CEAS-GNC-2022-044

CEAS EuroGNC 2022
"Conference on Guidance, Navigation and Control"
3-5 May 2022 @ Technische Universität Berlin, Germany

Toward Onboard Trajectory Optimization for Fuel-Saving Climb of Aircraft with Automatic Flight Control

Daniel M. Gierszewski Research Associate, Institute of Flight System Dynamics, Technische Universität München, 85748, Garching, Germany. daniel.gierszewski@tum.de

Vincent Fafard Graduate Student, Institute of Flight System Dynamics, Technische Universität München, 85748, Garching, Germany.

Felix Schweighofer Research Associate, Institute of Flight System Dynamics, Technische Universität München, 85748, Garching, Germany.

Patrick Piprek Dr.-Ing., Institute of Flight System Dynamics, Technische Universität München, 85748, Garching, Germany.

Moritz Speckmaier Research Associate, Institute of Flight System Dynamics, Technische Universität München, 85748, Garching, Germany.

Florian Holzapfel Institute Chair and Professor, Institute of Flight System Dynamics, Technische Universität München, 85748, Garching, Germany.

ABSTRACT

An approach to decrease fuel consumption during climb by using trajectory optimization and automatic flight path control is presented. The proposed concept prepares onboard generation of optimal flight trajectories which accounts for current atmospheric conditions, available aircraft performance, and operational constraints. The optimization algorithm is decoupled from the safety-critical flight control algorithms by means of a gateway which checks the feasibility of the generated trajectories. As part of a modular flight guidance and control system, a path controller for tracking the fuel-optimal trajectories is presented. The approach is demonstrated in simulation using a closed-loop model of a high-altitude utility aircraft. The aircraft model is described along with the optimal control problem formulation. Simulation results demonstrate the fuel savings of the optimal climb mode compared to a standard autopilot climb procedure.

Keywords: Fuel-Optimal; Trajectory Optimization; Path Control; Autothrust

Nomenclature

A	=	Aerodynamic Reference Frame / Aerodynamic Force or Motion
E	=	Earth Centered Earth Fixed Reference Frame
K	=	Kinematic Reference Frame / Kinematic Force or Motion
N	=	Navigation Reference Frame
O	=	North-East-Down (NED) Frame

P	=	Propulsive Force
cmd	=	Command
est	=	Estimation
meas	=	Sensor Measurement
opt	=	Optimal
SP	=	Short-Period Mode
x	=	Position in Longitudinal Direction
z	=	Downward Position
h	=	Altitude
α	=	Angle of Attack
γ	=	Flight Path (Climb) Angle
χ	=	Course Angle
D	=	Drag Force
L	=	Lift Force
f	=	Specific Force
T	=	Thrust Force
C_L	=	Lift Coefficient
C_D	=	Drag Coefficient
$\overline{q}$	=	Dynamic Pressure
$\mathbf{M}$	=	Rotation Matrix
$\vec{\mathbf{F}}$	=	Force Vector
X, Y, Z	=	Force components
$(\vec{\mathbf{V}})^E$	=	Velocity Vector w.r.t. E-frame
V	=	Velocity magnitude
u, v, w	=	Velocity components
CAS	=	Calibrated Airspeed
IAS	=	Indicated Airspeed
TAS	=	True Airspeed
W	=	Wind Speed
τ_P	=	Time Constant
k	=	Gain
g	=	Acceleration due to Gravity
S_{ref}	=	Reference Area
ρ	=	Air Density
t	=	Time
τ	=	Trajectory Running Parameter
m	=	Aircraft Mass
n	=	Load Factor
ω_0	=	Natural Frequency
ζ	=	Damping Ratio
ν	=	Pseudo Control
$\mathbf{x}$	=	State Vector
$\mathbf{u}$	=	Control Vector
J	=	Cost
$x_{\Delta\delta_T}$	=	Integrator State of Thrust Controller

1 Introduction

Reducing fuel consumption is of high interest in the aviation industry due to growing concerns about the environmental impact of aviation and the rising contribution of fuel costs to the total operating expenses. In the ICAO Continuous Climb Operations (CCO) Manual [1], the climb is identified as "the phase of operations using the highest rate of fuel use in flight". With the deployment of CCO, optimal climb profiles accounting for aircraft performance become possible. Optimizing vertical profiles has been subject to research for decades, e.g., [2–5]. A commonly applied method is to formulate and solve an Optimal Control Problem (OCP). Compared to *offline* optimization prior to flight, onboard optimization is favorable as it allows to account for

- current atmospheric conditions (e.g., wind, air density)
- changes in aircraft performance (e.g., varying mass, system degradation)
- varying mission objectives
- varying operational constraints (e.g., imposed by air traffic control)

However, common optimal control methods for aircraft trajectory optimization are computationally expensive and cannot guarantee convergence within a certain time horizon. This makes them unsuitable for real-time application onboard the aircraft. One strategy to overcome these issues is to solve relevant instances of the OCP offline and store the results onboard. Update techniques can then provide approximate solutions of perturbed problems in real-time during the flight. Jardin and Bryson [6] suggest a similar approach for optimal guidance in wind fields, which is based on neighboring optimal feedback control. Tsuchiya et al. [7] present a real-time optimization method for emergency landing trajectories minimizing either flight time or fuel consumption. The optimization is stopped either upon convergence or after a time limit. The solution is displayed on the flight instruments such that the pilots can check the trajectory while following it manually. Verhoeven et al. [8], present a real-time trajectory optimization algorithm to compute fuel-optimal descent flight paths, while complying with air traffic time constraints. Although the method includes acceleration strategies, convergence cannot be guaranteed and a nominal flight plan has to be provided as a backup. Nolan et al. [9] provide a proof-of-concept for solving optimal control problems onboard in real-time using indirect methods.

In this paper, a concept for a safe method for onboard trajectory optimization is proposed, which decouples the optimization task from the safety-critical flight control task and provides a reliable and deterministic fall-back function. An OCP is formulated such that it takes into account current environmental conditions and operational constraints as well as the dynamic and limitations of the closed-loop aircraft. The optimization results have to pass a feasibility check before being forwarded to the flight control system, which is used for automatic following of the fuel-optimal trajectory.

The paper is structured as follows: First, the proposed concept for onboard trajectory optimization is introduced in Section 2. In Section 3, the control environment is presented. In Section 4, the aircraft model for optimization is described in detail and the OCP is formulated. The automatic flight control approach for fuel optimal climb is described in Section 5. In Section 6, simulation results are shown and finally, in Section 7 concluding remarks are given.

2 System Concept for Onboard Optimization

The overall system, shown in Fig. 1, consists of a Flight Control Computer (FCC), which accommodates the safety-critical flight guidance and control software, and a mission computer on which the trajectory optimization is performed. As explained by Nolan et al. [9], there are two major problems with onboard optimization: the computational burden and the reliability of the solver to converge to a feasible solution within a time period acceptable for in-flight use. A further obstacle arises when it comes to certification: Many trajectory optimization problems of interest for in-flight application cannot be solved

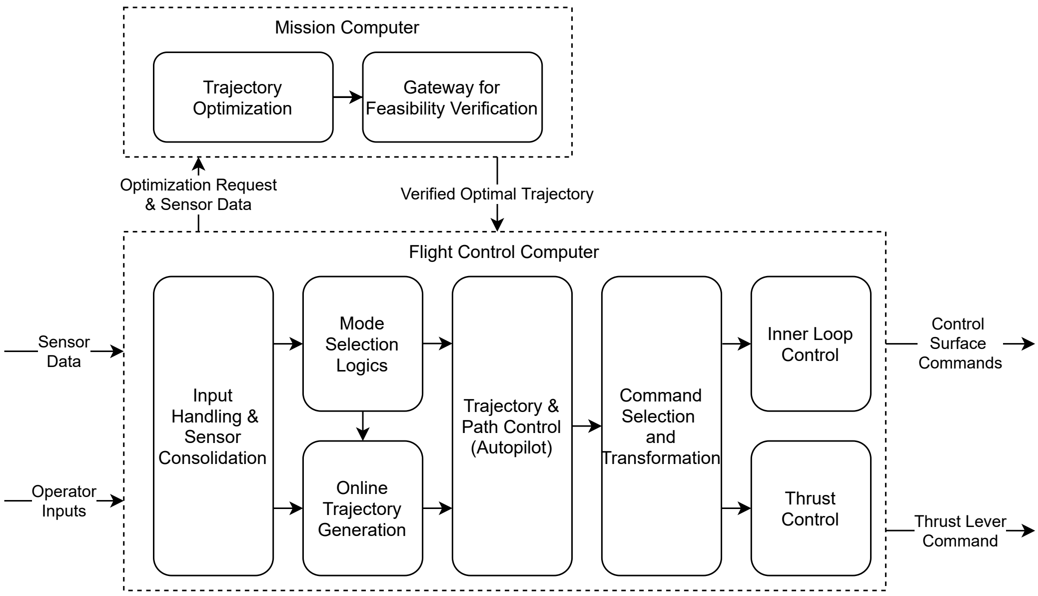

Fig. 1 Functional modules for online trajectory optimization and automatic flight control

analytically and their numerical solution requires iterative nonlinear optimization algorithms unsuitable for use in a safety-critical context. The certification of such solvers following current aerospace standards and guidelines is considered difficult and/or cost prohibitive. In the proposed system concept, the optimization algorithm is executed on a separate hardware, the mission computer, which provides the required computational power. Additionally, the optimization is functionally decoupled from the safety-critical part of the avionics via a *Gateway for Feasibility Verification*, which verifies optimal trajectories before forwarding them to the flight control functions on the FCC. All control input limitations and envelope protections of the flight control functions remain active while tracking the optimized trajectory to avoid unsafe flight conditions and to stay within the aircraft's operational envelope. The optimization is either triggered manually by the operator, e.g., by demanding a fuel-optimal climb, or automatically by the trajectory generation module if automatic flight guidance for lateral and/or vertical navigation is active. On demand, the FCC sends an *Optimization Request* including information on the required maneuver and associated *Sensor Data* for initialization to the mission computer, which solves the optimization task asynchronously and returns the result via the safety gateway back to the FCC. If the FCC receives a verified feasible trajectory from the gateway within a given time period, it engages the trajectory following mode. Otherwise, a safe sub-optimal fall-back strategy is applied, for example when the optimization fails to converge or feasibility cannot be verified. For an optimal climb maneuver, the fall-back strategy is a standard flight level change performed with a fixed climb thrust setting and the speed-by-pitch control mode of the autopilot. In case of automatic flight guidance, e.g., for automatic flight plan flying, the fallback mechanism is to use non-iterative, deterministic trajectory planning methods, e.g, [10, 11].

3 Control Environment

The modular flight guidance and control system developed at the Institute of Flight System Dynamics, see Fig. 1, constitutes the environment in which the proposed concept for onboard trajectory optimization and automatic trajectory flight is to be applied. The functional algorithms of the flight guidance and control system are executed on the FCC. The *Input Handling and Sensor Consolidation* module monitors signal integrity and processes signals from sensors and Human-Machine Interface (HMI) devices before use in the functional flight control modules. Based on operator inputs via the HMI, the *Mode Selection Logics* module [12] determines the internal operating mode of the automatic flight control system. The *Online Trajectory Generation* module [11] calculates commands for the trajectory controller [13] to enable automatic way-point based trajectory flight. The path controller [14] provides state-of-the-art autopilot functions. It is integrated with the trajectory controller in the *Trajectory & Path Control* module which provides the outer-loop control on a kinematic level. A *Command Selection and Transformation* module [15] transforms the kinematic outer-loop commands into inputs for the configuration-specific *Inner Loop Control* [16] and *Thrust Control* modules commanding the physical actuators.

4 Generation of Fuel-Optimal Climb Trajectories

For demonstration, the proposed approach is applied to a CS-23 [17] class Grob G-520T single-engine turboprop aircraft with a wingspan of 33 m and a Maximum Take-Off Mass (MTOM) of 5170 kg. The aircraft has been equipped with a digital flight control system within a research project, [18], to be operated as an optionally piloted vehicle platform by the company H3 HATS[1]. As this aircraft is designed for high altitude missions, fuel-efficient climb is of particular interest.

4.1 Simulation Model for Optimization

For climb trajectory optimization an extended point mass model of the G-520T in the vertical plane is used. A propulsion subsystem is included to compute fuel flow and thrust. Linear transfer functions represent the longitudinal inner-loop and thrust dynamics. The autothrust controller as well as the feed-forward paths of the outer-loop control law for climb angle and airspeed are incorporated in the model. This partially accounts for the closed-loop dynamics and thereby reduces tracking errors when the generated optimal control histories are used as command inputs to the path controller of the true auto-flight system.

4.1.1 Position Equations of Motion

The position is propagated in the local navigation frame N. Its origin is a fixed point at mean sea level and the orientation of the frame is derived by rotating the North-East-Down (O) frame by the desired kinematic course angle $\chi_{K,\mathrm{cmd}}$ about the z_O-axis.

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix}_N^E = \begin{bmatrix} V_K \cos \gamma_K \\ -V_K \sin \gamma_K \end{bmatrix}_N^E \tag{1}$$

Here, γ_K is the climb angle, V_K is the absolute kinematic velocity, x_N corresponds to the ground distance flown and z_N describes the negative value of the altitude h.

4.1.2 Translation Equations of Motion

The motion of a point mass aircraft in the vertical plane, described in the kinematic frame K under the assumption of a non-rotating, flat earth and quasi-steady mass change, is governed by [14]:

$$\dot{V}_K = \Delta f_{x,K} \qquad \dot{\gamma}_K = \frac{-\Delta f_{z,K}}{V_K} \tag{2}$$

[1]The H3 HATS, "High Altitude Technologies and Services" is a company of the H3 Aerospace Group concerned with the development, production, distribution and operation of high altitude platforms, so called HAP systems (High altitude platform systems). These systems are intended for various applications such as extension of telecommunication networks, research, agricultural monitoring, natural disaster monitoring and ensuring communication.

Here, the specific force increments Δf are defined as:

$$\Delta f_{x,K} := \frac{(X_A + X_P)_K}{m} - g\sin\gamma_K \qquad \Delta f_{z,K} := \frac{(Z_A + Z_P)_K}{m} + g\cos\gamma_K \tag{3}$$

The aircraft mass is denoted as m, g is the acceleration due to gravity, $(X_A)_K$, $(Z_A)_K$ are the aerodynamic force components and $(X_P)_K$, $(Z_P)_K$ represent the propulsion force components in the kinematic frame.

4.1.3 Aerodynamic Forces

The lift and drag forces are calculated in the aerodynamic reference frame A, whose x_A-axis is collinear to the aerodynamic velocity vector:

$$D = \overline{q} S_{\text{ref}} C_D = \overline{q} S_{\text{ref}} (C_{D0} + kC_L^2) \qquad L = \overline{q} S_{\text{ref}} C_L = \overline{q} S_{\text{ref}} (C_{L0} + C_{L\alpha}\alpha_A) \tag{4}$$

The dynamic pressure is given by $\overline{q} = \frac{1}{2}\rho V_A^2$. Since the model is restricted to the vertical plane, no lateral force is considered. The resulting aerodynamic force vector is transformed into the kinematic frame using the rotation matrix $\mathbf{M}_{KA}$ calculated from the aerodynamic and kinematic path angles γ_A, γ_K:

$$(\vec{\mathbf{F}}_A)_K = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix}_K = \mathbf{M}_{KA} \begin{bmatrix} -D \\ 0 \\ -L \end{bmatrix}_A \tag{5}$$

4.1.4 Propulsion Model

The propulsion model is based on table data representing the static characteristics of the G-520T turboprop engine. The thrust force and fuel flow are smoothly approximated by polynomials f_T, f_m in air density ρ, true airspeed V_A, and thrust lever position δ_T:

$$T = f_T(\rho, V_\text{A}, \delta_T) \qquad \dot{m}_f = f_m(\rho, V_\text{A}, \delta_T) \tag{6}$$

The dynamic of the engine is modeled by a first order lag,

$$\dot{T} = \tau_P^{-1}(T_{\text{cmd}} - T), \tag{7}$$

where τ_P is a time constant and T_{cmd} is computed from the power lever command $\delta_{T,\text{cmd}}$ by Eq. (6). It is assumed that the thrust force acts along the x_A-axis of the aerodynamic frame A. The resulting propulsive force vector $(\vec{\mathbf{F}}_P^G)_A = [T, 0, 0]_A^\top$ is transformed into the kinematic frame using the rotation matrix $\mathbf{M}_{KA}$. The differential equation for the aircraft mass is given by $\dot{m} = -\dot{m}_f$.

4.1.5 Linear Transfer Function for Closed-loop Rotation Dynamics

The short-period mode of the closed-loop aircraft is approximated by linear second-order dynamics for the aerodynamic Angle of Attack (AoA) α_A:

$$\frac{\text{d}}{\text{d}t}\begin{bmatrix} \Delta\alpha_A \\ \Delta\dot{\alpha}_A \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_{0,SP}^2 & -2\zeta_{SP}\omega_{0,SP} \end{bmatrix} \begin{bmatrix} \Delta\alpha_A \\ \Delta\dot{\alpha}_A \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_{0,SP}^2 \end{bmatrix} \Delta\alpha_{A,\text{cmd}} \tag{8}$$

Here, $\omega_{0,\text{SP}}$ is the natural frequency, ζ_{SP} is the damping ratio and $\Delta\alpha_{A,\text{cmd}}$ represents the deviation in AoA from the trim point. The variation of the parameters $\omega_{0,\text{SP}}$ and ζ_{SP} is smoothly approximated by polynomials based on straight and level trim conditions at varying static and dynamic pressures.

4.1.6 Command Mapping for the Linear Transfer Function

To match the feed-forward branch of the path controller, the aerodynamic AoA command, $\Delta\alpha_{A,\mathrm{cmd}}$, is computed from the lift force associated with a desired vertical path curvature $\dot{\gamma}_K$. Using the inversion control law of the outer-loop described in [14], a specific force command in the kinematic frame can be computed for a desired path curvature given by the pseudo-control $\nu_{\dot{\gamma}_K}$:

$$\Delta f_{z,K,\mathrm{cmd}} = -V_K \nu_{\dot{\gamma}_K} \tag{9}$$

The specific force command is transformed into the aerodynamic frame A using the rotation matrix $\mathbf{M}_{AK} = \mathbf{M}_{KA}^{\mathsf{T}}$:

$$(\Delta\vec{\mathbf{f}}_{\mathrm{cmd}})_A = \mathbf{M}_{AK} \begin{bmatrix} 0 \\ 0 \\ \Delta f_{z,K,\mathrm{cmd}} \end{bmatrix}_K \tag{10}$$

The resulting z_A-component of $(\vec{\mathbf{f}}_{\mathrm{cmd}})_A$,

$$\Delta f_{z,A,\mathrm{cmd}} = -\frac{\Delta L}{m} = \Delta f_{z,K,\mathrm{cmd}}(\sin\gamma_A \sin\gamma_K + \cos\gamma_A \cos\gamma_K), \tag{11}$$

can be used to calculate the AoA command applying Eq. (4):

$$\Delta\alpha_{A,\mathrm{cmd}} = -\frac{m}{\bar{q} S_{\mathrm{ref}} C_{L\alpha}} \Delta f_{z,K,\mathrm{cmd}} \cos(\gamma_A - \gamma_K) \tag{12}$$

4.1.7 Speed and Thrust Control

The speed control loop of the considered flight guidance and control system issues an aerodynamic linear specific force command $\Delta f_{x,A,\mathrm{cmd}}$ to the thrust control loop which controls the linear aerodynamic acceleration of the aircraft via the thrust lever setting $\delta_{T,\mathrm{cmd}}$. Assuming a constant wind vector, one can derive the linear aerodynamic acceleration

$$\dot{V}_A^G = \Delta f_{x,A}. \tag{13}$$

Inversion of Eq. (13) yields the speed control law [19]

$$\Delta f_{x,A,\mathrm{cmd}} = \nu_{\dot{V}_A} \tag{14}$$

with the pseudo control $\nu_{\dot{V}_A}$ for the desired aerodynamic acceleration. According to [14], the required thrust change for a desired acceleration is given by

$$\Delta T = m(\Delta f_{x,A,\mathrm{cmd}} - \Delta f_{x,A}). \tag{15}$$

The required change in thrust $\Delta\delta_{T,\mathrm{cmd}}$ is calculated by dividing the desired thrust change ΔT by the maximum available thrust. The final thrust command $\delta_{T,\mathrm{cmd}}$ is then obtained by applying a proportional-integral control law regulating the required change in thrust. The integrator part introduces another state in the optimization model, which is denoted as $x_{\Delta\delta_T}$. The thrust, speed, and flight path angle control loops are shown as block diagrams in Fig. 2. A cross-feed from the vertical path control to the integrator of the thrust loop compensates for additional required thrust during vertical maneuvering. A detailed description of the speed and thrust controller can be found in [14].

4.1.8 Atmosphere

The International Standard Atmosphere (ISA) [20] approximates static pressure, temperature and density as functions of altitude. Deviations from ISA are taken into account by adjusting the respective reference values at mean sea level using available sensor data for altitude, outside air temperature and static pressure. For onboard trajectory optimization, the wind vector can be estimated from available sensor data, e.g., using model-based methods [21]. The wind velocity components in the vertical plane relevant for the climb maneuver are considered in the optimization according to

$$\left(\vec{\mathbf{V}}_A\right)_N^E = \left(\vec{\mathbf{V}}_K\right)_N^E - \begin{bmatrix} u_W \\ 0 \\ w_W \end{bmatrix}_N^E . \tag{16}$$

The wind is assumed to be constant.

4.2 Optimal Control Problem

In summary, the state and control vectors of the aircraft model described in Section 4.1 are:

$$\mathbf{x} = \left[x_N, z_N, \gamma_K, V_K, \alpha_A, \dot{\alpha}_A, T, x_{\Delta\delta_T}, v_{\dot{\gamma}_K}, v_{\dot{V}_A}, m\right]^\mathsf{T} \qquad \mathbf{u} = \left[\dot{v}_{\dot{V}_A}, \dot{v}_{\dot{\gamma}_K}\right]^\mathsf{T} \tag{17}$$

Note that the first order time derivatives of the pseudo control variables are used as control vector while the actual pseudo control variables $v_{\dot{\gamma}_K}$ and $v_{\dot{V}_A}$ are defined as states in the OCP. This results in a twice continuously differentiable time history for the optimal pseudo controls as required for the feed-forward commands in the path control loop. By this, it is ensured that the feed-forward term for vertical path curvature yields a sufficiently smooth body load factor command which corresponds to a step-free elevator deflection command. The resulting longitudinal acceleration feed-forward term yields a step-free thrust command.

The objective is to minimize a weighted sum J of the total fuel consumption and an integral quadratic penalty on $\dot{\alpha}_A$:

$$J = w_m\left(m(t_0) - m(t_f)\right) + w_{\dot{\alpha}} \int_{\tau=t_0}^{t_f} \dot{\alpha}_A^2(\tau)\,\mathrm{d}\tau, \tag{18}$$

$$w_m = 10\,\mathrm{kg}^{-1}, \qquad w_{\dot{\alpha}} = 0.5\,\mathrm{s\,rad}^{-2} \tag{19}$$

The quadratic Lagrange cost term for $\dot{\alpha}_A$ reduces oscillations in $\dot{\alpha}_A$ observed at the end of the optimal climb trajectories. It is negligibly small compared to the fuel cost term.

The trajectory is subject to the path constraints

$$\begin{cases} \gamma_{K,\min} \le \gamma_K(t) \le \gamma_{K,\max} & \dot{\gamma}_{K,\min} \le v_{\dot{\gamma}_K}(t) \le \dot{\gamma}_{K,\min} \\ \alpha_{A,\min} \le \alpha_A(t) \le \alpha_{A,\max} & \dot{V}_{A,\min} \le v_{\dot{V}_A}(t) \le \dot{V}_{A,\max} \\ \delta_{T,\min} \le \delta_{T,cmd}(t) \le \delta_{T,\max} & V_{\mathrm{CAS},\min} \le V_{\mathrm{CAS}}(t) \le V_{\mathrm{CAS},\max} \\ \delta_{T,\min} \le x_{\Delta\delta_T}(t) \le \delta_{T,\max} & n_{z,\min} \le n_z(t) \le n_{z,\max} \\ 0 \le x_N(t) \le s_{\max} & \end{cases} \tag{20}$$

where $\dot{\gamma}_{K,\min/\max}$ are calculated as a function of the vertical load factor limits $n_{z,\min/\max}$. These are either derived from the pilot operating handbook of the G-520T or set to the limit values used within the flight control laws. The maximum ground distance $s_{\max}$ needs to be chosen large enough to accommodate the climb maneuver. This constraint should remain inactive in most practical cases.

Besides, boundary conditions enforce that the maneuver starts and ends in a trimmed state at a the given Calibrated Airspeed (CAS). At the initial boundary, the flight path angle, altitude and mass are fixed to the values measured or estimated upon engagement of the optimal climb mode. The integrator state $x_{\Delta\delta_T}$ is required to match the thrust setting to prevent the optimization algorithm from choosing an arbitrary value unrelated to the actual aircraft state. At the final boundary, level flight at the target altitude is prescribed. The boundary conditions are given by:

$$\begin{aligned}
\gamma_K(t_0) &= \gamma_{K,\text{meas}} & \gamma_K(t_f) &= 0 \\
\dot{\gamma}_K(t_0) &= 0 & \dot{\gamma}_K(t_f) &= 0 \\
\dot{\alpha}_A(t_0) &= 0 & \dot{\alpha}_A(t_f) &= 0 \\
V_{\text{CAS}}(t_0) &= V_{\text{CAS,cmd}} & V_{\text{CAS}}(t_f) &= V_{\text{CAS,cmd}} \\
\dot{V}_K(t_0) &= 0 & \dot{V}_K(t_f) &= 0 \\
x_{\Delta\delta_T}(t_0) &= \delta_T(t_0) & & \\
z_N(t_0) &= -h_{\text{meas}} & z_N(t_f) &= -h_{\text{cmd}} \\
m(t_0) &= m_{\text{est}} & &
\end{aligned} \tag{21}$$

The OCP is modeled using FALCON.m[2] [22]. Applying a trapezoidal collocation scheme, the OCP is discretized and transcribed into a sparse Nonlinear Program (NLP), which is then solved numerically by the interior-point filter line-search algorithm Ipopt (Interior Point OPTimizer) [23], using the MUMPS [24, 25] sparse linear solver within Newton iterations. The toolchain runs in a Matlab environment. For future onboard application, C/C++ code for integration with Simulink is going to be generated.

5 Automatic Flight Control for Fuel-Optimal Climb

Conventional trajectory-following control aims at reducing the spatial deviations between the aircraft and a reference point on the desired trajectory. Conversely, in case of fuel-optimal climb geometrically accurate tracking of the target trajectory is not the primary goal. Displacements from the planned trajectory arise from external disturbances, model errors, neglected higher-order dynamics and environmental uncertainties. Wind uncertainties can be expected to have a particularly significant effect since neither spatial nor temporal variations in the wind field are considered in our OCP. Given that the fuel consumption strongly depends on the aerodynamic state of the aircraft and that the most prominent deviations are expected to be of an aerodynamic nature, we prioritize the tracking of the optimal aerodynamic state over geometric accuracy. Consequently, we describe the fuel-optimal trajectory as time series of the optimal aerodynamic flight path angle γ_A^* and True Airspeed (TAS) V_A^*.

5.1 Command Generation

As a result of the optimization, optimal time histories for $\nu_{\dot{\gamma}_K}^*$, γ_A^*, $\nu_{\dot{V}_A}^*$, and V_A^* are obtained as discrete data points referenced to the n-point time grid $\mathbf{t} = [t_0, \ldots, t_{n-1}]$. These cannot be used directly as control inputs, as the path controller needs to generate sufficiently smooth and step-free inner-loop commands, which the controlled aircraft is able to follow. For that reason, the input signals for the path controller are derived from the optimization result by cubic Hermite interpolation for a query time point τ_{ref}. The time point τ_{ref} represents the running parameter of the optimized trajectory and can be interpreted as the time which has passed during the optimal climb maneuver. In the ideal case, where the optimal trajectory is

[2] FSD OptimAL CONtrol tool for Matlab, `https://www.falcon-m.com/`

perfectly tracked, the evolution of the running parameter τ_{ref} would match the evolution of the real time, i.e., $\dot{\tau}_{\text{ref}} = 1$. In reality, deviations between the optimal trajectory and the actual flown trajectory occur for several reasons, e.g., due to neglected dynamics and modeling uncertainties in the optimization model, sensor errors, inaccurate wind estimates as well as disturbances. As a result, the aircraft will not be at its desired altitude given by the optimal vertical trajectory at a given point in time. Instead of compensating for this deviation by sub-optimal correction maneuvers, we accelerate or decelerate the evolution of the running parameter τ_{ref}:

$$\dot{\tau}_{\text{ref}} = 1 + k_\tau \left(\tau_{\text{est}} - \tau_{\text{ref}}\right) \tag{22}$$

Here, $k_\tau < 0$ is the error controller gain and τ_{est} is determined from the optimal altitude history and the current altitude measurement. The value of the running parameter τ_{ref} is then calculated using forward Euler integration of $\dot{\tau}_{\text{ref}}$ with the sample time of the flight control application software. This method is similar to the concept of Pseudo-Control Hedging (PCH), where a reference model is adapted to account for the actual plant response.

5.2 Controller Structure

The proposed control structure preserves the structure of the path controller from [14] with few modifications. The optimal pseudo controls $\nu^*_{\dot{\gamma}_K}$ and $\nu^*_{\dot{V}_A}$ are used as inputs to the inversion. Compared

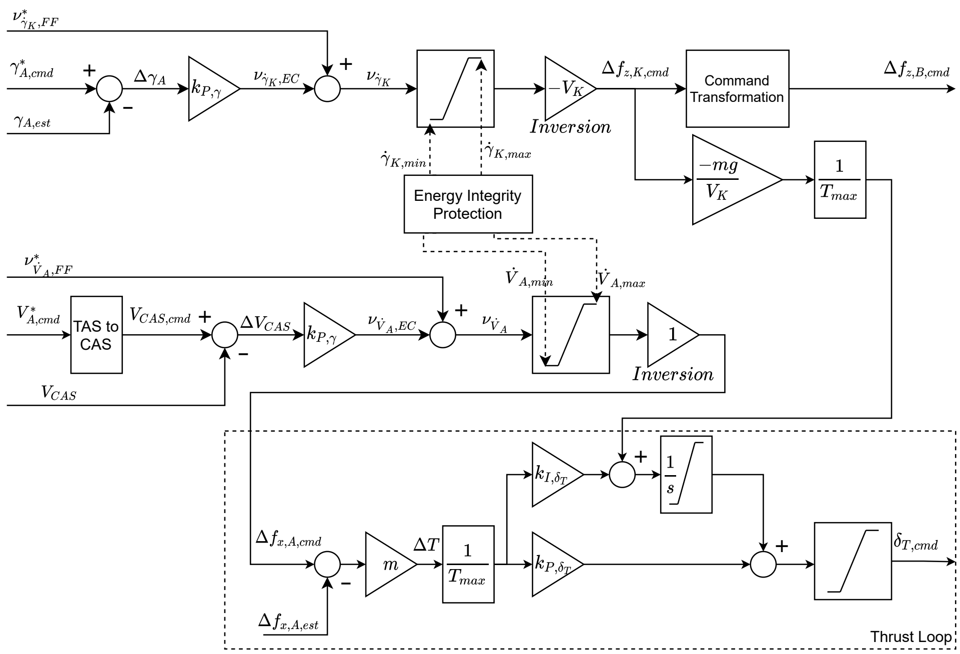

Fig. 2 Longitudinal path control structure with energy integrity protection and thrust controller

to [14], no reference model is used since the optimized time histories are already taking into account the desired aircraft dynamics and smooth pseudo-controls are guaranteed by the cubic interpolation. Due to the simplified model used for optimization as well as disturbances, sensor errors, etc., the real aircraft will deviate from the reference flight path. Proportional feedback controllers for flight path and speed regulate the tracking error between the optimal and estimated aerodynamic flight path angle and between the optimal and measured CAS. The conversion of the reference TAS profile to CAS allows the optimal climb mode to use the same speed control law as the other standard autopilot modes with active speed control, thereby avoiding switching transient issues and additional gain design and assessment effort.

5.3 Energy Integrity Protection

The nominal path controller which was extended to provide the optimal climb functionality incorporates active energy rate distribution prioritization to ensure airspeed integrity at the envelope boundary [19]. Originally, the reference values and pseudo controls for flight path angle and airspeed are produced by reference models and limited according to the energy integrity protection function. This function calculates lower and upper acceleration limits $\dot{V}_{A,\mathrm{prot,min/max}}$ as a function of the margin to the airspeed limits $V_{\mathrm{CAS,min/max}}$, as well as flight path angle limits $\gamma_{K,\mathrm{prot,min/max}}$, to prevent violations of the airspeed limits. In the optimal climb mode, which does not use the reference models, the acceleration limits $\dot{V}_{A,\mathrm{prot,min/max}}$ from the energy protection function can be directly applied to the pseudo control $\nu_{\dot{V}_A}$. The flight path angle limits $\gamma_{K,\mathrm{prot,min/max}}$ need to be converted to path curvature limits $\dot{\gamma}_{K,\mathrm{prot,min/max}}$ according to

$$\dot{\gamma}_{K,\mathrm{min/max}} = k_{\gamma,\mathrm{min/max}}(\gamma_{K,\mathrm{prot,min/max}} - \gamma_K). \tag{23}$$

The parameters $k_{\gamma,\mathrm{min/max}}$ determine the aggressiveness of approaching the speed envelope limit. The curvature limits are imposed on the path loop pseudo control $\nu_{\dot{\gamma}_K}$.

5.4 Altitude Capture

Except for the adaption of the reference time τ_{ref}, where the time estimate τ_{est} is determined from the current altitude, there is no actual feedback control of the altitude. In order to avoid high overshoots of the target altitude resulting from deviations of the reference time or disturbances, a dedicated altitude capture mode is activated once the kinematic flight path angle equals the capture flight path angle command

$$\gamma_{K,\mathrm{capture}}(\Delta h) = \frac{1}{V_K}\sqrt{2g(\Delta n_Z)_K \Delta h}. \tag{24}$$

The resulting capture trajectory is characterized by a constant incremental load factor $(\Delta n_Z)_K$ which is a controller design parameter. A larger value of $(\Delta n_Z)_K$ yields a faster capture maneuver and thus the optimal commands can be followed longer. However, a capture trajectory with a higher load factor increment reduces pilot comfort and increases control effort as well as the risk of reaching controller limits. For the considered utility aircraft, the capture load factor was set to 0.15 g. After capturing the altitude up to a given tolerance, altitude hold mode is activated and a different control loop regulates the deviation from the target altitude.

6 Simulation Results

The simulation results were obtained using a closed-loop simulation environment of the G-520T, based on the work by [26], which is used for design and verification of flight control software for the experimental auto-flight system of the G-520T. The simulation consists of a rigid-body flight dynamics model of the G-520T, actuation models, sensor models and a FCC model, which accommodates the flight guidance and control algorithms, as depicted in Fig. 1.

6.1 Optimal Climb Trajectory Tracking

To demonstrate the automatic tracking of the optimized trajectory, we simulated a climb from 5000 ft to 10 000 ft altitude. The optimal climb is initiated during steady-state horizontal flight at 95 KIAS, the recommended climb speed of the G-520T [27], with autopilot active in track hold, altitude hold and speed hold mode. The simulation is performed at 5 kts headwind, vertical updraft of 300 ft/min and no turbulence.

Figure 3 shows the simulation results. The aircraft accurately tracks the optimal references for the path angle $\gamma_{A,\mathrm{cmd}}$ and airspeed $V_{A,\mathrm{cmd}}$ and captures the target altitude with an overshoot of 21 ft. At $t = 144$ s the altitude capture mode engages, overriding the path loop command by $\gamma_{K,\mathrm{capture}}$ and resetting the speed command to the initial $V_{CAS,cmd}$. The inner-loop command variable, $(\Delta f_{z,\mathrm{cmd}})_B/g$, is step-free and shows acceptable tracking behavior. The thrust command activity at the start and end of the maneuver is higher than predicted by the optimization. After the capture maneuver, at $t = 150\,\mathrm{s}$, the autopilot switches to altitude hold mode.

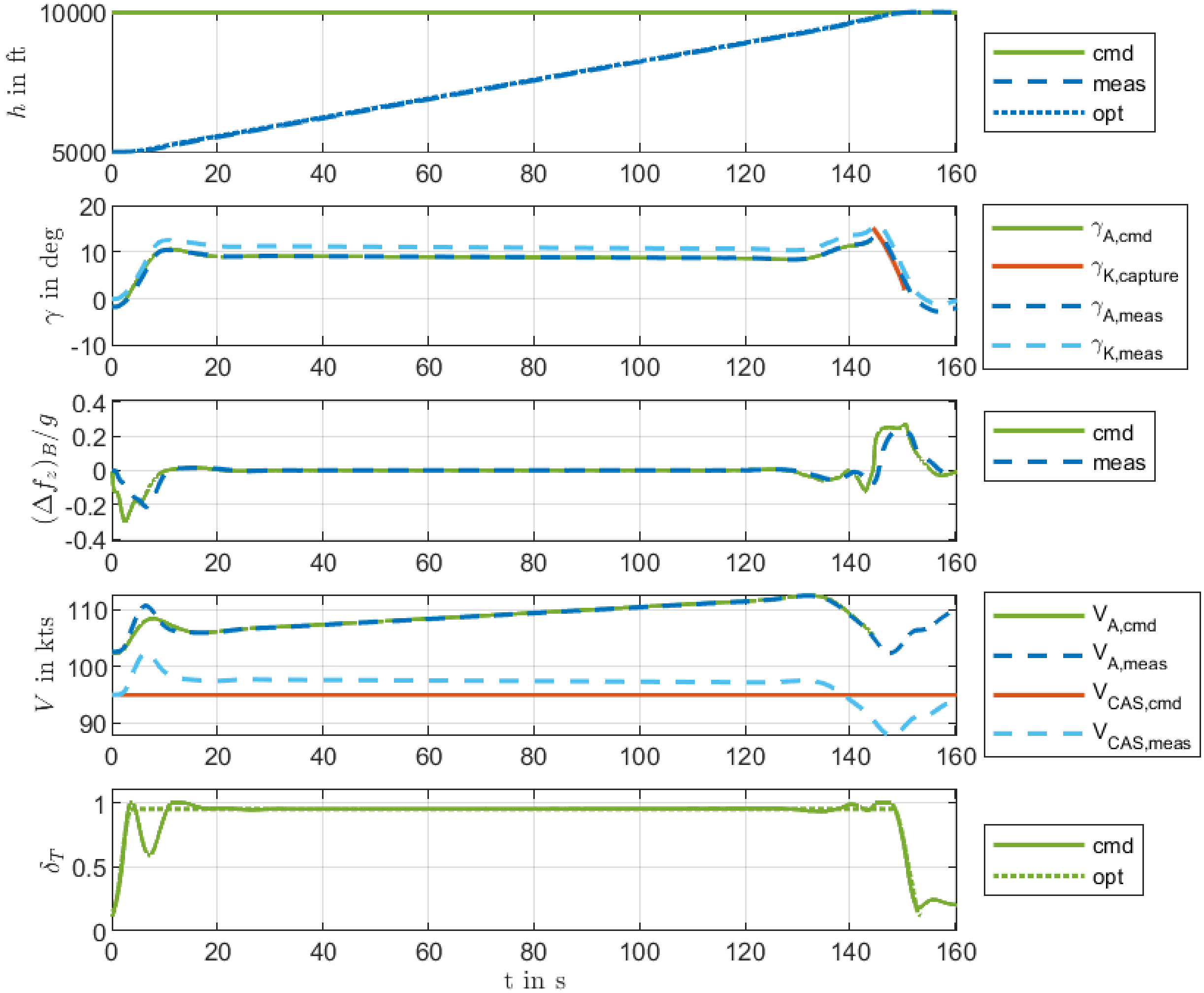

Fig. 3 Optimal climb trajectory tracking in presence of wind

6.2 Comparison to a Standard Autopilot Climb Procedure

As an in-depth evaluation of the expected fuel saving is beyond the scope of this paper, example scenarios are chosen to show the trend of the fuel saving potential. The benchmark for comparison is the standard autopilot flight level change function, which uses a speed-by-pitch controller to hold a constant airspeed with thrust at a fixed climb setting. Two scenarios are considered for comparison: a short climb from 5000 ft to 10 000 ft and a long climb from 1000 ft to 30 000 ft. In both scenarios, the initial speed is the recommended climb speed of 95 KIAS and there is no wind. The conditions at initiation of the climb maneuver are fixed and the climb maneuver is considered as completed when the autopilot engages altitude hold mode after capturing the target altitude. Figure 4 compares the trajectories obtained from the standard and optimal climb procedures for the long climb to 30 000 ft. While deviations in the flight path are most prominent at the start and end of the maneuver and relatively small in between, the speed profile exhibits crucial differences. Where the standard climb procedure tracks a constant CAS, the optimal climb is initially performed at a higher CAS which gradually decreases. The resulting higher average

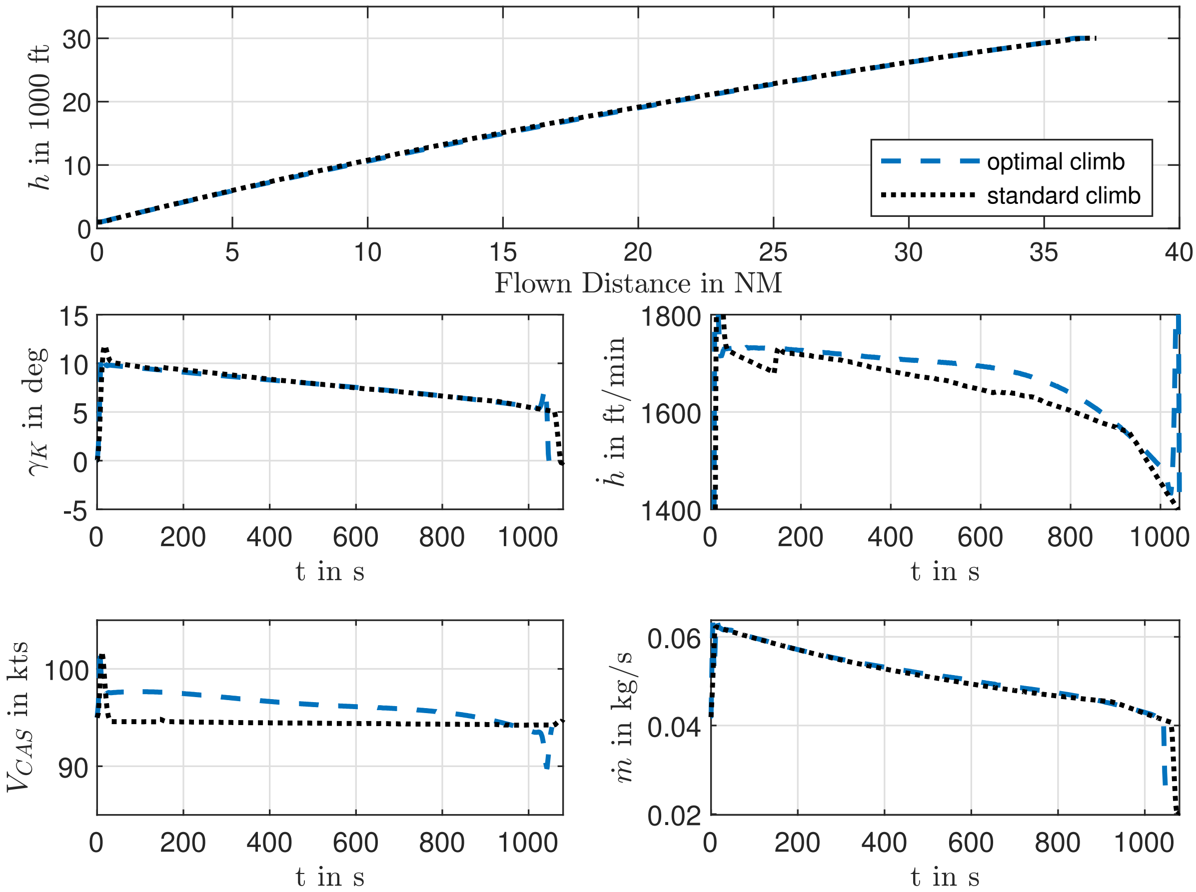

Fig. 4 Comparison of a flight level change with optimal climb control and reference speed by pitch control

climb rate yields a faster capture of the target altitude. In the optimal climb mode, the aircraft starts the capture maneuver at $t = 1040$ s and tracks the altitude at $t = 1054$ s, which is 16 s earlier compared to the benchmark. The faster climb, is found as the main cause for the fuel saving, as the required thrust and thereby the fuel flow drop strongly as soon as the capture maneuver is initiated. Table 1 lists the corresponding fuel savings compared to the standard flight level change. For a climb from 5000 ft to 10 000 ft both the optimization result and a closed-loop simulation of the optimal climb mode predict a reduction of fuel consumption by 2 % to 4 %. For a longer climb from 1000 ft to 30 000 ft the predicted fuel savings reduce to 1 % to 2 %, revealing that the standard climb procedure performed at the recommended climb speed is already close to the optimal solution. One reason for the gap between the optimal control result and the closed-loop simulation is the activation of the altitude capture mode which leads to a deviation from the optimal path at the end of the climb. Another reason, which explains the even higher gap for the longer climb, is the reduced quality of the polynomial fit to the fuel consumption data at high altitudes. Based on the expected increase in endurance, see Table 1, we conclude that the predicted fuel savings have only a small operational impact for the considered application. But when applying the proposed approach to a fleet of aircraft with large numbers of flights, we see a great potential to significantly reduce the environmental footprint and the total operating costs. This holds especially for short-distance flights where the climb segment represents a large share of the overall flight.

7 Conclusion and Future Work

The proposed concept for an optimal climb function prepares the application of onboard trajectory optimization methods in the autopilot of a CS-23 high-altitude utility aircraft. Simulations indicate a fuel-saving potential of about 1 % to 3 %, depending on the altitude difference. Further studies will be required to predict the fuel savings more accurately and consistently over a broad range of initial/final

Table 1 Predicted Fuel Savings Relative to Standard Climb Procedure

Scenario	Simulation Model	Saved Fuel		Expected Increase in Endurance
		in kg	in %	in s
5000 ft to 10 000 ft	Optimization Model	0.37	3.87	10
	Closed-loop Simulation	0.27	2.57	8
1000 ft to 30 000 ft	Optimization Model	1.28	2.34	64
	Closed-loop Simulation	0.55	1.01	28

altitudes and environmental conditions. The optimization approach should then be implemented on the mission computer of a G-520T to validate and refine the fuel-optimal climb mode in real flight tests. The computational efficiency and robustness of the optimization method should be assessed and improved, in particular for the planned onboard implementation. Alternative approaches to generate sufficiently smooth feed-forward signals might be incorporated, such as trigonometric series parametrizations [28] of the control histories in the OCP. The effect of uncertainties on the performance and reliability of the optimal climb mode should be assessed in future works. Relevant uncertainties include inaccurate wind estimates, aerodynamic coefficients, aircraft mass and propulsion models, among others. To fully exploit the fuel-saving potential in real-world scenarios, future work should account for cross-wind as well as for spatial and temporal variations of the wind field. While wind estimates obtained from onboard sensors may be valid in the vicinity of the aircraft, they cannot cover the long time horizon and altitude range of typical climb procedures. To improve on this, time-varying 3D wind forecasts from numerical weather models may be considered. By means of cyclic re-optimization during the climb, possibly supported by post-optimal sensitivity updates, the presented approach can be gradually refined. Ultimately, this may lead to a real-time Model-Predictive Control (MPC) method.

Author Contributions

System concept: D.G., F.H.; Optimization model and problem: P.P., D.G., V.F., F.S.; Optimization solver: P.P., F.S.; Flight control law: D.G., M.S.; Simulation: D.G., V.F.; Analysis: D.G., V.F., F.S.; Writing—original draft preparation: D.G.; Writing—review and editing: F.S., V.F., M.S., F.H., P.P., D.G.; Supervision, resources and project administration: F.H.;

Acknowledgments

Part of this research was funded by the German Federal Ministry for Economic Affairs and Energy (BMWi) within the Federal Aeronautical Research Program LuFo V-II through the project FADEAPC under grant no. 20Y1510C.

Supported by:

Federal Ministry
for Economic Affairs
and Climate Action

on the basis of a decision
by the German Bundestag

The authors would like to thank H3HATS (High Altitude Technologies and Services) for their valuable cooperation on this research project. Special thanks go to Robert Schmoldt, Frank Demmler, and Marius Koop from H3 HATS GmbH as well as Markus Ornigg (formerly H3 HATS GmbH), for their assistance and for providing crucial data and technical documentation.

References

[1] *Continuous Climb Operations (CCO) Manual*. International Civil Aviation Organization, 2013.

[2] James W. Burrows. Fuel optimal trajectory computation. *Journal of Aircraft*, 19(4):324–329, 1982. DOI: 10.2514/3.44756.

[3] Heinz Erzberger. Optimum climb and descent trajectories for airline missions. 1981.

[4] Matthew Sammut, David Zammit-Mangion, and Roberto Sabatini. *Optimization of Fuel Consumption in Climb Trajectories using Genetic Algorithm Techniques*. DOI: 10.2514/6.2012-4829.

[5] Ryota Mori. Fuel-saving climb procedure by reduced thrust near top of climb. *Journal of Aircraft*, 57(5):800–806, 2020. DOI: 10.2514/1.C035200.

[6] Matthew Jardin and Arthur Bryson. Neighboring optimal aircraft guidance in winds. *Journal of Guidance Control and Dynamics*, 24:710–715, 07 2001. DOI: 10.2514/2.4798.

[7] Takeshi Tsuchiya, Masahiro Miwa, Shinji Suzuki, Kazuya Masui, and Hiroshi Tomita. Real-time flight trajectory optimization and its verification in flight. *Journal of Aircraft*, 46(4):1468–1471, 2009. DOI: 10.2514/1.41709.

[8] Ronald Verhoeven, Ramon Dalmau-Codina, Xavier Prats, and Nico Gelder. Real-time aircraft continuous descent trajectory optimization with atc time constraints using direct collocation methods. *29th Congress of the International Council of the Aeronautical Sciences, ICAS 2014*, 01 2014.

[9] Sean M. Nolan, Clayton A. Smith, and Jacob D. Wood. Real-time onboard trajectory optimization using indirect methods. In *AIAA Scitech 2021 Forum*, Reston, Virginia, 01112021. American Institute of Aeronautics and Astronautics. DOI: 10.2514/6.2021-0106.

[10] Daniel Gierszewski, Volker Schneider, Patrick Lauffs, Lars Peter, and Florian Holzapfel. Clothoid-augmented online trajectory generation for radius to fix turns. *IFAC-PapersOnLine*, 51:174–179, 01 2018. DOI: 10.1016/j.ifacol.2018.07.029.

[11] Volker Schneider. *Trajectory Generation for Integrated Flight Guidance*. Dissertation, Technische Universität München, München, 2018.

[12] Christoph Krause and Florian Holzapfel. *System Automation of a DA42 General Aviation Aircraft*. DOI: 10.2514/6.2018-3984.

[13] Simon Schatz. *Development and Flight-Testing of a Trajectory Controller Employing Full Nonlinear Kinematics*. Dissertation, Technische Universität München, München, 2018.

[14] Erik Karlsson, Simon P. Schatz, Thaddäus Baier, Christoph Dörhöfer, Agnes Gabrys, Markus Hochstrasser, Christoph Krause, Patrick J. Lauffs, Nils C. Mumm, Kajetan Nürnberger, Lars Peter, Volker Schneider, Philip Spiegel, Lukas Steinert, Alexander W. Zollitsch, and Florian Holzapfel. Development of an Automatic Flight Path Controller for a DA42 General Aviation Aircraft. In *Advances in Aerospace Guidance, Navigation and Control*, pages 121–139, 2018. DOI: 10.1007/978-3-319-65283-2_7.

[15] Simon P. Schatz, Agnes C. Gabrys, Daniel M. Gierszewski, and Florian Holzapfel. Inner loop command interface in a modular flight control architecture for trajectory flights of general aviation aircraft. In *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 86–91. DOI: 10.1109/CoDIT.2018.8394801.

[16] Agnes Gabrys, Florian Holzapfel, Rasmus Steffensen, and Christian Merkl. *Flight Test Based Gain Tuning using non-parametric Frequency Domain Methods*. DOI: 10.2514/6.2021-1424.

[17] European Union Aviation Safety Agency (EASA). Certification Specifications for Normal-Category Aeroplanes (CS-23). Standard.

[18] Holzapfel, Florian and Schropp, Christopher and Gabrys, Agnes and Gierszewski, Daniel and Speckmaier, Moritz. Verbundvorhaben HOPLA - Hybrid optionally piloted long endurance aircraft; Teilvorhaben: Entwicklung eines hybriden Flugsteuerungssystems mit mechanisch-elektrischer Übertragung für den Einsatz in einem OPV : LuFo V-1 : Schlussbericht. Technical report, Technische Universität München, Lehrstuhl für Flugsystemdynamik, Garching bei München, 2019. DOI: 10.2314/KXP:1667449346.

[19] Erik Karlsson, Thaddäus Baier, Christoph Doerhoefer, Agnes Steinert, Markus Hochstrasser, Christoph Krause, Patrick Lauffs, Nils Mumm, Kajetan Nürnberger, Lars Peter, Simon Schatz, Volker Schneider, Philip Spiegel, Lukas Steinert, Alexander Zollitsch, and Florian Holzapfel. *Active Control Objective Prioritization for High-Bandwidth Automatic Flight Path Control*, pages 141–161. 01 2018. DOI: 10.1007/978-3-319-65283-2_8.

[20] ISO International Organization for Standardization. Standard atmosphere. Standard, 1975. ISO 2533-1975.

[21] Haichao Hong, Mengmeng Wang, Florian Holzapfel, and Shengjing Tang. Wind estimation for fixed-wing aircraft using command tracking approach. In *2018 26th Mediterranean Conference on Control and Automation (MED)*, pages 1–6, 2018. DOI: 10.1109/MED.2018.8443055.

[22] Matthias Rieck, Matthias Bittner, Benedikt Grüter, Johannes Diepolder, and Patrick Piprek. Falcon.m user guide. `https://www.falcon-m.com/`.

[23] Andreas Wächter and Lorenz T. Biegler. On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming*, 106(1):25–57, 2006. DOI: 10.1007/s10107-004-0559-y.

[24] Patrick R. Amestoy, Iain S. Duff, Jean-Yves L'Excellent, and Jacko Koster. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23(1):15–41, 2001. DOI: 10.1137/S0895479899358194.

[25] Patrick R. Amestoy, Alfredo Buttari, Jean-Yves L'Excellent, and Theo Mary. Performance and scalability of the block low-rank multifrontal factorization on multicore architectures. *ACM Trans. Math. Softw.*, 45(1), feb 2019. DOI: 10.1145/3242094.

[26] Alexander W. Zollitsch, Simon P. Schatz, Nils C. Mumm, and Florian Holzapfel. *Model-in-the-Loop Simulation of Experimental Flight Control Software*. DOI: 10.2514/6.2018-0425.

[27] *G-520 T "EGRETT" Pilot's Operating Handbook, Revision 2*. Grob Aircraft SE, 1997.

[28] Haichao Hong, Patrick Piprek, Rubens Junqueira Magalhães Afonso, and Florian Holzapfel. Trigonometric series-based smooth flight trajectory generation. *IEEE Transactions on Aerospace and Electronic Systems*, 57(1):721–728, 2021. DOI: 10.1109/TAES.2020.3008576.