FliPASED
WWW.FLIPASED.EU

D4.2 Scale-up aircraft re-design and control system layout

Matthias Wuestenhagen (DLR), Balint Vanek, Béla Takarics (SZTAKI), Charles Poussot-Vassal, Pierre Vuillemin (ONERA), Fanglin Yu (TUM)

GA number:	815058
Project acronym:	FLIPASED
Project title:	FLIGHT PHASE ADAPTIVE AEROSERVO-ELASTIC AIRCRAFT DESIGN METHODS
Funding Scheme: H2020	**ID:** MG-3-1-2018
Latest version of Annex I:	1.2 released on 13/12/2022
Start date of project: 01/09/2019	**Duration:** 46 Months

Lead Beneficiary for this deliverable:	ONERA
Last modified: 28/06/2023	**Status:** Submitted
Due date: 30/09/2022	

Project co-ordinator name and organisation:	Bálint Vanek, SZTAKI
Tel. and email: +36 1 279 6113	vanek@sztaki.hu
Project website:	www.flipased.eu

Dissemination Level		
PU	Public	X
CO	Confidential, only for members of the consortium (including the Commission Services)	

“This document is part of a project that has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 815058.”

Glossary

AC	aircraft
AFS	Active Flutter Suppression
ASE	Aero-Servo-Elastic
CAD	Computer-aided Design
CFD	Computational Fluid Dynamics
CFRP	carbon-fiber-reinforced polymers
CPACS	Common Parametric Aircraft Configuration Schema
EAS	Equivalent Airspeed
SMR	Flight Control Computer
GLA	Gust Load Alleviation
HIL	Hardware-in-the-Loop
HTP	Horizontal Tail Plane
LPV	Linear Parameter-varying
LTI	Linear Time-invariant
MDO	Multidisciplinary Design Optimization
MPC	Model Predictive Control
MIMO	Multi-Input Multi-Output
MLA	Manoeuvre Load Alleviation
PID	Proportional-Integral-Derivative
RCE	Remote Component Environment
ROM	Reduced Order Model
SFC	Specific Fuel Consumption
SMR	Short and Medium Range
UAV	Unmanned Aerial Vehicle
VTP	Vertical Tail Plane
ROM	Reduced Order Model
LFT	Linear-Fractional-Transformation
WRBM	Wing Root Bending Moment
WRTM	Wing Root Torsional Moment

Table of contents

List of Figures

1 Executive Summary

The deliverable “D4.2 Scale-up aircraft re-design and control system lay-out” details the design (and re-design) process for the control system and feedback laws design. As a specificity of the FLiPASED project, which aims at encapsulating the control design in the aircraft MDO process, this report details how such a control laws may be as much as possible automatically computed to satisfy the flight and safety requirements. The present report focuses then on the automated control design process for the Maneuver Load Alleviation (MLA), Gust Load Alleviation (GLA) and flutter control functions. These features are classical functionalities that are expected in most of the long range aircraft and are therefore crucial.

The report sequentially details all these three features. Here attention is given to the methodology development and the numerical robustness / versatility of the proposed process rather than on a specific performance oriented control tuning. Indeed, ultimately, the tuning of the macro parameters of each single control function shall be performed in view of a global objective under constraints, mastered by the MDO algorithm.

The contributors of the deliverable is equally balanced between DLR (on GLA), ONERA (MLA) and SZTAKI (Flutter). The integration of the different control design steps within the RCE platform in view of MDO, is performed by DLR and SZTAKI.

2 Introduction

Numbers of tasks in the automatic control (AC) domain remain driven by human ”expertise” which represents the main challenge for its integration in a automated Multi-Disciplinary Optimisation (MDO) tool-chain. From the authors perspective, this stems to two major elements:

- **Loose specifications.** The overall objective for the AC engineer is to make some system as well behaved as possible. Yet this qualitative objective generally involves antagonist elements which hardly translate uniquely into quantitative specifications. This leaves a great amount of freedom in the process and several radically different approaches can lead to equally satisfactory results from a practical perspective.
- **Division of the global AC process.** The AC engineer process is not monolithic. It is made of several sequential elementary tasks (model pre-processing, reduction, synthesis, etc.) which frameworks are globally disconnected of the final objective. Should the aforementioned specifications be given, they still do not decline naturally throughout the AC process and additional choices (tolerances, filters, etc.) must be made. Besides, it is particularly difficult to determine the impact of some (minor) modification of the process on the final result of the full chain. A well known example concerns the model reduction / control steps: a more accurate model do not automatically lead to a better or worth control. The connection is hard or even impossible to make.

To mitigate those two issues within the context of MLA, GLA and flutter control systems, we have to develop simple and intelligible processes based on established methods and our knowledge from the practical constraints of the aeronautical industry. Alternative approaches are clearly possible and the choices and approaches selected may be discussed and amended, but to the authors feeling, they are a good trade-off between optimality and versatility/numerical robustness. Besides, the focus has been placed on robustness of the process more than the performance of the solution. The resulting control-law is therefore not expected to perform as well as a carefully hand-tuned one. One major importance is that the process is fully automated and each block now is ”parametrized” with macro scalars, that may be fixed or optimized later on. The control design process follows the frequency grid depicted in figure 1.

It starts with the rigid dynamics flight control (in the project they are fixed and not optimized). Then, the MLA and GLA are constructed to handle the loads and tracking aspects. Finally, the flutter function, higher in frequency, is designed. The vibration is not considered in the considered aircraft as close to the flutter frequency.

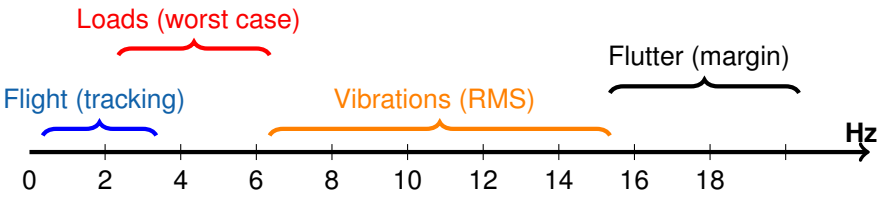

Figure 1: Schematic frequency decomposition of cascaded architecture.

The Scale-up task includes many different tools which have to be connected. The interactive workflow is shown in Figure 2. The synthesis of the control functions for MLA, GLA and flutter suppression have to be incorporated with respect to the overall workflow. In the sequel, the description of the blocks relevant for control law synthesis are described.

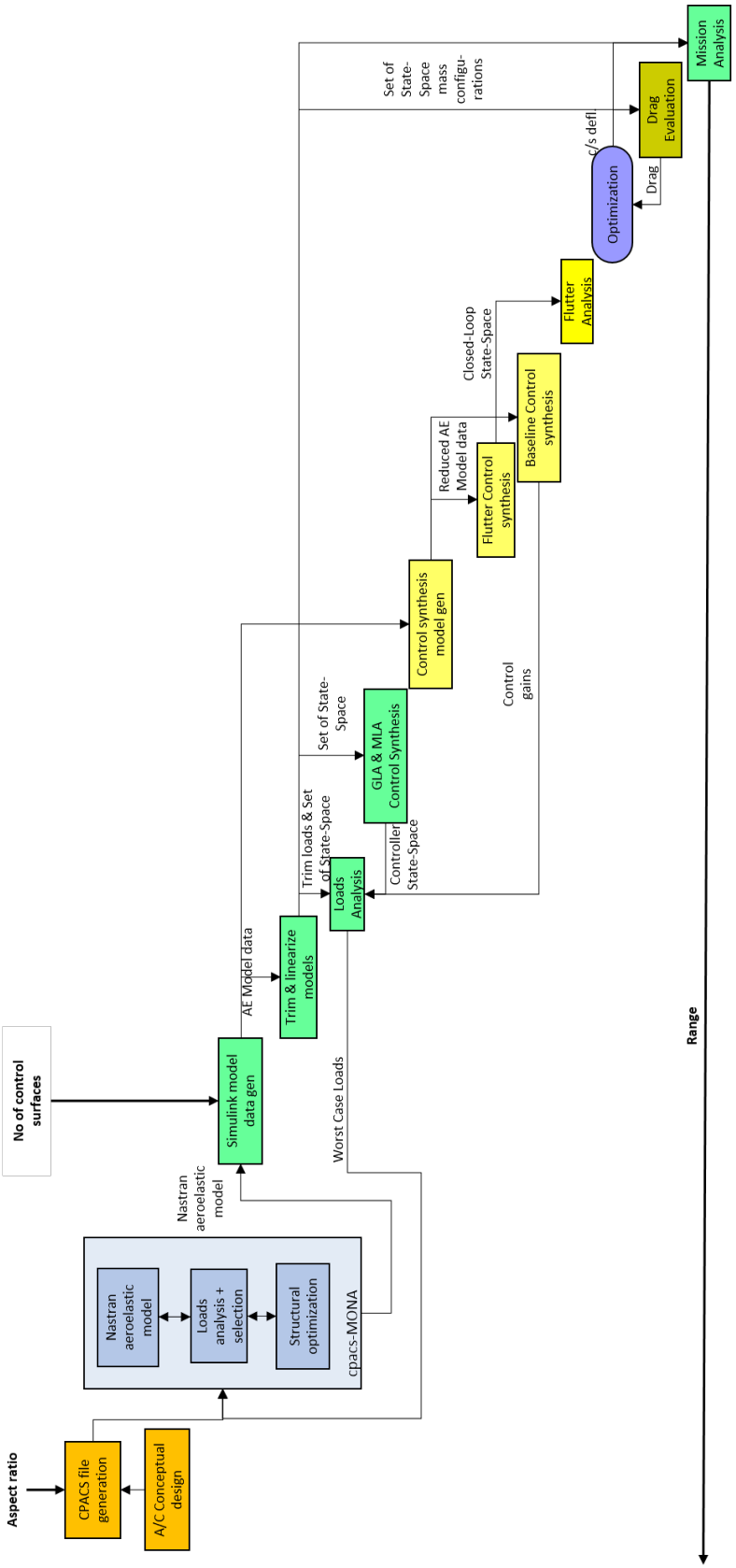

Figure 2: Diagram of the scale-up Workflow

3 MLA control layout and design

3.1 Introduction

The retained architecture for the MLA module is schematised in figure 3. For sake of simplicity and ease of interconnection in the MDO tool-chain, its inputs are reduced to the bare minimum:

- the initial large-scale aero-servo-elastic (ASE) model coming from the physical modelling of the aircraft. Note that what *large-scale* means largely depends on the considered domain. For AC, a few hundred of states is already considered as large-scale and prevents from exploiting modern synthesis or analysis tools which can be numerically demanding. This seldom explains the need for the reduction sub-module.
- The specifications for the MLA control-law are simply the target response time for the tracking and the sought complexity of the controller. Filters and other tuning parameters specific to the retained synthesis framework are kept internal to the module.

In output, the module returns

- the control-law K given by its state-space realisation,
- a performance indicator indicating whether the objective response time and dimension of the control-law are achieved.

The module itself is functionally divided in three blocks:

- model reduction: reduce the number of state of the ASE model to a tractable number,
- control synthesis: find a stabilising and structured control-law to ensure tracking while minimising the loads,
- analysis: determine whether the resulting control-law achieves the global tracking objective on the large-scale model.

Each block is detailed further in sections 3.2, 3.3 and 3.4. The whole module is then illustrated in section 3.5. The limitations and leads for improvement are then discussed in conclusion.

3.2 Model reduction

In order to reach the simplicity and robustness objective mentioned in the introduction, only well established reduction frameworks have been considered [1, 2]. Those frameworks are generally restricted to Linear Time Invariant (LTI) models and stand as follows:

Considering a stable large-scale LTI model H or dimension n, find a reduced-order model H_r of dimension r so that $\|H - H_r\|$ is small in some sense.

This readily sketches some limitations:

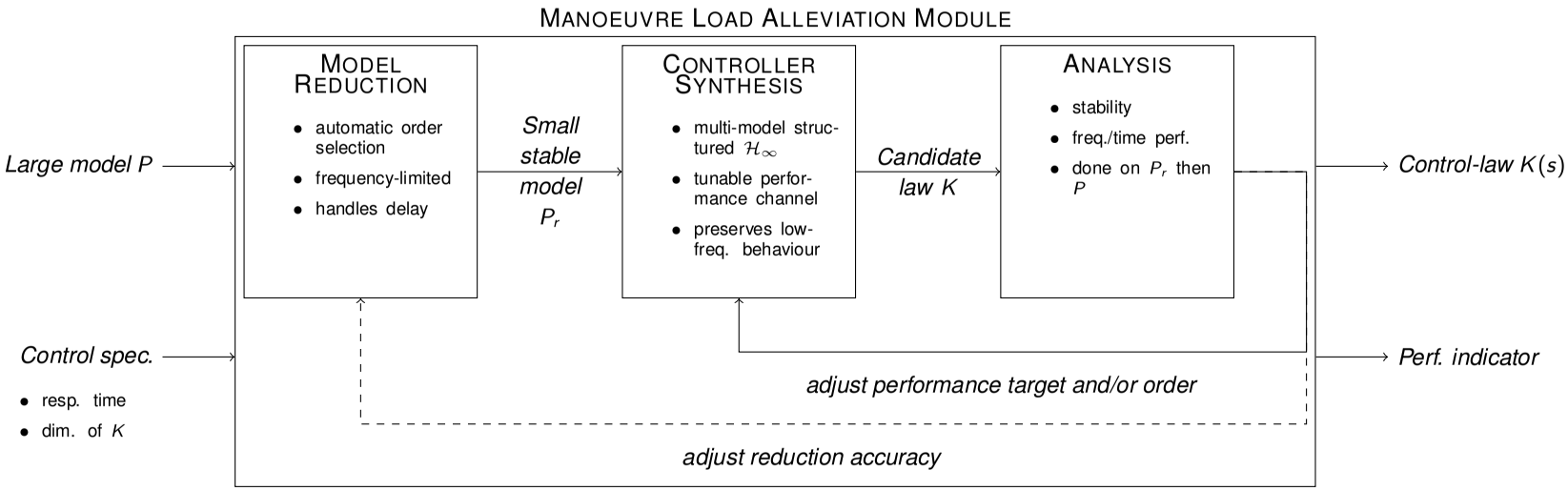

Figure 3: Overall scheme of the automated MLA process

- In practice, most methods assume that the reduction order r is given[1]. Yet it is not a very meaningful decision parameter as different models may require very different complexity to be represented accurately.
- Usual systems norms considered to evaluate the error ($\mathcal{H}_2$-norm or more rarely $\mathcal{H}_\infty$-norm) do not necessarily translate easily some practical constraints such as: preserving static gain, preserve equally all channels, etc. Part of these issues can be address by proper filtering which add more tuning parameters.
- The initial model is generally assumed to be stable, and most method also needs it to be well-conditioned. This may not be the case and several spurious low frequency poles are common with ASE models.
- As the open-loop error is considered, a low error does not necessarily translate in similar closed-loop behaviour with some controller K.

First point is addressed by a dedicated order selection approach, the second point by implementing a framework that has proven suitable for past aeronautical applications and the third point is dealt with a pre-treatment process. The last point can be dealt with using robust control techniques (see e.g. [22]) yet this generally induces some conservatism and is not exploited here. Instead, an a posteriori validation of the law on the large-scale model is preferred and usually performed.

Two reduction techniques have been considered in the module: the Balanced Truncation (BT) and the Loewner Framework (LF). These methods are thoroughly described in the literature (see e.g. [1] and [2]) and only the basic ideas are recalled:

- BT consists in truncating the state-space representation in the balanced basis so that only the most observable and controllable states are kept. In addition to preserve the stability of the large-scale model, the $\mathcal{H}_\infty$ approximation error is bounded by twice the sum of the discarded singular values σ_i of H, i.e.
$$e_\infty(r) = \|H - H_r\|_\infty \leq 2\sum_{r+1}^{n} \sigma_i = \overline{e}_\infty(r). \tag{1}$$
- From a set of SISO[2] frequency-domain data, the LF enables to build a m-th order descriptor model G_m that interpolates the initial data. Provided that there is enough data and under some rank assumptions involving the Loewner matrix $\mathbb{L}$ and shifted Loewner matrix $\mathbb{L}_s$, the realisation can be projected to an order $k \leq m$ without affecting the interpolation. The resulting model G_k is minimal with a McMillan degree given by $rank(\mathbb{L})$. In practice, this rank is computed numerically and thus involves some tolerance.

Both frameworks are thus quite different but the next sections show how they complement each other to form the main elements of the MLA module.

3.2.1 Pre-processing of the model

In addition to their dimension, three issues are generally encountered with ASE models: instability, presence of delays and difference in magnitude of the inputs and outputs. Each point is described below with an adequate counter-measure.

[1] To mitigate this assertion:*(i)* Balanced Truncation (BT) and Loewner Framework (LF) require rather a tolerance on some singular values which translates into an order and *(ii)* with the Optimal Hankel Truncation, the given order is a maximum value, which is already better. Unfortunately, the method has not proven very efficient in practice

[2] In the MIMO case, the interpolation is fulfilled along some prescribed tangential directions.

Instability: ASE models may embed low frequency poles which are either marginally stable or unstable. These either correspond to rigid-body dynamics of to numerical artifacts. For the MLA case and following the cascaded control architecture schematised by figure 1, true unstable dynamics should be taken care by other control modules and they should thus not be modified by the MLA. Therefore, the ASE model for the MLA synthesis should be expected to be stable. Unstable components should thus be discarded.

Stable/unstable decomposition of finite dimensional LTI models is available in Matlab. It is performed prior to the embedding of delays into the ASE model.

Delays: Due to the modelling of the aerodynamic forces on the fuselage or due to the embedding of other control systems loops (c.f. figure 1), input ASE models can contain input/output or internal delays.

While time-delay systems have been widely studied in the literature, associated reduction or control methods remain complex and difficult to implement in an automated manner. Therefore, a practical and standard approach is used instead. It consists in approximating delays through rational functions to fall back on a finite dimensional LTI system. In practice, this approximation step is usually handled through Padé approximation which is available in Matlab. Here: *(i)* Padé is indeed used prior to the BT and *(ii)* as the LF works with frequency-domain data, it is directly fed with the data embedding the delays.

Input-output magnitudes The inputs and outputs gather quantities which magnitudes are largely different (e.g. speed, angle, etc.). In order to preserve equally well all the transfer of the model during reduction, it is necessary to add input/output weighting matrices so that the norm of each channel is comparable.

This is done in the MLA module by adding diagonal scaling matrices in input and in output of the model. The weights are selected to normalise first the norm of each row (i.e. output) and then each column (input) considering either the 2 or ∞ norm. While it is not possible to normalise perfectly all the channels through this process, it significantly decreases the discrepancies that can appear between various channels during reduction thus achieving a better matching from a practical point of view.

Note that this normalisation process is also very useful for the synthesis process as it eases the selection of weighting functions.

3.2.2 Automatic order selection

The most straightforward approach comes from the BT technique which offers an interesting upper bound on the approximation error through the Hankel Singular Values (HSV). Note that the LF directly comes with an estimation of the adequate order to interpolate some given frequency-domain data. Still, it sometimes requires further reduction and what follows can thus be exploited in combination. In particular, the LF and the BT criterion are used jointly in the data-driven approach sketched below.

Dense delay-free case. Due to the bound (2), fast-decaying HSV is generally considered as a relevant indicator to assess the potential for reducing some LTI model. Note that based on Proposition 8.3 of [1], the upper bound (1) can be completed by the following lower bound,

$$\underline{e}_\infty(r) = \sigma_{r+1} \leq e_\infty(r). \tag{2}$$

The bounds (1) and (2) readily suggest a pessimistic or optimistic approach to select the adequate approximation order. Indeed, considering some target relative error e:

- if the $\mathcal{H}_\infty$-norm $N_\infty = \|H\|_\infty$ of the large-scale model is available, one can seek for the order r

such that

$$\underline{re}_\infty(r) = \frac{\underline{e}_\infty(r)}{N_\infty} \leq e \quad \text{or} \quad \overline{re}_\infty(r) = \frac{\overline{e}_\infty(r)}{N_\infty} \leq e. \tag{3}$$

- if N_∞ is not available due to the dimension of the model, then one can combines the bounds (1) and (2) with

$$\sigma_1 \leq \|H\|_\infty \leq \|\sigma\|_1, \tag{4}$$

to obtain the following relation for the relative error re_∞

$$\underline{\underline{re}}_\infty(r) = \frac{\underline{e}_\infty(r)}{\|\sigma\|_1} \leq re_\infty \leq \frac{\overline{e}_\infty(r)}{\sigma_1} = \overline{\overline{re}}_\infty(r). \tag{5}$$

Obviously, (5) has an increased conservatism in comparison to (3) but it is simpler to compute as it does not involve N_∞.

To highlight the conservatism of these bounds, let us consider the following set $\mathcal{M}$ of test models from COMPLieb [12]: LAH, CDP, DLR2, DLR3, ISS1, CM3 and CM4 (the model TL has been discarded, see remark 1). They have been selected for their resonant nature, a characteristic that is often shared by aeroelastic models. These models are reduced with BT for various orders ranging from 1 to $min(n/2, 50)$ and the relative error re_∞ is computed together with the various bounds (3) and (5). The ratios of the upper bound with the true relative error is reported in figure 4 and the ratio of the relative error with the lower bound is reported in figure 5.

One can see that the conservatism of the upper bound increases with the approximation order while the lower bound has a more constant conservatism. As expected, using N_∞ in (3) is more accurate (blue dots) than using its bounds in (5) which increases even more the conservatism (red dots). Again, the effect is more visible on the upper bound which is on average 8 times larger than the true error when using N_∞ and 15 times larger when it is not exploited. With the lower bounds, the mean values are 2 and 3, respectively.

Remark 1 (Numerical issues associated with the model TL)**.** *Despite its resonant nature, the model TL has been discarded of the results as it led to various numerical issues. In particular, for approximation orders larger than* 35*, both the upper bounds and lower bounds were invalid. For* $r = 35$*, the reduced-order model is already extremely accurate. This illustrates that choosing an unnecessary large order* r *can be counter-productive as it renders the reduction process numerically more sensitive.*

This also shows that the bounds (3) *and* (5) *should be considered with care in practice as they involve quantities (the HSV) that may be numerically sensitive (see [1, chap. 7]).*

Let us now consider the 393-th order MLA model from Flipased H which has been pre-processed. The model is reduced for various orders ranging from 1 to 40 with BT and the approximation error is computed together with the different bounds presented above. The results are reported on figure 6 and are coherent with the previous observations. In particular, avoiding N_∞ increases the conservatism and the lower bounds are closer than the upper bounds to the true error.

Suppose that a relative approximation error $e = 5\%$ is sought. This level of accuracy is reached with a 4-th order reduced model. This order is also suggested by both lower bounds while the upper bounds $\overline{re}_\infty$ and $\overline{\overline{re}}_\infty$ suggest 12 and 15, respectively.

All in all, these tests show that the HSV can provide meaningful information on the adequate approximation order r to reach some prescribed level of accuracy. Optimistic or pessimistic estimations are given through the upper/lower bounds (3) and (5). These bounds can be combined to derive a mixed criterion, e.g.

$$(1-\alpha)\underline{re}_\infty(r) + \alpha\overline{re}_\infty(r), \tag{6}$$

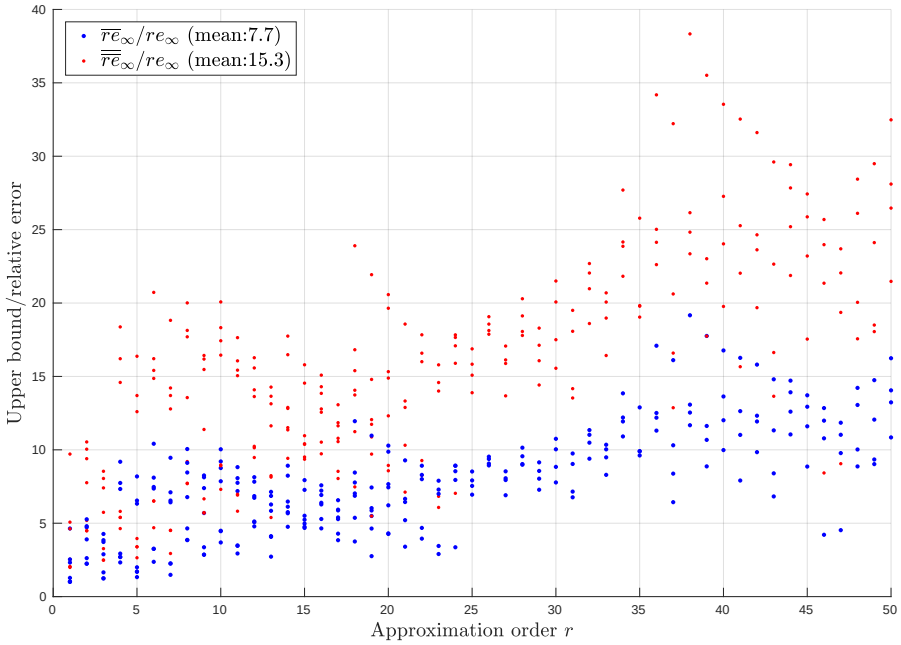

Figure 4: Ratio between the upper bounds of the approximation error and the actual error for various orders and models in $\mathcal{M}$.

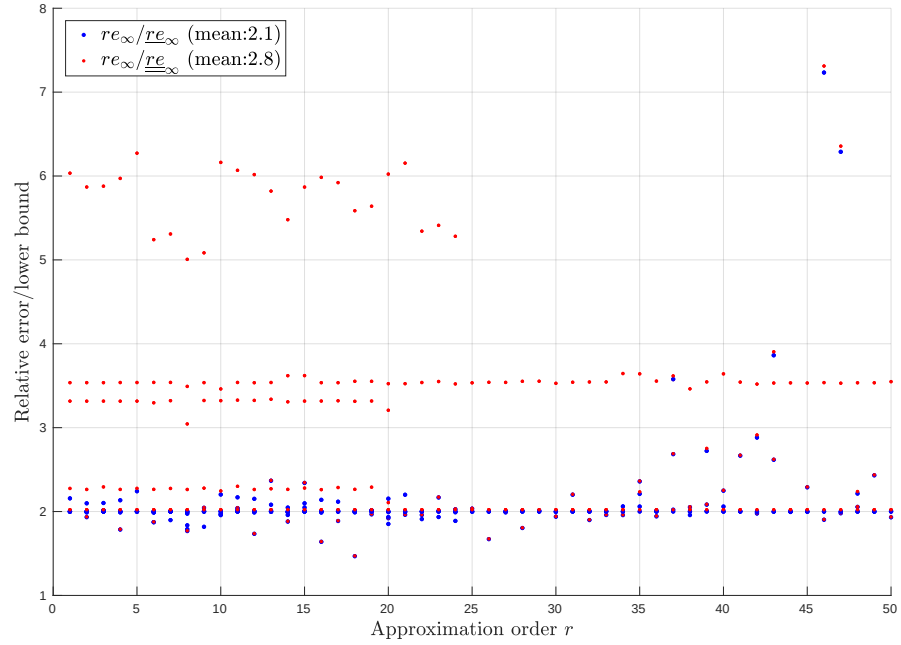

Figure 5: Ratio between the actual approximation error and lower bounds for various orders and models in $\mathcal{M}$.

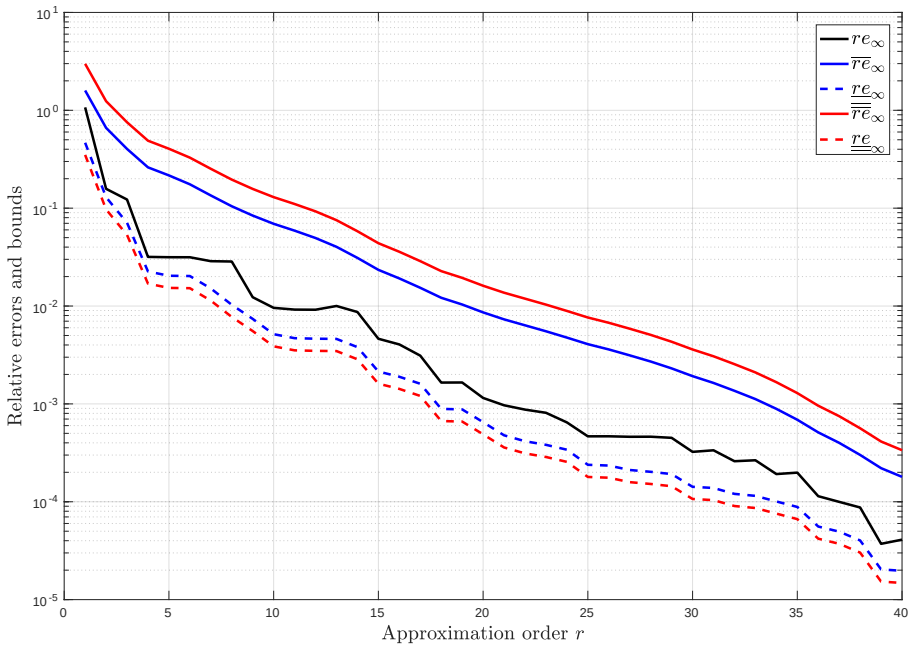

Figure 6: Approximation errors and associated bounds for the MLA model of Flipased.

where $\alpha \in [0, 1]$ is a tuning parameter to adjust the compromise of approximation order against the requirement that the resulting order enables to reach a prescribed approximation error.

Still, this approach implies a significant numerical burden as the HSV are obtained at the cost of solving two Lyapunov equations. The approach is thus only suited for dense models of moderate size. In addition, it does not handle delays which must be dealt with separately by ad-hoc methods (e.g. with Padé as above).

Data-driven case. A recent article [7] investigates the use of input-output data to approximate the gramians for use in BT. More specifically, it is shown how the evaluation of the transfer function can be exploited to approximate the integral definition of the gramians. The approach is also linked with a Loewner-based approach. In this context, various ideas stem for reduction purpose

- use the minimal model G_k from the LF to compute the gramians as a surrogate for H to compute the HSV and apply the approach detailed in the previous section. In the sequel, this is referred to as Loewner-HSV. Note that only stable HSV are considered as the unstable part of the model to be reduced needs to be kept anyway.
- Reason directly on the singular values of $\mathbb{L}$ (or the pencil $(\mathbb{L}, \mathbb{L}_s)$). While they are not the HSV of the model, their decay embeds the information about the minimal order k of the interpolating model (through the rank of the matrix) and can therefore also be relevant. In the sequel, this is referred to as Loewner-SV.

The second point is considered in [7] in comparison to their approach. It is illustrated that the singular values of the Loewner matrix follows the trend of the HSV but are not of the same magnitude. The first point on the other hand, has not been evaluated by the authors of [7].

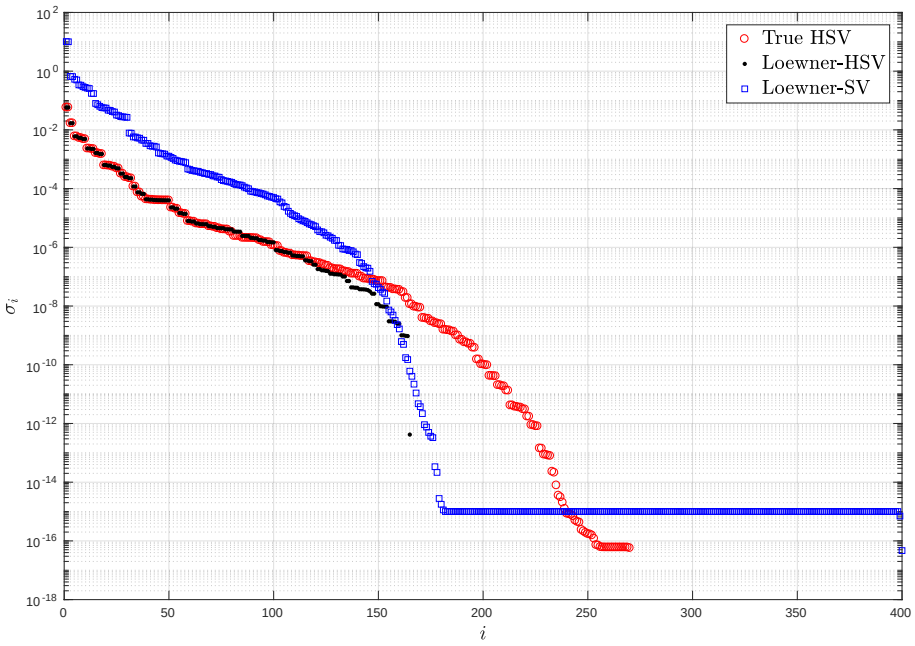

Figure 7: Comparison of the true HSV of the ISS1 model with the Loewner-HSV and the Loewner-SV.

To compare Loewner-HSV with the approach developed in [7], let us consider one of the example the authors give in section 3.4.1. It compares the HSV of the ISS1 model with their estimation. For the data-driven approaches, the authors use a grid of 400 frequency points logarithmically spread from 10^{-1} to 10^{2}. The resulting singular values are reported in figure 7.

First, note that the Loewner-SV reach machine precision after $i \sim 170$. This gives the minimal order k for the interpolant model G_k. This is why there are less HSV with Loewner-HSV. The Loewner-SV indicates that there is no additional information that can be extracted from this set of data. And globally, the model G_k is extremely accurate as $\|H - G_k\|_\infty / \|H\|_\infty = 0.07\%$. This is not surprising considering the number $N = 400$ of interpolation points in comparison to the dimension $n = 270$ of the model.

In addition, we retrieve the results from [7] and we can observe the same scaling factor between the true HSV (red circles) and the Loewner-SV (blue squares). On the other hands, the Loewner-HSV (black dots) appear to be extremely accurate (up to k) and comparable to the ones obtained with the dedicated method in [7].

3.2.3 Illustration of the reduction process

The reduction process is applied to the flipased ASE model with 426 states and 3 inputs, 4 outputs, 9 internal delays and 3 output delays (see the control section 3.3 for further information). Its frequency response is displayed together with the ones of the reduced-order models obtained with BT and LF process.

The corresponding models H_{71}^{BT} and H_{35}^{LF} have not the same dimension. This comes from the way delays are handled and the tolerance on the selection of the approximation order which is based on the HSV for the BT and on the Loewner-SV for the LF. Still, both models are representative of the main dynamics of the initial model up to the prescribed frequency of interest (shaded area). Note that thanks

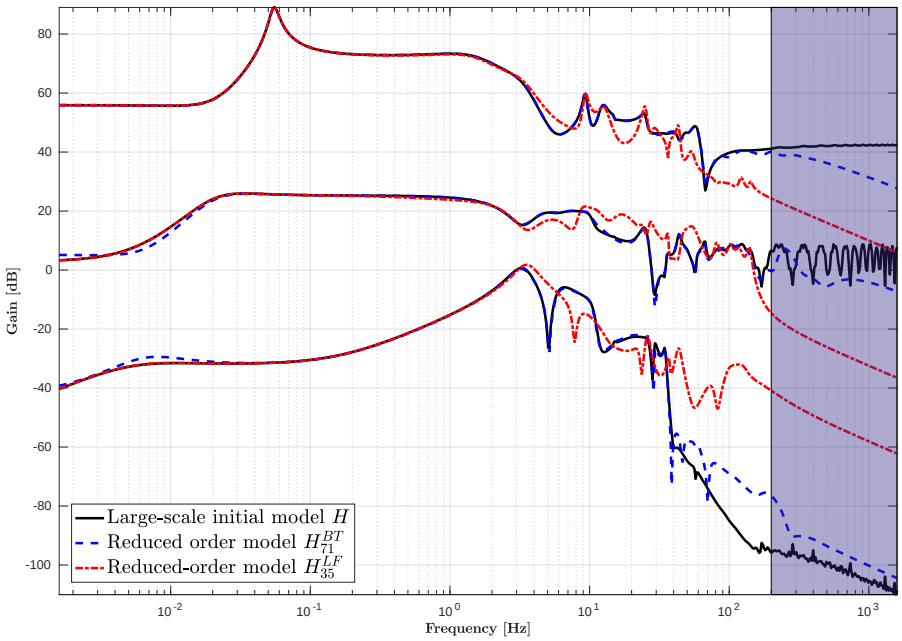

Figure 8: Reduction of the delayed MLA model with the BT and LF approaches.

to the normalisation process, the lower singular values are still matched.

3.3 Control synthesis

Pre-processing. Now a ROM is available, we are ready to process the control synthesis part. As a preliminary to the MLA optimization, the reduced order model input-output are first normalized. As in the reduction process, this input-output scaling allows dealing equally with all transfer in the optimisation process. Indeed, as detailed in what follows, it allows defining weighting / performance filters in an almost universal manner and unified.

Objective function. Given the obtained reduced order and normalized model, from now on denoted $\hat{H}$, the design of a MLA controller is now possible. We chose the $\mathcal{H}_\infty$ framework for this step. Such a framework aims at attenuating the worst-case of the (closed-loop) transfer $T_{zw}(K) : w \mapsto z$. Such a controller may be obtained using $\mathcal{H}_\infty$-norm oriented tools, e.g., through the solution of the following optimization problem:

$$K = \arg \min_{\bar{K} \in \mathcal{K} \subseteq \mathcal{H}_\infty} ||\mathcal{F}_l(G, \bar{K})||_{\mathcal{H}_\infty} \tag{7}$$

under the stabilizing constraint

$$K \text{ stabilizes } \mathcal{F}_l(G, K).$$

We also denote as γ,

$$\gamma = \min_{\bar{K} \in \mathcal{K} \subseteq \mathcal{H}_\infty} ||\mathcal{F}_l(G, \bar{K})||_{\mathcal{H}_\infty} = ||\mathcal{F}_l(G, K)||_{\mathcal{H}_\infty}. \tag{8}$$

Following (8) and the above notations, $\mathcal{F}_l(G, K)$ represents a lower LFT composed of G, a generalized plant that encompasses the ROM computed in the previous step plus the performance weighting functions, and the controller K. The set $\mathcal{K} \subseteq \mathcal{H}_\infty$ is meant to restrict the search of a controller to a given specific structure (this point is detailed in the following). In addition, w stands as the exogenous inputs while z are the performance output. All the magic of this framework stands in the definition of the weights and in the selection of input-outputs couple w, z. This boils down in defining G so that the problem to be solved is a single objective one.

Input output selection for the control set-up. In order to set up the control scheme, let us define the input and output signals selected:

1. the ailerons at patch #4, u_{a4} (sum right and left)
2. the rudders, u_r (sum right and left)
3. the gust input, w_g (the equivalent gust affecting each longitudinal patch is summed by considering the delay between each patch, function of the velocity and results in a single input)
4. the wing loads, l (computed as loads left plus right divided by two)
5. the pitch angle, θ
6. the pitch rate, q
7. the vertical acceleration, a_z
8. the vertical acceleration reference, $a_z^\star$

Then we denote the following generic signals

$$\begin{array}{rcll} w & = & \text{vec}(a_z^\star, w_g) & \text{(exogenous inputs)} \\ u & = & \text{vec}(u_{a4}, u_r) & \text{(control inputs)} \\ y & = & \text{vec}(\theta, q, a_z^\star - a_z) & \text{(system output measurements for the control)} \\ z & = & \text{vec}(z_1, z_2, z_3) & \text{(performance output, defined hereafter)} \end{array} \tag{9}$$

Then one defines the plant P as

$$\text{vec}(z, y) = P\text{vec}(w, u) \tag{10}$$

where P is simply a copy of the ROM, considering the input-output rearrangement and scaling (as defined earlier). This latter may be interconnected to the controller K leading to

$$z = \mathcal{F}(P, K)w = P(K)w. \tag{11}$$

We also denote $P_{i \mapsto o}$ the transfer from input i to output o.

Performance definition. The performance are then defined by channels mean of weight in the transfer from w to z (transfer from exogenous inputs to performance outputs). In the MLA they are meant to enforce the following three constraints:

C1- **Pilot load factor tracking error**:

$$z_1 = T_1(K)a_z^\star = W_e(H_r a_z^\star - P_{a_z^\star \mapsto a_z}(K)) \tag{12}$$

where

$$W_e(s) = g_e^{-1}\left(\frac{g_e/\omega_e s + 1}{1/\omega_e s + 1}\right)^2$$

is a weight that allows for ensuring low frequency attenuation (i.e. gain smaller that g_e for frequency below w_e). In addition

$$H_r(s) = \frac{1}{st_r/3 + 1}$$

is an input weight that suggests a tracking reference signal with a response of the form of a first order with response time t_r (seconds). In the considered setup, $t_r = 6$ (response time in 6 seconds), $g_e = 0.1$ (tracking mismatch in low frequency below 10%) and lower limit $\omega_e = 0.1$ rad/s.

C2- **Attenuation of wind to load transfer peaks**:

$$z_2 = T_2(K)w_g = W_p P_{w_g \mapsto l}(K) \tag{13}$$

where

$$W_p(s) = \| T_2(0)\|_\infty^{-1}$$

is the worst-case open-loop gain of $P_{w_g \mapsto l}$. This simple weight aims only at attenuating the load worst case amplification.

C2- **Stability and roll-off of the controller**:

$$z_3 = T_3(K)y = W_k K y, \tag{14}$$

where

$$W_k(s) = \frac{s^2/\omega_k}{s^2/(g_k\omega_k)^2 + 2s/(g_k\omega_k) + 1}$$

is a high-pass filter, with positive parameters $g_k = 0.1$ and $\omega_k = 1000$ rad/s. These latter are also fixed.

Note that all parameters $\{g_e, \omega_e, t_r, g_k, \omega_k\}$ may be optimized but are chosen to be fixed in the process. Indeed, many other elements may be tuned.

Multi-channel optimization. Constraint C1 relates the tracking objective of a MLA function. C2 relates to the main load attenuation objective, while constraint C3 imposes controller stability and constraints its high-frequency responses (avoiding un-modelled and noise excitements in its output). This series of constraints T then reads

$$T = \textbf{blkdiag}(T_1, T_2, T_3) \tag{15}$$

and is the one to be optimized to find the appropriate K.

Controller structure. Concerning the set $\mathcal{K}$, the chosen controller structure is a dynamic output-feedback controller without direct feed-through term, i.e.

$$K : \begin{cases} \dot{x}_c &= A_c x_c + B_c y \\ u &= C_c x_c \end{cases}$$

where A_c, B_c and C_c are matrices with appropriate dimensions defining a controller of rational order n_c to be determined and optimized in the MLA block. In addition, to ensure static gain tracking, an integral action is also imposed by adding the dynamic on $a_z^\star - a_z$.

"Optimization process". During the optimization proposed in figure 1, the only parameters to be optimized are n_c the dimension of the controller and γ the "optimality" of the performance. The former starts at a minimum value and the problem is solved with an objective $\gamma = 0$. Then, constraints are checked and iterations starts. The next section details this ad-hoc process as well as the analysis performed.

3.4 Analysis and iterations

Stability. After the initial optimization one obtains an attenuation level γ. If $\gamma > 1$, then the controller dimension is increased. Otherwise, the first property to be evaluated is the stability. This latter is easily checked on the ROM by analyzing the closed-loop eigenvalues. Then, in a second step, the stability is evaluated on the FOM including measurement and internal delays. This second step is performed either by approximating the delays via a Padé rational approximation or using a dedicated stability tool developed in [14], applicable to irrational functions. If the stability on the original model is satisfied, the performance is then analyzed.

Performance. Applying the controller on the full original model, and very that the weight constraints are satisfied, i.e. $\gamma < 1$. If true, then the controller has been found. Otherwise, increase the γ objective in the optimization, i.e. decrease the expected performance while keeping $\gamma < 1$.

The process is illustrated in the next part.

3.5 Illustration of whole MLA module

The proposed Matlab code reads as follows.

```
load('+flipased/ss_flexop_1_wing_gla_38')
load('+flipased/1wing_sym_gust_38_2021_3_25_18_45')
speed           = 38; % m/s
measDelay       = 200e-3;
trep            = 6; % MLA response time [seconds]
structure       = 3;
[K,CL,gam,info] = mla.main(sys,x_gust,speed,measDelay,trep,structure,false);
```

Note that the speed and measurement delays are the configuration parameters while `terp` is the t_r coefficient. `structure` is the original complexity of the model, n_c. Running the above code leads to:

```
------------------------------------------------------------
>> CONSTRUCT MODEL
------------------------------------------------------------
>> Select and merge input/output sets
>> Model informations
        * H2 unstable
        * 427 internal variables
        * 12 inputs, 7 outputs
        * 0 internal delays
        * 0 output delays
```

Which loads the model, set the input-output model without any delay. Then, on the rational form, the spurious poles are removed to avoid numerical issues. This performed as follows.

```
------------------------------------------------------------------
>> REMOVE SPURIOUS POLES
------------------------------------------------------------------
Should be (almost) zero: 1.06e-09
Should also be (almost) zero: 3.32e-08
Should be moderately large: 2.29e+01
Should be moderately large: 2.29e+01
>> Model informations
        * H2 stable
        * 426 internal variables
        * 12 inputs, 7 outputs
        * 0 internal delays
        * 0 output delays
```

Then, the internal and external delays are added to the model.

```
------------------------------------------------------------------
>> ADD DELAYS
------------------------------------------------------------------
>> Model informations
        * Internal delays, no stability check
        * 426 internal variables
        * 3 inputs, 4 outputs
        * 9 internal delays
        * 3 output delays
```

The resulting non rational model is then approximated to a rational form. Here, either the Robust Control Toolbox or the MOR Toolbox is used. The resulting model is finally normalized to the appropriately used in the control optimization step.

```
------------------------------------------------------------------
>> RATIONAL APPROXIMATION AND ORDER REDUCTION
------------------------------------------------------------------
>> Using Robust Control Toolbox (rational Pade approximation)
>> Model informations
        * H2 stable
        * 670 internal variables
        * 3 inputs, 4 outputs
        * 0 internal delays
        * 0 output delays
>> Model informations
        * H2 stable
        * 100 internal variables
        * 3 inputs, 4 outputs
        * 0 internal delays
        * 0 output delays
------------------------------------------------------------------
>> CONTROL-ORIENTED MODEL (NORMALIZE)
------------------------------------------------------------------
>> Unstable part size: 0
```

Now the model is ready for optimization. The loop starts and results on the single model investigated are reported here after.

```
------------------------------------------------------------
>> MLA LOOPS START
------------------------------------------------------------
        > OUTER-LOOP: optimize a control structure of order 3
        >> INNER-LOOP: optimize with objective gamma=0.00
                >> Compute weights
                >> Interconnect
                >> Construct MLA controller
Final: Peak gain = 1.19, Iterations = 244
                 WARNING: no stability check possible
                -- Full closed-loop stable (score 100)
                -- Unsuccessful load attenuation control (gamma=1.19)
        > OUTER-LOOP: optimize a control structure of order 4
        >> INNER-LOOP: optimize with objective gamma=0.00
                >> Compute weights
                >> Interconnect
                >> Construct MLA controller
Final: Peak gain = 0.989, Iterations = 233
                 WARNING: no stability check possible
                -- Full closed-loop stable (score 100)
                -- Unsuccessful load attenuation control (gamma=1.02)
        > OUTER-LOOP: optimize a control structure of order 5
        >> INNER-LOOP: optimize with objective gamma=0.00
                >> Compute weights
                >> Interconnect
                >> Construct MLA controller
Final: Peak gain = 0.859, Iterations = 503
                 WARNING: no stability check possible
                -- Full closed-loop stable (score 100)
                -- Successful load attenuation control (gamma=0.90)
```

In this case, the controller dimension is increased from 3 to 5 and, at the end the controller of dimension 5 is able to provide stability and performance. The controller γ value is 0.859 on the reduced model and 0.9 on the full order one.

3.6 Conclusion

The MLA process presented in this section provides a simple way to compute such a function, with few parameters. It only requires the ASE model, the starting order of the controller. The rest is iteratively computed. The output of this computation is a controller for MLA $K(s)$ and a stability guarantee, together with an attenuation level γ. Of course, as mentioned in the introduction, every step may be amended or at least discussed. Still, when applied in the overall process, it allows generating the MLA function.

4 GLA control layout and design

Like the MLA and flutter suppression control, the synthesis of the GLA controller has to be viewed from the perspective of the overall workflow. Therefore, the embedment of the GLA controller synthesis is described. In accordance to Figure 2 the first four green boxes are explained in greater detail.

4.1 Simulink Model Data Generation

The block ”Simulink Model Data Generation” receives Nastran decks from the ”cpacs-MONA” block and subsequently calculates the relevant aeroelastic model data by means of a Matlab based tool [9]. The aeroelastic model data can then be forwarded and processed by the block ”Trim and linearize models”.

4.2 Trim and Linearize Models

This block receives aeroelastic model data from the ”Simulink Model Data Generation” block. It trims and linearizes for different operating conditions and fuel levels, in order to cover the flight envelope. For the ”Loads Analysis” block trim loads are provided. The ”GLA and MLA Control Synthesis” block handles the linearized models for the synthesis of GLA and MLA control laws.

4.3 Loads Analysis

The ”Loads Analysis” block estimates the worst case loads of the system with the GLA and baseline controller active. Therefore, the GLA controller state-space system and the baseline controller gains are fed to the loads analysis. This block also receives the set of state-space models of the open-loop system with the trimmed loads in order to linearly simulate gust encounters. At different monitoring points the loads of the gust simulations are estimated. Adding them with the trimmed loads and extracting the maximum loads yields the worst case loads. As a first step the activity of the MLA and flutter controller are neglected. Fundamentally, however, all control law functions affect the loads. The worst case loads are needed by the ”Structural Sizing” block. This closes a loop, which converges the structural properties and the maximum expected loads.

The performance of the GLA control is judged based on the loads P_c, which the wing structure experiences during gust encounter. Especially the bending moment $P_{c,\mathrm{mx}}$ is of interest. The loads are determined based on the force summation method (FSM)

$$P_c = T_{cg}\left(P_g^{\mathrm{ext}} - P_g^{\mathrm{iner}}\right) \tag{16}$$

with external and inertial loads P_g^{ext} and P_g^{iner}. By means of T_{cg}, the incremental loads of all loads monitoring points are summed up and transformed to the loads coordinate system from the wing tip up to the considered wing position, which than provides the loads at a certain position [5, 11].

The gust input comes from various vertical 1-cosine gust profiles, which are defined by the gust zone

velocity and acceleration $U_{z,t}(t)$ and $\dot{U}_{z,t}(t)$

$$U_{z,t}(t) = \begin{cases} \frac{\bar{U}_t}{2}\left(1 - \cos\left(\frac{\pi}{H_t}\left(U_\infty t - x_z\right)\right)\right), & \text{if } \frac{x_z}{U_\infty} \leq t \leq \frac{2H_t + x_z}{U_\infty} \\ 0, & \text{otherwise} \end{cases}$$
$$\dot{U}_{z,t}(t) = \begin{cases} \frac{\bar{U}_t \pi}{2H_t} U_\infty \sin\left(\frac{\pi}{H_t}\left(U_\infty t - x_z\right)\right), & \text{if } \frac{x_z}{U_\infty} \leq t \leq \frac{2H_t + x_z}{U_\infty} \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

The maximum gust intensity and gust half length are $\bar{U}_t$ and H_t [6]. As time t evolves the aircraft moves through the gust profile from nose to aft, like shown in Figure 9. The aerodynamic model of the

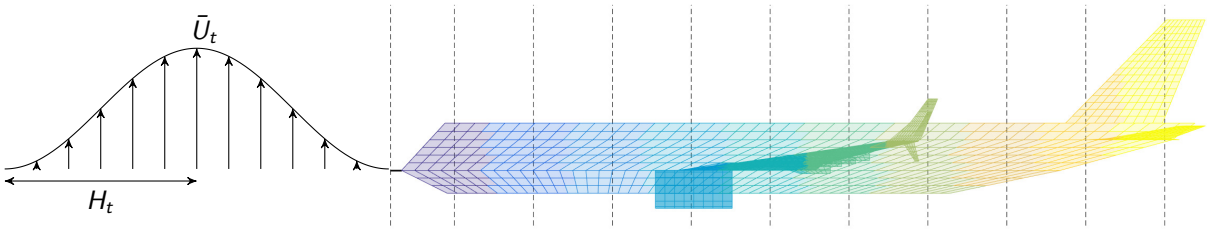

Figure 9: 1-cosine gust and aircraft gust zones.

reference aircraft is separated in gust zones as indicated by the different colours of the aerodynamic panel model. All panels within the same gust zone experience the gust velocity observed at the centre line defined at position x_z. It is indicated by the dashed vertical lines. Namely, within a gust zone the gust velocity is constant. The air data boom, which is at the nose of the aircraft, is specially treated through the introduction of another gust zone. Changes in the angle of attack α are recognised at the air data boom first. For GLA control this offers the opportunity to use a feedforward path [17]. The gust zone approach is an approximation. It saves computation time as it groups many aerodynamic panels into few zones. With ten gust zones the implementation was found to be accurate enough [10]. The gust velocity difference of two neighbouring zones is a simple time delay which is a function of the airspeed U_∞. As a transfer function a time delay is

$$G_{z,d}(s) = e^{-t_{z,d}s}, \tag{18}$$

where $t_{z,d}$ is the time delay in seconds and s is the Laplace variable [10]. The second-order Padé approximation of Equation (18) is

$$G_{z,d}(s) \approx \frac{s^2 - \frac{6}{t_{z,d}}s + \frac{12}{t_{z,d}^2}}{s^2 + \frac{6}{t_{z,d}}s + \frac{12}{t_{z,d}^2}}. \tag{19}$$

It converts to a linear state-space system with the additional states $x_{z,d}$ [8]. Thus, the inputs to the gust zones reduces to the inputs $U_{z,g}$ and $\dot{U}_{z,g}$ at the air data boom. The gusts then propagate over all gust zones.

4.4 GLA Control Synthesis

The ”GLA and MLA Control Synthesis” block uses the set of state-space models for different flight conditions and mass cases. The relevant inputs and outputs of the state-space models are selected and model order reduction and normalization are performed. Subsequently, a GLA controller is synthesized by means of model predictive control (MPC). With the resulting controllers the loads analysis is redone.

4.4.1 Idea of MPC

In Figure 10 the general principle of MPC is described. MPC manipulates a system so that certain

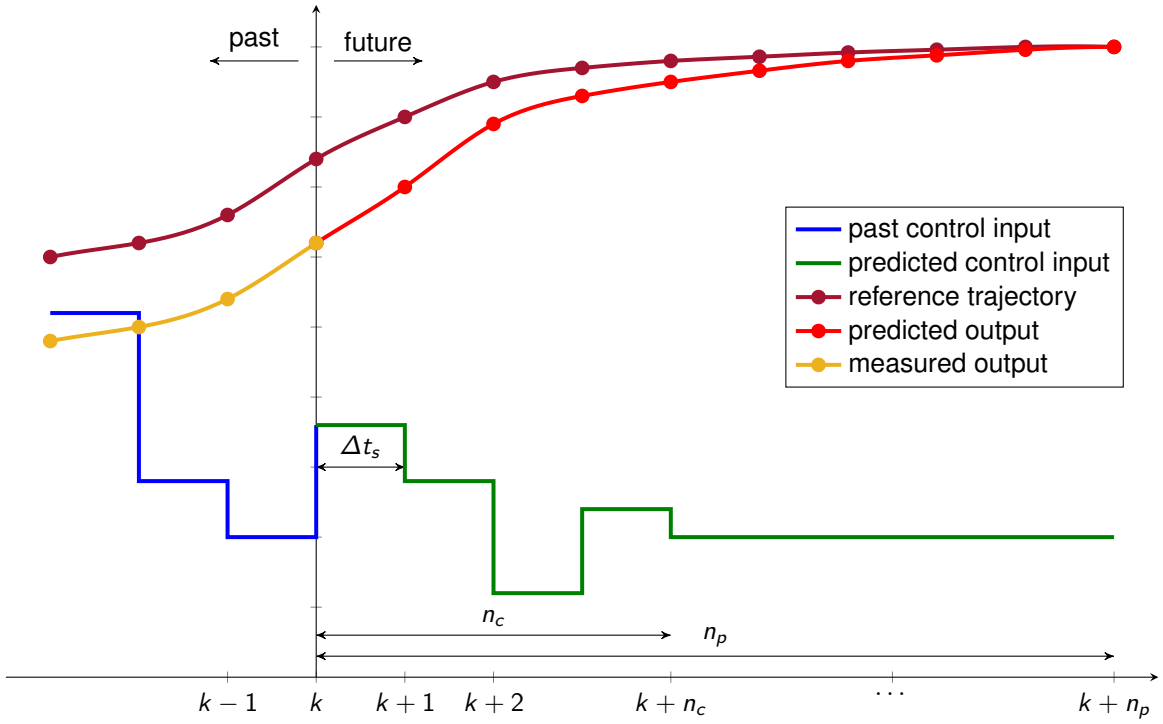

Figure 10: MPC principle [15].

outputs follow a predefined trajectory. Based on a plant model, MPC predicts at the k^{th} time step the output behaviour n_p time steps into the future, where n_p is the prediction horizon. It then optimises the input signals to achieve the desired trajectory. The change in input signals is optimised for n_c time steps and is constant for time steps between $k + n_c$ and $k + n_p$ [4]. Subsequent to the optimisation, MPC applies the first predicted control input increment. As soon as a time step of Δt_s has passed and the time $k + 1$ is reached, the prediction and control horizon are shifted by Δt_s and the optimisation is repeated [15].

4.4.2 Gust Load Alleviation with Model Predictive Control

The four ailerons of each wing and the elevators are used for GLA. The control surfaces are highlighted in magenta in Figure 11. Symmetric allocation of the control surfaces on the left and right side is possible as only vertical gust cases are taken into account. The outputs fed to the GLA controller are the α_a measurement at the air data boom, the z-accelerations and x-rotational rates taken from the fuselage IMU and the most inner and outer IMUs at the rear spar of each wing. The WRBM $P_{c,\text{mx}}$ is defined as an unmeasured output. MPC then reconstructs its value based on the given measurements.

Different mass cases, flight conditions and gust half lengths H_t are of interest for the reference aircraft. They differ in mass properties ranging from empty to the maximum take-off mass and different CG positions.
The relevant flight conditions were pinpointed down to three combinations of flight speeds U_∞ and altitudes h. Seven combinations of different gust half lengths H_t and maximum gust speeds $\bar{U}_t$ are considered for analysis [6]. For the nine mass cases and three flight conditions 27 linearised models

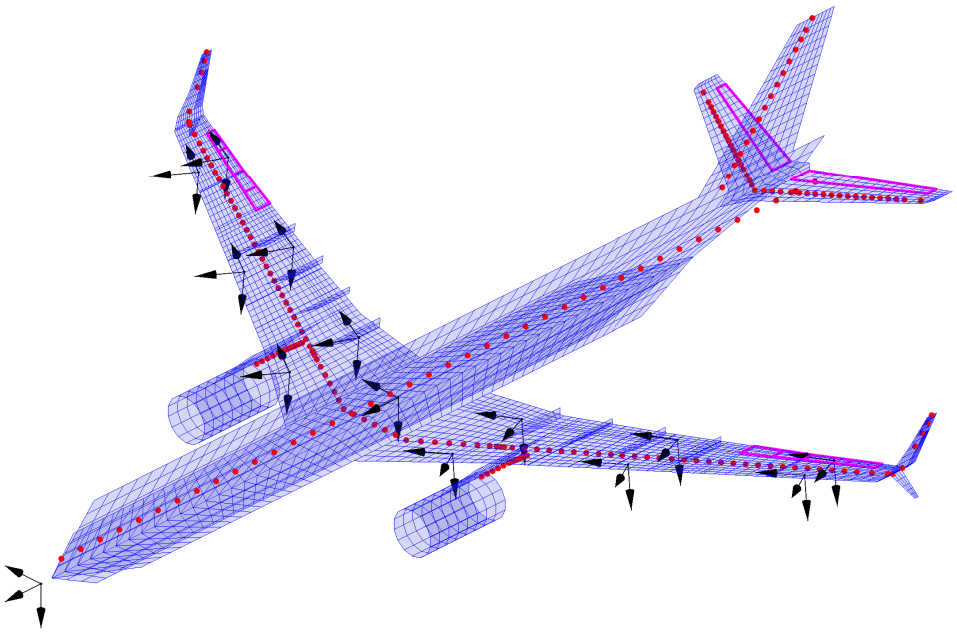

Figure 11: Reference flexible aircraft model defined by the structural grid (red), the aerodynamic panel model (blue), the deployed control surfaces for GLA (magenta) and the sensor coordinate system locations and orientations (black).

are collected. Subsequently, for each model seven different gust encounters are simulated resulting in 189 simulation cases for GLA.

In Figure 12 the effectiveness of the GLA controllers for mass case number five at different flight conditions is given. The WRBM includes the increment induced by the gust encounters and the steady trim value. In the shown case the WRBM can be reduced by roughly $13\,\%$ between open- and closed-loop. In order to examine the effect to the wing root torsional moment (WRTM) $P_{c,\mathrm{my}}$ in comparison to the WRBM $P_{c,\mathrm{mx}}$, in Figure 13 the maximum and minimum values of the WRTM are plotted over the maximum and minimum WRBM of 189 simulations. The values for the WRBM and the WRTM are represented relative to the maximum open-loop WRBM $\overline{\mathrm{WRBM}}$ over all considered cases. The grey diamonds represent the trim conditions. The blue circles and the green triangles are the open- and closed-loop results with gust encounter. All three data sets are enclosed by their convex hull. The maximum WRTM over the simulated gust encounters increases due to control surface deflection in closed-loop compared to open-loop. However, it changes by around $1\,\%$, which is insignificant. The maximum WRBM is reduced by $10\,\%$.

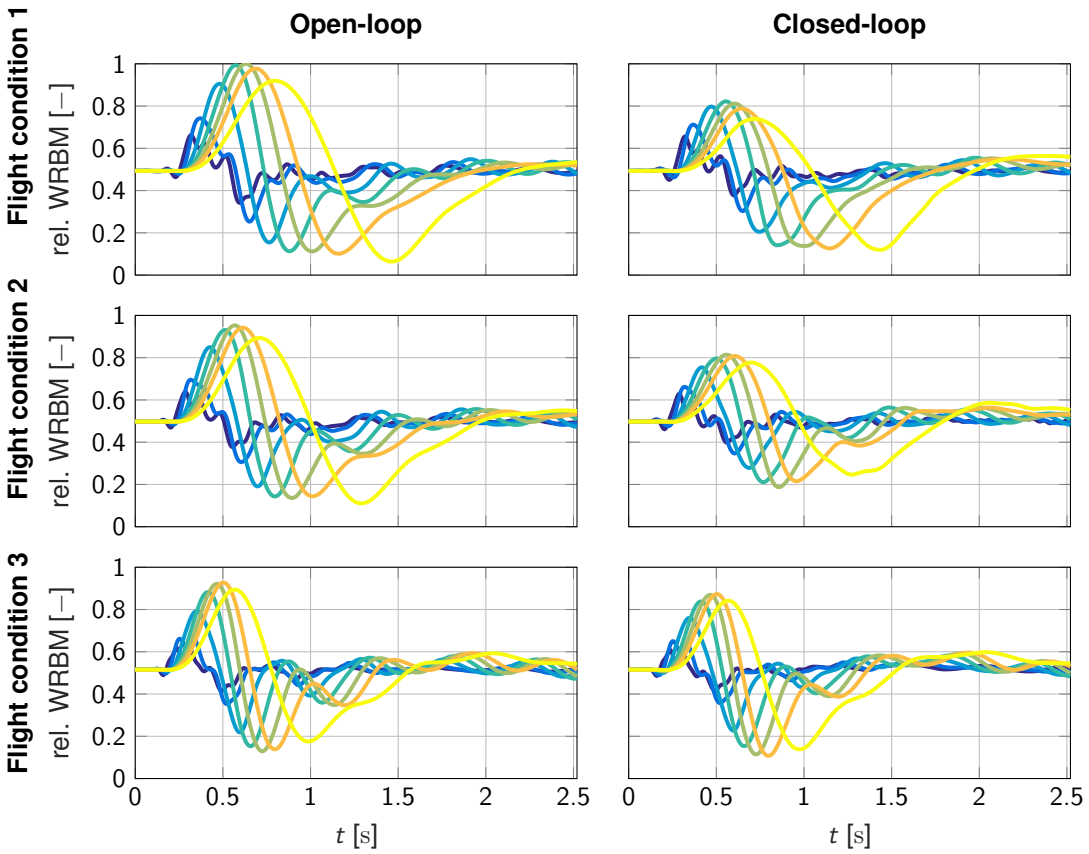

Figure 12: Open- and closed-loop gust simulations of the relative WRBM of the reference aircraft for mass case 5 and gust half lengths $9\,\mathrm{m}$ to $107\,\mathrm{m}$.

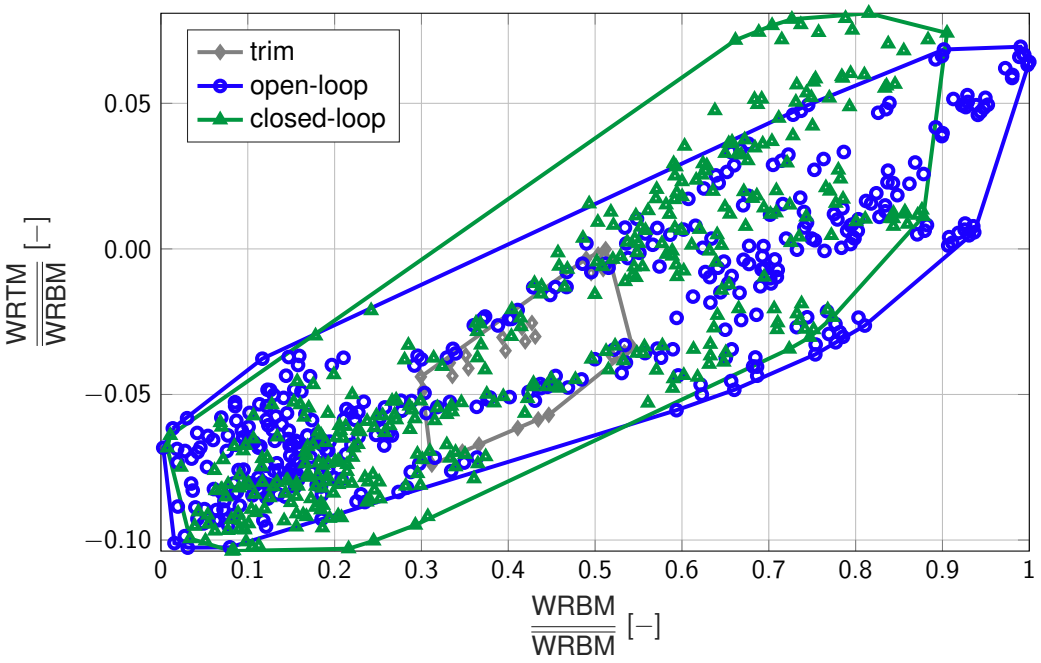

Figure 13: Relative WRTM over relative WRBM.

5 Flutter control layout and design

Flutter control is envisioned to be a feasibility check during conceptual design stage to be able to provide proff that the 15% flutter margin defined in CS-25 regulations can be decreased, by taking credit for active control systems.

5.1 Reduced order modelling for flutter control

Modeling block inputs
The modeling block takes the structural dynamics (M_{hh}, K_{hh}, B_{hh}) and aerodynamics data (Q_{hh}) as input via CPACS.

Modeling block main algorithms

The control oriented models are based on the linear parameter-varying (LPV) framework, [16, 3]. The LPV framework can serve as a good approach to model aeroservoelastic (ASE) systems for control design. The benefits of utilizing the LPV framework are the following; it can capture the parameter varying dynamics of the aircraft and many of the linear time-invariant (LTI) control design techniques have been extended to LPV systems. An LPV system is described by the state space model [20, 16]

$$\dot{x}(t) = A(\rho(t))\ x(t) + B(\rho(t))\ u(t) \tag{20a}$$
$$y(t) = C(\rho(t))\ x(t) + D(\rho(t))\ u(t) \tag{20b}$$

with the continuous matrix functions $A\colon \mathcal{P} \to \mathbb{R}^{n_x \times n_x}$, $B\colon \mathcal{P} \to \mathbb{R}^{n_x \times n_u}$, $C\colon \mathcal{P} \to \mathbb{R}^{n_y \times n_x}$, $D\colon \mathcal{P} \to \mathbb{R}^{n_y \times n_u}$, the state $x\colon \mathbb{R} \to \mathbb{R}^{n_x}$, output $y\colon \mathbb{R} \to \mathbb{R}^{n_y}$ input $u\colon \mathbb{R} \to \mathbb{R}^{n_u}$, and a time-varying scheduling signal $\rho\colon \mathbb{R} \to \mathcal{P}$, where $\mathcal{P}$ is a compact subset of $\mathbb{R}^N$. The system is called quasi LPV model if the parameter vector ρ includes elements of the state vector x. The system matrix $S(\rho(t))$ is defined as

$$S(\rho(t)) = \begin{bmatrix} A(\rho(t)) & B(\rho(t)) \\ C(\rho(t)) & D(\rho(t)) \end{bmatrix} \tag{21}$$

In a grid-based LPV representation ([20]), the system is described as a collection of LTI models (A_k, B_k, C_k, D_k) $= (A(\rho_k), B(\rho_k), C(\rho_k), D(\rho_k))$ obtained from evaluating the LPV model at a finite number of parameter values $\{\rho_k\}_1^{n_{\text{grid}}} = \mathcal{P}_{\text{grid}} \subset \mathcal{P}$.

The main milestones of the modeling block are the following. The ASE model is formed by combining the structural dynamics model, the aerodynamics model and the flight mechanics model. In order to obtain an ASE model suitable for control design, model order reduction needs to be carried out. The model order reduction is based on the bottom-up modeling approach, [18, 13, 21].

The key idea of the bottom-up modeling is the following. The subsystems of the ASE model in general have simpler structure than the nonlinear ASE model. Therefore, the subsystems containing the structural dynamics and aerodynamics model can be reduced by simpler, more tractable reduction techniques. Combining these reduced order subsystems results in a low order nonlinear ASE model upon which a nominal, low order, control oriented models can be obtained. The main measure of the accuracy of the low order model is the ν-gap metric, [19]. The control design is using the linear parameter-varying (LPV) framework, [16, 3]. Therefore grid-based LPV models need to be obtained via Jacobian linearization.

Reduction of the structural dynamics model

The structural dynamics of the aircraft are of the form

$$\mathcal{M}\ddot{\eta} + \mathcal{C}\dot{\eta} + \mathcal{K}\eta = F_{\text{modal}} \tag{22}$$

where F_{modal} is the force acting on the structure in modal coordinates, $\mathcal{M}$, $\mathcal{C}$ and $\mathcal{K}$ are the modal mass, damping and stiffness matrices respectively. The structural dynamics model is an LTI system, thus state truncation can be applied.

Reduction of the aerodynamics model

The aerodynamic lag terms take the state-space form

$$\begin{aligned} \dot{x}_{\text{aero}} &= \frac{2V_{\text{TAS}}}{\bar{c}} A_{\text{lag}} x_{\text{aero}} + B_{\text{lag}} \begin{bmatrix} \dot{x}_{\text{rigid}} \\ \dot{\eta} \\ \dot{\delta}_{\text{cs}} \end{bmatrix} \\ y_{\text{aero}} &= C_{\text{lag}} x_{\text{aero}} \end{aligned} \tag{23}$$

where V_{TAS} is the true airspeed, x_{rigid} is the rigid body state, η is the modal state of the structural dynamics, δ_{cs} is the control surface deflection and $\bar{c}$ is the reference chord. Using the aerodynamics model given by A_{lag}, B_{lag} and C_{lag} in (23) an LTI balancing transformation matrix T_{b} is computed. The balanced states of the aerodynamic model with the smallest Hankel singular values are residualized, leading to a reduced order aerodynamics model.

Uncertain low order model

The next step is to develop uncertain LPV models of the aircraft. Uncertain models can be developed by extending the structural dynamics model with the uncertain parameters. These uncertainties appear in the mass matrix $\mathcal{K}$ and in the damping matrix $\mathcal{C}$ in (22) of the nonlinear ASE model and are denoted by $\delta_{\mathcal{K}}$ and $\delta_{\mathcal{C}}$, respectively. Based on this uncertain, nonlinear model a grid-based uncertain LPV model is constructed. The grid-based uncertain LPV model is obtained over a 3 dimensional grid. The grid consists of equidistant points of the airspeed, 3 points of the natural frequency in the structural dynamics between $\pm 1\%$ of the nominal value, and 3 points of the damping in the structural dynamics between $\pm 10\%$ of the nominal value. The scheduling parameter ρ can then be defined as

$$\rho = \begin{bmatrix} \rho_{V_{TAS}} \\ \delta_{\mathcal{K}} \\ \delta_{\mathcal{C}} \end{bmatrix} \tag{24}$$

where $\rho_{V_{TAS}}$ is a measured parameter and $\delta_{\mathcal{K}}$ and $\delta_{\mathcal{C}}$ are unmeasured. These uncertainties have a significant effect on the flutter speeds and frequencies.

Modeling block robustness

As it can be seen, the bottom-up modeling approach involves a certain degree of heuristics. These heuristic steps include the selection of the structural dynamics states to retain and setting the the number of retained aerodynamic lag states. These parameters are hand tuned for the initial, reference aircraft model. The modeling tool needs to be adopted to the collaborative design in this respect. This means that the retained the initial structural modes to be retained are the ones of the reference aircraft. However, it is crucial that after every MDO iteration, the ν-gap metric is analyzed and that it does not exceed a threshold value. If this value is exceeded, it means that the bottom up-model is not accurate enough. Therefore, at the expense of increasing the order ot the resulting model, additional structural

modes need to be retained. The number of retained modes is increased until the ν-gap values are satisfactory. A similar approach is used for the order of the lag state aerodynamics model. In this case the number of the retained lag states is increased until a satisfactory ν-gap level is obtained.

Modeling block outputs

The modeling block provides the model for the flutter control design (FlexACModel). The FlexACModel is the low order, uncertain LPV model of the aircraft obtained by the steps described above. The resulting models are saved in the ToolSpecific section of CPACS.

5.2 Flutter control design

Flutter control design inputs

The flutter control design takes the outer aileron (denoted by L4 and R4) actuator dynamics and the flutter control design model FlexACModel as inputs via CPACS.

Flutter control design main algorithm

There are two main sub blocks in the flutter control design blocks. First, the design model is split into longitudinal and lateral. These models are then used to synthesize a stabilizing controller for the symmetric and asymmetric flutter mode respectively. Second, the control design consists of the construction of two uncertain plants, performance definitions, and the synthesis of two low-order controllers. These controllers are blended together to obtain the flutter controller. The stability of the resulting flutter controller and a couple of implementation criteria are also tested.

In case of the flutter suppression control desing, the airspeed and the uncertainties in the structural dynamics model are treated at parametric uncertainties and dynamic uncertainty is added to account for the model reduction. In order to reduce the computational time of the control synthesis, structured H_∞ design is chosen that result in an LTI flutter suppression controller. Similarly to the baseline control design algorithm, the flutter suppression control design block needs to be augmented with basic analysis algorithms to verify if the resulting controller satisfies the control performance specifications. As a main measure, the multi-input multi-output (MIMO) disc margins are selected.

Flutter control design outputs

The output of the block is the flutter suppression controller, saved via CPACS.

5.2.1 Closed loop analysis block

Closed loop analysis block inputs

The analysis block requires the baseline, the flutter controllers, the actuator and flexible aircraft dynamics. All the input data is handled via CPACS.

Closed loop analysis block main algorithm

The open and closed-loop flutter speed are determined by multi-loop input-output margin computations.

The open+loop flutter speed is the speed below which the margins obtained with only the baseline controller engaged are larger than a predefined threshold. The closed-loop flutter speed is defined similarly with both the baseline and the flutter controllers engaged.

Closed loop analysis block outputs

The closed loop analysis block provides the symmetric and asymmetric gain margins and the analysis runtime results as the main indicators for the designed controller's performance. The data are saved via CPACS.

5.2.2 Report generation block

A pdf report is generated containing key information about the synthesis and analysis: the open and closed-loop flutter speeds, the robustness margins, the gain of the controller, etc.

6 Conclusion

The deliverable D4.2 shows how the control system functions are integrated into the design objectives of the MDO process.

The overall scale-up design objective is to maximize the range of the chosen D150 aircraft. The combinations of the different technologies - aeroelastic tailoring, GLA, MLA, AFS and wingshape control - are demonstrated to provide additional design freedom during the conceptual design phase.

The process for the control system and feedback laws design are discussed from both ONERA? DLR and SZTAKI side, focusing on different aspects. As a specificity of the FLiPASED project, which aims at encapsulating the control design in the aircraft MDO process, this report details how such a control laws may be as much as possible automatically computed to satisfy the flight and safety requirements. The present report focuses then on the automated control design process required in the iterative execution of a conceptual phase design. These features are classical functionalities that are expected in most of the long range aircraft and are therefore crucial. The report sequentially details all these three features. Here attention is given to the methodology development and the numerical robustness / versatility of the proposed process rather than on a specific performance oriented control tuning. Indeed, ultimately, the tuning of the macro parameters of each single control function shall be performed in view of a global objective under constraints, mastered by the MDO algorithm later during the detailed design stages.

7 Bibliography

[1] A.C. Antoulas. *Approximation of large-scale dynamical systems*. SIAM, Philadelphia, 2005.

[2] A.C. Antoulas, C.A. Beattie, and S. Gugercin. *Interpolatory methods for model reduction*. SIAM Computational Science and Engineering, Philadelphia, 2020.

[3] G. Becker. *Quadratic stability and performance of linear parameter dependent systems*. PhD thesis, University of California, Berkeley, 1993.

[4] Alberto Bemporad, N. Lawrence Ricker, and Manfred Morari. Model predictive control toolbox user's guide. 2022.

[5] Raymond L. Bisplinghoff, Holt Ashley, and Robert L. Halfman. *Aeroelasticity*. Dover Publications, Inc., 1955.

[6] European Aviation Safety Agency. Certification specifications for large aeroplanes (cs-25). 2007.

[7] I.V. Gosea, S. Gugercin, and C. Beattie. Data-driven balancing of linear dynamical systems. *arXiv preprint arXiv:2104.01006*, 2021.

[8] Vladimir Hanta and Aleš Procházka. Rational approximation of time delay. 2009.

[9] Jeroen Hofstee, Thiemo Kier, Chiara Cerulli, and Gertjan Looye. A variable, fully flexible dynamic response tool for special investigations (varloads). In *International Forum on Aeroelasticity and Structural Dynamics*, 2003.

[10] M. Karpel, B. Moulin, and P. C. Chen. Dynamic response of aeroservoelastic systems to gust excitation. *Journal of Aircraft*, 42(5):1264–1272, 2005.

[11] Thiemo Kier and Gertjan Looye. Unifying manoeuvre and gust loads analysis models. In *International Forum on Aeroelasticity and Structural Dynamics*, 2009.

[12] F. Leibfritz and W. Lipinski. Description of the benchmark examples in $COMP_{e}ib$ 1.0. Technical report, University of Trier, 2003.

[13] Yasser M. Meddaikar, Johannes Dillinger, Thomas Klimmek, Wolf Krueger, Matthias Wuestenhagen, Thiemo M. Kier, Andreas Hermanutz, Mirko Hornung, Vladyslav Rozov, Christian Breitsamter, James Alderman, Bela Takarics, and Balint Vanek. Aircraft aeroservoelastic modelling of the FLEXOP unmanned flying demonstrator. In *AIAA Scitech 2019 Forum*. AIAA, jan 2019.

[14] C. Poussot-Vassal, P. Kergus, and P. Vuillemin. *Realization and Model Reduction of Dynamical Systems, A Festschrift in Honor of the 70th Birthday of Thanos Antoulas (eds. C. Beattie, P. Benner, M. Embree, S. Gugercin and S. Lefteriu)*, chapter Interpolation-based irrational model control design and stability analysis, pages 353–371. Springer, 2022.

[15] Max Schwenzer, Muzaffer Ay, Thomas Bergs, and Dirk Abel. Review on model predictive control: an engineering perspective. *The International Journal of Advanced Manufacturing Technology*, 117(5-6):1327–1349, 2021.

[16] J. S. Shamma. *Analysis and design of gain scheduled control systems*. PhD thesis, Massachusetts Institute of Technology, Cambridge, 1988.

[17] Sigurd Skogestad and Ian Postlethwaite. *Multivariable Feedback Control: Analysis and design*. John Wiley and Sons, 2 edition, 2005.

[18] Bela Takarics, Balint Vanek, Aditya Kotikalpudi, and Peter Seiler. Flight control oriented bottom-up nonlinear modeling of aeroelastic vehicles. In *2018 IEEE Aerospace Conference*. IEEE, mar 2018.

[19] G. Vinnicombe. *Measuring Robustness of Feedback Systems*. PhD thesis, Univ. Cambridge, Cambridge, 1993.

[20] Fen Wu. *Control of Linear Parameter Varying Systems*. PhD thesis, Univ. California, Berkeley, 1995.

[21] Matthias Wuestenhagen, Thiemo Kier, Yasser M. Meddaikar, Manuel Pusch, Daniel Ossmann, and Andreas Hermanutz. Aeroservoelastic modeling and analysis of a highly flexible flutter demonstrator. In *2018 Atmospheric Flight Mechanics Conference*. AIAA, jun 2018.

[22] K. Zhou and J C. Doyle. *Essentials Of Robust Control*. Prentice Hall, 1997.