FliPASED
WWW.FLIPASED.EU

D2.5 Aero-servo-structural design of the new advanced FliPASED wing

Fanglin Yu (TUM), Thiemo Kier, Yasser Meddaikar, Matthias Wuestenhagen (DLR), Balint Vanek, Bela Takarics (SZTAKI), Charles Poussot-Vassal (ONERA)

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 12/04/2019
Start date of project: 13/12/2022	**Duration:** 46 Months

Lead Beneficiary for this deliverable:	SZTAKI
Last modified: 28/06/2023	**Status:** Delivered
Due date: 30/09/2021	

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

ASE	Aeroservoelastic
CAD	Computer-aided Design
CPACS	Common Parametric Aircraft Configuration Schema
DLM	Doublet Lattice Method
FEM	Finite Element Model
SW	Soft-ware
HW	Hard-ware
VV	Verification and Validation
GLA	Gust Load Alleviation
MLA	Manoeuvre Load Alleviation
MDO	Multidisciplinary Design Optimization
PID	Proportional-Integral-Derivative
RCE	Remote Component Environment
HIL	Hardvare-in-the-loop
FCC	Flight Control Computer
TCL	Tool Command Language
FLEXOP	Flutter Free FLight Envelope eXpansion for ecOnomical Performance improvement
LPV	Linear Parameter-varying
LTI	Linear Time-invariant
BT	Balanced Truncation
LF	Loewner Framework
HSV	Hankel Singular Values
SV	Singular Values
ROM	Reduced Order Model
FOM	Full Order Model
WRBM	Wing Root Bending Moment
CFD	Computational Fluid Dynamics
CSM	Computational Structural Mechanics
DOF	Degree of Freedoms
WSC	Wing Shape Control
VLM	Vortex Lattice Method

Table of contents

List of Figures

1 Executive Summary

This deliverable will explain where does the retrofit plan comes from. Decision making process of wing selection for the retrofit is documented. Design study is conducted for different number of control surfaces to see which one is most suitable for the project goal (wing shape control). All the control law design for the retrofit wing is also documented in this deliverable.

2 Plan of Retrofit Wing

According to the initial plan, the new aero-servo-structural wing (-3 Wing) should be designed by the Multidisciplinary Design Optimization(MDO) toolchain. But the setup and validation of the MDO toolchian takes more time than the initially planned. In order to avoid further delay for the manufacturing and flight testing of new wing, the consortium decided to retrofit an old wing to decouple the MDO progress from the manufacturing and flight testing.

A feasibility check was conducted to make sure the retrofit plan viable from mechanical aspects.The retrofit wing should still be able to demonstrate the capabilities of maneuver load alleviation, gust load alleviation and wing shape control based drag reduction.

3 Selection of Retrofit Wing

This section will give brief introduction of all available wings and explain how was the retrofit wing selected.

3.1 Description of wings

3.1.1 -0 and -2 wings

The 'reference' or -0 wings and 'tailored' or -2 wings from the Flutter Free FLight Envelope eXpansion for ecOnomical Performance improvement (FLEXOP) project were considered for the retrofit design study. A selection between the two wings was to be made based on the higher potential of demonstrating induced drag reduction during cruise flight. Additionally, the existing 4 control surfaces on the chosen wing were to be replaced by a larger number of control surfaces, aimed at demonstrating better drag reduction. The design of the two wing pairs is presented in [23, 27] and is the outcome of an aeroelastic tailoring design toolchain used within the project. The term reference here denotes that the wing was designed using conventional industry-near balanced symmetric laminates as against its counterpart, the 'tailored' wing which demonstrated higher passive load alleviation through composite tailoring.

The planform of the wing is shown in Figure 1. The existing control surface layout consists of 4 equally-spaced control surfaces starting from 12% upto 98% of the wing semi-span.

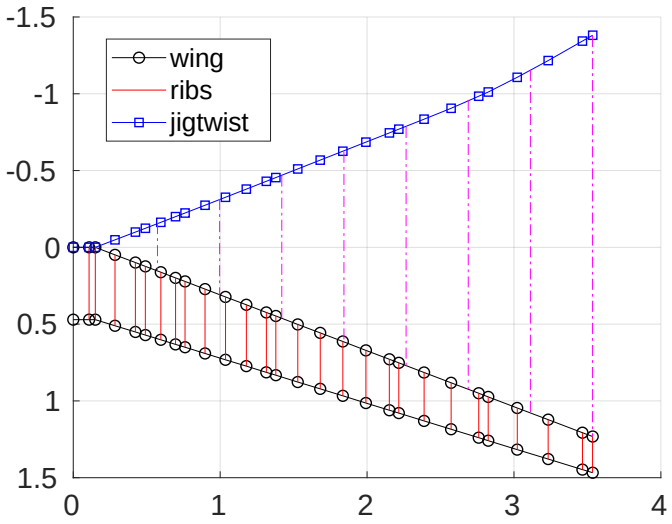

Figure 1: Planform of the reference wing used in this study

3.1.2 -1 Wing

-1 wing or flutter wing was designed in FLEXOP project in order to demonstrate the flutter suppression control algorithm. Because of constraints in concept of operations, special design study was conducted to bring down the flutter speed to make the flight testing of such wing feasible. For more design details please refer to the FLEXOP deliverables [11]. Because the intrinsic characteristics of -1 wing is very flexible, so -1 wing is ruled out for retrofitting.

3.2 Design study to select the retrofit wing

For this study, different potential flow tools and methods were considered for the estimation of wing induced drag, on the basis of the T-FLEX demonstrator wing - a Trefftz plane implementaiton in NASTRAN, a Vortex Lattice Method (VLM)-based pressure-integration approach, drag estimation using the PANUKL code, AVL [5] and Tornado [28]. The results presented in this section are obtained from the VLM-based pressure integration approach.

For this study, each of the existing four control surfaces on the wings were split into four control surfaces, that is 16 in total, in the simulation model. The drag minimization problem was solved for the reference and tailored wings at 1g trimmed flight for different flight points. The improvement in induced drag compared to the clean configuration (control surfaces at 0° deflection) attainable, for each of the respective wings is shown in Table 1. From this study, the following key inferences can be made.

- At the design speed of $45m/s$, both wing pairs show a relatively less benefit of using active control for induced drag reduction. This is because the aerodynamic characteristics of the wings developed earlier during the project were aimed at this design speed. Consequently, the potential of drag reduction using active control is less at these flight points.
- In both wing pairs, the drag reduction potential is more prominent at flight speeds exceeding the design speed, where the wing deformation is larger and hence the potential for better wing shape control.
- The reference or -0 wing shows a larger potential at demonstrating drag reduction when compared to the tailored or -2 wing. The tailored wing was designed to demonstrate passive load alleviation through composite tailoring. The inherent tendency of the wing to induce washout and shift loads towards the root results in a lift distribution that favors lower induced drag.

The lift distribution with and without active control together with the optimal control surface allocation at $50m/s$ is shown in Figures 2 and 3. The optimal control surface deflections corresponding to this flight point in the case of the reference wing are visualized in Figure 4.

Table 1: Comparison of induced drag improvement between reference and tailored wings (induced drag improvement is defined between clean and optimal control surface deflections of the respective wing)

	$20m/s$	$30m/s$	$40m/s$	$45m/s$	$50m/s$	$60m/s$
reference (-0) wings	3.6%	2.5%	4.3%	6.7%	9.9%	17.3%
tailored (-2) wings	4.8%	4.6%	4.6%	4.9%	5.5%	7.6%

3.3 Decision of retrofit wing

Consequently, the choice was made to use the reference wing for the flight tests. In order to further increase the visibility of a reduction in drag through active control, design studies on modifying the deformation of the wing such that the wing is further away from its design point could be considered, for instance by adding a suitable mass at a favorable position of the wing, favorable in the sense of increasing demonstrability of drag reduction, which is the focus of the exercise.

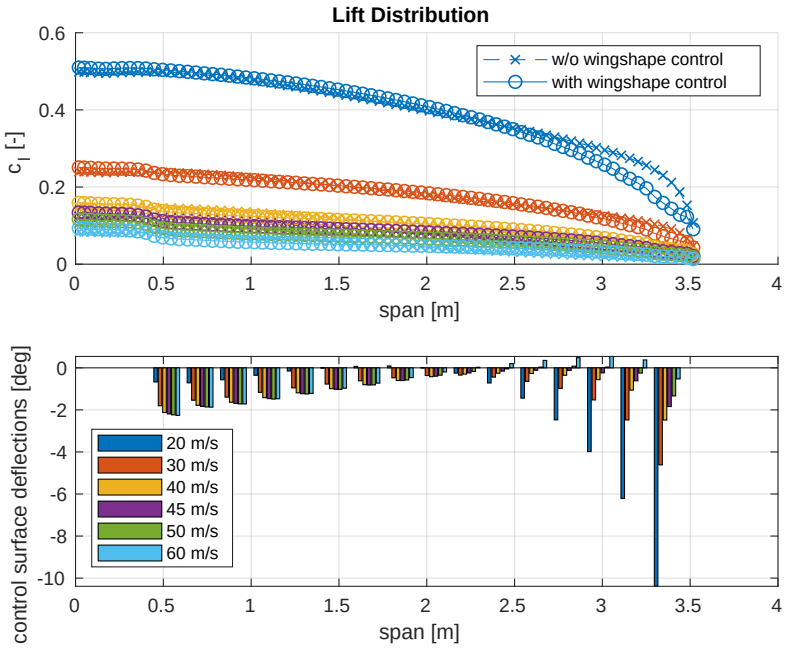

Figure 2: Lift distribution for the -0 reference wing without and with active control (above), optimal spanwise control surface deflections for minimum induced drag (below)

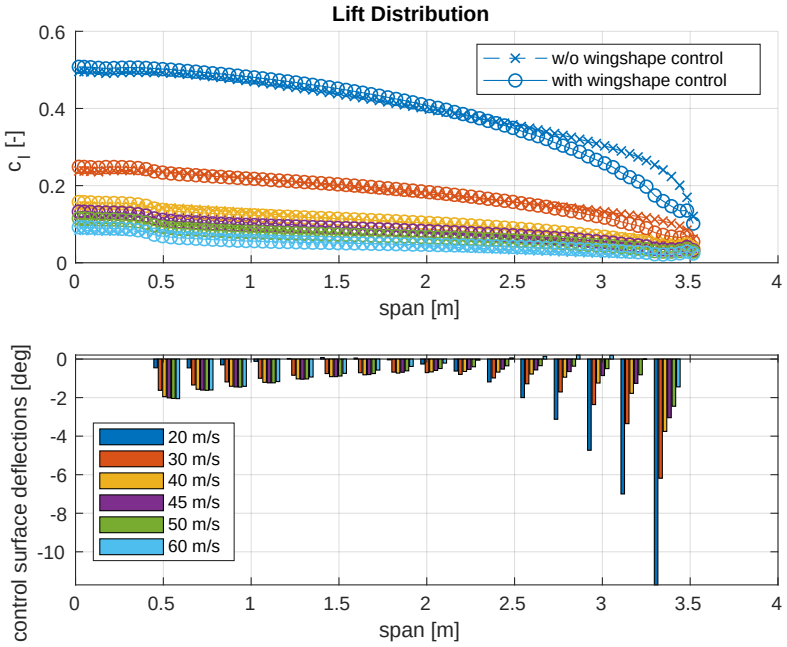

Figure 3: Lift distribution for the -2 tailored wing without and with active control (above), optimal spanwise control surface deflections for minimum induced drag (below)

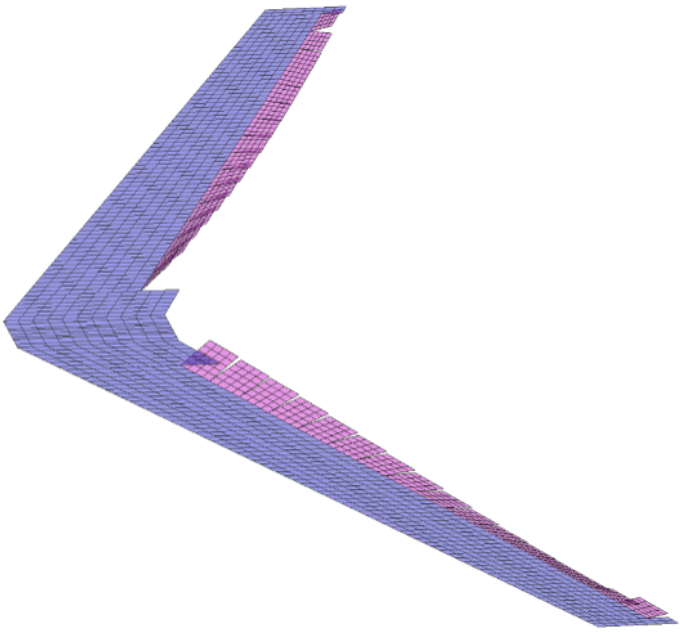

Figure 4: Optimal control surface deflections for the reference -0 wing at $50 m/s$ cruise (20x exaggerated control deflections)

4 Design Study for Number of Flaps

The optimal control surface allocation for the reference wing for minimal induced drag is shown in Figure 2. As mentioned earlier, the simulation here assumes that each of the four existing control surfaces is split into four independent control surfaces. It is seen that close to the root and the mid-span of the wing, the variation in control surface deflection is quite low. The strongest gradients in the deflection are visible at the outboard sections of the wing.

An efficient solution would be to define smaller discrete control surfaces near the tip section where differences between adjacent control surface deflections are largest. Consequently, after studying different combinations, a layout where the existing four control surfaces from the root to tip are split into 1, 2, 3, 3 control surfaces respectively was chosen. This was done considering the drag reduction potential on the one hand and manufacturing feasibility on the other, keeping in mind the existing hardware and systems on the wing. For more details about manufacturing please refer to the deliverable [20].

5 Control Design for Retrofit Wing

The advanced wing is flight worthy only if it is augmented with the corresponding control laws. The original baseline control law is active on flap 2 and 3 out of the 4 control surfaces, hence the corresponding baseline control with the advanced wing should also use the same wingspan sections. Control allocation on the other hand is a highly non trivial task, due to the interaction of baseline, MLA and GLA functions, and potentially flutter control.

5.1 Baseline control

The baseline control design algorithms have to be constructed in such a way that they can be integrated into the MDO framework. This means that they need to run in an automated way within the MDO iteration steps, i.e. it needs to be insensitive to the model changes that result from the MDO optimization.

The main idea of the baseline control design comes from the description given in Deliverable 1.2 [19].

The most critical considerations that have to be made for the new advanced FLiPASED wing are related to the control surfaces. The initial concept is to combine 4 control surfaces which would move together to get similar aileron and ruddervator surfaces as was the for the -1 wing design. In case there is enough control authority to achieve the desired baseline control goals with the combination of less than 4 flaps, this number can be reduced. That way more free surfaces remain for the active wing shape control, Manoeuvre Load Alleviation (MLA) and Gust Load Alleviation (GLA) control.

5.1.1 Control oriented modeling

Modeling block inputs

The modeling block takes the structural dynamics (M_{hh}, K_{hh}, B_{hh}) and aerodynamics data (Q_{hh}) as input via Common Parametric Aircraft Configuration Schema (CPACS).

Modeling block main algorithms

The control oriented models are based on the linear parameter-varying (LPV) framework, [34, 4]. 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 [37, 34]

$$\dot{x}(t) = A(\rho(t))\ x(t) + B(\rho(t))\ u(t) \tag{1a}$$
$$y(t) = C(\rho(t))\ x(t) + D(\rho(t))\ u(t) \tag{1b}$$

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

$$S(\rho(t)) = \begin{bmatrix} A(\rho(t)) & B(\rho(t)) \\ C(\rho(t)) & D(\rho(t)) \end{bmatrix} \tag{2}$$

In a grid-based LPV representation ([37]), 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, [35, 26, 38].

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 steps are given in Deliverable 1.2 [19].

Modeling block outputs

The modeling block provides a model for the baseline control design, RigACModel. The RigACModel is obtained from the nominal low order aircraft aeroservoelastic model by rezidualizing the structural and lag state dynamics. This model serves for the baseline control design, containing only the 12 rigid body states. The resulting model is saved in the ToolSpecific section of CPACS.

5.1.2 Baseline control design block

Baseline control design inputs

The baseline control design block takes the actuator dynamics and the baseline control design model RigACModel as inputs via CPACS.

Baseline control design main algorithm

The baseline control system features a classical cascade flight control structure with scheduled control loops to augment the lateral and longitudinal axis of the aircraft. As the cross-coupling between longitudinal and lateral axis is negligible, longitudinal and lateral control design is separated. The control loops use scheduled elements of proportional-integral-derivative (PID) controller structures with additional roll-offs in the inner loops to ensure that no aeroelastic mode is excited by the baseline controller. Scheduling with indicated airspeed V_{ias} is used to ensure an adequate performance over the velocity range from 30 m/s to 70 m/s.

The baseline control design needs to be augmented with verification/analysis algorithms that ensure that the resulting controllers after each MDO iteration satisfy the control performance specifications.

Baseline control design outputs

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

5.2 MLA

5.2.1 Introduction

The retained architecture for the MLA module is schematised in figure 5. 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 5.2.2, 5.2.3 and 5.2.4. The whole module is then illustrated in section 5.2.5. The limitations and leads for improvement are then discussed in conclusion.

5.2.2 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:

- 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.

[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

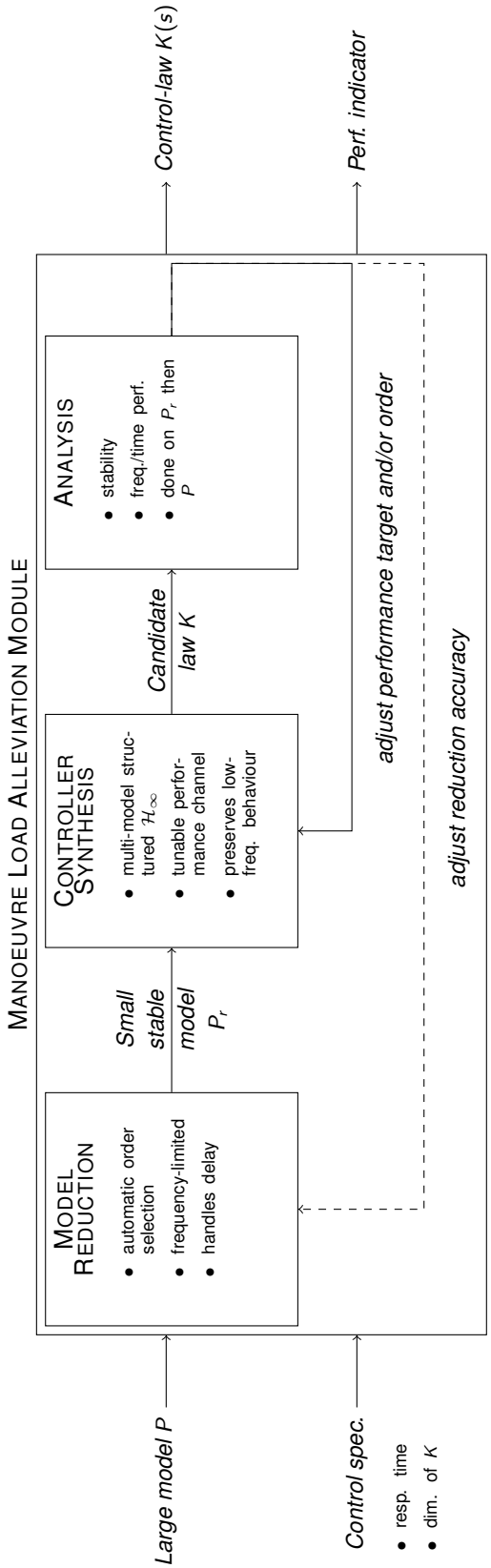

Figure 5: Overall scheme of the automated MLA process

- 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. [39]) 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.
$$e_\infty(r) = \|H - H_r\|_\infty \leq 2\sum_{r+1}^{n} \sigma_i = \overline{e}_\infty(r). \tag{3}$$
- 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.

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.

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, 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.

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

Delays: Due to the modelling of the aerodynamic forces on the fuselage or due to the embedding of other control systems loops (e.g. GLA and flutter), 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.

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 (4), 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 (3) can be completed by the following lower bound,

$$\underline{e}_\infty(r) = \sigma_{r+1} \leq e_\infty(r). \tag{4}$$

The bounds (3) and (4) 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
$$\underline{re}_\infty(r) = \frac{\underline{e}_\infty(r)}{N_\infty} \leq e \quad \text{or} \quad \overline{re}_\infty(r) = \frac{\overline{e}_\infty(r)}{N_\infty} \leq e. \tag{5}$$
- if N_∞ is not available due to the dimension of the model, then one can combines the bounds (3) and (4) with
$$\sigma_1 \leq \|H\|_\infty \leq \|\sigma\|_1, \tag{6}$$
to obtain the following relation for the relative error re_∞
$$\underline{\underline{re}}_\infty(r) = \frac{\underline{e}_\infty(r)}{\|\sigma\|_1} \leq re_\infty \leq \frac{\overline{e}_\infty(r)}{\sigma_1} = \overline{\overline{re}}_\infty(r). \tag{7}$$

Obviously, (7) has an increased conservatism in comparison to (5) 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 [24]: 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 (5) and (7). The ratios of the upper bound with the true relative error is reported in figure 6 and the ratio of the relative error with the lower bound is reported in figure 7.

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 (5) is more accurate (blue dots) than using its bounds in (7) 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 (5) *and* (7) *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 8 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 (5) and (7). These bounds can be combined to derive a mixed criterion, e.g.

$$(1-\alpha)\underline{re}_\infty(r) + \alpha\overline{re}_\infty(r), \tag{8}$$

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 [8] 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

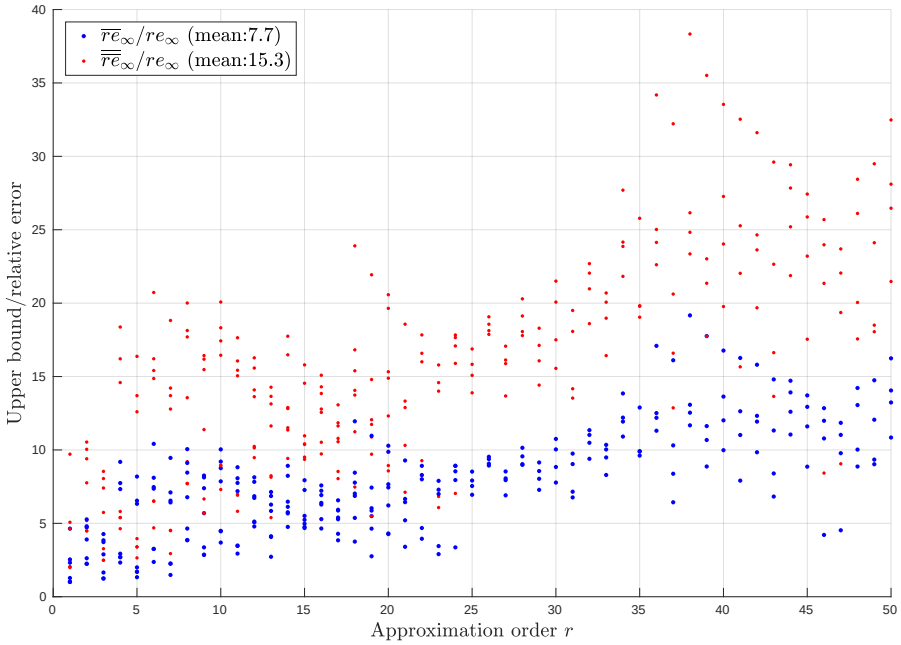

Figure 6: Ratio between the upper bounds of the approximation error and the actual error for various orders and models in $\mathcal{M}$.

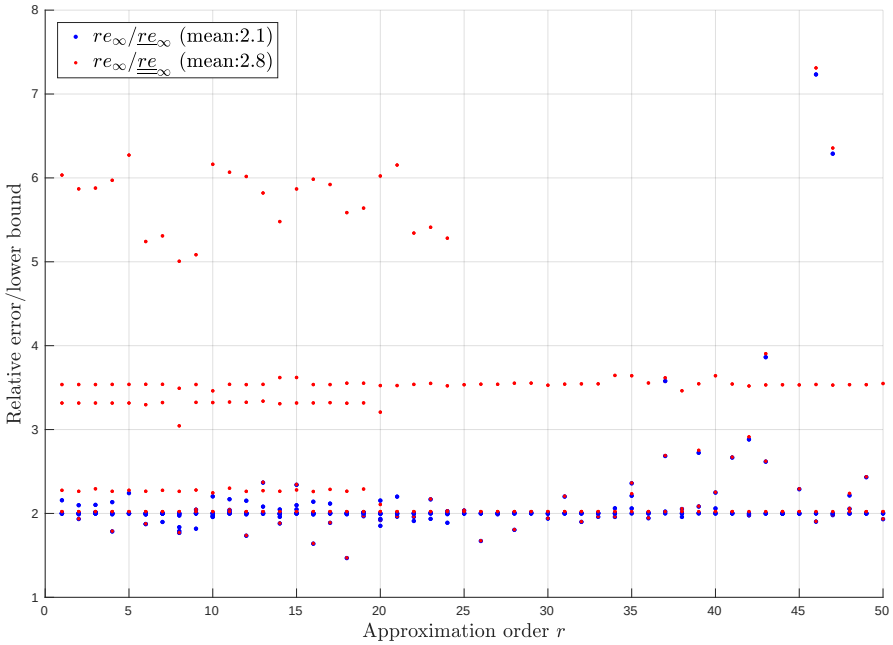

Figure 7: Ratio between the actual approximation error and lower bounds for various orders and models in $\mathcal{M}$.

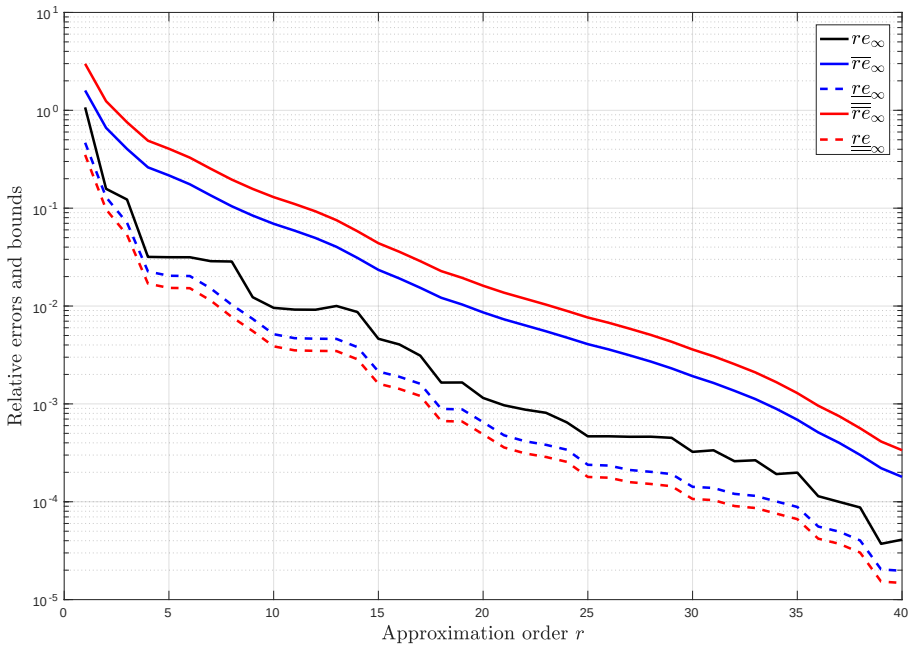

Figure 8: Approximation errors and associated bounds for the MLA model of Flipased.

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-Singular Value (SV).

The second point is considered in [8] 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 [8].

To compare Loewner-HSV with the approach developed in [8], 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 9.

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.

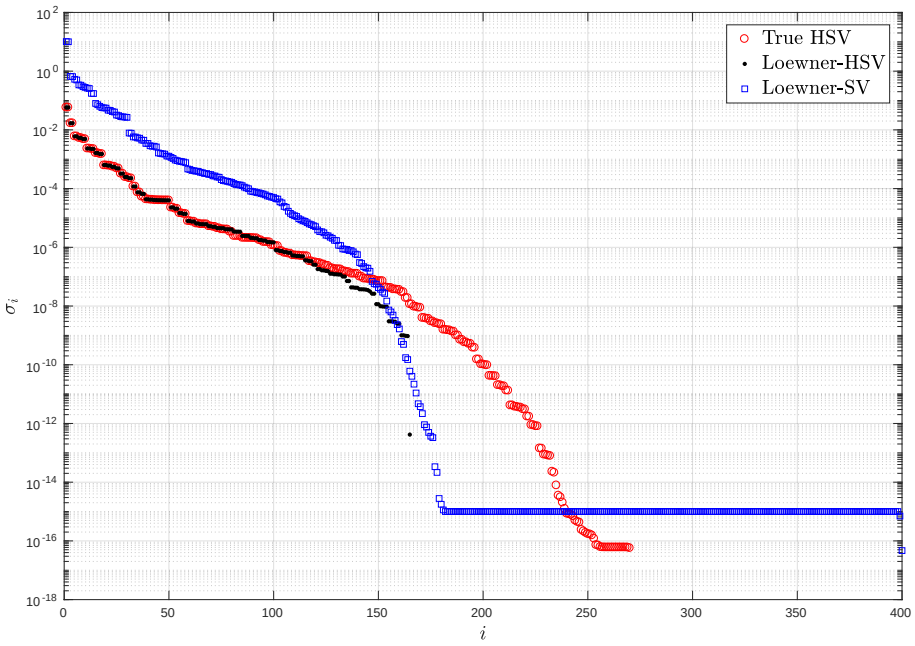

Figure 9: Comparison of the true HSV of the ISS1 model with the Loewner-HSV and the Loewner-SV.

In addition, we retrieve the results from [8] 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 [8].

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

5.2.3 Control synthesis

Pre-processing. Now a Reduced Order Model (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.

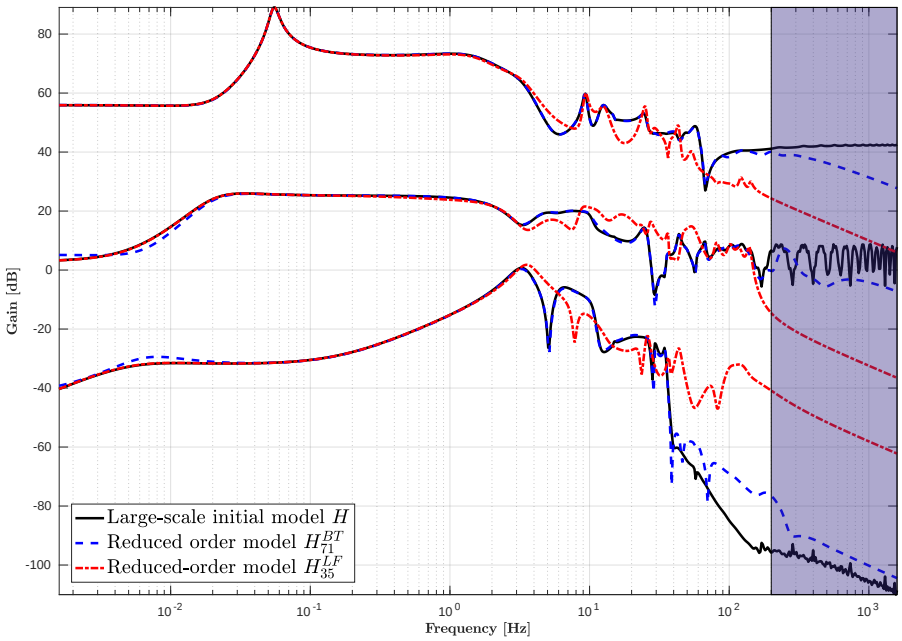

Figure 10: Reduction of the delayed MLA model with the BT and LF approaches.

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:

$$K = \arg \min_{\bar{K} \in \mathcal{K} \subseteq \mathcal{H}_\infty} ||\mathcal{F}_l(G, \bar{K})||_{\mathcal{H}_\infty} \tag{9}$$

under the stabilizing constraint

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

We also denote as γ,

$$\gamma = \min_{\bar{K} \in \mathcal{K} \subseteq \mathcal{H}_\infty} ||\mathcal{F}_l(G, \bar{K})||_{\mathcal{H}_\infty} = ||\mathcal{F}_l(G, K)||_{\mathcal{H}_\infty}. \tag{10}$$

Following (10) 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

$$\begin{array}{rcll} w & = & \text{vec}(a_z^\star, w_g) & \text{(exogenous inputs)} \\ u & = & \text{vec}(u_{a4}, u_r) & \text{(control inputs)} \\ y & = & \text{vec}(\theta, q, a_z^\star - a_z) & \text{(system output measurements for the control)} \\ z & = & \text{vec}(z_1, z_2, z_3) & \text{(performance output, defined hereafter)} \end{array} \tag{11}$$

Then one defines the plant P as

$$\text{vec}(z, y) = P\text{vec}(w, u) \tag{12}$$

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

$$z = \mathcal{F}(P, K)w = P(K)w. \tag{13}$$

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**:

$$z_1 = T_1(K)a_z^\star = W_e(H_r a_z^\star - P_{a_z^\star \mapsto a_z}(K)) \tag{14}$$

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**:

$$z_2 = T_2(K)w_g = W_p P_{w_g \mapsto l}(K) \tag{15}$$

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**:

$$z_3 = T_3(K)y = W_k K y, \tag{16}$$

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

$$T = \textbf{blkdiag}(T_1, T_2, T_3) \tag{17}$$

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, 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.

5.2.4 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 Full Order Model (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 [30], 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.

5.2.5 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.

5.2.6 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.

5.3 GLA

5.3.1 Gust Model

For the GLA controller synthesis the gust input is modelled by a discrete, vertical 1-cosine function. It is defined by the gust zone velocity

$$U_{z,\text{gust}}(t) = \begin{cases} \frac{U_{\text{gust,max}}}{2}\left(1 - \cos\left(\frac{\pi}{H_{\text{gust}}} U_\infty t\right)\right), & \text{if } t_{z,1} \leq t \leq t_{z,\text{end}} \\ 0, & \text{otherwise,} \end{cases} \tag{18}$$

where $U_{\text{gust,max}}$ is the maximum gust velocity, H_{gust} the gust half length [7] and U_∞ the airspeed. With increasing time t the aircraft moves through the gust from nose to aft, as shown in Figure 11. At time

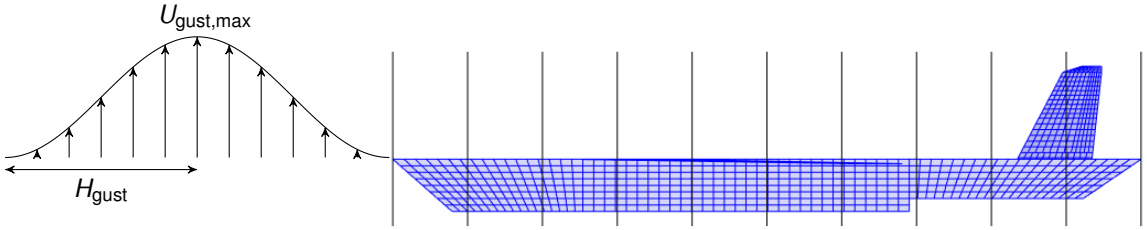

Figure 11: 1-cosine gust and aircraft gust zones.

$t_{z,1}$ the gust reaches the centre of a gust zone and leaves it at time $t_{z,\text{end}}$. The aerodynamic panels (small blue quadrangles) within the same gust zone experience the same gust velocity. This is an approximation which saves computation time. Ten gust zones were found to be accurate enough for

the demonstrator aircraft [12]. The difference in the gust zone velocity of two neighbouring gust zones is therefore describable by a time delay, which has a transfer function of the form

$$G_{z,\text{delay}}(s) = e^{-t_{z,\text{delay}}s}, \tag{19}$$

where $t_{z,\text{delay}}$ is the time delay in seconds and s is the Laplace variable [12]. To convert Equation (19) into a state-space format it is approximated by the second-order Padé approximation [9]

$$G_{z,\text{delay}}(s) \approx \frac{s^2 - \frac{6}{t_{z,\text{delay}}}s + \frac{12}{t_{z,\text{delay}}^2}}{s^2 + \frac{6}{t_{z,\text{delay}}}s + \frac{12}{t_{z,\text{delay}}^2}}. \tag{20}$$

5.3.2 Control Synthesis

The control surfaces chosen to react to the gust encounter are the most outer ailerons on both wings and all four ruddervators of the tail, as in Figure 12. The inputs to these control surfaces are allocated

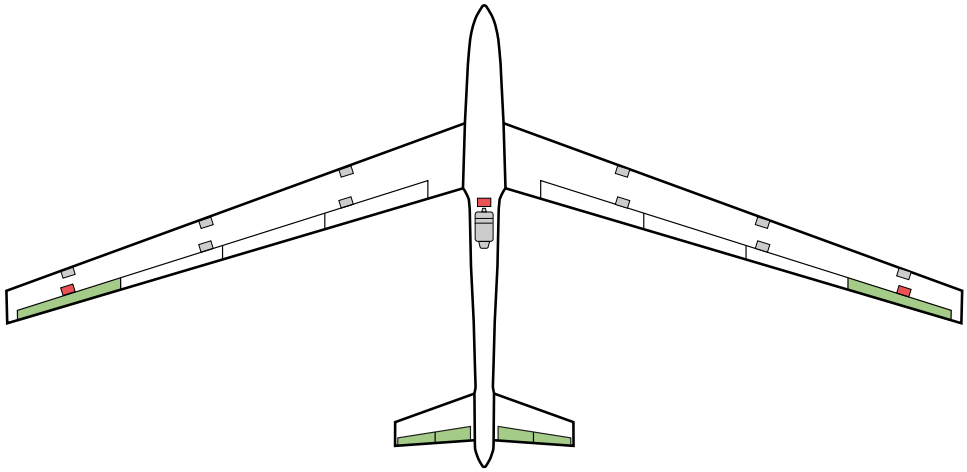

Figure 12: Demonstrator aircraft with IMUs (red) and control surfaces (green) for GLA control [25].

by utilising the longitudinal symmetry of the aircraft. This leads to two inputs u_{ail} and u_{rud}, one signal u_{ail} connected to both ailerons and signal u_{rud} connected to all four ruddervators.

The goal of the GLA controller is to reduce the wing root bending moment (WRBM) $M_{x,\text{wro}}$ due to a gust encounter. The measurements which are fed into the controller to fulfill this task are the pitch angle θ, the pitch rate q_{fu}, and the vertical accelerations $a_{z,\text{fu}}$ measured in the fuselage, $a_{z,\text{wtl}}$ and $a_{z,\text{wtr}}$ measured on the left and right wing, respectively. The controller then generates the commanded deflection signals u_{ail} and u_{rud}. The controllers are synthesised with the structured $\mathcal{H}_\infty$ method [3]. It generally solves an optimisation problem consisting of N_o equations of the form

$$\min_{K\in\mathcal{K}} ||W_{oi}G_{oi}(K)||_\infty,\ i = 1, ..., N_o, \tag{21}$$

for which the $\mathcal{H}_\infty$ norm of a weighted transfer function $W_{oi}G_{oi}(K)$ is minimised, while the state-space controller K is limited to the parameter space $\mathcal{K}$. The weighting function W_{oi} helps to limit the controller action to certain frequencies.
For the GLA controller K_{GLA} the parameter space $\mathcal{K}$ is reduced by the previously selected inputs and outputs and by limiting the number of states to 10. Furthermore, direct feedthrough is disabled for the controller in order to create a roll-off behaviour for the closed-loop dynamics. The overall control problem for the GLA control is outlined in form of a linear fractional transformation (LFT) in Figure 13. Three objectives ($N_o = 3$) are defined [31]:

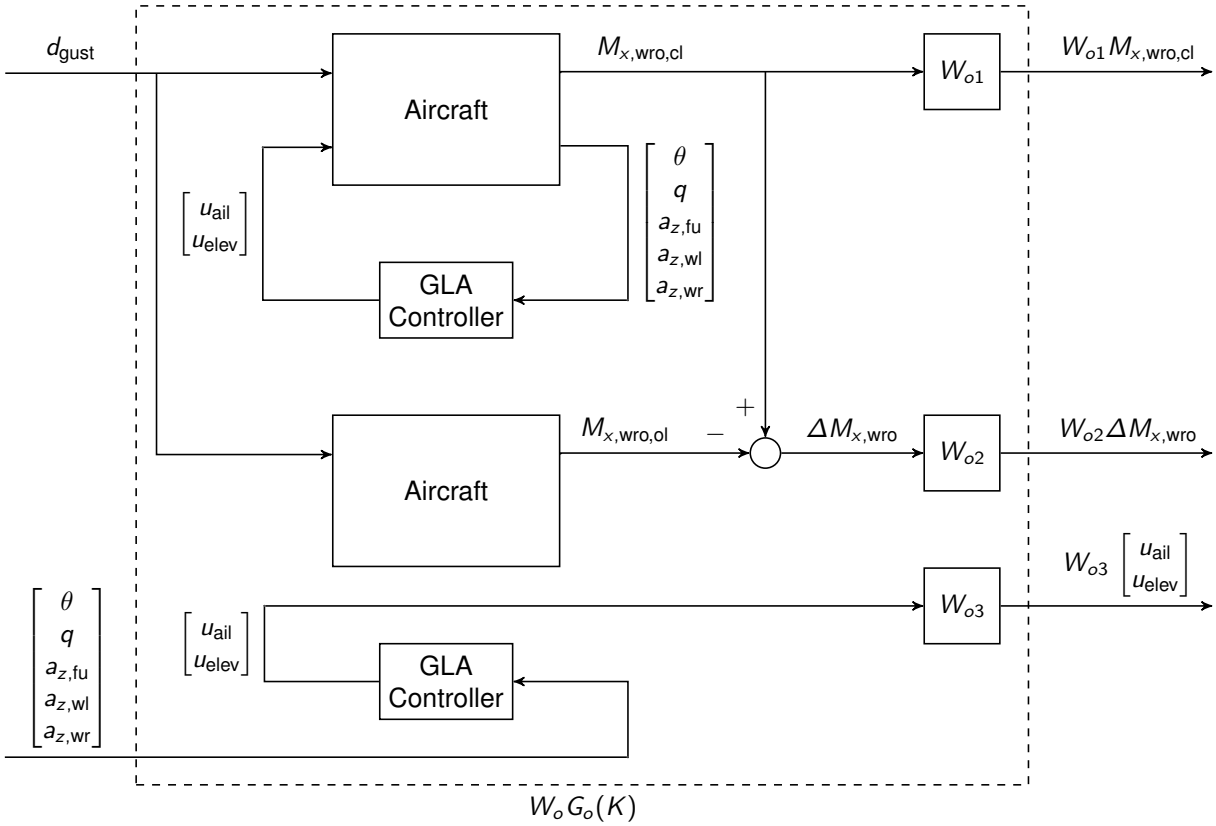

Figure 13: GLA control problem.

1. $\min_{K_{GLA} \in \mathcal{K}} ||W_{o1} G_{CL}(K_{GLA})||_\infty$
2. $\min_{K_{GLA} \in \mathcal{K}} ||W_{o2} (G_{CL}(K_{GLA}) - G_{OL})||_\infty$
3. $\min_{K_{GLA} \in \mathcal{K}} ||W_{o3} K_{GLA}||_\infty$

The first objective should minimise the closed-loop transfer function $G_{CL}(K_{GLA})$ from disturbance input d_{gust} to WRBM output $M_{x,wro}$. The second objective penalises the difference between $G_{CL}(K_{GLA})$ and the open-loop transfer function from disturbance input to WRBM output of G_{OL}. The weighting function W_{o2} puts the emphasis on low frequencies so that the GLA controller does not affect the flight dynamics. Eventually, the third objective represents a limitation of the controller K_{GLA} action to high frequency inputs. The weighting function W_{o3} rises with increasing frequency.

Based on these control objectives a GLA controller can be synthesised for the demonstrator aircraft with the set of -3 wings. This approach has successfully implemented for the demonstrator with the -0 wing.

5.4 Wing shape control with VarLoads

In the field of structural loads analysis, the focus is on accurate modelling of the lift forces, as they are the main driver of the structural sizing loads. Hence, in many aeroelastic implementations, forces acting in the longitudinal direction of the airframe are neglected. However, for flight mechanical assessments and cruise performance, the forces in the direction of the flow are essential. This issue can be addressed by either using higher fidelity methods or by extension of the potential flow based methods to account for longitudinal forces such as the induced drag.

A common practice is to pass the deformations of the structural model to the aerodynamic code and vice versa the pressure loading is passed to the structural code, which in turn is used to solve the flight dynamics equations of motion followed by a aircraft trimming procedure. This paper proposes an extension of the Vortex Lattice Method (VLM) implementation used in classical loads analysis. The resulting closed-form integral model allows for a fast execution without tedious iterations between the interfacing disciplinary codes.

The present implementation of the VLM accounts for the inherently nonlinear behavior of the induced drag and the dependence on the on-flow direction, while preserving the Aerodynamic Influence Coefficients and boundary conditions in matrix form, compatible with classical formulations.

The integral aircraft model is then used for wing shape control by optimizing the control surface scheduling of the trimmed, flexible aircraft to minimize the induced drag in off-design flight conditions. The optimizations of the control surface deflections regarding wing shape control are calculated with a model of the T-FLEX aircraft, the high aspect ratio subscale demonstrator for the FLIPASED project, which is equipped with multiple trailing edge control surfaces.

5.4.1 Overview

For many aircraft design analyses, mathematical models need to be evaluated numerous times, either to reflect changes during the design process or to evaluate certain criteria in the entire flight envelope. Therefore, loop capable models are essential which are fast to simulate, yet accurate enough for the specific analysis purpose.

A modelling approach for such applications was proposed in [15]. In [16] this modelling approach was used to optimize lift distributions for manoeuvre load and gust load alleviation. The in-plane forces for important flight dynamics and drag reduction were accounted for by the use of 3D panel methods [13, 18, 14]. In this paper, the integrated model has been improved to also support analyses regarding performance by accounting for induced drag forces with the Vortex Lattice Method. The current paper also features application examples pertaining to active manoeuvre load alleviation as well as performance improvement through optimization of the lift distributions for an high aspect ratio flexible aircraft configuration with multiple wing trailing edge flaps.

Optimal control surface scheduling to minimized drag of an elastic aircraft has been demonstrated in [22], which relies on a far field Trefftz plane analysis to evaluate the induced drag. An experimental test case was shown in [6], where a wind tunnel model of a flexible wing was used to validate the numerical model. This numerical model coupled a structural beam and a 3D potential flow panel method with an iterative scheme. In [36] a Vortex Lattice Method was used in combination with a 2D airfoil aerodynamic code to account for transonic and viscous effects. The aeroelastic and trim effects were regarded by iterations between structural deformation and aerodynamic lift distribution. A method for dynamic allocation for gust load alleviation is described in [32] with respect to optimal control surface scheduling for active load control.

The present aerodynamic model is based on a near field extension of the Vortex Lattice Method as commonly used for aerodynamic analyses. Instead of using an iterative exchange between the struc-

tural and aerodynamic model, the present method uses a matrix based formulation, which allows a closed form integral model formulation. This way, nonlinear effects like in-plane aerodynamic forces and induced drag components can be captured efficiently.

5.4.2 Quasi-Steady Aerodynamics: Vortex Lattice Method

As pointed out before, the large amount of design analysis cases that have to be considered in many aircraft design disciplines are still prohibitive for a wide spread adoption of costly Computational Fluid Dynamics (CFD) - Computational Structural Mechanics (CSM) coupled calculations. Therefore, usually classical methods derived from potential theory, such as the Vortex Lattice Method (VLM)[10] are still in use for many applications regarding flexible aircraft dynamics.

The Vortex Lattice Method (VLM) discretizes a lifting surface by trapezoidal shaped elementary wings, so called aerodynamic boxes. The aerodynamic lift is generated by placing a vortex along the quarter chord line of such an aerodynamic box. According to the Helmholtz theorems, such a vortex must either end at a solid surface, or extend to infinity. Hence, the bound vortex is extended at both corner points to infinity, forming the well known horseshoe shape with its legs pointing in free stream direction. The circulation strengths $\mathbf{\Gamma}_j$ of the individual horseshoe vortices are then determined by the Biot-Savart-Law and by meeting the flow compatibility condition, i.e. no perpendicular flow $\mathbf{v}_j$ through the solid surface at the control points at $3/4$ chord, according to Pistolesi's theorem [29].

$$\mathbf{v}_j = \mathbf{A}_{jj}\mathbf{\Gamma}_j \tag{22}$$

Figure 14 depicts the geometry of an aerodynamic box. The load acting point is located at mid span, quarter chord ($l-$set) and the control point at three quarter ($j-$set) chord point, respectively. The box reference point ($k-$set) is at the center of the box. The panel chord is c_j and the span is b_j. The vector of the bound vortex is denoted by $\mathbf{b}_l$.

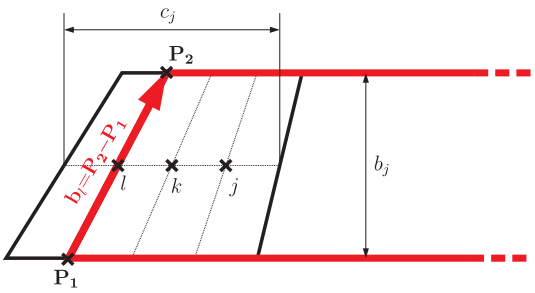

Figure 14: Geometrical properties of an aerodynamic box

The Kutta-Joukowski-Law relates the circulation to the lifting force.

$$L_j = \rho U_\infty \Gamma_j b_j \tag{23}$$

Normalizing the velocities at the control point $\mathbf{w}_j = \frac{\mathbf{v}_j}{U_\infty}$ and inversion of (22), yields the differential pressure coefficients with the aerodynamic AIC matrix.

$$\Delta\mathbf{c}_{\mathbf{p}_j} = \mathbf{Q}_{jj}\mathbf{w}_j \tag{24}$$

Reversely, the circulation of an individual horseshoe vortex can be calculated from the pressure coefficient as follows.

$$\Gamma_j = U_\infty \frac{c_j}{2} \Delta c_{p_j} \tag{25}$$

This is the ”classical” implementation of the VLM as employed for instance in the aeroelastic solutions of NASTRAN [33]. In the degree of freedom (DOF) notation for the j,k, and l set this implies :

$$\mathbf{u}_j = \begin{bmatrix} z \end{bmatrix}_j \; ; \; \mathbf{u}_k = \begin{bmatrix} z & \theta \end{bmatrix}_k^{\mathsf{T}} \; ; \; \mathbf{u}_l = \begin{bmatrix} z \end{bmatrix}_l \tag{26}$$

The boundary condition has to be met at $3/4$ chord location in z-direction ($j-\text{set}$). The resulting lift in the $l-\text{set}$ also acts in z-direction wrt the box geometry at the quarter chord, resulting in an additional moment about the y axis when regarding the box reference location $k-\text{set}$.

Alternatively, the Kutta-Joukowski-Law can be cast in the following form, involving a cross product instead of a scalar multiplication. This approach was proposed in [17] to enhance the flexible aircraft models for load analysis.

$$\mathbf{L}_l = \rho \mathbf{V}_l \times (\mathbf{b}_l \Gamma_j) \qquad \text{with} \;\; \mathbf{L}_l, \mathbf{V}_l, \mathbf{b}_l \in \mathbb{R}^3 \;\; \text{and} \;\; \Gamma_j \in \mathbb{R}^1 \tag{27}$$

The vector $\mathbf{b}_l$ is the vector quantity between the two corners of the horseshoe vortex, i.e. the bound vortex. The lift force $\mathbf{L}_l$ then acts, as aerodynamic theory predicts, perpendicular to the local stream velocity $\mathbf{V}_l$ at the quarter chord point. Consequently the DOF-set have to be extended:

$$\mathbf{u}_j = \begin{bmatrix} z \end{bmatrix}_j \; ; \; \mathbf{u}_k = \begin{bmatrix} x & y & z & \varphi & \theta & \psi \end{bmatrix}_k^{\mathsf{T}} \; ; \; \mathbf{u}_l = \begin{bmatrix} x & y & z \end{bmatrix}_l^{\mathsf{T}} \tag{28}$$

The lift at the quarter chord ($l-\text{set}$) is now a vector quantity in all three translational directions. The $k-\text{set}$ at the reference point was extended to all six DOF to account for moments arising from the directional lift.

Equation (27) can now be recast in matrix form by using eq. (25) and the skew matrix operator $\mathbf{sk}()$ for the cross product.

$$\mathbf{L}_l = q_\infty \left(\Big[-\mathbf{sk}(\mathbf{b}_l) \Big] \mathbf{w}_l \right) \odot \left(c_j \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^{\mathsf{T}} \Delta c_{p_j} \right) \tag{29}$$

The vector $\mathbf{w}_l$ is the velocity at quarter chord normalized by the free stream velocity. On individual box level, the quantities with index l are elements in $\mathbb{R}^3$, whereas index j denotes elements in $\mathbb{R}^1$. The operator $\odot$ in eq. (29) refers to an elementwise multiplication. A blockdiagonal extension of eq. (29) allows the use of the full DOF vectors in the $l-$ respectively $j-\text{set}$. Since both $\mathbf{w}_l$, as well as $\mathbf{w}_j$ (respectively Δc_{p_j}) depend on the boundary conditions, expression (29) is inherently nonlinear. Therefore, it is computationally more expensive compared to the linear expression in (23). If $\mathbf{w}_l = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^{\mathsf{T}}$, i.e. the free stream is assumed to be exclusively in x-direction, the analysis simplifies to the conventional case, with the lift $\mathbf{L}_l$ acting in the z-direction only.

In a last step, the lift vector $\mathbf{L}_l$ at the quarter chord is transformed to the reference point at the aerodynamic box centeroid.

$$\mathbf{P}_k^{\text{aero}} = \mathbf{T}_{lk}^{\mathsf{T}} \, \mathbf{L}_l \tag{30}$$

As mentioned before the $k-\text{set}$ has been extended to full 6 DOF to recover moments arising from the now directional lift vector.

5.4.3 Boundary Conditions: Differentiation Matrices

The boundary conditions for the Vortex Lattice Method are obtained with respect to a box reference point. Depending on the displacement, respectively motion of the reference point, the normal velocity $\mathbf{w}_j$ at the control point and velocity at the quarter chord point $\mathbf{w}_l$ need to be determined. The rotational displacement about the transverse axis contributes to the box angle of attack. The heaving motion, as well as rotational motion due to the offset of the reference point to the control point, also contribute to the effective box angle of attack.

These contributions can be accounted for by a differentiation wrt the x-direction, i.e. the change in slope θ_k, respectively wrt time for the normalized heaving motion $\dot{z}_k/U_\infty$ and pitching motion $\dot{\theta}_k/U_\infty$. These deformations and motions are represented in the vectors $\mathbf{u}_k$ and $\dot{\mathbf{u}}_k$. Thus, in matrix form this can be expressed as

$$\mathbf{w}_j = \mathbf{D^x}_{jk} \cdot \mathbf{u}_k \; + \; \mathbf{D^t}_{jk} \left(\frac{c_{ref}/2}{U_\infty} \right) \cdot \dot{\mathbf{u}}_k \tag{31}$$

The differentiation matrices are defined as

$$\mathbf{D^x}_{jk} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{32}$$

and

$$\mathbf{D^t}_{jk} = -\frac{2}{c_{ref}} \begin{bmatrix} n_x & 0 & n_z & 0 & -c_j/4 & 0 \end{bmatrix}. \tag{33}$$

In order to account for camber and incidence, the quantities n_x and n_z are the components of the normal vector at the control point. The classical implementation omits the component in x-direction, i.e. $n_x = 0$ and $n_z = 1$ and merely treats camber as addition to the differential pressure. In the present implementation an in-plane motion results in an increased circulation.

The differentiation matrix $\mathbf{D^t}_{lk}$ is set up equivalently for the three translational DOFs at the quarter chord.

$$\mathbf{w}_l = \mathbf{D^t}_{lk} \left(\frac{c_{ref}/2}{U_\infty} \right) \cdot \dot{\mathbf{u}}_k \tag{34}$$

with

$$\mathbf{D^t}_{lk} = -\frac{2}{c_{ref}} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -c_j/4 \\ 0 & 0 & 1 & 0 & c_j/4 & 0 \end{bmatrix} \tag{35}$$

Figure 15 illustrates the effects computed by equation (31) and (34).

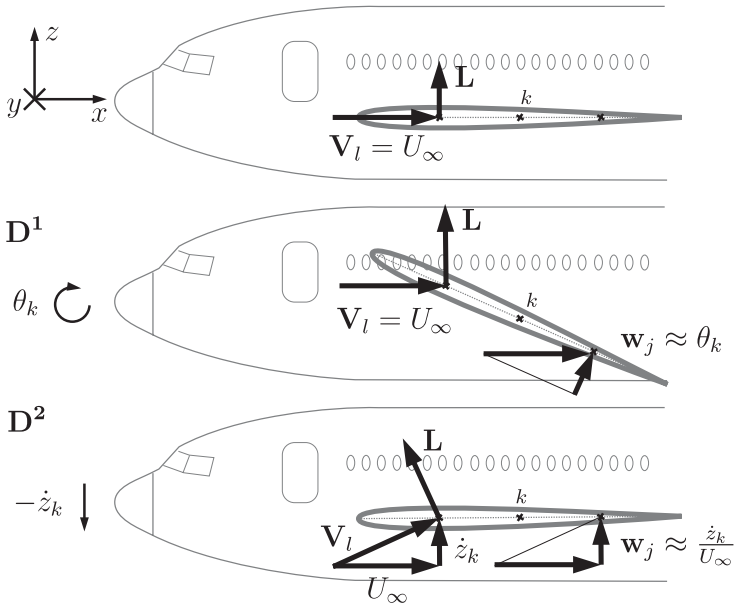

Figure 15: Illustration of the differentiation matrices $\mathbf{D^x}_{jk}$, $\mathbf{D^t}_{jk}$ and $\mathbf{D^t}_{lk}$

There is no matrix for $\mathbf{D^x}_{lk}$, as a steady deformation does not change the direction of flow. However, the heaving motion changes the direction of the lift vector. In the classical VLM implementation this contribution is neglected.

5.4.4 Induced Drag

Aerodynamic lift is generated by circulation Γ over the wing. Since the circulation distribution of a finite wing has to go to zero at the wing tips, a spanwise gradient in circulation results. Induced drag is generated by this spanwise change of the circulation of a 3D lifting surface, which leads to trailing vortices that are shed into the wake. The streamwise vorticity in the wake induces an additional downwash (36) on the wing, which turns the lift vector rearwards by the induced angle of attack (37). Figure 16 shows the the effect on an airfoil section of a finite wing.

Induced downwash is obtained by the integral equation:

$$w_i(y) = -\frac{1}{4\pi}\int_{-b/2}^{b/2} \frac{\mathrm{d}\Gamma(y_0)}{\mathrm{d}y}\frac{\mathrm{d}y_0}{y - y_0} \tag{36}$$

Using a small angle assumption, the induced angle of attack is then given by

$$\alpha_i(y) \approx \frac{-w_i(y)}{U_\infty}. \tag{37}$$

The overall induced drag can then be obtained by integrating over the wing span

$$D_i = \frac{1}{2}\rho U_\infty \int_{-b/2}^{b/2} \Gamma(y)\alpha_i(y)\,\mathrm{d}y. \tag{38}$$

Similarly, the total lift is obtained by

$$L = \rho U_\infty \int_{-b/2}^{b/2} \Gamma(y)\,\mathrm{d}y. \tag{39}$$

When comparing the integral equations, it can be seen that the lift (39) depends linearly on the overall circulation Γ, only. Whereas, in the drag integral (38), the induced angle of attack α_i is also a function of the circulation Γ, leading to the classical parabola shape of the drag polar.

To obtain the drag forces for the Vortex Lattice Method, there are two options:

- far field implementation:
  Solving the integral equations (38) and (39) in the so called Trefftz plane, which is located far downstream and perpendicular to the wake. The integration is carried out along the wake surface in this plane.
- near field implementation:
  Determining the induced downwash at the quarter-chord point of each box turning the local lift vectors as depicted in figure 16.

The near field implementation is very attractive, as the flexible aircraft equations of motion require a distributed aerodynamic loading. One drawback of it is that it is somewhat sensitive to the discretization of the lifting surface.

5.4.5 Near Field Induced Drag

The vorticity of the trailing vortices in the wake induces an additional downwash at the quarter chord of each box, which rotate the lift vector rearwards. The sum of the rearward component of all boxes constitute the induced drag. To evaluate the induced velocities at the quarter chord point, once again the Biot-Savart Law is employed. The contribution of the bound vortex of the current box itself needs to be omitted, due to the singularity at zero distance. The three velocity components normalized by

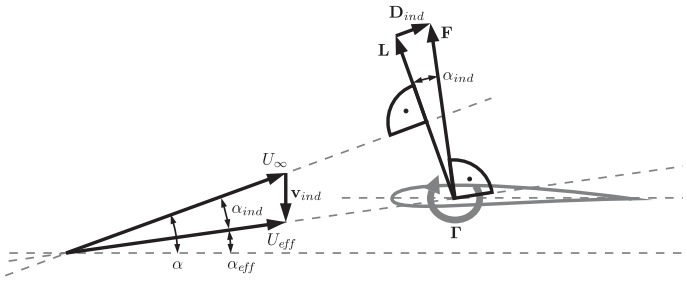

Figure 16: Illustration of induced drag and induced angle of attack

the free stream velocity are collected in the vector $\mathbf{w}_{l_{\text{ind}}}$. Similarly to eq. (22), this operation can be expressed as matrix equation.

$$\mathbf{w}_{l_{\text{ind}}} = \mathbf{A}_{lj} \frac{\Gamma_j}{U_\infty} \tag{40}$$

To build up the effective stream direction $\mathbf{w}_l$, the induced component $\mathbf{w}_{l_{\text{ind}}}$ is added to the free stream and motion induced components of equation (34). The subsequent force calculation in eq. (29) then includes the induced drag components.

5.4.6 Far Field Induced Drag

The evaluation of the far field is done by evaluation of eq. (38) in its discretized form. Once again the Biot-Savart Law is employed, but instead of calculating the downwash at the quarter chord, the induced velocity is evaluated on the wake surface far downstream in the so called Trefftz plane. The contributions of the individual box horseshoe vortices for each strip need to be summed up by multiplication of matrix $\mathbf{S}_{sj}$. The normalized velocity component perpendicular to the wake surface $\mathbf{w}_{s_{\text{ind}}}$ is then evaluated at the mid distance between these stripwise vortex pairs. The overall circulation of the wake strip is then multiplied with the perpendicular induced velocity and the wake strip width b_s. All wake strips are then summed up to the overall induced drag coefficient.

$$C_{D\,\text{FF}} = \frac{1}{S_{\text{ref}}} \sum_{n_s} b_s \cdot \mathbf{w}_{s_{\text{ind}}} \cdot \mathbf{S}_{sj} \frac{\Gamma_j}{U_\infty} \tag{41}$$

Equivalently, the overall lift is determined by a far field evaluation of eq. (39)

$$C_{L\,\text{FF}} = \frac{2}{S_{\text{ref}}} \sum_{n_s} b_s \cdot n_{z_s} \cdot \mathbf{S}_{sj} \frac{\Gamma_j}{U_\infty} \tag{42}$$

Similarly, the side force coefficient can be calculated in the far field

$$C_{Y\,\text{FF}} = \frac{2}{S_{\text{ref}}} \sum_{n_s} b_s \cdot n_{y_s} \cdot \mathbf{S}_{sj} \frac{\Gamma_j}{U_\infty} \tag{43}$$

The force components of each wake strip in the wind axis system are determined by n_{z_s} for the lift coefficient, respectively n_{y_s} for the side force coefficient.

5.4.7 Results: Optimal Control Surface Scheduling

A flexible aircraft model, which lends itself to demonstrate wing shaping control is the FLEXOP demonstrator aircraft T-FLEX [38]. The aircraft features a high aspect ratio and multiple trailing edge control surfaces distributed over the wing span. The original aircraft is equipped with four ailerons per wing. To allow for a finer granularity, each aileron has been split fourways for the simulation model used in

this study, resulting in overall 16 ailerons per wing. The VLM model of the wing is depicted in figure 17. Not shown is a V-tail equipped with two elevators per empennage surface, which allow for pitch and yaw control. The fuselage is not modelled aerodynamically and no empirical corrections for additional drag components have been employed in the simulation model. The propulsion system is based on a single jet engine mounted on top of the fuselage. The external propulsion forces are introduced into the structure accordingly, including the pitching moment contribution.

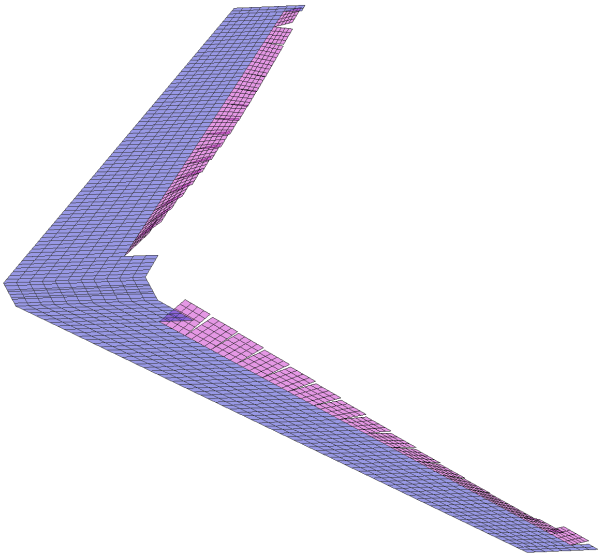

Figure 17: Flexible aircraft model with high aspect ratio wing and multiple tailing edge control surfaces

This use case demonstrates the capabilities of the proposed modelling approach with regards to performance improvement through lift re-distribution by optimal control surface deflection.

The application here is based on a flexible aircraft model with nonlinear equations of motion based on mean axes constraints. The model is trimmed by a nonlinear solver for horizontal flight condition. Optionally a linearized state space model can be obtained to assess the root loci.

5.4.8 Performance Improvement through Wing Shape Control

An optimal (preferably elliptic) lift distribution ensures low induced drag. During the mission of the aircraft, the fuel stored in the wings is consumed and therefore the mass properties change. Hence, the lift distribution changes due to aeroelastic effects. Usually, aircraft are designed for typical intermediate mass cases. However, away from this design point, either very heavy at the beginning, or very light at the end of the mission, the lift distribution is not optimal anymore. With adequate control surfaces available, the lift distribution can be modified to reduce the detrimental effect on the performance.

The T-FLEX aircraft does not have fuel tanks in the wing and due to the comparatively small fuel tank, the weight of the aircraft is only changing minimally during a test flight. Therefore, the aircraft speed is varied over a wide range to obtain off-design lift coefficients.

The objective is to minimize the required thrust in a trimmed horizontal flight condition for a given speed, by symmetrically deflecting the 16 trailing edge control surfaces per wing. To demonstrate the effectiveness of the Wing Shape Control (WSC), the results are compared to the induced drag values of the reference case without control surface deflections. It should be noted that the drag values also include the trim drag component of the overall flexible aircraft. Hence, the optimal wing lift-distribution

might differ from an elliptical shape, since the pitching moment has to be balanced by tail surface deflections. In that regard the obtained values correspond to an overall optimum of the entire flexible airframe in a trimmed flight condition.

One evaluation of a trimmed flight condition takes about 1.1 s. For the 16 symmetrically deflected control surfaces as optimization parameters, around 800 iterations are needed to a reach a flight state with minimal required thrust. A paralellized run for the 17 flight points between 35 and 75 m/s took approximately 20 minutes. The resulting control surface deflections of the 16 ailerons for the wing shape control over the wing span are depicted in figure 18.

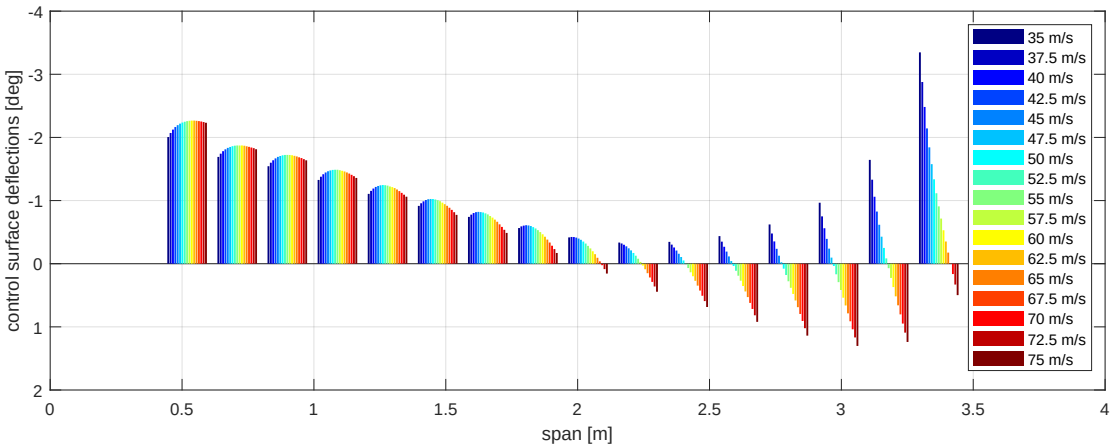

Figure 18: Optimal control surface deflections for different velocities

The associated drag polar is drawn in figure 19. The polar for the reference case is drawn in orange, the drag reduced polar with wing shape control in blue. The solid line are determined using the distributed near field implementation, which is used for trimming the flexible aircraft. The dashed lines are the integral far field evaluations in Trefftz plane of the total drag coefficient. While a difference between near and far field evaluation exists, the relative drag reduction of the wing shape control is nearly the same.

The lift and drag coefficients along with the relative drag improvements are summarized in table 2. The drag coefficients are represented in drag counts, i.e. $C_D \times 10^4$ for better readability.

V_{TAS} [m/s]	c_L	$c_{Di} \times 10^4$ ref.	$c_{Di} \times 10^4$ WSC	drag reduction ratio	$c_{Di} \times 10^4$ ref. (Far Field)	$c_{Di} \times 10^4$ WSC (Far Field)
35.0	0.37043	24.405	23.651	1.0319	25.59	24.87
37.5	0.32271	19.402	18.745	1.0351	20.61	19.959
40.0	0.28364	15.876	15.268	1.0398	17.101	16.48
42.5	0.25126	13.34	12.754	1.046	14.579	13.964
45.0	0.22412	11.486	10.902	1.0536	12.736	12.113
47.5	0.20115	10.109	9.516	1.0624	11.368	10.728
50.0	0.18154	9.0741	8.4635	1.0721	10.341	9.6771
52.5	0.16467	8.2866	7.6536	1.0827	9.5607	8.8693
55.0	0.15004	7.6816	7.0229	1.0938	8.9623	8.2411
57.5	0.13728	7.2129	6.526	1.1053	8.4997	7.747
60.0	0.12608	6.8472	6.1305	1.1169	8.1399	7.3545
62.5	0.1162	6.5602	5.8124	1.1286	7.8585	7.0397
65.0	0.10743	6.3338	5.5541	1.1404	7.6379	6.7849
67.5	0.099623	6.1547	5.3424	1.1521	7.4645	6.5768
70.0	0.092635	6.0128	5.1671	1.1637	7.3284	6.4053
72.5	0.086358	5.9003	5.0208	1.1752	7.2219	6.2628
75.0	0.080698	5.8113	4.8974	1.1866	7.1391	6.1433

Table 2: Lift and induced drag coefficients with and without Wing Shape Control. (Note: drag is given in drag counts $C_D \times 10^4$)

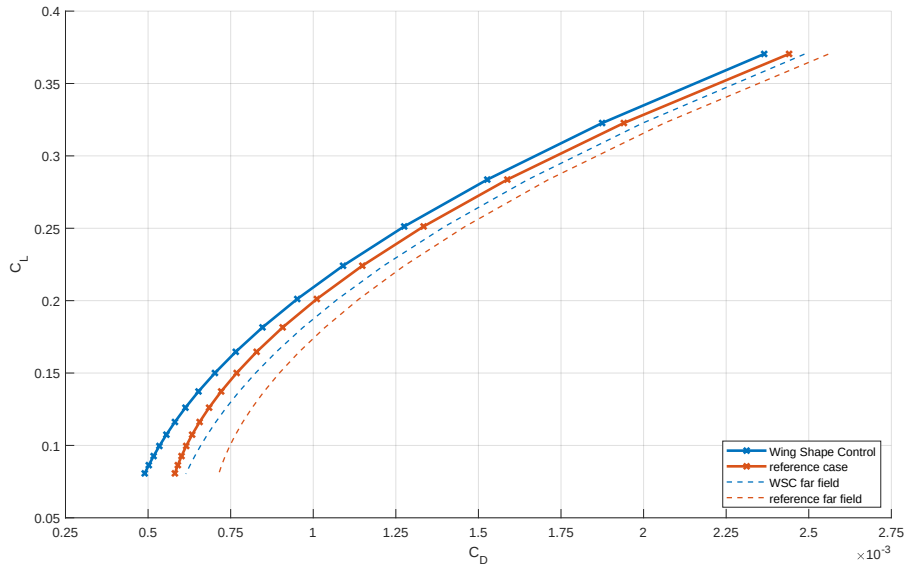

Figure 19: Drag polar with and without Wing Shape Control. (Far Field integral evaluations as dashed lines)

5.5 Wing shape control with PANUKL

For induced drag minimization via wing shape control, a surrogate drag model is needed first. This drag model can be constructed via the Panukl software for example. In the next part, the summary of the surrogate drag model development and the main ideas behind the open loop wing shape control are presented.

5.5.1 Surrogate drag model construction via the PANUKL software

In the following, the results of the simulations performed with the -0 wing are described, including the method we used as well as the challenges occurred during the process. To determine the optimal wing control surface deflections, PANUKL simulations were run with all possible combinations of the following aileron deflections (in degrees):

- -1, 0, 1;
- -2, -3, -4;
- -5.5, -5, -4.5;
- 2, 3, 4;
- -0.5, 0.5;
- -1.5, 1.5;
- 0, -0.1, -0.2, -0.3;

5.5.2 Aeroelastic trim via Panukl

For the simulations, an aeroelastic trim algorithm is implemented. Panukl software is used to model aerodynamics as detailed previously. This representation computes the aerodynamic load acting on the rigid panel model. The aerodynamic loads are given in Figure 20.

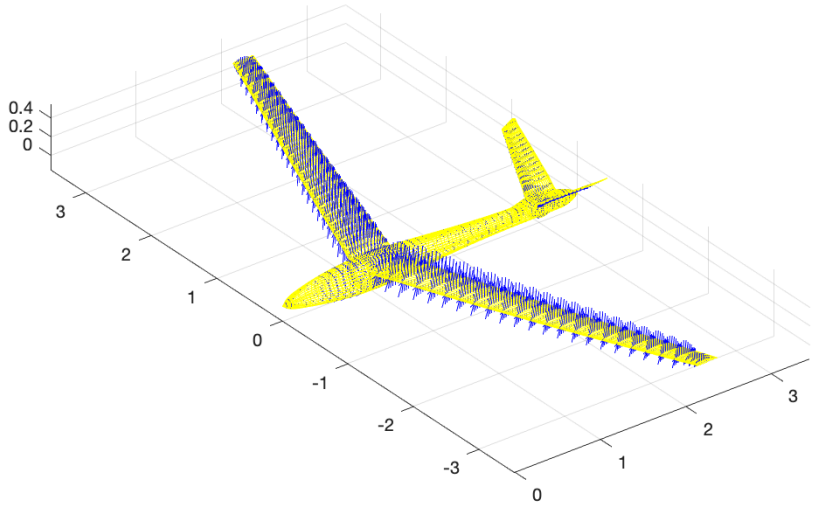

Figure 20: Aerodynamic load calculated from pressure coefficients

Pressure coefficients ($C_{p,i}$) are taken from the results to calculate the aerodynamic force (F_i) and moment (M_i) acting on each panel, as in equation (44) and (45):

$$F_{aero,i} = \frac{1}{2}\rho V^2 S_{panel,i} C_{p,i} \tag{44}$$

$$\begin{bmatrix} F_i \\ M_i \end{bmatrix} = \begin{bmatrix} 1 \\ c_{p,i}/4 \end{bmatrix} \tag{45}$$

where ρ represents air density, V is the airspeed, $S_{panel,i}$ and $c_{p,i}$ are the panel surface and the panel chord corresponding to the i^{th} panel respectively. These components are applied on the structural grid in the form of modal forces using the surface spline method and the transpose of the mode shape matrix. To convert this load into components that are applicable on the aircraft structure, the spline model has to be built. The spline nodes along the center of every component of the aircraft are geometrically identical to the structural grid points and the outer nodes were created in alignment with these central nodes based on the aerodynamic model. The presented spline model is able to account for forces in the z direction and moments acting in the $x-y$ plane. To create a system where the 'x' component of the aerodynamic force is taken into consideration, an additional vertical spline grid can be added to the spline model, which includes the moments acting in the $x-z$ plane as well. The complete spline representation is shown in Figure 21. This additional element promotes a more precise solution for the trim deformation calculation.

In the next step, the aerodynamic loads on the aerodynamic panels are converted to modal coordinates via the splines described above. These modal forces deform the structural grid, hence deforming the

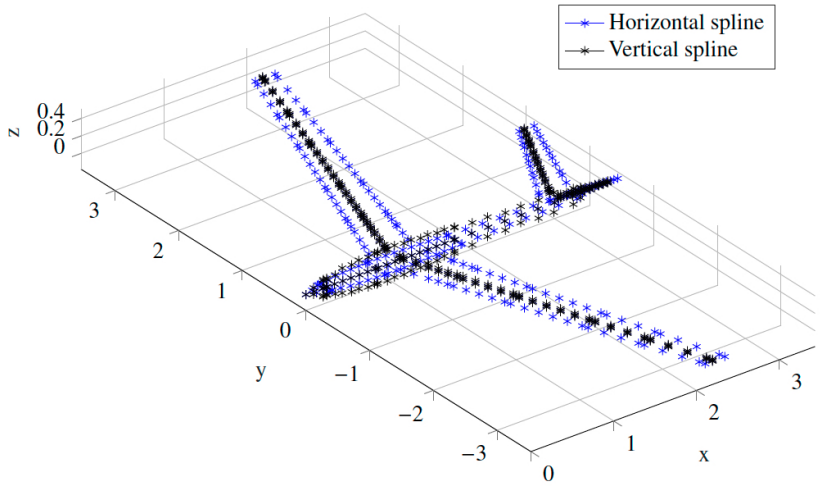

Figure 21: Horizontal and vertical spline

aerodynamic panels as well. The deformed aerodynamic geometry in Figure 22. will become the input for the next iteration step, where Panukl software takes the leading- and trailing edge coordinates to generate the a new panel model, for which a new set of aerodynamic load will be computed. The iteration (3-4 steps) is continued until an aeroelastic trim is reached.

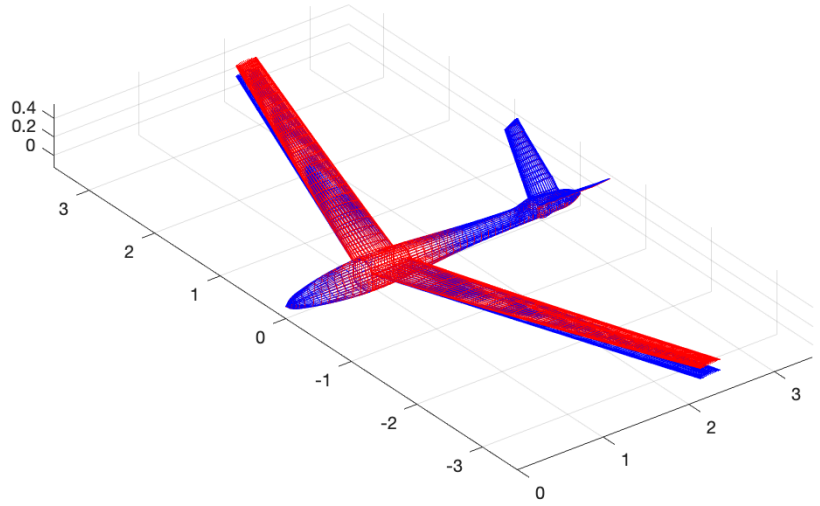

Figure 22: Aerodynamic grid and its deformation

The results of the Panukl computation are saved for this aeroelastic trim condition and these data are used to construct a surrogate aerodynamics model. The surrogate model is presented in the following section.

5.5.3 Surrogate aerodynamic model

The possible aileron configurations, as described in the previous section, result in a total of 612 Panukl simulation points for each airspeed values. The same aileron configurations are simulated for airspeed values $V = \begin{bmatrix}30 & 33 & 35 & 40 & 45 & 50 & 55 & 60 & 61\end{bmatrix} m/s$. The surrogate aerodynamics model that is fit on these results at each airspeed value is given as:

$$C_{Di} = f(\alpha, \delta_e, \delta_a) = C_{D0} + C_{D,\alpha}\alpha + C_{D,\delta_e}\delta_e + C_{D,\delta_a}\delta_a + C_{D,\alpha^2}\alpha^2 + C_{D,\delta_e^2}\delta_e^2 + C_{D,\delta_a^2}\delta_a^2 + C_{D,\alpha\delta_e}\alpha\delta_e + C_{D,\alpha\delta_a}\alpha\delta_a \tag{46}$$

where α is the angle of attack, δ_e is the elevator deflection and $\delta_a = [\delta_{a1}, \delta_{a2}, \delta_{a3}, \delta_{a4}]$ are the aileron deflections. A function was fit to the resulting 612 data points using pseudoinverse matrix, and then the aileron configuration that leads to the minimal induced drag values was found using Matlab's fminsearch function. In case the model was fitted to all 612 datapoints, it was found that the function fits the entire data set well (with an error of around 3%), but it is too "flat" around the minimum, so the aileron configuration of the minimum induced drag cannot be determined accurately. To overcome this problem, the simulation data was sorted according to C_{xi} and only the k smallest C_{xi} simulation points were used for the fitting. The number of data used (k) greatly influenced the optimal aileron deflections, therefore the value of k had to be chosen manually for each airspeed simulation. Such approach leads to optimal α, δ_e and δ_a values for each speed. The results can be seen below in Table 3.

Table 3: Induced drag values and optimal aileron configurations in degrees

V [m/s]	$C_{xi_{sim}}$	$C_{xi_{model}}$	δ_{a1}	δ_{a2}	δ_{a3}	δ_{a4}	α	δ_e
30	0,003950	0,003940	0,4748	0,6876	0,7639	0,1576	1,2886	-4,5776
33	0,002682	0,002678	0,5450	0,6414	0,8020	0,4491	0,2395	-3,4409
35	0,002110	0,002110	0,4049	0,6256	0,7668	0,2845	-0,14497	-2,6424
40	0,001220	0,001230	0,0877	0,4020	0,4853	0,1478	-0,82807	-1,1746
45	0,000766	0,000768	-0,4185	0,0451	0,0860	-0,2363	-1,1656	-0,1004
50	0,000519	0,000512	-0,8458	-0,2593	-0,2374	-0,8268	-1,366	0,818559
55	0,000371	0,000359	-1,1859	-0,4449	-0,5013	-0,9527	-1,5332	1,2973
60	0,000280	0,000271	-1,1790	-0,5360	-0,5693	-1,0175	-1,7398	1,78186
61	0,000266	0,000258	-1,1397	-0,5644	-0,5489	-0,9617	-1,7856	1,83084

The resulting model was then validated by comparing the C_{xi} values from the Panukl simulations and the surrogate model with non-deflected aileron configuration. The results are summarized in Table 4.

Table 4: Surrogate drag model evaluation

V [m/s]	30	35	40	45	50	55	60
$C_{xi_{sim}}$	3,935E-03	2,114E-03	1,236E-03	7,740E-04	5,310E-04	3,950E-04	3,070E-04
$C_{xi_{model}}$	3,940E-03	2,110E-03	1,230E-03	7,680E-04	5,120E-04	3,590E-04	2,710E-04
error [%]	-0,13	0,19	0,49	0,78	3,58	9,11	11,73

The trim angle of attach and elevator values, with the optimal aileron configurations and the resulting C_{xi} values as a function of airspeed are given in Figures 23–25.

In order to be able to properly interpolate between the above airspeed points, each variable is approximated by a function:

- A linear approximation was used for the ailerons between v=33...55 m/s;
- A quadratic function was used to estimate δ_e between v=30...61 m/s;

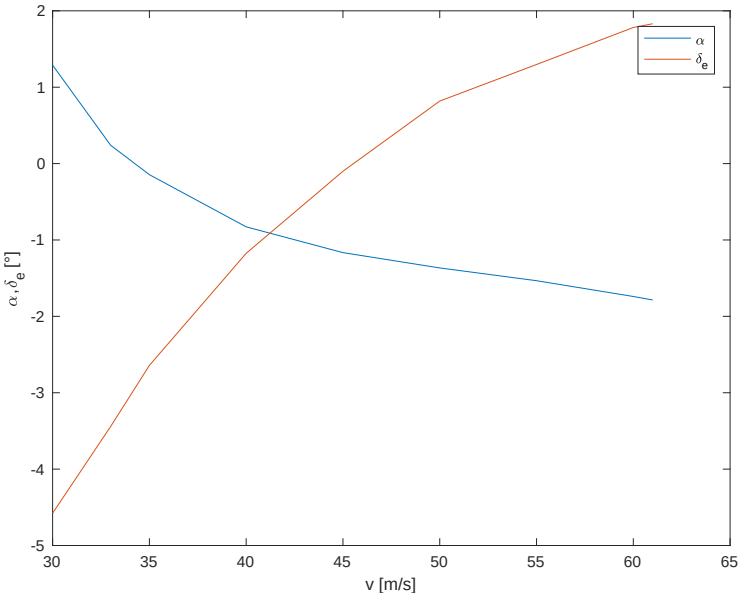

Figure 23: Trim angle of attack and elevator values

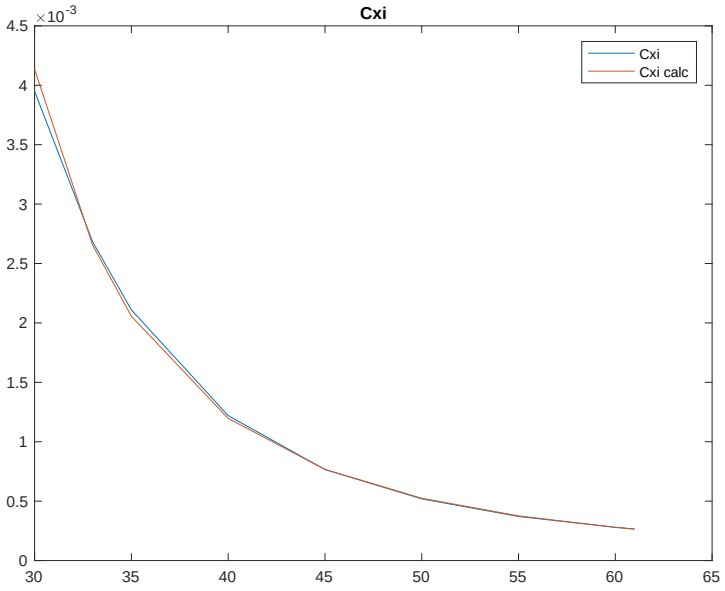

Figure 24: C_{xi} values

- A quadratic function was used to estimate α between v=30...61 m/s;
- Approximation with a quadratic (or higher) polynomial was not suitable for C_{xi}, but a fit to $1/\sqrt{(C_{xi})}$ resulted in a reasonably accurate approximate.

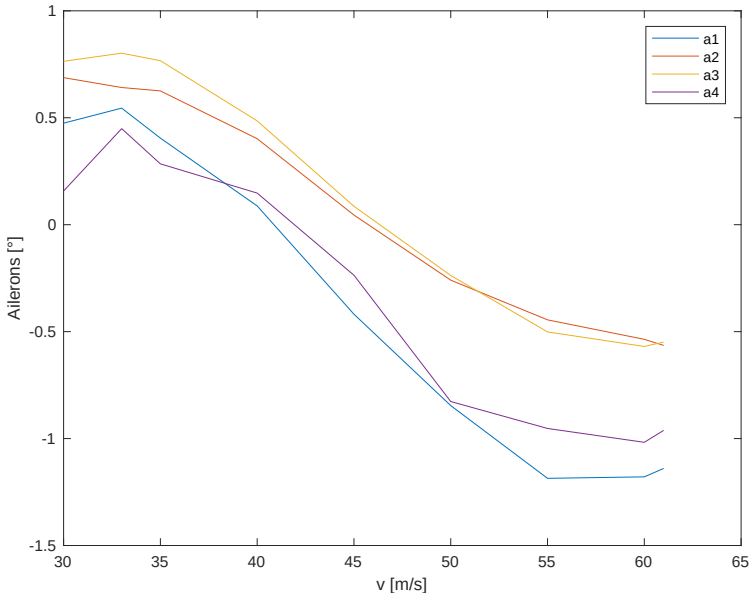

Figure 25: Optimal aileron configurations (in degrees)

5.5.4 Open loop wing shape control

The surrogate drag model presented above can be used for wings shape control based drag reduction. In the present case, a simple open loop control is considered. A look-up table is constructed based on the surrogate drag model which returns the optimal aileron configuration depending on the airspeed of the aircraft. The ailerons are then set to these values. At trim condition, the wing deflection is determined by these aileron settings and a wing shape resulting in a minimal induced drag is obtained. A special care needs to be taken to ensure that the optimal aileron settings not affect the baseline and flutter suppression controllers significantly.

In the future, a feedback based, dynamic wing shape controller design is planned. For this controller a more complex drag surrogate model is required, which provides information about the drag properties at off-trim conditions as well.

5.6 Wing shape control - Trefftz plane implementation in NASTRAN

A far-field Trefftz plane implementation similar to the one presented in [21] is implemented. The induced velocity and drag are calculated on a plane downstream of the aircraft known as the Trefftz plane, using the span-wise lift distribution projected onto the Trefftz plane as shown in Figure 27.

Panel lift forces are first obtained from an aerodynamic method, the doublet lattice method (DLM) - based aeroelastic solver in MSC.NASTRAN in this case. From the calculated strip-wise lift F_i, the strip circulation in the i^{th} strip is obtained through the Kutta–Joukowski equation as

$$\Gamma_i = \frac{F_i}{\rho V_0} \tag{47}$$

The total downwash on a strip i due to the circulation at all span-wise strips of the lifting surface is given

by

$$w_i = \left(\frac{1}{4\pi}\right)\sum_{k=1}^{N}(\Gamma_k - \Gamma_{k+1})\left[\frac{1}{y_i - y_k} - \frac{1}{y_i + y_k}\right] \qquad i,k = 1,....N \tag{48}$$

The coordinate y_i is calculated at the centre of the strip (where the lift and circulation are calculated) and y_k at the outer definition of the strip (where the trailing line vortex is modelled) as shown in Figure 27.

The induced angle of attack in strip i is calculated knowing the strip downwash as

$$\alpha_{i_{induced}} = -\frac{1}{V_0} w_i \tag{49}$$

The induced drag in the strip i is finally calculated as

$$D_i = [F_i]\alpha_{i_{induced}} \tag{50}$$

where F_i is the lift force in the i^{th} strip and V_0 is the free-stream velocity. The definitions of the various coordinates are shown in Figure 27. The total induced drag is obtained by summing the contributions from the N strips on the lifting surface.

$$D = \sum_{i=1}^{N} D_i \tag{51}$$

It is to be emphasized that the D and D_i in the above equations refer only to the induced drag, that is, the drag due to lift.

Equation 48 can be used for a single lifting surface aligned along the horizontal plane as shown in Figure 27. In this case, the wake of this single surface is projected onto the downstream Trefftz plane. When two or more lifting surfaces are present, the influence of the wake of each lifting surface must be calculated on every other lifting surface's wake. Additionally, in the presence of a dihedral, the distance terms y_i and y_k in Equation 48 are replaced by the vector distance between the various interacting lifting surfaces.

The entire routine is programmed within the SOL200 solution in MSC.NASTRAN, making use of appropriate cards to extract lift responses and to define the equations to compute the induced drag, that is Equations 47 - 51. The routine is coupled to an external Python script to perform the drag optimization. Several random distributions of the control surfaces are generated first, for each which the induced drag is calculated. The data points are used to construct a Kriging-Regression model, on which the minimization problem is solved.

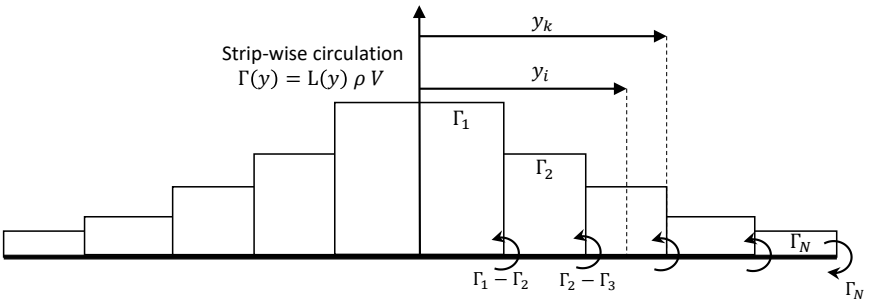

Figure 26: Definition of coordinates for the Trefftz plane implementation

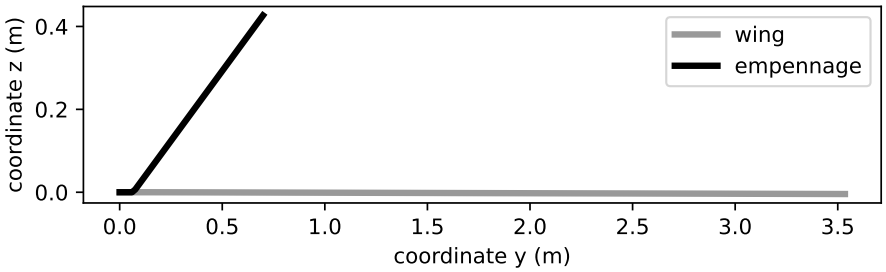

Figure 27: Wake at Trefftz plane from wing and empennage

6 Conclusion

A strategic decision has been reached to retrofit an existing wing, thereby separating the wing design process of the RCE workflow from the aspects of wing manufacturing and flight testing. This significant step aims to enhance efficiency and streamline the overall workflow. The retrofit plan, which outlines the specific actions to be taken, is presented in detail within this deliverable. By implementing this retrofit, the goal is to achieve greater flexibility in wing design iterations without disrupting the manufacturing and flight testing processes.

The rationale behind the selection of the retrofit wing is extensively documented in this report. It delves into the various factors considered during the decision-making process, including performance metrics and feasibility analysis. The argumentation provides a comprehensive overview of why the chosen retrofit wing is deemed optimal for achieving the desired outcomes. By presenting a well-supported case, the document aims to provide a clear understanding of the reasoning behind the selected wing and build confidence in the retrofit decision among stakeholders.

Moreover, this document offers a comprehensive elucidation of the Manuelor Load Alleviation (MLA), Gust Load Alleviation (GLA), and wing shape control design. These crucial design aspects are instrumental in ensuring that the retrofit wing operates in accordance with the desired objectives. The MLA system allows for precise control over the deflection of control surfaces, effectively reducing the load exerted on the wing. On the other hand, the GLA system aims to minimize the impact of gust loads on the wing, thereby enhancing safety and durability. Additionally, the wing shape control design expounds on the various mechanisms and technologies employed to achieve the desired changes in wing shape during flight. This aspect significantly contributes to improved aerodynamic performance and overall efficiency.

By expanding upon these topics within the deliverable, the aim is to provide comprehensive insights into the retrofit plan, the rationale behind the wing selection, and the intricate design considerations that will contribute to the successful implementation of the project.

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] P. Apkarian and D. Noll. Nonsmooth h ∞ synthesis. *IEEE Transactions on Automatic Control*, 51(1):71–86, 2006.

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

[5] Mark Drela. Avl.

[6] David Eller and Sebastian Heinze. Approach to induced drag reduction with experimental evaluation. *Journal of Aircraft*, 42(6):1478–1485, nov 2005.

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

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

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

[10] S. Hedman. Vortex Lattice Method for Calculation of Quasi Steady State Loadings on Thin Elastic Wings. Technical Report Report 105, Aeronautical Research Institute of Sweden, October 1965.

[11] Andreas Hermanutz. Flexop D1.4 report on design of -1 wings. Technical report, TUM, 2017.

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

[13] Thiemo M. Kier. Integrated Flexible Dynamic Maneuver Loads Models based on Aerodynamic Influence Coefficients of a 3D Panel Method. In *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 5-9 January 2015, Kissimmee, FL, USA*, number AIAA 2015-0185. AIAA, 2015.

[14] Thiemo M. Kier. An Integrated Model for Lateral Gust Loads Analysis and Dutch Roll Flight Dynamics using a 3D Panel Method. In *International Forum on Aeroelasticity and Structural Dynamics*, number IFASD-2017-107, 2017.

[15] Thiemo M. Kier. An integrated modelling approach for flight dynamics, manoeuvre- and gust-loads analysis. In *59th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 8-12 January 2018, Kissimmee, FL, USA*, number AIAA 2018-2209. AIAA, 2018.

[16] Thiemo M. Kier, Martin Leitner, Özge Süelözgen, and Manuel Pusch. An integrated flexible aircraft model for optimization of lift distributions. In *AIAA Scitech 2019 Forum*. American Institute of Aeronautics and Astronautics, jan 2019.

[17] Thiemo M. Kier and Gertjan H. N. Looye. Unifying Manoeuvre and Gust Loads Analysis. In *International Forum on Aeroelasticity and Structural Dynamics*, number IFASD-2009-106, 2009.

[18] Thiemo M. Kier, Mark J. Verveld, and Chris W. Burkett. Integrated Flexible Dynamic Loads Models Based on Aerodynamic Influence Coefficients of a 3D Panel Method. In *International Forum on Aeroelasticity and Structural Dynamics*, number IFASD-2015-179, 2015.

[19] Sebastian J. Koeberle, Julius Bartasevicius, Daniel Teubl, and Szabolcs Tóth. Flipased D1.2 requirements capture for a/c mdo design. Technical report, TUM, 2021.

[20] Sebastian J. Koeberle, Julius Bartasevicius, Daniel Teubl, and Szabolcs Tóth. Flipased D3.5 wing -3 manufacturing design for advanced wing. Technical report, TUM, 2022.

[21] Raymond M. Kolonay and Franklin E. Eastep. Optimal scheduling of control surfaces flexible wings to reduce induced drag. 43(6):1655–1661, 2006.

[22] Raymond M. Kolonay and Franklin E. Eastep. Optimal scheduling of control surfaces on flexible wings to reduce induced drag. *Journal of Aircraft*, 43(6):1655–1661, 2006.

[23] Wolf R. Krüger, Yasser M. Meddaikar, Johannes K. S. Dillinger, Jurij Sodja, and Roeland De Breuker. Application of Aeroelastic Tailoring for Load Alleviation on a Flying Demonstrator Wing. *Aerospace*, 9(10), 2022.

[24] F. Leibfritz and W. Lipinski. Description of the benchmark examples in *COMPl_eib* 1.0. Technical report, University of Trier, 2003.

[25] Manuel Pusch. *Blending of Inputs and Outputs for Modal Control of Aeroelastic Systems: Dissertation*. Verlag Dr. Hut, München, 2020.

[26] 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.

[27] Yasser M. Meddaikar, Johannes Dillinger, Jurij Sodja, and Roeland De Breuker. FLEXOP - Application of aeroelastic tailoring to a flying demonstrator wing. DGLR.

[28] Tomas Melin. A vortex lattice matlab implementation for linear aerodynamic wing applications, 12 2000.

[29] E. Pistolesi. Betrachtungen über die gegenseitige Beeinflussung von Tragflügelsystemen. In *Gesammelte Vorträge der Hauptversammlung 1937 der Lilienthal Gesellschaft*, 1937.

[30] 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.

[31] Charles Poussot-Vassal, Pierre Vuillemin, Olivier Cantinaud, and Florian Sève. Interpolatory methods for generic bizjet gust load alleviation function (accepted). *SIAM Journal on Applied Dynamical Systems*, 2021.

[32] Manuel Pusch, Thiemo M. Kier, Martin Tang, Johannes K. Dillinger, and Daniel Ossmann. Advanced gust load alleviation using dynamic control allocation. In *AIAA SCITECH 2022 Forum*. American Institute of Aeronautics and Astronautics, jan 2022.

[33] W. P. Rodden, R. L. Harder, and E. D. Bellinger. Aeroelastic Addition to NASTRAN. Technical Report NASA CR-3094, NASA, March 1979.

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

[35] 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.

[36] Eric Ting, Daniel Chaparro, Nhan Nguyen, and Gustavo E. C. Fujiwara. Optimization of Variable-Camber Continuous Trailing-Edge Flap Configuration for Drag Reduction. *Journal of Aircraft*, 55(6):2217–2239, 2018.

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

[38] 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.

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