Waitman, S., & Marcos, A. (2019). H Control Design for Active Flutter Suppression of Flexible-Wing Unmanned Aerial Vehicle Demonstrator. *Journal of Guidance, Control, and Dynamics*. https://doi.org/10.2514/1.G004618

JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS

H_∞ Control Design for Active Flutter Suppression of Flexible-Wing Unmanned Aerial Vehicle Demonstrator

Sérgio Waitman* and Andrés Marcos†
University of Bristol, Bristol, England BS8 1TR, United Kingdom

https://doi.org/10.2514/1.G004618

Flutter is an aeroelastic phenomenon affecting flexible structures in a fluid flow and may lead to unstable oscillations and to critical structural damage. This paper concerns the design of a controller to provide active flutter suppression on an unmanned flexible-wing demonstrator, currently under construction. The aircraft models used in the control design are obtained using a combination of balanced and modal reduction techniques from a high-fidelity nonlinear aeroservoelastic model, and include information about actuator and sensor dynamics as well as the phase loss introduced by the flight control computer. The controller is synthesized by posing and solving a weighted H_∞-norm optimization problem with the goal to provide damping for the flutter modes and extend the flight envelope above the open-loop flutter speed. The paper concludes with a thorough analysis and verification of the performance achieved by the closed-loop system, including simulations with the high-fidelity nonlinear model of the actual maneuvers that will be performed by the unmanned demonstrator during future flight tests.

Nomenclature

$a_{z,\mathrm{AS}}$ = antisymmetric wingtip vertical acceleration measurement
$a_{z,\mathrm{cg}}$ = vertical acceleration at the center of gravity
$a_{z,\mathrm{S}}$ = symmetric wingtip vertical acceleration measurement
$a_{z,\mathrm{wL}}$, $a_{z,\mathrm{wR}}$ = vertical acceleration at the left and right wingtips
G_{act} = model of the direct drive servo actuator dynamics
G_{chain} = aggregate model of the parasitic dynamics
G_{delay} = model of the computational delay
G_{IMU} = model of the inertial measurement unit sensor dynamics
G_N = Nth synthesis model, with $N \in \{1 \ldots, 26\}$
$G_{\mathrm{Padé}}$ = Padé approximation of the parasitic dynamics
G_{red} = reduced-order model of the parasitic dynamics
K_N = H_∞ controller designed for model N, with $N \in \{15, 19, 22\}$
$K_{r,N}$ = reduced H_∞ controller for model N, with $N \in \{15, 19, 22\}$
K_{sch} = scheduled flutter controller
$T_{L/R\to S/AS}$ = sensor coordinate transformation from left/right to symmetric/antisymmetric
$T_{S/AS\to L/R}$ = actuator coordinate transformation from symmetric/antisymmetric to left/right
V_{IAS} = indicated airspeed
V_{TAS} = true airspeed
δ_{AS} = antisymmetric flutter control input
δ_a = allocated aileron input
δ_e = allocated elevator input
δ_{iL}, δ_{iR} = left and right wing flaps, with $i \in \{1, \ldots 4\}$
δ_r = allocated rudder input
δ_{riL}, δ_{riR} = left and right ruddervators, with $i \in \{1, 2\}$
δ_S = symmetric flutter control input
δ_T = thrust input
η_1, η_2 = first generalized symmetric and antisymmetric flutter modes

I. Introduction

AEROSERVOELASTIC flutter is a phenomenon associated with the interaction of aerodynamic, inertial, and structural forces. It is characterized by the coupling between the aerodynamic loads acting on a flexible structure and its natural modes of vibration. This results in a self-excited unstable oscillatory motion that may lead to structural failure ([1] Chap. 5). Flutter could have catastrophic consequences in wings and airfoils and, as such, is critical for aircraft design. For this reason, airworthiness authorities require that aircraft are free from flutter across the flight envelope [2,3].

Traditionally, aircraft are designed so that flutter does not occur within normal flight conditions. This is normally achieved passively, that is, through the addition of structural mass to provide stiffness. This approach has the drawback of increasing the aircraft's weight and thus also its fuel consumption. The use of active flutter suppression (AFS) techniques could potentially lead to flight envelope expansion without the addition of stiffening structures, thus increasing fuel efficiency and performance. In addition to the economic benefits, this is aligned with the current trend of environmental impact reduction, such as the EU initiative to reduce greenhouse gas emissions by 20% and increase energy efficiency by 20% by 2020 [4].

The interest in the application of AFS in aircraft is not new, and a thorough historical review is provided in [5]. The advances in AFS come from the desire to reduce aircraft weight, which leads to a reduced separation between rigid-body and flexible modes. Because of this, AFS is connected with other active control technologies, such as gust alleviation and deformation control [6]. Early attempts to achieve AFS were mostly based on knowledge of the physics behind the flutter mechanisms. The design engineer would make use of this knowledge to choose an appropriate actuator/sensor positioning across the wing span to allow for simpler control strategies. The so-called *identically located actuators and forces* (ILAF) approach belongs to this category, where the sensor (accelerometer) is placed at the force application point, which allows the controller to provide structural damping (see, e.g., [7]). Such approaches have the advantage of using simple independent Single-Input Single-Output controllers, which could be designed using root-locus or other classic frequency-domain design techniques, but also the main drawbacks of potentially not appropriately addressing the Multiple-Input Multiple-Output nature of the problem and/or limiting the possible optimization of the performance versus robustness tradeoff. As computational power increased and more advanced control techniques were developed, attention was shifted toward new multivariable and robust approaches, such as linear quadratic Gaussian (LQG) [8], H_∞ [9,10],

---

Presented as Paper 2019 at the AIAA SciTech Forum, San Diego, CA, 7–11 January 2019; received 29 May 2019; revision received 12 November 2019; accepted for publication 14 November 2019; published online 19 December 2019.

*Research Associate, Aerospace Engineering Department; sergio.waitman@bristol.ac.uk.

†Senior Lecturer, Aerospace Engineering Department; andres.marcos@bristol.ac.uk. Senior Member AIAA.

and linear parameter-varying (LPV) [11] control. Seminal research efforts have seen these techniques applied to unmanned aerial vehicle (UAV) demonstrators, leading to important flight-test validation milestones. The X-56A Multi-Utility Technology Testbed (MUTT), designed by Lockheed Martin under contract to the Air Force Research Laboratory (AFRL), is a high-aspect-ratio flying wing developed with the goal of furthering research on the coupling between rigid and flexible modes as well as AFS [12]. This project has inspired the Performance Adaptive Aeroelastic Wing (PAAW) research effort by NASA, under which similar flying-wing UAVs were developed to test and validate AFS [13]. Another important experience, but with highly flexible UAVs and not directly focused on the AFS problem, was reported under the X-HALE project [14]. For more detail on the history of developments on AFS, the reader is referred to [5] and references therein. Even though considerable research effort has been devoted to the development of AFS, Livne ranks "control law design and implementation methods" as one of the five areas of importance for future research on AFS methods and adds that "it would [. . .] be an important contribution to the state of the art from the certification needs perspective to invest in the development and testing of such methods and architectures" [5].

The work reported in this paper is part of the Flutter Free Flight Envelope eXpansion for ecOnomical Performance improvement (FLEXOP) project, funded by the European Commission under the Horizon 2020 Research and Innovation Programme (Grant 636307). Its aim is to develop multidisciplinary aircraft design capabilities by achieving a closer coupling of wing aeroelasticity and flight control systems in the design process. A flexible-wing remotely piloted demonstrator is designed in the framework of this project [15,16] and will be used to achieve flight-test validation of aeroelastically tailored design [17] and AFS [18,19]. A concept drawing of the demonstrator is presented in Fig. 1. The overall goal of the project is to allow maturation of the technology readiness level (TRL) of passive and active strategies for flutter mitigation, thus helping to push them into more widespread use.

This paper concerns the design of an AFS controller with the goal to provide active damping and attenuate the flutter oscillations of the FLEXOP demonstrator. The controller is designed using classic H_∞ techniques ([20] Chap. 9) and scheduled according to the speed of the aircraft. This technique has been used before, in the frame of flutter suppression studies, in, for example, [9,21], including wind-tunnel validation, but they focus on AFS for wings and airfoils only. More recently, the H_∞ techniques have been applied to the design of AFS controllers for the X-56A model [11] as well as to the related Mini MUTT UAV [10]. The X-56A and the Mini MUTT are flying-wing UAVs and, as such, are subjected to a strong coupling between flexible and rigid modes, a phenomenon known as body freedom flutter (BFF) [5,22]. The FLEXOP demonstrator, on the other hand, has a more conventional configuration with a well-distinguishable fuselage and V-tail, which leads to flutter modes with a less pronounced coupling with the rigid dynamics. The main contribution of this paper is the application of well-established H_∞ control techniques to the problem of AFS for a relevant conventional aircraft system, thus advancing the TRL of the approach. The designed controller has successfully increased the flutter speed by around 30% in closed loop, as verified in simulation with high-fidelity linear and nonlinear models obtained from CFD/FEM.

Fig. 1 FLEXOP demonstrator.

This paper is divided as follows. Section II introduces the available models of the demonstrator and describes the model reduction performed on the high-fidelity models for the design of the controller. The synthesis framework of the H_∞ controller is described in Sec. III, and Sec. IV presents the analysis of the closed-loop system with the flutter controller. Finally, the verification of the performance of the closed-loop system with the addition of a baseline rigid-motion controller for handling and guidance is presented in Sec. IV.B.

II. FLEXOP Demonstrator Models

A. FLEXOP Demonstrator

The design of the FLEXOP UAV is reported in [23]. The flexible-wing demonstrator features a wingspan of 7 m, with an aspect ratio of 20. It is equipped with a 300 N jet engine, placed on the back of the fuselage, allowing a maximum takeoff mass of approximately 65 kg. An airbrake system is present, deflecting from the sides of the fuselage to allow fast deceleration for speed control and steep-angle approaches. Each wing features four control surfaces (see Fig. 2a) with the innermost used as high-lift devices for takeoff and landing. The two middle flaps are used as ailerons for roll motion control, and the outermost control surfaces are dedicated to flutter control. The empennage has a V-tail configuration to reduce interaction with the engine. Two pairs of ruddervators are present in the empennage for controlling pitch and yaw motions.

The demonstrator is equipped with a central inertial measurement unit (IMU) providing measurements of pitch, roll, and yaw angles (θ, ϕ, and ψ); pitch, roll, and yaw rates (p, q, and r); and forward, lateral, and vertical position (x, y, and z). An air data probe is used to measure airspeed (V_{IAS}), altitude (h), and angles of attack (α) and sideslip (β). Additionally, each wing is equipped with six additional IMUs providing acceleration and angular velocity measurements in different positions (see Fig. 2b).

Three sets of wings are planned to be developed for the demonstrator in the framework of the FLEXOP project. The first is a rigid wing to allow the validation of aircraft design, including its avionics and autopilot capabilities. A second set of flexible wings was designed such that flutter occurs below 55 m/s, with a low flutter frequency of less than 10 Hz. This is the set that will be used for verification and validation of the AFS technologies. Finally, a third set of wings was developed using aeroelastic tailoring to achieve maneuver load alleviation and for the validation of aeroelastic design frameworks [17].

For the purposes of this paper, only the set of flexible wings is considered. In the next sections the aircraft models used for control design are detailed.

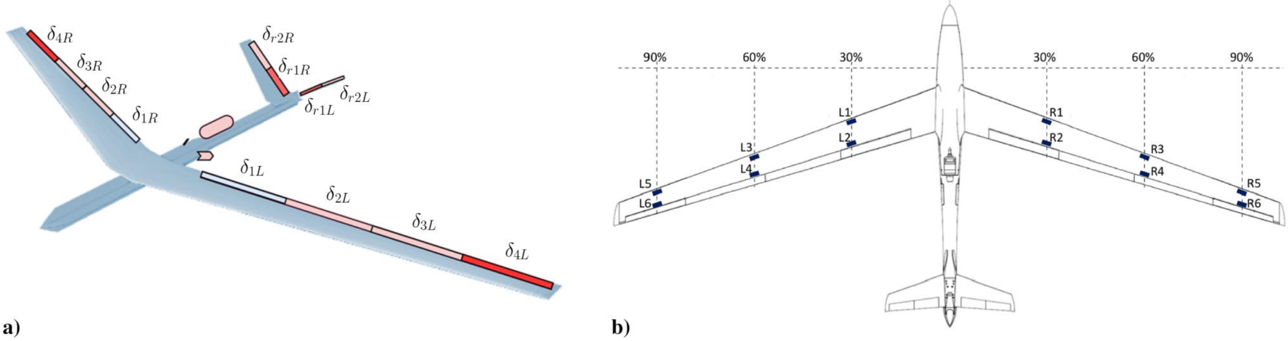


Fig. 2 FLEXOP demonstrator's actuators (a) and sensors (b) for flutter control.

B. Aeroelastic Aircraft Models

A fully flexible nonlinear model describing the rigid-body motion as well as the aeroelastic dynamics of the FLEXOP demonstrator was developed by the FLEXOP teams at the German Aerospace Center (DLR) and the Technical University of Munich (TUM) [24]. It is based on the interconnection between an aerodynamic model obtained by the doublet lattice method (DLM) and an elastic model representing the structural dynamics derived from a finite element (FE) model. This nonlinear model was subsequently trimmed and linearized in straight and level flight configuration at $N = 26$ different cruise speeds between 45 and 70 m/s (i.e., with 1 m/s increments), resulting in a set of 1152-state LTI models. The state space of these models contains the rigid-body states (forward, lateral, and vertical positions (x, y, and z); forward, lateral, and vertical velocities (u, v, and w); pitch, roll, and yaw angles (θ, ϕ, and ψ); and pitch, roll, and yaw rates (p, q, and r); 50 generalized modal coordinates ($\eta_1, \ldots, \eta_{50}$) and their derivatives ($\dot{\eta}_1, \ldots, \dot{\eta}_{50}$); as well as 1040 lag states ($x_{\mathrm{lag},1}, \ldots, x_{\mathrm{lag},1040}$). It should be noted that the trimming routine considers the static trim condition of the rigid and elastic modes, meaning that the linearization is performed around the deflected flight shape, and not around the jig shape (the interested reader is referred to [24] for further details). This is of importance when dealing with flexible wing configurations, as it can lead to an overestimation of the open-loop flutter speed of around 40%; see, for example, Ref. [25].

To obtain a tractable low-order model for controller synthesis, the Bristol University team applied model reduction techniques on these initial models ([20] Chap. 11). First, the states x, y, z, and ψ were removed by truncation. Subsequently, the 35 high-frequency modes ($\eta_{16}, \ldots, \eta_{50}$) and their respective derivatives were residualized, resulting in 1078-state models. These were subsequently converted to a balanced realization, and 1034 states were residualized to yield a set of 44-state models. Finally, the models were converted to a modal form, and an additional four high-frequency modes were residualized, resulting in a set of 40-state reduced-order models. This final set is used for controller synthesis purposes. The poles of the full- and reduced-order models are shown in Fig. 3 across speeds. Figure 4 shows a zoom of the pole map of the full-order (crosses) and reduced-order (squares) models close to the imaginary axis around the 50 rad/s frequency. It can be seen that, as the velocity increases, the damping of two pairs of conjugate poles deteriorates until instability is reached at $N = 8$. This point corresponds to a speed of $V_{\mathrm{TAS}} = 52$ m/s, and thus the open-loop flutter speed (i.e., the speed at which the flutter mode poles cross the imaginary axis) is very close to 52 m/s. These poles correspond to the first symmetric and antisymmetric flutter modes (η_1 and η_2, respectively). A comparison between the frequency responses of the two models is shown in Fig. 5 for a speed of $V_{\mathrm{TAS}} = 59$ m/s. It can be seen that the reduced model is a good approximation of the aircraft dynamics below 200 rad/s. It is important that it gives a good representation of the flutter dynamics, around 50 rad/s, as this is the model that will be used for active flutter control design.

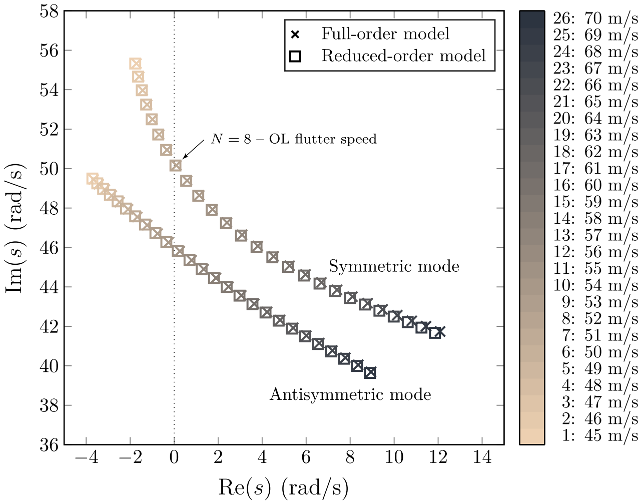


Fig. 4 Poles of the full-order and reduced-order models around the first two flutter modes.

The separation between the rigid-body dynamics and the elastic modes of the reduced-order model is highlighted in the pole map in Fig. 6. The short period and dutch roll modes, respectively, from the longitudinal and lateral dynamics, are shown to be in the frequency range below 15 rad/s, sufficiently away from the symmetric and antisymmetric flutter modes. One of the elastic modes is seen to be close to the 20 rad/s mark at lower speeds, but this is a stable mode associated with the first symmetric wing bending mode. As the airspeed increases, this mode moves toward a higher-frequency region.

C. Actuators and Sensors Models

Schematics of the aircraft control surfaces are seen in Fig. 2a. The outboard flaps (δ_{4L} and δ_{4R}) are reserved for flutter control, whereas the midboard flaps (δ_{2L}, δ_{3L}, δ_{2R}, and δ_{3R}) and ruddervators (δ_{r1L}, δ_{r2L}, δ_{r1R}, and δ_{r2R}) are used by the baseline rigid-motion controller. This separation eases the parallel synthesis of both the baseline and the flutter controllers and minimizes the risk of saturation due to concurrent action during maneuvers. Because the effect of both symmetric and antisymmetric flutter modes must be countered, the control action is split into symmetric (δ_S) and antisymmetric (δ_{AS}) inputs, which are allocated to δ_{4L} and δ_{4R} according to the relations

$$\begin{bmatrix}\delta_{4L}\\ \delta_{4R}\end{bmatrix} = \begin{bmatrix}\frac{1}{2} & \frac{1}{2}\\ \frac{1}{2} & -\frac{1}{2}\end{bmatrix}\begin{bmatrix}\delta_S\\ \delta_{\mathrm{AS}}\end{bmatrix} =: T_{S/AS\to L/R}\begin{bmatrix}\delta_S\\ \delta_{\mathrm{AS}}\end{bmatrix} \qquad (1)$$

The outboard flaps δ_{4L} and δ_{4R} are controlled via a direct drive mechanism. This allows a fast and precise actuation by the flutter controller while limiting the effect of mechanical freeplay and

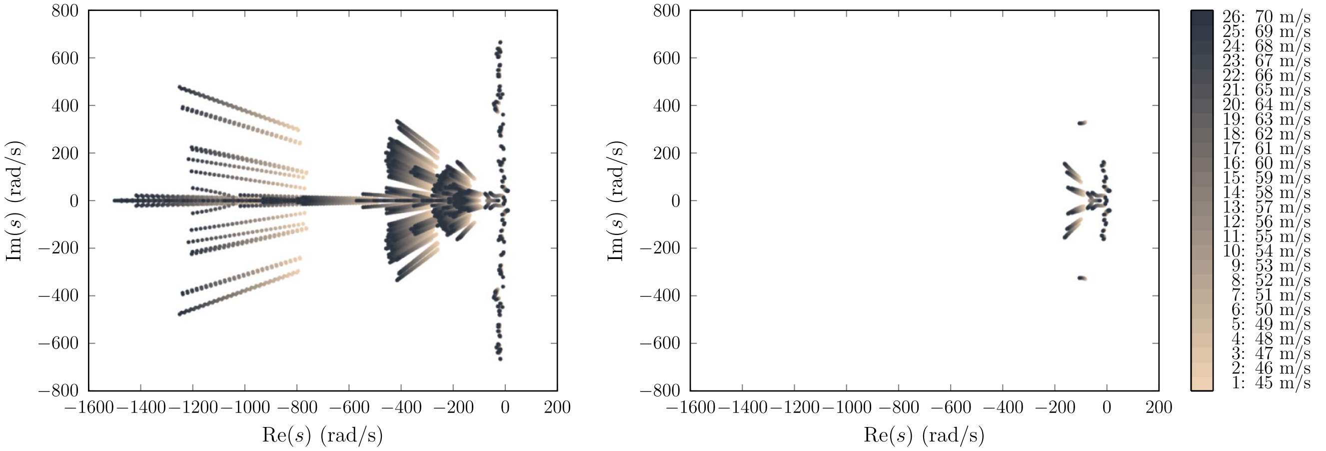


Fig. 3 Pole map of the full-order (left) and reduced-order (right) models.

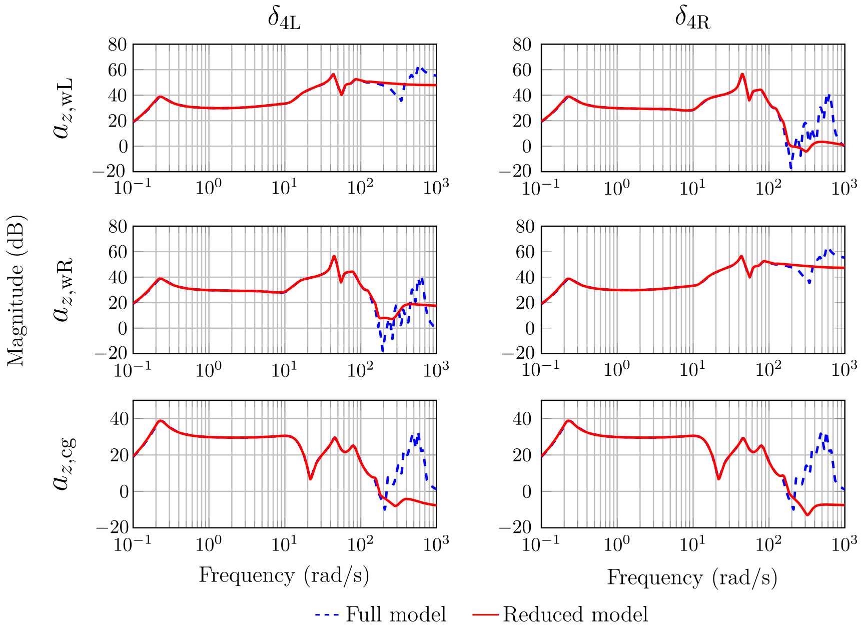


Fig. 5 Frequency responses of the full-order and reduced-order models at $V_{\mathrm{TAS}} = 59\ \mathrm{m/s}$.

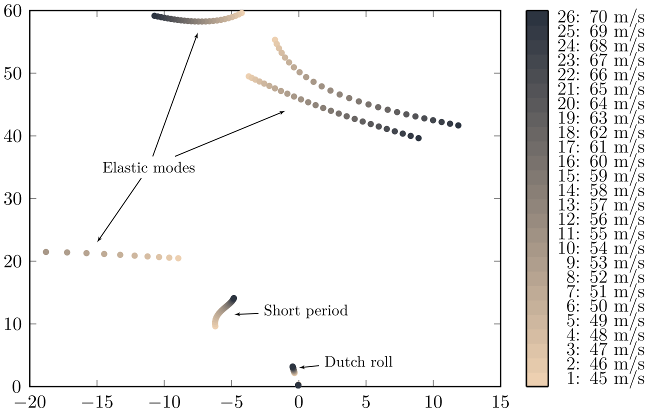


Fig. 6 Zoom on the low-frequency modes of the reduced-order model.

hysteresis associated with reduction mechanisms. A fourth-order model for this actuator was identified in [18], and is given by

$$G_{\mathrm{act}}(s) = e^{-0.0001s} \times \frac{2.741 \cdot 10^5 s^2 + 3.117 \cdot 10^6 s + 1.024 \cdot 10^8}{s^4 + 575.8 s^3 + 2.814 \cdot 10^5 s^2 + 3.324 \cdot 10^6 s + 1.025 \cdot 10^8} \tag{2}$$

The specifications of the control surfaces for flutter suppression are shown in Table 1. The minimal and maximal deflections are limited to −15° and 20°, respectively, with a maximum deflection rate estimated to be above 1000°/s. The model of the aircraft takes into account not only the flap deflection but also the deflection rate and acceleration. To take this into account, the first and second derivatives of the output of the actuator model are also used, that is, $sG_{\mathrm{act}}(s)$ and $s^2G_{\mathrm{act}}(s)$. This is possible because the actuator model has a relative degree of 2.

Table 1 Actuator limitations

Specification	δ_{4L}, δ_{4R}
Min. deflection	−15°
Max. deflection	20°
Max. deflection rate	> 1000°/s

As aforementioned, each wing of the demonstrator is equipped with 6 IMUs; see also Fig. 2b. The outboard sensors ($L5$, $L6$, $R5$, and $R6$), together with the IMU on the center of gravity, are used to provide measurements for the flutter controller. The IMUs used on the FLEXOP demonstrator have a bandwidth of 200 Hz and are modeled as simple unitary-gain first-order systems:

$$G_{\mathrm{IMU}}(s) = \frac{2\pi 200}{s + 2\pi 200} \tag{3}$$

The measured outputs used by the controller are the center of gravity acceleration $a_{z,\mathrm{cg}}$ and the left and right wingtip accelerations, $a_{z,\mathrm{wL}}$ and $a_{z,\mathrm{wR}}$, respectively. The latter accelerations are defined as the mean between the measurements provided by sensors 5 and 6 for each wing. To clearly separate the action of both flutter modes, the measurements are split into symmetric and antisymmetric accelerations, $a_{z,S}$ and $a_{z,\mathrm{AS}}$, defined as

$$\begin{bmatrix} a_{z,S} \\ a_{z,\mathrm{AS}} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & -1 \\ \frac{1}{2} & -\frac{1}{2} & 0 \end{bmatrix} \begin{bmatrix} a_{z,\mathrm{wL}} \\ a_{z,\mathrm{wR}} \\ a_{z,\mathrm{cg}} \end{bmatrix} =: T_{L/R \to S/AS} \begin{bmatrix} a_{z,\mathrm{wL}} \\ a_{z,\mathrm{wR}} \\ a_{z,\mathrm{cg}} \end{bmatrix} \tag{4}$$

The center of gravity acceleration is subtracted from the symmetric acceleration in order to decouple the measured symmetric acceleration at the wingtips from the rigid-body acceleration of the aircraft, as suggested in the seminal work by Wykes [7], as well as in more recent research efforts [26,27]. As a positive side effect, this guarantees that the gravity offset of the acceleration measurements is naturally canceled.

The flight computer of the FLEXOP demonstrator is responsible for the execution of the control laws, which are fed to a servo controller piloting the direct drive mechanism. The overall delay introduced by these elements is estimated to be around 15 ms, which is modeled as a pure delay $G_{\mathrm{delay}}(s) = e^{-0.015s} I_2$ acting on the controller output.

The complete feedback chain used for flutter suppression from the acceleration measurements ($a_{z,\mathrm{wL}}$, $a_{z,\mathrm{wR}}$, and $a_{z,\mathrm{cg}}$) to the flutter-dedicated control surfaces (δ_{4L}, δ_{4R}, and respective rates and accelerations) is presented in Fig. 7.

III. Controller Synthesis

In this section, the synthesis of a controller to achieve AFS is presented. The main goals of the controller are to provide additional

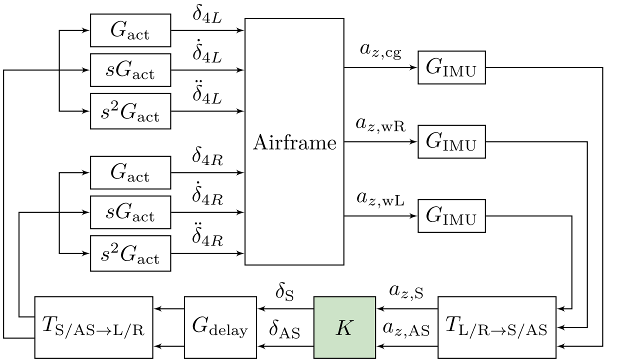


Fig. 7 Closed-loop system for flutter control.

damping to both the first symmetric and antisymmetric flutter modes below the open-loop flutter speed, and to extend the flight envelope by pushing the closed-loop flutter speed above 52 m/s. When in cruise level-flight at this speed, the aircraft is trimmed at about 22% of its full throttle, which indicates that there is margin to achieve this flight configuration from an aerodynamic point of view. With this objective in mind, the controller design is cast as an H_∞-norm problem, and suboptimal controllers are synthesized at three cruise speeds. These are subsequently scheduled with respect to the aircraft's true airspeed to obtain the final controller for AFS.

The designed flutter controllers will be implemented in the flight control computer, which operates at a sample time of 5 ms. This means that a discrete-time controller must be obtained. A choice must be made between discretizing the model and performing the design in discrete time, or synthesizing the controllers in continuous time and performing an a posteriori discretization. Although the former has the advantage of directly providing a discrete-time controller, the discretization of the plant and parasitic dynamics leads to some insight into the problem being lost. For this reason, the latter approach was chosen, and a Tustin transformation using the aforementioned sample time is performed to convert the controller from continuous to discrete time.

A. Synthesis Model

The controller is synthesized using the 40-state reduced-order model at cruise speeds of 59 m/s ($N = 15$), 63 m/s ($N = 19$), and 66 m/s ($N = 22$), all above the open-loop flutter speed. This is done to maximize the efficiency of the controller in this speed range and thus the flight envelope extension. The strategy for selecting the design points is based on the analysis of the performance achieved by each single-point controller, that is, the corresponding closed-loop flutter speed. Based on the performance of the controller at $N = 15$, the next point at $N = 19$ is selected such that the designed controller is able to stabilize a range of airspeeds overlapping with that of the first one and extending over higher speeds while maintaining performance. The same strategy is used to select the design point at $N = 22$. In these configurations, the eigenfrequencies of the unstable symmetric flutter mode are located between 43.9 and 45.7 rad/s, and those of the antisymmetric mode are between 41.7 and 44.1 rad/s.

As it was discussed at the beginning of this section, the goal of the controller is to damp the flutter oscillations. To translate this specification into the H_∞ optimization problem, the derivatives of the generalized coordinates corresponding to the first two flutter modes, $\dot{\eta}_1$ and $\dot{\eta}_2$, are chosen as performance measures. This results in a transfer matrix having resonance peaks at both of the flutter modes' eigenfrequencies (see Fig. 8). The damping objective can then be translated as a desired upper bound on these resonance peaks.

For the sake of controller synthesis, the effects of the deflection rate and acceleration of the control surfaces on the aircraft dynamics are ignored. In H_∞ control, the designed controller will have as many states as the synthesis model, and thus the dynamics of the IMU sensors are shifted to the input of the actuator models in order to reduce the plant size. This is possible because the model is linear and the bandwidth on every channel ($a_{z,\text{wL}}$, $a_{z,\text{wR}}$, and $a_{z,\text{cg}}$) is assumed to be the same. A reduced-order model of the combined effects of the delay with the IMU and actuator dynamics, G_{red}, is then obtained (see Fig. 9). For this, a fifth-order Padé approximation ($G_{\text{Padé}}$) of the sensor/delay/actuator chain (G_{chain}) is obtained, and then reduced via balancing and residualization. The result is a third-order model, which closely resembles the original model around the flutter eigenfrequencies (see Fig. 10).

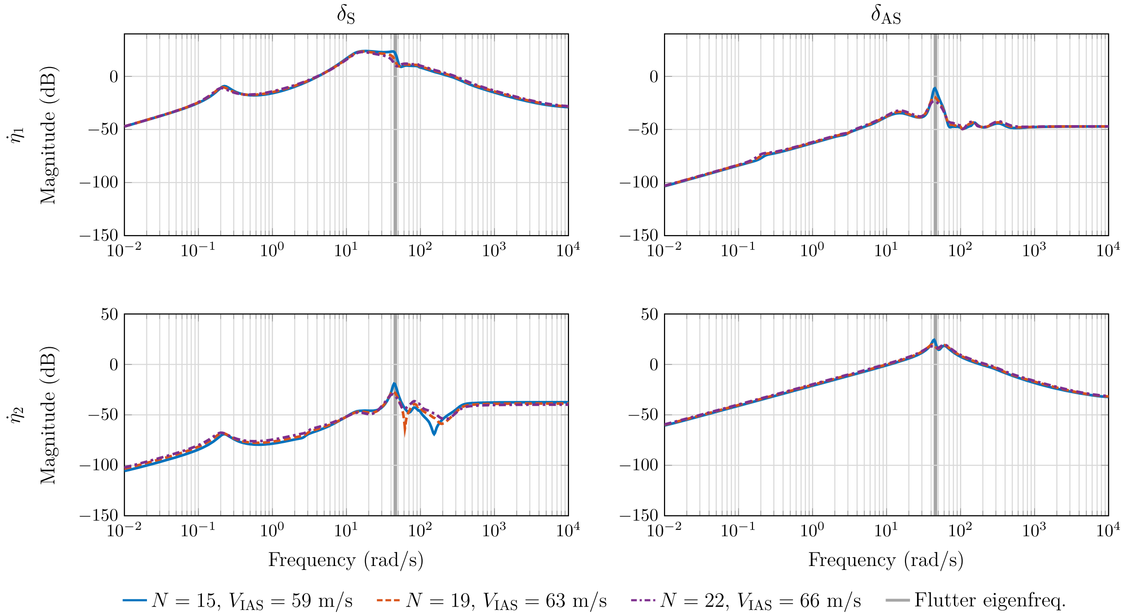


Fig. 8 Frequency response from the control inputs to $\dot{\eta}_1$ and $\dot{\eta}_2$.

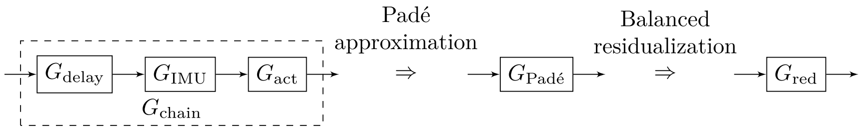


Fig. 9 Reduced model of the delay, IMU, and actuator dynamics.

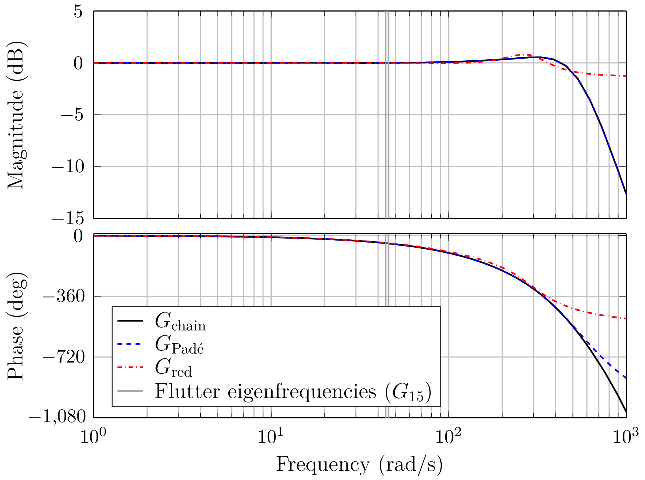


Fig. 10 Frequency response of actuator/sensor chain, Padé approximation, and reduced model.

The reduced-order model of the parasitic dynamics is appended to each of the 2 input channels in the 40-state model of the aircraft. Because the control action must be split to deal with both the symmetric and antisymmetric flutter modes, the allocation matrices $T_{L/R\to S/AS}$ and $T_{S/AS\to L/R}$ from Eqs. (1–4) are also included in the synthesis model $G_N(s)$, with $N \in \{15, 19, 22\}$. In this way, the controller sees $a_{z,S}$ and $a_{z,\text{AS}}$ as the measured outputs of the system and takes δ_S and δ_{AS} to be the controlled inputs (see Fig. 11).

B. H_∞ Loop Shaping

The design of H_∞ controllers requires the formulation of an H_∞ optimization problem whose solution, when found, is a controller that takes into account the system specifications. This section describes the formulation of the design problem and the numerical computation of the controller.

1. *Weighting Functions*

The performance specifications for the controller synthesis are translated into an H_∞ optimization problem via the addition of input and output frequency weights (see Fig. 12). There are four weights, W_z, W_u, W_n, and W_e, each focusing, respectively, on flutter performance (i.e., damping), control action, input signals scaling, and output signals scaling. The selection of the weights is explained next for the plant at 59 m/s, that is, G_{15}.

The damping provided by the controller for the unstable flutter modes is specified via the weight W_z. By increasing its magnitude around the flutter eigenfrequencies, a lower upper bound on the resonance peak of the respective transfer functions is imposed (as exemplified in Fig. 8), thus increasing the flutter damping. To reduce the number of states in the synthesis model, these weights are set to constant values, which are chosen based on the desired bound on the peaks of the corresponding transfer functions, leading to the final choice of $W_z = 0.01\ \text{diag}(5, 6)$.

The weight W_u is chosen so as to confine the control action around the flutter eigenfrequencies. This is done to avoid exciting high-frequency aeroelastic modes as well as limiting the coupling with the low-frequency dynamics that are handled by the baseline rigid-motion controller. The same weighting function for each control channel is then chosen and is given by

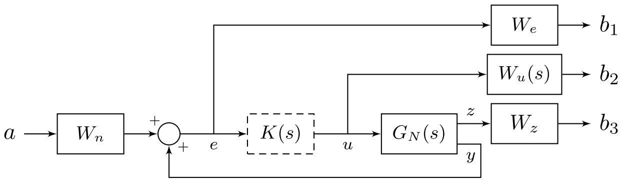


Fig. 12 Generalized plant for H_∞ optimization.

$$\tilde{W}_u(s) = 40\frac{s^2 + 101s + 2200}{s^2 + 127020s + 2200} \tag{5}$$

This transfer function is inspired by [10] and configures a band-stop filter centered around the eigenfrequency of the flutter modes, which imposes a wash-out and roll-off effect on the controller's frequency response, thus achieving the desired behavior.

The other weighting functions are chosen as constant matrices whose goal is to scale the input and output channels, typically based on normalizing the input–output relations, that is, by making every transfer function have unitary maximum magnitude. The values of the weights chosen after tuning for the synthesis at 59 m/s are

$$\begin{aligned} W_n &= 200I_2 \quad W_e = 0.001\text{diag}(6.7, 4.8) \\ W_u(s) &= \text{diag}(\tilde{W}_u(s), \tilde{W}_u(s)) \\ W_z &= 0.01\text{diag}(5, 6) \end{aligned} \tag{6}$$

The same weights are chosen for the synthesis at higher speeds (at 63 and 66 m/s), in order to obtain consistency across speeds. The only exception is W_e, which is changed to $W_e = 0.001\text{diag}(5.6, 10)$ for the synthesis at these two higher speeds. This was done because previous versions of these controllers, designed with the same weights as K_{15}, were not able to stabilize the plant in discrete time, due to a loss of damping for the antisymmetric channel arising from the controller discretization. The final W_e values were obtained after tuning, where the weight on the antisymmetric channel was increased with respect to its symmetric counterpart to increase the damping provided to the antisymmetric flutter mode, and the resulting controllers show similar adequate behavior in continuous and discrete time.

2. *Controller Synthesis and Reduction*

The generalized closed-loop system represented in Fig. 12 is given by the following transfer function matrix

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} W_e & & \\ & W_u & \\ & & W_z \end{bmatrix} \begin{bmatrix} S_o \\ KS_o \\ G_zKS_o \end{bmatrix} W_n a \tag{7}$$

where $S_o := (I - G_yK)^{-1}$ is the output sensitivity function and G_y (resp. G_z) denotes the transfer function from u to y (resp. to z). The transfer function from a to $b = [b_1 \quad b_2 \quad b_3]^T$ is then given by $\mathcal{F}_l(M, K)$, where $\mathcal{F}_l$ indicates a lower linear fractional transformation (LFT) ([28] Chap. 10) and M represents the system, augmented with the weights, in the standard H_∞ formulation (see Fig. 13). The synthesis model has 50 states, 40 coming from the reduced-order model of the aircraft, 6 from the reduced parasitic dynamics G_{red}, and additional 4 from the control effort weight W_u. The controller K is then found by solving the associated H_∞ optimization problem, that is,

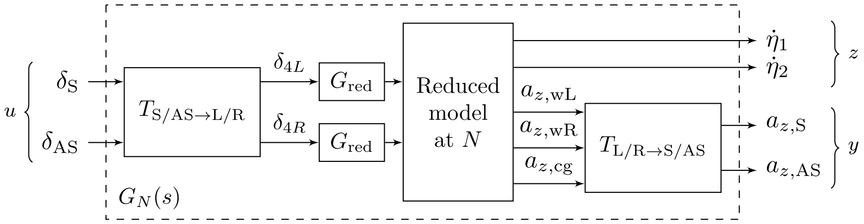


Fig. 11 Synthesis model for the H_∞ control design.

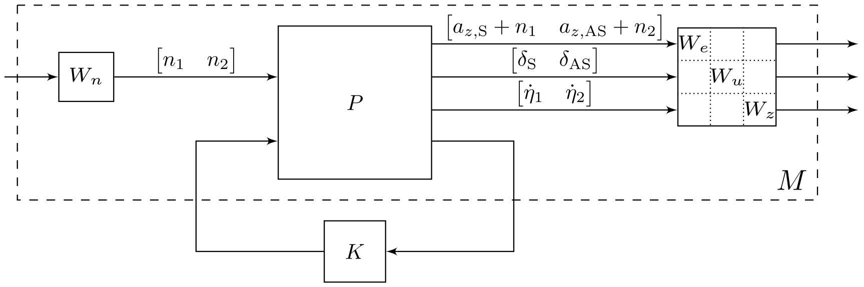


Fig. 13 Standard H_∞ problem.

$$K = \arg\min_K \|\mathcal{F}_l(M,K)\|_\infty \tag{8}$$

This is done numerically in MATLAB via the routine hinfsyn, and a suboptimal controller is synthesized for numerical stability. As an example of the results, the designed controller K_{15} ensures an H_∞ norm of 1.38 on the closed-loop system at 59 m/s.

As mentioned earlier, in H_∞ control the computed controller has the same number of states as the generalized plant, that is, 50 states. In view of the need for fast computation of the control law with limited processing power by the flight computer, a model reduction is performed by truncating a balanced representation of the controller. The dimension of the reduced controller is determined by analyzing the H_∞ norm of the weighted closed-loop system with the full-order controller replaced by its reduced version. Another important aspect is the frequency of the fastest pole of the controller, because the actual implementation of the control algorithm will be done in discrete time, with a sampling time of 5 ms. For this end, the reduced controllers $K_{r,i}$, with $i \in \{15, 19, 22\}$, are discretized with a Tustin transformation and frequency prewarping at the eigenfrequency of the first symmetric flutter mode. Based on this, the reduction is done so as to ensure that the poles of the reduced controller are slower than the Shannon frequency of 628 rad/s so that no dynamics of the controller are lost due to sampling or frequency warping. The order of the reduced controller is then chosen as to provide a compromise between its size, the frequency of its fastest pole, and the performance degradation. To further ensure proper roll-off of the controller, the direct feedthrough matrix D of the reduced controller is set to 0 after reduction. With this method, a 17-state reduced controller $K_{r,15}$ is obtained, which ensures an H_∞ norm of 1.46 on the weighted closed-loop system, a degradation of 5.96% on the performance obtained with the full-order controller K_{15}. A comparison between the frequency responses of both controllers is shown in Fig. 14a, together with the comparison at $N = 22$ in Fig. 14b. The frequency responses are quite similar around the flutter eigenfrequencies, where the controller action is most important. Although the frequency responses differ at low and high frequencies, the low-frequency degradation is still of sufficiently low magnitude to ensure that the flutter controller does not affect the rigid-motion baseline controller, as it will be shown in Sec. IV.

The pole-zero maps of both controllers, K_{15} and $K_{r,15}$, are shown in Fig. 15, with blue markers for the former and red for the latter. The highest-frequency pole of the full-order controller K_{15} is located at around 473 rad/s, whereas the fastest pole of $K_{r,15}$ is at 256 rad/s. The importance of the absence of high-frequency poles on the reduced controller is twofold. First, it ensures that the controller is not exciting the higher-frequency flexible modes of the aircraft. Second, because the flight computer has a sampling time of 5 ms, it is important to avoid the presence of poles above 630 rad/s. It can be seen from the pole-zero maps that the reduced controller has complex pairs of poles and zeros between 50 and 100 rad/s, which are very close to those of the full-order controller. These poles are the ones that are closer to the eigenfrequency of the open-loop flutter modes, located around 45 rad/s.

Table 2 shows the dimension of the three reduced controllers $K_{r,N}$, as well as the resulting performance degradation and their fastest poles. As mentioned earlier, there is a tradeoff between the size of the controller, the frequency of its fastest pole, and the performance degradation. This explains why the controllers have different dimensions, but with a consistent performance degradation in the cases of $K_{r,15}$ and $K_{r,19}$. In the case of $K_{r,22}$, it was not possible to obtain the same level of degradation without compromising the dimension of the controller, which is the reason why a degradation of 18% is observed in the chosen reduced controller.

The frequency responses and the pole-zero maps of the three reduced controllers $K_{r,N}$ are shown in Figs. 16 and 17. A clear difference can be spotted in the antisymmetric channel of the controller transfer functions between $K_{r,15}$ and the other reduced controllers. This is due to the change on the weight W_e between the synthesis of $K_{r,15}$ and the other controllers, as discussed in Sec. III.B.1. This leads to a sensible increase in the magnitude of the controller transfer function from $a_{z,\mathrm{AS}}$ to δ_AS for the two last

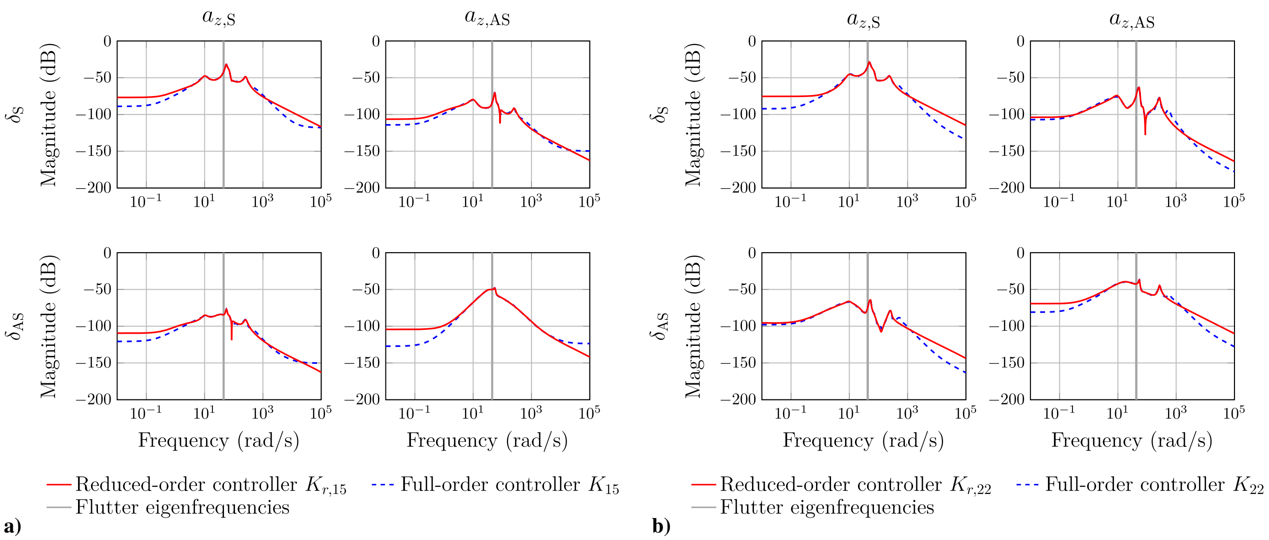


Fig. 14 Frequency responses of the full and reduced controllers at a) $N = 15$ and b) $N = 22$.

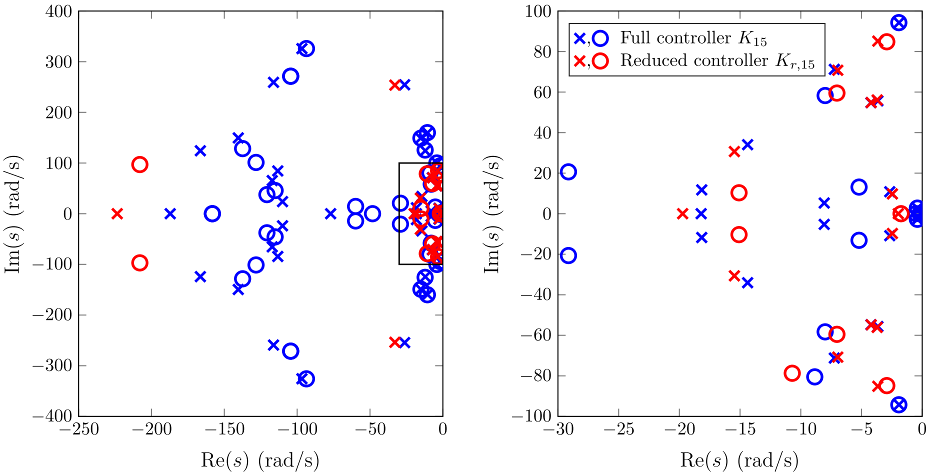


Fig. 15 **Pole-zero maps of the full-order and reduced-order controllers, with zoom on the right.**

controllers ($N \in \{19, 22\}$), which means that the controller is allowed more action in the antisymmetric channel at higher speeds. Moreover, additional peaks can be seen around the frequency of 270 rad/s, similar to the response of the symmetric channel ($a_{z,S}$ to δ_S).

C. Controller Scheduling

The single-point controllers synthesized in the previous section are scheduled to obtain a controller K_{sch} that provides better performance and robustness across the extended flight envelope. As discussed in the previous section, the controllers $K_{r,15}$, $K_{r,19}$, and $K_{r,22}$ are designed at, respectively, $V_{\text{TAS}} = \{59, 63, 66\}$ m/s using H_∞ design and subsequently reduced using balanced truncation, which leads to controllers of different dimensions. For this reason, an output interpolation scheme is selected for the scheduling of the controller. This has the additional advantage of avoiding numerical issues with the interpolation of the state-space matrices of the controllers [29]. Care must be taken so that the controller's outputs do not cancel each other during the interpolation phases; that is, that there is not a phase shift of 180° between them.

A piecewise-linear interpolation rule on the true airspeed V_{TAS} is chosen for simplicity. The scheduling rule is given by

$$\begin{aligned} K_{\text{sch}}(V_{\text{TAS}}) = {} & (1 - \lambda_1(V_{\text{TAS}}))(1 - \lambda_2(V_{\text{TAS}}))K_{r,15} \\ & + \lambda_1(V_{\text{TAS}})(1 - \lambda_2(V_{\text{TAS}}))K_{r,19} \\ & + \lambda_1(V_{\text{TAS}})\lambda_2(V_{\text{TAS}})K_{r,22} \end{aligned} \tag{9}$$

where

$$\lambda_1(V_{\text{TAS}}) \coloneqq \sigma\left(\frac{V_{\text{TAS}} - 59}{61 - 59}\right) \quad \lambda_2(V_{\text{TAS}}) \coloneqq \sigma\left(\frac{V_{\text{TAS}} - 64}{66 - 64}\right) \tag{10}$$

are the blending variables, with $\sigma(\cdot) \coloneqq \max(0, \min(1, \cdot))$ representing saturation between 0 and 1. The airspeed ranges where the controllers are blended ($V_{\text{TAS}} \in [59, 61]$ m/s for $K_{r,15}$ and $K_{r,19}$, and $V_{\text{TAS}} \in [64, 66]$ m/s for $K_{r,19}$ and $K_{r,22}$) are selected based on the ranges where the lower speed controller started to lose performance while the corresponding higher speed controller approached the performance achieved at its design speed. The scheduling strategy is represented in Fig. 18.

Table 2 **Number of states, degradation of performance, and frequency of the fastest pole of $K_{r,N}$**

N	Dimension	Degradation, %	Fastest pole, rad/s
15	17	5.96	256
19	16	3.99	259
22	15	18.1	269

IV. Closed-Loop Analysis of the Flexible Controller

In this section, the performance of the closed-loop system with the continuous-time scheduled controller K_{sch} is studied, first by itself and then together with the baseline rigid-motion controller designed by the Institute for Computer Science and Control (SZTAKI) and DLR teams [30]. All of the frequency- and time-domain analyses presented hereafter were performed with the continuous-time controller. The same analyses were also performed with the discrete-time version of the controller, obtained as described before in Sec. III.B.2, and very similar results were obtained (thus, the corresponding plots are omitted).

A. Without Baseline Rigid-Motion Controller

A thorough analysis of the closed-loop system with the scheduled controller K_{sch} is presented, first using the linearized models and then with the high-fidelity aeroelastic nonlinear model.

1. Linear Analysis

The first step is the analysis of the closed-loop system comprising the synthesis models G_N, for $N \in \{1, \ldots, 26\}$, with the scheduled controller. A pole-zero map is given in Fig. 19 for the models with $N \in \{1, 5, 10, 15, 20, 25\}$. The plot on the right shows the zoom around the flutter eigenfrequencies, with emphasis on the symmetric and antisymmetric flutter modes and without zeros for clarity. The controller stabilizes the reduced-order closed-loop system for the flying configurations from 45 m/s ($N = 1$) up to 68 m/s ($N = 24$). The non-minimum-phase zeros observed in the left plot are the result of the performed model reduction and are not present in the full-order models, as it will be seen next.

All of the subsequent analyses are performed with the full-order (1152 states) LTI aircraft models. The pole map of the closed-loop system (with the Padé approximation of the sensor/delay/actuator chain, $G_{\text{Padé}}$; see Fig. 9) is shown in Fig. 20, where the zeros are omitted for clarity. Once again, the right plot shows the zoom around the flutter eigenfrequencies. The scheduled controller is able to stabilize the flight configurations from $V_{\text{TAS}} = 45$ up to 68 m/s, effectively extending the flight envelope by 17 m/s, an increase of approximately 33% with respect to the open-loop flutter speed.

Figure 21 shows the maximum singular value from the inputs δ_S and δ_{AS} to the outputs $\dot{\eta}_1$ and $\dot{\eta}_2$ of the open-loop and closed-loop systems from $N = 1$ up to $N = 22$. It shows how the flutter controller damps the flutter modes without adding too much degradation in the low- and high-frequency dynamics.

To analyze the behavior of the closed-loop system in the time domain, the model represented in Fig. 22 is built. This model includes the aileron, elevator, and rudder commanded inputs, respectively, denoted δ_a, δ_e, and δ_r. The aileron action is allocated to the antisymmetric deflection of actuators δ_{2L}, δ_{3L} and δ_{2R}, δ_{3R}; that is,

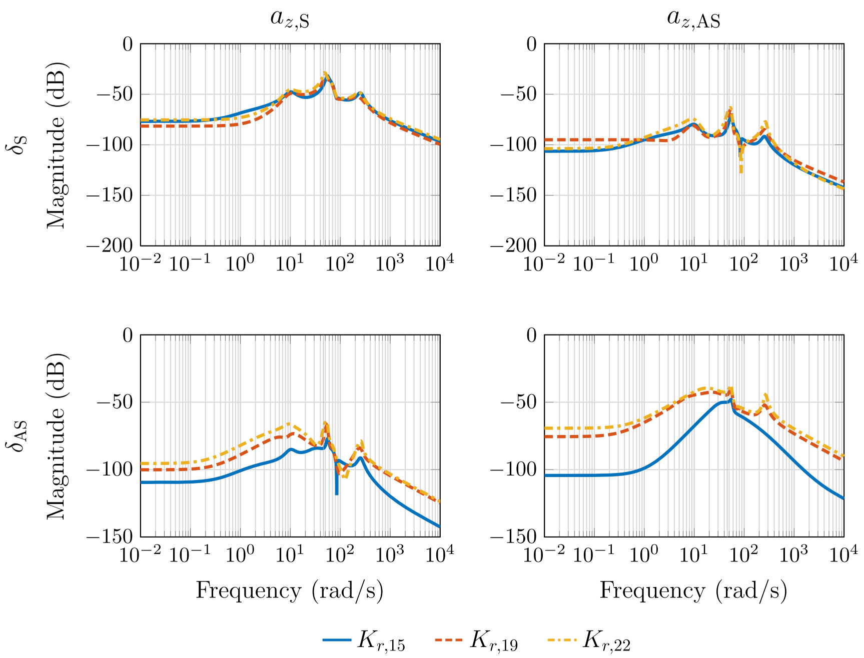


Fig. 16 Frequency responses of the reduced-order controllers $K_{r,N}$, for $N \in \{15, 19, 22\}$.

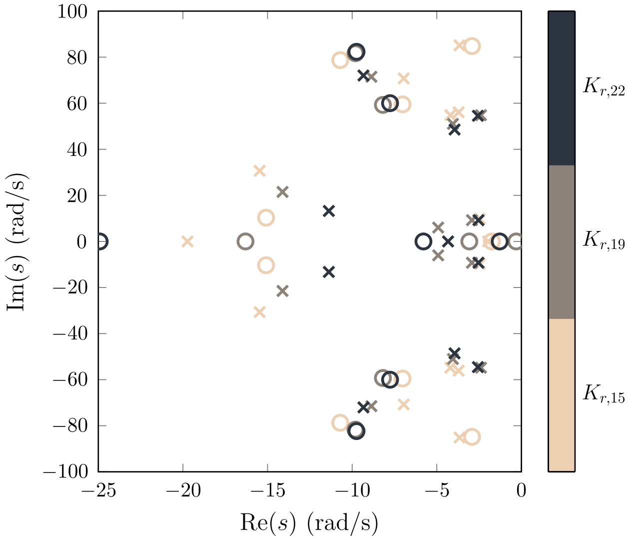


Fig. 17 Pole-zero maps of $K_{r,N}$, for $N \in \{15, 19, 22\}$.

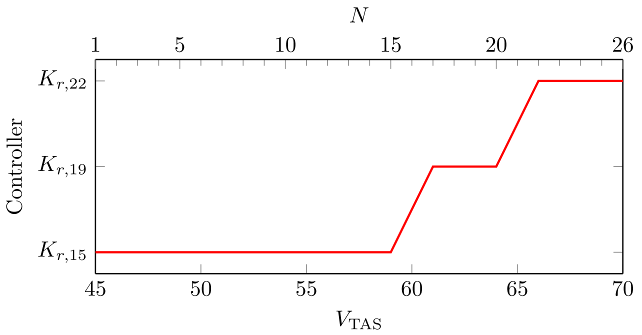


Fig. 18 Scheduling of K_sch on the aircraft true airspeed V_TAS.

$$\begin{bmatrix}\delta_{2L}\\ \delta_{3L}\\ \delta_{2R}\\ \delta_{3R}\end{bmatrix} = \frac{1}{4}\begin{bmatrix}1\\ 1\\ -1\\ -1\end{bmatrix}\delta_a \tag{11}$$

The elevator and rudder controls are allocated to the symmetric and antisymmetric deflection of the ruddervators δ_{r1L}, δ_{r2L} and δ_{r1R}, δ_{r2R} according to:

$$\begin{bmatrix}\delta_{r1L}\\ \delta_{r2L}\\ \delta_{r1R}\\ \delta_{r2R}\end{bmatrix} = \frac{1}{4}\begin{bmatrix}1 & 1\\ 1 & 1\\ -1 & 1\\ -1 & 1\end{bmatrix}\begin{bmatrix}\delta_r\\ \delta_e\end{bmatrix} \tag{12}$$

Figure 23 shows the response to a step input in the elevator δ_e of the open-loop (dashed lines) and closed-loop (solid lines) systems in cruise flight at the speeds of $V_\mathrm{TAS} = 46$ and $49\ \mathrm{m/s}$ (i.e., below the open-loop flutter speed). For $V_\mathrm{TAS} = 46\ \mathrm{m/s}$, the open- and closed-loop responses are very similar (almost indistinguishable in the plots), indicating that the flutter controller does not affect the maneuverability of the aircraft—that is, the closed-loop rigid-body dynamics remain virtually the same as the open-loop ones. This indicates that the flutter controller should not interact with the baseline rigid-body controller. For the case of $V_\mathrm{TAS} = 49\ \mathrm{m/s}$ it is seen (more easily on the bottom plots) that the flutter controller is able to damp the oscillations observed in the open loop. This is more clearly demonstrated in Fig. 24, which shows the same responses for $V_\mathrm{TAS} = 56\ \mathrm{m/s}$, that is, above the open-loop flutter speed. The simulation shows that the oscillations related to the symmetric flutter mode are effectively damped within 1 s. The antisymmetric flutter mode is also stabilized, although with smaller damping.

As a final linear time-simulation test, Fig. 25 shows the response of the closed-loop system to a doublet of $\pm 11°$ in the ailerons at the speed of $51\ \mathrm{m/s}$, just below the open-loop flutter speed, lasting for 1 s. As expected, this maneuver mainly excites the antisymmetric mode η_2 (see bottom-right plot). It is seen that no flutter oscillation is perceptible on the rigid-body dynamics (top plot) due to the action of the flutter controller. The deflections of the actuators δ_{4L} and δ_{4R} remain inside the saturation limits of $[-15°, +20°]$, with a maximum of $6.93°$. The maximum deflection rate is $239.57°/\mathrm{s}$, which is also below the saturation bound.

2. *Robustness Margins*

Robustness is an important property to be assessed in the closed-loop system as the design model is inherently not able to represent all of the actual dynamics and uncertainties of the physical aircraft. One of the more widespread indicator of robustness comes in the form of robustness margins, that is, the maximum allowed variation of gain and/or phase in the sensor and actuator channels of the control loop. Traditional gain and phase margins are obtained by breaking each channel of the feedback loop and finding the minimum *independent* gain and phase variation that will destabilize the closed loop. This approach alone can be inadequate as it does not account for correlated

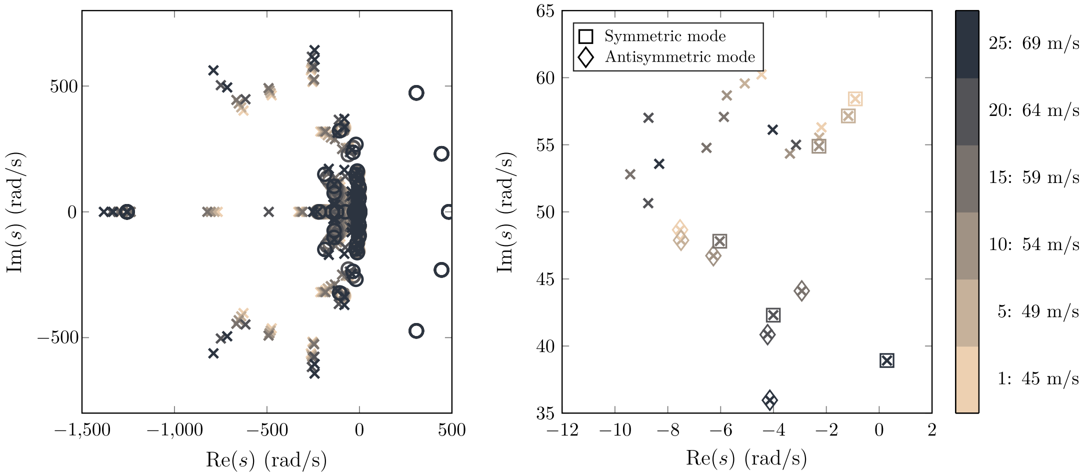


Fig. 19 **Pole-zero map of the reduced-order closed-loop system with zoom around the flutter eigenfrequencies.**

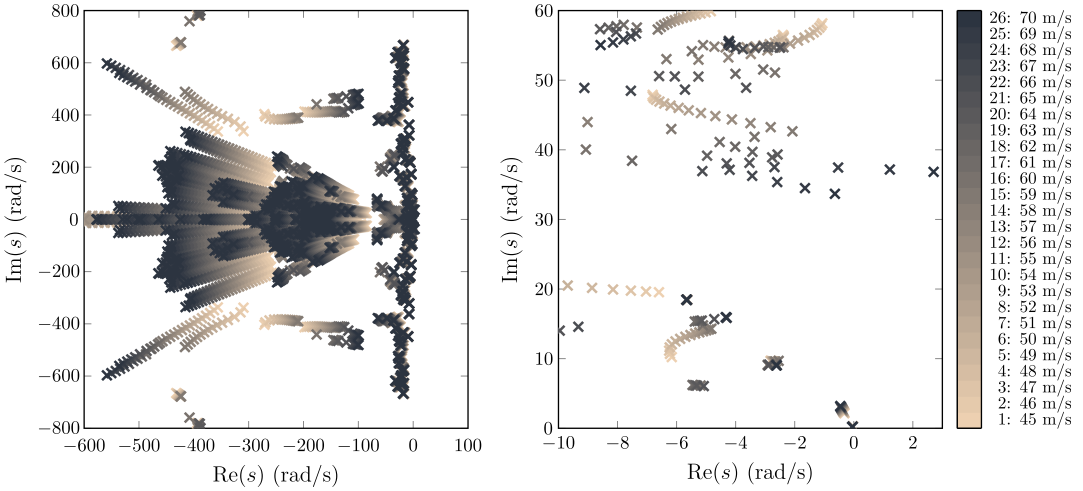


Fig. 20 **Pole map of the full-order closed-loop system with zoom around the flutter eigenfrequencies.**

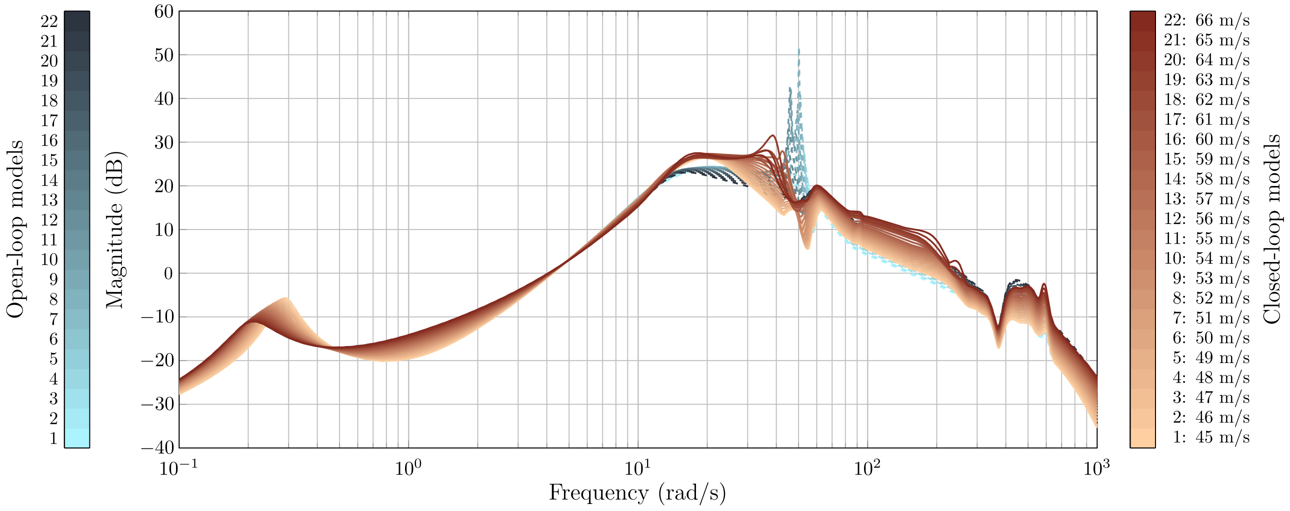


Fig. 21 **Maximum singular value for $N \in \{1, \ldots, 22\}$ from (δ_S, δ_{AS}) to $(\dot{\eta}_1, \dot{\eta}_2)$.**

gain and phase variations. This difficulty is overcome by considering disk margins, which consider simultaneous gain and phase perturbations. Because K_{sch} is multivariable, both single-loop and multiloop disk margins are assessed to ensure that the case of simultaneous perturbations on different channels of the control loop is also taken into account [31].

Robustness of the system against uncertain parameters, such as mass, position of center of gravity, and the eigenfrequencies and damping of the flutter modes, among others, could be assessed by making use of the structured singular value ([20] Chap. 8). However, no perturbed high-order CFD/FEM models (obtained by considering bounded uncertainties in critical parameters) were available. In addition, the reduced-order model used for design was obtained through a combination of modal and balanced reductions, which means that insight into the internal structure of the model is lost, and so it is impossible to perform this analysis with the present reduced models.

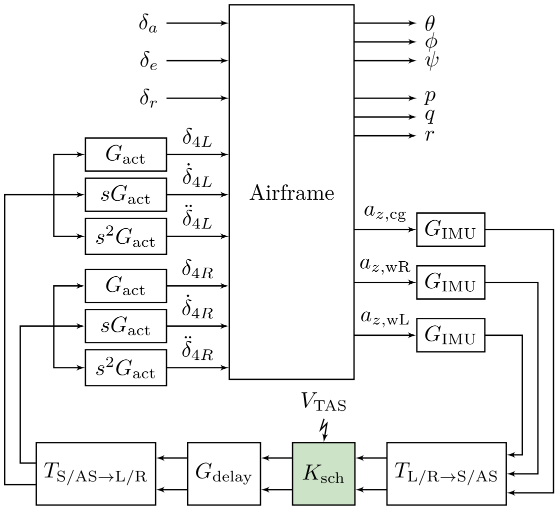


Fig. 22 Closed-loop system for analysis of the flutter controller.

Given that the full-order models have 1152 states, it is numerically impractical to perform such analysis with those models either. For this reason, the robustness analysis in this paper is restricted to the computation of minimum disk margins.

The selection of margin requirements for a flutter suppression controller is not straightforward, as it does not benefit from the extensive industrial experience that supports the design of controllers for rigid-body aircraft. For instance, the literature reports specifications of minimum phase margins ranging from 60° [32] down to 37° [10] and 30° [33]. Thus, in the present case, the requirements used in the design of the flutter controller are a minimum gain margin of 6 dB and a minimum phase margin of 35°. In addition, a minimum delay margin of 20 ms is required. This represents 4 skipped calculation steps of the flight control computer (in addition to the standard computational delay of 15 ms that is accounted for in $G_{\rm delay}$). These requirements are represented in Fig. 26 as red regions in each respective plot.

For the computation of the robustness margins, the coordinate transformations in Eqs. (1) and (4) are considered as part of the controller. This means that the loop breaks happen at the actual input (δ_{4L} and δ_{4R}) and output ($a_{z,\rm cg}$, $a_{z,\rm wL5}$, $a_{z,\rm wL6}$, $a_{z,\rm wR5}$, and $a_{z,\rm wR6}$) signals of the aircraft. This is done to ensure that the uncertainties are considered in the physical actuators and sensors and thus contemplate important cases such a phase discrepancy between the vertical acceleration measurements at the center of gravity and at the wingtips, for example.

Figure 26 shows the different robustness margins computed with the scheduled controller $K_{\rm sch}$. Figure 26a depicts the single- and multiloop (SL and ML, respectively) gain margins at the actuators and sensors. The single-loop input disk margin (blue right-facing triangles) remains above 13 dB up to $V_{\rm TAS} = 56$ m/s, whereas the corresponding output margin (orange left-facing triangles) is above 20 dB. When multiloop uncertainties are considered, some degradation of the gain margins is observed as expected. Nonetheless, the multiloop input disk margin (blue squares) is still above 6 dB up to 58 m/s. The corresponding output margin (orange circles) drops below the 6 dB threshold for lower speeds, although remaining close to the 5 dB mark. Finally, considering the multiloop input–output case (black crosses), it is seen that further degradation occurs, down to 3 dB. As pointed out in [10], this robustness margin can be overly conservative because it considers independent uncertainties in each actuator and sensor channel, whereas all measurements in the FLEXOP demonstrator come from the same type of IMU component and both actuators are the same.

A similar pattern is observed for the phase margins in Fig. 26b. In this case, the single-loop input and output margins, as well as the multiloop input case, are above the threshold of 35° for airspeeds

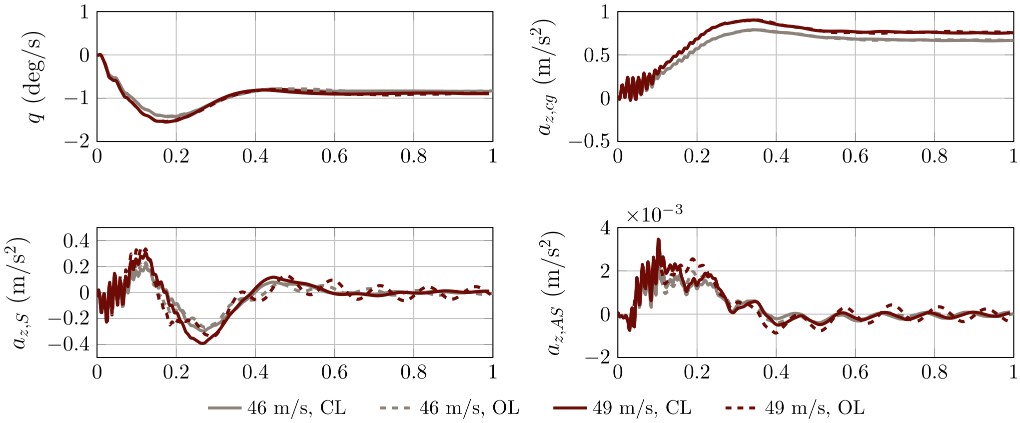


Fig. 23 Open-loop and closed-loop response to an elevator step input at $V_{\rm TAS} = 46$ and 49 m/s.

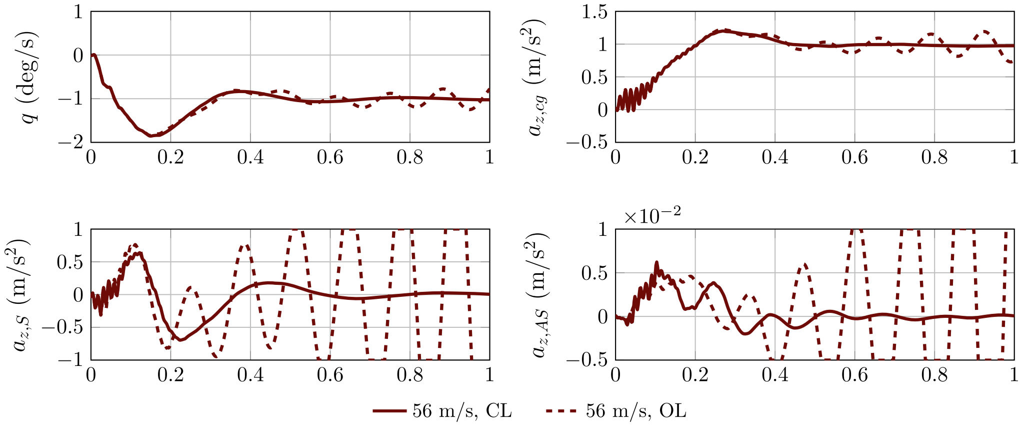


Fig. 24 Open-loop and closed-loop response to an elevator step input at $V_{\rm TAS} = 56$ m/s.

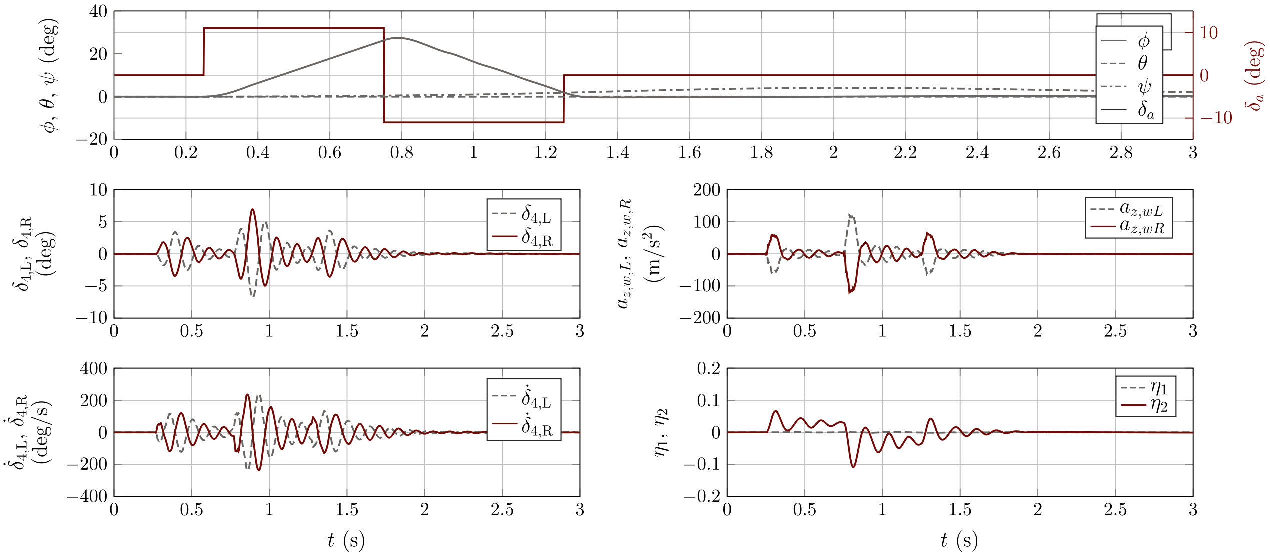


Fig. 25 Closed-loop response to aileron doublet input at the speed of 57 m/s.

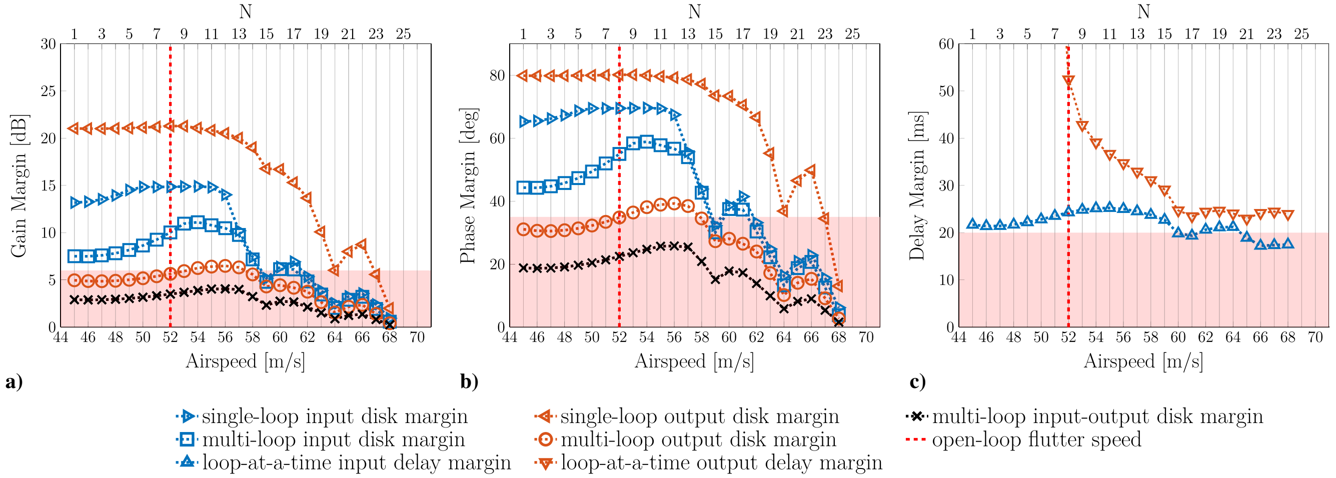


Fig. 26 Minimum a) gain, b) phase, and c) delay margins with respect to true airspeed.

below 58 m/s, whereas the multiloop output margin stays above 30°. With respect to the ML input–output disk margin, it is below 20° for all airspeeds except for a small region between 50 and 58 m/s, where it goes up to around 25°. Finally, Fig. 26c depicts the classic loop-at-a-time input and output delay margins. The latter is infinite below the open-loop flutter speed, and both stay above the threshold of 20 ms up to $V_{\mathrm{TAS}} = 59$ m/s and remain above 17 ms throughout the stability domain of the closed-loop system.

3. *Nonlinear Analysis*

This section presents the performance assessment of the flutter controller using the fully flexible nonlinear model. For this, the same interconnection presented in Fig. 22 is used, but the LTI model of the aircraft is replaced by the original nonlinear model. Then, the aircraft is trimmed in the same flying conditions that yielded the set of LTI models, and the response of the closed-loop system to step and doublet inputs is analyzed and compared with the corresponding linear responses.

Figure 27 shows the response of the aircraft to a step in the elevator input δ_e at the speed of $V_{\mathrm{TAS}} = 56$ m/s, corresponding to the linear time-domain test performed for Fig. 24. The simulations using both the nonlinear (solid lines) and linear (dashed lines) systems are shown, for comparison. The response of the nonlinear and linear systems are fairly similar, and they show that the controller performs well also in the nonlinear simulation. The guaranteed closed-loop flutter speed obtained with the nonlinear system is also $V_{\mathrm{TAS}} = 69$ m/s, the same as with the linear analysis of Sec. IV.A.1.

The above analysis using the fully flexible high-order nonlinear model allows the verification of the correct behavior of the controller and its flutter suppression capabilities without the actual interaction of a rigid-motion controller. Nonetheless, a coupled (baseline rigid-motion plus flutter controllers) analysis is necessary in order to fully assess the flutter controller in more realistic conditions. This coupled analysis is performed in the next section.

B. With Baseline Rigid-Motion Controller

The demonstrator built in the framework of the FLEXOP project will allow the validation of the AFS in flight test. For this, an autopilot system has been developed by the SZTAKI and DLR teams [30]. This autopilot is based on a baseline rigid-body controller, which consists of inner loops for the control of the longitudinal and lateral/directional motions of the aircraft, as well as a trajectory planner for commanding the altitude h, heading χ, and indicated airspeed V_{IAS} of the aircraft. After manual takeoff, the autopilot will be engaged with full-authority over the aircraft motion. The trajectory planner is designed to guide the aircraft along a horse-race-track shape, as represented in Fig. 28. In one of the straight legs of the flight course, denominated the test leg, the aircraft will be accelerated to incrementally increasing cruise flight velocities, until the open-loop flutter speed is attained. This will allow the assessment of the in-flight performance of the flutter controller in cruise conditions at different speeds.

The overall closed-loop system (including the baseline rigid plus flutter controllers) is shown in Fig. 29. In the previous sections, the flutter controller was designed and its performance analyzed across

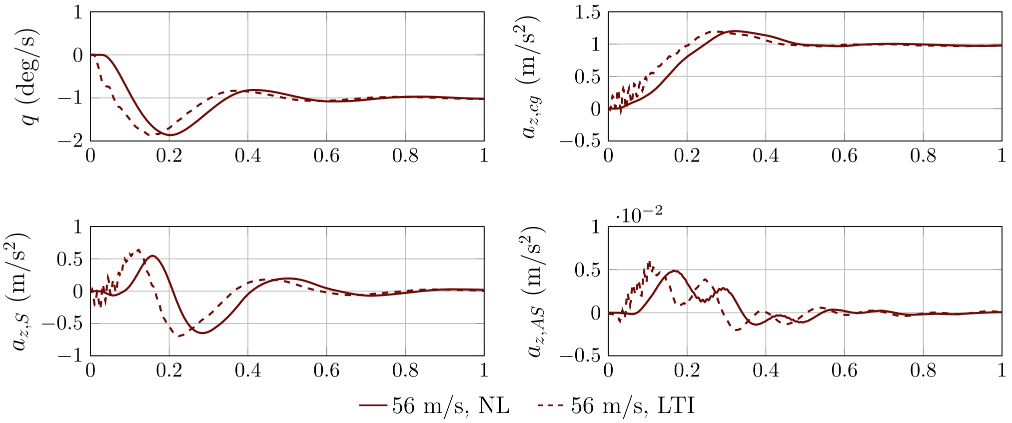


Fig. 27 **Nonlinear and linear responses to an elevator step input at 56 m/s.**

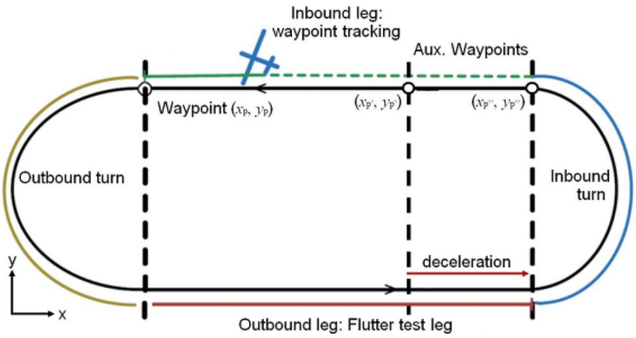


Fig. 28 **Horse-race track for evaluation of active flutter suppression capabilities [18].**

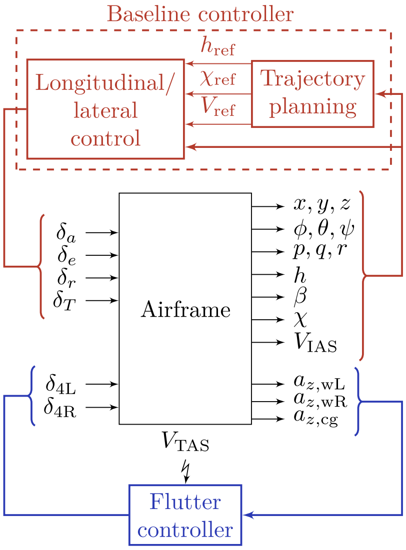


Fig. 29 **Closed-loop system including the guidance and active flutter control loops.**

the flight envelope with respect to the true airspeed V_{TAS}. The true airspeed was used because the LTI models were obtained by performing trimming and linearization with respect to V_{TAS}. Nonetheless, the baseline rigid-motion controller is designed to allow the aircraft to track indicated airspeed (V_{IAS}) references, and so the discussion in this section is based on V_{IAS} instead of V_{TAS}, with the conversion provided where appropriate. New versions of the nonlinear model should allow trimming to be done according to indicated airspeed (V_{IAS}), instead of true airspeed (V_{TAS}). This is important for the proper scheduling of the flutter controller, as the aircraft does not include static air temperature and pressure altitude measurements, and so the true airspeed is unknown to the flight computer. In this section the nonlinear simulator is now used with both controllers using two main sets of maneuvers, a so-called speed staircase (Sec. IV.B.1) and the horse-race track (Sec. IV.B.2). These analyses will serve to verify the controllers before hardware-in-the-loop (HIL) validation and subsequent flight tests.

1. Speed Staircase

The baseline and flutter controllers are implemented in the simulator of the nonlinear aerodynamic model of the aircraft developed by DLR. The simulator can then be used as a high-fidelity verification tool for the performance of both controllers as well as their interaction. As a first step in the assessment of the flutter controller performance, the baseline controller is set to attain level flight, that is, at constant height (348 m) and heading, and with a staircase speed reference increase starting from $V_{\mathrm{IAS}} = 38$ m/s up to 64 m/s in increments of 2 m/s every 20 s (see Fig. 30). It is seen that the steps tracking are homogeneous and adequate up to a speed of 64 m/s, where the simulation was stopped due to the appearance of unstable dynamics in the antisymmetric flutter modes, as it shall be seen next.

Figure 31 shows the corresponding time evolution of the symmetric ($a_{z,S}$) and antisymmetric ($a_{z,\mathrm{AS}}$) acceleration at the wingtips, without (gray dashed lines) and with (red solid lines) the flutter controller. In the former case, the system becomes unstable at $V_{\mathrm{IAS}} = 50$ m/s, or $V_{\mathrm{TAS}} \approx 52$ m/s, although with a very high time to double (nonperceptible in Fig. 31). This was expected from the open-loop flutter speed obtained from the analysis of the linearized systems. As V_{IAS} approaches 52 m/s ($V_{\mathrm{TAS}} \approx 54$ m/s), around the 150 s mark, the oscillations explode. The flutter controller is able to reduce the oscillations at the wingtips, even below the open-loop flutter speed. The behavior above this speed remains quite similar to those at lower speeds, which is a an indicator that the flutter controller allows the expansion of the flight envelope without affecting maneuverability. The closed-loop system with flutter controller is stable up to $V_{\mathrm{IAS}} = 60$ m/s ($V_{\mathrm{TAS}} \approx 62$ m/s), and at 62 m/s ($V_{\mathrm{TAS}} \approx 64$ m/s) it can be seen that the antisymmetric flutter mode becomes unstable, although with a very large time to double. When V_{IAS} reaches 64 m/s ($V_{\mathrm{TAS}} \approx 66$ m/s), the time to double becomes much smaller, and the flutter oscillations quickly diverge. This indicates that there is interaction between the flutter and baseline controllers at higher speeds, as the closed-loop flutter speed is reduced in the presence of the latter.

The deflection of the wingtip ailerons, δ_{4L} and δ_{4R}, is shown in Fig. 32. Both deflections are quite similar during the greater part of this simulation, as seen on the zoom over the 80–85 s time region (plot a), which indicates that the symmetric flutter mode is being excited. By the end of the simulation, the antisymmetric flutter mode becomes unstable, and this is observed in the antisymmetric action of the control surfaces on the zoom over the 255–260 s time region (plot b). For this level-flight simulation, the maximum deflection below the closed-loop flutter speed is lower than 0.002°, even during the acceleration phase. This can be explained by the fact that a linear elastic model is used to describe the structural dynamics, which are damped and stabilized by the flutter controller, resulting in stable

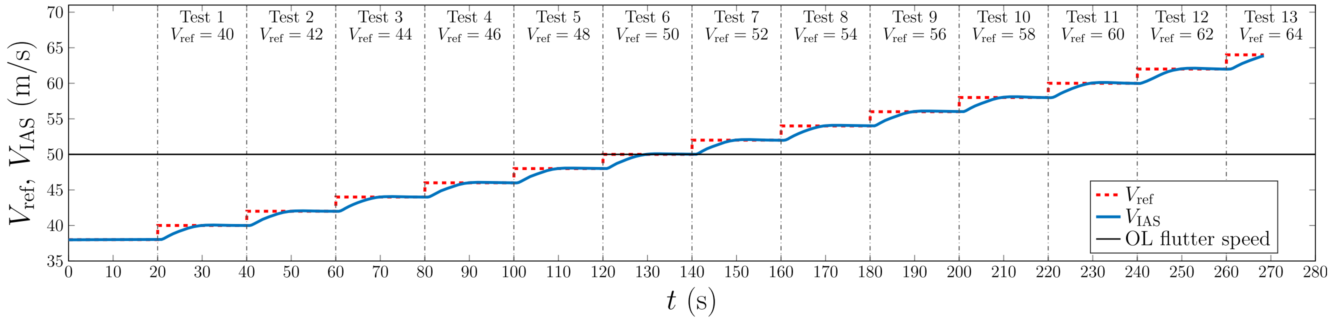


Fig. 30 **Speed reference tracking with flutter controller engaged.**

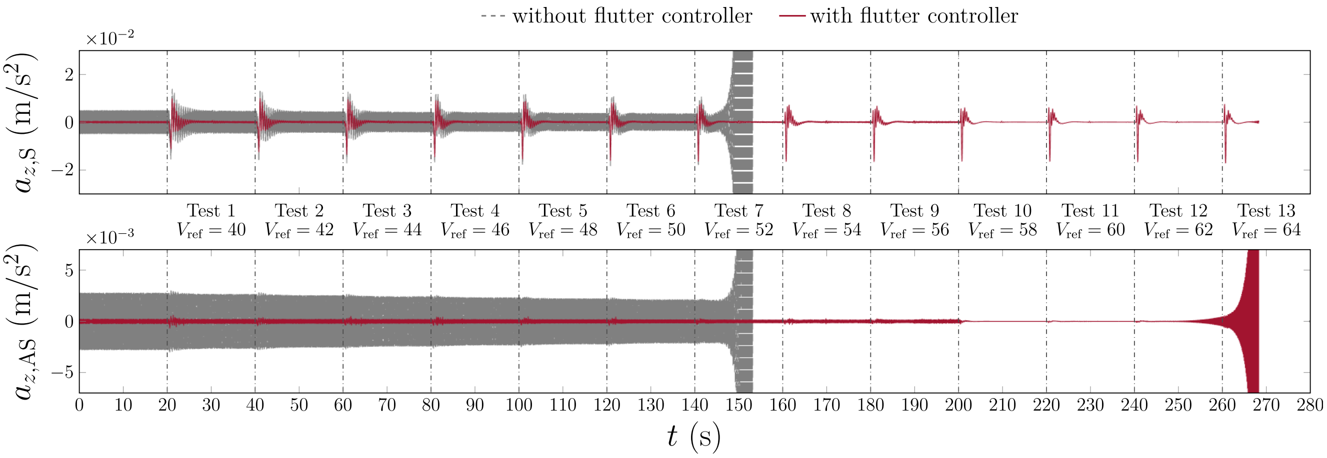
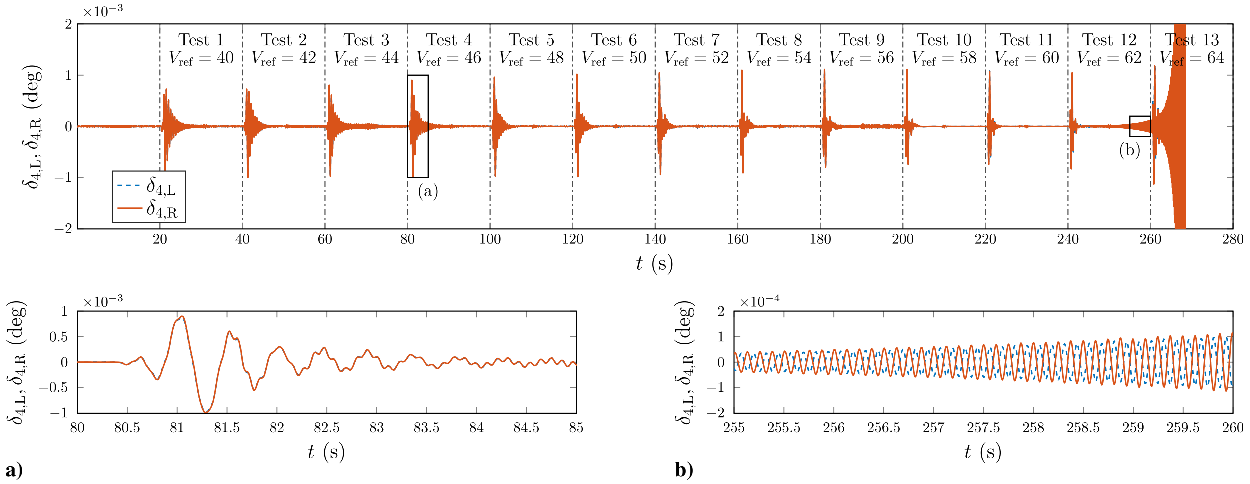


Fig. 31 **Symmetric and antisymmetric wingtip acceleration for the speed staircase simulation.**



a) b)

Fig. 32 **Deflection of the wingtip ailerons for the speed staircase simulation.**

linear dynamics (in level cruise flight). Because this staircase maneuver does not directly excite the flutter modes, very little control action is required of the controller. This indicates that the control effort is very low when in cruise-flight conditions, and then that the AFS has not a big impact on the energetic efficiency of the aircraft. The deflection rates in this simulation (not shown) are also very small with respect to the limits of ±1000°/s, and thus the flutter controller is far from saturation. As the aircraft approaches the speed of $V_{\mathrm{IAS}} = 62$ m/s, the control effort increases and becomes much higher at 64 m/s until reaching saturation, as the flutter controller is no longer able to ensure stability.

As discussed in Sec. IV.A, in the absence of the baseline rigid-motion controller, the closed-loop flutter speed was identified in the simulations with the nonlinear model at $V_{\mathrm{TAS}} = 69$ m/s. With the addition of the baseline controller, the closed-loop flutter speed has dropped and is around a speed of $V_{\mathrm{IAS}} = 62$ m/s (or $V_{\mathrm{TAS}} \approx 64.5$ m/s). This indicates that there is interaction between the controllers for rigid motion and AFS, at least at higher speeds. It should be noted that this interaction is not between the flutter controller and the rigid-motion dynamics, but rather with the baseline rigid-motion control action, as this is in contrast with the results presented in Sec. IV.A. Nonetheless, the flight envelope is expanded up to $V_{\mathrm{IAS}} = 60$ m/s, or equivalently $V_{\mathrm{TAS}} = 62.4$ m/s, an increase of around 22% with respect to the open loop. Future research will be directed at understanding this coupling, and how to improve the behavior of the complete closed-loop system.

2. *Horse-Race Track*

This section presents the analysis of the closed-loop nonlinear system as it follows the trajectory reference of the horse-race track. At each lap, the aircraft is accelerated up to a higher speed in the test leg in 4 m/s increments. This simulation is very close to the mission that will be undertaken by the demonstrator in the actual flight test and thus is of vital importance for the verification of the controller.

Figure 33 shows the three-dimensional xyz trajectory of the aircraft as it follows the horse-race track for the cases without (gray dashed lines) and with (red solid lines) flutter controller. Both simulations start at point $(x, y, z) = (790, 0, 350)$ m, which is marked with a black dot on the front-left side of the track, and with a blue cylinder to indicate the test leg. Outside of the test leg cylinder, both trajectories are virtually the same, which once again indicates that the flutter controller does not have an impact on the rigid-motion dynamics. During the test leg, some deviations between both trajectories can be seen. This is mostly due to the attempt to fly the aircraft without the flutter controller above the open-loop flutter speed, which leads to high-magnitude oscillations on the wings that impact the rigid dynamics (low) frequency region (this is more clearly seen in Fig. 34b).

Figure 34a shows the speed profile during the horse-race-track simulation with the flutter controller engaged, along with the speed reference. The aircraft goes through the test leg six times during the mission, with the first test at $V_{\mathrm{ref}} = 42$ m/s, and all subsequent tests with 4 m/s increments. It is seen that the response of the speed tracking presents some overshoot, especially at the highest speed ($V_{\mathrm{ref}} = 62$ m/s).

In Fig. 34b, the pitch angle of the aircraft is plotted against time for the cases without and with the flutter controller. The flutter-induced oscillations in the rigid dynamics in the former case can be clearly seen in this figure during the tests from $V_{\mathrm{ref}} = 54$ m/s onward. These are not present with the flutter controller engaged, as expected. The performance of the controller can be further assessed in Fig. 34c, where the symmetric wingtip accelerations are plotted for the cases without and with flutter controller. The controller is able to significantly reduce the flutter oscillations as the aircraft goes through the test leg, thus allowing it to fulfill the mission designated by the autopilot. In test 6, that is, at $V_{\mathrm{ref}} = 62$ m/s, some level of flutter oscillation is present. This is due to the overshoot in the speed tracking, as seen in Fig. 34a, which results in the aircraft accelerating to speeds above $V_{\mathrm{IAS}} = 62$ m/s. According to the results presented in the previous section, the closed-loop system is only able to effectively damp the flutter modes up to $V_{\mathrm{IAS}} = 60$ m/s, and so these oscillations could be expected.

The controller performance is now considered in the presence of disturbances: measurement noise, turbulence, and gusts. The noise in the sensor measurements is modeled with the use of coloring filters to achieve realistic results, as described in ([34] Sec. 3.2.2). For this simulation, the presence of noise in the acceleration measurements

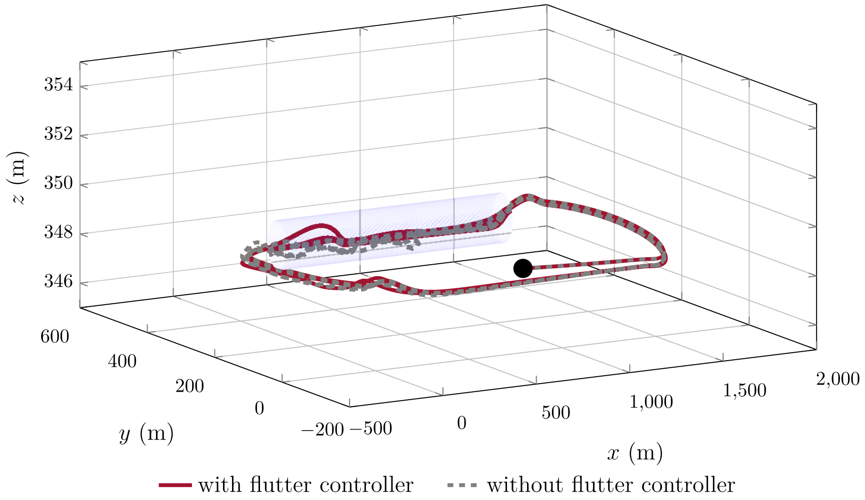


Fig. 33 *xyz* **trajectory on the horse-race track, with and without flutter controller.**

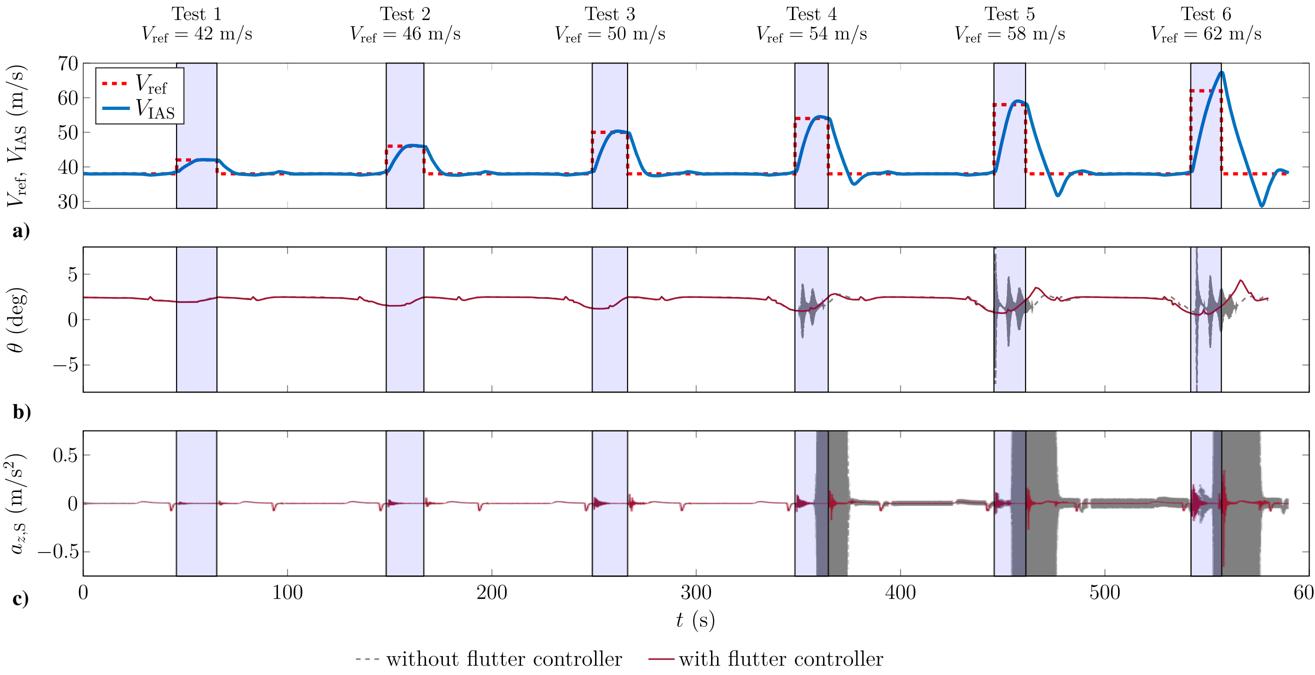


Fig. 34 **Speed reference tracking with and without flutter controller.**

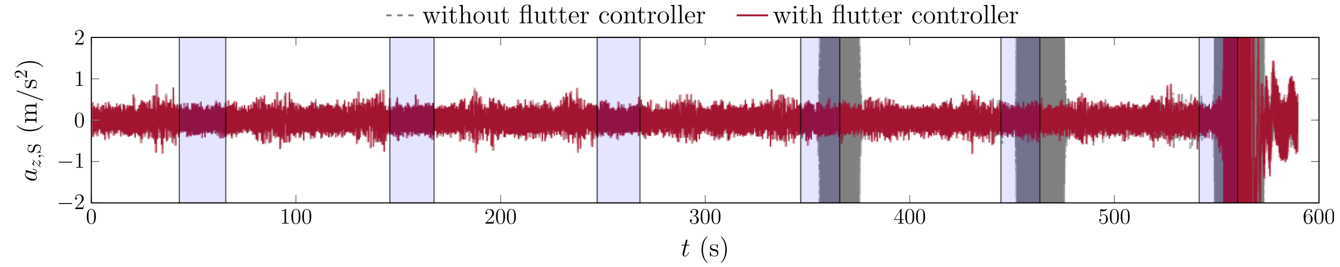


Fig. 35 Symmetric wingtip acceleration in the horse-race track, with and without flutter controller, in the presence of gusts and sensor noise.

provided by the IMUs is considered (both in the center of gravity and in the wingtips), as well as in the measurement of speed, used for scheduling purposes. The turbulence is modeled by a Dryden filter, which is set up using the dimensions of the aircraft and actual wind data obtained gathered at the site where the flight tests will take place. In addition, discrete vanishing 1-cosine wind gusts are added along the y and z axes. These are added at 40 and 100 s in the vertical direction, and at 165 and 200 s in the lateral direction.

Figure 35 shows the symmetric acceleration at the wingtips of the aircraft during the simulation of the horse-race track in the presence of noise, turbulence, and gusts. The case with the flutter controller shows that it is able to maintain performance below the speed of $V_{\mathrm{IAS}} = 60$ m/s, but some degradation is observed above this mark. Unlike the nondisturbed case, the flutter controller is not able to maintain stability in the last test track, that is, when the autopilot attempts to impose a speed of $V_{\mathrm{IAS}} = 62$ m/s. Nonetheless, as discussed before when analyzing the speed staircase simulation, the closed-loop system is unstable at this speed (even if with a very high time-to-double; see Fig. 31), and so this result is not unexpected.

V. Conclusions

This paper presents the results of the synthesis of a controller to provide AFS for the demonstrator aircraft developed in the framework of the FLEXOP project. The controller proposed is composed of three single-point H_∞ controllers, scheduled with respect to the true airspeed V_{TAS}. The proposed controller was shown to extend the flight envelope above the open-loop flutter critical speed by providing sufficient damping to the first symmetric and antisymmetric vibration modes. The results obtained are satisfactory and promising, even in simulation with a high-fidelity nonlinear model and in the presence of gusts and sensor noise. The flight-test campaigns to validate the flutter suppression capabilities of the controller are scheduled for the spring/summer of 2019.

Acknowledgments

This work has received funding from the Horizon 2020 Research and Innovation Programme under grant agreement number 636307, project Flutter Free Flight Envelope eXpansion for ecOnomical Performance improvement (FLEXOP). The authors are thankful to Matthias Wüstenhagen and Thiemo Kier from the German Aerospace Center (DLR) for providing the high-fidelity nonlinear model of the FLEXOP demonstrator, and also to Tamás Luspay from the Institute for Computer Science and Control (SZTAKI) and Daniel Ossmann from DLR for providing the baseline rigid-motion controller.

References

[1] Hodges, D. H., and Pierce, G. A., *Introduction to Structural Dynamics and Aeroelasticity*, Cambridge Aerospace Series, Vol. 15, 2nd ed., Cambridge Univ. Press, New York, 2011.

[2] "Aeroelastic Stability Substantiation of Transport Category Airplanes," Advisory Circular 25.629-1B, Federal Aviation Administration, Oct. 2014.

[3] "Certification Specifications and Acceptable Means of Compliance for Normal, Utility, Aerobatic, and Commuter Category Aeroplanes," CS-23—Amendment 4, European Aviation Safety Agency, July 2015.

[4] COM(2010) 2020, "A Strategy for Smart, Sustainable and Inclusive Growth," European Commission, March 2010, COM(2010) 2020.

[5] Livne, E., "Aircraft Active Flutter Suppression: State of the Art and Technology Maturation Needs," *Journal of Aircraft*, Vol. 55, No. 1, 2018, pp. 410–452.
https://doi.org/10.2514/1.C034442

[6] Regan, C. D., and Jutte, C. V., "Survey of Applications of Active Control Technology for Gust Alleviation and New Challenges for Lighter-Weight Aircraft," NASA TM-2012-216008, 2012.

[7] Wykes, J., "Structural Dynamic Stability Augmentation and Gust Alleviation of Flexible Aircraft," *5th Annual Meeting and Technical Display*, AIAA Paper 1968-1067, 1968.
https://doi.org/10.2514/6.1968-1067

[8] Gangsaas, D., and Ly, U.-L., "Application of a Modified Linear Quadratic Gaussian Design to Active Control of a Transport Airplane," *Guidance, Navigation, and Control and Co-Located Conferences*, AIAA Paper 1979-1746, 1979.
https://doi.org/10.2514/6.1979-1746

[9] Waszak, M. R., "Robust Multivariable Flutter Suppression for Benchmark Active Control Technology Wind-Tunnel Model," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 1, 2001, pp. 147–153.
https://doi.org/10.2514/2.4694

[10] Theis, J., Pfifer, H., and Seiler, P. J., "Robust Control Design for Active Flutter Suppression," *AIAA SciTech Forum*, AIAA Paper 2016-1751, 2016.
https://doi.org/10.2514/6.2016-1751

[11] Hjartarson, A., Seiler, P. J., and Balas, G. J., "LPV Aeroservoelastic Control Using the LPVTools Toolbox," *AIAA Atmospheric Flight Mechanics (AFM) Conference*, AIAA Paper 2013-4742, 2013.
https://doi.org/10.2514/6.2013-4742

[12] Burnett, E. L., Beranek, J. A., Holm-Hansen, B. T., Atkinson, C. J., and Flick, P. M., "Design and Flight Test of Active Flutter Suppression on the X-56A Multi-Utility Technology Test-Bed Aircraft," *The Aeronautical Journal*, Vol. 120, No. 1228, 2016, pp. 893–909.
https://doi.org/10.1017/aer.2016.41

[13] Danowsky, B. P., Kotikalpudi, A., Schmidt, D., Regan, C., and Seiler, P., "Flight Testing Flutter Suppression on a Small Flexible Flying-Wing Aircraft," *AIAA AVIATION Forum*, AIAA Paper 2018-3427, 2018.
https://doi.org/10.2514/6.2018-3427

[14] Cesnik, C. E. S., Senatore, P. J., Su, W., Atkins, E. M., and Shearer, C. M., "X-HALE: A Very Flexible Unmanned Aerial Vehicle for Nonlinear Aeroelastic Tests," *AIAA Journal*, Vol. 50, No. 12, 2012, pp. 2820–2833.
https://doi.org/10.2514/1.J051392

[15] Meddaikar, Y. M., Dillinger, J., Klimmek, T., Krüger, W., Wüstenhagen, M., Kier, T. M., Hermanutz, A., Hornung, M., Rozov, V., Breitsamter, C., et al., "Aircraft Aeroservoelastic Modelling of the FLEXOP Unmanned Flying Demonstrator," *AIAA SciTech Forum*, AIAA Paper 2019-1815, 2019.
https://doi.org/10.2514/6.2019-1815

[16] Rößler, C., Stahl, P., Sendner, F., Hermanutz, A., Koeberle, S., Bartasevicius, J., Rozov, V., Breitsamter, C., Hornung, M., Meddaikar, Y. M., et al., "Aircraft Design and Testing of FLEXOP Unmanned Flying Demonstrator to Test Load Alleviation and Flutter Suppression of High Aspect Ratio Flexible Wings," *AIAA SciTech Forum*, AIAA Paper 2019-1813, 2019.
https://doi.org/10.2514/6.2019-1813

[17] Sodja, J., Werter, N., and De Breuker, R., "Design of a Flying Demonstrator Wing for Manoeuvre Load Alleviation with Cruise Shape Constraint," *AIAA SciTech Forum*, AIAA Paper 2018-2153, 2018.
https://doi.org/10.2514/6.2018-2153

[18] Luspay, T., Baár, T., Teubl, D., Vanek, B., Ossmann, D., Wüstenhagen, M., Pusch, M., Kier, T. M., Waitman, S., Iannelli, A., et al., "Flight Control Design for a Highly Flexible Flutter Demonstrator," *AIAA SciTech Forum*, AIAA Paper 2019-1817, 2019.
https://doi.org/10.2514/6.2019-1817

[19] Pusch, M., Ossmann, D., and Luspay, T., "Structured Control Design for a Highly Flexible Flutter Demonstrator," *Aerospace*, Vol. 6, No. 3, 2019, p. 27.
https://doi.org/10.3390/aerospace6030027

[20] Skogestad, S., and Postlethwaite, I., *Multivariable Feedback Control: Analysis and Design*, 2nd ed., Wiley, New York, 2001.

[21] Mukhopadhyay, V., "Transonic Flutter Suppression Control Law Design, Analysis and Wind-Tunnel Results," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 5, Sept. 2000, pp. 930–937. https://doi.org/10.2514/2.4635

[22] Iannelli, A., Marcos, A., and Lowenberg, M., "Study of Flexible Aircraft Body Freedom Flutter with Robustness Tools," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 5, 2018, pp. 1083–1094. https://doi.org/10.2514/1.G003165

[23] Stahl, P., Sendner, F.-M., Hermanutz, A., Rößler, C., and Hornung, M., "Mission and Aircraft Design of FLEXOP Unmanned Flying Demonstrator to Test Flutter Suppression Within Visual Line of Sight," *AIAA AVIATION Forum*, AIAA Paper 2017-3766, 2017. https://doi.org/10.2514/6.2017-3766

[24] Wüstenhagen, M., Kier, T., Meddaikar, Y. M., Pusch, M., Ossmann, D., and Hermanutz, A., "Aeroservoelastic Modeling and Analysis of a Highly Flexible Flutter Demonstrator," *2018 Atmospheric Flight Mechanics Conference*, AIAA Paper 2018-3150, 2018. https://doi.org/10.2514/6.2018-3150

[25] Frulla, G., Cestino, E., and Marzocca, P., "Critical Behaviour of Slender Wing Configurations," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 224, No. 5, 2009, pp. 587–600. https://doi.org/10.1243/09544100 JAERO553

[26] Jeanneau, M., Aversa, N., Delannoy, S., and Hockenhull, M., "AWIATOR's Study of a Wing Load Control: Design and Flight-Test Results," *IFAC Proceedings Volumes*, Vol. 37, No. 6, June 2004, pp. 469–474. https://doi.org/10.1016/S1474-6670(17)32219-X

[27] Wildschek, A., "An Adaptive Feed-Forward Controller for Active Wing Bending Vibration Alleviation on Large Transport Aircraft," Ph.D. Thesis, Technical Univ. of Munich, Munich, 2008.

[28] Zhou, K., Doyle, J. C., and Glover, K., *Robust and Optimal Control*, Prentice Hall, Upper Saddle River, NJ, 1996.

[29] Marcos, A., "A Gain Scheduled H-Infinity Controller for a Re-Entry Benchmark," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2010-7567, 2010. https://doi.org/10.2514/6.2010-7567

[30] Ossmann, D., Luspay, T., and Vanek, B., "Baseline Flight Control System Design for an Unmanned Flutter Demonstrator," *IEEE Aerospace Conference*, 2019. https://doi.org/10.1109/AERO.2019.8741853

[31] Blight, J. D., Dailey, R. L., and Gangsaas, D., "Practical Control Law Design for Aircraft Using Multivariable Techniques," *International Journal of Control*, Vol. 59, No. 1, 1994, pp. 93–137. https://doi.org/10.1080/00207179408923071

[32] Roger, K. L., Hodges, G. E., and Felt, L., "Active Flutter Suppression—A Flight Test Demonstration," *Journal of Aircraft*, Vol. 12, No. 6, 1975, pp. 551–556. https://doi.org/10.2514/3.59833

[33] Adams, J., William, M., Christhilf, D. M., Waszak, M. R., Mukhopadhyay, V., and Srinathkumar, S., "Design, Test, and Evaluation of Three Active Flutter Suppression Controllers," NASA, TM-4338, 1992.

[34] Marcos, A., "Aircraft Applications of Fault Detection and Isolation Techniques," Ph.D. Thesis, Univ. of Minnesota, Minneapolis, MN, Feb. 2004.