

Interner Bericht

DLR-IB-FT-BS-2024-83

**Comparing Incremental
Nonlinear Dynamic Inversion
and Total Energy Control for
Morphing Wing UAVs**

Hochschulschrift

Eric Schaar

Deutsches Zentrum für Luft- und Raumfahrt

Institut für Flugsystemtechnik
Braunschweig



DLR

**Deutsches Zentrum
für Luft- und Raumfahrt**

Institutsbericht
DLR-IB-FT-BS-2024-83

Comparing Incremental Nonlinear Dynamic Inversion and Total Energy Control for Morphing Wing UAVs

Eric Schaar

Institut für Flugsystemtechnik
Braunschweig

036 Seiten
013 Abbildungen
003 Tabellen
010 Referenzen

Deutsches Zentrum für Luft- und Raumfahrt e.V.
Institut für Flugsystemtechnik
Abteilung Unbemannte Luftfahrzeuge

Stufe der Zugänglichkeit: I, Allgemein zugänglich: Der Interne Bericht wird elektronisch ohne Einschränkungen in ELIB abgelegt.

Braunschweig, den 12.06.2024

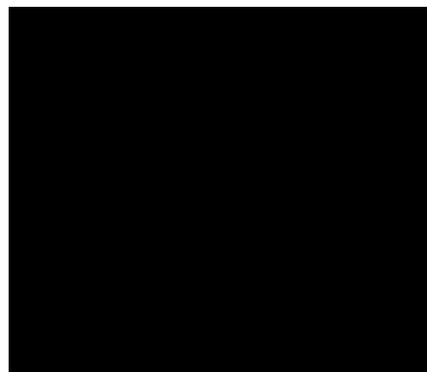
Unterschriften:

Institutsleitung: Prof. Dr.-Ing. S. Levedag

Abteilungsleitung: J. Dauer

Betreuer:in: L. Kracke

Verfasser:in: E. Schaar





Project Thesis

Comparing Incremental Nonlinear Dynamic Inversion and Total Energy Control for Morphing Wing UAVs

Eric Schaar

December 22, 2023

DLR Institut für Flugsystemtechnik
Prof .Dr.-Ing. Stefan Levedag

Supervisors:
Lennart Kracke
Dr. Mark Spiller

**Deutsches Zentrum
für Luft- und Raumfahrt e.V.**
in der Helmholtz-Gemeinschaft



Institut für
Flugsystemtechnik

DLR Institut für Flugsystemtechnik
Postfach 32 67, 38022 Braunschweig

Ihr Zeichen
Ihr Schreiben
Unser Zeichen

Ihr Gesprächspartner

Telefon 0531 295-
Telefax 0531 295-
E-Mail

01.10.2023

**Studienarbeit
für
Herrn Jerk Eric Schaar
Student an der Technischen Uni [REDACTED]**

Thema

Entwicklung und Implementierung einer Flugregelung zur automatischen Landung von Starrflügel-UAV

Erläuterung

Das interdisziplinäre Forschungsprojekt „Morphing Technologies & Artificial Intelligence Research Group“ (morphAIR) zielt darauf ab, die Verbesserung der Flugleistung durch formveränderliche Flügel zu demonstrieren und ihren Beitrag zur Effizienzsteigerung von Flächenflugzeugen zu bestimmen. Obwohl Morphing-Technologien in den letzten zehn Jahren in umfangreichen Boden- und Windkanaltests untersucht wurden, fehlen Flugtests zur Validierung dieser Ergebnisse und zur Bewertung der Veränderungen der Flugeigenschaften sowie der Flugleistungen. Darüber hinaus ist die Kombination einer auf künstlicher Intelligenz (KI) basierenden Regelung mit Morphing-Technologien nicht umfangreich erforscht, zeigt jedoch großes Potenzial für Verbesserungen der Flugsteuerung und -leistung.



Versuchsträger Proteus mit schematischer Darstellung potentieller Morphing-Elemente (Quelle: DLR)

Um diese wissenschaftlichen Ziele bei gleichzeitiger Risikominimierung zu erreichen, sind daher im morphAIR-Projekt Scaled Flight Tests mit dem unbemannten Luftfahrtsystem „Proteus“ geplant.

Bei den Flugtests ist die Landung aufgrund der veränderten Fahrwerkskonfiguration, des Wegfallens von Bremsklappen und des durch zusätzliche Mechanik und Flugtestinstrumentierung erhöhten Fluggewichts eine der kritischsten Phasen. Die hohe Landegeschwindigkeit und der flache Gleitwinkel steigern die Belastung der Sicherheitspiloten erheblich. Um die Sicherheit und Effizienz der Landungen zu verbessern, ist es von entscheidender Bedeutung, eine präzise und zuverlässige Flugregelung zu entwickeln, die eine automatische Landung ermöglicht.

Aufgabenstellung

Das Ziel dieser Studienarbeit besteht darin, eine Flugregelung für die automatische Landung von Starrflügel-UAVs zu entwickeln und zu implementieren. Die Landeregelung soll mithilfe von Simulink entwickelt und zunächst in einer der vorhandenen Simulationsumgebungen für den PX4-Autopiloten getestet werden. Die Flugregelung soll die Geschwindigkeit, den Gleitpfad und die Ausrichtung des UAVs während des Landeanflugs steuern, um eine sanfte und präzise Landung zu gewährleisten. Anschließend soll der Regler mithilfe der PX4-Simulink-Toolchain auf realer Hardware implementiert werden und in Flugtests auf mindestens zwei verschiedenen Flugzeugen erprobt werden.

Im Einzelnen sind folgende Punkte zu bearbeiten:

- Literaturrecherche zu vorhandenen Verfahren und Technologien zur automatischen Landung von Starrflügel-UAV (Regelstrategien, Navigationsmethoden, Sensortechnologien)
- Definition und umsetzen eines Konzepts für einen Landeregler in MATLAB/Simulink
- Erprobung des Reglers in eine der vorhandenen Simulationsumgebungen für den PX4-Autopiloten
- Übertragung des Reglers auf den PX4-Autopiloten und einer konkreten Hardwareumgebung
- Optionale Durchführung von Landeexperimenten mit Vergleich von Simulation und Experiment hinsichtlich Landegenauigkeit und Leistung des Reglers
- Dokumentation in einer wissenschaftlichen Form entsprechend den Vorgaben der TU Braunschweig zum Verfassen einer Studienarbeit und unter Berücksichtigung der Qualitätsvorgaben des DLR Instituts für Flugsystemtechnik

Literatur

Die Literaturliste dient lediglich einer ersten Orientierung und erhebt keinen Anspruch auf Vollständigkeit.

- Brockhaus, R., Alles, W., & Luckner, R. (2011). Flugregelung. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-01443-7>
- You, D. I., Jung, Y. D., Cho, S. W., Shin, H. M., Lee, S. H., & Shim, D. H. (2012). A Guidance and Control Law Design for Precision Automatic Take-off and Landing of Fixed-Wing UAVs. In AIAA Guidance, Navigation, and Control Conference. AIAA Guidance, Navigation, and Control Conference. American Institute of Aeronautics and Astronautics. <https://doi.org/10.2514/6.2012-4674>
- Adamy, J. (2014). Nichtlineare Systeme und Regelungen. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-45013-6>
- Trittler, M. (2018). Automatic landing for fixed-wing unmanned aerial vehicles with optical sensors. Aachen, Germany: Shaker Verlag.



Bearbeitungszeit

3 Monate

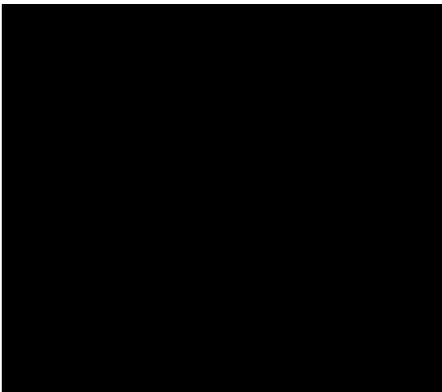
Betreuer

Lennart Kracke
Sebastian Cain

Ausgabe

_01.10.2023 _____

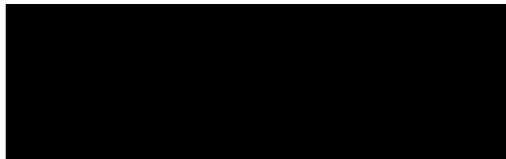
1.12.2023 _____



Statement of Originality

This thesis has been performed independently with the support of my supervisor/s. To the best of the author's knowledge, this thesis contains no material previously published or written by another person except where due reference is made in the text.

Braunschweig, December 22, 2023



Abstract

This paper explores the performance of a control system using incremental nonlinear dynamic inversion (INDI) for the longitudinal control of the morphing wing drone *Proteus* and compares it to a total energy control system (TECS). The main factors investigated are the decoupling of vertical flight path and horizontal speed, the tracking performance of a landing profile, and the robustness against changes in flight dynamics, caused by the morphing wing. Both systems showed similar and satisfying decoupling of speed and vertical flight path. The INDI system slightly outperformed the TECS in its path-tracking ability and was slightly more robust against perturbations of the eigendynamics. Both systems are considered suitable for the longitudinal control of the drone, with the INDI having slightly superior performance and robustness. When also accounting for the high disturbance rejection capabilities for INDI control systems outlined in the literature, the system is considered a capable candidate for the control of the drone. A difficulty when investigating the stability of the INDI controller was the nonlinearity of the control system, as the computation of margins is not directly possible. The Monte-Carlo parameter sampling approach was useful for gaining a general impression of the robustness. The INDI required additional computation performance compared to the TECS. Optimization of the control allocation algorithm by adding an analytical solution to the weighted least squares problem did not improve the computation time required.

Contents

List of Figures	i
List of Tables	ii
Abbreviations and Formulas	iii
1 Introduction	1
2 Derivation of Control Strategies	2
2.1 Incremental Non-Linear Dynamic Inversion (INDI)	2
2.2 Total Energy Control System (TECS)	4
3 Controller and Simulation Setup	6
3.1 State-Space Model	6
3.2 Position and Rate Controller for INDI	8
3.3 Design and Tuning of the Rate Controller for the TECS	9
3.4 Tuning of TECS and path angle commander	10
3.5 Actuators and Sensors	11
4 Simulation	12
4.1 Decoupling of Vertical and Horizontal Velocity	12
4.2 Landing Approach	13
4.3 Robustness against Model Uncertainty	17
4.4 Computational Effort	17
4.5 Combination of a TECS with INDI	21
5 Conclusion	23
Bibliography	24

List of Figures

2.1	TECS block diagram	5
3.1	Flight stack of PX4 autopilot [7]	6
3.2	State-space model	7
3.3	Structure of outer loop controller	8
3.4	Structure and root locus of base controller	9
3.5	Bode diagrams of $\ H\ _\infty$ criteria	10
4.1	Velocity step influence on velocity, altitude, and energy rates	14
4.2	Altitude step influence on velocity, altitude, and energy rates	15
4.3	Landing path tracking without disturbances	16
4.4	TECS landing path tracking with $K_H = 0.5$	16
4.5	Influence of parameter variation on altitude step response of INDI controller	18
4.6	Influence of parameter variation on altitude step response of TECS	19
4.7	Box plot of computation times for (1) analytical algorithm, (2) numerical algorithm, and (3) analytical algorithm with outsourced inverse calculation	21

List of Tables

- 3.1 Tuning parameters of outer loop controller 8
- 3.2 Tuning parameters of pitch angle controller 10
- 3.3 Tuning parameters of pitch angle controller 11

Abbreviations and Formulas

Abbreviation	Meaning	
DLR	German aerospace center	
EKF	Extended Kalman Filter	
INDI	Incremental nonlinear dynamic inversion	
NDI	Nonlinear dynamic inversion	
TECS	Total energy control system	
UAV	Unmanned aerial vehicle	
WLS	Weighted least squares	

Formula	Meaning	Unit
\mathbf{A}	State space system dynamics matrix	
\mathbf{B}	State space input matrix	
\mathbf{C}	State space output matrix	
C	Cost	
\mathbf{D}	State space feedforward matrix	
D	Drag force	N
E	Total energy	J
g	Gravitation constant	N/kg
H	Altitude	m
K	Gain	
L	Energy distribution	N
m	Mass	kg
M	Moment coefficient	
q	Pitch rate	rad/s
s	Complex Laplace parameter	
T	Thrust force	N
\vec{u}	Input vector	
V	Airspeed	m/s
W_v	Pseudocontrol weighing factor	
W_u	Control input weighing factor	
X	Longitudinal force/drag coefficient	
Z	Vertical force/lift coefficient	
\vec{x}	State vector	
\vec{y}	Output vector	
α	Angle of attack	
γ	Flight path angle, weighing factor	rad
δ_F	Relative thrust command	
η	Elevator deflection	rad
θ	Pitch angle	rad
\vec{v}	Pseudocontrol vector	
ω	Angular frequency	rad/s

1 Introduction

The pursuit of enhanced fuel efficiency in modern aircraft design has been a longstanding objective, propelled further by the escalating global efforts to mitigate climate change. Morphing wing structures are a technology, that in previous studies has shown promising reductions in aircraft's fuel consumption by up to 5% [2]. While the concept has been extensively investigated in theoretical frameworks and small-scale wind tunnel experiments, its practical application in system-critical function and real-world performance remains relatively unexplored.

To bridge the gap between theoretical conjectures and practical applications, flight tests are essential to gather empirical data on fuel consumption and dynamic property changes resulting from morphing wing structures. In response to this imperative, the German Aerospace Center (DLR) has launched the Morphair project. Its goal is the development of the Proteus unmanned aerial vehicle (UAV), which will be equipped with a morphing trailing edge wing. This UAV shall be used as a first flight test demonstrator, to test a morphing wing aircraft under real conditions. Importantly, the utilization of the Proteus UAV is anticipated to yield substantial cost-effectiveness in comparison to traditional flight test aircraft. This cost efficiency arises from circumventing expenditures associated with pilot involvement, certification procedures, and overall development costs.

One of the foremost challenges inherent in the multidisciplinary design of the Proteus drone is the effective control of flight dynamics, particularly in managing the intricate interplay of the multiple actuators embedded in each wing for morphing purposes. Addressing this complexity, the design incorporates an artificial intelligence (AI)-based controller to regulate the UAV's flight dynamics. Furthermore, the control of elevator and thrust, particularly during the critical phase of landing, introduces new challenges due to the morphing-induced alterations in aerodynamic and dynamic properties.

In addressing these challenges, the Incremental Nonlinear Dynamic Inversion (INDI) control strategy emerges as a promising solution. Renowned for its disturbance rejection capabilities and its ability to operate with minimal system dynamics information [8], INDI provides a robust foundation for managing the complexities introduced by morphing wings.

This thesis aims to contribute to the evolving field of morphing wing technology by focusing on the development and analysis of an INDI landing controller for the Proteus morphing wing drone. Emphasis is placed on achieving the decoupling of airspeed and altitude changes, crucial for safe and controlled landings, and on effectively mitigating disturbances arising from the dynamic morphing process. The obtained results will be systematically compared against the established standard Total Energy Control Systems (TECS) utilized in PX4 autopilots, providing a comprehensive assessment of the INDI approach's efficacy in addressing the unique challenges posed by morphing wing unmanned aerial vehicles.

2 Derivation of Control Strategies

This chapter is dedicated to presenting the mathematical derivation of the evaluated controllers, specifically the Incremental Nonlinear Dynamic Inversion (INDI) control system and the Total Energy Control System (TECS). The principle of INDI is explained in section 2.1, while section 2.2 focuses on the theory behind the TECS. Their implementation in the simulation is shown in chapter 3.

2.1 Incremental Non-Linear Dynamic Inversion (INDI)

Incremental non-linear dynamic inversion (INDI) is a control strategy, showing promising disturbance rejection capabilities and low model uncertainty sensitivity [8, 10]. These attributes hold particular significance for the control of morphing-wing aircraft during the landing phase, given the continuous variation in their aerodynamic properties throughout flight.

INDI extends from the concept of non-linear dynamic inversion (NDI) control. The core concept of NDI is the differentiation of plant dynamics until a direct relationship between input and output is established. Differentiating the system's dynamic equations $\dot{\vec{y}} = \dot{\vec{x}}$ and $\dot{\vec{x}} = \mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u}$ under the premise of $\mathbf{C} = \mathbf{I}$ and $\mathbf{D} = \mathbf{0}$ results in:

$$\frac{d}{dt} (\vec{y} = \vec{x}) \Leftrightarrow \dot{\vec{y}} = \dot{\vec{x}} \quad (2.1)$$

$$\frac{d}{dt} (\dot{\vec{x}} = \mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u}) \Leftrightarrow \dot{\vec{y}} = \mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u} \quad (2.2)$$

After the first differentiation, a direct relationship between input and output is obtained by solving for \vec{u} , resulting in an equation, that specifies the required controls to obtain the desired output signal:

$$\vec{u} = \mathbf{B}^{-1}(\dot{\vec{y}} - \mathbf{A} \cdot \vec{x}) \quad (2.3)$$

The accuracy of this correlation is dependent on precise knowledge about the system dynamics. As these are expected to be significantly altered through adaptations of the morphing-wing, this relationship is not directly applicable for the control of the aircraft. INDI dissolves this dependency by using the incremental form of the equations of motion. The incremental form is achieved through a Taylor series approach:

$$\begin{aligned} \dot{\vec{y}} &= \mathbf{A} \cdot \vec{x}_0 + \mathbf{B} \cdot \vec{u}_0 \\ &+ \frac{\partial}{\partial \vec{x}} (\mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u})_{\vec{x}=\vec{x}_0, \vec{u}=\vec{u}_0} \cdot (\vec{x} - \vec{x}_0) \\ &+ \frac{\partial}{\partial \vec{u}} (\mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u})_{\vec{x}=\vec{x}_0, \vec{u}=\vec{u}_0} \cdot (\vec{u} - \vec{u}_0) \end{aligned} \quad (2.4)$$

This can be simplified into eq. (2.5), as the first term is the derivative of y at $t = t_0$ and by calculating the partial derivatives.

$$\dot{\vec{y}} = \dot{\vec{y}}_0 + [\mathbf{A}]_{\vec{x}=\vec{x}_0, \vec{u}=\vec{u}_0} \cdot (\vec{x} - \vec{x}_0) + [\mathbf{B}]_{\vec{x}=\vec{x}_0, \vec{u}=\vec{u}_0} \cdot (\vec{u} - \vec{u}_0) \quad (2.5)$$

When considering small time increments, the third term is of much higher significance for the system dynamics as the second term, because the control inputs directly influence $\dot{\vec{y}}$. The system states \vec{x} are themselves only an integration of the control surface inputs, making their influence negligible for sufficiently small time increments. When the output $\dot{\vec{y}}$ is reformulated into the pseudo control \vec{v} and $\delta\vec{u} = \vec{u} - \vec{u}_0$ is defined, the equation simplifies further and becomes invertible.

$$\begin{aligned}\vec{v} &= \dot{\vec{y}}_0 + [\mathbf{B}]_{\vec{x}=\vec{x}_0, \vec{u}=\vec{u}_0} \cdot \delta\vec{u} \\ \delta\vec{u} &= [\mathbf{B}]_{\vec{x}=\vec{x}_0, \vec{u}=\vec{u}_0}^{-1} \cdot (\vec{v} - \dot{\vec{y}}_0)\end{aligned}\quad (2.6)$$

This control law eliminates the necessity for precise knowledge of the system dynamics, relying only on accurate information on control effectiveness. It is further necessary to have an accurate signal of the derivative of the output vector $\dot{\vec{y}}_0$. For an aerial vehicle, this means measurements of (angular) accelerations are required. These are not directly measured by accelerometers and gyroscopes, so estimation of angular accelerations is required. In [8], the use of a linear predictive filter is suggested. Further methods include Kalman filtering and the use of neural networks. For complexity reasons, the sensor dynamics are not included in this work but need to be considered for real-world application. When neglecting the actuator dynamics, the transfer function of the closed-loop INDI system reduces to a low-pass filter [8]. The constant K is equal to the gain applied to the open-loop system.

$$\mathbf{G} = \frac{K}{K + s}$$

In this work, the INDI is implemented with the dynamic control allocation method [9]. Instead of directly inverting the control effectiveness like shown in eq. (2.6), the control output is calculated through the least square optimization problem. The following cost function is defined:

$$C(\vec{u}) = \|\mathbf{W}_u(\vec{u} - \vec{u}_d)\|^2 + \gamma\|\mathbf{W}_v(\mathbf{G}\vec{u} - \vec{v})\|^2 \quad (2.7)$$

The factor \mathbf{W}_u penalizes the deviation of control output of the INDI compared to the desired control output, \mathbf{W}_v the deviation of expected system input compared to the virtual controls, γ is a weighing factor between both optimization goals. This cost function is solved through the active set method developed by HÄRKEGÅRD [4]. The weighing factors can be used to prioritize certain control modes over others. In [9] this is used on a quad-rotor drone to prioritize roll and pitch commands over yaw commands, which are essential for maintaining a stable flight state. For the application on a fixed-wing drone, it seems reasonable to prioritize the pitch rate q over horizontal velocity V , as 1 rad/s of pitch rate is of much higher impact than 1 m/s of velocity deviation. Thus, the weighing factor for velocity is reduced to a heuristic value of 0.1 (see eq. (2.8)).

$$\mathbf{W}_v = \begin{pmatrix} 1 & 0 \\ 0 & 0.1 \end{pmatrix} \quad (2.8)$$

2.2 Total Energy Control System (TECS)

Currently, the PX4 autopilot system uses a total energy control system (TECS) as standard for fixed-wing aircraft configurations. Thus, this controller is used to compare the performance of the INDI-controlled system. The basic concept of the TECS was initially introduced by LAMBREGTS for BOEING in 1982 [5] to address the challenge of decoupling the vertical flight path and airspeed, inherent in conventional autopilot architectures.

In flight control architectures like cascade control, a commanded change in flight path angle results in a change of airspeed, as the potential energy is redistributed to the kinetic energy of the aircraft. TECS circumvents this coupling by not independently controlling speed and altitude; instead, it governs the total energy of the aircraft, as expressed by (eq. (2.9)).

$$E_T = \frac{1}{2}mV^2 + mgH \quad (2.9)$$

Through derivation and restructuring, the relative energy rate is calculated:

$$\frac{\dot{E}_T}{mgV} = \frac{\dot{V}}{g} + \frac{\dot{H}}{V} = \frac{\dot{V}}{g} + \gamma \quad (2.10)$$

Since the difference between thrust T and drag D is the acceleration force of the aircraft, thrust command, and energy rate have a simple relationship:

$$T - D = mg\left(\frac{\dot{V}}{g} + \gamma\right) \quad (2.11)$$

When the initial thrust is trimmed and the incremental form of the equation is considered, the required change in thrust is obtained:

$$\Delta T = mg\left(\frac{\dot{V}_e}{g} + \gamma_e\right) \quad (2.12)$$

With this relation, the total energy can be exclusively controlled through the thrust. Now, the elevator can be used to distribute the total energy between its kinetic and potential fraction. The division of energy is expressed through the energy distribution rate \dot{L} , directly controllable by the elevator.

$$\dot{L} = \frac{\dot{V}}{g} - \gamma \quad (2.13)$$

Using PI control, control laws for the elevator and thrust are formulated [3]. To avoid unwanted zeros, the proportional feedback uses only the total energy \dot{E} and the distribution rate \dot{L} , while the integral feedback uses the error signals \dot{E}_e and \dot{L}_e , resulting in the control laws in eq. (2.14) and eq. (2.15)

$$T_C = \left(\frac{K_{TI}}{s}\right) \cdot \left(\frac{\dot{V}_e}{g} + \gamma_e\right) + K_{TP} \cdot \left(\frac{\dot{V}}{g} + \gamma\right) \quad (2.14)$$

$$\eta_C = \left(\frac{K_{EI}}{s}\right) \cdot \left(\frac{\dot{V}_e}{g} - \gamma_e\right) + K_{EP} \cdot \left(\frac{\dot{V}}{g} - \gamma\right) \quad (2.15)$$

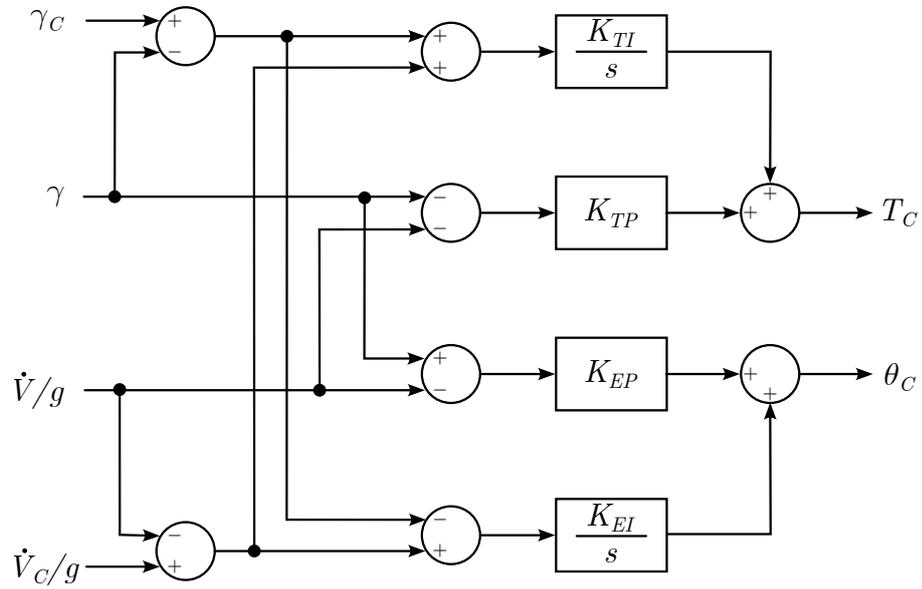


Figure 2.1: TECS block diagram

In the TECS controller however only \dot{L} and \dot{E} are used for the proportional feedback, to avoid unwanted zeros created by feedback of the error signals. The full control structure of the outer loop controller is displayed in fig. 2.1. The TECS requires an inner-loop controller for transforming the commanded pitch angle θ_C to an elevator command η .

3 Controller and Simulation Setup

The previous chapter described the mathematical derivation of both INDI and TECS. This chapter is dedicated to presenting their implementation in the simulation, including the required control structures and models. First, in section 3.1, the linear state-space model employed for tuning and testing the controllers is explained. The following section 3.2 and section 3.3 explain the control architectures used for INDI and TECS.

For both systems, the implementation of a rate and position controller is required. The general role of these controllers within an autopilot architecture is shown in fig. 3.1. A rate controller's function is to translate the commands given for the rates of the aircraft, in the case of longitudinal control for a fixed-wing plane the pitch rate q and velocity V into actuator commands. The position controller takes information from the navigator, in this case, the altitude H , and translates it into a rate command, then executed by the aforementioned rate controller. For simplification reasons, modelling of actuators and sensors and the required signal processing are beyond the scope of this work. However, in section 3.5, the consequences of non-ideal actuators and sensors and required steps regarding controller tuning and signal processing are briefly outlined.

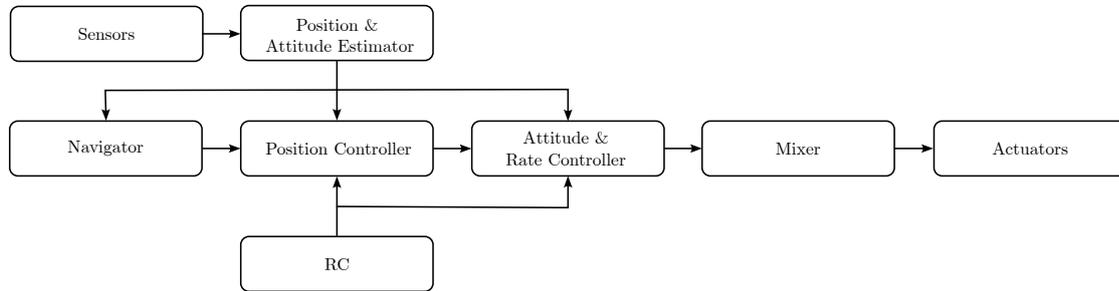


Figure 3.1: Flight stack of PX4 autopilot [7]

3.1 State-Space Model

To facilitate the testing and calibration of the controllers, a linear state space model is employed to represent the aircraft's flight dynamics. The state and output equations eq. (3.1) and eq. (3.2)

$$\dot{\vec{x}} = \mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u} \tag{3.1}$$

$$\vec{y} = \mathbf{C} \cdot \vec{x} + \mathbf{D} \cdot \vec{u} \tag{3.2}$$

are illustrated graphically by the block diagram depicted in fig. 3.2. Focusing exclusively on the longitudinal motion for this study, the state matrix \mathbf{A} and the input matrix \mathbf{B}

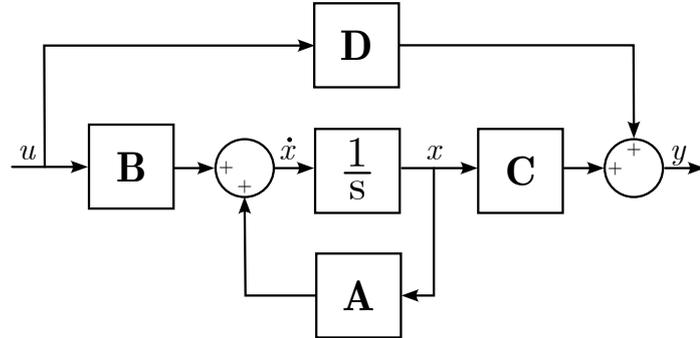


Figure 3.2: State-space model

are reduced to eq. (3.3)

$$\mathbf{A} = \begin{bmatrix} M_q & M_\alpha & M_u & 0 & 0 & 0 \\ 1 & Z_\alpha + Z_\theta & Z_u & Z_\theta & 0 & 0 \\ 0 & X_\alpha + X_\theta & X_u & X_\theta & 0 & 0 \\ 0 & Z_\alpha - Z_\theta & -Z_u & Z_\theta & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & V_{Tr} & 0 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} M_f & M_\eta \\ Z_f & Z_\eta \\ X_f & X_\eta \\ -Z_f & -Z_\eta \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \quad (3.3)$$

The state vector \vec{x} is composed of the pitch rate q , the angle of attack α , the velocity V , the path angle γ , the pitch angle θ and the altitude H . The two inputs in \vec{u} are the engine thrust δ_F and the elevator deflection η .

$$\vec{x} = (q \quad \alpha \quad V \quad \gamma \quad \theta \quad H)^T \quad \vec{u} = (\delta_F \quad \eta)^T \quad (3.4)$$

For the simulation, it is assumed that all states are directly measurable, rendering matrix $\mathbf{C} = \mathbf{I}$. No feedback of the input is applied, resulting in matrix

$$\mathbf{C} = \mathbf{I} \quad \mathbf{D} = \mathbf{0} \quad (3.5)$$

Given that the dynamic model of the Proteus drone was still in the developmental stage at the time of writing, coefficient data from BROCKHAUS [1], resembling a DORNIER DO 328, is adopted. As previously stated, it is assumed all states are measured and can be used for the control of the drone. Sensor behaviour is idealized, with dynamics, drifting, and noise disregarded. The actuators are also idealised, responding instantaneously to commands \vec{u} .

Both control architectures explained in the next sections (INDI in section 2.1 and TECS in section 2.2) do not directly control all state variables of the system, but instead rely on outer loop cascade structures. The variables directly controlled by both inner loop controllers are the pitch rate q and the velocity V . All further variables rely upon a hierarchical integration of these variables, subsequently being controlled by outer feedback loops.

3.2 Position and Rate Controller for INDI

INDI itself is not a controller, but a nonlinear feedforward, which inverts the system dynamics. Thus, both a rate and position controller are required upstream of the INDI. Different from the rate controller described in the introduction to this chapter, the function of the rate controller for the inverted system is not the generation of actuator signals, but the output of the pseudo-control vector \vec{v} , containing the derivatives of the pitch rate \dot{q} and airspeed \dot{V} . This rate controller uses a proportional gain, reducing the aircraft's dynamics to a PT1 system depending on the applied gain (see section 2.1). The attitude, in the specific case of the pitch angle θ , is also controlled with a proportional gain. To mitigate the risk of unsafe flight states and tail striking during landing, a limit is employed to the commandable pitch angle θ . For the control of the path angle γ within the position controller, a PID controller is used to achieve sufficient tracking performance and improve the transient behaviour. The path angle is derived by dividing the vertical speed by the airspeed. The vertical speed itself is computed based on the commanded altitude change, subject to an additional limitation to prevent exceeding the maximum climb or sink rate. The outer loop controller is heuristically tuned with the individual parameters listed in table 3.1

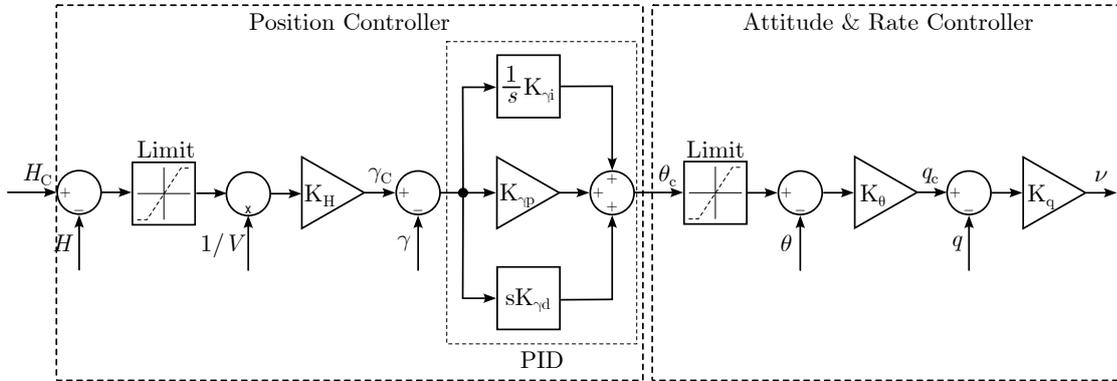


Figure 3.3: Structure of outer loop controller

Parameter	Value
K_q	5
K_θ	1.5
$K_{\gamma p}$	2
$K_{\gamma i}$	1
$K_{\gamma d}$	2
K_H	1

Table 3.1: Tuning parameters of outer loop controller

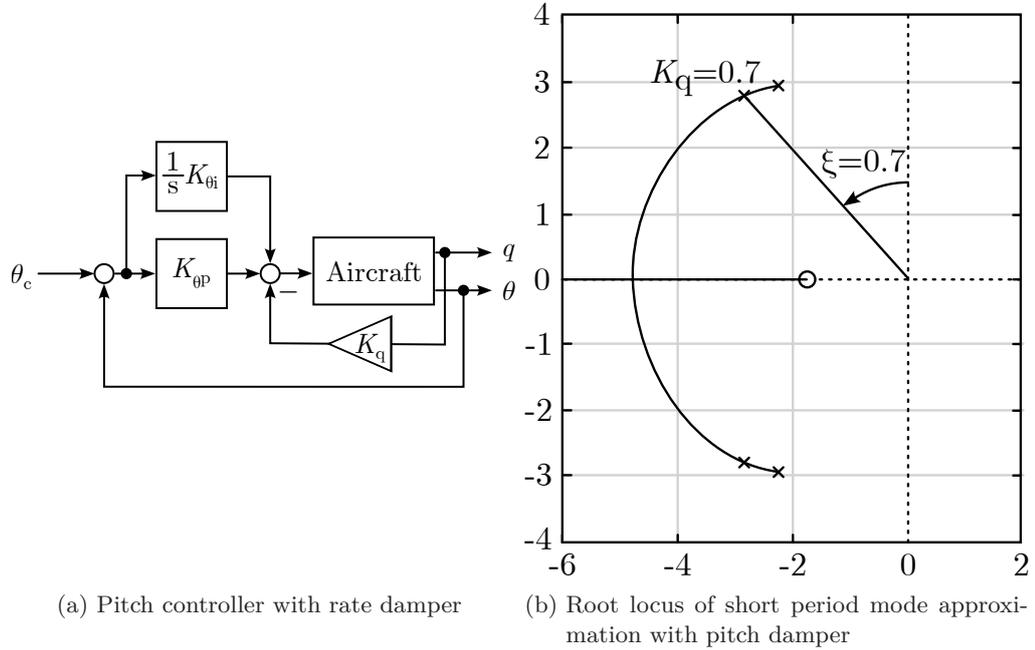


Figure 3.4: Structure and root locus of base controller

3.3 Design and Tuning of the Rate Controller for the TECS

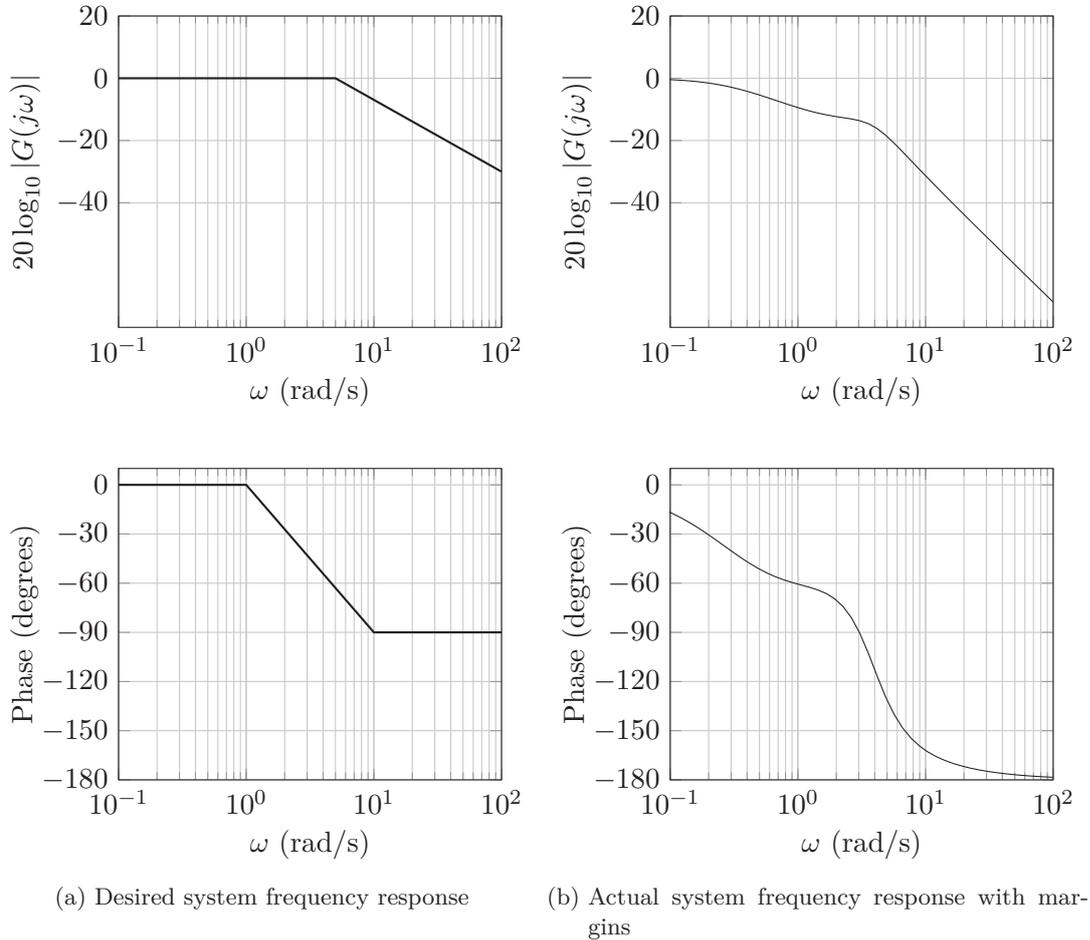
The Total Energy Control System takes the part of the position controller shown in fig. 3.1, thus requiring an inner control loop for attitude and rate control. However, the attitude and rate controller does not control the airspeed, as is the case for the INDI system, as this is handled already by the TECS (cf. fig. 2.1). The inner loop controller used for the TECS is composed of a pitch damper and a PI pitch angle controller (see fig. 3.4a). Tuning of the proportional pitch damper is done through the root locus method. For this, the state space model of longitudinal motion (eq. (3.3)) is reduced to the short-period mode approximation eq. (3.6), which only contains the states affected by aircraft's short-period mode oscillations q and α and the elevator input η .

$$\begin{pmatrix} \dot{q} \\ \dot{\alpha} \end{pmatrix} = \begin{bmatrix} M_q & M_\alpha \\ 1 & Z_\alpha + Z_\theta \end{bmatrix} \begin{pmatrix} q \\ \alpha \end{pmatrix} + \begin{bmatrix} M_\eta \\ Z_\eta \end{bmatrix} \eta \quad (3.6)$$

The root locus of this system with the pitch damper is then plotted, shown in fig. 3.4b. The gain is chosen to achieve the desired attenuation. In this case, an attenuation ξ of 0.7 is assumed appropriate, requiring a gain of $K_q = 0.1$. For the tuning of the PI-controller, the short-period-mode approximation from eq. (3.6) is extended by the pitch angle θ in eq. (3.7).

$$\begin{pmatrix} \dot{q} \\ \dot{\alpha} \\ \dot{\theta} \end{pmatrix} = \begin{bmatrix} M_q & M_\alpha & 0 \\ 1 & Z_\alpha + Z_\theta & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{pmatrix} q \\ \alpha \\ \theta \end{pmatrix} + \begin{bmatrix} M_\eta \\ Z_\eta \\ \theta \end{bmatrix} \eta \quad (3.7)$$

Two $\|H\|_\infty$ criteria are applied to this system to determine the tuning parameters. A PT1 behaviour with a rise time of 5 seconds is the first requirement, furthermore a gain

Figure 3.5: Bode diagrams of $\|H\|_\infty$ criteria

margin of 6dB and a phase margin of 45° is demanded. The desired frequency response is shown in fig. 3.5a, with the real frequency response shown in fig. 3.5b. As seen there, all margin requirements are fulfilled with a lower amplification of higher frequencies. This optimization results in the gains in table 3.3.

Parameter	Value
K_q	0.1
K_{θ_p}	-0.21
K_{θ_i}	$-2.7 \cdot 10^{-6}$

Table 3.2: Tuning parameters of pitch angle controller

3.4 Tuning of TECS and path angle commander

The tuning factors for the TECS are taken from the literature [3] and are listed in ???. To enable the TECS following altitude commands, a further control loop outside the TECS is implemented, to generate path angle commands from the altitude. This commander

loop follows a similar structure like for the INDI (cf. fig. 3.3). The gain K_H is also proposed by the literature as 0.1 [3].

Parameter	Value
K_{EI}	0.4
K_{TI}	0.4
K_{EP}	1
K_{TP}	1
K_H	0.1

Table 3.3: Tuning parameters of pitch angle controller

3.5 Actuators and Sensors

For simplicity, ideal actuators and sensors are assumed in this study. However, actuators and sensors have dynamic behaviour. The actuators define the closed-loop transfer function when INDI is applied, as their dynamics are not inverted [10]. INDI also requires the state of the actuator u_0 to determine the next actuator command u . While the elevator deflection can be measured directly with a potentiometer, the thrust is not trivially obtainable. Thus, a model-based estimation is required [10]. The sensors measuring the required state and derivated state vector (cf. section 2.1, section 3.2) are not ideal, thus delivering noisy and even drifting signals. For the fixed-wing drone, the state derivatives \dot{V} and \dot{q} are required. Especially the pitch acceleration signal, provided by the gyroscope, is expected to be noisy [10]. A second order filter eq. (3.8) was applied in [10] to reduce the noise. As this filter introduces a time delay, it has to be applied to the acceleration \dot{V} as well for both signals to be synchronous.

$$F(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (3.8)$$

SIEBERLING has further applied a linear predictive filter to the measurement data to eliminate the time-delay error [8].

The current control architecture uses feedback of the flight path angle γ . However, γ is not measured directly with the current sensor layout. Instead, feedback of the vertical speed \dot{H} could be used, as it can be estimated from the altitude and vertical acceleration using a complementary filter (eq. (3.9), [1])

$$\dot{H} = \frac{s}{1 + T_s} H_{baro} + \frac{T}{1 + T_s} \ddot{H} \quad (3.9)$$

At the time of writing, it is planned to implement the INDI controller on the PX4 firmware. Within the PX4 flight stack from fig. 3.1, an Extended Kalman Filter (EKF) is already implemented [7]. The EKF already implements, inter alia, estimations of speeds and turn rates, thus providing a potential alternative, which is already implemented within the software, to the aforementioned filters.

4 Simulation

In the present chapter, the performance and robustness against disturbances of the INDI controller and the TECS are compared. Firstly, it shall be investigated, whether the INDI controller can decouple airspeed and altitude sufficiently, as this is crucial for altitude changes of the aircraft, necessary during the landing approach. This is examined in section 4.1 and compared to the decoupling performance of the TECS from the literature. Furthermore, the UAV must be able to maintain a glide slope with minimal deviation for landing while also holding a steady velocity, examined in section 4.2. A central objective of this paper is to determine the control system's ability to mitigate the changes in dynamic properties induced through wing morphing. In section 4.3, an implementation of the disturbances into the flight physical model through parametric uncertainties with Monte-Carlo parameter sampling is described and the robustness of both controllers to these alterations is determined. The robustness against disturbing forces like wind is not investigated in this study, as within the existing literature on INDI, excellent rejection properties have already been identified [10]. As the execution of the iterative control allocation method requires potentially higher computational resources, the modifications of the algorithm are compared in section 4.4 regarding their required simulation time. In the last section of this chapter, a concept found in the literature [6], combining a TECS with INDI is critically examined, especially under consideration of the results from the study of decoupling performance of INDI in section 4.1.

4.1 Decoupling of Vertical and Horizontal Velocity

A fundamental challenge in the precise control of the longitudinal motion of an aircraft is the inherent coupling of airspeed and path angle. This coupling is due to the energy transfer between kinetic and potential energy, occurring in manoeuvres where the altitude is changing. Conventional control systems, which control airspeed and altitude separately, can show significant changes in airspeed, when the altitude is changing, as thrust will only be adjusted through an error in the aircraft's speed. The TECS controller was developed to precisely break up this coupling [5] by controlling the kinetic and potential energy instead of velocity and altitude. INDI is also decoupling the different inputs and outputs, as each virtual control ν_i only affects the output y_i . To compare the decoupling performances of both controller, the energy rates \dot{E} and \dot{L} introduced in section 2.2 are used. When a change in airspeed is commanded to the aircraft, the altitude should stay constant. This results in the total energy change \dot{E} and energy distribution rate \dot{L} being equal, when the change in airspeed does not cause any disturbances of

vertical flight path:

$$\dot{E} = \frac{\Delta V}{g} + \gamma \quad (4.1)$$

$$\dot{L} = \frac{\Delta V}{g} - \gamma \quad (4.2)$$

$$H = \text{const.} \implies \gamma = 0 \implies \dot{E} = \dot{L} = \frac{\dot{V}}{g}; \quad (4.3)$$

When a change in altitude is commanded and the airspeed shall remain constant, the total energy change \dot{E} is equal to the negative energy distribution rate \dot{L} .

$$V = \text{const.} \implies \dot{V} = 0 \implies \dot{E} = -\dot{L} = \gamma; \quad (4.4)$$

FALEIRO has demonstrated this measure in [3] for a TECS, showing that the remaining coupling of airspeed and altitude is minimal. When a step of 10m in altitude is commanded to the INDI-controlled system (cf. fig. 4.2b), a maximum velocity deviation of 0.08m/s is caused (cf. fig. 4.2a). Due to this low deviation in speed, the energy distribution rate \dot{L} resembles the inverse of the total energy rate \dot{E} closely (cf. fig. 4.2c). When a step in velocity of 10m/s is commanded, the maximum measured deviation in altitude is about 1m (cf. fig. 4.1b). In this case, \dot{L} and \dot{E} match each other closely (fig. 4.1c). Thus, the INDI control achieves a similar decoupling performance as achieved through a TECS in [3].

4.2 Landing Approach

To determine the controller performance during the landing approach of the morphing-wing aircraft, the ability to follow a landing approach path is determined. The commanded landing path is generated by applying a constant negative vertical acceleration to initiate the descent until the maximum vertical speed is reached. Before touchdown, a positive acceleration is used to reduce the vertical touchdown speed. The INDI controller accurately tracks the glideslope with only a minimal tracking error, shown in fig. 4.3a, while the TECS is subject to a significantly larger tracking error (cf. fig. 4.3c). This rather significant tracking error is due to the slower response behaviour of the TECS. The tracking performance can be improved by increasing the K_H parameter of the path angle command loop parameter of the TECS. Figure 4.4 shows the tracking performance with the K_H parameter increased from 0.1 to 0.5. The INDI also holds the horizontal speed accurately, even, when the vertical accelerations are applied, the error stays below 0.05m/s (cf. fig. 4.3b). The TECS also maintains airspeed accurately, with a maximum disturbance of less than 0.2m/s, when the constant K_H is increased to 0.5 for better path tracking, this deviation increases to about 0.4m/s. Thus, both controllers perform as expected from section 4.1 concerning their speed hold ability.

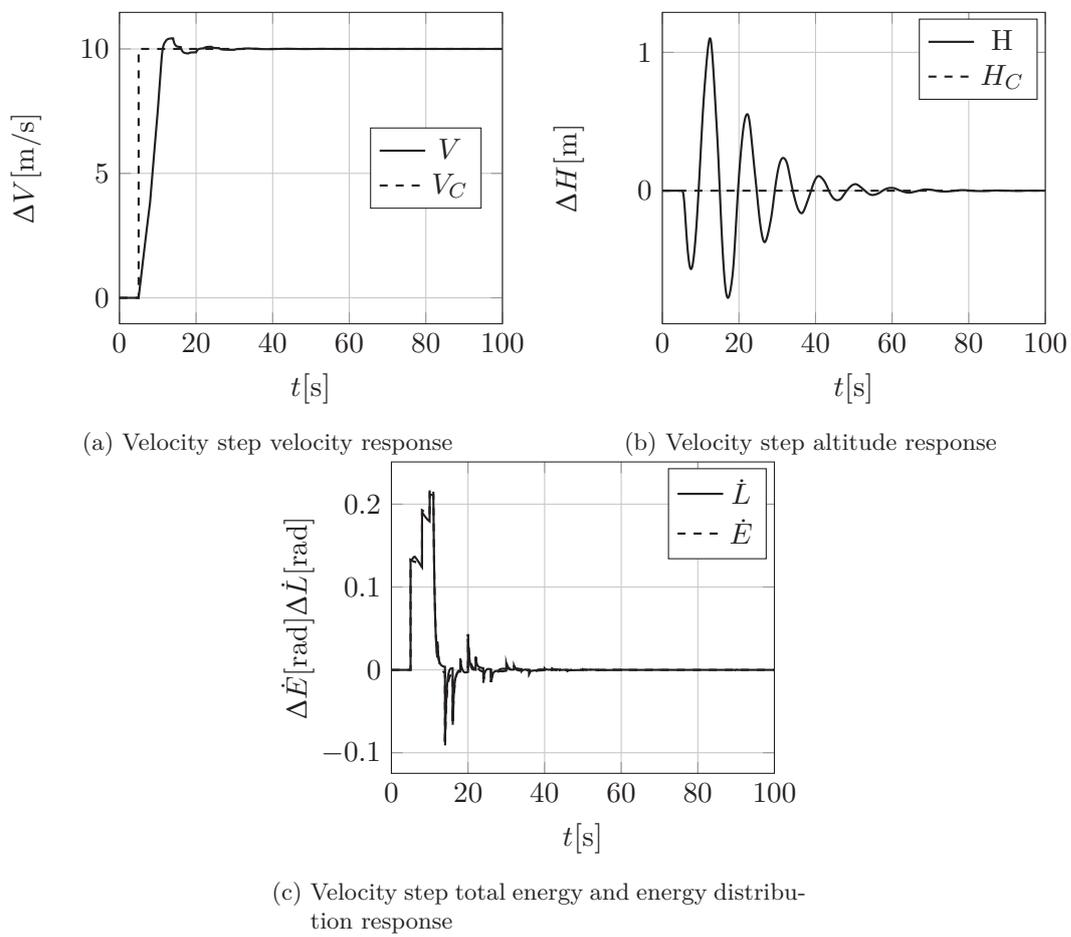
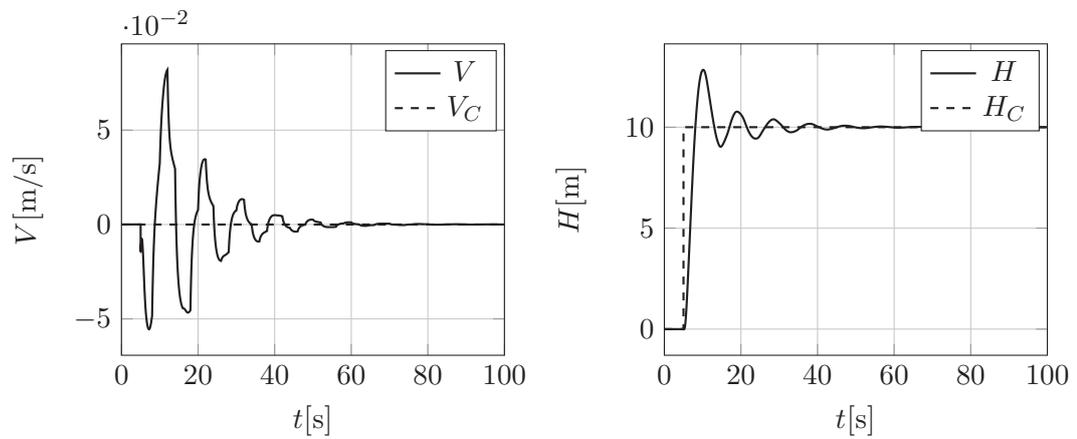
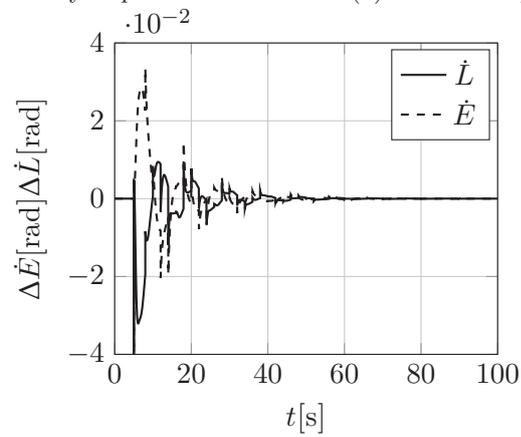


Figure 4.1: Velocity step influence on velocity, altitude, and energy rates



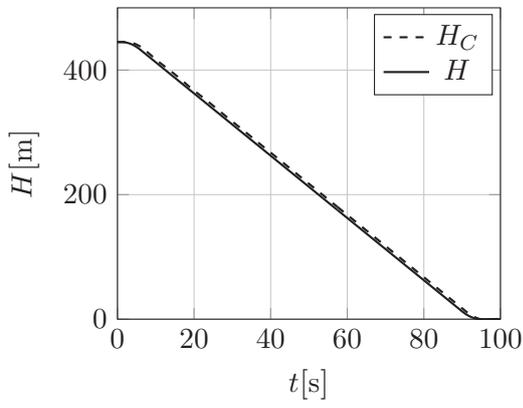
(a) Altitude step velocity response

(b) Altitude step altitude response

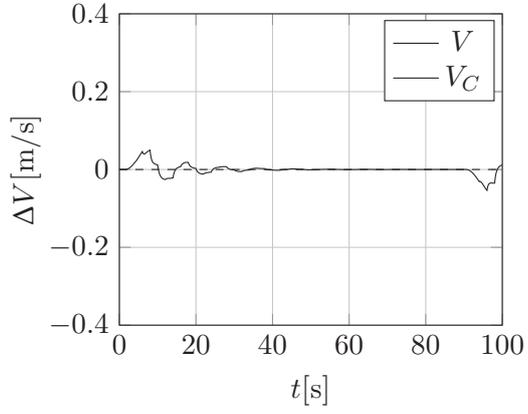


(c) Altitude step total energy and energy distribution response

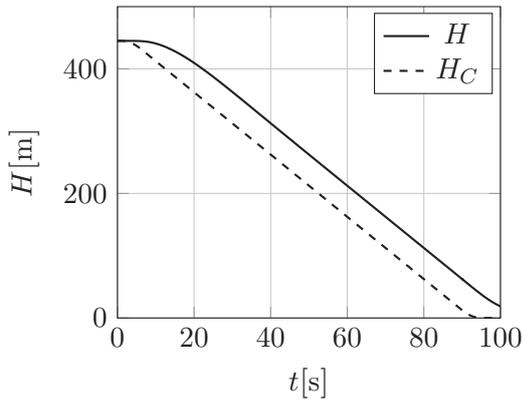
Figure 4.2: Altitude step influence on velocity, altitude, and energy rates



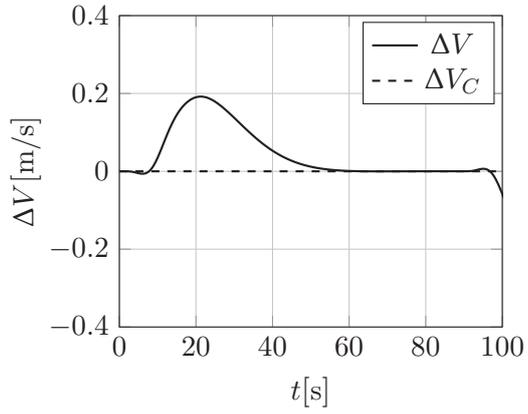
(a) INDI controller vertical flight path tracking



(b) INDI controller velocity hold

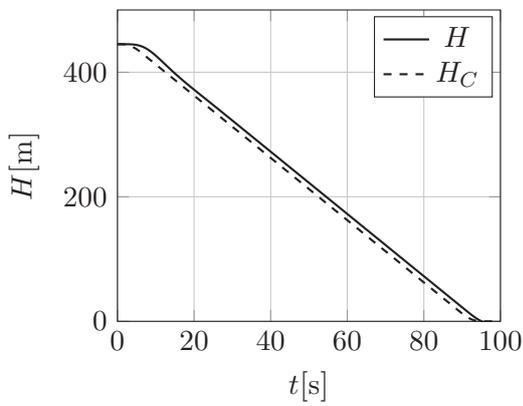


(c) TECS vertical flight path tracking

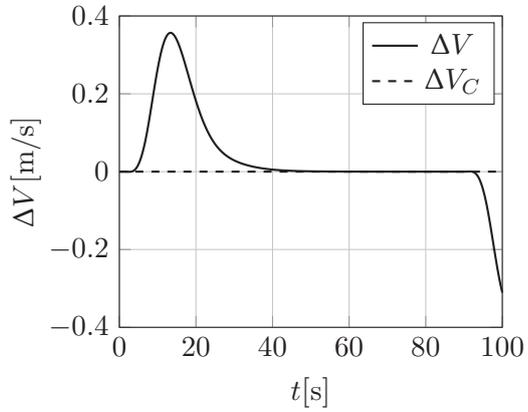


(d) TECS velocity hold

Figure 4.3: Landing path tracking without disturbances



(a) TECS vertical path tracking with $K_H = 0.5$



(b) TECS velocity hold with $K_H = 0.5$

Figure 4.4: TECS landing path tracking with $K_H = 0.5$

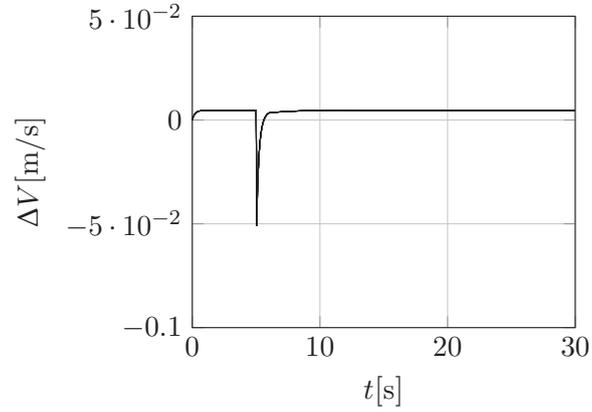
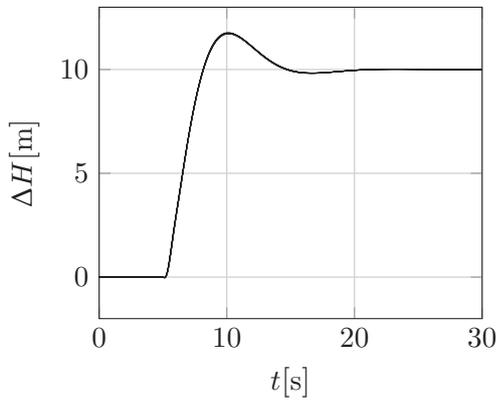
4.3 Robustness against Model Uncertainty

A key requirement for the longitudinal control of the morphing wing drone is high robustness against disturbance. Of particular importance is the ability to counteract the changes in flight dynamics, which are caused by the morphing wing. The effects of the morphing wing are not quantifiably known yet. It is assumed for this study, that aerodynamic parameters of lift and moment change through the morphing of the wing, while parameters related to drag stay relatively constant. In reality, parameter changes will most likely affect the dynamics of the aircraft, in the state-space model described by the matrix \mathbf{A} , without significant effects on the input dynamics denoted in \mathbf{B} . However, it is tested here, how well the controllers handle changes not only in the dynamic matrix \mathbf{A} , but also in the control effectiveness matrix \mathbf{B} . Due to the unknown changes in aerodynamics, the parameters subject to changes are modelled as unknown parameters. Monte Carlo parameter sampling is then applied to provide a general idea of the closed-loop behaviour. The test case used is a step in altitude, with a horizontal speed hold command. For each level of relative disturbance, 10 parameter variations are tested. Variabilities of up to 10%, 20%, and 50% of nominal value are tested for each controller. The INDI-controlled system does not show a significant spread in the step responses for all inaccuracy levels, as shown in fig. 4.5.

It has to be noted that a Monte Carlo parameter variation is only useful for a basic understanding of the system behaviour, when subject to model uncertainty. It is, however, not proof, that the system has sufficient margins and stays stable for all possible parameter combinations inside the uncertainty range. For the TECS, notable deviations only occur for 50% of parameter uncertainty (cf. fig. 4.6e and fig. 4.6f), however still with minor magnitude. For 10% and 20% of uncertainty, almost no effects are visible (cf. fig. 4.6a until fig. 4.6f).

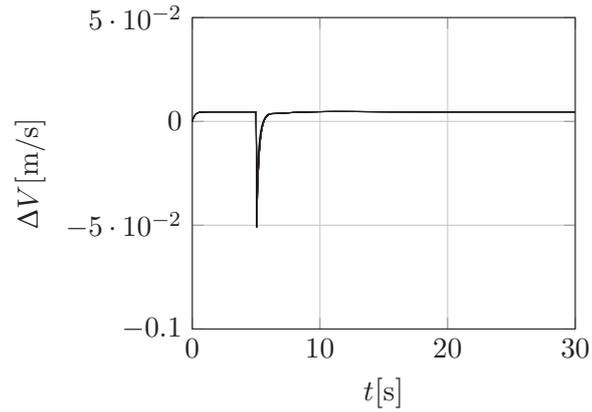
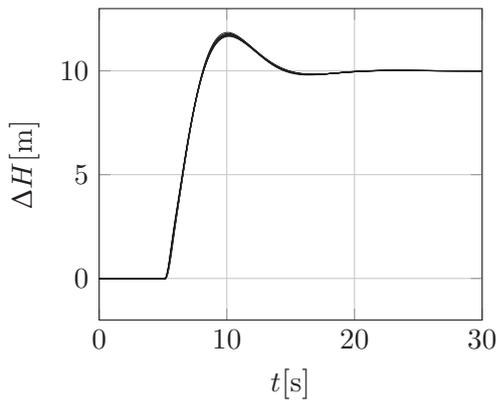
4.4 Computational Effort

During the simulations, it was noticed, that the simulation of the INDI system took significantly longer computation time than the TECS. This was to be expected, as the control vector \vec{u} has to be computed iteratively for each time increment when using the INDI control allocation process. This might be a challenge for implementation on a flight control computer like the *Pixhawk 6*, given its much lower processor performance, compared to a personal computer. For the simulations shown in chapter 4, a maximum iteration count of 100 is set for the optimization function. To potentially reduce the computational effort required, this count was decreased down to a single iteration. Only a slight deviation in the output signal in the magnitude of 10^{-4} occurred, but the overall simulation time did not significantly change. Increasing the maximum number of iterations up to three showed no deviations from the simulations with 100 iterations. Further increasing the iteration count also did not lead to changes in the results any more. Thus, the convergence for the simulation has not taken more than 3 iterations. Even though the number of required function iterations is low, this could potentially exceed the available processing power of commonly used autopilot boards. Reducing the maximum number of function iterations is not expected to significantly reduce the computational effort. A



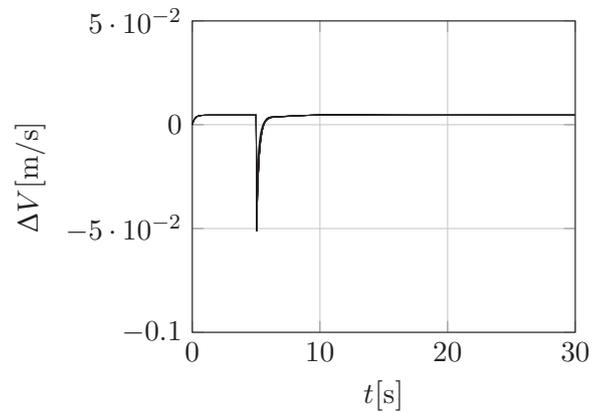
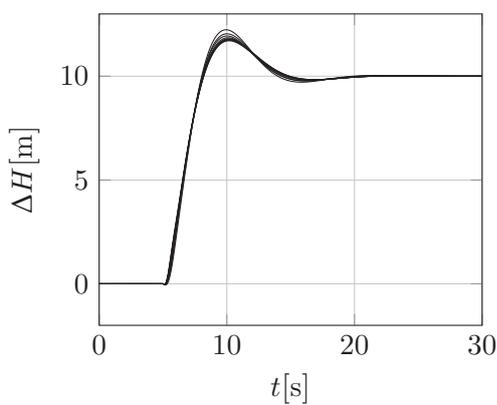
(a) INDI altitude step response with 10% relative model inaccuracy

(b) INDI velocity deviation with 10% relative model inaccuracy



(c) INDI altitude step response with 25% relative model inaccuracy

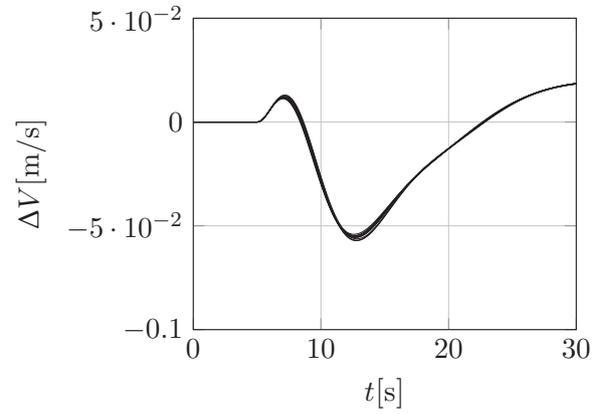
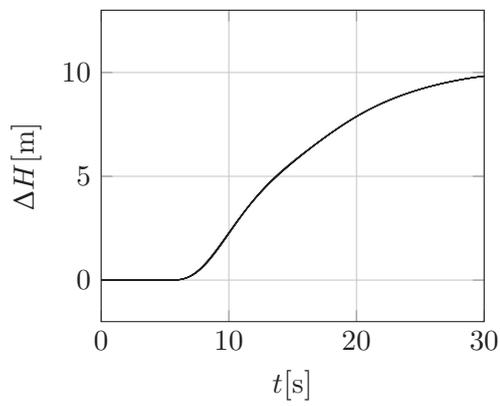
(d) INDI velocity deviation with 25% relative model inaccuracy



(e) INDI altitude step response with 50% relative model inaccuracy

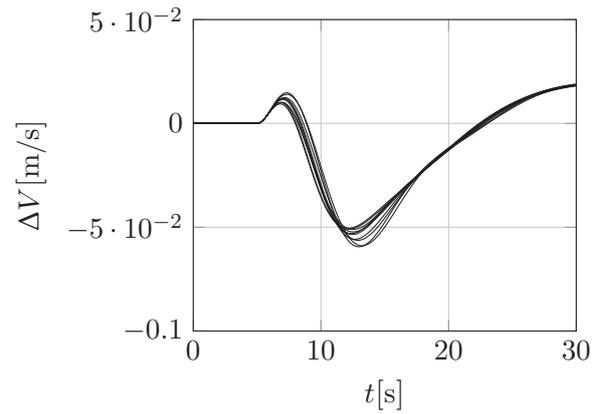
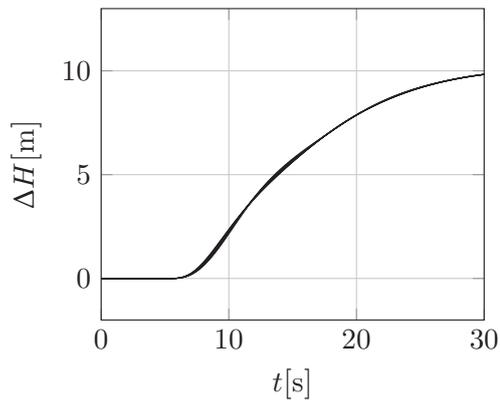
(f) INDI velocity deviation with 50% relative model inaccuracy

Figure 4.5: Influence of parameter variation on altitude step response of INDI controller



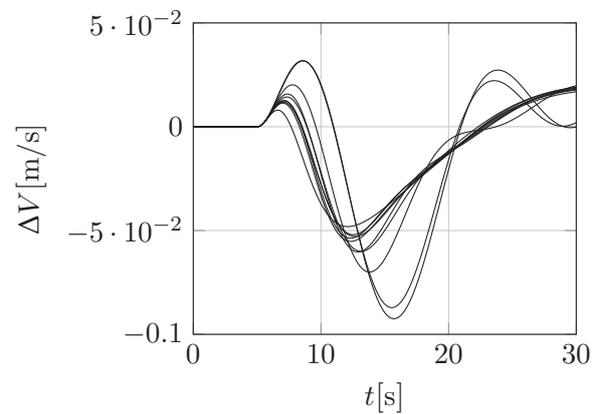
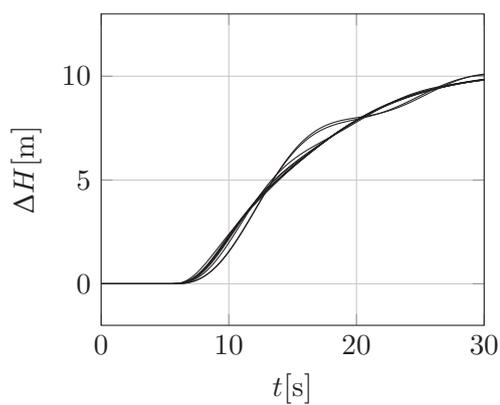
(a) TECS altitude step response with 10% relative model inaccuracy

(b) TECS velocity deviation with 10% relative model inaccuracy



(c) TECS altitude step response with 25% relative model inaccuracy

(d) TECS velocity deviation with 25% relative model inaccuracy



(e) TECS altitude step response with 50% relative model inaccuracy

(f) TECS velocity deviation with 50% relative model inaccuracy

Figure 4.6: Influence of parameter variation on altitude step response of TECS

distributed computation architecture is a possible solution if the processing power of the autopilot computer is not sufficient. In this architecture, the active set method could be outsourced to a companion computer, which will be integrated into the *Proteus* drone for inference and post-training of the machine-learning algorithm. An optimization of the code, which could reduce the iterations needed, and thus the computation time, is the use of an analytical algorithm to solve the weighted least squares (WLS) problem.

Analytical Solution of WLS problem

Take the cost function of the weighted least squares problem eq. (2.7). The first term of the cost function $\|\mathbf{W}_u(\vec{u} - \vec{u}_d)\|^2$ penalizes the actuator effort. This term is used in the cost function in [9] to allow the control of over-actuated systems like multicopters. As a fixed-wing aircraft is not over-actuated, this term can be removed from the equation, making it analytically solvable.

$$C_{red}(\vec{u}) = \|\mathbf{W}_\nu(\mathbf{G}\vec{u} - \vec{v})\|^2 \quad (4.5)$$

The analytical solution to eq. (4.5) is given by eq. (4.6)

$$\vec{u} = (\mathbf{B}^T \mathbf{W}_\nu \mathbf{B})^{-1} \mathbf{B} \mathbf{W}_\nu \vec{v} \quad (4.6)$$

However, the solution for \vec{u} might be out of the defined bounds. To deal with the possibility of actuator saturation, the analytical solution is implemented to the code in a way, that when the bounds are exceeded, the Active-Set-Method from [4] is still used.

Algorithm 1 Combined Active Set and Analytical Solution for WLS Problem

```

1: procedure WEIGHTEDLEASTSQUARES( $\mathbf{B}, \mathbf{W}_\nu, \vec{u}_{min}, \vec{u}_{max}, \mathbf{W}_u, \vec{u}_d, \gamma, \vec{v}$ )
2:   Try solving analytically:
3:    $\vec{u}_{analytical} \leftarrow \text{AnalyticalSolution}(\mathbf{B}, \mathbf{W}_\nu, \vec{v})$ 
4:   if IsWithinBounds( $\vec{u}_{analytical}, \vec{u}_{min}, \vec{u}_{max}$ ) then
5:     return  $\vec{u}_{analytical}$ 
6:   end if
7:   Fallback to Active Set Method:
8:    $\vec{u}_{activeSet} \leftarrow \text{ActiveSetMethod}(\mathbf{W}_u, \vec{u}_d, \gamma, \mathbf{W}_\nu, \mathbf{B})$ 
9:   return  $\vec{u}_{activeSet}$ 
10: end procedure
11: procedure ANALYTICALSOLUTION( $\mathbf{B}, \mathbf{W}_\nu, \vec{v}$ )
12:    $u \leftarrow (\mathbf{B}^T \mathbf{W}_\nu \mathbf{B})^{-1} \mathbf{B} \mathbf{W}_\nu \vec{v}$  ▷ Analytical solution formula
13:   return  $\vec{u}$ 
14: end procedure
15: procedure ACTIVESETMETHOD( $\mathbf{W}_u, \vec{u}_d, \gamma, \mathbf{W}_\nu, B$ )
16:   Active Set Method from [4]
17:   return Solution from Active Set Method
18: end procedure

```

The performance of the modified algorithm is tested by comparing the required computational time for a 30-second simulation of a step in altitude of 10m. Both algorithms

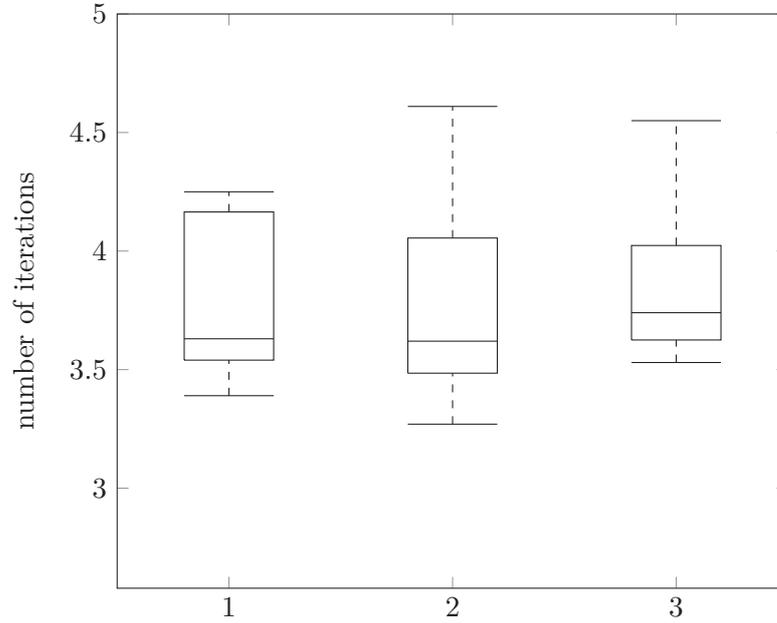


Figure 4.7: Box plot of computation times for (1) analytical algorithm, (2) numerical algorithm, and (3) analytical algorithm with outsourced inverse calculation

showed similar performance, even though the analytical solution always stayed inside bounds so that the numerical solution never had to be used. A possible explanation is that the analytical solution requires the inverse, which has to be calculated for every function call in algorithm 1. It is possible to precalculate the $\mathbf{B}^T \mathbf{W}_\nu \mathbf{B})^{-1}$ term beforehand to potentially reduce the computation time. However, this approach has not shown any positive change to the overall computation time, as can be seen from fig. 4.7, plot 3. It has also disabled the possibility of online adjustments of the matrix weights in W_ν . In general, the addition of an analytical algorithm to find a solution faster has not shown any benefit to the computation time, most likely due to the already low number of required iterations for the numerical solver. Whether the current performance of the INDI control system is compatible with the flight computer hardware of the *Proteus* drone has to be tested.

4.5 Combination of a TECS with INDI

A combination of INDI with an outer-loop TECS controller seems like a promising control architecture for fixed-wing UAVs, as it combines two state-of-the-art control strategies. This concept was analysed by [6] for a ducted fan UAV. They considered the performance of the combined controller superior to PID and PI TECS controllers regarding disturbance rejection as well as for changing airspeed or velocity without affecting the other. However, the performance of this architecture is not compared against a cascade control using INDI, as described in this and further papers [8, 10]. As shown in section 4.1, the decoupling of airspeed and vertical flight path is already satisfyingly fulfilled with this architecture, while also showing strong disturbance rejection properties (cf. section 4.3, [10]). Thus, the use of an outer loop TECS is considered obsolete by the author, when

INDI is applied. Furthermore, systematically tuning a TECS requires an eigenstructure assignment [3], but is often carried out through heuristics. For cascade control loops, used in this paper, in combination with INDI, controllers can be designed with far higher flexibility to achieve the desired behaviour of the aircraft.

5 Conclusion

This paper explored the performance of a control system using incremental nonlinear dynamic inversion (INDI) for the longitudinal control of the morphing wing drone *Proteus* and compared it to a total energy control system (TECS). The main factors investigated were the decoupling of vertical flight path and horizontal speed, the tracking performance of a landing profile, and the robustness against changes in flight dynamics, caused by the morphing wing. Both systems showed similar and satisfying decoupling of speed and vertical flight path. Thus, the combination of an inner loop rate controller using INDI and an outer loop TECS structure, as proposed in [6] is considered obsolete when introducing a complex tuning task. The INDI system slightly outperformed the TECS in its path-tracking ability and was slightly more robust against perturbations of the eigendynamics. Both systems are considered suitable for the longitudinal control of the drone, with the INDI having slightly superior performance and robustness. When also accounting for the high disturbance rejection capabilities for INDI control systems outlined in the literature, the system is considered a capable candidate for the control of the drone. A difficulty when investigating the stability of the INDI controller was the nonlinearity of the control system, as the computation of margins is not directly possible. The Monte-Carlo parameter sampling approach was useful for gaining a general impression of the robustness. To prove the controller's robustness to real-world perturbations, the implementation of the controller in a simulation using the occurring changes in dynamics is necessary. The INDI required additional computation performance compared to the TECS, which might not be available on a flight computer like the *Pixhawk 6*. Optimization of the control allocation algorithm by adding an analytical solution to the weighted least squares problem did not improve the computation time required. Thus, the INDI control allocation algorithm might have to be outsourced to a companion computer. Further changes required for a real-world flight test are the implementation of filters to remove sensor noise and predict states, which are not measured directly. It is also required to tune the outer loop control structure of the INDI control system in consideration of the actuator dynamics of the aircraft.

Bibliography

- [1] Rudolf Brockhaus, Wolfgang Alles, and Robert Luckner. *Flugregelung*. 3., neu bearb. Aufl. Berlin and Heidelberg: Springer, 2011. ISBN: 978-3-642-01442-0.
- [2] David A. Burdette and Joaquim R.R.A. Martins. “Design of a transonic wing with an adaptive morphing trailing edge via aerostructural optimization”. In: *Aerospace Science and Technology* 81 (2018), pp. 192–203. ISSN: 12709638. DOI: 10.1016/j.ast.2018.08.004.
- [3] L. F. Faleiro and A. A. Lambregts. “Analysis and tuning of a ‘Total Energy Control System’ control law using eigenstructure assignment”. In: *Aerospace Science and Technology* 3.3 (1999), pp. 127–140. ISSN: 12709638. DOI: 10.1016/S1270-9638(99)80037-6.
- [4] O. Härkegård. “Efficient active set algorithms for solving constrained least squares problems in aircraft control allocation”. In: *Proceedings of the 41st IEEE Conference on Decision and Control, 2002*. IEEE, 2002, pp. 1295–1300. ISBN: 0-7803-7516-5. DOI: 10.1109/CDC.2002.1184694.
- [5] A. A. Lambregts. “Vertical flight path and speed control autopilot design using total energy principles”. In: *Guidance and Control Conference*. Reston, Virginia: American Institute of Aeronautics and Astronautics, 1983. DOI: 10.2514/6.1983-2239.
- [6] Chengji Liu, Hailong Pei, and Zihuan Cheng. “A High-Performance Sensor-Based Control for Complex Maneuvers of Ducted Fan Unmanned Aerial Vehicles”. In: *2022 34th Chinese Control and Decision Conference (CCDC)*. IEEE, 2022, pp. 3271–3277. ISBN: 978-1-6654-7896-0. DOI: 10.1109/CCDC55256.2022.10034415.
- [7] *PX4 Architectural Overview / PX4 User Guide (main)*. 2023. URL: <https://docs.px4.io/main/en/>.
- [8] S. Sieberling, Q. P. Chu, and J. A. Mulder. “Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction”. In: *Journal of Guidance, Control, and Dynamics* 33.6 (2010), pp. 1732–1742. ISSN: 0731-5090. DOI: 10.2514/1.49978.
- [9] Ewoud Smeur, Daniel Höppener, and Christophe de Wagter. *Prioritized Control Allocation for Quadrotors Subject to Saturation*. 2017.
- [10] Ewoud J. J. Smeur, Qiping Chu, and Guido C. H. E. de Croon. “Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles”. In: *Journal of Guidance, Control, and Dynamics* 39.3 (2016), pp. 450–461. ISSN: 0731-5090. DOI: 10.2514/1.G001490.