CEAS-GNC-2024-042

EuroGNC Navigation Guidance Control CEAS Bristol, UK June 11th-13th 2024
University of BRISTOL ROYAL AERONAUTICAL SOCIETY AIAA

A Practical Approach to Automated Multiobjective Gust Load Alleviation Control Design in a Structured H_2/H_∞ Framework

Davide Cavaliere PhD Researcher, DLR (German Aerospace Center) Institute of Flight Systems, 38108, Braunschweig, Germany. Davide.Cavaliere@dlr.de

Nicolas Fezans Scientific Advisor, DLR (German Aerospace Center) Institute of Flight Systems, 38108, Braunschweig, Germany. Nicolas.Fezans@dlr.de

ABSTRACT

A gust load alleviation controller must reduce gust loads in the face of a challenging control problem and competing requirements. This may involve time- and frequency-domain objectives, limitations on the size and complexity of the controller, high-order aeroelastic aircraft models, limited system performance, and robustness against complex aeroelastic uncertainties. Tuning such a controller is often difficult and time consuming, and simplified approaches tend to fall short of the performance which can actually be reached. Using a weighted H_2 specification and a set of relatively simple heuristics, gust load alleviation controllers can be automatically iterated to reach a set of performance objectives and constraints. This method is shown to work fairly well and to consistently produce the same outcome, however the design process is sensitive to randomness in the synthesis tool.

Keywords: Gust load alleviation; automated control design; structured synthesis

1 Introduction

Active control technologies have long been recognized as a path to improved aircraft efficiency by easing the bare airframe's design requirements. One such technology, active load alleviation (ALA), seeks to actively limit the sizing loads upon which the structure is designed, reducing its required strength and hence its weight. When introduced relatively late in the development cycle of an aircraft, ALA can only offer modest improvements, e.g., reducing fatigue loads, increasing maximum takeoff weight, or enabling a wingtip extension. Earlier in the design cycle, however, the reduction in sizing loads can result in significant improvements, such as more efficient lift distributions or higher aspect ratio wings. The ability to realistically assess the potential impact and system requirements of ALA functions, and hence to rapidly iterate on their controllers as the aircraft evolves, is therefore of great interest.

Flight loads are mainly composed of maneuver loads and gust loads. Maneuver load alleviation can often be dealt with in a quasi-static manner and has already been widely adopted in industry, however gust load alleviation (GLA) still faces significant technical challenges. The performance limitations imposed by system delays and dynamics are at odds with the dynamic and unpredictable nature of atmospheric turbulence, and they tend to strongly limit realistically achievable performance. The need to deal with aeroelastic modes further complicates control design and introduces issues with sensor and actuator placement, flutter stability (in case of feedback control), and high-order models.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

In the current certification specifications [1, 2], gust loads are defined by frequency-domain continuous turbulence requirements as well as time-domain discrete gust requirements. Continuous turbulence requirements closely resemble weighted H_2 specifications and can be handled by frequency-domain control methods, however time-domain peak requirements are more difficult to work with.

Many examples of GLA control design in the literature have thus preferred to use continuous-turbulence definitions when formulating control specifications, e.g., [3–5]. Several studies have used a direct-optimization approach including time-domain optimization to directly target discrete-gust loads [6–8]. Others have instead opted for adaptive control [9, 10], model predictive control [11–16], and incremental nonlinear dynamic inversion [17].

Studies in which GLA control is included in multidisciplinary optimization (MDO) necessarily adopt some kind of automated control design. In [18], the gains of the feedforward angle of attack-based GLA controller are directly optimized to minimize the gust loads resulting from a set of discrete gusts. Reference [19] directly optimizes a proportional-derivative controller with gust-induced angle of attack as input. The study described in [20] uses a proportional controller receiving angle of attack and whose single gain is preselected by the designer based on potential aerodynamic theory, available control surfaces, and engineering judgement.

The control methods described above face significant barriers before they can be deployed for industrial problems. These include difficult and time-consuming tuning procedures, issues with implementation and certification, and poor scalability to large control problems. The simplified methods used for MDO are, for the most part, too simple, simultaneously ignoring much of the complexity of the control design problem while failing to exploit their full potential.

Multi-objective parameter synthesis (MOPS) methods, such as CONDUIT [21] and DLR's MOPS [22] were developed to address this type of problem in the domain of flight control. Such approaches use generalized optimization methods to deal with non-convex multi-objective control problems. This allows them to directly optimize the controller parameters for any kind of control requirement, including arbitrarily defined time-domain requirements. However, they are best suited to smaller control problems [21]; industrial-size GLA problems with high-order aeroelastic models may quickly become infeasible.

This work aims to help address these issues by automating the GLA control design process. We start with the control tuning loop from the GLA control design methodology used in [23], shown in Fig. 1. In that work, a lidar-based GLA controller for an industrial aeroelastic aircraft is tuned using nonsmooth optimization-based structured H_∞ synthesis. After each synthesis step, the control designer evaluates the behavior and performance of the controller using time-domain simulations, and, if necessary, updates the H_∞ specifications to correct any discrepancies. Due to the generally poor connection between the H_∞ norm and the time-domain performance requirements, this update is far from trivial. A successful control design typically requires skill, experience, and many iterations on the part of the designer. The shortcomings of the control specifications are therefore one of the main difficulties in tuning the controller. Reference [24] develops discrete gust impulse filter (DGIF)-based H_2 specifications which allow the designer to directly specify the signal energy of a discrete gust response. This quantity correlates well with the time-domain peak, and so can be fine-tuned to obtain a desired incremental peak value by varying a single scalar weight.

The main contribution of this work consists of a method to automate the update process by using simple heuristics to add and tune a set of these H_2-DGIF specifications. It relies on a structured H_2/H_∞ control synthesis framework (the same as in [23]), hence exploiting mature, efficient, industry-accepted, and accessible tools [25]. The resulting design process should thus be able to meet discrete gust requirements for an arbitrary number of performance outputs while still taking into consideration other types of non-iterated frequency-domain requirements, e.g., continuous turbulence H_2 specifications or stability margins.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

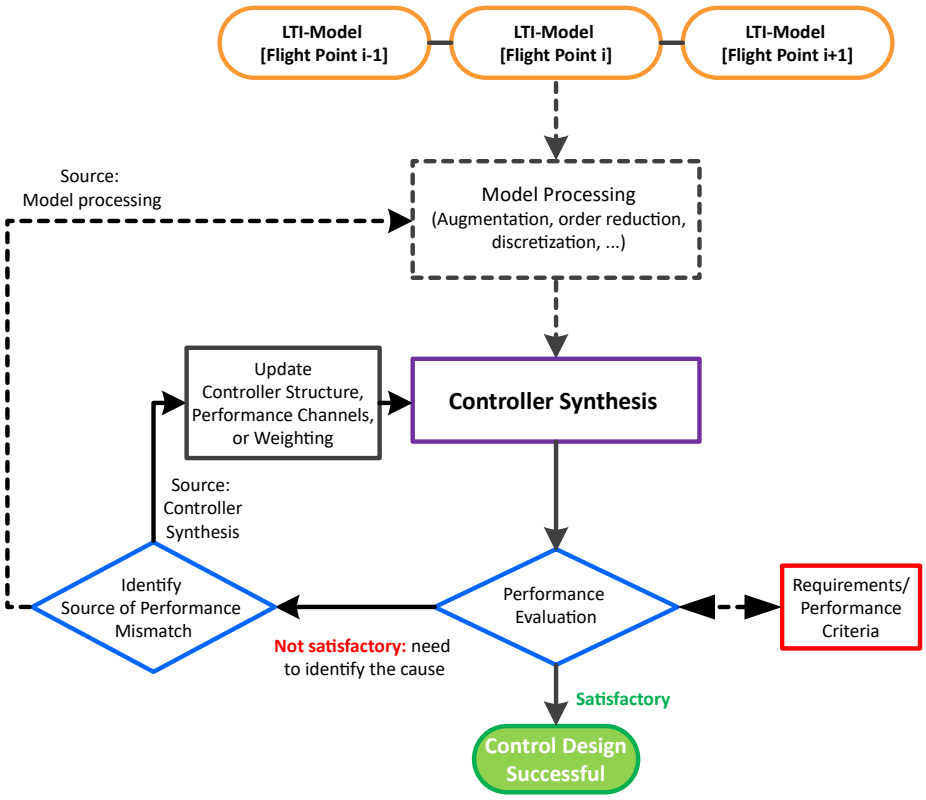


Fig. 1 Control design process activity diagram, modified from [23].

2 The GLA Control Problem

2.1 Gust load requirements

Gust load requirements are defined in EASA CS 25.341 [1]. They are defined both in terms of the peak time-domain responses to discrete gust encounters and the stochastic response to continuous turbulence fields. Here their definitions are briefly recalled.

2.1.1 Discrete gust loads

Discrete gusts are modeled as 1−cosine vertical uniform gusts:

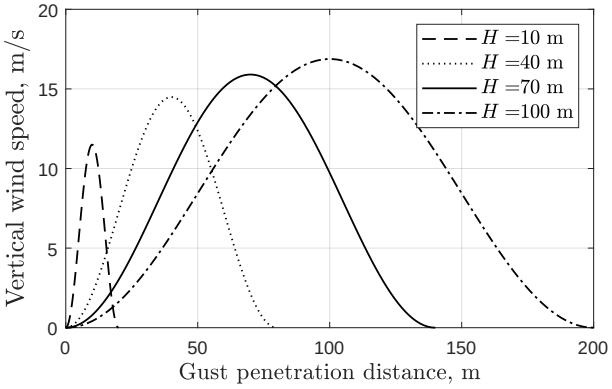
$$U = \frac{U_{ds}}{2}\left(1 - \cos\left(\frac{\pi\, x}{H}\right)\right) = \frac{U_{ds}}{2}\left(1 - \cos\left(\frac{\pi\, t\, V_{TAS}}{H}\right)\right), \quad \text{with} \quad U_{ds} = U_{ref} F_g \left(\frac{H}{107}\right)^{1/6} \tag{1}$$

The gust speed profile U depends on the peak gust velocity U_{ds}, gust gradient H, and gust penetration distance x, alternatively expressed as gust penetration time t multiplied by true airspeed V_{TAS}. The distance between the beginning and the end of the gust is $2H$. As per the certification specifications, H can vary between 9 m (30 ft) and 107 m (350 ft). Peak gust velocity U_{ds} depends on the flight profile alleviation factor F_g, which depends on aircraft design masses and maximum altitude, and reference velocity U_{ref}, which decreases with altitude. Figure 2a plots a few examples of 1−cosine gusts computed at sea level and with $F_g = 1$ to illustrate the relationship between U_{ds} and H.

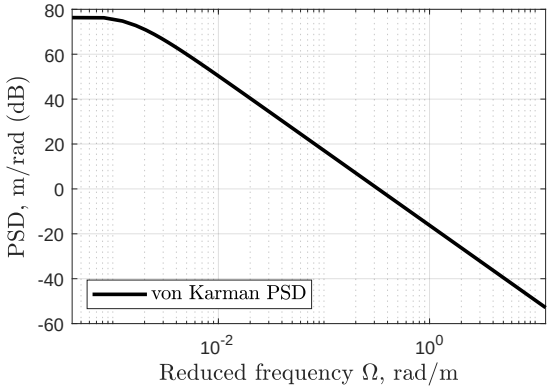
Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

Discrete gust loads must be determined in simulation. The limit loads are taken as the peak positive and negative loads across all values of H for both upward and downward gusts. The gust load envelope is then determined by examining the total loads, i.e., including trim loads. Typically this means that, e.g., upward bending moments on the wing are more critical than downward bending because the wing is subject to upward bending in trimmed level flight.



(a) Discrete gusts with varying values of H.



(b) von Kármán continuous turbulence PSD.

Fig. 2 Discrete gust and continuous turbulence definitions from the certification specifications.

2.1.2 Continuous turbulence

Continuous turbulence is defined as a stationary Gaussian random process [26] characterized by the normalized von Kármán power spectral density (PSD) Φ_{turb} for vertical turbulence, as in Eq. 2 and Fig. 2b. The von Kármán PSD is naturally expressed as a function of the reduced frequency Ω (and hence defined in space), but it also be expressed in terms of angular frequency (ω, rad/s) via the true airspeed V_{TAS}. This requires the PSD itself to be rescaled accordingly [26].

$$\Phi_{turb}(\Omega) = \frac{L}{\pi}\frac{1+\frac{8}{3}\left(1.339\,\Omega\,L\right)^2}{\left[1+\left(1.339\,\Omega\,L\right)^2\right]^{\frac{11}{6}}}, \quad \Phi_{turb}(\omega) = \frac{L}{\pi V_{TAS}}\frac{1+\frac{8}{3}\left(1.339\,\omega L/V_{TAS}\right)^2}{\left[1+\left(1.339\,\omega L/V_{TAS}\right)^2\right]^{\frac{11}{6}}} \tag{2}$$

In Eq. 2, $\Omega = \frac{\omega}{V_{TAS}}$ is the reduced frequency, $L = 762$ m (2,500 ft) is the turbulence scale length, and F_g is the same as for the discrete gust definitions.

For a given quantity of interest y, which could be, for instance, the wing root bending moment or the vertical acceleration at the cockpit, the limit incremental value Δy_{lim} in continuous turbulence is then calculated (for linear systems) as:

$$\Delta y_{lim} = U_\sigma \sqrt{\int_0^\infty |G(\Omega)|^2 \Phi_{turb}(\Omega) d\Omega} \tag{3}$$

$G(\Omega)$ is the frequency response of the quantity of interest in response to vertical turbulence and U_σ the limit turbulence intensity. The square root term in Eq. 3 represents, more generally, the root mean square (RMS) gain of the output quantity.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

2.2 Design considerations

2.2.1 Load objectives

The main objective of a GLA system is to reduce the sizing loads of the structure by reliably reducing the gust loads. This clearly implies that at any given point in the structure, there is little benefit to reducing the gust loads beyond the next-highest load type (e.g., maneuvers); at best, a modest over-reduction may improve robustness by acting as a buffer against performance degradation. In many practical cases, part or even most of the structure is sized by other loads (e.g., landing or maneuver loads) or by entirely different requirements (e.g., aeroelastic stability, manufacturing constraints). As a consequence, GLA requirements should only aim at achieving a limited reduction of the gust load envelope in specific parts of the structure.

2.2.2 Actuation limits

For the purpose of control design, commercial aircraft control surfaces are often modeled as linear 2nd-order systems with nonlinear position and rate saturation limits. Saturated actuators can strongly degrade stability in case of feedback control. Pure feedforward control by definition cannot destabilize the system (assuming the controller is stable), but it is still subject to performance degradation and may contribute to destabilizing other control loops. In the absence of control techniques which can explicitly deal with such nonlinearities (such as MPC), it is highly advisable to ensure, at a minimum, that the controller does not attempt to exceed the saturation limits.

2.2.3 Secondary objectives

GLA functions exist within the broader context of the aircraft system; although their main function is to reduce gust loads, their effect on other aspects of the system must not be neglected. Once the targeted gust load envelope can be achieved within the specified constraints, any residual control authority is better spent improving performance on secondary objectives. These can include, among others: passenger comfort, handling qualities, actuator wear, fatigue loads, energy consumption, and robustness.

2.3 Aircraft Model Description

2.3.1 SE^{2}A Mid-Range Aircraft Design

The aircraft configuration used in this work is the Sustainable and Energy-Efficient Aviation (SE^{2}A) Mid-Range (MR) Aircraft [27]. It is comparable in role and capabilities to an Airbus A320-200, and is designed to balance direct operating costs against CO_2-equivalent emissions (including contrail effects) by exploiting advanced technologies as well as an alternative mission profile. Compared to the A320-200, the resulting design notably features a lower and slower (and hence low-contrail) nominal mission profile, reduced wing sweep, increased wingspan and aspect ratio, lower wing loading, and over-wing engines. The structural sizing was performed under the assumption that load alleviation functions would be included, so only maneuver loads were considered, and the pull-up maneuvers were limited to $2g$ instead of the normal $2.5g$ to account for a hypothetical MLA function. Gust loads were not taken into account at all under the assumption that they would be reduced by active GLA functions.

2.3.2 Flight Dynamics Model

The aircraft is modeled in a nonlinear aeroelastic flight dynamics simulation environment, presented in [28]. This flight dynamics model (FDM) includes mid-fidelity unsteady aerodynamics and a flexible structure derived from the high-order finite-element structural model. It is capable of trimmming,

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

linearizing, and simulating the aircraft throughout the flight envelope. Using the Mode Displacement Method [29], cut loads (including axial, shear, bending moments, and torsional moment) are computed at 134 load stations evenly distributed along the elastic axes of the wings, tailplanes, and fuselage.

2.3.3 Mass Cases

In this work, four mass cases are considered: Maximum Takeoff Weight (MTOW), Operating Empty Weight (OEW), Maximum Zero-Payload Weight (MZPW), and Maximum Zero-Fuel Weight (MZFW). MTOW and OEW are 'classical' limit cases: at MTOW the aircraft's wing tanks are fully fueled and the fuselage is fully loaded with passengers and cargo, and at OEW the aircraft is emptied of all usable fuel, passengers, and cargo. MZFW and MZPW are intermediate configurations: at MZFW the fuselage is fully loaded (like at MTOW) and the wing tanks are empty, and at MZPW the fuselage is empty and the wings tanks are full. This set of mass cases should cover most of the realistic range of variations in symmetrical aeroelastic dynamics.[1]

2.3.4 Loads and critical flight point

Reference [28] describes the open-loop discrete gust load envelopes of the SE^{2}A MR aircraft. It was shown that for this aircraft, gust loads tend to increase with airspeed, such that the most critical gust cases tend to occur at the maximum cruise design speed V_C/M_C (see [1, § 335(a)]). The most critical flight point for the inner-wing bending loads corresponds to the MTOW mass case at the maximum-TAS point along the V_C boundary: altitude 6000 m, EAS 177 m/s, TAS 241 m/s, and Mach 0.76.

As discussed in the introduction, a GLA function should aim to ensure that the structure is no longer sized by gusts by reducing the gust loads until they are below the next-strongest load type. As a basis for comparison, a maneuver load envelope has therefore also been computed by performing 'balanced' (static) and 'checked' (dynamic) pitch maneuvers according to the certification specifications [1, § 331(b) and (c)(2)]. The open-loop gust and maneuver load envelopes can be seen in Fig. 15 (blue and red curves, respectively). The gust loads are sizing in practically all load stations, and there is a fairly substantial 'gust load margin' on the inner half of the wing semispan. Though unusual, this is not entirely surprising given the gust-prone design characteristics (e.g., low wing loading) and simplified structural sizing procedure.

2.3.5 System dynamics

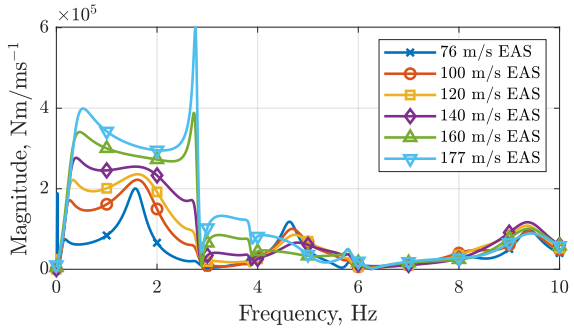
To illustrate some dynamics of the problem, Fig. 3 plots a series of frequency responses of the transfer function from vertical wind input to wing root bending moment at varying flight conditions. Figure 3a shows the variation in response with changing airspeed at a constant altitude of 6000 m and with the MTOW mass case, whereas Figure 3b shows the variation with changing mass distribution at the critical flight point 6000 m and 177 m/s EAS.

Most of the principal dynamic modes of the flexible aircraft can be observed in the resonance peaks in these plots. The phugoid (*PH*) mode is barely visible around 0 Hz and the short period (*SP*) mode is found between 0.1 and 1 Hz. The first symmetrical aeroelastic wing bending mode (*1WB*) lies between 1 and 2 Hz at low speeds, and becomes faster and more strongly damped with increasing airspeed. The mode between 2.5 and 3 Hz, referred to here as *1WT*, involves symmetrical wing torsion and rear fuselage bending, and is closely related to the unfavorable aeroelastic characteristics of engines mounted

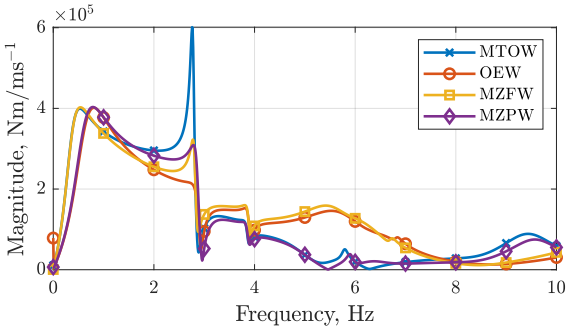
[1] Variations in rigid-body dynamics due to center of gravity (CG) position are somewhat neglected in this set of mass cases. They can significantly affect control performance if pitching dynamics are involved, however CG position can be reliably estimated in-flight and used for gain scheduling.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.



(a) Varying EAS at altitude 6000 m and with MTOW.



(b) Varying mass cases at altitude 6000 m and airspeed 177 m/s

Fig. 3 Frequency responses of the transfer function between vertical gust input and wing root bending moment.

behind the elastic axis of the wing.[2] The in-plane wing bending mode (*1IP*) is just under 4 Hz, and the second symmetrical aeroelastic wing bending mode (*2WB*) varies between approximately 4.5 and 5.5 Hz, depending largely on mass case. With increasing airspeed, *1WB* notably disappears as it becomes faster and better-damped while the adjacent *SP* and *1WT* modes increase in magnitude, and *1WT* becomes particularly poorly damped for the MTOW mass case.

2.3.6 Control surfaces

The aircraft only uses conventional trailing-edge flap control surfaces. Their actuation dynamics are modeled using second-order low pass transfer functions with natural frequency of 20 rad/s and a damping ratio of 1. Their rate and deflection limits are set to ±40 deg/s and ±20 deg, respectively, based on common values for civil airliners [30]. Figure 4 illustrates the control surface configuration used in this study. Four sets of control surfaces are included: elevators (δe), inner and outer ailerons (δa^{in} and δa^{out}, respectively), and mid-wing flaperons (δa^{flp}).

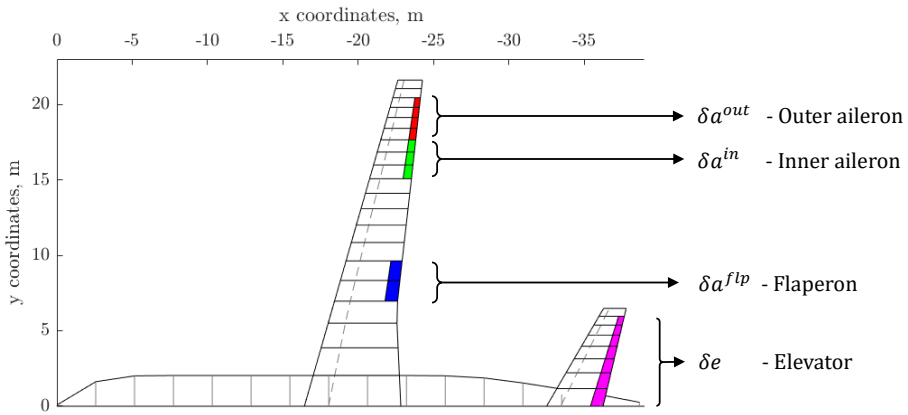


Fig. 4 Control surface and onboard sensor configuration. All control surfaces are symmetrical.

[2]This can be understood in terms of the 'deadweight' moment produced by the engine mass. In particular, an increase in lift which creates an upward wing bending motion also creates a leading-edge-up torsional moment (due to the engine mass). This acts in opposition to a typical (stabilizing) bending-torsion coupling, effectively reducing the damping of this mode and leading to a lower flutter speed.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

2.3.7 Lidar-based wind estimation system

The aircraft is also equipped with a lidar-based wind estimation system. It is based on a Doppler wind lidar capable of detecting the relative wind speed along its line-of-sight up to a couple hundred meters away. Combined with a scanning system and a wind estimation algorithm, it provides an estimate of the vertical wind field profile along the predicted flight path. The combination of sensor, scan pattern, and estimation algorithm results in an imperfect estimate of the true wind field, such that it effectively behaves like a low-pass filter [31]. The inner workings of this system are fairly complex and fall well beyond the scope of this work; for more details, interested readers may refer to [4, 31–33].

Figure 5 schematically illustrates the structure of the wind field estimate. It is defined by a reference position located at the aircraft nose, h_{pre} evenly-spaced points ahead of the reference position, and h_{post} points to the rear. Here, this wind field estimate is refreshed at the control system's sampling rate (100 Hz), with a discrete number of evenly spaced points separated in time by a single time step, i.e. the distance between adjacent points is 10 ms $\cdot$ V_{TAS} meters. This choice of spacing facilitates the design of preview controllers in a linear discrete-time framework: 10 ms is equivalent to a single discrete time delay. In this work, $h_{pre} = 40$ and $h_{post} = 25$, such that the controller may take advantage of an anticipation time of 0.4 s (nearly 100 m at maximum TAS) as well as some knowledge of the wind field acting on the aircraft.

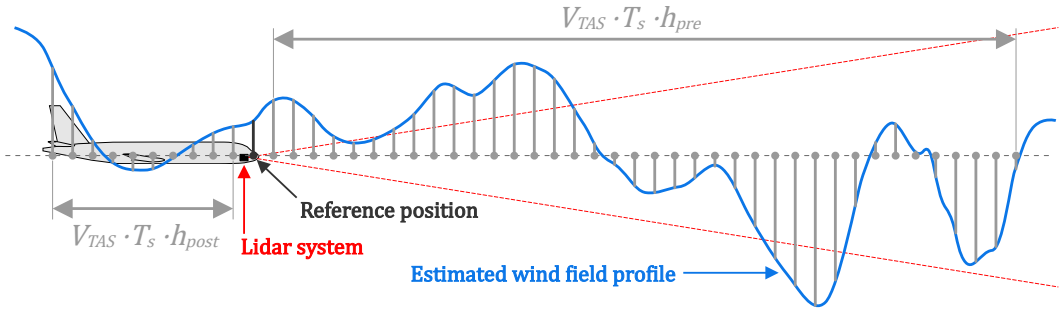


Fig. 5 Schematic of the vertical wind field estimate provided by the lidar system.

3 Control design

3.1 Multimodel structured H_2/H_∞ synthesis

Building on the methods developed and exploited in references [23, 24, 31, 34–36], Fig. 6 illustrates the GLA synthesis problem in standard form with disturbance input **w**, control inputs **u**, measured outputs **y**, and performance outputs **z**. The control inputs include the commanded pairwise-symmetrical deflections of the aircraft's control surfaces. The disturbance input consists of a single vertical gust signal well ahead of the aircraft. The lidar system and wind field estimation model F_{lidar}, developed in Ref. [31], applies the wind field estimation system's dynamics to produce the estimated wind field vector $\mathbf{w}_{est}$ as well as the delayed model gust input w_{turb}. To accommodate this filter and to ease the implementation of the final controller as a digital control system, the synthesis problem is assembled and tuned in discrete time with a sampling frequency of 100 Hz. Aside from the estimated wind field, the measured outputs also include measured outputs from sensors aboard the aircraft $\mathbf{y}_{sens}$, such as accelerometers, rate gyroscopes, and angle of attack vanes. The performance outputs include any model outputs upon which a control specification may be placed, usually including (but not limited to) cut loads, local accelerations, and control surfaces rates and positions.

This work relies on a structured H_2/H_∞ synthesis framework based on the highly efficient non-smooth optimization techniques included in the Matlab Control System Toolbox function *systune*. Aside from allowing the designer to directly specify the order and structure of the controller, such non-smooth

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

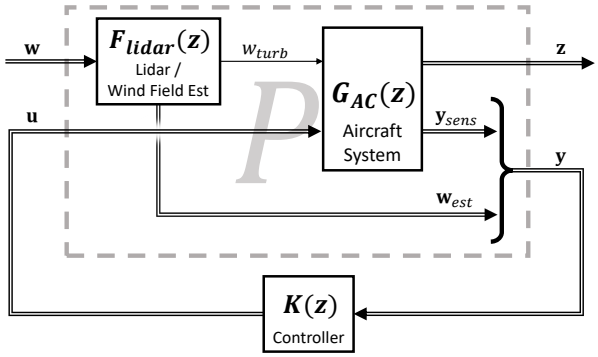


Fig. 6 Control synthesis problem including lidar-based wind estimation.

optimization tools have shown themselves to be effective at dealing with relatively high-order systems [25]. Bearing in mind the high order of the linear models considered here (full-order models over 1000 states, reduced for control design to around ≈100 states), this represents an enormous advantage over older full-order synthesis methods. It is important to note, however, that unlike some full-order synthesis methods, these techniques do not come with any guarantee of reaching a global optimum. If the synthesis process fails to reach the specified stopping condition on the first run, it is common practice to attempt a few more random starts to try to find a more favorable local optimum.

Using a multi-model formulation, as in Fig. 7, a single controller can be tuned against multiple system models at the same time. This is particularly useful when attempting to deal with certain kinds of uncertainties, for instance aeroelastic uncertainties caused, e.g., by variations in the aircraft's mass distribution. Such uncertainties can change the shape and natural frequencies of the flexible modes as well the aircraft's trimmed flight shape, causing complex variations in actuator, sensor, and aircraft dynamics. Such uncertainties are difficult to accurately identify and model using parametric uncertainty methods (such as μ-analysis/synthesis), whereas it is fairly simple to generate multiple system models with varying mass distributions. Although multi-model robustness may not provide any concrete guarantees outside of the specific models used for synthesis, the guarantees of robustness obtained with a naive implementation of μ-methods may be entirely disconnected from the physical reality of the control problem.

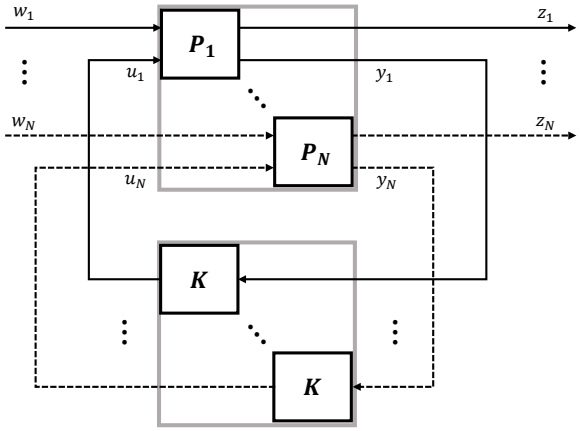


Fig. 7 Multi-model control synthesis for a single controller.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

Performance specifications may be defined in terms of frequency-weighted H_2 or H_∞ norms of transfer functions from $\mathbf{w}$ to $\mathbf{z}$. *Systune* distinguishes between *hard* and *soft* specifications, i.e., *soft* specifications are minimized subject to the *hard* specifications being met. More formally, this is:

$$\begin{aligned} \underset{k}{\text{minimize}} \quad & \max_i \left\| T_{w_i \to z_i} \right\| \\ \text{subject to} \quad & \max_j \left\| T_{w_j \to z_j} \right\| \leq 1, \end{aligned} \tag{4}$$

where k is the set of all tunable controller parameters, $T_{w \to z}$ denotes a weighted transfer function from w to z, $\|\cdot\|$ indicates an H_2 or H_∞ norm, and i and j refer to the sets of soft and hard specifications, respectively. In practice, the optimization is usually stopped when the *soft* requirements pass below 1 to avoid overdesign and to speed up the design process.

In this work, all such specifications are only defined using SISO transfer paths, and the weights are normalized such that the specification is met if the weighted norm is ≤ 1.[3] In the multi-model case, specifications may be defined for all models at the same time or for a subset of them.

3.2 Defining discrete gust specifications

3.2.1 H_2-DGIF specifications

The discrete gust specifications used here are based on the work reported in [24]. In brief, the H_2 norm may be interpreted as the signal energy (i.e. the L_2 norm) of the time-domain impulse response of a system [37]. Using a weighting filter which produces a $1-\cos$ine shape in response to an impulse, a weighted H_2 specification may be defined which specifies the signal energy of the response to a discrete gust of that length.

Reference [24] provides an expression for precisely such a filter, named the discrete gust impulse filter (DGIF) $F_{DG}(s)$. The DGIF is of order 5 and is parametrized with airspeed (V_{TAS}), gust length (H), and peak velocity (U_{ds}), so it can be directly computed for any given flight point and gust case. For a generic SISO transfer function $T_{w \to z}(s)$, the specification is thus expressed:

$$||W_{DG}(s) T_{w \to z}(s)||_2 \leq 1, \quad \text{with} \quad W_{DG}(s) = K_{norm} K_{weight} F_{DG}(s) \tag{5}$$

The normalization factor K_{norm} is computed, if possible, such that $||K_{norm} F_{DG} T_{w \to z}(s)||_2 = 1$. The value of the weighting gain K_{weight} is then chosen such that for a desired relative change η in the L_2 norm of the discrete gust response, $K_{weight} = 1/\eta_{obj}$. If $||K_{norm} F_{DG}(s) T_{w \to z}(s)||_2 \approx 0$, as is typically the case for actuator deflections of the open-loop aircraft, the specification cannot be normalized and $K_{norm} = 1$.

3.2.2 Iterative tuning for peak load requirements

Seeing as the quantity of interest is not the L_2 norm, but rather the time-domain peak response, the final value of K_{weight} cannot (reliably) be chosen a-priori and must be iteratively adjusted to reach the desired limit. The L_2 norm of the discrete gust response is fortunately well-correlated with the time-domain peak [24], so increases in K_{weight} result in approximately proportional decreases in the time-domain peak. Based on this idea, the full iteration and update rules are developed in Sec. 3.4.

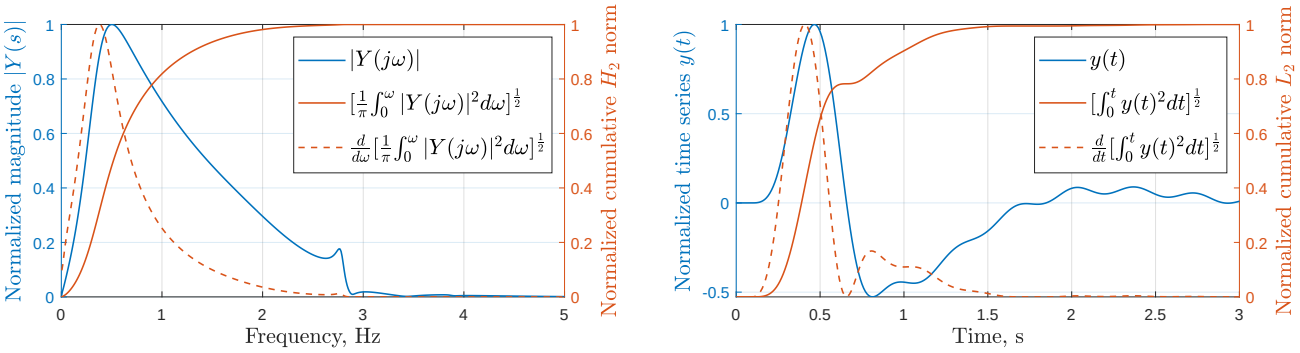
A full demonstration of these properties is beyond the scope of this paper, however Fig. 8 may provide some intuition as to the underlying causes. In both the time- and frequency-domain, it compares

[3] It is more common to normalize the model rather than the specifications, however to accommodate the possibility of applying multiple specifications using different norms to the same transfer function, it is simpler to normalize the weighting functions instead. This is particularly helpful when working with frequency-weighted H_2 specifications, as the H_2 norm does not satisfy the multiplicative property.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

the response of a DGIF-weighted transfer function with its cumulative 2-norm (i.e., with a progressively increasing upper integration limit) and its derivative. The question is: which part of the response contributes the most to the cost function, and hence what will the optimization algorithm try to reduce first? The answer is indicated by the derivative of the cumulative norm: in sections where this is high, the norm increases quickly, so a descent gradient which induces a reduction in that part of the response is likely to be steeper. In Fig. 8a, the norm is dominated by the first peak, and 80% of the contribution to the total norm occurs below 1 Hz. In Fig 8b, the time-domain is also clearly dominated by the first, positive peak, which alone contributes approximately 80% of the overall norm. It is interesting to note how disproportionately the absolute value of the peak affects the norm derivative: the second (negative) peak is approximately half of the first, but the norm derivative's peak value during the first peak is over 5 times greater than during the second.



(a) Frequency-domain **(b) Time-domain**

Fig. 8 Comparison of the DGIF-weighted (with with $H = 70$ m) frequency response and impulse response (blue) with their respective cumulative 2-norms (solid red) and their derivatives (dashed red). Based on the transfer function from vertical gust input to wing root bending moment at 177 m/s EAS, 6000 m altitude, and MTOW.

3.3 Other specifications

3.3.1 Continuous turbulence

Continuous turbulence requirements for a linear system can be directly expressed as H_2 requirements, as explained in [24]. Using the same notation as above, the specification reads:

$$||W_{CT}(s)T_{w\to z}(s)||_2 \leq 1, \quad \text{with} \quad W_{CT}(s) = K_{norm}K_{weight}F_{CT}(s) = \frac{\sqrt{2\pi}U_\sigma}{z_{obj}}F_{CT}(s) \tag{6}$$

The continuous turbulence filter $F_{CT}(s)$, an example of which can be found, e.g., in [38, p.46], approximates the von Kármán PSD. Here, again, K_{norm} may be computed such that $||K_{norm}F_{CT}(s)T_{w\to z}(s)||_2 = 1$ so as to specify a relative variation via $K_{weight} = 1/\eta_{obj}$. Unlike the discrete gust specifications, the rightmost expression of Eq. 6 permits the absolute desired value z_{obj} to be specified directly. Consequently, no iterations are necessary to meet continuous turbulence objectives.

3.3.2 Ride quality

Following the example of [39], ISO 2631-1 allows ride quality/passenger comfort metrics to be addressed. This standard expresses human sensitivity to vertical vibration, horizontal vibration, and motion sickness as a set of frequency weightings. Much like the continuous turbulence requirements, the criteria in ISO 2631-1 are defined as weighted 2-norms in the frequency domain. They may therefore be expressed as H_2 specifications in much the same way as the continuous turbulence specifications above.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

As weighting functions, the low-order approximations for the ISO2631-1 frequency weightings derived in [40] may be used. For example, the fifth-order approximation of the motion sickness weighting is shown below [40]:

$$W_f(s) = \frac{0.1457s^4 + 0.2331s^3 + 13.75s^2 + 1.705s + 0.3596}{s^5 + 7.757s^4 + 19.06s^3 + 28.37s^2 + 18.52s + 7.23} \tag{7}$$

3.3.3 H_∞ templates and bandwidth

H_∞ specifications can be chosen by the designer to impose upper bounds on the frequency response of a transfer function. For a generic SISO transfer function $T_{w\to z}(s)$, the expression is:

$$||K_{norm}W(s)T_{w\to z}(s)||_\infty \leq 1 \tag{8}$$

These types of specifications are not necessary for achieving the load requirements, however they may still be generally useful for enforcing hard bounds on the behavior of the system. One such application is the use of an H_∞ template to bound controller's bandwidth, for instance:

$$W_{bw}(s) = K_{weight} \underbrace{\left(\frac{s+\omega_{ro}}{s}\right)^{n_{ro}}}_{\text{Roll-on}} \underbrace{\left(\frac{s+\omega_{bw}}{\omega_{bw}}\right)^{n_{bw}}}_{\text{Roll-off}} \tag{9}$$

This template imposes an $n_{ro} \times 20$ dB/decade roll-on slope ending at ω_{ro} rad/s to avoid static deflections and measurement bias errors, and an $n_{bw} \times -20$ dB/decade roll-off slope starting at ω_{bw} rad/s to limit its sensitivity to higher-frequency modes, noise, and unmodelled uncertainties. This sort of specification is a simple, common-sense way of robustifying the controller, however selecting or tuning the parameters of this weighting function is not trivial.

3.4 Automated design process

3.4.1 Concept and definitions

At this point, it is useful to introduce a few definitions. Firstly, we distinguish between *soft* requirements, which define the desired performance of the system, and *hard* requirements, which define acceptable limits. A controller which meets both *soft* and *hard* requirements is a *success*; if it does not meet the *hard* requirements, it is a *hard failure*; and if it meets the *hard* but not the *soft* requirements, it is a *soft failure*.

A clear distinction is also made between the terms *requirement* and *specification*. A *requirement* (also denoted *Reqt*) refers to objectives/constraints expressed in the natural form, e.g. time-domain response peaks for discrete gust requirements. A *specification* (*Spec*) refers to the frequency-domain control specifications used by the synthesis tool. Requirements have a multiplicity of specifications: each specification belongs to only one requirement, however each requirement may have any number of specifications. Depending on whether they derive from a *hard* or *soft* requirement, specifications are themselves *hard* or *soft*

A final distinction must be made between *iterable* (*Iter*) and *noniterable* (*Noniter*) requirements. In this work, the only type of iterable requirement is the discrete-gust time-domain peak requirement, for which the number of specifications is determined during the control design process via the iteration rules, described below. All other requirements are noniterable, i.e., they are directly expressed as a single frequency-domain specification.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

The concept for the automated design loop can thus be represented as in Fig. 9. The designer defines requirements, which the design loop converts into specifications (iteratively, if necessary); *systune* synthesizes controllers, and once their performance meets all requirements or other stopping conditions are met, the final controller is delivered to the designer.

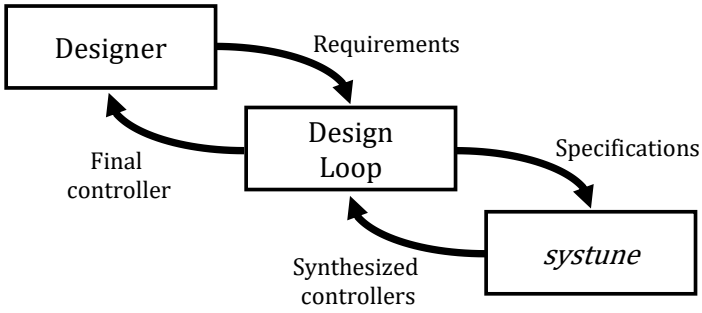


Fig. 9 Automated design loop concept.

3.4.2 Control design framework

Figure 10 provides a more detailed overview of the proposed GLA control design process.

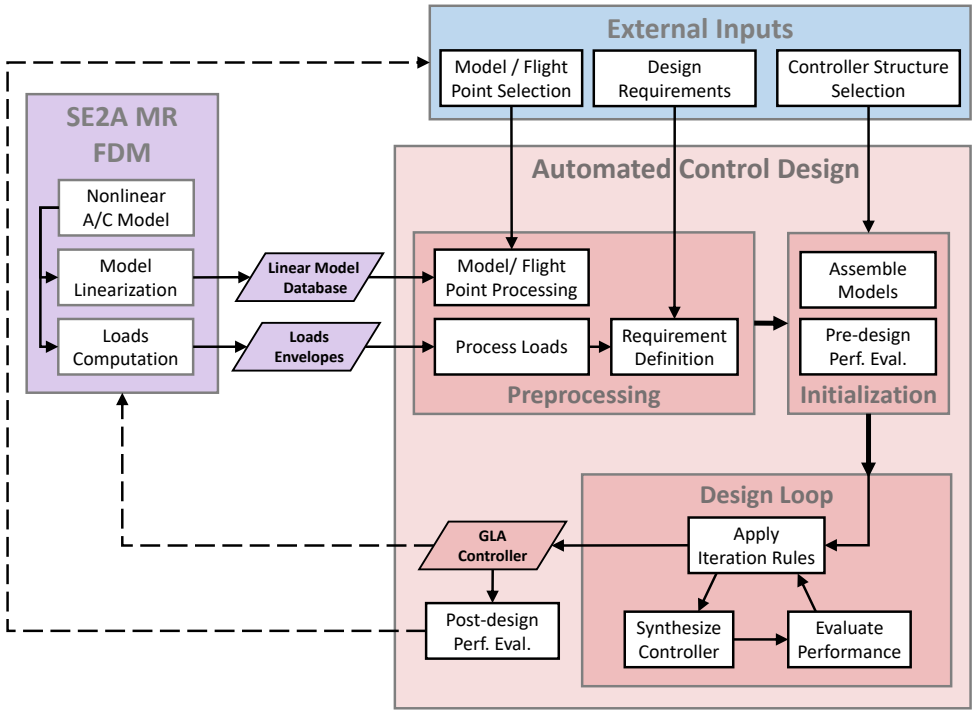


Fig. 10 Automated GLA control design framework.

The full nonlinear flight dynamics model is used to compute load envelopes and a database of linear models covering the full flight envelope and all relevant mass cases.

A set of top-level inputs to the process must be provided by an external source, for instance the control designer or an outer optimization loop. These include the untuned controller, which defines its inputs (**y**), outputs (**u**), internal structure, tunable parameters and their initial values, and bounds on the tunable parameters, if applicable. The flight points, mass cases, and system configurations (e.g., time

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

delays and lidar settings) included in the control design process must also be chosen. Finally, the design requirements in terms of gust load reduction, actuator limits, and custom frequency-domain specifications (e.g., bandwidth limitations) are selected.

In the preprocessing step of the control design process, the external inputs are checked against available models and loads them. The load envelopes are processed such that each model load output is associated with the maximum and minimum gust and maneuver loads, the trimmed load, and the targeted limit loads. The design requirements are then assembled: for each load station (e.g. wing root), type (e.g. vertical bending moment) and load requirement (e.g. discrete gusts, continuous turbulence), a single requirement is defined.

For example, each discrete gust requirement defines its criticality (hard or soft), the name of the output signal, its trimmed value, its open-loop value, its targeted value, and whether it is an absolute or relative value. A relative value is interpreted as a change relative to its open-loop value, and if the trimmed value is 0, the open-loop and targeted values are interpreted as incremental values.

The initialization step begins by preparing the linear design and evaluation models. This process, also described in some detail in, e.g., [41], includes input and output reduction according to the inputs/outputs specified by the controller and by the requirements, model order reduction, conversion to discrete time, integration of the lidar model, and additional augmentations, e.g., time delays or model uncertainties. The (untuned) controller is then connected to the design models and an initial performance evaluation (denoted later as iteration 0) is conducted. This establishes a starting point for the design loop; depending on the initial values of the controller parameters, this may be equivalent to an open-loop evaluation.

The control design loop can then begin, alternating between applying the iteration rules (described below), synthesizing controllers, and evaluating the system's performance. When the stopping conditions defined in the iteration rules are met, the loop ends, producing the final tuned controller and a final post-design performance evaluation.

3.4.3 Performance evaluations

The performance evaluations mentioned above serve to establish the system's performance both in terms of the specifications and of the requirements. Here, frequency-domain evaluations use the (reduced-order) design model(s) and time-domain evaluations use full-order linear models, but other choices are possible; for instance, the full nonlinear flight dynamics model from [28] or a multirate evaluation environment such as the one in [41] could be used for time-domain simulations.

Specification performance η_S corresponds to the values of the frequency-domain norms through which the specifications are defined, discussed above in 3.1. Requirement performance η_R is instead linked to requirement type, and is defined such that a requirement is met when $\eta_R \leq 1$. For noniterable requirements, $\eta_R = \eta_S$. For iterable (i.e., discrete gust) requirements, η_R refers to the ratio between the actual and the targeted incremental time-domain peaks, respectively Δy_{peak} and Δy_{lim}. Time-domain peaks are found via a series of gust encounter simulations, with the range of gust lengths and corresponding peak gust velocities computed according to the certification specifications [1]. If the targeted value is specified as an absolute value, the incremental target can be found by subtracting the trim value. The performance is then computed as $\eta_R = \Delta y_{peak}/\Delta y_{lim}$. When a specification or requirement is applied to more than one model in a multimodel control problem, η_S and η_R are computed individually for each model.

3.4.4 Design loop iteration

Figure 11 is a flowchart depicting the iteration rules of the control design loop. For brevity, not all aspects of the rules are described here, however a few main points are discussed. For a more intuitive view of the properties of these rules, see the examples presented in Section 4.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

Firstly, as explained above, three outcomes are possible: *hard fail* if the *hard* requirements cannot be met, *soft fail* if the *soft* requirements cannot be met, and *success* if all requirements are met. *Hard* requirements are given absolute priority over *soft* ones to allow them to converge to acceptable values first. Failure to meet a noniterable requirement, even for a *soft* requirement, immediately results in failure. If noniterable requirements are met, it implies that at least one iterable requirement is not, so then a set of specification update rules are applied.

Figure 12 shows the specification update rules in more detail. Two types of operations are foreseen in updating the iterable specifications: *tightening* and *relaxing*. These operations, which are identical for both hard and soft requirements, are shown in detail in Fig. 13. *Tightening* involves increasing specification weights so as to drive down the time-domain peak. For each requirement, only the specification corresponding to the critical gust length H_{crit} is tightened. H_{crit} is identified from the results of the latest performance evaluation as the gust length corresponding to the largest peak response. If none yet exists for that gust length, a new one is added. K_{weight} for new specifications is a guess, so they are then *pretightened* to ensure that their performance $\eta_S \geq 1$ at the beginning of the next synthesis step, hence forcing the controller to be retuned.

Relaxation instead reduces K_{weight} for all specifications with $\eta_S > 1$ (before relaxation) such that they have an $\eta_S \approx 0.99$ at the beginning of the next synthesis step. This serves to 'reset' the specifications if K_{weight} is too high. For *hard* requirements, each specification can only be relaxed once to help protect against an accidental 'overshoot' triggering a hard fail. *Soft* requirements are relaxed any time the *hard* specifications need to be tightened to 'get them out of the way' so as to minimize the time, computational effort, and iterations required for the *hard* requirements to converge.[4] Accordingly, the *tightening* rules are different before and after a specification has been *relaxed*: before, K_{weight} is increased as aggressively as possible to 'close the gap'; after, it is gradually raised to avoid overshooting the limit.

4 Application and results

4.1 GLA design case

4.1.1 Design goals

To provide a reasonably realistic example of a gust load alleviation design problem, the following design goals are chosen:

1) Reduce inner-wing vertical bending gust loads to those of the symmetrical maneuver loads.
2) Limit inner-wing torsional loads to those of the open-loop gust load envelope.
3) Limit aft fuselage bending gust loads to those of the open-loop gust load envelope.
4) Limit HTP bending and torsional loads to those of the open-loop gust load envelope.
5) Avoid degrading ride quality with respect to the open-loop aircraft.
6) Limit the control surface rates to ±40 deg/s and deflections to ±20 deg.
7) Ensure the frequency response of control surface commands tends to 0 at frequencies $\omega \to 0$ and $\omega \to \infty$.

The open-loop gust and maneuver load envelopes referred to here are computed for all mass cases across the entire flight envelope, whereas the design is performed and evaluated only at the critical flight point. To simplify the problem and focus on the main capability of the method presented here, gust loads are computed using only discrete gusts. It is worth noting that at least some of these requirements can be

[4] In the situation in which the *hard* specifications can be met but the *soft* specifications cannot, the synthesis tool makes a costly effort to reach the local minimum. Once it fails, it repeats the process with random restarts up to a prespecified limit.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

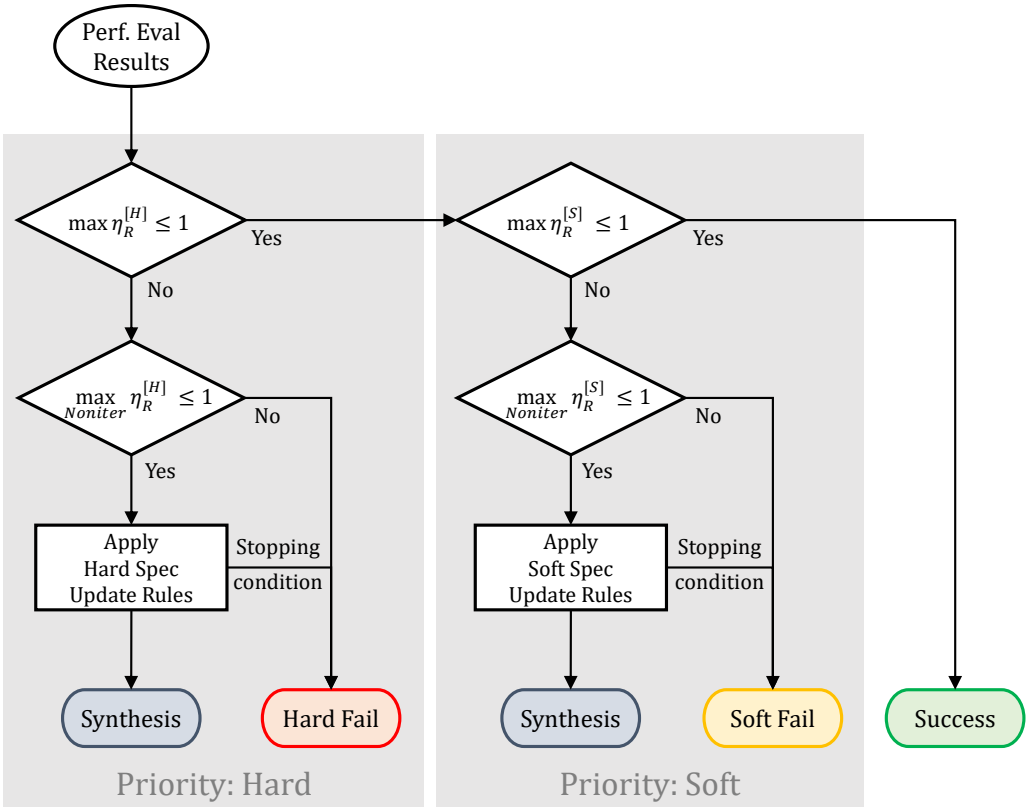


Fig. 11 Flowchart of the control design loop iteration rules. Superscripts $[H]$ and $[S]$ indicate *hard* and *soft* requirements.

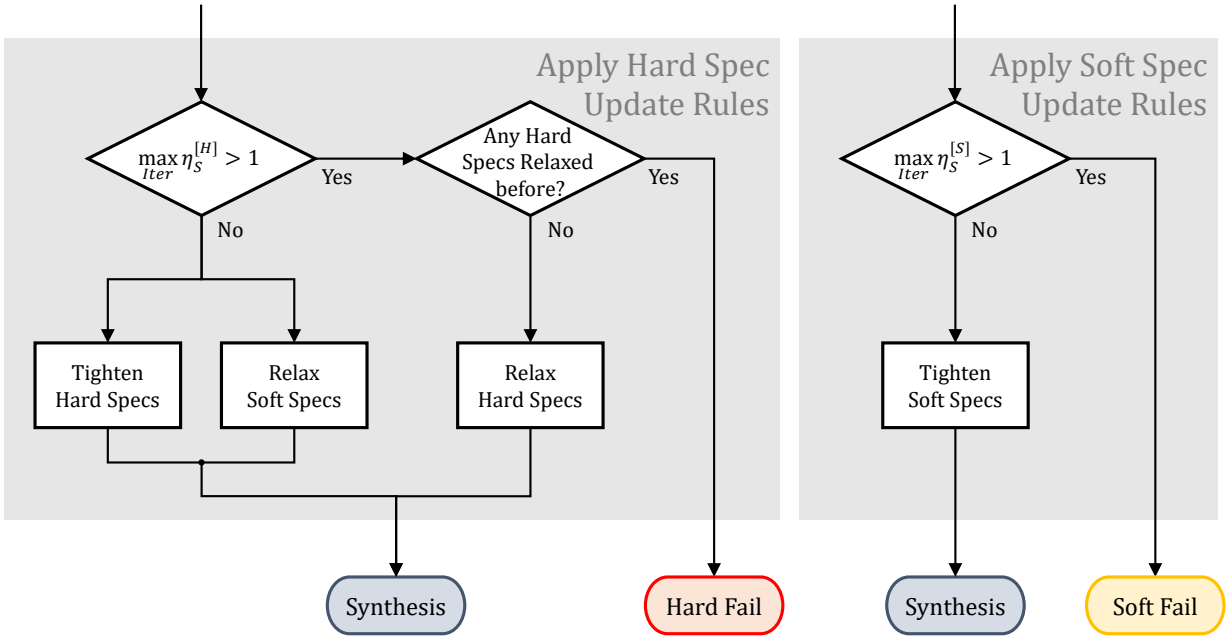
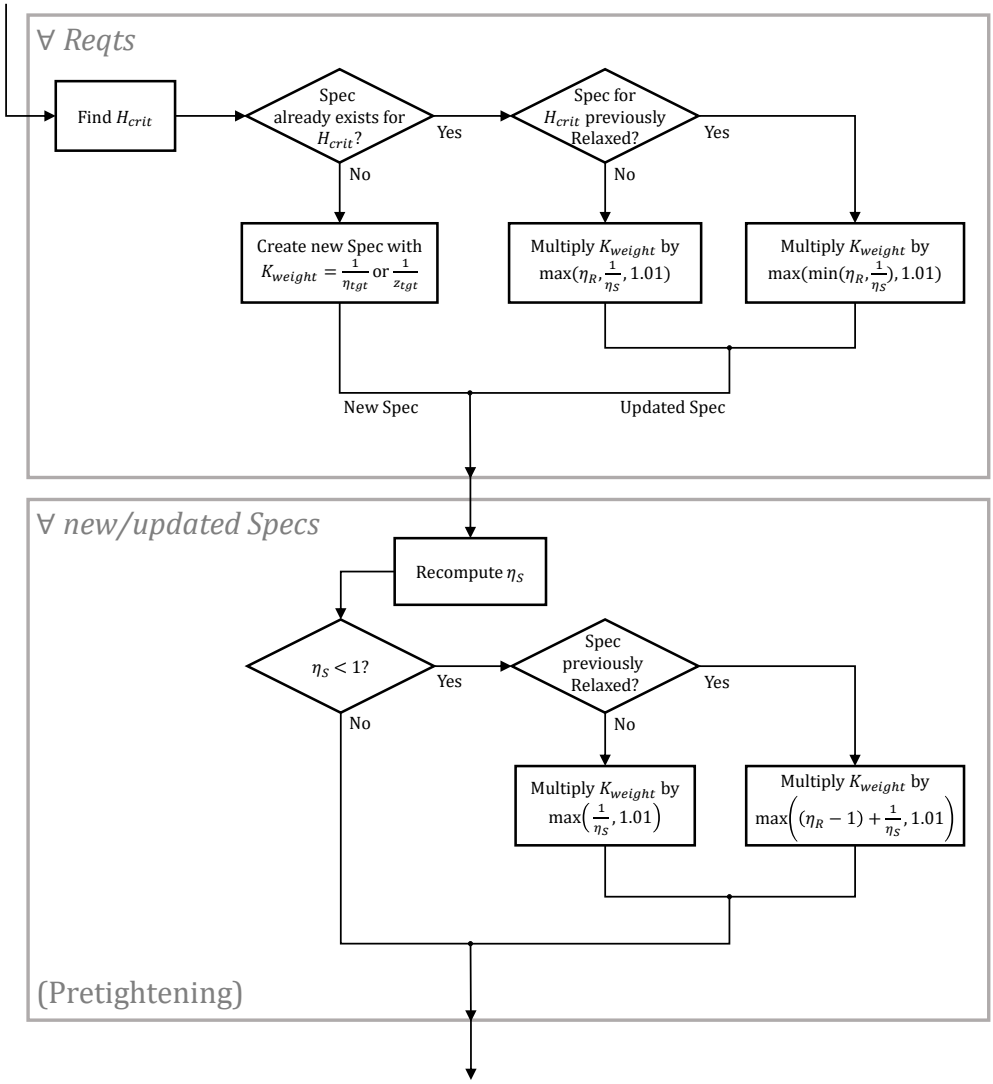


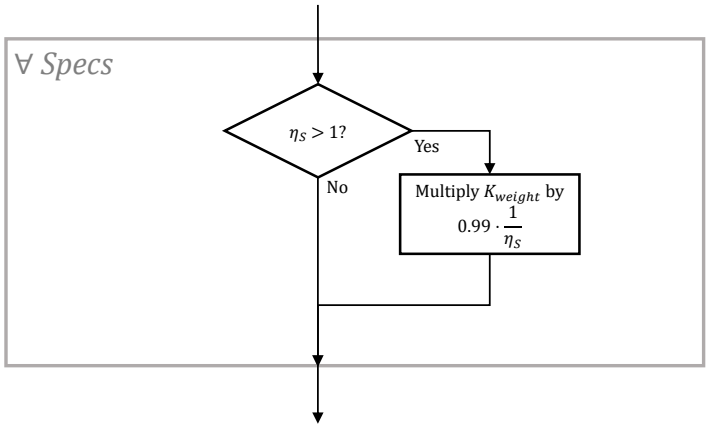
Fig. 12 Flowchart of the specification update rules.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.



(a) Tighten Specs



(b) Relax Specs

Fig. 13 Flowchart describing iterable specification update rules.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

expected to conflict with one another. For instance, lidar-based controllers tend to make heavy use of the elevator to pitch into gusts, but this tends to generate additional HTP and fuselage loads.

4.1.2 Control requirements

The corresponding control requirements are listed in Table 1. Inner wing loads are specified via two load stations: the wing root ($WL, 76$) and an inner-mid-wing station slightly outboard of the engine nacelles ($WL, 85$). Fuselage loads are specified at a load station just ahead of the HTP ($FU, 35$), and HTP loads are specified only at the root ($HL, 60$). The load requirements are also visually represented as black arrows in Figure 15. Ride quality requirements are evaluated at the pilot station and at the aft end of the cabin; they are defined in terms of the peak response to discrete gusts and of the ISO 2631-1 motion sickness metric. Actuator constraints are only specified for their deflection rates, as the deflections themselves rarely reach their limits.

Table 1 Control requirements

Goal	Reqt. ID	Output	Description	Requirement Type	Performance Target
Soft Requirements					
1)	S1	$M_{x,WL,76}$	Wing root bending moment	CS 25.341(a) Discrete Gusts	≤ Maneuver load envelope
	S2	$M_{x,WL,85}$	Inner mid-wing bending moment		
2)	S3	$M_{y,WL,76}$	Wing root torsional moment	CS 25.341(a) Discrete Gusts	≤ Open-loop gust load envelope
	S4	$M_{y,WL,85}$	Inner mid-wing torsional moment		
3)	S5	$M_{y,FU,35}$	Aft fuselage bending moment	CS 25.341(a) Discrete Gusts	≤ Open-loop gust load envelope
4)	S6	$M_{x,HL,60}$	HTP root bending moment	CS 25.341(a) Discrete Gusts	≤ Open-loop gust load envelope
	S7	$M_{y,HL,60}$	HTP root torsional moment		
5)	S8	$a_{z,fus,fwd}$	Vertical accel. at pilot station.	CS 25.341(a) Discrete Gusts	≤ Open-loop peak
	S9	$a_{z,fus,aft}$	Vertical accel. at aft cabin.		
	S10	$a_{z,fus,fwd}$	Vertical accel. at pilot station.	ISO 2631-1 Motion Sickness	≤ Open-loop performance
	S11	$a_{z,fus,aft}$	Vertical accel. at aft cabin.		
Hard Requirements					
6)	H1	$\delta\dot{a}^{out}_{sym}$	Symmetrical aileron deflection rates	CS 25.341(a) Discrete Gusts	≤ 40°/s
	H2	$\delta\dot{a}^{in}_{sym}$			
	H3	$\delta\dot{a}^{flp}_{sym}$			
	H4	$\delta\dot{e}$	Elevator deflection rate		
7)	H5	$\delta\dot{a}^{out}_{sym}$	Symmetrical aileron deflection rates	H_∞ Bandwidth (Eq. 9)	$K_{weight} = 20$, $n_{ro} = 1$, $\omega_{ro} = 0.05 \times 2\pi$ rad/s, $n_{bw} = 2$, $\omega_{bw} = 10 \times 2\pi$ rad/s
	H6	$\delta\dot{a}^{in}_{sym}$			
	H7	$\delta\dot{a}^{flp}_{sym}$			
	H8	$\delta\dot{e}$	Elevator deflection rate		

4.1.3 Control function parameters

A pure feedforward controller using only lidar-derived wind estimates and consisting of a simple gain matrix is selected. The estimated wind field $\mathbf{w}_{est}$ has 71 elements, with $h_{pre} = 45$ and $h_{post} = 25$, and the sensor output vector $\mathbf{y}_{sens}$ is empty (see Fig. 6). The gain matrix thus has 71 columns and 4 rows, one for each set of symmetrically deflected control surfaces. System time delays are represented by a discrete 50 ms delay (i.e., $5z^{-1}$) on all control inputs $\mathbf{u}$.

4.2 Design example

A multi-model control design run is performed using all four mass cases simultaneously. Figure 14 shows the progression of the requirement performance η_R over the course of the design process. Figures 14a and 14b present a detailed view of η_R for all control requirements at the beginning and end of the

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

design process. Note that iteration 0 represents the initial conditions; in this case, all control gains start at 0, so Fig. 14a represents the open-loop performance and requirements H1-H8 have $\eta_R = 0$ because the actuators are not used. If the initial values of the control gains were nonzero, the initial performance would be affected.

In Fig. 14c, the worst-performing requirement for each iteration is traced by the black line, whereas all other requirements are plotted in light gray. It is clear that for this design run, the first synthesis results in excessive actuator rates, and the successive 10 iterations are spent gradually restricting them.

Figure 15 plots the load envelopes for all loads specified in the requirements. Control requirements are indicated by black arrows, the open-loop gust load envelopes are in blue, the maneuver envelopes in red, and the responses of the linear models corresponding to the design points in purple. This controller meets and exceeds all load requirements. In fact, the loads are reduced almost everywhere except for the aft fuselage, where they are slightly increased with respect to the open loop. This includes, surprisingly, the HTP loads, which are reduced despite the use of the elevator.

Figure 16a shows the wing root bending moment frequency response and Fig. 16b its time-domain response to a 70 m gust. This controller mainly reduces the loads around the short-period mode, and above 3 Hz, it is essentially inactive. From Fig. 16b, we see that it is taking advantage of the preview time provided by the lidar system to generate an opposite load before the gust impact (between 0.5 and 0.8 s).

4.3 Consistency of design process

A certain degree of randomness is inherent to the control design process, as the synthesis algorithm makes some use of random numbers (e.g., through random restarts when it fails to reach the design objective). The iteration rules are entirely deterministic, but the requirement performance obtained at the end of each iteration and the local optimum of an iteration's initial controller can vary significantly. This then affects which specifications are added, their initial gains, and whether (or how many times) the controller parameters are randomized, potentially causing each run to take a very different path. A tight set of requirements and/or a controller structure which strongly favors a particular control strategy always tend to improve the consistency of the outcome. To give some indication of the consistency of the design process, the design process is repeated 10 times, and the worst-case η_R curves for each run are plotted.

Figure 17 shows this plot for the baseline design case. All runs had the same final result, i.e., success; in terms of the number of iterations before stopping, the initial design is more or less in the middle of the pack. As in the initial design, most iterations are spent enforcing the constraints, as the critical gust length may shift from one iteration to the next.

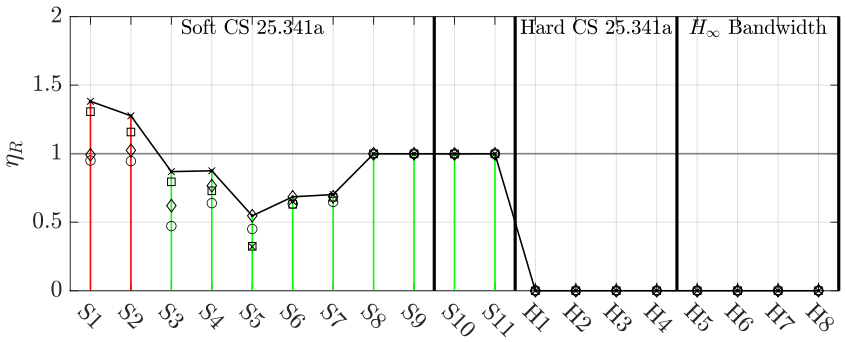
To help improve the consistency of the process and reduce the number of iterations, continuous-turbulence requirements are added to the control surface rates, described in Table 2. The resulting noniterable H_2 specifications should help constrain the actuator rates regardless of gust length, hopefully speeding up the process.

Table 2 Additional continuous-turbulence requirements

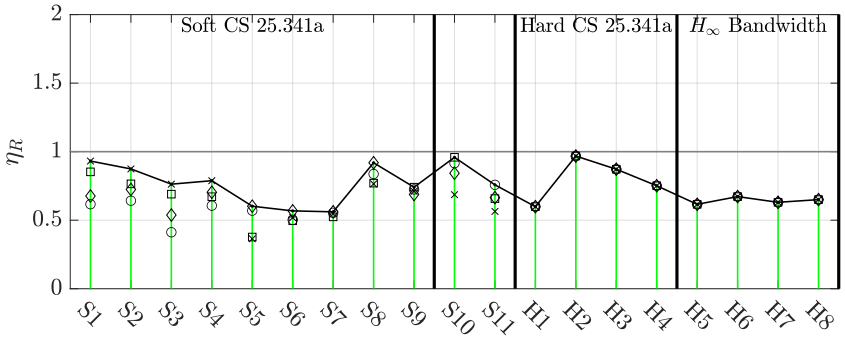
Goal	Reqt. ID	Output	Description	Requirement Type	Performance Target
Hard Requirements					
6)	H9 H10 H11	$\delta\dot{a}^{out}_{sym}$ $\delta\dot{a}^{in}_{sym}$ $\delta\dot{a}^{flp}_{sym}$	Symmetrical aileron deflection rates	CS 25.341(b) Continuous Turbulence	≤ 40°/s
	H12	$\delta\dot{e}$	Elevator deflection rate		

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

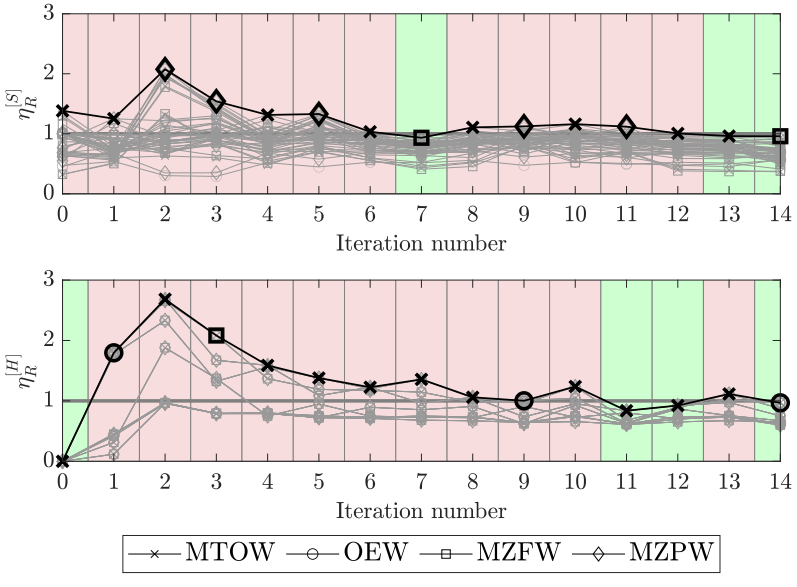
The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.



(a) Initial requirement performance η_R.



(b) Final requirement performance η_R.

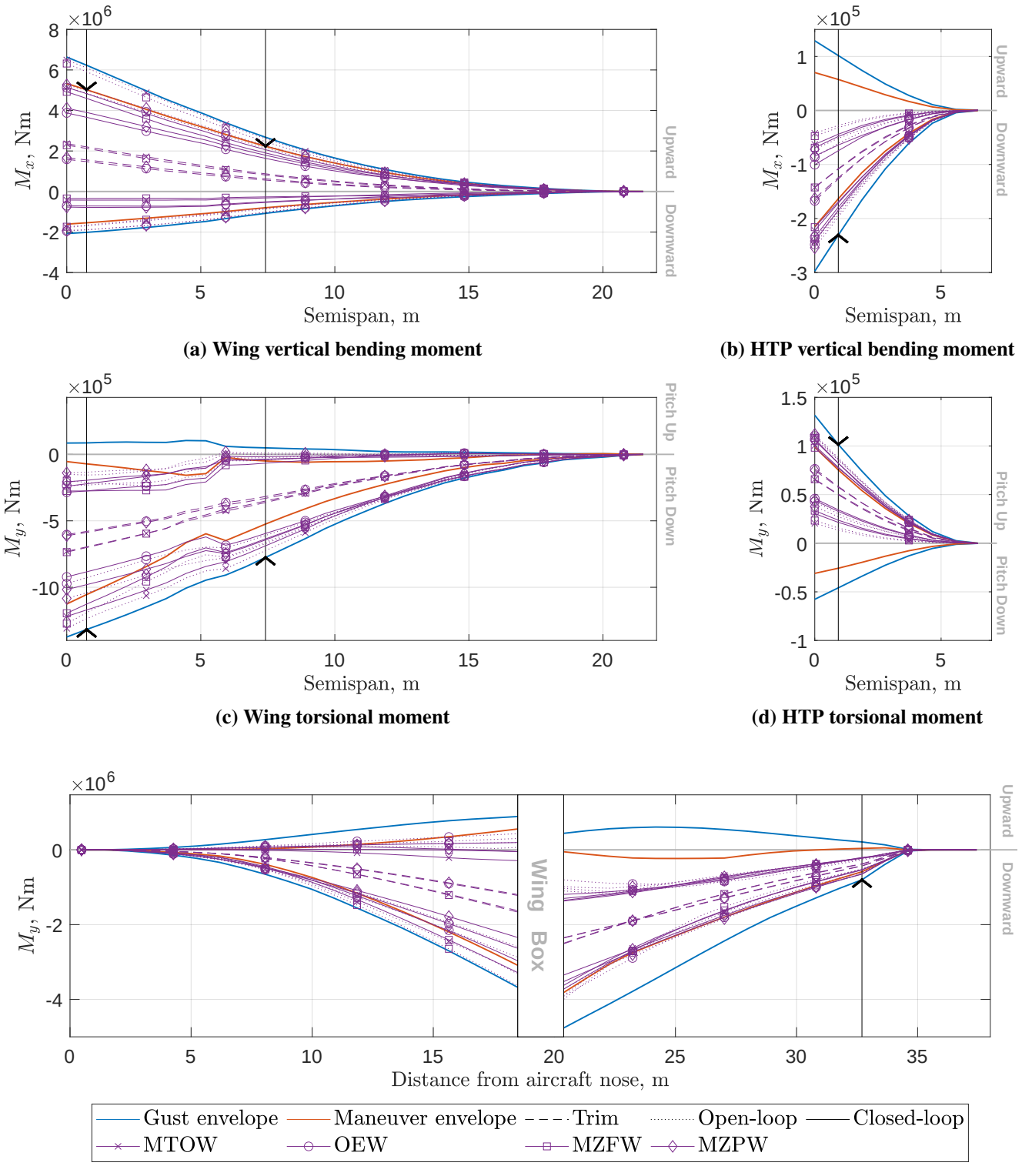


(c) Evolution of the soft (upper) and hard (lower) requirement performance η_R over the course of the design process.

Fig. 14 Control design progress of the lidar-based FF GLA multi-model design case. Mass cases (indicated by markers) include: MTOW (×), OEW (○), MZFW (□), and MZPW (◇)

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.



(a) Wing vertical bending moment

(b) HTP vertical bending moment

(c) Wing torsional moment

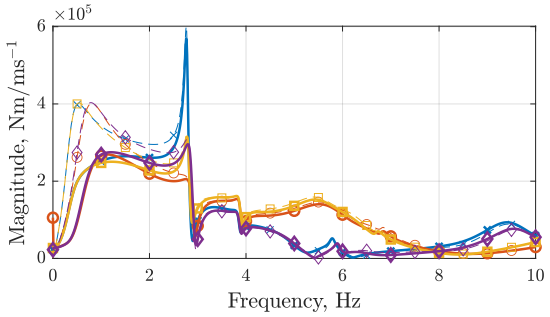
(d) HTP torsional moment

(e) Fuselage bending moment

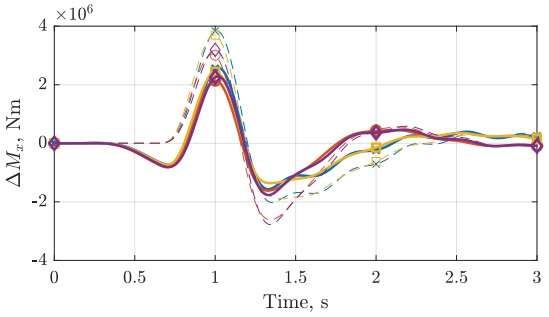
Fig. 15 Open-loop gust and maneuver load envelopes compared to the gust load envelopes computed for the lidar-based FFGLA controller. Control requirements indicated by black arrows.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.



(a) Frequency response



(b) Time series of incremental response to discrete gust with $H = 70$ m.

Fig. 16 Response of wing root bending moment ($M_{x,WL,76}$) in open-loop (dashed lines) and with lidar-based FF GLA controller (continuous lines). Mass cases include: MTOW (blue, ×), OEW (red, ○), MZFW (yellow, □), and MZPW (purple, ◇)

The results are shown in Fig. 18. As anticipated, the additional constraints reduce the average number of iterations before the stopping conditions are met, and the variations in η_R are generally more contained.

5 Conclusion and Outlook

An automated multiobjective gust load alleviation (GLA) control design process based on nonsmooth optimization-based frequency-domain control synthesis tools has been presented. It is capable of iterating on the control specifications to obtain a controller capable of meeting time-domain gust load objectives in the presence of conflicting constraints. It has been successfully demonstrated here on a reasonably complex gust load alleviation control problem for a high-order aeroelastic aircraft.

The randomness and uncertainty inherent in the synthesis method (e.g., due to random restarts) prevent the control process from consistently and deterministically converging to the same result in the same number of iterations, but rerunning the design process several times appears to always produce the same outcome (i.e., success or failure). These shortcomings of nonsmooth optimization-based control synthesis tools are largely outweighed by their practical advantages, e.g., structured controllers, multichannel specifications, and the ability to deal with high-order control problems. The issue with random restarts may be addressed by developing a (potentially deterministic) scheme for selecting the restarted parameter values instead of allowing *systune* to do so on its own.

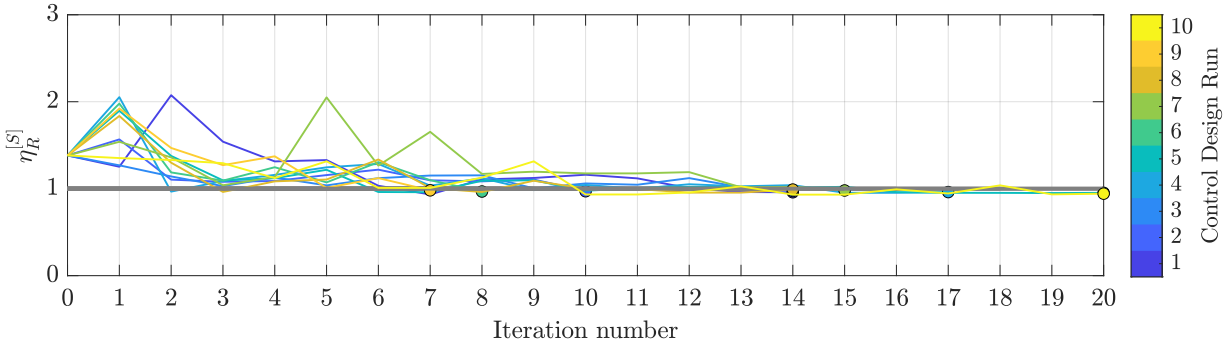
Finally, for the case in which control authority exceeds what is needed to meet the requirements, the control design process cannot prevent the controller from (wastefully) overperforming on its objectives. It is currently up to the designer to manually add secondary requirements (e.g., passenger comfort) or tighten the constraints. A future development could add 'open' iterable requirements which become active only once all objectives and constraints are met. They would then be gradually tightened until the system reaches its limits.

Acknowledgments

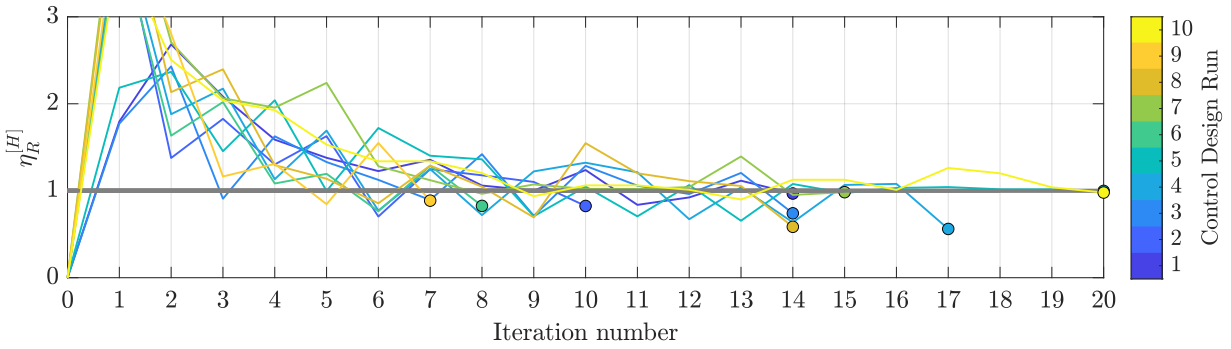
The authors would like to acknowledge the funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2163/1 - Sustainable and Energy Efficient Aviation – Project-ID 390881007.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

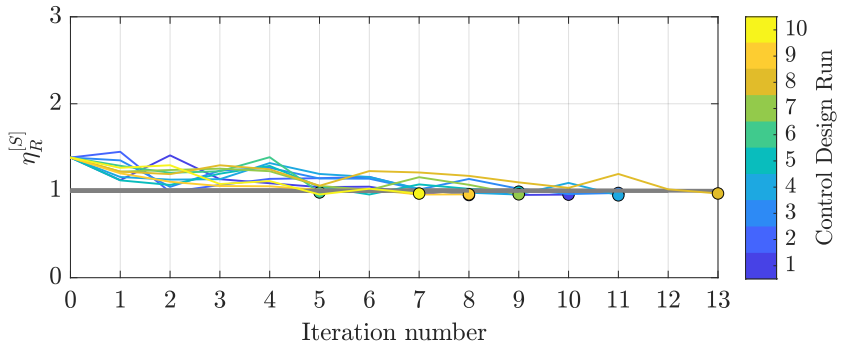


(a) Soft requirements

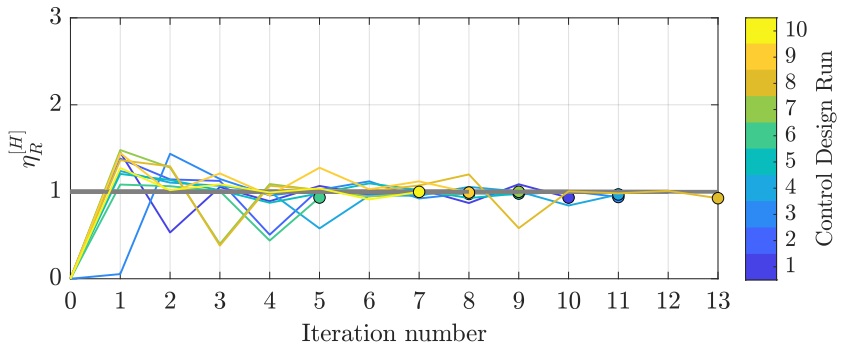


(b) Hard requirements

Fig. 17 Evolution of maximum requirement performance η_R over 10 control design runs. Individual runs are indicated by color.



(a) Soft requirements



(b) Hard requirements

Fig. 18 Evolution of maximum requirement performance η_R over 10 control design runs, including continuous-turbulence requirements on actuator rates. Individual runs are indicated by color.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

References

[1] EASA. *CS-25: Certification Specifications and Acceptable Means of Compliance for Large Aeroplanes*, volume Amendment 27. European Union Aviation Safety Agency, Cologne, Germany, Jan. 2023.

[2] FAA. CFR Title 14 Vol 1 Part 25 - Airworthiness Standards: Transport Category Airplanes, 2020.

[3] Hugo Fournier, Paolo Massioni, Minh Tu Pham, Laurent Bako, Robin Vernay, and Michele Colombo. Robust Gust Load Alleviation of Flexible Aircraft Equipped with Lidar. *Journal of Guidance, Control, and Dynamics*, pages 1–15, Sept. 2021. ISSN: 1533-3884. DOI: 10.2514/1.G006084.

[4] Davide Cavaliere, Nicolas Fezans, Daniel Kiehn, David Quero, and Patrick Vrancken. Gust Load Control Design Challenge Including Lidar Wind Measurements and Based on the Common Research Model. In *AIAA SCITECH 2022 Forum*, San Diego, CA, USA & Virtual, Jan. 2022. American Institute of Aeronautics and Astronautics. ISBN: 978-1-62410-631-6. DOI: 10.2514/6.2022-1934.

[5] Natascha Jansson and David Eller. Robust Turbulence Load Alleviation. In *IFASD 2011*, Paris, France, June 2011.

[6] Simon Hecker and Klaus-Uwe Hahn. Advanced Gust Load Alleviation System for Large Flexible Aircraft. In *1st CEAS Conference*, page 8, Berlin, Germany, 2007.

[7] Nicolas Fezans and Hans-Dieter Joos. Combined Feedback and LIDAR-Based Feedforward Active Load Alleviation. In *AIAA Atmospheric Flight Mechanics Conference*, Denver, Colorado, USA, June 2017. American Institute of Aeronautics and Astronautics. ISBN: 978-1-62410-493-0. DOI: 10.2514/6.2017-3548.

[8] A. Wildschek, T. Haniš, and F. Stroscher. L_∞-Optimal feedforward gust load alleviation design for a large blended wing body airliner. In C. Vallet, D. Choukroun, C. Philippe, G. Balas, A. Nebylov, and O. Yanova, editors, *Progress in Flight Dynamics, Guidance, Navigation, Control, Fault Detection, and Avionics*, pages 707–728, St. Petersburg, Russian, 2013. EDP Sciences. ISBN: 978-2-7598-0878-6 978-2-7598-1107-6. DOI: 10.1051/eucass/201306707.

[9] Andreas Wildschek, Rudolf Maier, Klaus-Uwe Hahn, Dirk Leissling, Michael Press, and Adolf Zach. Flight Test with an Adaptive Feed-Forward Controller for Alleviation of Turbulence Excited Wing Bending Vibrations. In *AIAA Guidance, Navigation, and Control Conference*, Chicago, Illinois, USA, Aug. 2009. American Institute of Aeronautics and Astronautics. ISBN: 978-1-60086-978-5. DOI: 10.2514/6.2009-6118.

[10] Yongzhi Wang, Andrea Da Ronch, and Maryam Ghandchi Tehrani. Adaptive Feedforward Control for Gust-Induced Aeroelastic Vibrations. *Aerospace*, 5(3):86, Aug. 2018. ISSN: 2226-4310. DOI: 10.3390/aerospace5030086.

[11] Matthew Dillsaver, Carlos Cesnik, and Ilya Kolmanovsky. Gust Load Alleviation Control for Very Flexible Aircraft. In *AIAA Atmospheric Flight Mechanics Conference*, Portland, Oregon, Aug. 2011. American Institute of Aeronautics and Astronautics. ISBN: 978-1-62410-153-3. DOI: 10.2514/6.2011-6368.

[12] Matthias Wuestenhagen. Gust Load Alleviation Control of Aircraft with Varying Mass Distribution. In *AIAA SCITECH 2023 Forum*, National Harbor, MD & Online, Jan. 2023. American Institute of Aeronautics and Astronautics. ISBN: 978-1-62410-699-6. DOI: 10.2514/6.2023-0371.

[13] John H. Hansen, Molong Duan, Ilya V. Kolmanovsky, and Carlos E. S. Cesnik. Load Alleviation of Flexible Aircraft by Dynamic Control Allocation. *Journal of Guidance, Control, and Dynamics*, pages 1–9, June 2022. ISSN: 0731-5090, 1533-3884. DOI: 10.2514/1.G006577.

[14] Sohrab Haghighat, Hugh H. T. Liu, and Joaquim R. R. A. Martins. Model-Predictive Gust Load Alleviation Controller for a Highly Flexible Aircraft. *Journal of Guidance, Control, and Dynamics*, 35(6):1751–1766, Nov. 2012. ISSN: 0731-5090, 1533-3884. DOI: 10.2514/1.57013.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

[15] H.-G. Giesseler, M. Kopf, P. Varutti, T. Faulwasser, and R. Findeisen. Model Predictive Control for Gust Load Alleviation. In *IFAC Proceedings Volumes*, volume 45, pages 27–32, 2012. DOI: 10.3182/20120823-5-NL-3013.00049.

[16] Michael Kopf, Eric Bullinger, Hans-Gerd Giesseler, Stephan Adden, and Rolf Findeisen. Model Predictive Control for Aircraft Load Alleviation: Opportunities and Challenges. In *2018 Annual American Control Conference (ACC)*, pages 2417–2424, Milwaukee, WI, June 2018. IEEE. ISBN: 978-1-5386-5428-6. DOI: 10.23919/ACC.2018.8430956.

[17] X. Wang, E. Van Kampen, Q. P. Chu, and Roeland De Breuker. Flexible Aircraft Gust Load Alleviation with Incremental Nonlinear Dynamic Inversion. *Journal of Guidance, Control, and Dynamics*, 42(7):1519–1536, July 2019. ISSN: 0731-5090, 1533-3884. DOI: 10.2514/1.G003980.

[18] Simon Binder, Andreas Wildschek, and Roeland De Breuker. The interaction between active aeroelastic control and structural tailoring in aeroservoelastic wing design. *Aerospace Science and Technology*, 110:106516, Mar. 2021. ISSN: 12709638. DOI: 10.1016/j.ast.2021.106516.

[19] Jia Xu and Ilan Kroo. Aircraft Design with Active Load Alleviation and Natural Laminar Flow. *Journal of Aircraft*, 51(5):1532–1545, Sept. 2014. ISSN: 0021-8669, 1533-3868. DOI: 10.2514/1.C032402.

[20] Vega Handojo, Jan Himisch, Kjell Bramsiepe, Wolf Reiner Krüger, and Lorenz Tichy. Potential Estimation of Load Alleviation and Future Technologies in Reducing Aircraft Structural Mass. *Aerospace*, 9(8):412, July 2022. ISSN: 2226-4310. DOI: 10.3390/aerospace9080412.

[21] Mark B. Tischler, Tom Berger, Christina M. Ivler, Mohammadreza H. Mansur, Kenny K. Cheung, and Jonathan Y. Soong. *Practical Methods for Aircraft and Rotorcraft Flight Control Design: An Optimization-Based Approach*. American Institute of Aeronautics and Astronautics, Inc., Reston, Virginia, Apr. 2017. ISBN: 978-1-62410-443-5 978-1-62410-661-3. DOI: 10.2514/4.104435.

[22] H.-D. Joos. A methodology for multi-objective design assessment and flight control synthesis tuning. *Aerospace Science and Technology*, 3(3):161–176, Apr. 1999. ISSN: 12709638. DOI: 10.1016/S1270-9638(99)80040-6.

[23] Nicolas Fezans, Christian Wallace, Daniel Kiehn, Davide Cavaliere, and Patrick Vrancken. Lidar-based Gust Load Alleviation - Results Obtained on the Clean Sky 2 Load Alleviation Benchmark. In *IFASD 2022*, Madrid, Spain, June 2022. Paper No. 155.

[24] Davide Cavaliere and Nicolas Fezans. Toward Automated Gust Load Alleviation Control Design via Discrete Gust Impulse Filters. *Journal of Guidance, Control, and Dynamics*, pages 1–14, Feb. 2024. ISSN: 0731-5090, 1533-3884. DOI: 10.2514/1.G007762.

[25] Pierre Apkarian and Dominikus Noll. Optimization-Based Control Design Techniques and Tools. In John Baillieul and Tariq Samad, editors, *Encyclopedia of Systems and Control*. Springer London, London, 2021. ISBN: 978-1-4471-5102-9. DOI: 10.1007/978-1-4471-5102-9.

[26] Frederic M. Hoblit. *Gust Loads on Aircraft: Concepts and Applications*. AIAA Education Series. American Institute of Aeronautics and Astronautics, 1988. ISBN: 978-1-60086-188-8. DOI: 10.2514/4.861888.

[27] Stanislav Karpuk, Rolf Radespiel, and Ali Elham. Assessment of Future Airframe and Propulsion Technologies on Sustainability of Next-Generation Mid-Range Aircraft. *Aerospace*, 9(5):279, May 2022. ISSN: 2226-4310. DOI: 10.3390/aerospace9050279.

[28] Yannic Beyer, Davide Cavaliere, Kjell Bramsiepe, Khalid Khalil, André Bauknecht, Nicolas Fezans, Meiko Steen, and Peter Hecker. An Aeroelastic Flight Dynamics Model for Gust Load Alleviation of Energy-Efficient Passenger Airplanes. In *AIAA AVIATION 2023 Forum*, San Diego, CA and Online, June 2023. American Institute of Aeronautics and Astronautics. ISBN: 978-1-62410-704-7. DOI: 10.2514/6.2023-4452.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.

[29] Christian Reschke. *Integrated Flight Loads Modelling and Analysis for Flexible Transport Aircraft*. PhD thesis, Universität Stuttgart, Stuttgart, Germany, 2006.

[30] NRC. Aviation Safety and Pilot Control: Understanding and Preventing Unfavorable Pilot-Vehicle Interactions. NRC, National Research Council, Washington, D.C., 1997.

[31] Davide Cavaliere, Nicolas Fezans, Daniel Kiehn, Julius Schultz, and Ulrich Römer. Linear Modeling of Doppler Wind Lidar Systems for Gust Load Alleviation Design. *Journal of Guidance, Control, and Dynamics (under review)*, 2024.

[32] Nicolas Fezans, Patrick Vrancken, Philippe Linsmayer, Christian Wallace, and Christoph Deiler. Designing and Maturating Doppler Lidar Sensors for Gust Load Alleviation: Progress Made Since AWIATOR. In *1st Aerospace Europe Conference*, page 12, Bordeaux, France, 2020.

[33] Daniel Kiehn, Nicolas Fezans, Patrick S Vrancken, and Christoph Deiler. Parameter Analysis of a Doppler Lidar Sensor for Gust Detection and Load Alleviation. In *IFASD 2022*, Madrid, Spain, June 2022.

[34] Ahmed Khalil and Nicolas Fezans. A multi-channel H_∞ preview control approach to load alleviation design for flexible aircraft. *CEAS Aeronautical Journal*, 12(2):401–412, Apr. 2021. ISSN: 1869-5582, 1869-5590. DOI: 10.1007/s13272-021-00503-z.

[35] Ahmed Khalil and Nicolas Fezans. Gust load alleviation for flexible aircraft using discrete-time preview control. *The Aeronautical Journal*, 125(1284):341–364, Feb. 2021. ISSN: 0001-9240, 2059-6464. DOI: 10.1017/aer.2020.85.

[36] Christian Wallace and Nicolas Fezans. Lidar-based Gust Load Alleviation - Results Obtained on a Generic Long Range Aircraft Configuration. In *EUCASS*, page 23 pages. Proceedings of the Aerospace Europe Conference - EUCASS - CEAS - 2023, 2023. DOI: 10.13009/EUCASS2023-564.

[37] Sigurd Skogestad and Ian Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. Wiley, Chichester, 2. ed., repr edition, 2010. ISBN: 978-0-470-01168-3.

[38] Neal M. Barr, Dagfinn Gangsaas, and Dwight R Schaeffer. Wind Models for Flight Simulator Certification of Landing and Approach Guidance and Control Systems. FAA FAA-RD-74-206, Federal Aviation Administration, Washington, D.C., Dec. 1974.

[39] François Kubica and Béatrice Madelaine. Passenger Comfort Improvement by Integrated Control Law Design. In *RTO AVT Specialists' Meeting on Structural Aspects of Flexible Aircraft Control*, Ottawa, Canada, Oct. 1999.

[40] L. Zuo and S.A. Nayfeh. Low order continuous-time filters for approximation of the ISO 2631-1 human vibration sensitivity weightings. *Journal of Sound and Vibration*, 265(2):459–465, Aug. 2003. ISSN: 0022460X. DOI: 10.1016/S0022-460X(02)01567-5.

[41] Christian Wallace, Simon Schulz, Nicolas Fezans, Thiemo Kier, and Guido Weber. Evaluation Environment for Cascaded and Partly Decentralized Multi-Rate Load Alleviation Controllers. In *ICAS 2022*, Stockholm, Sweden, Sept. 2022.

Except where otherwise noted, content of this paper is licensed under a Creative Commons Attribution 4.0 International License.

The reproduction and distribution with attribution of the entire paper or of individual pages, in electronic or printed form, including some materials under non-CC-BY 4.0 licenses is hereby granted by the respective copyright owners.