Article

Model-Based Robust Control Design for a Highly Flexible Flutter Demonstrator

Manuel Pusch [1], Daniel Ossmann [1,*], and Tamas Luspay [2]

[1] Institute of System Dynamics and Control, German Aerospace Center (DLR), 82234 Wessling, Germany; {manuel.pusch, daniel.ossmann}@dlr.de

[2] Systems and Control Lab, Institute for Computer Science and Control, Budapest, Hungary 1111; luspay.tamas@sztaki.mta.hu

* Correspondence: daniel.ossmann@dlr.de; Tel.: +49-8153-282683

Version January 18, 2019 submitted to Aerospace

Abstract: This article presents the model based design and verification of the flight control system of a highly flexible flutter demonstrator developed in the European FLEXOP project. The flight control system includes a baseline controller to operate the aircraft fully autonomously and an active flutter controller to stabilize the unstable aeroelastic modes and extend the aircraft's operational range. The baseline control system features a classical cascade flight control structure with scheduled control loops to augment the lateral and longitudinal axis of the aircraft. The flutter controller uses an advanced blending technique to blend the flutter relevant sensor and actuator signals. These blends decouple the unstable modes from each other and allow controlling them individually by scheduled single loop controllers. For the tuning of the free controller parameters a model based approach is presented, solving multi-objective, non-linear optimization problems via robust control techniques. The developed control system, including baseline and flutter control algorithms, is verified in an extensive simulation campaign using a high fidelity, non-linear simulator of the flutter demonstrator.

Keywords: flutter control; flight control, model based control design; robust control; optimal blending; non-linear simulation.

1. Introduction

Today's aircraft manufacturers are eager to fulfill the greener imperative demanded by today's society and allow for a more economic operation of aircraft. Besides the efficiency of engines and aerodynamics, the aircraft weight and the wing aspect ratio have a major impact on the fuel consumption. A reduction of aircraft weight is achieved by using new materials like carbon composites, as it has been successfully achieved for example with the Airbus A350 or the Boeing 787, while larger wings increase the aspect ratio. These approaches, however, decrease the aircraft velocity at which undesired effects like flutter, i.e. the unstable coupling between the aerodynamics and the aircraft structure, occur. If the trend of reducing the aircraft structure is continued, these effects will appear in the desired operational aircraft envelopes. Possible countermeasures are active control techniques which allow to stabilize these unstable dynamics and extend the operational region of the aircraft. Such advanced control algorithms require model based design methods which call for adequate models of the aeroservorelastic effects. Thus, the design of active flutter controllers is a challenging task and has raised attention to the academic community, see for example [1–3] for valuable contributions. In this article the design of a flight control system including the baseline controller and the active flutter controller for a highly flexible flutter demonstrator is presented. The considered aircraft, depicted in Figure 1, is the main demonstrator of the Horizon 2020 project Flutter Free FLight Envelope eXpansion for ecOnomic Performance improvement (FLEXOP) to develop and test active flutter suppression control algorithms [4,5].

The proposed control system features two main parts, the baseline flight control system to navigate the aircraft fully autonomously around the predefined flight test track and the active flutter control algorithms to stabilize the aircraft's flutter modes and thereby extend its operational range. The architecture of the presented baseline controller to augment the rigid body motion features a classical cascaded flight controller architecture [6,7] with well proven feedback loops of proportional-integral-derivative (PID) controllers together with damping augmentation. These control loops provide capabilities for augmented pilot-in-the-loop flights as well as for autonomous flights. A major task during the design of active flutter control algorithms is the adequate fusion of the numerous available measurements on the wings and the different control inputs. This is commonly done in a pre-processing step by the control engineer before deriving the control algorithm. In Section 2 of this article we present a new systematic approach to analytically blend the available inputs and outputs to isolate the modes to be stabilized. This enables a simple control design for each individual mode, using parametrized single-input single-output (SISO) controllers of a predefined structure.

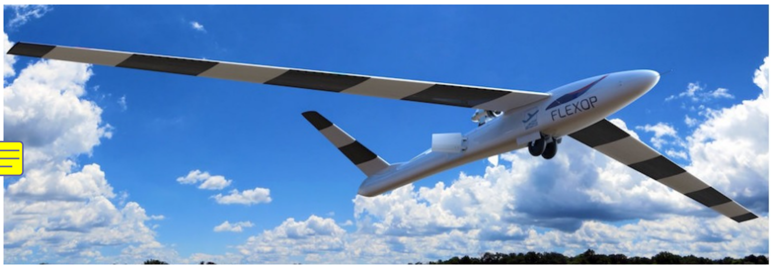

Figure 1. FLEXOP flutter demonstrator.

By defining the structure of the controllers in advance, the design of the active flutter controller as well as the baseline controller in this article comes down to the selection of adequate control gains. For this selection a model based approach using robust control techniques is proposed in Section 3. Two generic control design problems are defined: The first problem defines a multi-model, multi-objective optimization for deriving a controller which is robust against parameter variations. The second design is posted as parametric multi-objective problem for designing gain-scheduled controllers. Both problem can be solved using non-smooth optimization based robust control algorithms [8]. In the case of the gain-scheduled controller this approach enables the direct design of gain scheduled controller parameters single design step, i.e., avoiding the classically applied point-wise design. The presented tool chain for the controller design is applied to the FLEXOP flutter demonstrator in Section 4. A detailed overview of the baseline controller functionality and the employed loops and design criteria is provided. Further, the application of the proposed blending vector design and parameter tuning to derive the controller gains is discussed for the active flutter controller, providing valuable insight to the reader on how such algorithms are developed. The designed flight control system of the FELXOP flutter demonstrator is verified in an extensive verification campaign and its results are reported in Section 5. For the verification campaign a high fidelity, non-linear simulator of the closed-loop aircraft [9,10], including structural and aerodnymical effects as well as detailed actuator, sensor, engine models, is available. The verification includes wind scenarios to test the disturbance attenuation, acceleration scenarios to verify the stabilization capabilities of the active flutter control algorithm, and flights along the predefined flight test pattern on which the real flight tests will be performed.

2. H_2timal Input and Output Blending

In this section the theoretical background to optimally blend inputs and outputs of a system is provided. The approach blends the inputs and outputs in a way, that the H_2-norm of the mode to be controlled is maximized. This approach is especially applicable for aeroelastic problems, for which commonly a high number of sensors, e.g. strain or acceleration measurements, are available and need to be fused accordingly within the control algorithm.

2.1. Modal Control of Linear Time-Invariant Systems

An linear time-invariant linear time-invariant (LTI) system with n_u inputs, n_y outputs and n_x states which is physically realizable can be described by the transfer function matrix

$$G(s) = C\left(sI - A\right)^{-1} B + D, \tag{1}$$

where $A \in \mathbb{R}^{n_x \times n_x}, B \in \mathbb{R}^{n_x \times n_u}, C \in \mathbb{R}^{n_y \times n_x}, D \in \mathbb{R}^{n_y \times n_u}$ and s denotes the Laplace variable. Assuming that A is diagonalizable, a modal decomposition of $G(s)$ is possible such that

$$G(s) = \sum_{i=1}^{n_i} M_i(s) + D,$$

where the individual modes $i = 1, ..., n_i$ are given as

$$M_i(s) = \begin{cases} \dfrac{R_i}{s - p_i} & \text{if } \Im(p_i) = 0 \\ \dfrac{R_i}{s - p_i} + \dfrac{\overline{R}_i}{s - \overline{p}_i} & \text{otherwise.} \end{cases} \tag{2}$$

According to (2), a mode i is either described by a single real pole p_i with an imaginary part $\Im(p_i) = 0$ or a conjugate complex pole pair p_i and $\overline{p}_i$. Hence, the number of modes n_i does not necessarily equal the number of states n_x, i.e. $n_i \leq n_x$. Each pole p_i is associated with a residue $R_i = c_i b_i^T \in \mathbb{C}^{n_y \times n_u}$, where $b_i \in \mathbb{C}^{n_u}$ and $c_i \in \mathbb{C}^{n_y}$ are the pole input and output vectors, respectively. For a real pole, the corresponding pole vectors and residue are real, and for a conjugate complex pole pair the pole vectors and residues are conjugate complex. Achtung, c sind ne halbe Seite weiter unten auch die SISO Regler.

In general, a mode i is considered to be asymptotically stable if $\Re(p_i) < 0$ and unstable if $\Re(p_i) > 0$. In case $\Re(p_i) = 0$, the mode is considered to be undamped, which also includes a pole in the origin. Furthermore, the natural frequency of a mode is given as $\omega_{\mathrm{n},i} = |p_i|$ and for $\omega_{\mathrm{n},i} \neq 0$, the corresponding relative damping is $\zeta_i = -\Re(p_i)/\omega_{\mathrm{n},i}$. Note that for a conjugate complex pole pair, the corresponding real parts $\Re(p_i) = \Re(\overline{p}_i)$ and magnitudes $|p_i| = |\overline{p}_i|$ are equal. For more information on modal decomposition and the properties of individual modes see, for instance, [11].

The task of controlling a single mode $M_j(s) \in \{M_i(s)\}$ of a high order dynamic system can quickly become challenging when the number of control inputs or measurement outputs is increased. In order to reduce the complexity of the control problem, it is proposed to weight and sum up the measurement signals such that the resulting virtual measurement output $v_{y,j}$ represents the response of the mode to be controlled. Similarly, it is proposed to generate a virtual control input $v_{u,j}$ which is distributed to available control inputs such that the target mode can be individually controlled. In other words, the mode to be controlled is isolated by blending inputs and outputs. The corresponding input and output blending vectors $k_{u,j} \in \mathbb{R}^{n_u}$ and $k_{y,j} \in \mathbb{R}^{n_y}$ depend on the shape of the targeted mode and can be seen as directional filters. This implies a high robustness against frequency variations as the blending vectors are independent of the mode's natural frequency. Blending the inputs and outputs as proposed, a simple single input and single output (SISO) controller $c_j(s)$ can be designed to control the isolated mode. Hence, the multiple-inout multiple-output control design problem becomes a SISO one with the challenge to find adequate blending vectors.

In Figure 2, the resulting feedback interconnection is depicted, where the modes $j = 1, .., n_j$ are subject to be controlled. Summarizing the input and output blending vectors in $K_u = [k_{u,1} \ \cdots \ k_{u,n_j}]$ and $K_y = [k_{y,1} \ \cdots \ k_{y,n_j}]$, the overall controller is

$$K(s) = K_u C(s) K_y^T,$$

where the SISO controllers are collected on the diagonal of $C(s) = \operatorname{diag}\left(c_1(s), \cdots, c_{n_j}(s)\right)$.

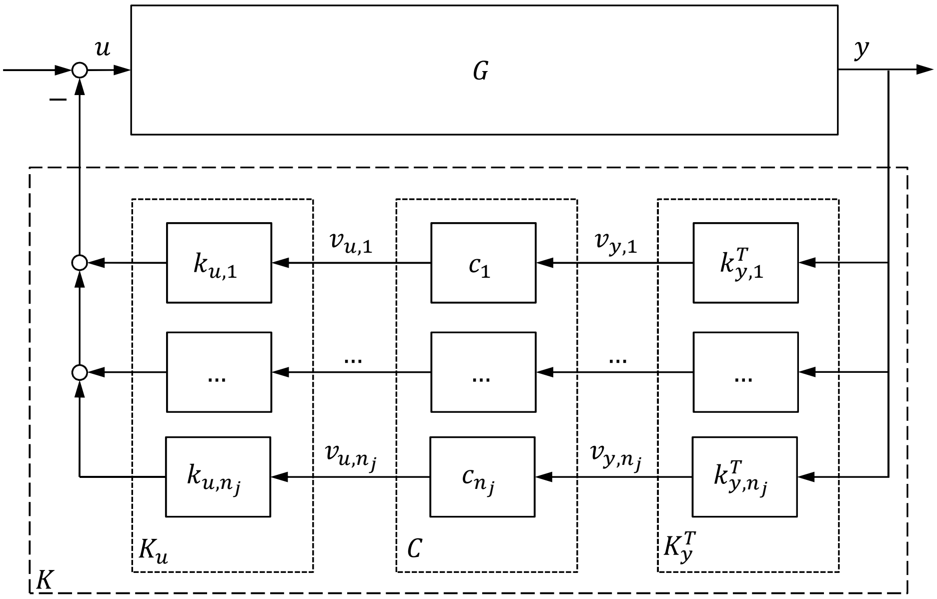

Figure 2. Closed-loop interconnection of plant G with flutter controller K, output blending matrix K_y, input blending matrix K_u, and controller C.

2.2. Maximization of Modal Controllability and Observability

The goal addressed herein is to find blending vectors which yield a maximum modal controllability and observability in terms of the $\mathcal{H}_2$ norm. [illegible]s requires a joint design of the input and output blending vectors as controllability and observability cannot regarded independent of each other. Furthermore, a sufficient mode decoupling needs to be considered in order to enable the proposed SISO controller design. Eventually, the proposed method is extended to undamped and unstable modes, for which the $\mathcal{H}_2$ norm becomes infinite by definition.

The combined controllability and observability of an asymptotically stable mode $M(s) \in \{M_j(s)\}$ is quantified in terms of the $\mathcal{H}_2$ norm. Hence, the goal is to stay as close as possible to the original controllability and observability of the targeted mode when blending inputs and outputs with real-valued unit vectors k_u and k_y, respectively. This gives rise to quantify the loss of controllability and observability via the efficiency factor

$$\eta = \frac{\left\| k_y^T M(s) k_u \right\|_{\mathcal{H}_2}}{\| M(s) \|_{\mathcal{H}_2}}, \tag{3}$$

where $\eta \in [0\ 1]$ for $M(s)$ being fully controllable and observable. Based on that, a pair of input and output blending vectors is considered as $\mathcal{H}_2$-optimal when the efficiency factor η is maximized. The resulting optimization problem can hence be formulated as

$$\begin{array}{ll} \underset{k_u \in \mathbb{R}^{n_u}, k_y \in \mathbb{R}^{n_y}}{\text{maximize}} & \left\| k_y^T M(s) k_u \right\|_{\mathcal{H}_2} \\ \text{subject to} & \| k_u \|_2 = 1 \\ & \| k_y \|_2 = 1. \end{array} \tag{4}$$

In order to efficiently solve the nonlinear optimization problem (4), the findings in [12] are applied to the objective function (4) giving

$$\left\| k_y^T M(s) k_u \right\|_{\mathcal{H}_2} = |k_y^T M(j\omega_{\mathrm{n}}) k_u| \sqrt{\omega_{\mathrm{n}} \zeta}, \tag{5}$$

where the term $\sqrt{\zeta\omega_\mathrm{n}}$ is actually independent of the blending vectors. Hence, the original problem of maximizing the $\mathcal{H}_2$ norm can be turned into a problem of maximizing the magnitude of the complex scalar $k_y^T M(j\omega_\mathrm{n})k_u$. Computing this magnitude according to [12] and factoring out the real-valued blending vectors k_y and k_u, it can be written

$$|k_y^T M(j\omega_\mathrm{n})k_u| = \max_{\phi}\left(k_y^T F(\phi)k_u\right), \tag{6}$$

where $F(\phi) : \mathbb{R} \to \mathbb{R}^{n_y \times n_u}$ is defined as

$$F(\phi) = \Re(M(j\omega_\mathrm{n}))\cos\phi + \Im(M(j\omega_\mathrm{n}))\sin\phi. \tag{7}$$

Recalling that the actual goal is to find a maximum of (6) gives

$$\begin{aligned}\max_{k_u,k_y}\left|k_y^T M(j\omega_\mathrm{n})k_u\right| &= \max_{k_u,k_y}\max_{\phi}\left(k_y^T F(\phi)k_u\right)\\ &= \max_{\phi}\max_{k_u,k_y}\left(k_y^T F(\phi)k_u\right).\end{aligned} \tag{8}$$

In (8), the term

$$\max_{k_u,k_y}\left(k_y^T F(\phi)k_u\right) = \|F(\phi)\|_2 = \sigma_\mathrm{max} \tag{9}$$

can be directly computed for a given value of ϕ by applying a singular value decomposition (SVD) on

$$F(\phi) = U\Sigma V^T = \begin{bmatrix}k_{y,\mathrm{max}} & \bullet\end{bmatrix}\begin{bmatrix}\sigma_\mathrm{max} & 0\\ 0 & \bullet\end{bmatrix}\begin{bmatrix}k_{u,\mathrm{max}} & \bullet\end{bmatrix}^T, \tag{10}$$

where the placeholder $\bullet$ denotes a matrix of adequate size. In (10), both $U \in \mathbb{R}^{n_y \times n_y}$ and $V \in \mathbb{R}^{n_u \times n_u}$ are orthogonal matrices which are real-valued as $F(\phi)$ is also real-valued. Furthermore, $\Sigma \in \mathbb{R}^{n_y \times n_u}$ is a rectangular diagonal matrix with the singular values of $F(\phi)$ in descending order on its diagonal. Selecting only the largest singular value $\sigma_\mathrm{max} \in \mathbb{R}_{\geq 0}$, the corresponding input and output singular vectors $k_{u,\mathrm{max}} \in \mathbb{R}^{n_u}$ and $k_{y,\mathrm{max}} \in \mathbb{R}^{n_y}$ directly yield the input and output blending vectors which solve (9) for a given value of ϕ.

Finally, inserting (9) into (8), an equivalent formulation of the optimization problem (4) is given as

$$\max_{k_u,k_y}\left\|k_y^T M(s)k_u\right\|_{\mathcal{H}_2} \Leftrightarrow \max_{\phi}\|F(\phi)\|_2, \tag{11}$$

where the optimization variables $k_u \in \mathbb{R}^{n_u}$ and $k_y \in \mathbb{R}^{n_y}$ are constrained by $\|k_u\|_2 = 1$ and $\left\|k_y\right\|_2 = 1$ while $\phi \in \mathbb{R}$ is unconstrained. Solving $\max_\phi \|F(\phi)\|_2$ yields an optimal phase angle ϕ^* for which the $\mathcal{H}_2$-optimal blending vectors can be directly determined according to (10). Hence, the number of optimization variables is reduced from $n_u + n_y$ to a single one, or, in other words, the difficulty of finding a solution of (4) becomes independent of the actual number of inputs and outputs. Finally, the optimization problem (4) has been transformed to the numerically tractable problem (11). The latter can be easily solved using readily available numerical software tools, further discussed and described in [12].

3. Optimization-Based Control Design

The controller structures of the baseline controller are defined based on classical flight-mechanical considerations. The blending vectors in the active flutter control algorithm allow defining a generic, parametrized SISO controller structure to control the flutter modes. Thus, for both design tasks only the actual gains have to be selected. These controller gains are derived by solving one of the two robust control design problems specified herein. The presented model-based gain optimizations pose non-convex design problems which are solved using Matlab's `systune` routine based on non-smooth optimization techniques [8]. The software tools allow an intuitive definition of optimization criteria in the frequency domain (e.g., bandwidth) and in the time domain (e.g. rise time) as either minimization criteria (soft constraint) or as minimization constraint (hard criteria).

For the model based approach, a low order, Linear Parameter-Varying (LPV) model of the FLEXOP demonstrator has been derived in [13] via linearization and advanced model reduction techniques from the full aeroservoelastic model described in [9]. It serves as basis for the design herein. The derived LPV model of the form

$$G(\rho(t)) : \begin{array}{rcl} \dot{x}(t) & = & A(\rho(t))x(t) + B(\rho(t))u(t) \\ y(t) & = & C(\rho(t))x(t) + D(\rho(t))u(t) \end{array} \tag{12}$$

has the grid based representation

$$\mathcal{G} = \left\{ G_i \mid G_i = \left[\begin{smallmatrix} A_i & B_i \\ C_i & D_i \end{smallmatrix}\right], \begin{smallmatrix} A_i = A(\rho_i) & B_i = B(\rho_i) \\ C_i = C(\rho_i) & D_i = D(\rho_i) \end{smallmatrix} \right\}. \tag{13}$$

In (12) $G(\rho(t))$ is the LPV model depending on the parameter $\rho(t)$ with the state vector x, the input vector u, the output vector y, and the state space matrices $A(\rho)$, $B(\rho)$, $C(\rho)$, and $D(\rho)$. In (13) $\mathcal{G}$ defines the set of $i = 1, \ldots, n_m$ linear time invariant models on the n_m grid points. Thus, the model $G(\rho(t))$ is evaluated with the n_m constant parameter values ρ_i, giving the LTI models G_i with the space matrices A_i, B_i, C_i, and D_i. For the LPV model of the flutter demonstrator the scheduling parameter is the indicated airspeed, i.e., $\rho(t) = V_{\text{ias}}(t)$, in an interval between 32 m/s and 70 m/s. The derived low order model of the aircraft dynamics has a state dimension of $n_x = 35$. Second order actuator models, first order linear sensors model, and a first order LPV engine model with time delay are explicitly added to the design dynamics.

Depending on the variability of the aircraft dynamics to be considered for the underlying control design we distinguish two control design problems to be solved. In case of low variations in the aircraft dynamics over the aircraft velocity, the goal is to design a constant controller for the whole velocity range via a multi-model approach. Larger variations in the aircraft dynamics call for a scheduled controller design to achieve better performance.

3.1. Constant Controller Design

The multi-model, multi-objective optimization problem to derive constant gains of a predefined controller structure [14] is stated by

$$\begin{aligned} &\min_{\Lambda} \max_{i,j} f_j^{(i)}(\Lambda) \\ &\text{s.t.} \max_{i,k} g_k^{(i)}(\Lambda) < 1 \\ &\Lambda_{\min} < \Lambda < \Lambda_{\max}, \end{aligned} \tag{14}$$

where $f_j(\Lambda)$ are the $j = 1, \ldots, n_s$ posed soft constraints, and $g_k(\Lambda)$ are the $k = 1, \ldots, n_h$ hard constraints. The upper index (i) indicates that the constraints are evaluated for all $i = 1, \ldots, n_m$ models. The free controller gains k_l to be optimized, with $l = 1, \ldots, n_k$, are staked in the vector Λ and tuned over all models and are limited by the upper and lower bounds $\Lambda_{\min}$ and $\Lambda_{\max}$. The software normalizes the soft and hard constraints and applies non-smooth optimization techniques to solve the corresponding multi-objective problem [8].

3.2. Scheduled Controller Design

The scheduled controller design problem [15] is similar to the one presented in (14). The main difference is that the controller gains in K depend on the scheduling variables described in the vector π. This vector belongs to the bounded region $\Pi \in \mathcal{P}$, where $\mathcal{P}$ is the n_p-dimensional parameter space. The design problem is defined by

$$\begin{aligned} &\min_{\Lambda(\pi)} \max_{i,j} f_i^{(j)}(\Lambda(\pi)) \\ &\text{s.t.} \max_{i,k} g_k^{(i)}(\Lambda(\pi)) < 1 \\ &\Lambda_{\min} < \Lambda(\pi) < \Lambda_{\max}. \end{aligned} \tag{15}$$

To avoid the necessity to optimize over the multi-dimensional function space $\Lambda(\pi)$, the gains in $\Lambda(\pi)$ are restricted to polynomial basis functions of the parameters in π. For example, the l^{th} element of the vector $\Lambda(\pi)$ is described by

$$k_l = a_{0,l} + a_{1,l}\pi + \cdots + a_{n_o,l}\pi^{\circ n_o}, \tag{16}$$

where n_o defines the polynomial order of the basis function. The vectors $a_{m,l}$ with $m = 1, \ldots, n_o$ and $l = 1, \ldots, n_k$, are constant and have the size $1 \times n_p$. The notation $\circ$ is used to indicate that the exponent is used on each element of the parameter vector π. For the control designs herein the indicated airspeed is the only scheduling parameter of the controller, i.e., $\pi = V_{\text{ias}}$ and $n_p = 1$. Also, the scheduling parameter of the controller is equal to the parameter of the underlying LPV design model described in (13), i.e., $\rho = \pi = V_{\text{ias}}$.

3.3. Design requirements

The soft and hard design constraints f and g in (14) and (15), respectively, are defined using classical control objectives in the frequency and time domain. This includes desired bandwidth, robustness margins, overshoot, tracking error, rise time, maximum loop gains, and desired loop shapes. Another possibility used in this article is to provide a *reference model* and use the error between this reference model and the resulting dynamics as criteria to be minimized in either (14) or (15). Such a model matching setup provides an elegant way to get as close as possible to the desired dynamics over the whole parameter range.

4. Application to the FLEXOP Demonstrator

The presented approaches to optimally blend input and outputs in Section 2 and the optimization to determine the controller parameters in Section 3 are applied to the FLEXOP aircraft. The single-engined FLEXOP flutter demonstrator features a wing span of 7 m and is illustrated in Figure 1. The takeoff weight is typically 55 kg but can be increased by up to 11 kg of ballast. Two wing-sets are designed and manufactured for the aircraft. The first one features a rigid structure with a flutter speed far beyond the operational aircraft velocity. This wing-set is mainly used for basic flight testing and rigid model verification. The second wing-set, which is considered in the models in this article, is flexible and has two main flutter modes within the operational velocity range.

The rigid body motion of this aircraft is described by a standard nonlinear six-degrees-of-freedom flight mechanics model (e.g. [16]) in terms of translational velocities u, v, w and angular velocities p (roll), q (pitch), r (yaw) in the body-fixed frame. Orientation in the earth-fixed reference frame is described in terms of Euler angles Φ (bank), Θ (pitch), and Ψ (heading). The angles between body-fixed frame and wind axes are angle of attack α and side-slip angle β. The flight path is described with respect to earth by the path angle γ and the course angle χ. To describe the flutter phenomena, the structural dynamics from a reduced finite element model are coupled with aerodynamics derived via the doublet lattice method. This coupling to derive the aeroelastic model is achieved via splining. For the flexible wing-set the first flutter mode, i.e., symmetric bending mode, becomes unstable at around

52 m/s with 8 Hz, while the second one, i.e., the asymmetric bending mode, follows at 56 m/s with 7.3 Hz [10,18]. A detailed description of the aircraft modeling and its analysis is also provided in [9,17].

As control inputs the aircraft features four ruddervators on the aircraft's V-tail, two on the left ($\delta_{rv,l1}$, $\delta_{rv,l2}$) and two on the right side ($\delta_{rv,r1}$, $\delta_{rv,r2}$) as illustrated in Figure 3. These ruddervators are combining the functionalities of classical rudders and elevators. The symmetric deflections of the ruddervator correspond to classical elevator deflections, while asymmetric deflections exhibit rudder deflections. Additionally, the aircraft has four pairs of ailerons. The most outer pair ($\delta_{a,l1}$, $\delta_{a,r1}$) is used for flutter control while the most inner pair ($\delta_{a,l4}$, $\delta_{a,r4}$) is used as high lift devices during takeoff and landing. The inner two pairs ($\delta_{a,l2}$, $\delta_{a,r2}$, $\delta_{a,l3}$, $\delta_{a,r3}$) are used in the baseline control law to control the aircraft's roll motion. The actuators to steer the control surfaces are modeled as second order systems with rate and position limits to realistically reflect the actuator behavior. These models have been obtained through frequency-based system identification and data gathered on the various servos. The sensors of the aircraft are modeled as first order linear models including time delays. The aircraft is equipped with a 300 N jet engine [19], located on the fuselage back. A high fidelity, non-linear simulation model of the engine is available. Consequently, a simplified, control-oriented model has been developed and is considered in the controller design tasks. It features a dominant time delay of 1 s, a non-linear mapping from the engine's revolution-speed to thrust (and versa), and a rather slow second-order dynamic. In addition, a velocity dependent saturation limit is considered. It describes how the available thrust decreases with increased inflow speeds.

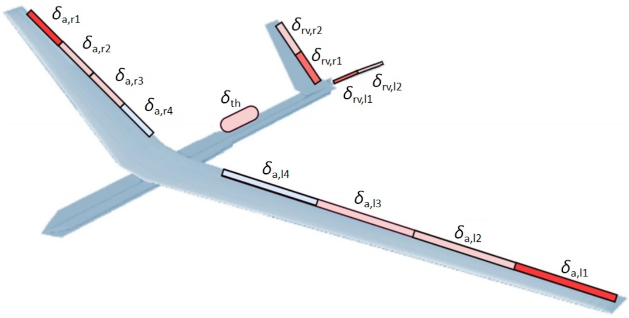

Figure 3. Control surface configuration.

4.1. Baseline Controller

Three different modes to control the aircraft are considered in the flight control system. These modes facilitate a stepwise augmentation of the aircraft during the flight test campaign:

(i) **Direct Mode**: The direct mode allows the pilot on the ground to bypass the flight control system. The only part active in the flight control computer is the mapping from the received remote-control signals to the commanded surface deflections. The pilot controls the pitch, roll and yaw axis directly via the aircraft's control surface deflections and its velocity via the thrust setting.

(ii) **Augmented Mode**: The augmented mode switches on basic augmentation for the pilot. Instead of directly controlling the surfaces the pilot inputs pitch- and roll-attitude commands. The side-slip angle is automatically regulated to zero, reducing the pilots need to control the yaw axis separately. Velocity control remains in direct control, i.e., the pilot controls the velocity via the thrust setting.

(iii) **Autopilot Mode**: In this mode the pilot fully delegates the aircraft control to the flight control system. Altitude, course angle, velocity and side-slip angle are automatically controlled. To fly along the defined test pattern, reference commands based on the aircraft position are generated in a navigation module.

The inner loops of the control system in roll, pitch and yaw provide the basis for the operational model (ii) and (iii). Mode (iii) is the core element of the autopilot adding the outer loops for course angle, altitude and speed control (autothrottle) as illustrated in Figure 4. Thus, a series of cascaded control loops is used to facilitate the control design task. As the cross-coupling between longitudinal and lateral axis is negligible, longitudinal and lateral control design is separated. Thrust commands δ_{th} which are transferred to an engine revolution command δ_ω via a nonlinear mapping and the elevator δ_e are the available actuators for longitudinal control. The available bandwidths for throttle and elevator differ considerably such that a combined control design does not promise any advantages. Thus, the reference V_{ref} for the indicated airspeed V_{ias} is controlled solely by the use of the throttle command δ_{th}. The elevator command δ_e is used to control the attitude in the inner loop and the vertical position of the aircraft in the outer loop. The pitch-attitude controller in the most inner feedback loop tracks the pitch-attitude (Θ), attenuates wind disturbances, and improves short period damping with the pitch rate (q) measurement as an auxiliary feedback signal. The cascaded outer loop establishes control of the altitude (H). Both controllers are scheduled with velocity (V_{ias}), indicated by $\nearrow$ in Figure 4, to achieve optimal performance over the required velocity range.

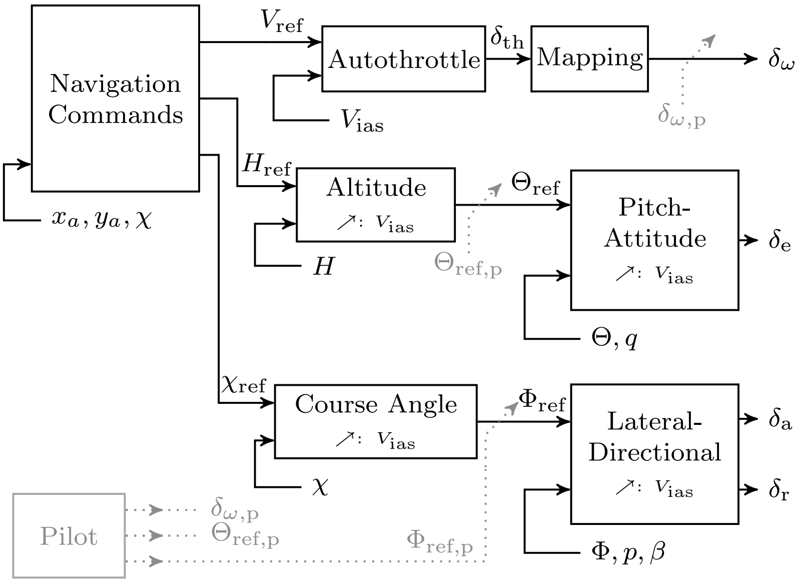

Figure 4. Control architecture for fully automated flight (mode (iii)), and augmented flight (mode (ii)), indicated in gray.

Lateral-directional control generates aileron (δ_a) and rudder commands (δ_r). The lateral-directional control problem is necessarily multivariable and requires the coordinated use of aileron command δ_a and rudder command δ_r. The most inner loop features roll-attitude (Φ) tracking, roll-damping augmentation via the roll rate (p), and coordinated turn capabilities, i.e. turns without side-slip, via feedback of the side-slip angle (β). The outer loop establishes control of the course angle (χ). Again, all controllers are scheduled with velocity to increase performance over the velocity range. Within the fully automated flight mode (iii) the reference signals for the velocity (V_{ref}), altitude (H_{ref}), and course angle (χ_{ref}) are provided by a dedicated navigation algorithm. It uses the GPS longitudinal and lateral position of the aircraft (x_a and y_a) as well as the current course angle (χ) to provide the commands. More details on the algorithm can be found in [20].

Structure wise, the control loops use scheduled elements of proportional-integral-derivative (PID) controller structures with additional roll-offs in the inner loops to ensure that no aeroelastic mode is excited by the baseline controller. A scheduling in dependence of the indicated airspeed V_{ias} is used to ensure an adequate performance over the velocity range from 32 m/s to 70 m/s. For the scheduling a first or second order polynomial in V_{ias} following (16) is applied. As an example the proportional gain $k_p = a_0 + a_1 V_{\mathrm{ias}} + a_2 V_{\mathrm{ias}}^2$ is depending quadratically on V_{ias} with the free parameters a_0, a_1, and a_2. These free parameters are directly included in the optimization problem (15). A comprehensive summary of the used controller structures for each cascaded loop is provided in Table 1, including the channel description in the controller architecture and the implemented scheduling.

Table 1. Summary of the control loops of the FLEXOP baseline flight control system with the inner loop functions (first part) and autopilot functions (second part).

Control Loop	Channel	Structure	Scheduling
Pitch Attitude Control	$(\Theta_{\text{ref}} - \Theta) \rightarrow \delta_e$	PI	2^{nd}-order polyn. in $V_{\textbf{ias}}$
Pitch Damping	$q \rightarrow \delta_e$	P	1^{st}-order polyn in $V_{\textbf{ias}}$
Roll Attitude Control	$(\Phi_{\text{ref}} - \Phi) \rightarrow \delta_a$	P	1^{st}-order polyn in $V_{\textbf{ias}}$
Roll Damping	$p \rightarrow \delta_a$	P	1^{st}-order polyn. in $V_{\textbf{ias}}$
Yaw Control	$\beta \rightarrow \delta_r$	PID	2^{nd}-order polyn. in $V_{\textbf{ias}}$
Autothrottle	$(V_{\text{ref}} - V_{\text{ias}}) \rightarrow \delta_{\text{th}}$	2 DOF-PID	none
Altitude	$(H_{\text{ref}} - H) \rightarrow \Theta_{\text{ref}}$	PI	2^{nd}-order polyn. in $V_{\textbf{ias}}$
Course Angle	$(\chi_{\text{ref}} - \chi) \rightarrow \Phi_{\text{ref}}$	PID	2^{nd}-order polyn. in $V_{\textbf{ias}}$

Note that these controller outputs δ_e, δ_a, and δ_r defer from the actual surface inputs to ease the actual control design task. Thus, they need to be transformed to physical actuator commands via an adequate control allocation. The FLEXOP aircraft has multiple control surfaces and features combined rudder and elevator surfaces (ruddervators) as depicted in Figure 3. The commands to the actuators of the two aileron pairs are determined by

$$\begin{array}{ccccc} \delta_{a,l2} & = & \delta_{a,l3} & = & 0.5\delta_a \\ \delta_{a,r2} & = & \delta_{a,r3} & = & -0.5\delta_a \end{array} \tag{17}$$

to generate the required differential aileron deflections for roll motion control. For the ruddervators superposition of the elevator command δ_e and the rudder command δ_r is applied by

$$\begin{array}{ccccc} \delta_{elev,l1} & = & \delta_{elev,l2} & = & \delta_e + 0.5\delta_r \\ \delta_{elev,r1} & = & \delta_{elev,r2} & = & \delta_e - 0.5\delta_r. \end{array} \tag{18}$$

Thus, symmetric deflections on the left and right of the ruddervators correspond to elevator commands while differential deflections establish rudder commands.

4.1.1. Parameter Tuning

With the baseline controller structure available, the next step is to tune the free parameters of the individual control loops. Following the ideas presented in Section 3, an individual optimization problem is set up for the tuning of each control loop. This results in six optimization problems to be solved and summarized in Table 2. Note that the proportional damping augmentations in roll and pitch are not tuned separately but included in the optimization problems of the corresponding tracking loops. For the inner loops a phase margin of at least 45 deg is demanded. As short period damping is relevant, a minimum of 0.6 is set as optimization constraint. For the roll motion a fast response time of 1 s with good tracking capabilities (steady state error of 0.1) is defined. For the coordinated turn capabilities via the side slip angle feedback a single constraint on the disturbance rejection gain is applied. For the outer loops an adequate frequency separation commonly used in a cascade controller design is applied. The outer loops for controlling attitude and course angle are designed to be five times slower than the inner loops, leading to a corresponding bandwidth or response time constraint. Finally, the auto-throttle is a little more involved due to the complex engine dynamics. Therefore, a model matching problem using the non-linear simulator is used which aims to minimize the recorded error between the desired and achieved response in the simulation.

Table 2. Overview of the six defined optimization problems with the number of free parameters and optimization criteria within the model based design procedure of the baseline controller.

Channel	Structure	Free Parameters	Criteria
Pitch Attitude Control	PI	8	Damping ration of 0.6 &
incl. Pitch Damping	P		Phase margin of 45 deg
Roll Attitude Control	P	4	Response time of 1s & steady state
incl. Roll Damping	P		Error of 0.1 & phase margin of 45 deg
Yaw Control	PID	9	Disturbance rejection gain
Auto-Throttle	2 DOF-PID	5	Model matching error
Altitude	PI	6	Bandwidth criterion
Course Angle	PID	9	Response time of 5 s

4.2. *Flutter Controller*

The main two flutter modes to be stabilized in order to expand the operational velocity range of the aircraft are the first symmetric bending and first asymmetric bending modes. The proposed $\mathcal{H}_2$-optimal blending approach described in Section 2 is applied to the FLEXOP demonstartor to blend the available output measurements and allow for a subsequent SISO controller design. The resulting blending vectors enable a straight forward design of two dedicated SISO control loops, one for each of the two unstable aeroelastic modes.

4.2.1. Input-Output Blending

hier schrieben wie blendign ausgelegt wird, randbedningungen, irgendwas Konkreted für das bsp. Wird blending gescheduled??? wenn nicht, argument über richtung,..... eventull ist das bode am auslegungspunkt und dann die auswertug am grid interessant?

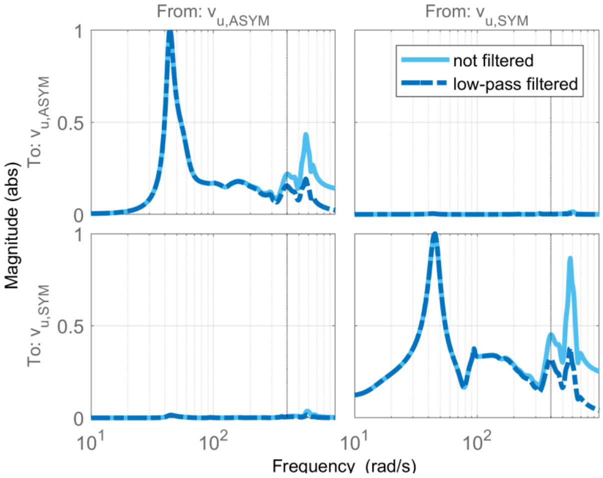

Figure 5. Bode-magnitude plots from virtual inputs to virtual outputs showing the decoupling of the two unstable flutter modes at an velocity of xx m/s.

4.2.2. Single-Input Single-Output Controllers

With the designed blending vectors it is possible to design two dedicated SISO controllers. In the ideal case, i.e. when the input and output blending fully decouples a conjugate complex mode $M(s)$ from its parent system $G(s)$, the input-output behavior from the blended input to the blended output can be described by

$$M(s) := k_y^T G(s) k_u = \frac{\alpha s + \beta}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \tag{19}$$

In (19), ω_n and ζ denote the natural frequency and relative damping of the decoupled mode. Furthermore, the scalar coefficients α and β result from blending inputs and outputs with the vectors k_u and k_y, respectively. In order to increase the damping of the decoupled mode (19), a SISO feedback controller needs to be designed. In general, the relative damping of a mode can be increased by a so-called direct velocity feedback, see [21]. For the decoupled mode (19), this can be achieved by filtering the blended output signal with

$$h(s) = \begin{cases} 1 & \text{if } \beta = 0 \\ s & \text{if } \alpha = 0 \\ \dfrac{\beta}{\alpha\omega_\mathrm{n}^2}s + 1 & \text{otherwise.} \end{cases} \tag{20}$$

In case $\beta = 0$, modal velocity is already measured by the blended output and hence, a proportional feedback directly allows increasing relative damping. In case $\alpha = 0$, the blended output represents modal deflection, which needs to be differentiated in order to obtain a velocity signal. In any other case, the proposed high-pass filter adds a zero such that a static feedback of the resulting signal directly affects the relative damping of the underlying mode.

If the target mode cannot be fully decoupled, which is generally the case, additional filtering may be required. For instance, an excitation of unmodeled system dynamics is avoided by limiting the frequency range of the controller. Therefore an additional first order roll-off filter is considered in the SISO controller. With the aforementioned considerations the form of the SISO controller can be defined by

$$c(s) = kh(s)\frac{1}{\tau s + 1}, \tag{21}$$

where $\tau \in \mathbb{R}^+$ is the time constant of the roll-off filter, and k is a static feedback gain. This feedback gain is the additional design freedom which is subject to tuning, allowing to adjust the control performance such that common constraints like actuator limitations or robustness criteria are met.

For the FLEXOP model the first bending symmetric bending and the first asymmetric bending modes need to be stabilized. Thus, two SISO controllers of the form (21) need to be designed. As a large velocity range needs to be considered, the gain k in (21) is scheduled as presented in (16) as linear function in dependence of the indicated airspeed, i.e., $k(V_\mathbf{ias}) = a_0 + a_1 V_\mathbf{ias}$. The parameters a_0 and a_1 are the free parameters when solving the optimization problems. The two optimization problems to design the two SISO controllers have the form (15). As explicit optimization criteria sufficient robustness margins and xxxx????xxxxx are demanded in the optimization. The two problems are again solved using non-smooth optimization techniques [15]. In Table 3 the resulting parameter values for the two SISO controller are summarized.

Table 3. Parameter values of the two SISO controllers stabilizing the two flutter modes.

Controller for	ω_n	α	β	τ	a_0	a_1
Symmetric Bending Mode	xx.xx	xx.xx	xx.xx	xx.xx	xx.xx	xx.xx
Asymmetric Bending Mode	xx.xx	xx.xx	xx.xx	xx.xx	xx.xx	xx.xx

Closing the two SISO loops stabilizes the two flutter modes as it is illustrated in the pole migration plot in Figure 6. The plot compares the open-loop poles in gray to the closed-loop pole depicted in color in dependence of the airspeed. Clearly visible is the unstable behavior, i.e., the crossing to the right half plain of the symmetric and asymmetric bending modes in the open-loop. With the two developed controller the symmetric flutter mode can be stabilized up to an airspeed of 65.5 m/s. The asymmetric mode is stabilized even beyond 70 m/s. Demanding additional single-loop robustness margins of 6 dB in gain and 45 deg in phase to the critical point, leads to an operational speed of about

60 m/s, This still results in an increase in allowable speed of more than 15 % compared to the open-loop flutter behavior of the aircraft. Also noticeable is that the other poles of the systems are not largely effected by the two introduced SISO controller. While a full decoupling is not achieved, the migration of the poles does not show any decrease in damping and therefore is acceptable.

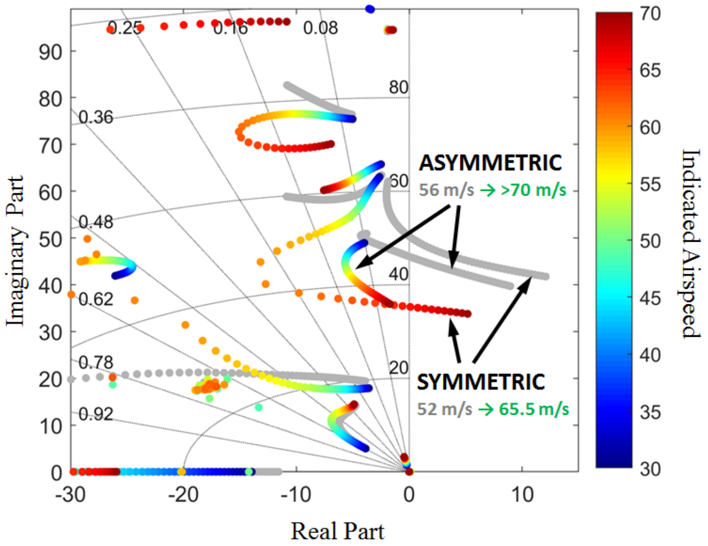

Figure 6. Comparison of the open- and closed-loop poles. Only the positive part imaginary axis is depicted for a better overview.

The linear analysis results discussed in this section provide an initial verification of the controller. The next mandatory step on the way to the implementation of the control algorithms on the aircraft is to test them in a non-linear simulation environment of the aircraft to gather further insight into the performance and robustness of the developed algorithms.

5. Verification

The developed flight control system including the basic augmentation to autonomously fly the aircraft and the flutter control algorithm are verified in a non-linear simulator and a summary of the important results is provided in this section. The baseline controller shall work for both wing-set configurations, i.e., the rigid and the elastic one. A detailed description of the baseline control architecture and a coherent analysis with the rigid wing set is provided in [20]. The advancement herein is to fly the aircraft with the flexible wing set beyond the flutter speeds of approximately 52 m/s and 54 m/s. Thus, the findings from the linear analysis in Section 4 which indicate an extension of the flutter phenomena from 54 m/s to 66 m/s for the asymmetric bending mode and from 52 m/s to 70 m/s for the symmetric bending mode are verified. Therefore, a simulation based verification using the developed high fidelity simulator presented in [9] is performed. To provide some insight in the open-loops flutter behavior, i.e., without active flutter control, the aircraft is accelerated from its trim condition at 38 m/s to 50 m/s. From there on the speed is increased by 4 m/s to enter the flutter region.

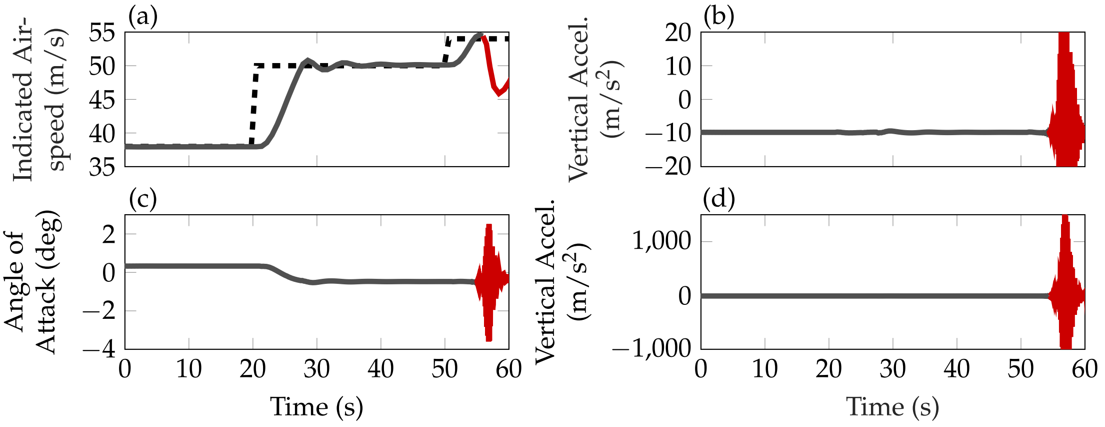

Figure 7. Simulation results for an acceleration scenario for indicated airspeed (a), angle of attack (c), accelerations on the wing root (b), and on the wing tip (d) without flutter controller. The flight in the stable regime is indicated in gray (▬), the unstable situation in red (▬).

Figure 7 depicts the aircraft speed in the diagram (a), the aircraft's angle of attack in diagram (c), the vertical accelerations on the wing root of the left wing in diagram (b), and the vertical accelerations on the left wing tip in diagram (d). The first step in the reference airspeed happens at 20 s simulation time. The aircraft accelerates, leading to a reduction of the required angle of attack (c) to hold the altitude, and reaches the commanded speed of 50 m/s. At 50 s simulation time the reference speed is increased by 4 m/s. The aircraft reaches the flutter speed and the wings start to oscillate, indicated in red (▬) in the diagrams of Figure 7. This leads to high accelerations on the wings, which is depicted in the diagrams (b) for the left wing root and in diagram (d) for the left wing tip. In reality, the aircraft would have been lost at this point, but the resulting non-linear behavior is not covered by the simulation. In Figure 8 the same scenario is simulated with the flutter controller enabled. The velocity is increased step-wise until flutter occurs. The aircraft is stabilized up to an indicated airspeed of about 65 m/s, as predicted by the linear analysis. The accelerations on the wing are kept close to their trim conditions by the flutter controller. After initiating the velocity step from 64 m/s to 66 m/s the first symmetric bending mode becomes unstable. This instability is indicated in the diagrams of Figure 8 by the changing the line color to red.

Next, the aircraft is simulated on the predefined flight test pattern. For the description of the pattern we are assuming without loss of generality that the north direction is equal to the y-axis of the defined coordinate system. The main waypoint to be tracked is chosen before the turn of the actual flutter test. Thus, the inbound leg is dedicated to the waypoint tracking to ensure a uniform start

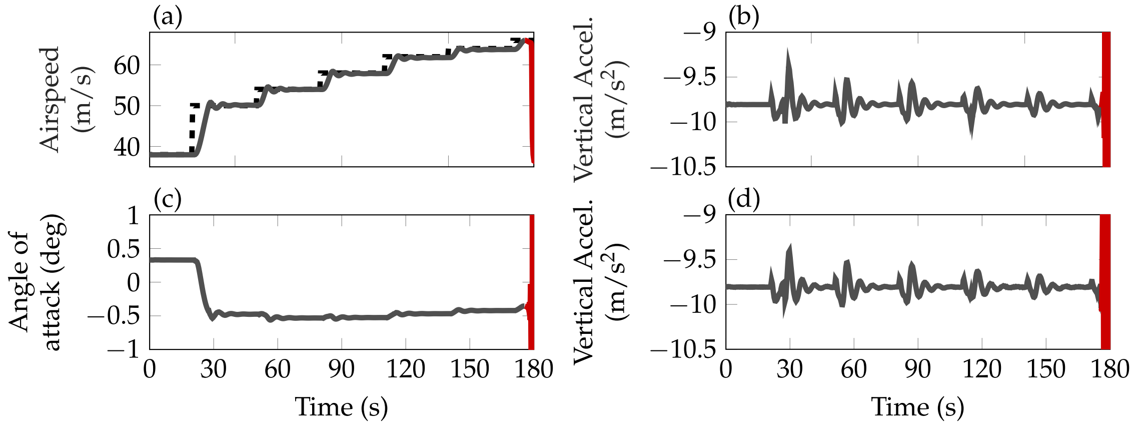

Figure 8. Simulation results for an acceleration scenario for indicated airspeed (a), angle of attack (c), accelerations on the wing tip (b) and the wing root (d) with active flutter controller. The flight in the stable regime is indicated in gray (▬), the unstable situation in red (▬).

of the outbound turn. After the outbound turn the aircraft reaches the outbound leg, on which the aircraft velocity is increased to test the active flutter control. On the last part of the outbound leg the aircraft is decelerated to avoid flying the turns above the open-loop flutter speed. Overall, this results in four main segments, for which the reference signals need to be provided. To generate the reference signals a state-machine with sub-tasks, which are selected based on switching criteria, is implemented. This state-machine together with the presented baseline and flutter controller allow to navigate the aircraft around the test pattern fully autonomously. In Figure 9 relevant flight parameters during a simulated flight of 200 s along the described test pattern is depicted. In diagram (a) the longitudinal and lateral position of the aircraft is shown. The 200 s flight time correspond to approximately two laps on the pattern. On the test leg of each lap the aircraft is brought into the open-loop flutter regime. The indicated airspeed (▬) is depicted in diagram (c) together with it reference command (┅). The pattern is flown with a nominal speed of 38 m/s. On the test leg the speed is increased to 54 m/s in the first lap and 58 m/s in the second lap. The altitude is maintained by the control system at 348 m shown in diagram (b). The visible spikes at about 65 s and 160 s in the altitude and airspeed are due to vertical wind gusts simulated on the test leg to verify the control system's robustness against disturbances. The control system maintains the angle of side-slip around zero during the whole flight including the turn maneuvers. Thus, the baseline controller is able to adequately track the demanded values in the relevant flight parameters.

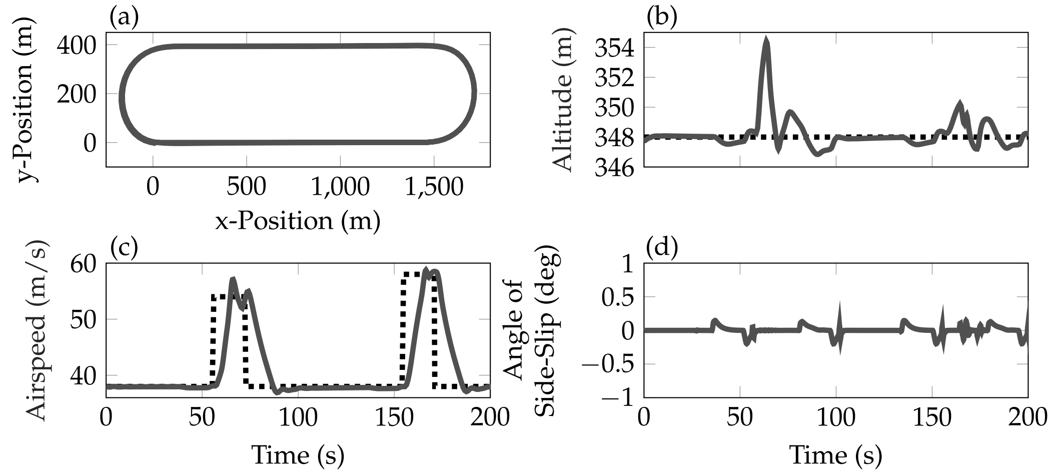

Figure 9. Simulated aircraft position during three laps on the test track in (a), reference and aircraft altitude in (b), reference and aircraft velocity in (c), and angle of attack in (d).

In Figure 10 the corresponding control surface deflections of the ailerons and the ruddervators are depicted. In diagram (a) the deflections of the two left ($\delta_{a,l2}$, $\delta_{a,l3}$) and the two right ($\delta_{a,r2}$, $\delta_{a,r3}$) aileron pairs used to control the roll motion of the aircraft are shown. Due to selected aileron control allocation described in (17), the deflections are identical for the two left and two right ailerons, respectively. A moderate deflection of $\approx \pm 3$ deg is necessary to bring the aircraft to the desired bank angle of 45 deg during the turns and back to 0 deg after the turns. In diagram (b) the deflections of the ruddervators are shown, which are the result of the superposition of the rudder and elevator commands from the baseline controller, following from the control ruddervator allocation in (18). Again, the deflections on each side are identical so that only two lines are visible. The deflections of about ± 3 deg around the trim value of ≈ 8 deg are required to ensure the coordinated turn without side slip angle ($\beta \approx 0$) and to track the reference altitude during the turns and on the test leg. Diagram (c) depicts the deflections of the outer aileron pair ($\delta_{a,l4}$, $\delta_{a,r4}$) between 62 s and 71 s simulation time, i.e., when the aircraft is flown within the open-loop flutter regime. The depicted deflections are required to stabilize the aircraft. Only a single line is visible as in the velocity range up to 58 m/s the symmetric bending mode is the predominant, unstable mode. Diagram (d) depicts the deflections of the outer aileron pair between 164 s and 173 s simulation time, i.e., when the aircraft is accelerated to 58 m/s reference speed. The required deflections for stabilizing the aircraft are comparable to the ones at 54 m/s airspeed.

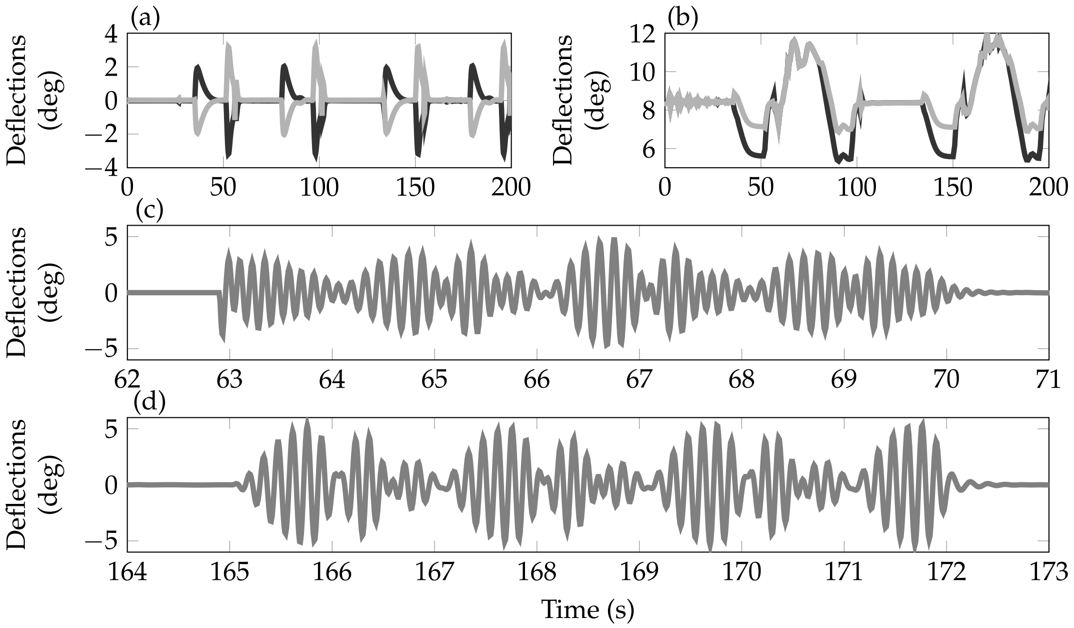

Figure 10. Control surface activity during the simulated flight for the two left and two right ailerons controlling the roll motion in (a), the two left and right ruddervators in (b), and the two ailerons stabilizing the two flutter in (c) and (d) for the phases when the flutter controller is activated.

Figure 11 depicts the resulting accelerations on the tip of the left wing in the vertical direction (━━) and horizontal direction (━━) , where the latter is directed into the flying direction. The flutter controller is able to stabilize both flutter modes during the whole test, leading to deviations of $\pm 15\,\mathrm{m/s^2}$ from the trim point for the vertical accelerations and $\pm 5\,\mathrm{m/s^2}$ for the horizontal accelerations. To summarize the performed verification campaign, the developed control system including the baseline and flutter controllers is able to satisfactorily navigate along the defined flight test pattern and stabilize the flutter as predicted with the linear analysis also in wind scenarios. These results provide confidence that the developed system will also work satisfactorily during the real flight tests.

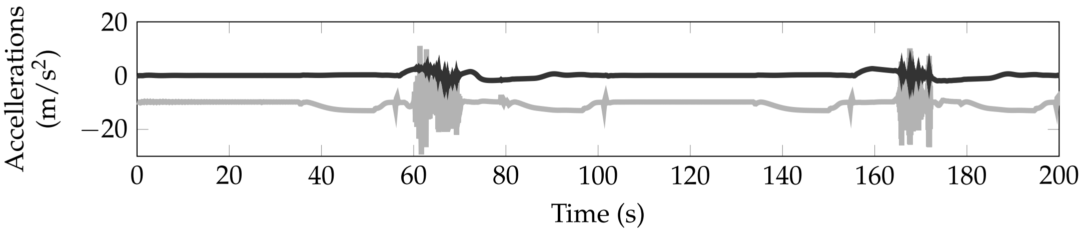

Figure 11. Simulated vertical (▬) and horizontal (▬) accelerations on the left wing tip.

6. Conclusions

In this article the design of a flight control system for the FLEXOP flutter demonstrator, a highly flexible aircraft designed to test active flutter control algorithms, has been presented. The control system includes a baseline controller to autonomously navigate the aircraft and an active flutter control algorithm, to stabilize the unstable flutter modes and thereby extend the aircraft's operational range. The baseline controller features a classical cascaded controller structure with various scheduled PID control loops. The flutter controller is based on a novel optimal blending of the available inputs and outputs to structurally isolate the flutter modes in fictitious feedback signals. This allows designing simple SISO controllers for each of the modes, which has been carried out in detail in the paper. The controller gains for both, the baseline and the flutter controller, have been selected in a model based optimization setup using robust control techniques. Therefore, two dedicated optimization setups have been proposed and solved via robust control techniques. The closed-loop has been verified using a high fidelity non-linear simulator of the test aircraft in various scenarios and results have been reported. The results show that with the developed control system it is possible to extend the aircraft's operational range beyond the open-loop flutter speed.

Author Contributions: Conceptualization: D. Ossmann, M. Pusch and T. Luspay; methodology and software: M. Pusch and T. Luspay; validation: D. Ossmann and T. Luspay; writing—original draft preparation: D. Ossmann; writing—review and editing: D. Ossmann, M. Pusch and T. Luspay; visualization: M. Pusch and D. Ossmann; supervision: D. Ossmann.

Funding: This work was performed in the framework of the European Union's Horizon 2020 research and innovation programme and is part of the Flutter Free FLight Envelope eXpansion for ecOnomic Performance improvement (FLEXOP) project with the grant agreement No 636307.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Theis, J.; Pfifer, H.; Seiler, P. Robust control design for active flutter suppression. Proc. of Atmospheric Flight Mechanics Conference, AIAA SciTech Forum; AIAA: San Diego, CA, 2016. doi:10.2514/6.2016-1751.
2. Danowsky, B.P. Flutter Suppression of a Small Flexible Aircraft using MIDAAS. Proc. of Atmospheric Flight Mechanics Conference, AIAA AVIATION Forum; AIAA: Denver, CO, 2017. doi:10.2514/6.2017-4353.
3. Danowsky, B.P.; Kotikalpudi, A.; Schmidt, D.; Regan, C.; Seiler, P. Flight Testing Flutter Suppression on a Small Flexible Flying-Wing Aircraft. Proc. of Multidisciplinary Analysis and Optimization Conference, AIAA AVIATION Forum; AIAA: Atlanta, GA, 2018. doi:10.2514/6.2018-3427.
4. Stahl, P.; Sendner, F.M.; Hermanutz, A.; Rößler, C.; Hornung, M. Mission and Aircraft Design of FLEXOP Unmanned Flying Demonstrator to Test Flutter Suppression within Visual Line of Sight. Proc. of 17th AIAA Aviation Technology, Integration, and Operations Conference, AIAA AVIATION Forum; AIAA: Denver, CO, 2017. doi:10.2514/6.2017-3766.
5. Roessler, C.; Stahl, P.; Sendner, F.; Hermanutz, A.; Koeberle, S.; Bartasevicius, J.; Rozov, V.; Breitsamter, C.; Hornung, M.; Meddaikar, Y.M.; Dillinger, J.; Sodja, J.; Breuker, R.D.; Koimtzoglou, C.; Kotinis, D.; Georgopoulos, P. Aircraft Design and Testing of FLEXOP Unmanned Flying Demonstrator to Test Load Alleviation and Flutter Suppression of High Aspect Ratio Flexible Wings. Proc. of AIAA Scitech Forum; AIAA: San Diego, CA, 2019. doi:10.2514/6.2019-1813.

6. Brockhaus, R.; Alles, W.; Luckner, R. *Flugregelung*, 3 ed.; Springer Verlag Berlin Heidelberg, 2011.
7. Stevens, B.L.; Lewis, F.L.; Johnson, E.N. *Aircraft control and simulation: dynamics, controls design, and autonomous systems*; John Wiley & Sons, 2015.
8. Apkarian, P.; Noll, D. Nonsmooth H_∞ Synthesis. *IEEE Transactions on Automatic Control* **2006**, *51*, 71–86. doi:10.1109/TAC.2005.860290.
9. Wüstenhagen, M.; Kier, T.; Pusch, M.; Ossmann, D.; Meddaikar, M.Y.; Hermanutz, A. Aeroservoelastic Modeling and Analysis of a Highly Flexible Flutter Demonstrator. Proc. of Atmospheric Flight Mechanics Conference, AIAA AVIATION Forum; AIAA: Atlanta, GA, 2018. doi:10.2514/6.2018-3150.
10. Meddaikar, Y.; Dillinger, J.; Klimmek, T.; Krueger, W.; Wuestenhagen, M.; Kier, T.; Hermanutz, A.; Hornung, M.; Rozov, V.; Breitsamter, C.; Alderman, J.; Takarics, B.; Vanek, B. Aircraft Aeroservoelastic Modelling of the FLEXOP Unmanned Flying Demonstrator. Proc. of AIAA Scitech Forum; AIAA: San Diego, CA, 2019. doi:10.2514/6.2019-1815.
11. Skogestad, S.; Postlethwaite, I. *Multivariable Feedback Control – Analysis and Design*; John Wiley & Sons, 2005.
12. Pusch, M.; Ossmann, D. $\mathcal{H}_2$-optimal Blending of Inputs and Outputs for Modal Control. *Submitted to Transaction of Control System Technology* **2018**.
13. Luspay, T.; Péni, T.; Vanek, B. Control oriented reduced order modeling of a flexible winged aircraft. Proc. of IEEE Aerospace Conference; IEEE: Big Sky, MT, 2018. doi:10.1109/AERO.2018.8396496.
14. Apkarian, P.; Gahinet, P.; Buhr, C. Multi-model, multi-objective tuning of fixed-structure controllers. Proc. of European Control Conference; IEEE: Strasbourg, France, 2014. doi:10.1109/ECC.2014.6862200.
15. Apkarian, P.; Dao, M.N.; Noll, D. Parametric Robust Structured Control Design. *IEEE Transactions on Automatic Control* **2015**.
16. McRuer, D.T.; Graham, D.; Ashkenas, I. *Aircraft Dynamics and Automatic Control*; Vol. 740, Princeton University Press, Princeton, NJ, 2014.
17. Rozov, V.; Hermanutz, A.; Breitsamter, C.; Hornung, M. Aeroelastic Analysis of a Flutter Demonstrator with a very Flexible High-Aspect-Ratio Swept Wing. Proc. of International Forum on Aeroelasticity and Structural Dynamics; IFASD: Como, Italy, 2017.
18. Iannelli, A.; Marcos, A.; Lowenberg, M. Aeroelastic modeling and stability analysis: A robust approach to the flutter problem. *International Journal of Robust and Nonlinear Control* **2018**, *28*, 342–364. doi:10.1002/rnc.3878.
19. Sendner, F.M.; Stahl, P.; Rößler, C.; Hornung, M. Designing an UAV Propulsion System for Dedicated Acceleration and Deceleration Requirements. Proc. of 17th AIAA Aviation Technology, Integration, and Operations Conference, AIAA AVIATION Forum; AIAA: Denver, CO, 2017. doi:10.2514/6.2017-4105.
20. Ossmann, D.; Luspay, T.; Vanek, B. Baseline Flight Control System Design for an Unmanned Flutter Demonstrator. Proc. of IEEE Aerospace Conference; IEEE: Big Sky, MT, 2019.
21. Preumont, A. *Vibration control of active structures*; Vol. 2, Springer, 1997.