \dot{\alpha}, q, \dot{q}, V_t, \rho, n_z, \delta_c, \delta_e, \delta_{le}, \delta_{ldg}) \tag{20}$$

A couple of modifications was conducted on the values of the lookup tables. The lookup tables that were modified was the contribution of pitch coefficient due to pitch rate, elevon deflection and mach, $C_m(M, q, \delta_e)$ and contribution of normal coefficient due to pitch rate, elevon deflection and mach, $C_n(M, q, \delta_e)$. To ensure C_1 continuity, no discontinuity, of the linearized system any lookup table with inputs of the pitch rate requires C_2 continuity along the axis of q. To solve for this while still ensuring the same value of the original lookup tables, Hermite splines where used to solve for the values along the axis of q for the new uniform lookup tables. This does not fully ensure C_2 continuity, since the lookup tables are linear interpolated, however the tables are finely gridded to ensure a sufficient smoothness in the linearized coefficients.

Figure 5, shows the top model of the Simulink version of the ADMIRE. The model contains three distinct parts, the actuator dynamics, the aerodynamic model with the engine model, and finally the sensor dynamics with delays. The original model contained inputs for all six control surfaces, throttle and two for the thrust vectoring. Since this model is only in the longitudinal plane a reduction in inputs is possible to only the elevon, canard, throttle and pitch axis thrust vectoring is present.

The actuator dynamics for the control surfaces and thrust vectoring are first order transfer functions, Equation 21, with a time constant τ. The original model of the ADMIRE included maximum and minimum values for which the control surfaces. This was excluded for the current version of the model however in the future this is planned to be added. Rate limiters for the actuators are included with values of ±50 deg/sec for both canard and elevon deflections.

$$G_{act} = \frac{1}{1 + \tau s} \tag{21}$$

The sensor dynamics used for this model is the same as the original ADMIRE and the sensor dynamics used in the HIRM [22], and are described in the original manual [19] and the set of lecture notes [6]. To model the calculation times of the output of the system, time delays are also included in the system. The Simulink block, `Transport Delay`, was used to model the 0.02 [sec] included in the model. This allows for the use Padé approximation of the transport delay if it is linearized. The longitudinal model also includes inputs for the disturbances to three of the states, q, u and w.

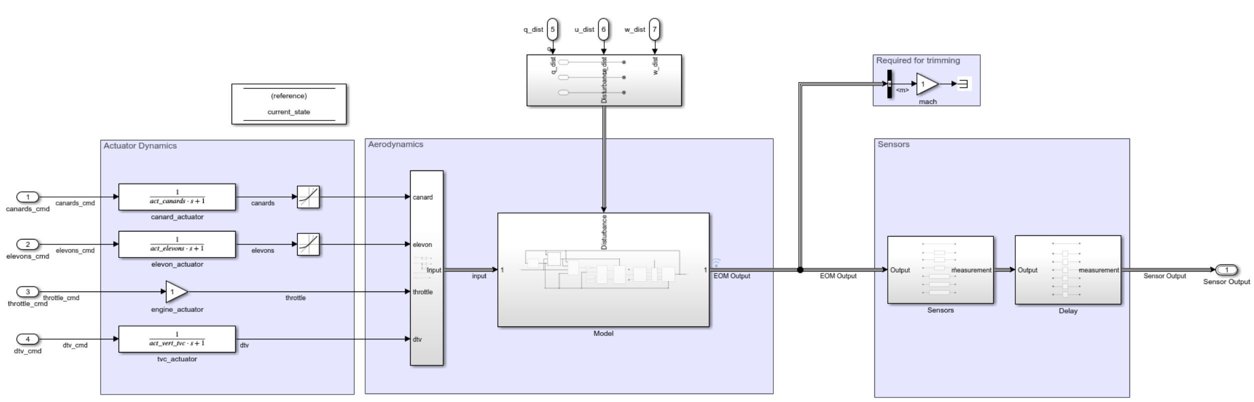


Figure 5 Longitudinal open-loop ADMIRE Aircraft in Simulink

This ported version of the model can be found at a hosted Git-Hub repository *. This repository contains this longitudinal model described, as well as a preliminary version of a full six degree of freedom version of the model. The model uses sub-system references to the coefficients which are stored separately in different files to be used in the different version of the model. The lookup-tables are stored in Simulink Data-Dictionary files and are referenced by each coefficient sub-system. Instead of using individual signals for each value of the model Simulink's `Bus` objects are used to represent groups of signals. Examples of these include the output bus which contains all the measured outputs of the system, input bus and the disturbance bus.

Overall the newer version of the ADMIRE contains many of the modern features available in Simulink compared to when the original model was constructed. This will hopefully allow easier access to the model and allow for a more efficient use of it.

B. Linearized Flight Dynamics

In order to linearize the model for H_∞ control synthesis, trimmed flight points are needed. For the trimmed condition the aircraft is kept at a steady state with the properties seen in Equation 22. As can be seen in the equation all inputs are kept at zero except for the elevon deflection. This was chosen as the control design focuses on the SISO system.

$$\begin{aligned} u &\neq 0 & \dot{u} &= 0 & w &\neq 0 & \dot{w} &= 0 \\ \theta &\neq 0 & \dot{\theta} &= 0 & \delta_e &\neq 0 & \dot{\delta}_e &= 0 \\ \delta_c &= 0 & \dot{\delta}_c &= 0 & \delta_{tvc} &= 0 & \dot{\delta}_{tvc} &= 0 \end{aligned} \tag{22}$$

The algorithm used to solve for the trim point was a gradient descent elimination algorithm, and is included in MATLAB's `findop` function [23]. The domain for which the aircraft was trimmed across was the altitude and mach domain. This was chosen instead of the alpha domain since when gain-scheduling the values of the gains should change slowly.

Using the values trim point linearization can be conducted. Linearization was conducted using MATLAB's `ulinearize` which is similar to the `linearize` command but handles uncertainty, discussed in subsection V.C. This results in a state-space model seen in Equation 23 with all states and all input of the model.

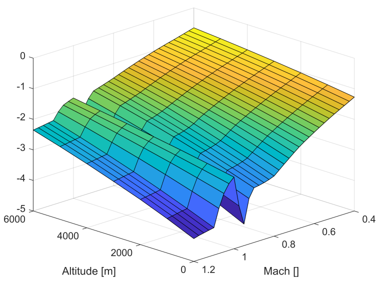
$$\begin{gathered} \dot{x} = Ax + Bu \\ x = \begin{bmatrix} u & w & \theta & q & \delta_e & \delta_c & \delta_{tvc} \end{bmatrix}^T \quad u = \begin{bmatrix} \delta_e & \delta_c & \delta_{tvc} \end{bmatrix}^T \end{gathered} \tag{23}$$

As this paper focuses on the design of a pitch rate controller the principle of short period approximation is valid [24]. One assumption that holds when reducing to a short period approximation is that the velocity of the aircraft is kept constant. Figures of each of the parameters can be found in Figure 6. The resulting state space expression can be of the short period motion can be seen in Equation 24.

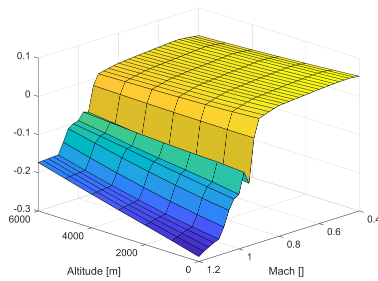
$$\begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{\delta}_e \end{bmatrix} = \begin{bmatrix} z_w & z_q & z_{\delta_e} \\ m_w & m_q & m_{\delta_e} \\ 0 & 0 & \tau_{\delta_e} \end{bmatrix} \begin{bmatrix} w \\ q \\ \delta_e \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} \delta_e \end{bmatrix} \tag{24}$$

In Figure 6 the parameters of the reduced linear short period model are presented. From these figures we see that the parameters are continuous except for in the transonic region as the dynamics of the model changes drastically when close to Mach 1. Therefore a controller will be synthesis only in the subsonic region.

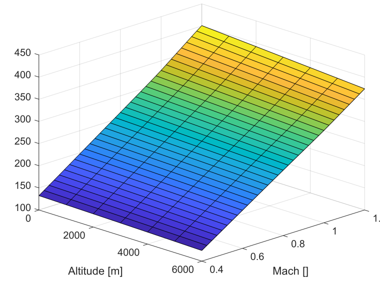
*https://github.com/ADMIRE-REDUX/ADMIRE-REDUX



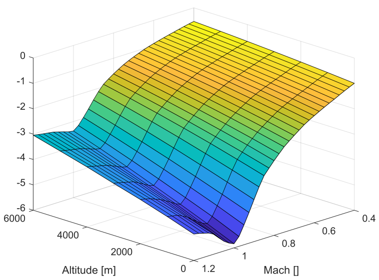
(a) z_w



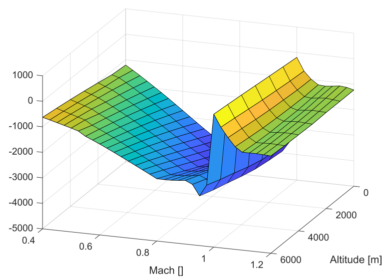
(b) m_w



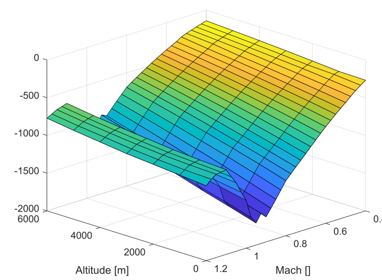
(c) z_q



(d) m_q



(e) z_{δ_e}



(f) m_{δ_e}

Figure 6 Short Period reduction parameters for the ADMIRE

C. Uncertainty Modeling

The modified model also includes the use of Matlab's `ureal` values. This allows uncertainty to be incorporated into the linearization phase of the controller synthesis using `ulinearize` function. For the longitudinal model of the ADMIRE, uncertainty parameterize for center of gravity along the x-axis X_{cg}, moment of inertia in the y-axis I_{yy}, mass m. Supplemental uncertainty for aerodynamic coefficients δC_{m_α}, δC_{m_q} and δC_{δ_e} are implemented.

Equation 25, represents a system of Figure 7 which shows how the uncertainty of the model is defined.

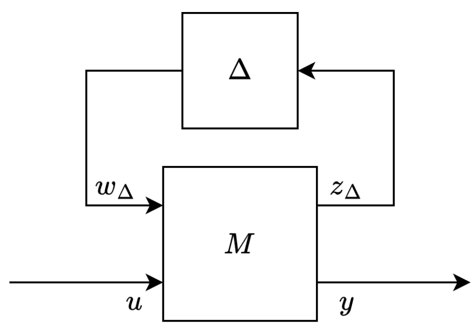


Figure 7 Upper LFT of a plant with structured uncertainty

$$\begin{bmatrix} z_\Delta(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} M_{11}(s) & M_{12}(s) \\ M_{21}(s) & M_{22}(s) \end{bmatrix} \begin{bmatrix} w_\Delta \\ u(s) \end{bmatrix} \tag{25}$$

Figure 8 shows an example of an uncertain short period model, we can see that the DC gain of the system varies greatly as the parameters of I_{yy}, CG position and, mass effect it, however at higher frequencies the uncertainty in the system is less. We see this decrease in uncertainty at higher frequencies because unmodeled dynamics where not incorporated into the uncertainty modeling.

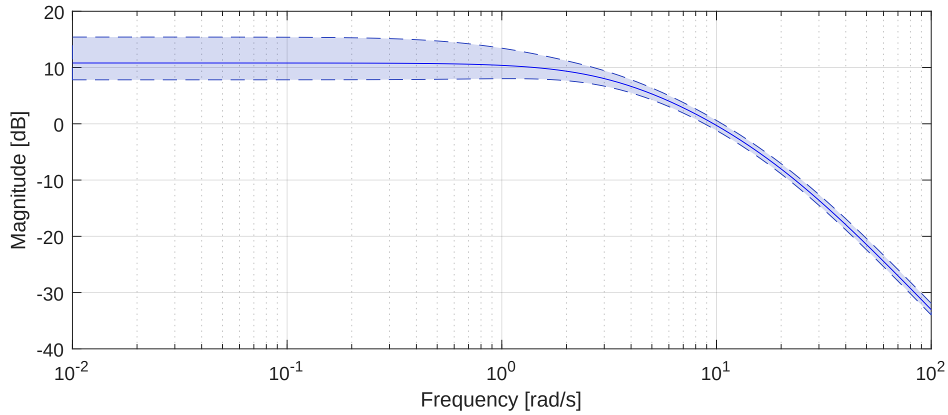


Figure 8 Example of the short period bode plot with its nominal model and a uncertainty component generated by sampling the upper LFT with 100 samples

VI. Controller Synthesis

This section will outline the process of designing a pitch-rate controller using the co-design filters outlined in section IV. The controller will be designed across the domain from mach 0.35 to 0.75 and 0 m to 6000 m in altitude. This mach range was chosen as at a lower speed the aircraft is close to its stall speed at higher altitude and the maximum mach value was chosen as after this point the parameters of the linear model changes rapidly as the ADMIRE approaches the transonic region.

In this section three controllers are derived. These three controllers are determined if the min-max algorithm is used for that synthesis. Table 1 shows how these controllers are derived. "Nominal" denotes a regular multi-objective approach, Equation 4 and Equation 5, while MPCP denotes the mixed parametric control problem seen in Equation 6.

Table 1 Overview of the synthesis types conducted for each controller

	Nominal Model	**Feedback Worst Case**	**2DoF Worst Case**
Feedback synthesis	Nominal	MPCP	MPCP
Feed-forward Synthesis	Nominal	Nominal	MPCP

A. Design Objective

To achieve a good closed loop system which achieve good performance and robustness, design objectives on the frequency domain behaviour of the system can be defined. They are derived from [15] and are split into two categories low frequency domain Table 2 and high frequency domain Table 3.

Table 2 Design objective for fundamental transfer functions at low frequencies

Design Objective	**Reasoning**
$\bar{\sigma}(S_O G)$ is to be minimized	reduce the attenuation of input disturbance at the plant output
$\bar{\sigma}(S_O)$ is to be minimized	reduce attenuation of output disturbance at plant output
$\bar{\sigma}(M)$ is to be minimized	to acquire good handling qualities
$\bar{\sigma}(K S_O F)$ is to be minimized	reduce attenuation of pilot input to the control surfaces

Table 3 Design objective for fundamental transfer functions at high frequencies

Design Objective	Reasoning
$\bar{\sigma}(KS_O)$ is to be minimized	reduce noise attenuation at plant input
$\bar{\sigma}(T_O)$ is to be minimized	reduce noise attenuation at plant output

Other design objectives which are not frequency domain based can be listed as:

- The classical gain and phase margin shall be at least 6 dB and 45 deg, this comes from the U.S. military requirements [25]
- The control surface shall not exceeded its deflection limits of 25 deg and -20 deg and its deflection rate of ±50 deg/sec [19]
- The closed loop system shall be internally stable

B. Normalization

Scaling is important in practical applications as it makes model analysis and controller design much simpler [14]. This section will outline the reasoning why scaling is an important concept. The differences between scaling and normalizing and the rational behind choosing one. A discussion on the two different ways this is implemented on the controller synthesis process and the authors chosen method.

One issue that arises when using frequency based design methods comes from evaluating the DC gain of transfer functions which compares the input of the plant to the output of the plant such as KS_o and KS_oF or vice versa. As seen in the previous chapter the DC gain of the short period plant varies across the flight envelope, this introduces issues with keeping constant requirements across each flight point. Therefore a way of making the requirements imposed uniform across each gain scheduling point a way of scaling the plant or normalizing is needed.

Normalizing and scaling are two different ways of imposing this uniform behaviour. Normalizing imposes a constraint on the DC gain of the useful transfer functions KS_o and KS_oF to be zero decibels. This allows a better comparison of the behaviour of the dynamics along the frequency domain.

One issue with this way of normalizing is the loss of relevancy of the magnitude of the transfer functions. In this case an option would be to use scaling which is presented in Skogestad [14]. Scaling of the model directly relates the requirements defined by the user to the magnitude of the signals. The designer would set requirements on the maximum allowed input change D_u and the largest expected change in error D_e and scale the input and output of the system or model by these values. One benefit that this allows for compared to normalizing is that it can work for MIMO systems. On the other hand a loss of frequency behaviour of the system is lost as more focus is put on the magnitude of the transfer functions.

In this synthesis normalizing of the problem will be used. This allows for easier comparisons of KS_o, KS_oF and S_oG along the frequency spectrum between each flight point but less information can be displayed when comparing the magnitude of the signals.

To accomplish this the short period model is normalized by adding a static gain to the model such that the DC gain is zero decibels. This can be seen in Figure 9. When doing this one has to add a static gain to the controller after it is synthesised such that the output is proper. All values from the controllers in the future will include this scaling.

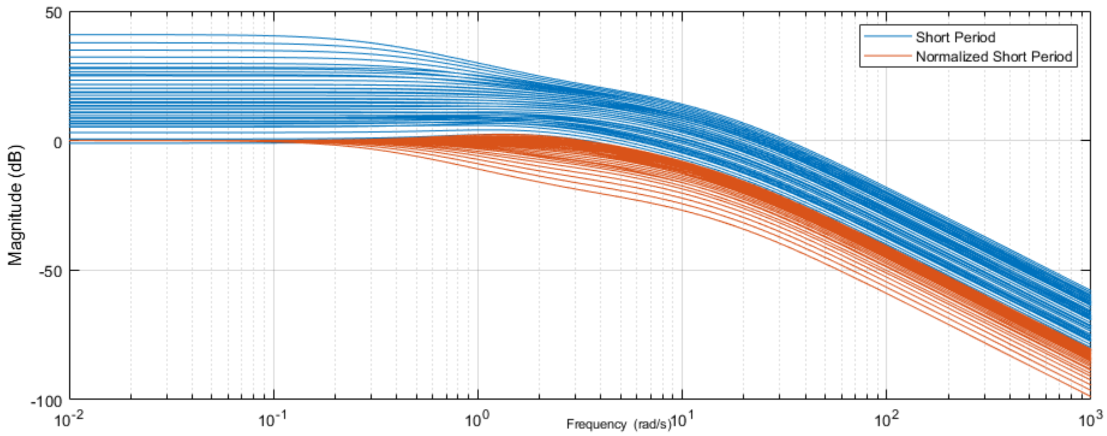


Figure 9 Bode plot of the magnitude of a normalized short period model and regular

C. Feedback Design

The first step in the two part controller synthesis is tuning the feedback loop of the design. The structure of the feedback synthesis can be seen in Figure 10. Here three objective functions are selected to be shaped, S_o which is the output disturbance to the output signal z_1, KS_o which is the noise input n to z_2 and finally the load sensitivity function which is d_I to z_1.

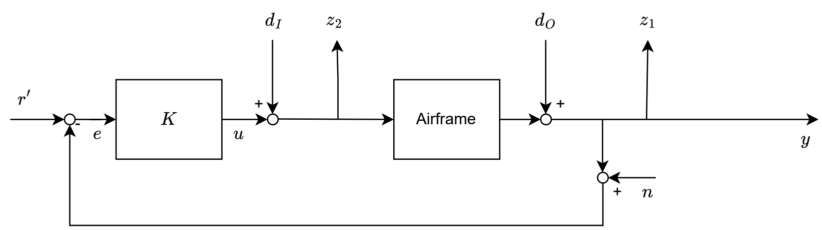


Figure 10 Feedback Synthesis Structure

The structure of the feedback controller was chosen as an integrator with a static gain k_i multiplied by a first order transfer function with zero location a_{fb} and pole location b_{fb}. This was selected so that at higher frequencies a good roll-off of the noise sensitivity function, KS_o, is achieved.

$$K_{fb}(s) = \frac{(s/a_{fb}+1)}{(s/b_{fb}+1)}\frac{k_i}{s} \tag{26}$$

To attain good output disturbance a low pass filter W_{S_o} is added after the signal z_1. Additionally a low pass filter W_{GS_o} is added to the load sensitivity function to reduce the affects input disturbance on the output. The third filter used is a high pass filter to reduce the control actuation due to noise and the reference signal W_{KS_o}.

The Co-Design parameters for the first stage synthesis can be found in Table 4. The values where chosen to meet the design requirements set out. For the output sensitivity function a value of 1.5 was selected for M_{S_o} to allow for adequate classical phase margin as the peak of the S_o directly relates to the phase margin [14]. The Co-Design settings are selected such that the attenuation of disturbance is at low-frequencies are reduced and the noise sensitivity is reduced at high-frequencies.

Table 4 Co-Design weighting filter parameters

Weighting Filter	Filter Type	Parameters	Initial Condition	Optimization Type
W_{So}	low pass	M_{So}	1.5	Fixed
		ϵ_{So}	40 dB	Fixed
		ω_{So}	2 rad/s	Maximization
W_{KSo}	high pass	M_{KSo}	40 dB	Fixed
		ϵ_{So}	40 dB	Fixed
		ω_{KSo}	20 rad/s	Minimization
W_{SoG}	low pass	M_{SoG}	1.5	Fixed
		ϵ_{SoG}	40 dB	Fixed
		ω_{SoG}	2 rad/s	Maximization

D. Feed-Forward Design

The second stage of the two stage process is synthesizing the feed-forward component of the controller. The overall structure of the synthesis can be seen in Figure 11. In this case only two objective functions are accounted for, the control input due to a reference signal r and the model matching transfer function from r to z_1.

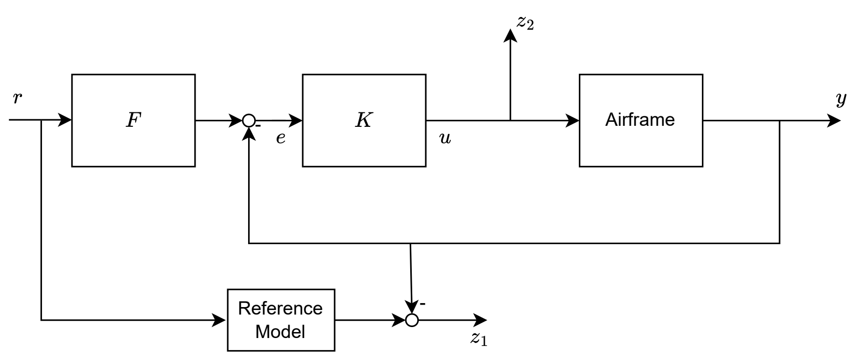


Figure 11 Block diagram of the feed-forward synthesis

The feed forward controller was selected as a first order transfer function, Equation 27, this allowed for adequate shaping of the final system while still reducing the complexity of the controller with only two variable the zero, a_{ff} and the pole, b_{ff}.

$$K_{ff}(s) = \frac{(s/a_{ff} + 1)}{(s/b_{ff} + 1)} \tag{27}$$

In order to improve the transient response of the system a reference model is added with the dynamics of a second order system, Equation 28. The natural frequency, ω_0 was selected as 5 [rad/sec] and the damping ratio was selected as 0.95.

$$G_{ref}(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \tag{28}$$

Two weighting filters are added to the synthesis of the controller. W_{MM} is a low pass filter to allow for model matching and W_{KS_oF} to reduce the control actuation due to tracking at higher frequencies.

The Co-Design parameters of the weighting filters can be seen in Table 5. These where selected to meet the design objectives of the system to meet adequate performance and reduce the actuator deflection.

7

Table 5 Co-Design feed forward weighting filter parameters

Weighting Filter	Filter Type	Parameters	Initial Condition	Optimization Type
W_{MM}	low pass	M_{MM}	40 dB	Fixed
		ϵ_{MM}	0 dB	Fixed
		ω_{MM}	5 rad/s	Maximization
W_{KSoF}	high pass	M_{KSoF}	40 dB	Fixed
		ϵ_{KSoF}	40 dB	Fixed
		ω_{KSoF}	20 rad/s	Minimization

E. Synthesising Across Multiple Models

To evaluate the differences between using multiple models in the synthesis a domain of the parameters of the model must be selected. From robustness analysis of the original controller [19] of the ADMIRE the parameters where selected. These parameters for the longitudinal model can be seen in Table 6.

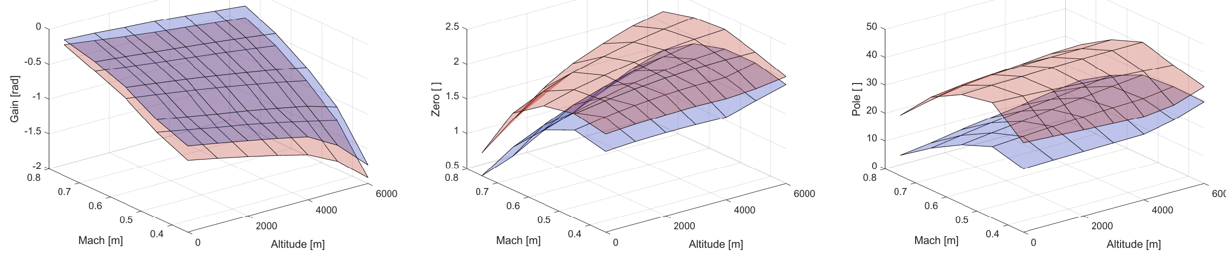
Table 6 Parameter variation for the multiple models used in the synthesis and analysis

Parameter	Minimum	Nominal	Maximum
m [kg]	8190	9100	10010
I_{yy} [kg/m^2]	76950	81000	85050
Xcg [m]	-0.1	0	0.1
δC_{m_α} [rad^{-1}]	-0.01	0	0.01
δC_{m_q} [sec/rad]	-0.001	0	0.001
δC_{δ_e} [rad^{-1}]	-0.01	0	0.01

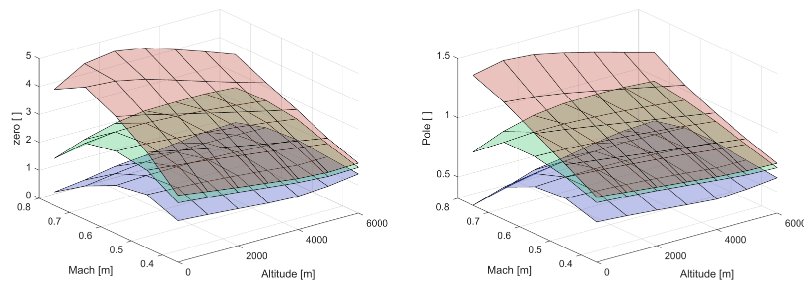
To evaluate the performance and robustness differences when synthesizing across a domain of models three different synthesis are completed. The first one is with nominal models for both stages represented by "nominal model". The second synthesis, "Feedback Worst Case", only uses min-max synthesis in the first stage. Finally, "2DoF Worst Case" uses the min-max synthesis for both stages of the design.

F. Results

The gains achieved from the synthesis can be seen in Figure 12. Since the feedback gains of the feedback worst case and 2DoF worst case are the same, 2DoF worst case are not shown in the figures for the gain surfaces. Overall smooth gain surfaces are achieved which further validates the acceptability of using gain-scheduling for this type of model. For all the flight points and controllers both the lead-lag controllers have suitable margin between the pole and zero to avoid pole-zero cancellation.



(a) K_i **(b)** a_{fb} **(c)** b_{fb}



(d) a_{ff} **(e)** b_{ff}

Figure 12 Synthesized gains at all flight points; blue are the nominal model gains, red are feedback worst case and green are for 2DoF worst case

Already in the values of the gains we can see a difference in the results achieved to what type of synthesis was completed. In the case of the feedback controller the integral gain, when using the min-max algorithm, is greater than the nominal case.

VII. Controller Analysis

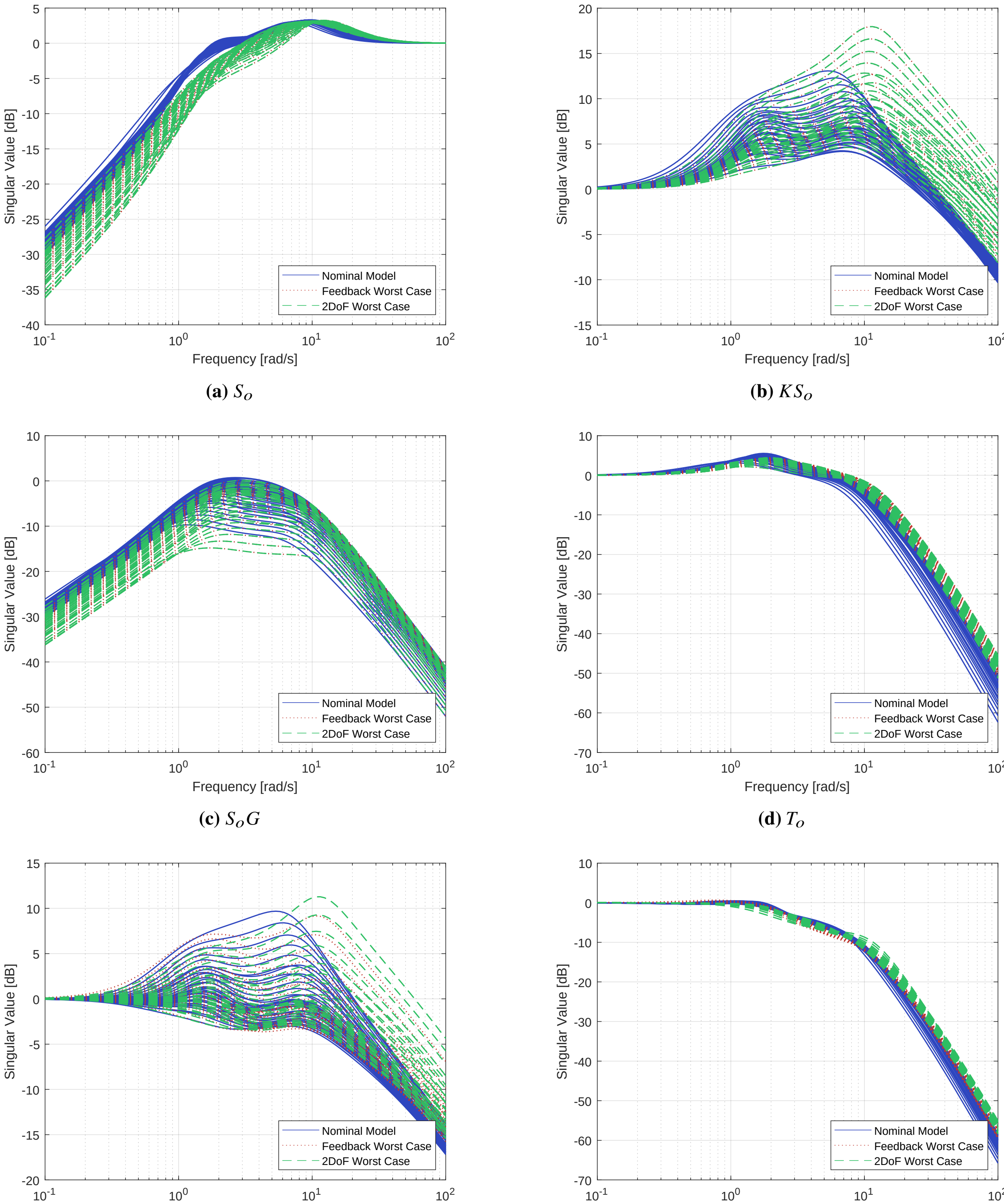
This section will show the performance and robustness of the controllers synthesised. This analysis includes frequency domain analysis, stability margins, linear simulations, non-linear simulation and finally μ-analysis. A comparison of using multiple models in the optimization is included. Three different synthesis where completed one, the nominal, with only the nominal values of the parameters, another synthesis where only multiple models where used in the synthesis for the first stage of the design and finally a synthesis where multiple models where used in both stages.

A. Frequency Domain

The Gang of Six are six transfer functions which show the majority of the relations in a 2DoF control system [26]. Theses transfer functions for the three synthesis can be seen in Figure 13. All three designs show a rejection of low-frequency disturbances, as S_o the output sensitivity, S_oG the load sensitivity function are limited at these frequencies. Since this is a single-input single-output design (SISO) we can also infer that S_I is limited at the low frequencies as the output sensitivity would be equal to the input sensitivity function.

The three designs also show an adequate roll-off at higher frequencies to both reduce attenuation of output disturbance at plant input, KS_o and reduce attenuation of pilot input to the control surfaces KS_oF.

The main differences in the designs can be seen in the S_o, KS_o and KS_oF. The direct relation between the cross-over frequencies of S_o and KS_o is shown, as we shift S_o further to the right, speeding up the dynamics of the final system, a corresponding shift in the noise sensitivity can be seen. Comparing the three synthesis at the KS_oF function we can see that when both stages use a region of models we see the highest cross-over frequency. This might be due to the relatively hard requirement to match the desired model to the entire collection of models leading to high-frequency signals at the plant input.



(a) S_o

(b) KS_o

(c) S_oG

(d) T_o

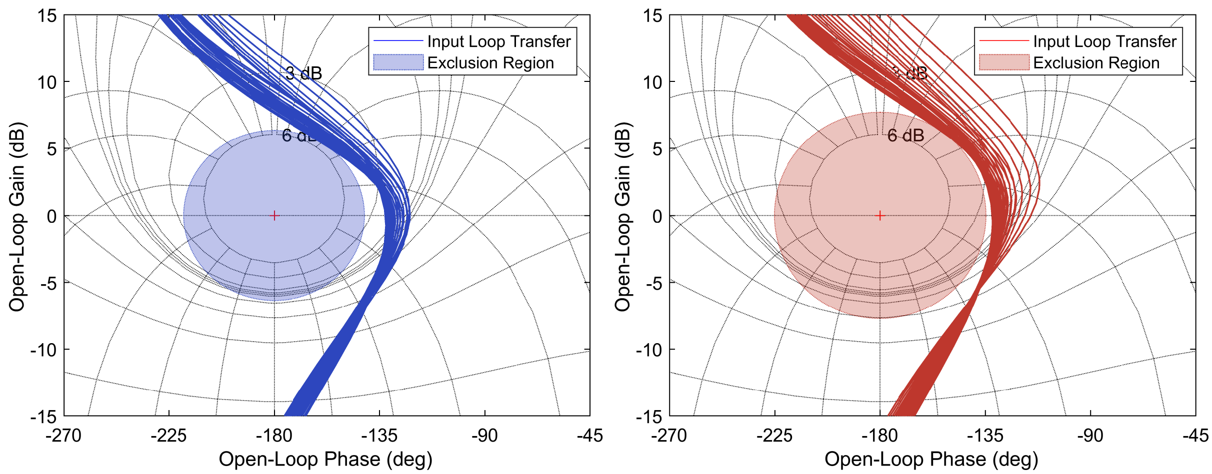
(e) KS_oF

(f) T_oF

Figure 13 Gang of six of all gain scheduled flight points with nominal plant conditions

B. Stability Margins

The Nichols plot for the two first stage synthesis are shown in Figure 14. An exclusion region is included in these plots to show the more conservative disk-based stability margins. Since the structure of the controller are the same, the asymptotes of the graphs are the same. The main difference of the two designs is the distance between the Nichols curve and the instability point.



(a) Nominal Model   (b) Feedback Worst Case

Figure 14 Nichols plots of all flight points. Exlusion Region [shaded] is based on the worst-case disk margin of all flight points

To show that both designs meet the stipulated requirement and to highlight the differences the classical and disk-based gain and phase margins are presented in Table 7.

The Nichols plots and corresponding margin extracted from them used the controller synthesised and the nominal models of the aircraft. This was done as the military requirements do not suggest that the margin requirement should encompose a region of models.

Table 7 Worst-case classical and disk based margins

	Classical		Disk Based	
	Gain [dB]	**Phase** [deg]	**Gain** [dB]	**Phase** [deg]
Nominal	7.61	47.5	6.34	38.5
Feedback Worst Case	9.34	47.9	7.68	45.2
2DoF Worst Case	9.34	47.9	7.68	45.2

From both the Nichols plots and the presented classical and disk-based margins we can stipulate that in the case of using multiple models we can see a greater improvement in the disk-based margin requirement than the classical.

C. Linear Simulation

Linear simulation in the time domain was completed for a simple step response at all flight points in, Figure 15. The figure shows the controller with the nominal model in solid lines and a shaded area showing the region of responses for a set of uncertain models. This region was constructed by conducting 100 Monte Carlo simulations for each flight point with a random set of parameters from Table 6.

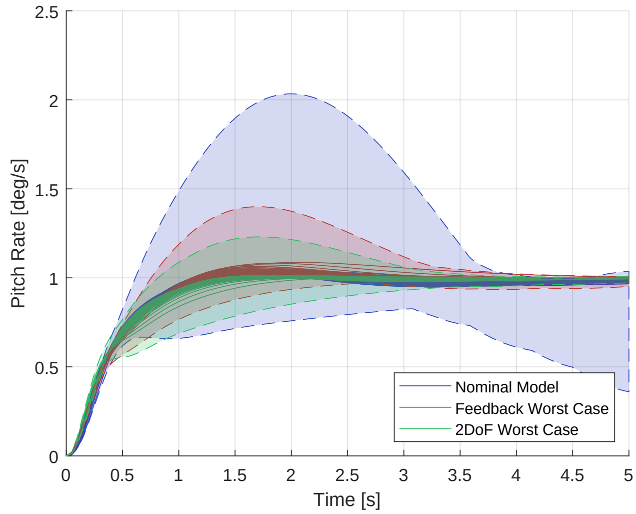


Figure 15 Linear step response of all flight points and bounds of a Monte Carlo simulation of 100 random sampled models per flight point

Figure 15 shows substantially the trade-off a designer has for nominal performance and robust performance. We can see that in the case for synthesising with only the nominal models the step response at each flight point is relatively good however the worst case response is poor.

To investigate the disturbance rejection of the controllers two linear simulations where completed one with a step response to the input of the plant, Figure 16, and one simulation with a disturbance to the output of the system, Figure 17.

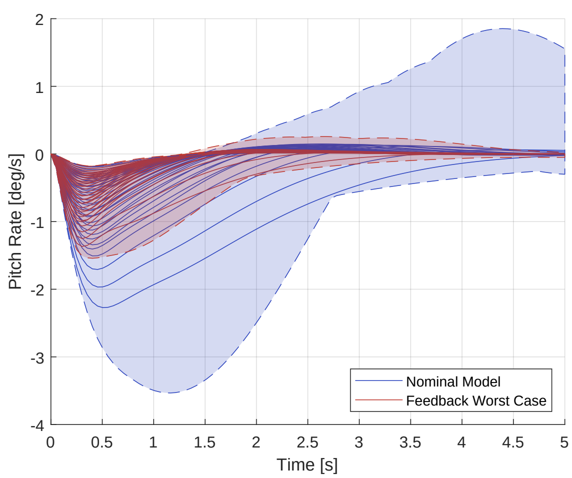


Figure 16 Linear response of a step input disturbance of 0.2 [deg] **with all flight points and bounds of a Monte Carlo simulation of 100 random sampled models per flight point**

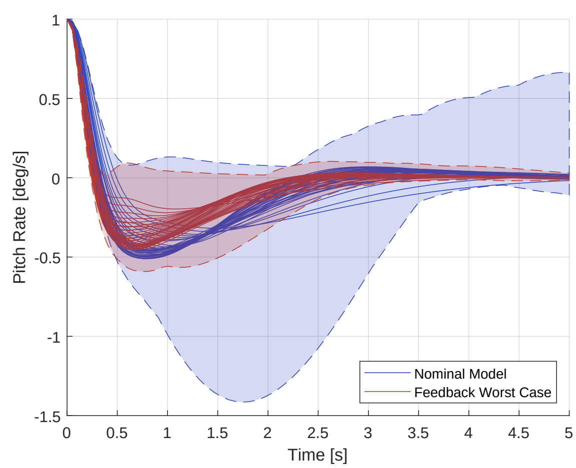
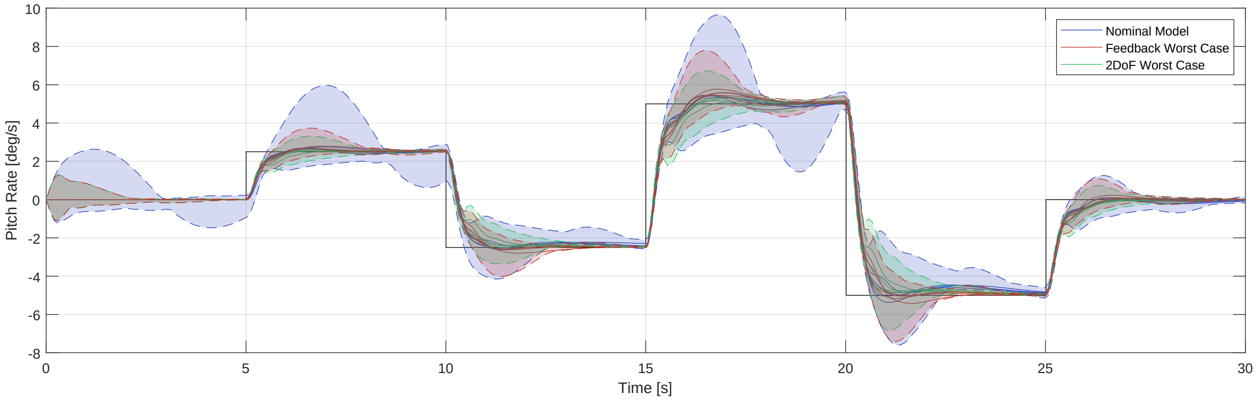


Figure 17 Linear response of a step output disturbance of 1 [deg/s] **with all flight points and bounds of a Monte Carlo simulation of 100 random sampled models per flight point**

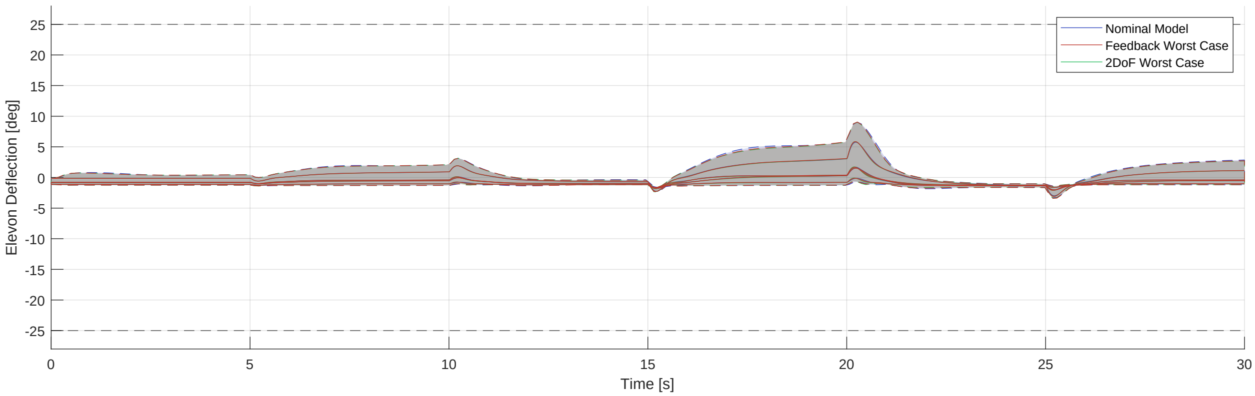
These two figures shows both the performance in nominal and robustness when a syntheses is completed across multiple models.

D. Nonlinear Simulation

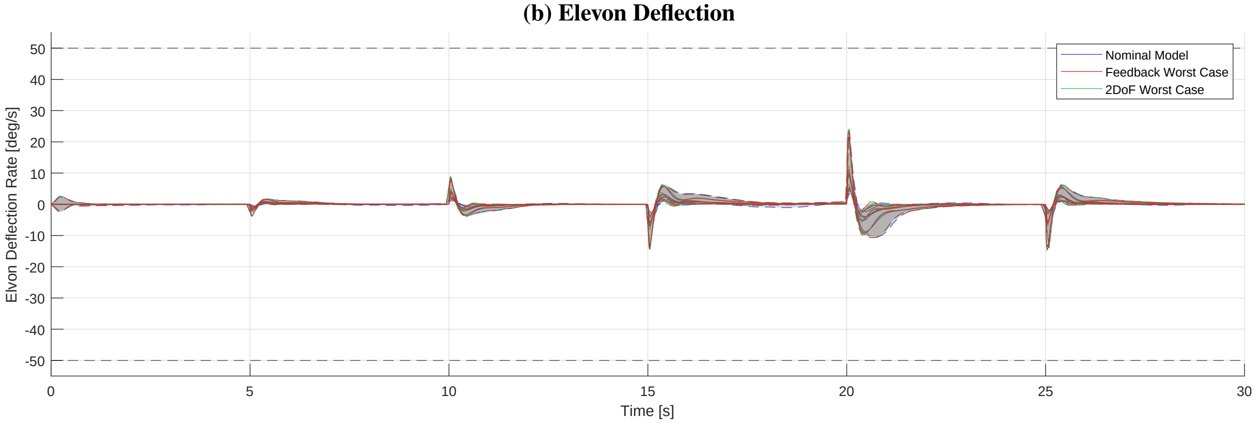
To verify the controllers behave as expected, a non-linear simulation was completed, Figure 18. The maneuver consists of a slow pitch-up and than leveling off the aircraft following this a faster pitch-up is commanded and subsequently a level-off is commanded. Four simulation where completed with the nominal parameters of the aircraft at a grid of mach 0.55 and 0.75 at altitude 1000 m and 5000 m. A subsequent Monte-Carlo simulation is overlaid on the figure to show the behaviour with different model parameters. These parameters where randomly selected with bounds of $X_{cg} \in$ [-0.1 0.1] [m], $m \in$ [8190 10010] [kg] and $I_{yy} \in$ [76950 85050] [kg/m^2].



(a) Pitch Rate



(b) Elevon Deflection



(c) Elevon Deflection Rate

Figure 18 Non-Linear Simulation of the synthesised controller, with all flight points and bounds of a Monte Carlo Simulation of 100 random sampled models per flight point (shaded)

From the non-linear simulations we can see that the elevon deflection and the elevon deflection rate do not exceed the limits imposed for the design. From the elevon deflection plots very little difference in the behavior is visible, this might be due the actuator of the elevon behaving like a low-pass filter limiting the effect that a faster controller has.

The non-linear simulations show very similar results to the linear simulations. There is a clear result showing that when synthesizing with multiple models at both stages a more robust controller is achieved however nominal performance is lost.

E. μ-analysis

μ-analysis also known as structured singular value analysis is a widely used tool to analyse robust stability and performance of a system [6]. In this section a robust evaluation of the final plant will be completed using this tool.

The parametric uncertainty, Δ, used to generate the value of μ stems from the same parametric uncertainty used to synthesis the models, Table 6. The value of μ was computed using MATLAB's function `lftdata` and `mussv` [27].

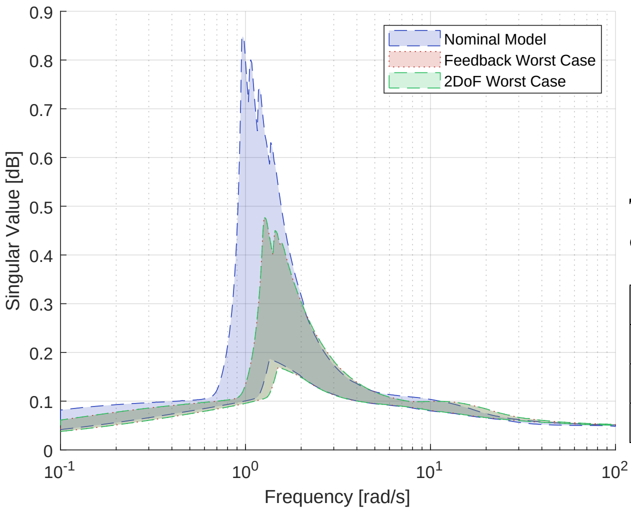


Figure 19 μ-analysis plot at each designed flight point with the upper bound of μ value plotted

Table 8 Maximum μ value for each of the different controllers

	μ [dB]	Frequency [rad/sec]
Nominal	0.849	0.968
Feedback Worst Case	0.476	1.26
2DoF Worst Case	0.476	1.26

The upper and lower bound of the μ are shown in Figure 19 with there respective maximum value and the corresponding frequency in Table 8. The μ values are constructed with the short-period reduced model rather than the full longitudinal state-space. This might be the cause of the narrow banding that is shown in the plot as only significant values of μ are present around the 1 rad s^{-1} frequency.

For both designs the value of μ is less than one, resulting in closed-loop stability in the presence of the parametric uncertainty. Comparing the two varying feedback designs, when a multi-model design is used the value of μ is decreased but the frequency corresponding to that value is increased. This is understandable as with multi-model the worst case parametric uncertainty model is used for each tuning goal relating to a higher degree robust stability.

VIII. Conclusion

The purpose of this paper was to investigate the use of automatic tuning weighting filter and the benefit of using multiple models in a multi-model multi-object tuning of a structured controller. This would lead to a better understanding of the use of this type of controller synthesis of uncertain plants.

To validate the synthesis method the ADMIRE aircraft model was ported from C to Simulink to allow for modern techniques in the MATLAB/Simulink environment. One of the benefits from this was allowing the representation of the linear system in a uncertain state-space class.

Using the technique of automatic tuning weighting filter a two stage gain scheduled pitch rate controllers was synthesised for a section of the subsonic region. First tuning an inner feedback loop controller for and secondly tuning a feed-forward controller. This showed promising results in both nominal stability and performance but also robust stability and performance.

Three different controllers where synthesised to investigate varying if multiple-models was used at each stage of the two stage process. Results showed clear trade-off between the three synthesis techniques as with multiple models the nominal performance is sacrificed for higher robust stability and performance of the system.

References

[1] Sagliano, M., Farì, S., Hernandéz, J. M., Heidecker, A., Garrido, J., Winter, M., Seelbinder, D., Schlotterer, M., Woicke, S., and Dumont, E., "Structured Robust Control for the Aerodynamic Steering of Reusable Rockets," *IFAC-PapersOnLine*, Vol. 55, No. 25, 2022, pp. 31–36. https://doi.org/10.1016/j.ifacol.2022.09.319.

[2] Apkarian, P., "Tuning Controllers Against Multiple Design Requirements," *2012 16th International Conference on System Theory, Control and Computing (ICSTCC)*, 2012.

[3] Gahinet, P., and Apkarian, P., "Decentralized and fixed-structure H control in MATLAB," *Proceedings of the IEEE Conference on Decision and Control*, Institute of Electrical and Electronics Engineers Inc., 2011, pp. 8205–8210. https://doi.org/10.1109/CDC.2011.6160298.

[4] Apkarian, P., Dao, M. N., and Noll, D., "Parametric Robust Structured Control Design," *IEEE Transactions on Automatic Control*, Vol. 60, No. 7, 2015, pp. 1857–1869. https://doi.org/10.1109/TAC.2015.2396644.

[5] Apkarian, P., and Noll, D., "The H ∞ Control Problem is Solved," , 2017. https://doi.org/10.12762/2017.AL13-01{ï}, URL https://hal.archives-ouvertes.fr/hal-01653161.

[6] Bates, D., and Hagström, M., *Nonlinear Analysis and Synthesis Techniques for Aircraft Control*, Springer, 2007.

[7] Doyle, J. C., Glover, K., Khargonekar, P. P., and Francis, B. A., "State-Space Solutions to Standard H2and H, Control Problems," *IEEE Transactions on Automatic Control*, Vol. 34, No. 8, 1989, pp. 831–847. https://doi.org/10.1109/9.29425.

[8] Apkarian, P., and Noll, D., "Nonsmooth H synthesis," *IEEE Transactions on Automatic Control*, Vol. 51, No. 1, 2006, pp. 71–86. https://doi.org/10.1109/TAC.2005.860290.

[9] Apkarian, P., Gahinet, P., and Buhr, C., "Multi-model, multi-objective tuning of fixed-structure controllers," *2014 European Control Conference, ECC 2014*, 2014, pp. 856–861. https://doi.org/10.1109/ECC.2014.6862200.

[10] Mackenroth, U., *Robust Control Systems Theory and Case Studies*, Springer-Verlag, 2004.

[11] Da Silva De Aguiar, R. S., Apkarian, P., and Noll, D., "Structured robust control against mixed uncertainty," *IEEE Transactions on Control Systems Technology*, Vol. 26, No. 5, 2018, pp. 1771–1781. https://doi.org/10.1109/TCST.2017.2723864.

[12] Belletti Araque, J. P., Zavoli, A., Trotta, D., and De Matteis, G., "Advanced H-infinity Synthesis for Launch Vehicle Attitude Control in Atmospheric Flight," *AIAA Scitech 2021 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2021. https://doi.org/10.2514/6.2021-1217, URL https://arc.aiaa.org/doi/10.2514/6.2021-1217.

[13] Gahinet, P., and Apkarian, P., "A linear matrix inequality approach to H control," *International Journal of Robust and Nonlinear Control*, Vol. 4, No. 4, 1994, pp. 421–448. https://doi.org/10.1002/rnc.4590040403.

[14] Skogestad, S., and Postlethwaite, I., *Multivariable Feedback Control: Analysis and design*, 2nd ed., John Wiley and Sons, 2005.

[15] Bates, D., and Ian, P., *Robust Multivariable Control of Aerospace System*, Delft University Press, 2002.

[16] Gonzalez, J. A. P., "Commande Robuste Structurée : Application Co-Design Mécanique / Contrôle d'Attitude d'un Satellite Flexible," Ph.D. thesis, l'Institut Supérieur de l'Aéronautique et de l'Espace, 2016.

[17] Pérez, C. A., Theodoulis, S., Sève, F., and Goerig, L., "Automatic weighting filter tuning for robust flight control law design," *IFAC-PapersOnLine*, Vol. 55, No. 16, 2022, pp. 400–405. https://doi.org/10.1016/J.IFACOL.2022.09.057.

[18] Fielding, C., Varga, A., Bennani, S., and Selier, M., *Advanced Techniques for Clearnace of Flight Control Laws*, Springer, 2002.

[19] Forssel, L., and Nilsson, U., *ADMIRE The Aero-Data Model in a Research Environment Version 4.0, Model Description*, 2005.

[20] Zipfel, P. H., *Modeling and Simulation of Aerospace Vehicle Dynamics*, American Institute of Aeronautics and Astronautics Inc., 2007.

[21] "Simulink User Guide," , 2023. URL https://nl.mathworks.com/help/pdf_doc/simulink/simulink_ug.pdf.

[22] Amato, F., Mattei, M., Scala, S., and Verde, L., "Robust Flight Control Design for the HIRM based on Linear Quadratic Control," *Aerospace Science and Technology*, Vol. 4, No. 6, 2000, pp. 423–438. https://doi.org/https://doi.org/10.1016/S1270-9638(00)01067-1, URL https://www.sciencedirect.com/science/article/pii/S1270963800010671.

[23] "Simulink Control Design User's Guide," , 2023. URL https://nl.mathworks.com/help/pdf_doc/slcontrol/slcontrol.pdf.

[24] Stevens, B. L., Lewis, F. L., and Johnson, E. N., *Aircraft Control and Simulation*, 3rd ed., John Wiley & Sons, Inc, 2016.

[25] "Flight Control Systems-Design, Installation and Test of Piloted Aircraft, General Specification For," Tech. Rep. MIL-DTL-9490E, Department of Defence, 2008.

[26] Åstrom, K. J., and Murray, R. M., *Feedback Systems An Introduction for Scientists and Engineers*, Princeton University Press, 2009.

[27] Balas, G. J., Doyle, J. C., Glover, K., Packard, A., and Smith, R., *μ-Analysis and Synthesis Toolbox*, The Mathworks, 2001.