CEAS-GNC-2024-054

EuroGNC

Bristol, UK

June 11th-13th 2024

Flight Testing of Real-Time Model Predictive Flight Control for Unmanned Flexible Aircraft

Benjamin Herrmann Research Engineer, Institute of Aircraft Systems Engineering, Hamburg University of Technology, Nesspriel 5, 21129 Hamburg, Germany. benjamin.herrmann@tuhh.de

Leif Rieck Research Engineer, Institute of Aircraft Systems Engineering, Hamburg University of Technology, Nesspriel 5, 21129 Hamburg, Germany. leif.rieck@tuhh.de

Frank Thielecke Professor, Institute of Aircraft Systems Engineering, Hamburg University of Technology, Nesspriel 5, 21129 Hamburg, Germany. frank.thielecke@tuhh.de

ABSTRACT

This paper describes the design, real-time implementation, and flight testing of a model predictive flight controller for the longitudinal and lateral-directional control of a slightly flexible 25kg unmanned aircraft. The controller is responsible for stabilization of the aircraft and tracking of reference commands. Reduced-order linear prediction models are derived from a high-fidelity nonlinear model of the flexible aircraft at different points in the flight envelope. To allow for the consideration of aeroelastic effects throughout the control design process, relevant structural modes are included in these models. From the set of prediction models, multiple linear model predictive controllers are designed off-line, applying the MPC formulation using a velocity-form model for offset-free tracking. During on-line operation, they are scheduled similar to classical gain-scheduling. The number of decision variables is effectively reduced by approximating the incremental control trajectory with Laguerre functions. An observer is designed to estimate non-measureable states. The predictive flight controller is finally validated in multiple flight test experiments with predefined tracking scenarios. The controller is shown to provide very good tracking performance even in the presence of wind and gust disturbances.

Keywords: Model Predictive Flight Control; Real-Time Implementation; Flight Testing; Flexible UAV

Nomenclature

A, B, C, D	=	System state-space matrices
$\bar{A}, \bar{B}, \bar{C}$	=	Augmented state-space matrices
A_L	=	Laguerre state transition matrix
a	=	Laguerre pole
c	=	Constant
c_m	=	Laguerre coefficient of the mth Laguerre function
d	=	Vector of translational strip deformations, m
d_i	=	Elastic translational deformation vector of mass element of a strip, m
dm_i	=	Mass element of a strip
e	=	Tracking error

F^{ext}, M^{ext}	=	Vector of external forces and moments, N and N·m
G	=	Gravitation acceleration vector, m/s^2
G_u, G_u^L	=	Selector matrix of input constraints in original and Laguerre formulation
$G_{\Delta x}$	=	Selector matrix of incremental state constraints
g	=	Linear cost term
H	=	Hessian matrix
h_u	=	Vector of control input limits
$I_{n \times n}$	=	Unit matrix with dimension $n \times n$
J	=	Inertia tensor, kg·m^2
J_k	=	Cost function
K_k	=	Kalman gain
L_i	=	Vector/matrix of Laguerre functions evaluated at time step i
l_m	=	mth Laguerre function
m	=	Aircraft mass, kg
N, N_c	=	Prediction and input constraint horizon
N_L	=	Number of Laguerre functions
n_s, n_a, n_f	=	Number of strips, ailerons, flaperons
n_u, n_x, n_y	=	Number of inputs, states, outputs
n_z	=	Load factor
n_ρ	=	Number of scheduling parameters
O_{B^R}, O_I	=	Body reference frame and inertial frame
P	=	Terminal penalty
$\mathcal{P}$	=	Compact subset of admissible values for scheduling parameters
p_i	=	Position of mass element of a strip relative to O_{B^R}, m
Q_1, Q_2	=	Incremental state and tracking error weighting matrices
Q_η	=	Generalized force, N·m
$\tilde{Q}, \tilde{R}$	=	Horizon weighting matrices
R	=	Input weighting matrix
R_L	=	Weighting matrix of Laguerre coefficients
r	=	Reference vector
r_i	=	Position vector of mass element of a strip, m
r_0	=	Position vector of the origin of O_{B^R}, m
s_i	=	Undeformed position vector of mass element of a strip relative to O_{B^R}, m
$T_{B^R I}$	=	Transformation matrix from O_I to O_{B^R}
T_s	=	Sampling time, s
$\mathcal{U}$	=	Input constraint set
ΔU	=	Incremental control trajectory vector
u	=	Control input vector
Δu	=	Incremental control input vector
V	=	Translational velocity vector, m/s
V_A	=	Airspeed, m/s
w	=	Sink rate in O_I, m/s
X^a	=	Augmented state trajectory vector
x	=	State vector
x^a	=	Augmented state vector
$x_{B^R}, y_{B^R}, z_{B^R}$	=	Coordinates of body reference frame, m
x_I, y_I, z_I	=	Coordinates of inertial frame, m
$\bar{x}, \bar{u}, \bar{y}$	=	Trim condition vectors
Δx	=	Incremental state vector
$\Delta \hat{x}$	=	Estimated incremental state vector

y	=	Output vector
β	=	Angle of sideslip, rad
$\delta_e, \delta_a, \delta_f, \delta_r, \delta_t$	=	Deflections (elevator, aileron, flaperon, rudder) and throttle position, rad or %
γ	=	Vector of Laguerre coefficients
η	=	Modal coordinate
μ	=	Generalized mass, kg·m^2
Ω_L	=	Matrix of Laguerre functions
ω	=	Angular velocity vector, rad/s
ω_n	=	Undamped natural frequency, rad/s
ρ	=	Scheduling parameter vector
ξ	=	Modal damping ratio
Φ, Γ	=	Prediction matrices
Φ_j	=	Mode shapes
ϕ	=	Bank angle, rad
$0_{n\times m}$	=	Zero matrix with dimension $n \times m$
$(\cdot)\|_{B^R}$	=	$(\cdot)$ expressed in O_{B^R}

1 Introduction

The efforts to enhance aircraft efficiency by reducing structural weight and improving the aerodynamic performance have led to consider aircraft with lighter structures and high aspect ratio wings. These wing structures are usually subject to larger loads. They are also more flexible which causes larger in-flight deformations and decreases the separation of the rigid-body and aeroelastic dynamics in the frequency domain. From a flight control perspective, this poses various challenges since undesired coupling of rigid-body dynamics and elastic deformation can occur through aerodynamic forces and control systems, and render the traditional strategy of treating rigid-body and aeroelastic dynamics separately during control design obsolete [1, 2]. Therefore, new control approaches are needed that are capable of integrating multiple requirements such as stabilization, maneuver demand, and load control.

One promising method for this task is model predictive control (MPC), also known as receding horizon control (RHC). By performing a constrained optimization over a finite horizon each sampling time interval, the control inputs are calculated based on the predictions of a known model of the system. Its ability to control multi-input multi-output (MIMO) systems, capability to explicitly handle pointwise-in-time constraints, as well as its advantageous performance with nonlinear systems makes it particularly interesting for flight control applications. The use of model predictive control in flight control has been previously studied, e.g. in [3–10]. Many of these works consider fault-tolerant systems by reconfiguration of the controller in case of actuator fault or failure [5–7], while others focus on path planning problems [9]. A popular approach has also been to use MPC in systems for envelope protection, maneuver limiting, or load alleviation, as these type of problems can suitably be formulated as constrained control problems. Two main control architectures have proven effective for these applications: reference governors [11–13], where the MPC alters the reference input of inner loop controllers such that the system output remains within certain limits, and integrated MPC concepts [14–19], where the MPC is responsible for both command tracking and enforcing secondary control objectives such as load alleviation. In [11], the authors use a model-predictive-control-like scheme to design a reference governor for enforcing maneuver limits of a fighter aircraft. In [12, 13], a linear MPC-based reference governor is proposed for maneuver load alleviation (MLA) of a flexible aircraft. Load constraints are enforced by adapting the inner loop controller's reference commands while also manipulating additional control surfaces specifically assigned to perform MLA. A well-known application of linear MPC for integrated flight control is presented in [14], where the authors address the gust load alleviation (GLA) of very flexible aircraft. Aeroservoelastic effects are accounted for throughout the control design process by incorporating flexible modes in the linearized state-space representation of the aircraft. The work was later extended in [15] to enhance the controller's performance and robustness by introducing an additional feedback loop in the model prediction. Other examples of integrated MPC concepts for gust load alleviation of flexible aircraft include [16, 17] in combination with light detection and ranging (LIDAR), [18] applying linear MPC in comparison to an equivalent linear quadratic regulator (LQR), and [19] comparing linear and nonlinear MPC. Although existing research has demonstrated that model predictive control can improve the closed-loop performance in different flight control applications, one of the main challenges remains real-time implementation [20]. It results from the high computational load that is required to repeatedly solve a constrained optimization problem each sampling time interval. Concerning flexible aircraft, additional challenges are posed by the high orders of aircraft models and long prediction horizons that are needed to sufficiently capture rigid-body and aeroelastic dynamics. Different methods to reduce the computational complexity have been proposed, such as in [13] by applying move blocking of inputs to reduce the number of decision variables or in [21] by aggregating constraints using Kreisselmeier-Steinhauser (KS) function. However, most of the existing works are sole simulation studies and only few experimental results, e.g. [22], or even in-flight demonstrations, e.g. [23, 24], are available.

In view of the potential and challenges of MPC in flight control applications, the objective of this work is the design, real-time implementation, and flight testing of a model predictive flight controller for the longitudinal and lateral-directional control of a slightly flexible 25kg unmanned aircraft. The controller is responsible for the stabilization of the aircraft and tracking of reference commands. Aeroelastic effects are taken into account in the predictions by including relevant structural modes in the prediction models. To achieve a reduction of computation time and thus enable the real-time implementation of the controller, an approach to approximate the control trajectory with Laguerre functions [25, 26] is employed. They provide a flexible framework to model control trajectories with a limited number of functions. In this way, the number of decision variables of the optimization problem is significantly reduced. Successful demonstration of the controller in flight was achieved in September 2023 using the G-Flights Dimona aircraft. The paper is organised as follows. First, the test aircraft, nonlinear model, and linear prediction models are introduced in Section 2, followed by the design of the controller in Section 3. Finally, the tuning, real-time implementation, and flight testing of the controller is presented in Section 4.

2 Test Aircraft and Model Formulations

The aircraft used for flight testing of the controller is the slightly flexible G-Flights Dimona, depicted in Figure 1. It is a 25kg unmanned replica of the *HK36 Super Dimona* at a scale of 1:3. An electric motor drives the aircraft with a maximum power of 4kW. Battery capacity allows for flight times between 20 and 30 minutes. It has a length of 2.4m and features custom spar, rib, and foil manufactured wings with increased span and flexibility. With a total span of 5.4m and a total surface area of 1.68m^2, the wing aspect ratio results to 17.35. At 2g, the wing tip deflection reaches approx. 10% of the half span. The aircraft is equipped with multiple trailing edge control surfaces on the wings. Each wing comprises two ailerons and two flaperons. They are designed as multi-functional control surfaces, providing lateral-directional control, high lift control, and future load control. The aircraft is further equipped with a rudder for lateral-directional control and two elevators for longitudinal control and future load control.

The aircraft can either be controlled by a safety-pilot via remote control or by a real-time capable flight control computer. A safety switch ensures that the pilot can take over control at any time. The flight control computer, a *MicroAutoBox II* from dSpace, serves as the central component for hosting guidance, navigation, and control (GNC) applications and for processing measurements. Further, it issues control outputs to the control surface servo actuators during automatic flight. The on-board avionics are completed by several sensors, radio equipment, and a separate computer for data recording. An industry-grade high-precision inertial navigation platform (INS), supported by dual-antennas, is used to measure GPS position, heading, attitude, velocities, and accelerations. An increased position accuracy is

Fig. 1 G-Flights Dimona

achieved with a third antenna on ground providing Differential GPS (DGPS) correction. Airdata such as airspeed, angle of attack, angle of sideslip, static air pressure, and air temperature is measured by three in-house developed five-hole-probes [27] located at the left wing, right wing, and vertical tail. Additional inertial measurement units (IMUs) and strain gauges for shear, bending, and torsion are integrated into the aircraft structure to measure structural dynamics and loads. They are distributed at relevant stations along the fuselage, empennage, and wings. Data distribution between the sensors and computers is handled via Controller Area Network (CAN) and Ethernet. A telemetry link enables a flight test engineer on ground to control and monitor the aircraft during flight experiments. The overall concept and main parts of the aircraft instrumentation are described in detail in [28, 29].

2.1 Nonlinear Flexible Aircraft Model

For model-based development of flight control laws, a high-fidelity nonlinear full-flexible flight dynamics model of the G-Flights Dimona was developed in a previous work using system identification techniques [29]. The model formulation combines linear structural dynamics with distributed quasi-steady strip aerodynamics. Decoupled nonlinear equations of motion are developed with the choice of the body reference frame to satisfy the linearized mean axes constraints [30] and by representing the elastic deformation of the strips in terms of free vibration modes. The body reference frame O_{B^R} is located at the instantaneous center of mass of the aircraft, see Figure 2. The position r_i of a mass element dm_i of an arbitrary strip of the flexible aircraft in an inertial reference frame O_I can be expressed in terms of the relative position p_i to this body reference frame and the position r_0 of the origin of O_{B^R}.

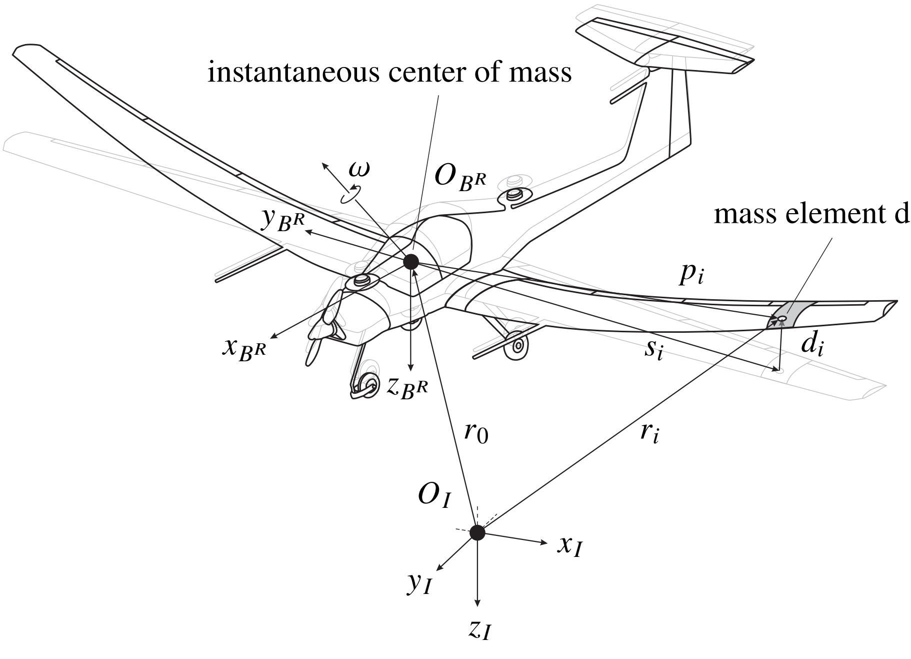

Fig. 2 Elastic strip deformation and definition of reference frames

With the free vibration modes, the elastic deformation of the strips can be written as a summation of mode shapes Φ_j multiplied by their respective modal coordinates η_j

$$d = \begin{bmatrix} d_1 \\ \vdots \\ d_{n_s} \end{bmatrix} = \sum_j \Phi_j \eta_j, \tag{1}$$

where d_i are the elastic translational deformation vectors of the $i = 1, \ldots, n_s$ strips. Then, the position p_i of the mass element can be separated into its undeformed part s_i and its deformation part d_i. Using O_{B^R} to derive the equations of motion of the flexible aircraft and only assuming small deformations, the decoupled equations of motion are given as follows:

$$\dot{V}|_{B^R} = -\omega|_{B^R} \times V|_{B^R} + T_{B^R I} G|_I + \frac{1}{m} F^{ext}|_{B^R} \tag{2}$$

$$\dot{\omega}|_{B^R} = -J^{-1}(\omega|_{B^R} \times (J\omega|_{B^R})) + J^{-1} M^{ext}|_{B^R} \tag{3}$$

$$\ddot{\eta}_j = -2\xi_j \omega_{n,j} \dot{\eta}_j - \omega_{n,j}^2 \eta_j + \frac{1}{\mu_j} Q_{\eta_j}. \tag{4}$$

The first two equations are the nonlinear equations for the rigid-body translational and rotational degrees of freedom. Therein, $V|_{B^R}$ and $\omega|_{B^R}$ denote the translational and angular velocity vectors of the body reference axes, respectively, $G|_I$ is the gravity vector, $T_{B^R I}$ is the transformation matrix from O_I to O_{B^R}, m is the aircraft mass, and J the inertia tensor (constant assuming only small deformations). The vectors $F^{ext}|_{B^R}$ and $M^{ext}|_{B^R}$ represent the sum of external forces and moments, respectively. The last equation (4) represents the linear differential equations for the aeroelastic dynamics in modal coordinates. Within these equations, η_j is the modal coordinate, $\omega_{n,j}$ the undamped natural frequency, ξ_j the modal damping ratio, μ_j the generalized mass, and Q_{η_j} the generalized force of each mode. The aerodynamic strip forces and moments are modeled using quasi-steady stability and control derivatives. A total number of 48 wing strips, 8 horizontal tail strips, and 5 vertical tail strips are considered. The aircraft model is implemented in MATLAB/Simulink with additional models of the propulsion system, actuator and control surface dynamics, earth and atmosphere, wind and turbulence, sensors, and filters. For model identification, several test activities were performed. Seven structural modes and the corresponding mode shapes were identified from a modal analysis based on ground vibration test (GVT) data and a finite element (FE) model. Initial distributions for the stability and control derivatives were derived from three-dimensional vortex-lattice-method steady-flow calculations. They were subsequently adapted during parameter estimation based on flight test data obtained from an extensive flight test campaign. Details on the modeling framework, model identification, and results are presented in [29].

2.2 Linear Prediction Models

The model predictive flight controller designed in this work is based on linear models. Thus, linearization of the nonlinear model is necessary. Since the aircraft dynamics and control surface effectiveness vary substantially across the flight envelope, the nonlinear aircraft model is trimmed and linearized in straight level flight and coordinated turn for different values of the airspeed $V_A \in [18, 34]$m/s in steps of 2m/s and bank angle $\phi \in [-40, 40]^\circ$ in steps of 10° or 15°. First-order Taylor approximation of the nonlinear model with respect to the different trim conditions yields a set of linear time-invariant (LTI) models. A single LTI model with respect to the trim condition $(\bar{x}, \bar{u}, \bar{y})$ is obtained as

$$\begin{aligned} \dot{x}(t) &= A(x(t) - \bar{x}) + B(u(t) - \bar{u}) \\ y(t) &= C(x(t) - \bar{x}) + D(u(t) - \bar{u}) + \bar{y}, \end{aligned} \tag{5}$$

where $x \in \mathbb{R}^{n_x}$ is the state, $u \in \mathbb{R}^{n_u}$ the input, and $y \in \mathbb{R}^{n_y}$ the output vector. Each LTI model consists of 58 states comprising 22 actuator states, 14 modal amplitudes and rates of the structural dynamics, 12 states of the rigid-body flight dynamics, and five states due to first order Padé approximation of five internal downwash delays. The inputs include the 11 control surfaces of the aircraft and the throttle. The outputs consist of typical flight dynamic parameters and the control surface deflections. For the design of the controller, the inputs are reduced to the control surfaces. Further, the left and right elevator are preallocated as one input to achieve a synchronised movement. The outputs are selected to contain the controlled outputs load factor, bank angle, and angle of sideslip. In addition, the control surface deflections of ailerons and flaperons are selected as auxiliary outputs. Model order reduction is performed for all LTI models by means of classical truncation and residualization [31]. A consistent state space is preserved for all models. Only one state per actuator is retained, effectively reducing the actuator dynamics to first order. Further, the rigid-body flight dynamics are reduced to the states capturing the relevant dynamics of the controlled output. From the structural modes, only the first symmetric wing bending mode is considered in the prediction models as its natural frequency of 3.97Hz is below the maximum servo actuator bandwidth of 6.74Hz [29]. The resulting reduced-order linear prediction models consist of 24 states each. The models are finally converted to discrete-time using Euler's method for simplicity, such that $A_d = I + T_s A_r$, $B_d = T_s B_r$, $C_d = C_r$, and $D_d = D_r$, where T_s is the controller sampling time and (A_r, B_r, C_r, D_r) are the reduced-order system state space matrices.

3 Model Predictive Flight Control Design

3.1 Problem Formulation

From the set of reduced-order prediction models, multiple linear model predictive controllers are designed for the different points in the flight envelope. The control architecture in this work uses the MPC for longitudinal and lateral-directional control of the aircraft. That is, it is responsible for stabilization of the aircraft and tracking of reference commands by computing optimal control inputs for the elevators δ_e, ailerons δ_a, flaperons δ_f, and rudder δ_r based on the predictions of rigid-body and aeroelastic dynamics. The reference commands are given in terms of piece-wise constant reference signals for the controlled output load factor r_{n_z}, bank angle r_ϕ, and angle of sideslip r_β. The load factor reference is computed by an outer control loop comprising a gain-scheduled P controller to track a sink rate reference r_w and a turn compensation. The control architecture is completed by an observer for the estimation of non-measurable states and a gain-scheduled PI controller to track the airspeed reference r_{V_A} via the throttle δ_t. Figure 3 displays the overall control architecture.

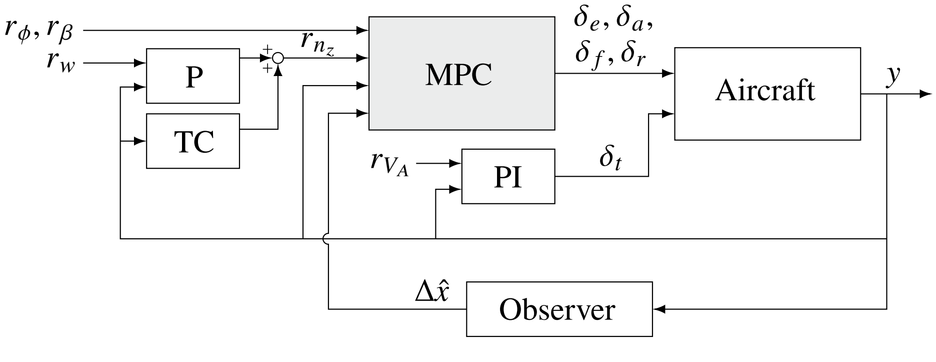

Fig. 3 Flight control architecture

Given the reference signals for the controlled output and choosing a reference for the auxiliary output, the MPC reference vector r is assembled. A zero reference is chosen for the auxiliary output such that ailerons and flaperons return back to neutral position in steady-state in order to minimize drag. Thus, the reference vector is defined as

$$r = \begin{bmatrix} r_{n_z} & r_\phi & r_\beta & 0_{1\times(n_a+n_f)} \end{bmatrix}^\mathsf{T}, \tag{6}$$

where n_a and n_f denote the number of ailerons and flaperons, respectively. To achieve offset-free tracking of the controller, the MPC formulation using a velocity-form model is employed [32]. For each point in the flight envelope, the velocity-form model is obtained by augmenting the respective discrete-time linear prediction model as follows. To simplify the notation, u_k, x_k, and y_k are used to describe the input, state, and output of the reduced-order prediction models at time step k. Let $\Delta u_k = u_k - u_{k-1}$ and $\Delta x_k = x_k - x_{k-1}$. Further, define the new augmented state vector

$$x_k^a = \begin{bmatrix} \Delta x_k^\mathsf{T} & e_k^\mathsf{T} \end{bmatrix}^\mathsf{T} \tag{7}$$

where $e_k = y_k - r$ is the tracking error. Then, the velocity-form model is given as

$$\begin{aligned} \begin{bmatrix} \Delta x_{k+1} \\ e_{k+1} \end{bmatrix} &= \overbrace{\begin{bmatrix} A_d & 0_{n_x\times n_y} \\ C_d A_d & I_{n_y\times n_y} \end{bmatrix}}^{\bar{A}} \overbrace{\begin{bmatrix} \Delta x_k \\ e_k \end{bmatrix}}^{x_k^a} + \overbrace{\begin{bmatrix} B_d \\ C_d B_d + D_d \end{bmatrix}}^{\bar{B}} \Delta u_k \\ e_k &= \underbrace{\begin{bmatrix} 0_{n_y\times n_x} & I_{n_y\times n_y} \end{bmatrix}}_{\bar{C}} \begin{bmatrix} \Delta x_k \\ e_k \end{bmatrix}. \end{aligned} \tag{8}$$

This augmentation of the original discrete-time models leads to a natural embedding of integrators and, under reasonable assumptions, ensures zero steady-state error even in case of a mismatch between the prediction model and the actual system [1]. The predictive control law is computed by minimizing the quadratic cost function

$$J_k = \sum_{i=1}^{N-1} \left(\|\Delta x_{k+i}\|_{Q_1}^2 + \|e_{k+i}\|_{Q_2}^2 \right) + \|e_{k+N}\|_P^2 + \sum_{i=0}^{N-1} \left(\|\Delta u_{k+i}\|_R^2 \right) \tag{9}$$

according to the RHC principle, where N denotes the prediction horizon, Q_2, R, and $P = cQ_2$ with $c > 0$ are positive definite weighting matrices and Q_1 is a positive semi-definite weighting matrix. The weighting matrices can be chosen individually for each point in the flight envelope to tune closed-loop performance and control effort. Note that in this formulation of the cost function, the prediction horizon equals the control horizon. The optimization problem to be solved at each time step k is given by

$$\min_{\Delta U_k} \quad J_k \tag{10a}$$

$$\text{s.t.} \quad x_{k+i+1}^a = \bar{A} x_{k+i}^a + \bar{B} \Delta u_{k+i} \tag{10b}$$

$$u_{k+i} = u_{k-1} + \textstyle\sum_{j=0}^{i} \Delta u_{k+j} \in \mathcal{U}, \quad i \in [0 \;\; N-1] \tag{10c}$$

$$\Delta x_{k+N} = 0 \tag{10d}$$

with the vector of future incremental control inputs $\Delta U_k = \begin{bmatrix} \Delta u_k^\mathsf{T} \dots \Delta u_{k+N-1}^\mathsf{T} \end{bmatrix}^\mathsf{T}$ as the decision variables and the input constraint set $\mathcal{U}$. Only input constraints are considered in this work. Stability is addressed in terms of the terminal equality constraint (10d) and the terminal penalty P on the final tracking error in equation (9). The advantage of using the terminal equality constraint is that all equilibria are mapped to the origin of the incremental states $\Delta x = 0$. Hence, the predicted terminal state is not required to reach a certain terminal region in the state space but only a steady state, and $\Delta u = 0$ can be used to

[1] Including D_d in $\bar{B}$ leads to a small approximation error with regard to Δu_k, since Δu_{k+1} is not known at time step k.

guarantee stability even for unreachable reference commands, provided any equilibrium is reachable within the prediction horizon [33]. A proof of stability with respect to the more general case of linear parameter-varying (LPV) systems is given in [34].

Using ΔU_k and the vector of the predicted states $X_k^a = \left[{x_{k+1}^a}^\mathsf{T} \dots {x_{k+N}^a}^\mathsf{T}\right]^\mathsf{T}$, cost function (9) can be written in vector form as

$$J_k = (X_k^a)^\mathsf{T} \tilde{Q} (X_k^a) + \Delta U_k^\mathsf{T} \tilde{R} \Delta U_k, \tag{11}$$

where

$$\tilde{Q} = \begin{bmatrix} \text{diag}_{N-1}(Q_1, Q_2) & \\ & \text{diag}_1(Q_1, P) \end{bmatrix}, \quad \tilde{R} = \text{diag}_N(R) \tag{12}$$

are the weighting matrices over the horizon. Further, the predicted state trajectory X_k^a can be expressed as a function of the control trajectory ΔU_k, i.e,

$$X_k^a = \Phi x_k^a + \Gamma \Delta U_k \tag{13}$$

with the prediction matrices

$$\Phi = \begin{bmatrix} \bar{A} \\ \bar{A}^2 \\ \bar{A}^3 \\ \vdots \\ \bar{A}^N \end{bmatrix}, \quad \Gamma = \begin{bmatrix} \bar{B} & 0 & 0 & \dots & 0 \\ \bar{A}\bar{B} & \bar{B} & 0 & \dots & 0 \\ \bar{A}^2\bar{B} & \bar{A}\bar{B} & \bar{B} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{A}^{N-1}\bar{B} & \bar{A}^{N-2}\bar{B} & \bar{A}^{N-3}\bar{B} & \dots & \bar{B} \end{bmatrix}. \tag{14}$$

Inserting expression (13) of X_k^a into the cost function (11) effectively eliminates constraint (10b) and one obtains the dense formulation. The resulting optimization problem is given by

$$\min_{\Delta U_k} \quad \frac{1}{2} \Delta U_k^\mathsf{T} H \Delta U_k + g^\mathsf{T} \Delta U_k \tag{15a}$$

$$\text{s.t.} \quad G_u \Delta U_k \leq h_u \tag{15b}$$

$$G_{\Delta x} \Gamma \Delta U_k = -G_{\Delta x} \Phi x_k^a, \tag{15c}$$

where G_u and $G_{\Delta x}$ are selector matrices of the input and terminal equality constraints, respectively, and h_u contains the upper and lower limits of the control input. The Hessian H and linear cost term g are given by

$$H = \Gamma^\mathsf{T} \tilde{Q} \Gamma + \tilde{R}, \quad g = \Gamma^\mathsf{T} \tilde{Q} \Phi x_k^a. \tag{16}$$

Problem (15) is a quadratic programming (QP) problem and can be solved efficiently using fast QP solvers. Note that in order for problem (15) to be set up, the initial condition x_k^a has to be known. From the vector of future incremental control inputs ΔU_k, only the first element Δu_k is implemented to obtain the current control input according to $u_k = u_{k-1} + \Delta u_k$, given the previous input u_{k-1}.

3.2 Laguerre Functions

Using MPC in aircraft flight control often requires long prediction horizons because typical sampling times are short compared the time constants of the governing dynamics. This leads to a high number of decision variables and significantly complicates real-time implementation of the controller. In order to reduce the number of decision variables, in this work, the incremental control trajectory is approximated by a set of Laguerre functions [26]. Laguerre functions are orthonormal functions that can be suitably described in discrete-time using only two parameters. They are also exponentially decaying functions, such that by using these functions to describe ΔU_k, it is assumed that the incremental control trajectory within the moving horizon window behaves like the impulse response of a stable system, that is, exponentially decays after some initial time period [25]. With this design methodology, the problem of finding the incremental control trajectory ΔU_k is converted into a problem of finding coefficients for the set of Laguerre functions. Typically, a significant reduction of decision variables can be achieved [35].

Consider the single-input case for simplicity. The incremental control trajectory can be approximated by N_L discrete-time Laguerre functions as

$$\Delta u_{k+i} \approx \sum_{m=1}^{N_L} l_{m,i}\, c_{m,k} = L_i^{\mathsf{T}} \gamma_k, \quad i \in [0 \;\; N-1]\,, \tag{17}$$

where $l_{m,i}$ are the $m = 1, \ldots, N_L$ discrete-time Laguerre functions evaluated at future time step i of the moving horizon window starting from the current time step k, and $c_{m,k}$ are the corresponding coefficients. Note that the coefficients are constant over the prediction horizon. The vectors $L_i^{\mathsf{T}} = [l_{1,i} \ldots l_{N_L,i}]$ and $\gamma_k = [c_{1,k} \ldots c_{N_L,k}]^{\mathsf{T}}$ summarize the Laguerre functions and coefficients in vector form, respectively. The discrete-time Laguerre functions can be described with the state space realization

$$L_{i+1} = A_L L_i, \tag{18}$$

where

$$A_L = \begin{bmatrix} a & 0 & 0 & \ldots & 0 & 0 \\ \epsilon & a & 0 & \ldots & 0 & 0 \\ -a\epsilon & \epsilon & a & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & a & 0 \\ (-1)^{N_L-2}a^{N_L-2}\epsilon & (-1)^{N_L-3}a^{N_L-3}\epsilon & (-1)^{N_L-4}a^{N_L-4}\epsilon & & \epsilon & a \end{bmatrix} \tag{19}$$

with $\epsilon = 1 - a^2$ and initial condition

$$L_0^{\mathsf{T}} = \sqrt{\epsilon}\begin{bmatrix} 1 & -a & a^2 & -a^3 & \ldots & (-1)^{N_L-1}a^{N_L-1} \end{bmatrix}. \tag{20}$$

The pole a, also called scaling factor, and the number of Laguerre functions N_L can be used as additional tuning knobs in the design. With a, the exponential decay rate of the functions is set. It is used to tune the control horizon. For $a = 0$, the Laguerre functions become a set of pulses and the description of the incremental control trajectory becomes equivalent to the original approach. With N_L, the number of Laguerre functions to approximate the incremental control trajectory and hence the number of decision variables is determined. It is used to control the complexity of the incremental control trajectory.

Let the incremental control trajectory be described by the set of Laguerre functions. Expressed in vector form, ΔU_k is given as

$$\Delta U_k = \begin{bmatrix} L_0 & L_1 & \ldots & L_{N-1} \end{bmatrix}^{\mathsf{T}} \gamma_k = \Omega_L \gamma_k. \tag{21}$$

Extending this to the multi-input case, each input $j = 1, \ldots, n_u$ is described by a separate set of Laguerre functions $N_{L,j}$ with pole a_j, i.e.

$$\Delta u_{k+i,j} = L_{i,j}^{\mathsf{T}}\, \gamma_{k,j}, \tag{22}$$

and the description of the incremental control trajectory in equation (21) is modified such that

$$L_i = \begin{bmatrix} L_{i,1} & 0 & \ldots & 0 \\ 0 & L_{i,2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \ldots & 0 & L_{i,n_u} \end{bmatrix}, \quad \gamma_k = \begin{bmatrix} \gamma_{k,1} \\ \gamma_{k,2} \\ \vdots \\ \gamma_{k,n_u} \end{bmatrix}. \tag{23}$$

The pole and number of Laguerre functions can be chosen individually for each input. Thus, the total number of decision variables results in $N_L = \sum_{j=1}^{n_u} N_{L,j}$. Inserting expression (21) of ΔU_k into problem (15), the optimization problem is rewritten as

$$\min_{\gamma_k} \quad \frac{1}{2}\gamma_k^\mathsf{T} H \gamma_k + g^\mathsf{T}\gamma_k \tag{24a}$$

$$\text{s.t.} \quad G_u^L \gamma_k \leq h_u \tag{24b}$$

$$G_{\Delta x}\Gamma\Omega_L\gamma_k = -G_{\Delta x}\Phi x_k^a, \tag{24c}$$

with

$$H = \Omega_L^\mathsf{T}\Gamma^\mathsf{T}\tilde{Q}\Gamma\Omega_L + R_L, \quad g = \Omega_L^\mathsf{T}\Gamma^\mathsf{T}\tilde{Q}\Phi x_k^a, \tag{25}$$

where $R_L \in \mathbb{R}^{N_L \times N_L}$ is the weighting matrix of the Laguerre coefficients. The input constraints are expressed in terms of the Laguerre coefficient vector γ_k. Noting that $u_{k+i} = u_{k-1} + \sum_{j=0}^{i} \Delta u_{k+j}$, the constraint matrix becomes

$$G_u^L = \begin{bmatrix} L_{\sum,0} & L_{\sum,1} & \dots & L_{\sum,N_c} \end{bmatrix}^\mathsf{T} \tag{26}$$

with

$$L_{\sum,i} = \begin{bmatrix} \sum_{j=0}^{i} L_{j,1} & 0 & \dots & 0 \\ 0 & \sum_{j=0}^{i} L_{j,2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \sum_{j=0}^{i} L_{j,n_u} \end{bmatrix}. \tag{27}$$

Input constraints are enforced on the first $N_c \leq N$ elements of the incremental control trajectory. By describing ΔU_k using exponentially decaying Laguerre functions, the number of constraints can potentially be reduced, since the incremental control input is ensured to converge to zero after some time period. Thus, it is often sufficient to impose the constraints only on the transient period of the trajectory [25]. Based on the RHC principle, the applied input at each time step is $u_k = u_{k-1} + L_0^\mathsf{T}\gamma_k$. Note that once the Laguerre parameters are selected in the control design, the matrix Ω_L and initial condition L_0 do not change. This approach of using Laguerre functions to describe ΔU_k effectively reduces the number of decision variables and thus on-line computation time. Moreover, the number of decision variables becomes independent of the prediction horizon such that sufficiently long horizons can be realized.

3.3 Observer Design

The construction of the optimization problem requires the knowledge of the initial augmented state vector x_k^a. The majority of the augmented states can be either be measured by the on-board instrumentation of the aircraft (rigid-body flight dynamics states) or calculated on-line from the control input and models of the actuator and control surface dynamics (actuator states). However, the states related to the structural dynamics and downwash delays are not measurable and thus have to be estimated using an observer. In this work, a linear parameter-varying Kalman filter is designed. To this end, a grid-based LPV model is constructed from the set of LTI models in Equation 5. Model reduction is performed to remove all states associated with the slow rigid-body flight dynamics. Further, rigid-body accelerations, velocities, and airspeed are selected as the model outputs. Lastly, the models are discretized using Euler's method with sampling time T_s to obtain the discrete-time observer state space matrices A_d^O, B_d^O, C_d^O, and D_d^O. The Kalman filter is designed to estimate the incremental state vector. Hence, the equation for the prediction step is

$$\Delta\hat{x}_{k|k-1} = A_d^O(\rho_{k-1})\Delta\hat{x}_{k-1|k-1} + B_d^O(\rho_{k-1})\Delta u_{k-1}, \tag{28}$$

where $\Delta\hat{x}_{k|k-1}$ denotes the *a priori* estimate of the incremental state vector and $\rho : \mathbb{R} \longrightarrow \mathcal{P}$ with $\mathcal{P} \subset \mathbb{R}^{n_\rho}$ is the vector of time-varying scheduling parameters. Airspeed is chosen as the only scheduling parameter.

The equation for the correction step is given by

$$\Delta\hat{x}_{k|k} = \Delta\hat{x}_{k|k-1} + K_k(\rho_k)\left(\Delta y_k - C_d^O(\rho_k)\Delta\hat{x}_{k|k-1} - D_d^O(\rho_k)\Delta u_k\right), \tag{29}$$

where $\Delta\hat{x}_{k|k}$ is the *a posteriori* estimate of the incremental state vector [2]. The Kalman gain K_k is determined from the solution of the discrete Riccati equation for each linear model of the LPV interpolation grid. The process and measurement noise covariances are chosen based on the measurement noise characteristics and the LPV model uncertainty.

4 Real-Time Implementation and Flight Testing

The presented MPC formulation using Laguerre functions to describe the incremental control trajectory provides a powerful framework for designing model predictive flight control for large scale systems such as flexible aircraft. In particular, the reduction of decision variables effectively reduces on-line computation time such that real-time implementation of the controller is feasible. The model predictive flight controller is finally tested in flight test experiments under real world conditions using the G-Flights Dimona. In the following, the tuning, real-time implementation, and flight test results are discussed.

4.1 Tuning and Real-Time Implementation

From the set of linear prediction models, 42 linear model predictive controllers covering the entire flight envelope are designed off-line. For each controller, the sampling time is set to $T_s = 0.04$s. First, closed-loop performance and control effort are initially tuned by selecting the weighting matrices $Q_1 = 0_{n_x \times n_x}$, $Q_2 = I_{n_y \times n_y}$, $P = 50 \cdot Q_2$, and $R_L = I_{N_L \times N_L}$. Next, the number of Laguerre functions to describe the incremental control trajectory of each input is chosen to $N_{L,j} = 5$. Thus, the total number of decision variables results to $N_L = 50$ (compared to 700 decision variables without the use of Laguerre functions). The corresponding Laguerre poles are set to $a = 0.88$ based on an estimate of the dominant eigenvalues of the closed-loop system [26]. Subsequently, the prediction horizon is fixed to $N = 70$ steps. This means, the controllers predict the aircraft response for 2.8s ahead of the current time step to determine the optimal control inputs. With this choice of Laguerre parameters and prediction horizon, sufficiently complex incremental control trajectories can be modeled that stabilize the aircraft within the prediction horizon. Further, the incremental control trajectory decays to zero at the end of the prediction horizon. In a last step, closed-loop performance and control effort are further tuned in nonlinear closed-loop simulations by adjusting the weighting matrices Q_1, Q_2, and R_L as a function of airspeed to account for the varying dynamics and control surface effectiveness. The auxiliary outputs of aileron and flaperon deflections are assigned low weights in order to not affect the transient response of the control trajectory. Input constraints are enforced on the first $N_c = 18$ samples, capturing the transient maxima of the control trajectory. Given the horizon weight, prediction, Laguerre, and constraint matrices for the set of controllers, the quadratic programs are constructed. The resulting Hessian, linear cost term (except for x_k^a), and terminal equality constraint matrices are stored as 4-dimensional matrices representing the grid of airspeed and bank angle. In this way, the construction of the QP matrices is performed off-line which reduces computation time. However, more on-line memory is required. The input constraint matrix G_u^L and initial condition L_0 are stored as constant matrices. Lastly, the LPV Kalman filter is constructed off-line based on the process and measurement noise covariance matrices. The resulting Kalman gains and LPV model system matrices are stored as 3-dimensional matrices representing the grid of airspeed.

Next, the controller is integrated into the real-time application of the flight control computer in MATLAB/Simulink among the existing applications for measurement processing, guidance, navigation,

[2] Including D_d in (29) generally causes an algebraic loop, since the calculation of Δu_k is based on $\Delta\hat{x}_{k|k}$ itself. However, in this case, Δu_k is taken from the low-level control output of the flight control computer which introduces a small delay.

and control. During on-line operation, the stored Hessian, linear cost term, and equality constraint matrices are interpolated based on the current measurements of airspeed $V_{A,k}$ and bank angle ϕ_k, and the QP assembled given the initial augmented state vector x_k^a. The input constraints are set up based on the previous control input u_{k-1}. The resulting QP is solved using the interior-point (IP) solver *qpSWIFT* [36]. The maximum number of iterations is limited to five. This early termination potentially produces suboptimal solutions since the QPs are not solved to full accuracy. However, it is found that the solver already provides good solutions that satisfy the constraints after this number of iterations. Further, considering the predictive control law is computed according to the RHC principle, the quality of control obtained still is high [37]. Given the solution vector of the Laguerre coefficients γ_k, the control input is computed. After code generation, the overall maximum computation time of the controller on the flight control computer, including all other applications running in parallel, results to 0.036s.

4.2 Flight Test Results

The controller is eventually tested in flight test experiments with four different tracking scenarios. Each scenario consists of a predefined reference maneuver in terms of sink rate r_w, bank angle r_ϕ, angle of sideslip r_β, and airspeed r_{V_A} commands. Scenario 1 is a sink rate doublet, scenario 2 a bank angle doublet, scenario 3 a simultaneous sink rate and bank angle pulse command, and scenario 4 an airspeed pulse command. In September 2023, a total of 60 maneuvers were successfully performed at different airspeeds. On the flight test day, wind and gust speeds between 2-5m/s were measured. Figure 4 shows the flight test result of tracking scenario 1-4 executed at medium airspeed reference commands between 22m/s and 28m/s for the controlled output and input. In the plots, the measurements of load factor and airspeed are filtered at 15Hz cut-off frequency to enhance visibility. The control surfaces are denoted according to their position on the aircraft, that is, subscript "l" is used for left, "r" for right, "out" for outer, and "in" for inner position. The measurements show a very good tracking performance of the model predictive flight controller. In particular, the tracking of bank angle is excellent with steady tracking errors not exceeding $\pm 1^\circ$. The controller effectively minimizes the strong coupling of roll and yaw motion characteristic of this aircraft, thus allowing for an accurate tracking of bank angle while keeping angles of sideslip small. This is achieved by computing optimal control inputs for ailerons, flaperons, and rudder based on the prediction of the aircraft response. A maximum transient tracking error of -6.8° for the angle of sideslip is observed at the 60° bank angle step change. However, considering the wind and gust conditions on the flight test day, this is deemed acceptable. Sink rate is tracked well with steady tracking errors predominantly within ± 1m/s and a transient maximum error of -2.2m/s during the 60° bank angle step change. A similar result is achieved for the tracking of load factor, where the maximum tracking error of 0.46 is also observed during the 60° bank angle step change but otherwise stays within ± 0.2. The tracking of airspeed is acceptable although the airspeed controller could be tuned more dynamically in the future. Wind and gust disturbances are most visible in the tracking of load factor and angle of sideslip, leading to continuous control action of elevators and rudder throughout the maneuvers. Looking at the wing control surface deflections, ailerons and flaperons are primarily used for roll control with the control effort distributed among the control surfaces based their roll moment effectiveness. That is, outer left and right ailerons deflect the most. Opposite pairs do not necessarily deflect in a synchronised fashion, as they are manipulated individually by the controller. After a transient response during control action, ailerons and flaperons return back to neutral position in steady-state. This confirms that the inclusion of the auxiliary output in the design of the controller works as intended. From all control surfaces, only the rudder approaches its limit during the transient response to the 60° bank angle step change. However, no constraints are violated. A small positive offset in the rudder deflection is observed throughout all maneuvers. This indicates a possible offset in the angle of sideslip measurement.

In order to evaluate the performance of the controller across the entire flight envelope, additional plots of the flight test result for tracking scenario 1-4 executed at low (below 22m/s) and high airspeed reference commands (above 28m/s) are presented in the appendix in Figure 5 and Figure 6, respectively.

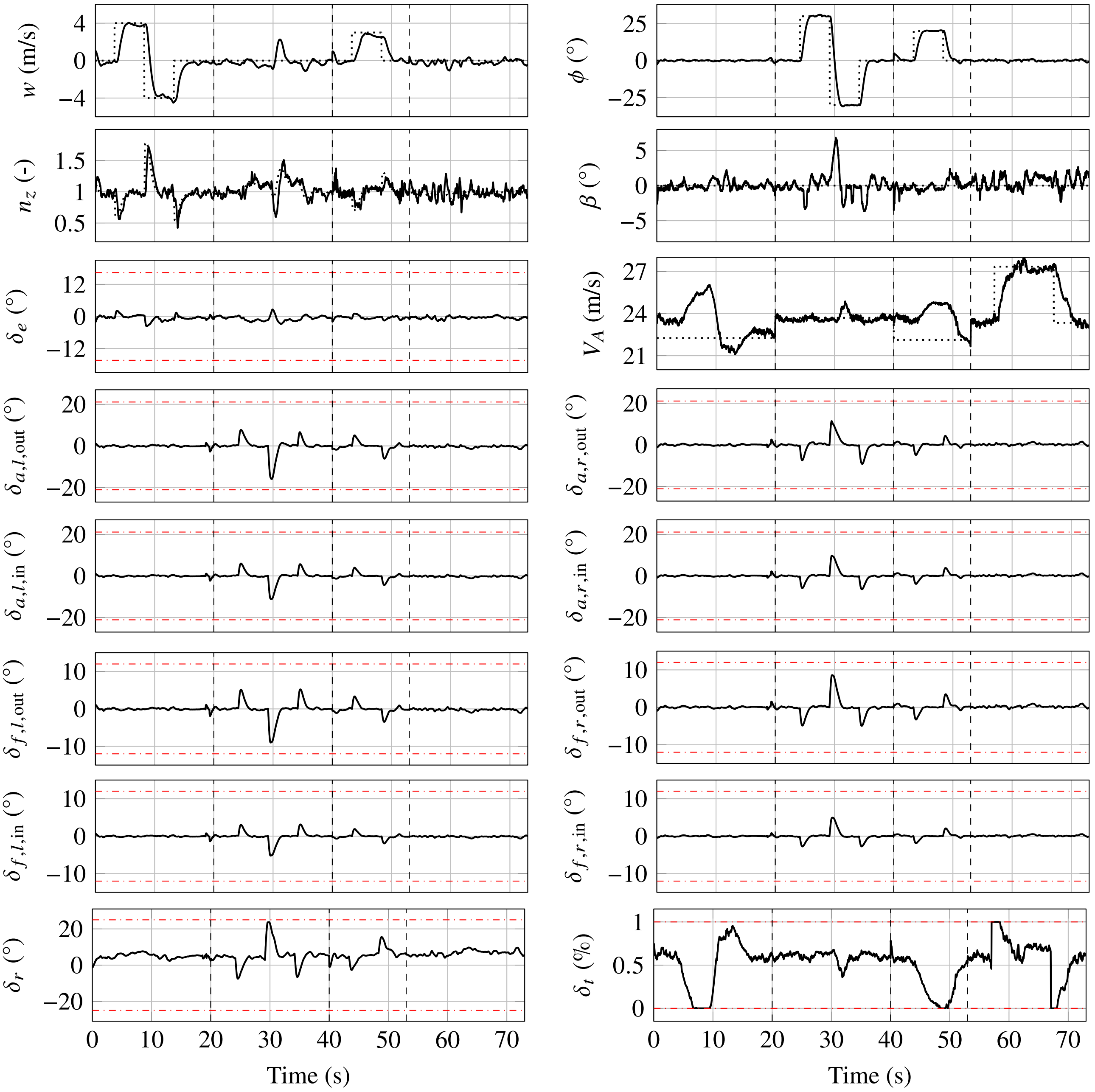

Fig. 4 Flight test results for tracking scenarios 1-4 at medium airspeed reference commands (reference: dotted, measured: solid, constraints: dash-dotted, maneuver separation: dashed)

As expected, less control effort is required at higher airspeeds due to the increase of control surface effectiveness. At lower airspeeds, the damping of rigid-body modes decreases such that a higher damping was targeted during controller tuning, resulting in slightly slower response times. However, overall, similar tracking performances are achieved. These results confirm that the scheduling of multiple linear model predictive controllers designed at different points in the flight envelope is an effective method for achieving good tracking performance across the entire flight envelope. Across all airspeed, no strong excitation of the first symmetric wing bending mode is observed during the maneuvers. This indicates that the contribution of this structural mode in the predictions of the maneuvers alone is small compared to the rigid-body modes. However, in a subsequent work the controller will be extended with maneuver and gust load alleviation functions where fast actuator response times are needed. Moreover, nonlinear simulations of various gust interactions in preparation of the flight tests have shown a significant excitation of the structural mode with considerable effect on the wing loads.

5 Conclusion

A model predictive flight controller for the longitudinal and lateral-directional control of a slightly flexible 25kg unmanned aircraft is proposed, implemented in real-time, and validated in flight test experiments. The controller is responsible for the stabilization of the aircraft and tracking of reference commands by computing optimal control inputs for the 11 control surfaces of the aircraft. Reduced-order linear prediction models are derived from a high-fidelity nonlinear model of the aircraft at different points in the flight envelope. Aeroelastic effects are accounted for in the predictions by including relevant structural modes in the reduced-order models. From the set of prediction models, multiple linear model predictive controllers are first designed off-line, applying the MPC formulation using a velocity-form model for offset-free tracking, and later scheduled during on-line operation. To achieve a reduction of computation time and thus enable real-time implementation of the controller, orthonormal Laguerre functions are used to approximate the incremental control trajectory. This approach proves effective in significantly reducing the number of decision variables and constraints of the optimization problem. The controller is successfully validated in various flight test experiments at different airspeeds comprising four different tracking scenarios. The results show a very good tracking performance of the controller even in the presence of wind and gust disturbances. In particular, the tracking of the bank angle using multiple wing control surfaces is very accurate. Overall, the scheduling of multiple linear controllers designed at different points in the flight envelope proves effective in achieving good tracking performance across the entire flight envelope. Future work will consider the augmentation of the controller with load control functions to design an integrated flight and loads controller.

Appendix

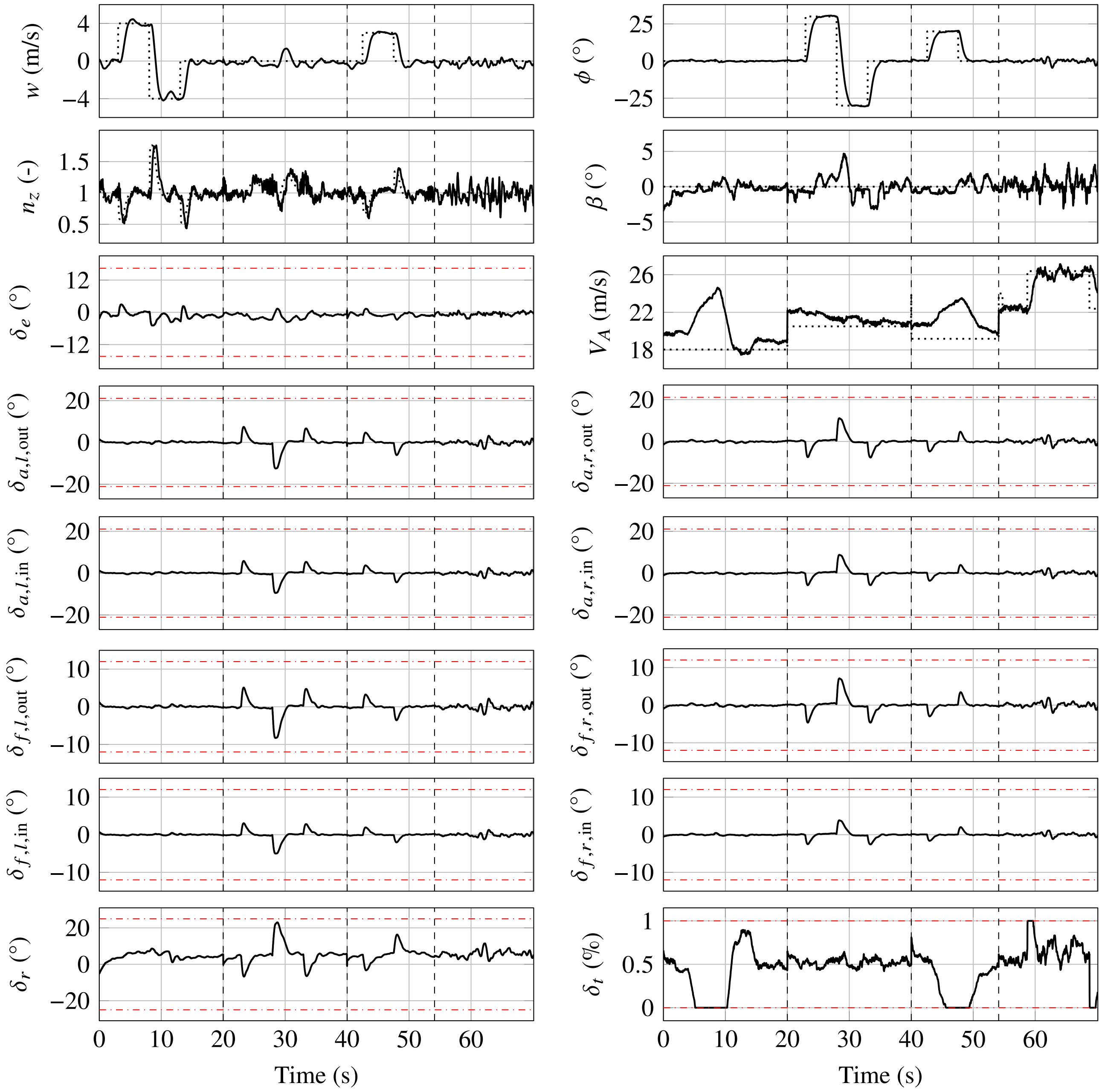

Fig. 5 Flight test results for tracking scenarios 1-4 at low airspeed reference commmands (reference: dotted, measured: solid, constraints: dash-dotted, maneuver separation: dashed)

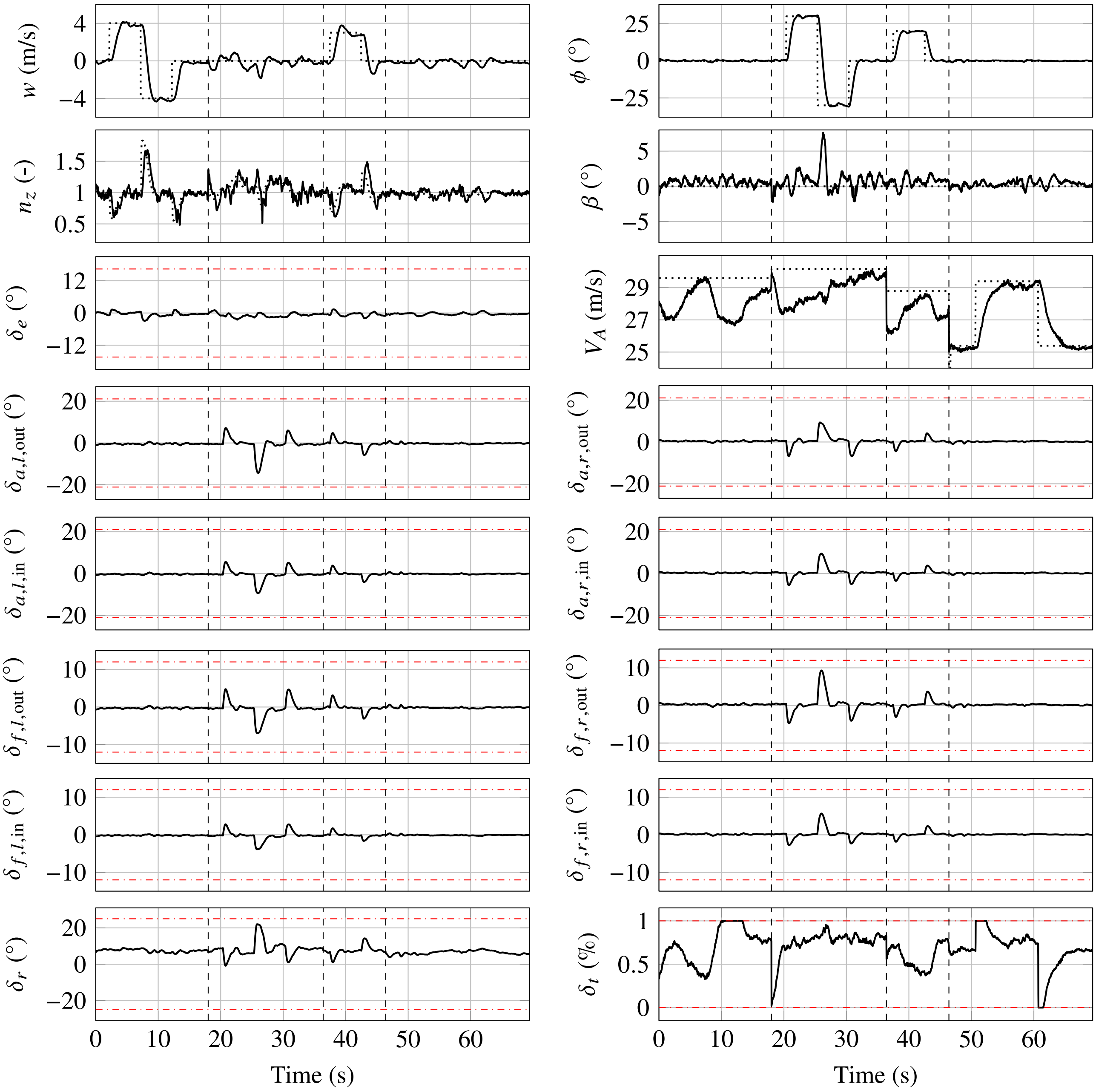

Fig. 6 Flight test results for tracking scenarios 1-4 at high airspeed reference commands (reference: dotted, measured: solid, constraints: dash-dotted, maneuver separation: dashed)

Acknowledgments

The authors would like to thank Lothar Desel and Rasmus Köhler for their assistance during preparation and execution of the flight tests. This work was funded by the research project "Integrated flight and loads control of very flexible aircraft for the multidisciplinary aircraft design of future climate-neutral aircraft" (FlexFuture), which is supported by the German Federal Ministry for Economic Affairs and Climate Action in the national LuFo VI-2 program. Any opinions, findings and conclusions expressed in this document are those of the authors and do not necessarily reflect the views of the other project partners.

Supported by:

Federal Ministry
for Economic Affairs
and Climate Action

on the basis of a decision
by the German Bundestag

References

[1] Ilhan Tuzcu and Leonard Meirovitch. Effects of flexibility on the stability of flying aircraft. *Journal of Dynamic Systems, Measurement, and Control*, 127(1):41–49, June 2004. ISSN: 1528-9028. DOI: 10.1115/1.1870040.

[2] Flávio J. Silvestre, Antônio B. Guimarães Neto, Rafael Mendes Bertolin, Roberto Gil Annes da Silva, and Pedro Paglione. Aircraft control based on flexible aircraft dynamics. *Journal of Aircraft*, 54(1):262–271, Jan. 2017. ISSN: 1533-3868. DOI: 10.2514/1.c033834.

[3] C. Shearer and S. Heise. Constrained model predictive control of a nonlinear aerospace system. In *Guidance, Navigation, and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, aug 1998. DOI: 10.2514/6.1998-4235.

[4] R. Bhattacharya, G. J. Balas, M. A. Kaya, and A. Packard. Nonlinear receding horizon control of an f-16 aircraft. *Journal of Guidance, Control, and Dynamics*, 25(5):924–931, sep 2002. DOI: 10.2514/2.4965.

[5] J. M. Maciejowski and C. N. Jones. MPC fault-tolerant flight control case study: flight 1862. *IFAC Proceedings Volumes*, 36(5):119–124, jun 2003. DOI: 10.1016/s1474-6670(17)36480-7.

[6] D. K. Kufoalor and T. A. Johansen. Reconfigurable fault tolerant flight control based on nonlinear model predictive control. In *2013 American Control Conference*. IEEE, jun 2013. DOI: 10.1109/acc.2013.6580635.

[7] M. M. Kale and A. J. Chipperfield. Stabilized MPC formulations for robust reconfigurable flight control. *Control Engineering Practice*, 13(6):771–788, jun 2005. DOI: 10.1016/j.conengprac.2004.09.001.

[8] P. W. Gibbens and E. D. B. Medagoda. Efficient model predictive control algorithm for aircraft. *Journal of Guidance, Control, and Dynamics*, 34(6):1909–1915, nov 2011. DOI: 10.2514/1.52162.

[9] A. Abate and L. El Ghaoui. Robust model predictive control through adjustable variables: an application to path planning. In *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*. IEEE, 2004. DOI: 10.1109/cdc.2004.1428786.

[10] S. H. Mathisen, K. Gryte, T. Johansen, and T. I. Fossen. Non-linear model predictive control for longitudinal and lateral guidance of a small fixed-wing UAV in precision deep stall landing. In *AIAA Infotech @ Aerospace*. American Institute of Aeronautics and Astronautics, jan 2016. DOI: 10.2514/6.2016-0512.

[11] D. Simon, O. Härkegård, and J. Löfberg. Command governor approach to maneuver limiting in fighter aircraft. *Journal of Guidance, Control, and Dynamics*, 40(6):1514–1527, jun 2017. DOI: 10.2514/1.g002272.

[12] M. de F.V. Pereira, I. Kolmanovsky, C. E. Cesnik, and F. Vetrano. Model predictive control architectures for maneuver load alleviation in very flexible aircraft. In *AIAA Scitech 2019 Forum*. American Institute of Aeronautics and Astronautics, jan 2019. DOI: 10.2514/6.2019-1591.

[13] M. de F.V. Pereira, I. Kolmanovsky, C. E. Cesnik, and F. Vetrano. Model predictive control for maneuver load alleviation in flexible airliners. In *International Forum on Aeroelasticity and Structural Dynamics (IFASD)*, pages 420–432. Curran Associates, Inc. Red Hook, NY, 2019.

[14] S. Haghighat, H. Liu, and J. R. R. A. Martins. Application of model predictive control to gust loads alleviation systems. In *AIAA Atmospheric Flight Mechanics Conference*. American Institute of Aeronautics and Astronautics, jun 2009. DOI: 10.2514/6.2009-5929.

[15] S. Haghighat, H. H. T. Liu, and J. R. R. A. Martins. Model-predictive gust load alleviation controller for a highly flexible aircraft. *Journal of Guidance, Control, and Dynamics*, 35(6):1751–1766, nov 2012. DOI: 10.2514/1.57013.

[16] H.-G. Giesseler, M. Kopf, P. Varutti, T. Faulwasser, and R. Findeisen. Model predictive control for gust load alleviation. *IFAC Proceedings Volumes*, 45(17):27–32, 2012. DOI: 10.3182/20120823-5-nl-3013.00049.

[17] H.-G. Giesseler, M. Kopf, T. Faulwasser, P. Varutti, and R. Findeisen. Gust load alleviation based on model predictive control. In *International Forum on Aeroelasticity and Structural Dynamics 2013 (IFASD)*, number CONF, 2013.

[18] R. J. Simpson, R. Palacios, H. Hesse, and P. Goulart. Predictive control for alleviation of gust loads on very flexible aircraft. In *55th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. American Institute of Aeronautics and Astronautics, jan 2014. DOI: 10.2514/6.2014-0843.

[19] Y. Wang, A. Wynn, and R. Palacios. Model-predictive control of flexible aircraft dynamics using nonlinear reduced-order models. In *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. American Institute of Aeronautics and Astronautics, jan 2016. DOI: 10.2514/6.2016-0711.

[20] M. Kopf, E. Bullinger, H.-G. Giesseler, S. Adden, and R. Findeisen. Model predictive control for aircraft load alleviation: Opportunities and challenges. In *2018 Annual American Control Conference (ACC)*. IEEE, jun 2018. DOI: 10.23919/acc.2018.8430956.

[21] M. de F.V. Pereira, I. Kolmanovsky, and C. E. Cesnik. Model predictive control with constraint aggregation applied to conventional and very flexible aircraft. In *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, dec 2019. DOI: 10.1109/cdc40024.2019.9029769.

[22] W. B. Dunbar, M. B. Milam, R. Franz, and R. M. Murray. Model predictive control of a thrust-vectored flight control experiment. *IFAC Proceedings Volumes*, 35(1):355–360, 2002. DOI: 10.3182/20020721-6-es-1901.00965.

[23] T. Keviczky and G. J. Balas. Flight test of a receding horizon controller for autonomous UAV guidance. In *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005. DOI: 10.1109/acc.2005.1470518.

[24] T. J. Stastny, A. Dash, and R. Siegwart. Nonlinear MPC for fixed-wing UAV trajectory tracking: Implementation and flight experiments. In *AIAA Guidance, Navigation, and Control Conference*. American Institute of Aeronautics and Astronautics, jan 2017. DOI: 10.2514/6.2017-1512.

[25] L. Wang. Discrete time model predictive control design using laguerre functions. In *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*. IEEE, 2001. DOI: 10.1109/acc.2001.946117.

[26] L. Wang. Discrete model predictive controller design using laguerre functions. *Journal of Process Control*, 14(2):131–142, mar 2004. DOI: 10.1016/s0959-1524(03)00028-3.

[27] C. Niemann, M. Montel, and F. Thielecke. Development of an air data system for an unmanned research aircraft. In *Deutscher Luft-und Raumfahrtkongress*, 2014.

[28] M. Krings, B. Annighoefer, and F. Thielecke. Ultra - unmanned low-cost testing research aircraft. In *American Control Conference ACC*, 2013.

[29] B. Herrmann, J. Theis, and F. Thielecke. Nonlinear system identification of a UAV model with distributed aerodynamics and flexible structure. *CEAS Aeronautical Journal*, 14(3):661–677, jul 2023. DOI: 10.1007/s13272-023-00674-x.

[30] Martin R. Waszak and David K. Schmidt. Flight dynamics of aeroelastic vehicles. *Journal of Aircraft*, 25(6):563–571, June 1988. ISSN: 1533-3868. DOI: 10.2514/3.45623.

[31] A. Hjartarson, P. J. Seiler, A. Packard, and G. J. Balas. LPV aeroservoelastic control using the LPVTools toolbox. In *AIAA Atmospheric Flight Mechanics (AFM) Conference*. American Institute of Aeronautics and Astronautics, aug 2013. DOI: 10.2514/6.2013-4742.

[32] L. Wang. A tutorial on model predictive control: Using a linear velocity-form model. *Developments in Chemical Engineering and Mineral Processing*, 12(5-6):573–614, jan 2004. DOI: 10.1002/apj.5500120511.

[33] P. S. G. Cisneros, A. Sridharan, and H. Werner. Constrained predictive control of a robotic manipulator using quasi-LPV representations. *IFAC-PapersOnLine*, 51(26):118–123, 2018. DOI: 10.1016/j.ifacol.2018.11.158.

[34] P. S. G. Cisneros and H. Werner. A velocity algorithm for nonlinear model predictive control. *IEEE Transactions on Control Systems Technology*, 29(3):1310–1315, may 2021. DOI: 10.1109/tcst.2020.2979386.

[35] L. Rieck, B. Herrmann, F. Thielecke, and H. Werner. Efficient quasi-linear model predictive control of a flexible aircraft based on laguerre functions. In *2023 American Control Conference (ACC)*. IEEE, may 2023. DOI: 10.23919/acc55779.2023.10156329.

[36] A. G. Pandalaand Y. Ding and H.-W. Park. qpswift: A real-time sparse quadratic program solver for robotic applications. *IEEE Robotics and Automation Letters*, 4(4):3355–3362, 2019.

[37] Y. Wang and S. Boyd. Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, 18(2):267–278, mar 2010. DOI: 10.1109/tcst.2009.2017934.