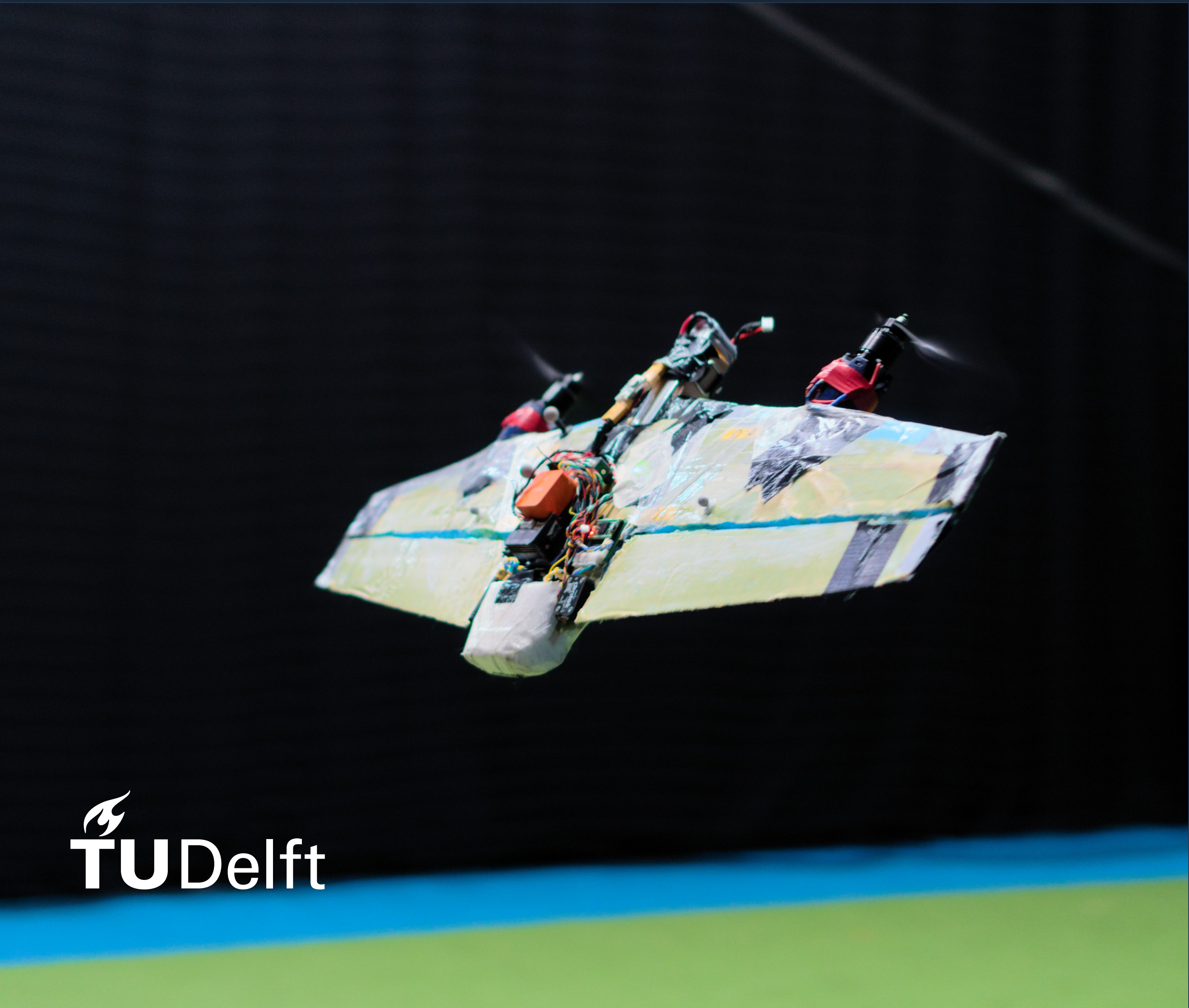
Full Actuator Nonlinear Dynamic Inversion for Enhanced Hybrid UAV Control

Justin Dubois



Full Actuator Nonlinear Dynamic Inversion for Enhanced Hybrid UAV Control

Thesis report

by

Justin Dubois

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on February 11, 2026

Thesis committee:

Chair:	Dr. Ir. E. van Kampen
Supervisors:	Dr. Ir. E.J.J. Smeur
	E. Ntouros
External examiner:	Dr. S. Theodoulis
Place:	Faculty of Aerospace Engineering, Delft
Project Duration:	March 2025 - February 2026
Student number:	5217091

Faculty of Aerospace Engineering · Delft University of Technology

Cover: Photograph of the Cyclone tail-sitter MAV, courtesy of Bartosz Jemioł.

TU Delft
Delft
University of
Technology

Preface

Completing this thesis marks the end of a chapter that began back in September 2020 with my Bachelor's in Aerospace Engineering. The past years have been full of challenges, exciting projects, and incredible people who made this journey both meaningful and enjoyable.

I would like to sincerely thank my supervisors, Ewoud Smeur and Evangelos Ntouros, for their guidance, feedback, and encouragement throughout this project. During our numerous discussions, it became clear to me that they are not only knowledgeable experts in their field but also genuinely passionate about working together with me on this project.

I also want to thank everyone at the MAVLab for their technical help, insightful discussions, and the great atmosphere in the lab. From the MAVLab team, I would especially like to thank Erik van der Horst for his assistance and invaluable hardware expertise. I learned a great deal from him during our many late evenings at the lab, where he patiently helped me debug countless unexplainable issues. His knowledge and problem-solving skills saved me more than once, and probably kept me sane in the process.

Lastly, I'd like to thank my friends for their positivity and encouragement, and my family for their endless support, patience, and understanding throughout these past years.

Justin Dubois
Delft, February 2026

Contents

List of Acronyms

ANDI Actuator Nonlinear Dynamic Inversion 1–3, 10, 14, 15, 18, 37–41, 43, 45, 49–51, 53, 55–57

HITL Hardware-In-The-Loop 56

INDI Incremental Nonlinear Dynamic Inversion 1–3, 10, 12–15, 18, 37, 49–51, 55–57

ISTAR Intelligence, Surveillance, Target Acquisition, and Reconnaissance 7

LQR Linear-Quadratic Regulator 11

MAV Micro Air Vehicle i, 1–3, 7, 18, 37, 41–43, 45, 49–51, 53, 56

MIMO Multiple Input Multiple Output 11, 12

NDI Nonlinear Dynamic Inversion 11–13, 18

OBM On-Board Model 1, 11–13, 38, 42

PID Proportional-Integral-Derivative 11

SISO Single Input Single Output 11

SUAV Small Unmanned Aerial Vehicle 7, 12, 14–16, 18

TBV Tilt-Body Vehicle 8

UAV Unmanned Aerial Vehicle 1, 2, 7–11, 14–18, 37, 55, 56

VTOL Vertical Takeoff and Landing 8

1

Introduction

1.1. Research Background

Micro Air Vehicles (MAVs) are experiencing rapid growth and adoption across an increasingly wide range of industrial applications. These vehicles address the need for agile, cost-effective, and versatile aerial platforms in roles where traditional larger aircraft are impractical or uneconomical. To satisfy different mission requirements, these vehicles are often tailored to best fulfill a specific purpose, resulting in a diverse landscape of vehicle types, including fixed-wing, rotary-wing, and hybrid configurations [1].

This diversity in vehicle design brings with it a unique challenge in the development and deployment of flight control systems for MAVs. The low cost and small size of these platforms means they are often developed under tight budget constraints and require rapid prototyping cycles. As a result, there is a strong demand for generic, adaptable control solutions that can be quickly implemented across different platforms.

However, the unique and often unconventional airframe designs of MAVs may lead to highly nonlinear and coupled dynamics, making robust and high-performance control a significant challenge [1]. Traditional linear controllers, which rely on local linearization around an operating point, struggle to provide the required performance and robustness across the wide range of operating conditions encountered by these vehicles. There is therefore a need for nonlinear control approaches that are both easy to implement and sufficiently generic to be adapted to various MAV configurations.

Sensor-based Incremental Nonlinear Dynamic Inversion (INDI) has emerged as a promising control technique to address these challenges. Sensor-based INDI leverages sensor feedback to locally linearize the nonlinear dynamics of the vehicle, effectively transforming the system into an equivalent linear one over a broad operating envelope [2]. This enables the use of classical linear control techniques for systems with significant nonlinearities. Furthermore, sensor-based INDI offers the ability to decouple airframe-dependent characteristics from the controller design [2], thus facilitating the reuse of control architectures across different MAVs.

Sensor-based INDI has demonstrated robust performance on a variety of MAVs [3–8]. This success is particularly due to its reliance on measured state derivatives, rather than a detailed On-Board Model (OBM). This makes the approach inherently resilient to modeling errors and unmodeled disturbances [2].

Despite its strengths, the standard sensor-based INDI approach has limitations. The closed-loop system's response to disturbances and unmodeled dynamics is shaped by the actuator dynamics and any filtering present on the output [9, 10]. When actuator bandwidth is limited relative to the dynamics of the Unmanned Aerial Vehicle (UAV), the performance of sensor-based INDI degrades, leading to tracking errors and less predictable error dynamics.

To overcome these limitations, Steffensen, Steinert, and Smeur [11] developed a control approach that explicitly models both the state-dependent dynamics and actuator dynamics, known as Actuator Nonlinear Dynamic Inversion (ANDI). Actuator Nonlinear Dynamic Inversion (ANDI) compensates for these dynamics directly, enabling more precise compensation of nonlinearities and improved tracking performance, especially when state-dependent dynamics and actuator dynamics are significant [11].

While ANDI has been validated in simulation and shown to offer theoretical advantages such as well-defined error dynamics, practical experimental validation remains limited. Previous work by De Ponti, Smeur, and Remes [12] has demonstrated ANDI's effectiveness in handling actuators with differing bandwidth. However, the ability to compensate state-dependent dynamics has yet to be tested on a real MAV.

1.2. Research Formulation

To systematically address the research gap identified above, the following research objective has been defined:

Research Objective

> The research objective is to implement, and experimentally validate a full ANDI stabilization controller with state-dependent dynamics compensation on a tail-sitter MAV, assessing its feasibility as a generic and adaptable control approach for MAVs, and to quantify the performance benefits of state-dependent dynamics compensation in real-world flight conditions compared to standard INDI.

To achieve this objective, the research is structured around research questions 1 to 3 that guide the investigation from theoretical foundations to practical implementation and experimental validation.

Research Question 1

> How does full ANDI with state-dependent dynamics compensation address the limitations of sensor-based INDI for control of a hybrid MAV?

Research Question 2

> How does explicit compensation of state-dependent dynamics affect reference tracking accuracy and disturbance rejection performance when deployed on a tail-sitter MAV in real-world flight conditions?

Research Question 3

> How can ANDI with state-dependent dynamics compensation be systematically implemented on generic MAVs, considering practical constraints such as limited model knowledge, and available sensor measurements?

These questions progressively work towards achieving the research objective: Research Question 1 establishes the theoretical foundation by reviewing existing control methods and positioning ANDI within the broader landscape of nonlinear control. Research Question 2 evaluates the actual performance benefits through experimental validation, providing empirical evidence of the approach's effectiveness in real-world conditions. Finally, research question 3 aims to provide a guide to facilitate the adoption of ANDI for various MAVs.

1.3. Structure of the Report

This report is organized into four main parts:

Part I **Preliminary Analysis** provides the theoretical foundation necessary to understand the research context and motivation. The literature study in Chapter 2 positions sensor-based INDI and full actuator ANDI within the broader landscape of nonlinear control methods for UAVs. This part directly addresses Research Question 1 by examining how ANDI with state-dependent dynamics compensation addresses the limitations of standard sensor-based INDI and how it can be applied to hybrid MAVs.

Part II **Scientific Article** presents the core contribution of this thesis in the form of a scientific article. This article documents the design, implementation, and experimental validation of the full ANDI stabilization controller on the *Cyclone* tail-sitter MAV platform. It encompasses the vehicle dynamics modeling, detailed controller design, theoretical equivalence analysis between ANDI and INDI, and comprehensive experimental results. This part directly addresses Research Question 2 by providing empirical evidence of the performance benefits of state-dependent dynamics compensation in real-world flight conditions.

Part III **Additional Results** provides additional work not part of the scientific article. This includes practical implementation guidelines for deploying ANDI on generic hybrid MAVs platforms, addressing Research Question 3. In addition, the initial development of a guidance controller based on the ANDI framework is presented as a work in progress.

Part IV **Closure** provides a conclusion to the report, revisiting the research objective and questions, and summarizes the key findings. It also offers recommendations for future research directions based on the insights gained throughout this study.

Part I

Preliminary Analysis

2 Literature Review

2.1. A Brief Overview of Hybrid UAVs

Recent advancements in Unmanned Aerial Vehicle (UAV) technology have expanded their civil, commercial, and military applications. Small Unmanned Aerial Vehicles (SUAVs), also referred to as Micro Air Vehicles (MAVs), now enable diverse operations across many sectors. Individuals and small businesses commonly employ SUAVs for photography, videography, and creative media. Aerial imagery from these platforms has enabled new applications in content production. Filmmakers and creators can now achieve perspectives that were previously unavailable due to the high cost and logistical complexity [13].

In civil domains, SUAVs have been applied in emergency response operations. Following natural disasters, they can be deployed to map flood zones, locate survivors, and assess infrastructure damage, thereby reducing exposure for human operators [14]. Their versatility also benefits environmental monitoring, where optical, multi-spectral, and thermal sensors track habitat changes, vegetation health, and pollution sources [15]. Agriculture has embraced these systems for rapid field surveys, enabling precise irrigation, fertilization, and early crop stress detection [16].

Infrastructure management and urban development increasingly rely on these compact aerial systems. Power line inspections, bridge surveys, and construction monitoring exploit SUAVs that hover near structures [17]. Urban planners apply UAV-based photogrammetry to generate high-resolution models of expanding cities [18]. Healthcare providers use small drones to deliver medical supplies between hospitals or to remote areas difficult to access via traditional means [19].

In defense and security roles, SUAVs have become indispensable for Intelligence, Surveillance, Target Acquisition, and Reconnaissance (ISTAR) operations. The same attributes driving civil adoption, namely low cost, rapid deployment, and operational versatility, make these platforms valuable for military reconnaissance, border patrol, and coastal monitoring where coverage and response time are critical [20].

The range of these applications is reflected in the diversity of SUAV configurations. The reduced cost and shorter development cycles of SUAVs enable designs optimized for specific mission profiles. Depending on operational needs, SUAVs may prioritize endurance, maneuverability, payload capacity, or minimizae cost and complexity [13, 21].

2.1.1. Fixed-Wing and Rotary-Wing UAVs

Many of these missions require long endurance and extended range, which fixed-wing UAVs provide through aerodynamic lift. Fixed-wing designs excel at covering large areas efficiently, making them ideal for mapping, surveillance, and long-haul transport [1]. However, they demand additional infrastructure such as runways or catapults, and open spaces for takeoff and landing, severely limiting operational flexibility as they can not be used in confined environments like urban rooftops, ships, disaster zones, or cluttered construction sites.

On the other hand, rotary-wing UAVs, such as multi-rotors and helicopters, offer Vertical Takeoff and Landing (VTOL) capability, precise hovering, and superior maneuverability without infrastructure needs. These qualities make them indispensable for infrastructure inspections, emergency mapping, precision agriculture, and urban operations where centimeter-level positioning is required [21]. Yet rotary-wing platforms suffer from limited endurance, short range, and high energy consumption due to continuous powering of the rotors to counteract gravity, making them less suitable for demanding missions requiring long flight times or extensive coverage.

2.1.2. Hybrid UAVs Configurations

The solution lies in hybrid UAVs, which combine rotary-wing VTOL versatility with fixed-wing cruise efficiency. These platforms enable operations from any location while achieving the endurance needed for extended missions [1]. Engineers have pursued this through diverse configurations, each balancing mechanical complexity, weight, control requirements, and mission suitability.

Dual-system hybrids use separate propulsion sets for each flight regime: dedicated lift rotors for hover (often multi-rotor style) and forward-facing motors for cruise. Examples include quad-planes like the *Evo* by Deltaquad [22], shown in Figure 2.1a. This approach provides robust, independent operation in each flight mode with straightforward controls. Advantages include reliability (failure in one system does not necessarily compromise the other) and simpler transitions; the main disadvantage is the increased weight from carrying the additional propulsion system [1].

Tilt-rotor and tilt-wing designs reorient either propulsion units or entire wing structures between hover and cruise modes. Tilt-rotors physically pivot motors from vertical to horizontal positions, sharing thrust systems across flight regimes. An example of the configuration is the *Trinity Pro* tilt-rotor by Quantum Systems [23], shown in Figure 2.1b. By reusing motors across both flight modes, tilt-rotors minimize redundancy and weight. Advantages include efficient resource utilization and simple transition; disadvantages involve complex actuators, structural reinforcements for rotating joints, and vulnerability to tilting mechanism failures [1]. Tilt-wing designs rotate entire wings (with fixed motors attached) to redirect aerodynamic lift and thrust, simplifying motor control and optimizing wing aerodynamics in both modes. Advantages include better aerodynamic performance and simple transitions; disadvantages include high gust sensitivity during hover due to large vertical wing area, and additional structural demands and strong actuators needed [1].

Tail-sitters, also known as Tilt-Body Vehicle (TBV), pitch the entire airframe from vertical takeoff/landing to horizontal cruise using fixed propulsion, without dedicated actuators. Platforms like the *WingtraOne* by Wingtra [24] in Figure 2.1c and the *Cyclone* by Bronz et al. [25] in Figure 2.1d exemplify this. They minimize mechanical complexity approaching that of a fixed-wing. Advantages are simplicity, low weight and cost, and full VTOL capabilities; disadvantages include complex control requirements across the full flight envelope, payload integration challenges due to the tilting body, and reduced gust tolerance during hover mode. [1, 26].

2.2. Tail-Sitter UAVs

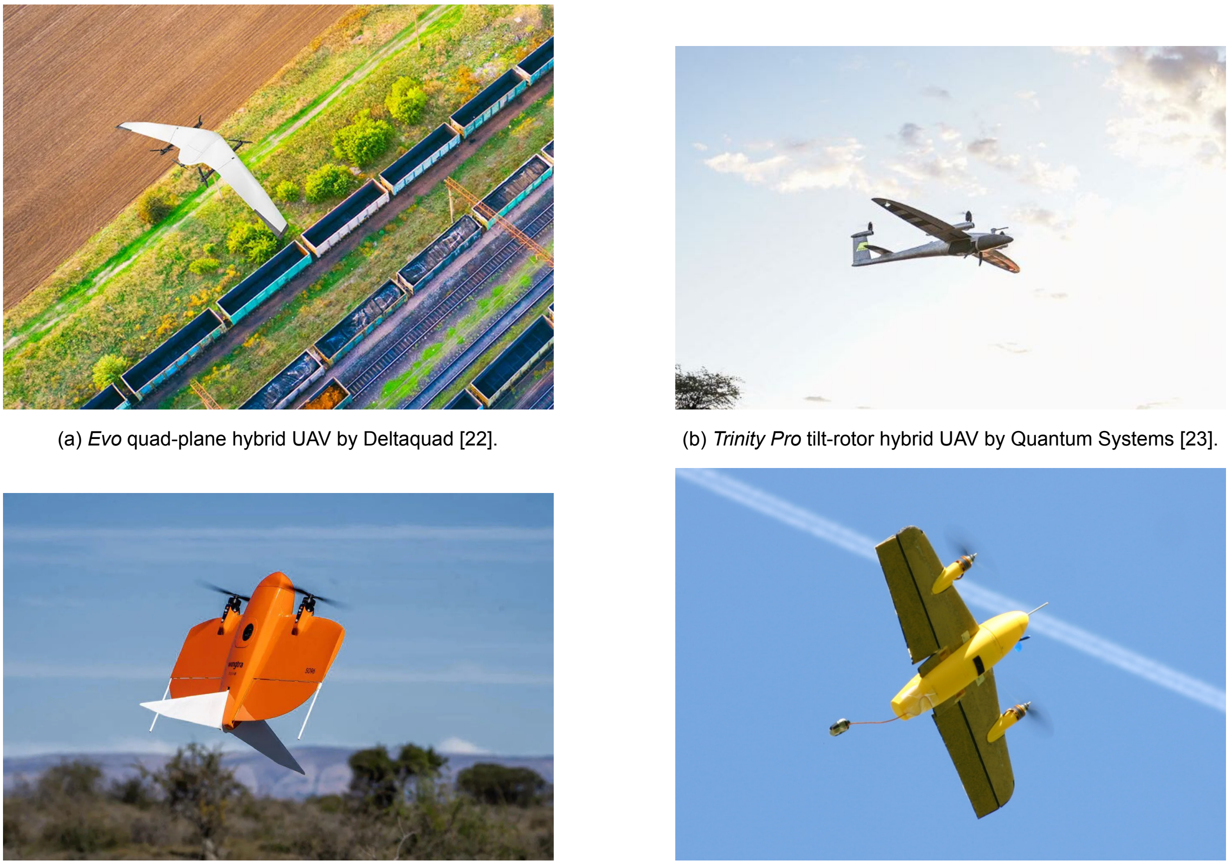
This work focuses on tail-sitter hybrid UAVs as the test platform for implementing and testing the control framework. Tail-sitters are suited for applications requiring high endurance and long-range coverage, as their simple mechanical design, low weight, and efficient fixed-wing cruise performance provide advantages over more complex hybrid configurations. Despite these benefits, they present formidable control challenges due to their need for stable operation across the entire flight envelope from hover through transition to forward flight.

Tail-sitters typically employ two counter-rotating tractor propellers mounted on the fuselage to provide thrust and lift during hover. Control in pitch and yaw is achieved through elevons, which are control surfaces located on the trailing edge of the wing. These elevons deflect symmetrically to generate pitch moments and differentially to produce yaw moments, remaining effective in hover due to their placement in the propeller slipstream [25].

The body-fixed coordinate frame is defined with respect to the vehicle's hover orientation rather than forward flight attitude. This choice is fundamental: the body frame has its x-axis pointing forward (out of the fuselage belly), the y-axis pointing to the right, and the z-axis pointing downward (toward the tail). Unlike conventional fixed-wing aircraft where the body frame is aligned with forward-flight cruise conditions, the hover-based definition for tail-sitters ensures consistent axis interpretation throughout all flight regimes, simplifying control formulation and physical interpretation when the aircraft is pitched nearly horizontal during cruise.

Control Challenges of Tail-Sitters

Controlling tail-sitters across their full flight envelope requires nonlinear control approaches. Tail-sitters must operate continuously across a wide flight envelope, from vertical hover through all intermediate attitudes to horizontal forward flight without mode switching.



(a) *Evo* quad-plane hybrid UAV by Deltaquad [22].

(b) *Trinity Pro* tilt-rotor hybrid UAV by Quantum Systems [23].

(c) *WingtraOne* tail-sitter hybrid UAV by Wingtra [24].

(d) *Cyclone* tail-sitter hybrid UAV during transition [25].

Figure 2.1: Different Hybrid UAVs Configurations: (a) Quad-plane, (b) Tilt-rotor, (c) Tail-sitter, (d) Tail-sitter.

During the transition from vertical hover to forward flight, angle-of-attack changes rapidly, encountering post-stall aerodynamic regimes where the relationship between control inputs and resulting forces becomes highly nonlinear.

A critical technical constraint is the singularity problem in attitude representation. Euler angles, the classical representation used in aviation, become singular at certain attitudes (gimbal lock), making them unsuitable for a controller that must operate globally across all pitch angles. Quaternion-based attitude representations are required to avoid these singularities, adding mathematical complexity to the control formulation.

Actuator limitations further complicate control design. The elevons have finite deflection ranges and limited effectiveness, particularly at low airspeed or low thrust settings. Saturation of these surfaces is frequently encountered during aggressive maneuvers. Additionally, the motors and elevons exhibit heterogeneous actuator dynamics with different bandwidths, although these are typically not strongly coupled for tail-sitters.

Of particular importance for this work is the state-dependent dynamics exhibited by tail-sitters. During aggressive maneuvers in the transitional flight regime, aerodynamic forces and moments are substantial and change rapidly. These effects degrade the tracking performance if unaccounted for.

Tail-Sitters as a Research Platform

Tail-sitters provide a suitable platform for validating advanced control techniques in practical hybrid vehicle applications. The control challenges discussed above, including state-dependent dynamics, heterogeneous actuator dynamics, and the requirement for global control across a wide range of attitudes, make tail-sitters a challenging test case. The tail-sitter configuration is thus chosen as a representative platform for evaluating the effectiveness of the proposed Actuator Nonlinear Dynamic Inversion (ANDI) control framework with state-dependent dynamics compensation for hybrid UAVs.

2.3. Nonlinear Control

Despite their operational advantages, hybrid UAVs present control challenges. Unlike traditional fixed-wing or rotary-wing aircraft, which operate within a defined flight envelope, hybrid aircraft must transition between two different aerodynamic regimes. During hover, they rely on rotor-based lift and control, while during forward flight, they exploit aerodynamic surfaces and forward thrust. Each regime exhibits distinct aerodynamic characteristics and typically requires different control strategies. The transition between regimes introduces additional control complexity.

Traditional approaches to hybrid UAV control employ composite controllers, where different linear controllers are blended for each flight mode. While linear control methods have demonstrated success in industry, they have significant limitations [27]. These methods are typically complex, vehicle-specific, and costly to develop and maintain. More critically, linear controllers designed for specific operating points often cannot maintain satisfactory performance across the full flight envelope, particularly during aggressive maneuvers or in the presence of disturbances. The nonlinear dynamics of hybrid vehicles across multiple flight regimes motivates the adoption of global nonlinear control strategies. Approaches such as Incremental Nonlinear Dynamic Inversion (INDI) can operate effectively across all flight conditions using a single, unified controller architecture [28].

Nonlinear control theory provides the mathematical framework for managing systems whose dynamics cannot adequately be approximated by linear models [29]. Aircraft are inherently nonlinear systems: as they traverse different speeds, altitudes, and attitudes, aerodynamic forces, moments, and actuator responses vary dramatically and nonlinearly with the system states and inputs. Traditional linear control approaches are inadequate for these systems.

Linear controllers work by assuming the plant behaves linearly, an assumption rarely valid for real aircraft. Weakly nonlinear systems can be controlled within a small operational domain around an equilibrium point, where local linearization approximates the dynamics. However, this approach is ineffective when the system operates outside design conditions [29]. Although linear control methods have been effective for small perturbations and provided the foundation for early autopilot systems, they cannot guarantee stability or performance across the entire flight envelope. Consequently, they are unsuitable for many real-world systems, such as hybrid vehicles that operate across a wide flight envelope [30].

2.3.1. Nonlinear Control Methodologies

To address these challenges, several nonlinear control methodologies have been developed:

Gain Scheduling combines multiple linear controllers in order to make them suited for controlling weakly nonlinear systems. This is achieved by interpolating between a set of linear controllers, each one designed and tuned for a local linear time-invariance approximation of the systems [29, 31]. Commercial aircraft flight control systems employ this strategy to handle varying aerodynamic conditions across flight regimes (e.g., takeoff, cruise, and landing). Controllers are tuned for discrete airspeed and altitude ranges, with interpolation ensuring smooth transitions as parameters like dynamic pressure slowly change [29, 32]. However, this approach is primarily applicable for systems with weakly nonlinear dynamics dominated by parameter variations rather than complex cross-coupling. In practice, it introduces high development costs from tuning and validating dozens of controller variants and risks of transient discontinuities during regime transitions [29, 31–33].

Adaptive Control addresses systems with uncertain or slowly varying parameters (e.g. fuel consumption altering aircraft mass). By updating controller gains in real time, adaptive methods maintain performance across varying conditions. These approaches are particularly useful for aviation systems with linearly parameterizable dynamics, such as altitude control in commercial aircraft. However, they require measurable states and struggle with rapidly changing disturbances, limiting their use in highly dynamic maneuvers. Adaptive methods can be especially useful when they compliment other (model based) nonlinear control methods [29].

Robust control techniques can account for nonlinearities by treating them as uncertainties in a simplified linear model. These methods design linear controllers based on worst-case bounds on a nominal linear model, ensuring stability even with unmodeled dynamics [29]. Robust control can provide formal guarantees of robustness to bounded disturbances and model uncertainties, making them suitable for safety-critical applications. However, they often lead to conservative designs that sacrifice performance for robustness, and their effectiveness diminishes for strongly nonlinear systems where linear approximations are inadequate [34].

Backstepping does not require linearization. Instead, it uses a recursive design methodology to stabilize a nonlinear system by systematically designing controllers for each subsystem, introducing pseudo-control inputs at each step. Backstepping can retain beneficial nonlinear terms, which can improve the controller's performance. It does require the system to be in strict-feedback form, which limits it use for systems with complex dynamics [35].

Feedback Linearization techniques, such as Nonlinear Dynamic Inversion (NDI), transform the nonlinear dynamics into a single equivalent linear system which is valid over a large range of operating conditions, without the need for gain scheduling [29, 36]. By algebraically canceling nonlinear terms, this approach enables the use of classical linear control techniques such as Proportional-Integral-Derivative (PID) or Linear-Quadratic Regulator (LQR) for highly nonlinear systems. In addition, feedback linearization decouples the airframe dependent characteristics from the controller design [37], making the approach generic and modular, resulting in a controller which is adaptable to a wide range of different systems [38, 39]. For example, hybrid aircraft like the *F-35B* leverage feedback linearization to manage transitions between hover and forward flight regimes [40].

2.4. Feedback Linearization

Feedback linearization offers several advantages for controlling hybrid UAVs. First, by providing global linearization across a large operating range, it enables a single unified controller architecture that maintains consistent performance from vertical takeoff through transition to cruise flight [29, 36]. Second, the approach naturally decouples vehicle-specific characteristics from the controller design [37], yielding a generic architecture that adapts readily to different configurations without extensive redesign, significantly reducing development cost in the small platform market [38, 39]. Third, certain variants of feedback linearization, particularly sensor-based approaches, require minimal prior knowledge of vehicle dynamics, making them practical for small unmanned systems where detailed aerodynamic modeling may be prohibitively expensive or unavailable.

2.4.1. Model-Based Nonlinear Dynamic Inversion

Model-based Nonlinear Dynamic Inversion (NDI), also referred to as feedback linearization or just NDI, in the most general form, is a widely used model-based control technique that aims to transform a nonlinear system into an equivalent linear one through algebraic cancellation of nonlinearities using state feedback. This approach allows the subsequent application of common linear control design methods to nonlinear systems, provided that the On-Board Model (OBM) is sufficiently accurate and certain structural conditions are met [29].

Theoretical Foundations

The NDI control law for a Multiple Input Multiple Output (MIMO) system starts with a general state-space form assuming the equations of motion are affine in the inputs, as given in Equation (2.1) [29].

$$\begin{cases} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + G(\mathbf{x})\mathbf{u} \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}) \end{cases} \tag{2.1}$$

In Equation (2.1), the vector $\mathbf{x} \in \mathbb{R}^n$ denotes the system's state, the control input vector is represented by $\mathbf{u} \in \mathbb{R}^k$. The function $\mathbf{f}(\mathbf{x}) \in C^r(\mathbb{R}^n; \mathbb{R}^n)$ describes the system's state-dependent dynamics, independent of the control input. The matrix $G(\mathbf{x}) \in C^1(\mathbb{R}^n; \mathbb{R}^{n\times m})$ characterizes the control effectiveness, mapping the control inputs to their influence on the state derivatives. The output vector, $\mathbf{y} \in \mathbb{R}^m$, represents the measured or controlled quantities of interest, the function $\mathbf{h}(\mathbf{x}) \in C^r(\mathbb{R}^n; \mathbb{R}^m)$ defines how the states are mapped to these outputs. For this derivation the assumption is made that the system is fully actuated, meaning that the number of inputs and outputs is equal, $k = m$. The affine structure of the dynamics, where the control input appears linearly as $G(\mathbf{x})\mathbf{u}$, is a required for the application of NDI, as it enables the direct inversion of the system dynamics for feedback linearization [29].

The concept of relative degree needs to be introduced. For a Single Input Single Output (SISO) system, the relative degree $r \leq n$ is the minimal number of times the output needs to be differentiated to time in order for it to be a direct function of the input. The relative degree should be constant and defined over the region of interest around $\mathbf{x_0}$. For MIMO systems, each output y_i has a partial relative degree $r_i \in \mathbb{N}$. This partial relative degree is the minimum number of times the output y_i needs to differentiated to time for it to be a direct function of $\mathbf{u}$. The total relative $r \leq n$ degree is the sum of the partial relative degrees, assuming the outputs are independent of each other. In the case where $r = n$, full linearization can be achieved, and all nonlinearities can be canceled. In contrast, when $r < n$, only partial linearization is possible since the internal dynamics remain nonlinear and must exhibit intrinsic stability to ensure overall system stability.

Without loss of generality, assume the system has relative degree $r \in \mathbb{N}$ with respect to $\mathbf{u}$ for all elements of $\mathbf{y}$ around $\mathbf{x}_0$. Equation Equation (2.2) expresses the derivative of order r of the output equation, this is achieved using Lie derivatives in order to express the rate of change of $\mathbf{h}(\mathbf{x})$ along a vector field.

$$\mathbf{y}^{(r)} = L_f^r \mathbf{h}(\mathbf{x}) + L_g L_f^{r-1} \mathbf{h}(\mathbf{x})\mathbf{u} \tag{2.2}$$

The linearizing NDI control law is obtained in Equations (2.3) to (2.5) by solving Equation (2.2) for the input command $\mathbf{u}_c = \mathbf{u}$, and introducing the pseudo-control vector $\boldsymbol{\nu} \in \mathbb{R}^m$ by choosing $\mathbf{y}^{(r)} = \boldsymbol{\nu}$. The dependence on the OBM is clear from the presence of $A(\mathbf{x})$ and $\mathbf{b}(\mathbf{x})$ in the control law.

$$A(\mathbf{x}) = L_g L_f^{r-1}\mathbf{h}(\mathbf{x}) \tag{2.3}$$

$$\mathbf{b}(\mathbf{x}) = L_f^r \mathbf{h}(\mathbf{x}) \tag{2.4}$$

$$\mathbf{u}_c = A(\mathbf{x})^{-1}(\boldsymbol{\nu} - \mathbf{b}(\mathbf{x})) \tag{2.5}$$

The pseudo-control vector $\boldsymbol{\nu}$ can be designed using linear control techniques. In the case of a reference tracking problem, a linear control law could be as in Equation (2.6). With $\mathbf{y}_{ref}$ the desired reference to track and k_i the controller gains.

$$\boldsymbol{\nu} = \mathbf{y}_{ref}^{(r)} + \sum_{i=0}^{r-1} k_i(\mathbf{y}_{ref}^{(i)} - \mathbf{y}^{(i)}) \tag{2.6}$$

Properties of Model-Based Nonlinear Dynamic Inversion

The primary theoretical advantage of NDI is that it can invert the system's nonlinear dynamics, resulting in a linear cascade of integrators in the closed-loop response [29, 36]. This perfect linearization, achieved without gain scheduling, substantially simplifies controller design and enables precise tracking and stabilization across the entire operating envelope [29].

In practice, NDI has a significant limitation: its performance depends on the accuracy of the OBM. The control law explicitly relies on precise model inversion to cancel nonlinearities; any discrepancy between model and reality, whether from unmodeled dynamics, parameter uncertainties, or external disturbances, directly degrades inversion quality and can cause performance degradation or even instability [29, 41]. Creating a model of sufficient accuracy for reliable operation can become prohibitively expensive for small unmanned systems, where detailed system identification and validation consume resources that might be better allocated to other aspects of development.

Beyond model accuracy, NDI success depends entirely on high-quality state measurements. Sensor noise, measurement delays, or misalignment between sensors can significantly compromise the control law's ability to achieve the intended linearization.

These practical constraints reveal the limitations of NDI for SUAVs. While NDI offers theoretically perfect inversion, its requirement for a detailed and accurate OBM makes it poorly suited to many small platform applications where development resources are limited and vehicle configurations may vary widely.

2.4.2. Sensor-Based Incremental Nonlinear Dynamic Inversion

Sensor-Based Incremental Nonlinear Dynamic Inversion (INDI), also referred to as simplified feedback linearization or just INDI, is an incremental controller that completely forgoes the need for the OBM of the state-dependent dynamics and instead relies on direct sensor measurements to correct for the unmodeled dynamics implicitly. By relying on sensor measurements, only minimal prior knowledge of the system dynamics is required [2, 42].

Theoretical Overview of Sensor-Based Incremental Nonlinear Dynamic Inversion

INDI can be applied to a generic system as shown in Equation 2.7. In contrast to NDI, the system no longer needs to be affine in the input.

$$\begin{cases} \dot{\mathbf{x}} & = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} & = \mathbf{h}(\mathbf{x}) \end{cases} \tag{2.7}$$

In Equation (2.7), the vector $\mathbf{x} \in \mathbb{R}^n$ denotes the system's state, the control input vector is represented by $\mathbf{u} \in \mathbb{R}^k$. The function $\mathbf{f}(\mathbf{x}, \mathbf{u}) \in C^{r+1}(\mathbb{R}^n \times \mathbb{R}^k; \mathbb{R}^n)$ describes the system's dynamics. The output vector, $\mathbf{y} \in \mathbb{R}^m$, represents the measured or controlled quantities of interest, the function $\mathbf{h}(\mathbf{x}) \in C^{r+1}(\mathbb{R}^n; \mathbb{R}^m)$ defines how the states are mapped to these outputs.

Without loss of generality, assume the system has a relative degree of $r \in \mathbb{N}$ with respect to $\mathbf{u}$ for all elements of $\mathbf{y}$ around $\mathbf{x}_0$. Equation (2.8) expresses the derivative of order r of the output equation.

$$\mathbf{y}^{(r)} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{2.8}$$

To linearize this system, a first-order Taylor expansion is used around the previous state and input, $[\mathbf{x}_0, \mathbf{u}_0]$, where the subscript 0 denotes values from one sampling interval Δt earlier. Equation (2.9) shows this expansion.

$$\mathbf{y}^{(r)} \approx \underbrace{\mathbf{f}(\mathbf{x}_0, \mathbf{u}_0)}_{\mathbf{y}_0^{(r)}} + \underbrace{\frac{\partial}{\partial \mathbf{u}}\mathbf{f}(\mathbf{x}_0, \mathbf{u}_0)}_{F_u} \cdot (\mathbf{u} - \mathbf{u}_0) + \underbrace{\frac{\partial}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x}_0, \mathbf{u}_0)}_{F_x} \cdot (\mathbf{x} - \mathbf{x}_0) \tag{2.9}$$

The term $\mathbf{y}_0^{(r)} = \mathbf{f}(\mathbf{x}_0, \mathbf{u}_0)$ is the measured output derivative from the previous control step, which can be obtained directly from sensor measurements. The matrix F_u is the control effectiveness matrix relating changes in the input to changes in the output derivative. The matrix F_x relates state changes to output changes.

The assumption of timescale separation is made: when sampling at very high frequencies, the state change $(\mathbf{x} - \mathbf{x}_0)$ is negligible compared to the input change. As a result, the state-dependent term $F_x(\mathbf{x} - \mathbf{x}_0) \approx \mathbf{0}$ is dropped. This assumption is equivalent with assuming the actuators have infinitely high bandwidth [11]. Neglecting the state-dependent term simplifies the equation to Equation (2.10).

$$\mathbf{y}^{(r)} \approx \mathbf{y}_0^{(r)} + F_u \cdot (\mathbf{u} - \mathbf{u}_0) \tag{2.10}$$

This linear approximation forms the basis for the INDI control law, where the only required model knowledge is the control effectiveness matrix F_u.

The linearizing INDI control is obtained in Equation (2.11) by solving Equation (2.8) for the input command $\mathbf{u}_c = \mathbf{u}$, and introducing the pseudo-control vector $\boldsymbol{\nu} \in \mathbb{R}^m$ by choosing $\mathbf{y}^{(r)} = \boldsymbol{\nu}$. The system can be solved with any right inverse matrix $F_u^\dagger$.

$$\mathbf{u}_c = F_u^\dagger(\boldsymbol{\nu} - \mathbf{y}_0^{(r)}) + \mathbf{u}_0 \tag{2.11}$$

INDI is an incremental controller, this means that a control increment is computed at each iteration instead of an absolute control input. To get the total control input, the control increment needs to be added to the previous input command. This is reflected in Equation (2.11) where $\mathbf{u}_0$ is added to the control increment.

The pseudo-control vector $\boldsymbol{\nu}$ can be designed using linear control techniques. In case of reference tracking problem, this could be as shown in Equation (2.6).

Properties of Sensor-Based Incremental Nonlinear Dynamic Inversion

The sensor-based approach is a significant advantage over model-based NDI. Rather than requiring an accurate OBM, the control law uses measurements of the system's response. This makes sensor-based INDI robust to model uncertainties and external disturbances, since unmodeled effects and disturbances are reflected in the sensor feedback and compensated for in the next control increment [42].

Another advantage is the resulting reduction in implementation cost and complexity, as there is no need to develop and maintain a high-fidelity OBM of the full system dynamics. The control effectiveness matrix, which relates incremental control inputs to measured state derivatives, remains the only required model component, which can often be identified or adapted online. This simplicity has contributed to the growing popularity of INDI for SUAVs [42].

Despite these benefits, sensor-based INDI has several important limitations. One of these is that it does not account for heterogeneous actuator dynamics, instead assuming that the actuator dynamics are identical and can instantaneously follow control commands. The closed-loop system's response to disturbances, and the unmodeled dynamics, is shaped by the actuator dynamics and any filtering present on the output [9, 10]. When actuator bandwidth is limited relative to the dynamics of the UAV, the INDI controller performance is affected, resulting in tracking errors and ill-defined error dynamics. In case coupled heterogeneous actuator dynamics are present, the closed-loop system can become unstable [10, 43].

Furthermore, because of the reduced model dependence, the standard INDI formulation omits state-dependent terms in the system dynamics [42]. This can reduce accuracy during aggressive maneuvers where such terms are significant. In many cases there is at least some knowledge available about the system state-dependent dynamics which INDI cannot benefit from.

The performance of INDI depends on accurate sensor measurements. Time delays and sensor data mismatches can reduce performance or cause instability [44]. This issue becomes particularly critical when measurement filtering is necessary to mitigate sensor noise, as such filtering introduces additional delays. These delays can negatively impact the closed-loop response, potentially causing instability or degraded tracking performance [10, 42].

Application of Sensor-Based Incremental Nonlinear Dynamic Inversion on Small Unmanned Aerial Vehicle

Sensor-based INDI has been successfully applied to many SUAVs and has demonstrated robust performance across a diverse range of platforms and configurations. The approach has been validated on fixed-wing, rotary-wing, and hybrid UAVs, demonstrating its versatility and applicability beyond a single platform type [3–8, 28].

The adoption of INDI in research and industry reflects its practical applicability: it provides stable control under model uncertainties, compensates for disturbances through sensor feedback, and requires limited prior knowledge of system dynamics. This robustness and simplicity have made it the control method of choice for many small platform developers where detailed aerodynamic modeling is impractical or unavailable. However, despite these successes, INDI still faces challenges related to heterogeneous actuator dynamics and the inability to leverage state-dependent system knowledge, motivating further advancements in the field.

2.4.3. Actuator Nonlinear Dynamic Inversion

To address the limitations of sensor-based INDI, arising from neglecting state-dependent dynamics and heterogeneous actuator dynamics, Steffensen, Steinert, and Smeur [11] propose ANDI, a hybrid control law which can compensate for state-dependent dynamics in the feedforward path while also explicitly accounts for heterogeneous first order actuator dynamics [11].

Overview of Actuator Nonlinear Dynamic Inversion

Actuator Nonlinear Dynamic Inversion (ANDI) controller is an incremental controller that linearizes a generic system as given in Equation (2.7) with first order linear actuator dynamics Equation (2.12) which shows the first order actuator dynamics with the commanded input $\mathbf{u}_c \in \mathbb{R}^k$. The diagonal matrix $\boldsymbol{\varepsilon} \in \mathbb{R}^{(k \times k)}$ defines the actuator bandwidth for each actuator.

$$\dot{\mathbf{u}} = \boldsymbol{\varepsilon}(\mathbf{u_c} - \mathbf{u}) \tag{2.12}$$

Without loss of generality, assume the system has a relative degree of $r \in \mathbb{N}$ with respect to $\mathbf{u}$ for all elements of $\mathbf{y}$ around $\mathbf{x}_0$. Equation (2.13) expresses the output equation's time derivative of order r.

$$\mathbf{y}^{(r)} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{2.13}$$

The first order actuator dynamics are introduced explicitly in this relation by differentiating once more with respect to time as shown in Equations (2.14) and (2.15).

$$\mathbf{y}^{(r+1)} = \underbrace{\frac{\partial}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x}, \mathbf{u})}_{F_x}\dot{\mathbf{x}} + \underbrace{\frac{\partial}{\partial \mathbf{u}}\mathbf{f}(\mathbf{x}, \mathbf{u})}_{F_u}\dot{\mathbf{u}} \tag{2.14}$$

$$\mathbf{y}^{(r+1)} = F_x\dot{\mathbf{x}} + F_u\boldsymbol{\varepsilon}(\mathbf{u_c} - \mathbf{u}) \tag{2.15}$$

The linearizing ANDI control law is obtained in Equation (2.16) by solving Equation (2.15) for the commanded input $\mathbf{u}_c$, and introducing the pseudo-control vector $\boldsymbol{\nu} \in \mathbb{R}^m$ by choosing $\mathbf{y}^{(r+1)} = \boldsymbol{\nu}$. The system can be solved with any right inverse matrix $(F_u\boldsymbol{\varepsilon})^\dagger$.

$$\mathbf{u}_c = (F_u\boldsymbol{\varepsilon})^\dagger(\boldsymbol{\nu} - F_x\dot{\mathbf{x}}) + \mathbf{u} \tag{2.16}$$

The pseudo-control vector $\boldsymbol{\nu}$ can be designed using linear control techniques. In case of reference tracking problem, this could be as shown in Equation (2.6).

Properties of Actuator Nonlinear Dynamic Inversion

ANDI provides inversion for systems with first-order actuator dynamics. By modeling and inverting the actuator dynamics in the control law, ANDI allows specification of error dynamics consistent with physical actuator limitations. This produces predictable closed-loop behavior, even when actuators have different bandwidths or are slow compared to the system dynamics [11]. Standard INDI, by contrast, assumes infinitely fast actuators and does not account for actuator limitations and heterogeneity, which can result in degraded tracking performance and less predictable error dynamics [10, 43].

Another advantage is explicit compensation of state-dependent terms. While sensor-based INDI omits these terms and relies on sensor feedback for correction, ANDI includes them directly, resulting in improved tracking accuracy and robustness [11]. This allows ANDI to achieve perfect tracking of a reference under ideal conditions, independent of the chosen error controller gains.

Like sensor-based INDI, ANDI uses sensor feedback to reject disturbances and unmodeled effects. Unmodeled effects are reflected in the measurements, and are compensated in the next control increment. The advantages of disturbance rejection and robustness to model uncertainties thus apply to ANDI as well [11].

However, ANDI introduces additional complexity and implementation cost. ANDI requires more model knowledge than sensor-based INDI, increasing the cost and effort for system identification and model validation. Additionally, ANDI requires accurate measurements of state derivatives, which can be challenging to obtain [11].

In summary, ANDI provides improvements over sensor-based INDI in scenarios where actuator limitations and state-dependent dynamics are significant. It offers enhanced tracking, predictable error dynamics, and adaptability to actuators with different bandwidths, at the cost of increased modeling requirements and complexity. The approach combines elements of the simplicity and robustness of sensor-based INDI extended with compensation of state-dependent dynamics and actuator heterogeneity.

Application of Actuator Nonlinear Dynamic Inversion on Small Unmanned Aerial Vehicle

ANDI is a relatively recent development and has not yet seen widespread application of SUAVs. However, De Ponti, Smeur, and Remes [12] has applied the ANDI controller on a variable skew quad-plane drone with coupled heterogeneous actuator dynamics. The results demonstrate the ability of ANDI to deal with these heterogeneous actuator dynamics compared to sensor-based INDI. The ability to compensate for state-dependent dynamics has been omitted in this work and remains to be validated on real-world SUAVs.

2.5. System Modeling

The different feedback linearization methods discussed in this chapter all rely on some form of system modeling to varying degrees. The accuracy and structure of these models directly impact the performance and stability of the resulting controllers. Therefore, a deep understanding of system modeling is crucial for implementing the nonlinear control strategies on UAVs.

2.5.1. Model Structure

This thesis limits itself to parametric model estimation methods, where the model structure is known a priori and only the parameters are identified from data. The model structure defines how the forces and moments acting on the vehicle are related to its states and inputs through parameterized equations.

The model structure is chosen based on a trade-off between complexity and fidelity. A more complex model structure can capture more physical effects but may require more parameters to be estimated, increasing the effort and amount of data needed for identification. Conversely, a simpler model structure may be easier to identify but may neglect important dynamics, leading to poor model accuracy.

In addition to capturing the relevant dynamics, the model structure must be differentiable and also be singularity free over the entire flight envelope of the SUAV. For tail-sitter SUAVs, this is particularly challenging due to the wide range angles of attack and sideslip encountered, especially during transitions between hover and forward flight.

Classical Parametrization

The Buckingham-π-theorem based approach [45] represents the classical method for modeling aerodynamic forces and moments, and is widely applied in aviation. The method groups physical parameters into dimensionless combinations, yielding expressions for forces and moments in terms of dynamic pressure, vehicle geometry, and aerodynamic angles. This is described in Equations (2.17) and (2.18) [26].

$$\mathbf{F}_b = \frac{1}{2}\rho S \mathbf{v}_\infty^2 \mathbf{C}_b(\alpha, \beta, M, Re) \tag{2.17}$$

$$\mathbf{M}_b = \frac{1}{2}\rho S \mathbf{v}_\infty^2 B \mathbf{K}_b(\alpha, \beta, M, Re) \tag{2.18}$$

$$\tag{2.19}$$

The aerodynamic forces $\mathbf{F}_b \in \mathbb{R}^3$ and moments $\mathbf{M}_b \in \mathbb{R}^3$ acting on a finite wing are governed by the dynamic pressure $\frac{1}{2}\rho\mathbf{v}_\infty^2$, where ρ is the air density and $\mathbf{v}_\infty \in \mathbb{R}^3$ is the freestream velocity. The wing's planform area S scales these forces, directly influencing lift and drag generation. The force coefficient vector $\mathbf{C}_b \in \mathbb{R}^3$ and moment coefficient vector $\mathbf{K}_b \in \mathbb{R}^3$ are parameterized functions that depend on the angle of attack α, sideslip angle β, Mach number M, and Reynolds number Re. When considering SUAVs flying at low velocities, compressibility and viscosity can usually be neglected, it can thus be assumed that $\mathbf{C}_b$ and $\mathbf{K}_b$ are solely functions of α and β.

For moments, the diagonal matrix B in Equation (2.20), introduces geometric scaling. The wingspan b scales rolling and yawing moments, while the mean aerodynamic chord $\overline{c}$ scales pitching moments.

$$B = \begin{bmatrix} b & 0 & 0 \\ 0 & \overline{c} & 0 \\ 0 & 0 & b \end{bmatrix} \tag{2.20}$$

These parameters collectively capture the wing's aerodynamic response to varying flight conditions and geometry, with $\mathbf{C}_b$ and $\mathbf{K}_b$ are derived from experimental or computational data. This is usually represented by finite truncated sums of Fourier or MacLaurin series, with the series coefficients serving as the tunable parameters of the model.

This classical approach to modeling poses fundamental problems when applied to tail-sitter UAVs. The trigonometric relation of the free-stream and the angle of attack and sideslip, given in Equations (2.21) and (2.22), make this apparent.

$$\alpha = \tan^{-1}\left(\frac{v_{\infty,b3}}{v_{\infty,b1}}\right) \tag{2.21}$$

$$\beta = \sin^{-1}\left(\frac{v_{\infty,b2}}{||\mathbf{v}_\infty||}\right) \tag{2.22}$$

During hover or low-speed flight, the free-stream velocity approaches zero, rendering these trigonometric relations singular. Small measurement errors or computational noise are significantly amplified when dividing by near-zero values, producing erratic estimates of aerodynamic angles. This singularity fundamentally undermines the model's reliability in hover and low-speed regimes.

An additional limitation emerges when the tail-sitter descends vertically. It encounters angles of attack approaching $\alpha = \pi$ radians $180°$, far beyond the normal operating envelope of fixed-wing aircraft and beyond the scope of this parametrization's validity.

These limitations render the classical model unsuitable for designing controllers that must operate globally across the tail-sitter's full flight envelope.

Φ-Theory Parametrization

The limitations of classical parametrization motivated the development of alternative model structures specifically suited to vehicles that operate across a wide flight envelope. Φ-theory parametrization is one such approach, designed explicitly for hybrid UAVs with significant transition capabilities [26]. Rather than relying on aerodynamic angles (α, β) that become singular at low speeds, Φ-theory uses a velocity-based parametrization that remains well-defined and numerically stable across the entire flight envelope.

Instead of expressing forces and moments as functions of aerodynamic angles, Φ-theory directly uses the body frame velocity and angular velocity components. This angle-free parametrization eliminates singularities and naturally accommodates the full range of flight conditions experienced by tail-sitters. The model structure yielding the aerodynamic forces and moments is expressed in Equation (2.23) [26].

$$\boldsymbol{\tau}_b = -\frac{1}{2}\rho S \eta C \Phi(\boldsymbol{\eta}_b) C \boldsymbol{\eta}_b \tag{2.23}$$

Where $\boldsymbol{\tau}_b \in \mathbb{R}^6$ is the aerodynamic wrench with respect to the center of mass. The vehicle's linear and angular velocity components with respect to the freestream are combined in the state vector $\boldsymbol{\eta}_b \in \mathbb{R}^6$. The scalar $\eta \in \mathbb{R}_+$ represents the aerodynamic ϕ-norm. The air density ρ wing planform area S, and wing screw reference matrix $C \in \mathbb{R}^{6\times6}$ are also included in the model structure to allow the parameters to be dimensionless. The core parameters of the model are the aerodynamic Φ-coefficients, these define the function $\Phi : \mathbb{R}^6 \rightarrow \mathbb{R}^{6\times6}$.

The state vector $\boldsymbol{\eta}_b$ in Equation (2.24) combines the vehicle's freestream linear and angular velocity components. The aerodynamic ϕ-norm η in Equation (2.25) serves as a generalized speed metric that remains well-defined even when linear velocity approaches zero during hover. The tunable parameter ϕ weights the relative importance of linear and angular velocity components.

$$\boldsymbol{\eta}_b = [\mathbf{v}_{\infty,b}^\top, \boldsymbol{\omega}_{\infty,b}^\top]^\top \tag{2.24}$$

$$\eta = \sqrt{\mathbf{v}_\infty^2 + \phi\boldsymbol{\omega}_\infty^2}\,, \quad \phi > 0 \tag{2.25}$$

The matrices C is defined in Equation (2.26) and incorporates geometric scaling for moments through the matrix B defined in Equation (2.20).

$$C = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & B \end{bmatrix} \tag{2.26}$$

$$\tag{2.27}$$

The Φ-matrix in Equation (2.28) is decomposed into four 3×3 sub-matrices that separately capture the effects of linear and angular velocity on forces and moments.

$$\Phi = \begin{bmatrix} \Phi^{(fv)} & \Phi^{(f\omega)} \\ \Phi^{(mv)} & \Phi^{(m\omega)} \end{bmatrix} \tag{2.28}$$

Each component in Equation (2.28) represents a specific physical mechanism and is a constant function linear in the parameters.

- $\Phi^{(fv)}$: forces due to linear velocity (lift, drag)
- $\Phi^{(f\omega)}$: forces due to angular velocity (cross-coupling effects, damping forces)
- $\Phi^{(mv)}$: moments due to linear velocity (pitching moments from aerodynamic forces)
- $\Phi^{(m\omega)}$: moments due to angular velocity

The key advantage of Φ-theory is that it provides a singularity-free mathematical model structure from hover through transition to forward flight. This enables the development of genuinely global controllers that do not need to rely on mode switching. Moreover, because the model is structured using velocity components rather than aerodynamic angles, it accommodates vehicles with unconventional geometries like tail-sitters, where traditional aerodynamic conventions are not applicable. This makes Φ-theory particularly well-suited as the basis for implementing hybrid control approaches like INDI or ANDI on hybrid UAVs, which require a valid model across the full operating envelope.

2.6. Conclusion

This chapter has presented an overview of SUAVs and the control frameworks essential for their operation. The operational diversity and mechanical complexity of hybrid platforms necessitate generic and adaptable nonlinear control approaches. Feedback linearization techniques emerge as particularly compelling for this task, offering the ability to achieve global linearization across the entire flight envelope using a single, generic controller architecture. Three principal variants have been examined: Model-based NDI offers theoretical perfect inversion but demands expensive, high-fidelity models unsuitable for small platforms. Sensor-based INDI represents a practical alternative by forgoing detailed models in favor of real-time sensor feedback, providing practical robustness to model inaccuracies and cost-effectiveness. Additionally, the chapter discussed model structures for hybrid UAVs, identifying the singularity-free Φ-theory parametrization as particularly suitable for tail-sitter configurations across the full flight envelope.

Sensor-based INDI is widely used for hybrid SUAV control, demonstrating practical applicability and performance across diverse platforms. However, this approach has limitations that affect SUAVs with slow actuators or significant state-dependent dynamics. Standard INDI assumes infinitely fast actuators and cannot explicitly handle heterogeneous actuator dynamics, leading to degraded tracking performance and unpredictable error dynamics when actuator bandwidth is limited. Furthermore, INDI neglects all state-dependent terms in the system dynamics, instead treating them as disturbances to be implicitly compensated through sensor feedback alone. These limitations motivate the need to investigate enhanced control approaches that can address both heterogeneous actuator dynamics and state-dependent effects while retaining the flexibility and simplicity of sensor-based methods.

ANDI addresses both identified limitations of sensor-based INDI. First, regarding actuator dynamics: ANDI explicitly models and inverts first-order actuator dynamics within the control law, enabling designers to specify desired closed-loop error dynamics consistent with actual actuator bandwidths. This results in defined behavior even with slow or heterogeneous actuators, in contrast to the implicit treatment of actuator limitations in standard INDI. Second, regarding state-dependent dynamics: ANDI includes the state-dependent term $F_x\dot{\mathbf{x}}$ directly in the control law for explicit compensation, rather than relying solely on sensor feedback to correct for these effects. This inclusion improves tracking accuracy and robustness in nonlinear regimes where state-dependent contributions are significant. Like sensor-based INDI, ANDI provides disturbance rejection and robustness to unmodeled effects, offering a hybrid approach for hybrid MAVs with significant actuator limitations and state-dependent dynamics.

Part II

Scientific Article

Full Actuator Nonlinear Dynamic Inversion for Enhanced Hybrid UAV Control

Justin P. G. Dubois*

Abstract—**Expanding the operational capabilities of Micro Air Vehicles (MAVs) hinges on control systems that manage highly nonlinear dynamics across broad flight envelopes. Incremental Nonlinear Dynamic Inversion (INDI) is popular for its simplicity and modest modeling needs, but its assumption of infinitely fast actuators and neglect of state-dependent effects limit performance when actuators have slow or heterogeneous dynamics or when aerodynamic effects are significant. Actuator Nonlinear Dynamic Inversion (ANDI) overcomes these limitations by explicitly incorporating state-dependent dynamics and finite actuator bandwidth into the control law, enabling improved tracking performance across diverse actuator configurations.**

This work implements the full ANDI stabilization controller on the *Cyclone*, a hybrid MAV tail-sitter, using cascaded complementary filtering for state estimation. Simulation and flight experiments validate the approach and assess whether this compensation yields practical performance gains, establishing ANDI as a viable, generic control solution for MAVs. Code is available at `https://github.com/tudelft/paparazzi/tree/feat_stabilization_andi_controller`

Index Terms—**actuator nonlinear dynamic inversion, feedback linearization, hybrid UAV, incremental nonlinear dynamic inversion, nonlinear control, tail-sitter**

I. Introduction

MAVs are experiencing rapid growth and adoption in an increasingly wide range of industrial applications where they address the need for agile, cost-effective, and versatile aerial platforms. To satisfy the mission requirements, these vehicles are often tailored to best fulfill a specific mission, resulting in a diverse landscape of vehicle types, including fixed-wing, rotary-wing, and hybrid configurations [1]. Hybrid Unmanned Aerial Vehicles (UAVs), such as tail-sitters and tiltrotors, combine the vertical takeoff and landing capabilities of rotary-wing aircraft with the efficient forward flight characteristics of fixed-wing designs, enabling a broader range of operational scenarios

This diversity in airframe design, made possible by low costs, brings a unique challenge in the development and deployment of flight control systems for MAVs. The low cost and small size mean these vehicles are often developed under tight budget constraints and require rapid prototyping cycles. As a result, there is a strong demand for generic, adaptable control solutions that can quickly be implemented across different platforms. However, the unique and often unconventional airframe designs of these vehicles may lead to highly nonlinear and coupled dynamics, making reliable and high-performance control a significant challenge [1]. Traditional linear controllers, which rely on a local linearization, struggle to provide the required performance across the wide range of operating conditions encountered by these vehicles. Therefore, there is a need for nonlinear control approaches that are both easy to implement and sufficiently generic to be adapted to various MAV configurations.

INDI has emerged as a promising control technique to address these challenges. INDI leverages sensor feedback to locally linearize the nonlinear dynamics of the vehicle, effectively transforming the system into an equivalent linear one over a broad operating envelope without needing a full model of the system dynamics [2]. This enables the use of classical linear control techniques for systems with significant nonlinearities. Furthermore, INDI decouples airframe-dependent characteristics from the controller design, thus facilitating the reuse of control architectures across different UAV platforms. INDI has demonstrated excellent performance on a variety of MAVs [3–8], particularly due to its reliance on measured state derivatives (such as angular accelerations) rather than a detailed On-Board Model (OBM), making it resilient to modeling errors and unmodeled disturbances.

Despite its strengths, INDI neglects the influence of actuator dynamics and state-dependent terms in its formulation, which can limit performance in practical scenarios with slow actuators or significant state-dependent dynamics. In INDI, any unmodeled dynamics, such as state-dependent effects (e.g., aerodynamic terms) or deviations in actuator behavior, are treated as disturbances and must be corrected through the feedback path. However, this feedback correction is inherently limited by actuator bandwidth and filtering. This results in slow error rejection, leading to increased tracking errors and less predictable error dynamics [9].

To overcome these limitations, Steffensen et al. [10] developed ANDI, which explicitly models both state-dependent drift dynamics and finite actuator dynamics directly in the feedforward path of the control law. By compensating for these effects in the feedforward path, rather than relying on feedback correction, ANDI enables faster error rejection and improved tracking performance. ANDI has been validated in simulation and shown to offer theoretical advantages such as well-defined error dynamics and exact reference tracking [10]. Previous experimental work by De Ponti et al. [11] has demonstrated ANDI's effectiveness in compensating for actuators with differing bandwidth. However, the ability to compensate state-

*J. P. G. Dubois is with Faculty of Aerospace Engineering, Control & Simulation, Delft University of Technology, 2629HS Delft, The Netherlands(Corresponding author).

This paper is part of the author's M.Sc. thesis supervised by Dr. Ir. E.J.J. Smeur and E. Ntouros, at Delft University of Technology. Full thesis is available at `https://repository.tudelft.nl`.

dependent effects through the feedforward path has yet to be experimentally validated on a real MAV platform.

This project aims to bridge this gap by implementing and experimentally validating full ANDI with explicit state-dependent compensation on a tail-sitter MAV. The *Cyclone* tail-sitter exhibits strong state-dependent effects, including lift and drag caused by the large wing, providing a representative testbed for evaluating feedforward state-dependent compensation. The work comprises: (i) designing and implementing a full ANDI stabilization (attitude) controller with explicit state-dependent compensation; (ii) generating undelayed feedback estimates through a cascaded complementary filter as proposed by Steffensen et al. [9]; and (iii) conducting flight tests to experimentally validate the compensation of state-dependent effects and compare performance against the INDI baseline, extending beyond prior actuator-only studies [11].

II. Experimental Platform and Configuration

The experimental platform for this work is the *Cyclone* tail-sitter UAV, a hybrid Vertical Takeoff and Landing (VTOL) platform developed by Bronz et al. [12]. The *Cyclone* combines the hover capabilities of a rotary-wing aircraft with the efficient forward flight of a fixed-wing design. The vehicle operates in two primary flight modes: in hover, it stands vertically on its tail and relies on two tractor propellers for lift; in forward flight, it pitches to a horizontal attitude where the wing provides the majority of lift and the propellers generate forward thrust. Transitioning between these modes involves significant changes in aerodynamic loading and control authority, making it a representative platform for evaluating advanced nonlinear control strategies. The body frame is defined with respect to the vehicle's hover orientation, with the x-axis pointing forward (out of the belly), the y-axis pointing to the right, and the z-axis pointing downward (to the tail). This is shown in fig. 1.

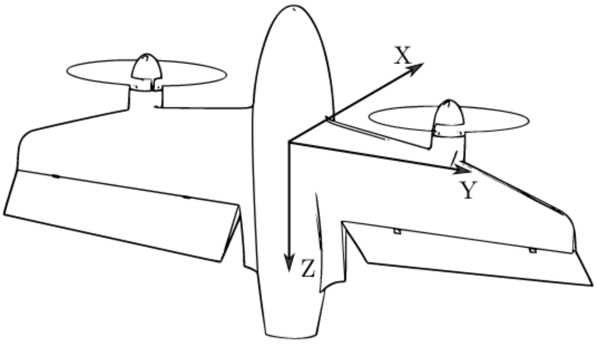


Fig. 1: Diagram of the *Cyclone* tail-sitter UAV with its body axis convention. [12]

The *Cyclone* operates across a continuous spectrum of flight attitudes from vertical hover to horizontal forward flight. Transition from hover to forward flight is achieved by continuously pitching the vehicle down from the vertical hover orientation to horizontal forward flight; the opposite transition from forward flight to hover is achieved by pitching up. The vehicle can operate at any pitch angle, requiring a globally valid control law, which is capable of maintaining stability and tracking performance across all attitudes.

The *Cyclone* is equipped with actuators and sensors to facilitate control and navigation. Actuation is provided by two counter-rotating tractor propellers driven by electric motors and a pair of elevons. The motors provide the main thrust and are used differentially to generate roll moments. Pitch and yaw moments are controlled by the elevons via symmetric and differential deflections, respectively. These surfaces remain effective in hover because they operate in the propeller slipstream. Actuator feedback is available in the form of motor rotational speed and elevon angle feedback from the servos. The vehicle is outfitted with an Inertial Measurement Unit (IMU) that provides measurements of angular rates and specific forces. An onboard flight computer, running the `Paparazzi Autopilot` [13] software, manages sensor data acquisition, state estimation, and control law execution. A positive elevon deflection corresponds to a trailing-edge-down command, generating a pitch-down moment about the body y-axis. The left motor rotates counterclockwise and the right motor clockwise, when viewed from behind the vehicle. The servos can directly be commanded with a desired angle, while the motors only support commands as a percentage of maximum throttle. This limitation is addressed by linearly mapping the throttle command to the motor rotational speed. A better approach is to implement a closed-loop speed controller, which is left for future work.

III. Vehicle Dynamics Model

This section details the vehicle dynamics model structure and the identified coefficients used for the *Cyclone* tail-sitter. The model is based on the Φ-theory aerodynamic framework. For this research, flight tests have been exclusively conducted in an indoor environment where only hover and low-speed forward flight conditions were achievable. Consequently, the model has been simplified to focus on these regimes by retaining only the most significant terms identifiable by the experimental data.

The model structure can be extended in future work to include high-speed flight regimes, should additional data become available.

A. Specific Thrust Modelling

The total specific thrust τ generated by the two propellers is approximated as a linear function of the squared motor rotational speeds, $\omega_{m_l}^2$ and $\omega_{m_r}^2$ for the left and right motors, respectively. The resulting model is expressed in eq. (1).

$$f_\tau(\mathbf{u}) = \tau = c_\tau(\omega_{m_l}^2 + \omega_{m_r}^2) \tag{1}$$

This simple representation is valid for hover and low-speed flight, where inflow angle and induced-velocity coupling are small. The identified specific thrust coefficient c_τ is $7.35 \times 10^{-6}\,\mathrm{m\,rad^{-2}}$.

B. Φ-Theory Aerodynamic Model

The aerodynamic model follows the Φ-theory formulation for tail-sitter aircraft. This framework expresses specific aerodynamic forces and moments directly in terms of body-axis velocity components and their magnitude, thereby avoiding the singularities and discontinuities associated with traditional angle-of-attack and sideslip parameterizations. [14, 15]

Φ-theory describes the forces and moments in the forward-flight body frame, which is obtained by a 90° rotation about the y-axis relative to the hover frame. The relationship between the hover body-frame velocities $(\mathbf{v}_b, \boldsymbol{\omega}_b)$ and their forward-flight equivalents $(\mathbf{v}_{\text{ff}}, \boldsymbol{\omega}_{\text{ff}})$ is given by this rotation in eq. (2).

$$\begin{bmatrix} v_{\text{ff}_x} \\ v_{\text{ff}_y} \\ v_{\text{ff}_z} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}}_{R_{\text{ff}\leftarrow\text{b}}} \begin{bmatrix} v_{b_x} \\ v_{b_y} \\ v_{b_z} \end{bmatrix}, \quad \begin{bmatrix} \omega_{\text{ff}_x} \\ \omega_{\text{ff}_y} \\ \omega_{\text{ff}_z} \end{bmatrix} = R_{\text{ff}\leftarrow\text{b}} \begin{bmatrix} \omega_{b_x} \\ \omega_{b_y} \\ \omega_{b_z} \end{bmatrix} \tag{2}$$

This rotation reorients the velocity and angular velocity vectors such that the x_{ff}-axis aligns with the vehicle's forward direction, while the z_{ff}-axis points downward, maintaining consistency with conventional aircraft notation.

The Φ-theory representation of the specific aerodynamic forces and moments is shown in eqs. (3) to (5) and eqs. (6) to (8). These terms are linear in the parameters Φ_i. The resulting specific aerodynamic forces and moments in the forward-flight frame are subsequently rotated back into the hover frame according to eqs. (9) and (10).

$$f_{\text{ff}_x} = \Phi_{fx_{m\Sigma}} \left(\omega_{m_l}^2 + \omega_{m_r}^2\right) + \Phi_{fx_{vx}} \|\mathbf{v}\| v_{\text{ff}_x} \tag{3}$$

$$f_{\text{ff}_y} = \Phi_{fy_{vy}} \|\mathbf{v}\| v_{\text{ff}_y} \tag{4}$$

$$\begin{aligned} f_{\text{ff}_z} = {} & \Phi_{fz_{m\Sigma}} \left(\omega_{m_l}^2 + \omega_{m_r}^2\right) + \Phi_{fz_{vx}} \|\mathbf{v}\| v_{\text{ff}_x} \\ & + \Phi_{fz_{vz}} \|\mathbf{v}\| v_{\text{ff}_z} + \Phi_{fz_{ev\Sigma}} \left(\delta_{e_l} + \delta_{e_r}\right) \|\mathbf{v}\| v_{\text{ff}_x} \\ & + \Phi_{fz_{em\Sigma}} \left(\delta_{e_l} \omega_{m_l}^2 + \delta_{e_r} \omega_{m_r}^2\right) \end{aligned} \tag{5}$$

$$\begin{aligned} m_{\text{ff}_x} = {} & \Phi_{mx_{em\Delta}} \left(\delta_{e_l} \omega_{m_l}^2 - \delta_{e_r} \omega_{m_r}^2\right) \\ & + \Phi_{mx_{ev\Delta}} \left(\delta_{e_l} - \delta_{e_r}\right) \|\mathbf{v}\| v_{\text{ff}_x} + \Phi_{mx_{\omega 2}} \|\omega_{\text{ff}_x}\| \omega_{\text{ff}_x} \\ & + \Phi_{mx_{\omega\times}} \omega_{\text{ff}_y} \omega_{\text{ff}_z} \end{aligned} \tag{6}$$

$$\begin{aligned} m_{\text{ff}_y} = {} & \Phi_{my_{vf}} \|\mathbf{v}\| v_{\text{ff}_x} + \Phi_{my_{vz}} \|\mathbf{v}\| v_{\text{ff}_z} + \Phi_{my_{m\Sigma}} \left(\omega_{m_l}^2 + \omega_{m_r}^2\right) \\ & + \Phi_{my_{em\Sigma}} \left(\delta_{e_l} \omega_{m_l}^2 + \delta_{e_r} \omega_{m_r}^2\right) \\ & + \Phi_{my_{ev\Sigma}} \left(\delta_{e_l} + \delta_{e_r}\right) \|\mathbf{v}\| v_{\text{ff}_x} + \Phi_{my_{\omega\times}} \omega_{\text{ff}_x} \omega_{\text{ff}_z} \end{aligned} \tag{7}$$

$$\begin{aligned} m_{\text{ff}_z} = {} & \Phi_{mz_{vy}} \|\mathbf{v}\| v_{\text{ff}_y} + \Phi_{mz_{m\Delta}} \left(\omega_{m_l}^2 - \omega_{m_r}^2\right) \\ & + \Phi_{mz_{v\omega x}} \|\mathbf{v}\| \omega_{\text{ff}_x} + \Phi_{mz_{\omega\times}} \omega_{\text{ff}_x} \omega_{\text{ff}_y} \end{aligned} \tag{8}$$

$$\mathbf{f}_f(\mathbf{x}, \mathbf{u}) = \mathbf{f}_{\text{b}} = R_{\text{ff}\leftarrow\text{b}}\, \mathbf{f}_{\text{b}} \tag{9}$$

$$\mathbf{f}_m(\mathbf{x}, \mathbf{u}) = \mathbf{m}_{\text{b}} = R_{\text{ff}\leftarrow\text{b}}^{\top}\, \mathbf{m}_{\text{ff}} \tag{10}$$

The coefficient estimates resulting from a least-squares identification on hover and low-speed flight data are summarized in table I. The coefficients are all expressed in SI units.

TABLE I: Φ-Theory Model Coefficients.

Coefficient	**Value**	**Description**
$\Phi_{fx_{m\Sigma}}$	$7.35 \times 10^{-6}\,\mathrm{m\,rad^{-2}}$	Fwd thrust
$\Phi_{fx_{vx}}$	$-0.030\,\mathrm{m^{-1}}$	Drag
$\Phi_{fy_{vy}}$	$-0.008\,\mathrm{m^{-1}}$	Side force
$\Phi_{fz_{m\Sigma}}$	$0\,\mathrm{m\,rad^{-2}}$	Vert thrust
$\Phi_{fz_{vx}}$	$0\,\mathrm{m^{-1}}$	Lift (fwd speed)
$\Phi_{fz_{vz}}$	$1.9 \times 10^{-5}\,\mathrm{m^{-1}}$	Lift (vert speed)
$\Phi_{fz_{ev\Sigma}}$	$0\,\mathrm{m^{-1}\,rad^{-1}}$	Elevon-speed sum
$\Phi_{fz_{em\Sigma}}$	$0\,\mathrm{m\,rad^{-3}}$	Elevon-motor sum
$\Phi_{mx_{em\Delta}}$	$1.90 \times 10^{-5}\,\mathrm{rad^{-2}}$	Elevon-motor diff
$\Phi_{mx_{ev\Delta}}$	$0.344\,\mathrm{m^{-2}}$	Elevon-speed diff
$\Phi_{mx_{\omega 2}}$	$-0.4940\,\mathrm{rad^{-1}}$	Roll damping
$\Phi_{mx_{\omega\times}}$	$-2.18\,\mathrm{rad^{-1}}$	Roll cross-coupling (yz)
$\Phi_{my_{vf}}$	$0\,\mathrm{rad\,m^{-2}}$	Fwd damping
$\Phi_{my_{vz}}$	$-0.0888\,\mathrm{rad\,m^{-2}}$	Vert damping
$\Phi_{my_{m\Sigma}}$	$0\,\mathrm{rad^{-1}}$	Motor sum
$\Phi_{my_{em\Sigma}}$	$-4.24 \times 10^{-5}\,\mathrm{rad^{-2}}$	Elevon-motor sum
$\Phi_{my_{ev\Sigma}}$	$0.2525\,\mathrm{m^{-2}}$	Elevon-speed sum
$\Phi_{my_{\omega\times}}$	$1.262\,\mathrm{rad^{-1}}$	Pitch cross-coupling (xz)
$\Phi_{mz_{vy}}$	$-3.71 \times 10^{-3}\,\mathrm{rad\,m^{-2}}$	Side damping
$\Phi_{mz_{m\Delta}}$	$3.9 \times 10^{-5}\,\mathrm{rad^{-1}}$	Motor diff
$\Phi_{mz_{v\omega x}}$	$-1.29 \times 10^{-2}\,\mathrm{m^{-1}}$	Roll coupling
$\Phi_{mz_{\omega\times}}$	$-0.4827\,\mathrm{rad^{-1}}$	Yaw cross-coupling (xy)

IV. ANDI Controller Design

ANDI extends INDI by controlling the system one derivative order higher, enabling explicit compensation for both first-order actuators with finite actuator bandwidth and state-dependent dynamics [10]. This section derives the ANDI stabilization control law for the *Cyclone* platform and presents the design of the reference model and error controller that shapes these closed-loop dynamics. The derivation unfolds in four steps: the system dynamics are formulated in terms of state and input vectors, the linearizing control law is derived through dynamic inversion, reference and error controllers are designed to achieve desired closed-loop performance, and cascaded complementary filtering is introduced to filter the feedback without introducing delay.

A. Control Law Derivation

This subsection presents the derivation of the ANDI attitude control law for the *Cyclone* platform, following the framework established by Steffensen et al. [10]. The derivation begins by defining the state, input, and output vectors in eqs. (11) to (13) respectively.

$$\mathbf{x} = \begin{bmatrix} \omega_x & \omega_y & \omega_z & v_x & v_y & v_z \end{bmatrix}^{\top} \tag{11}$$

$$\mathbf{u} = \begin{bmatrix} \delta_{e_l} & \delta_{e_r} & \omega_{m_l}^2 & \omega_{m_r}^2 \end{bmatrix}^{\top} \tag{12}$$

$$\mathbf{z} = \begin{bmatrix} \dot{\omega}_x & \dot{\omega}_y & \dot{\omega}_z & \tau \end{bmatrix}^{\top} \tag{13}$$

The state vector $\mathbf{x}$ contains the body-frame angular rates $\boldsymbol{\omega}$ and linear velocities $\mathbf{v}$, the output vector $\mathbf{z}$ contains the body-frame angular accelerations $\dot{\boldsymbol{\omega}}$ and specific thrust τ, and the control input vector $\mathbf{u}$ contains the left and right servo deflections $\delta_{e_l}, \delta_{e_r}$ and the squared motor rotational speeds $\omega_{m_l}^2, \omega_{m_r}^2$. Squared motor rotational speed is chosen as the control input to linearize the command-to-thrust relationship, exploiting the quadratic relationship between propeller thrust and rotational speed.

A fundamental requirement of ANDI is explicit modeling of actuator dynamics as first-order, which distinguishes it from classical INDI that assumes infinitely fast actuators. The actuators are modeled as first-order systems characterized by their respective bandwidths. Specifically, motor dynamics are approximated by a bandwidth of $35\,\mathrm{rad\,s^{-1}}$, while the elevon servos exhibit slower dynamics with a bandwidth of $20\,\mathrm{rad\,s^{-1}}$. These first-order actuator dynamics are formulated in eq. (14), where the actuator bandwidths are collected in the diagonal matrix $\boldsymbol{\varepsilon}_u$.

$$\dot{\mathbf{u}} = \boldsymbol{\varepsilon}_u(\mathbf{u}_c - \mathbf{u}), \quad \boldsymbol{\varepsilon}_u = \mathrm{diag}(\varepsilon_{e_l}, \varepsilon_{e_r}, \varepsilon_{m_l^2}, \varepsilon_{m_r^2}) \tag{14}$$

The controlled outputs are angular acceleration and specific thrust, defined in eq. (15).

$$\mathbf{z} = \begin{bmatrix} \dot{\boldsymbol{\omega}} \\ \tau \end{bmatrix} = \begin{bmatrix} \mathbf{f}_m(\mathbf{x}, \mathbf{u}) \\ f_\tau(\mathbf{u}) \end{bmatrix} \tag{15}$$

In Equation (15), both $\mathbf{f}_m(\mathbf{x}, \mathbf{u})$ and $f_\tau(\mathbf{u})$ are derived from the aerodynamic and thrust models detailed in section III. The reactive torque from the motors, and the moment generated by elevon angular velocity are neglected in this formulation. Although compensation terms proportional to motor angular accelerations and elevon angular rates can be included [16], these contributions are small compared to the dominant aerodynamic and thrust moments and are further minimized by the slow actuator dynamics.

Differentiating eq. (15) with respect to time expresses the output dynamics as functions of the state and input derivatives, yielding the state-dependent contribution F_x and the control effectiveness matrix F_u. The Jacobians are computed symbolically using the `MATLAB Symbolic Toolbox` from the aerodynamic model.

$$\dot{\mathbf{z}} = \begin{bmatrix} \ddot{\boldsymbol{\omega}} \\ \dot{\tau} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\partial}{\partial \mathbf{x}} \mathbf{f}_m(\mathbf{x}, \mathbf{u}) \\ \frac{\partial}{\partial \mathbf{x}} f_\tau(\mathbf{u}) \end{bmatrix}}_{F_x} \dot{\mathbf{x}} + \underbrace{\begin{bmatrix} \frac{\partial}{\partial \mathbf{u}} \mathbf{f}_m(\mathbf{x}, \mathbf{u}) \\ \frac{\partial}{\partial \mathbf{u}} f_\tau(\mathbf{u}) \end{bmatrix}}_{F_u} \dot{\mathbf{u}} \tag{16}$$

The term F_x represents the sensitivity of the dynamics to state changes. In prior work by De Ponti et al. [11], this term is neglected under the assumption that the contribution of the state-dependent dynamics $F_x\dot{\mathbf{x}}$ is significantly smaller than $F_u\dot{\mathbf{u}}$. This work retains $F_x\dot{\mathbf{x}}$ to evaluate its impact on tracking performance when state-dependent effects are significant.

The magnitude of the state-dependent term $F_x\dot{\mathbf{x}}$ scales with the rate of state change. For the attitude control problem, this becomes significant during aggressive maneuvers where angular accelerations are large, as the aerodynamic forces and moments that comprise F_x scale with both angular velocity and angular acceleration. Conversely, in quasi-steady-state flight with small angular rates, the state-dependent contribution becomes negligible.

The linearizing control law is obtained by solving eq. (16) for the commanded actuator rate $\dot{\mathbf{u}}_c$, assuming F_u has full row rank in the operational region, admitting a right pseudo-inverse. Introducing the pseudo-control vector $\boldsymbol{\nu} = [\ddot{\boldsymbol{\omega}}^\top \; \dot{\tau}]^\top$ yields the control law in eq. (17).

$$\dot{\mathbf{u}}_c = F_u^\dagger \left(\boldsymbol{\nu} - F_x\dot{\mathbf{x}}\right) \tag{17}$$

Incorporating the actuator dynamics from eq. (14) by substituting into eq. (17) and solving for $\mathbf{u}_c$ yields the incremental control law in eqs. (18) and (19).

$$\mathbf{u}_c = \boldsymbol{\varepsilon}_u^{-1}\dot{\mathbf{u}}_c + \mathbf{u} \tag{18}$$

$$\mathbf{u}_c = \boldsymbol{\varepsilon}_u^{-1} F_u^\dagger \left(\boldsymbol{\nu} - F_x\dot{\mathbf{x}}\right) + \mathbf{u} \tag{19}$$

The commanded input $\mathbf{u}_c$ is computed by adding an increment to the current actuator state $\mathbf{u}$. This increment comprises two components: the first term $\boldsymbol{\varepsilon}_u^{-1}F_u^\dagger\boldsymbol{\nu}$ responds to the reference jerk and specific thrust rate, while the second term $-\boldsymbol{\varepsilon}_u^{-1}F_u^\dagger F_x\dot{\mathbf{x}}$ compensates for state-dependent dynamics. The bandwidth matrix $\boldsymbol{\varepsilon}_u$ within the pseudo-inverse accounts for heterogeneous actuator dynamics: faster actuators (larger bandwidth) receive proportionally smaller commands to match the response of slower actuators.

To verify the inversion of the system, substituting eq. (19) into eq. (16) yields eq. (20), confirming that the output derivative $\dot{\mathbf{z}}$ equals the pseudo-control input $\boldsymbol{\nu}$, given no disturbances and perfect model and state knowledge.

$$\begin{aligned} \dot{\mathbf{z}} &= F_x\dot{\mathbf{x}} + F_u\dot{\mathbf{u}} \\ &= F_x\dot{\mathbf{x}} + F_u\boldsymbol{\varepsilon}_u(\mathbf{u}_c - \mathbf{u}) \\ &= F_x\dot{\mathbf{x}} + F_u\boldsymbol{\varepsilon}_u\boldsymbol{\varepsilon}_u^{-1}F_u^\dagger\left(\boldsymbol{\nu} - F_x\dot{\mathbf{x}}\right) \\ &= F_x\dot{\mathbf{x}} + I\left(\boldsymbol{\nu} - F_x\dot{\mathbf{x}}\right) \\ &= \boldsymbol{\nu} \end{aligned} \tag{20}$$

This confirms that the ANDI control law, under ideal conditions, successfully linearizes and decouples the system dynamics, achieving the desired output behavior defined by the pseudo-control input $\boldsymbol{\nu}$, despite the presence of state-dependent dynamics and first-order actuators with finite bandwidths.

In practice, perfect inversion fails due to actuator saturation and model-plant mismatches from unmodeled effects and disturbances. To address these practical limitations, Weighted Least Squares (WLS) control allocation is employed for allocating actuator commands, while a reference model smooths pilot inputs and an error controller compensates for remaining model mismatches by shaping closed-loop disturbance rejection dynamics.

B. WLS Control Allocation

The system is fully-actuated, providing four independent control effectors for four controlled outputs (angular acceleration in three axes and specific thrust). However, during aggressive maneuvers or transitions between flight modes, actuator saturation is likely to occur, effectively rendering the system underactuated. To maintain predictable and stable behavior, WLS control allocation is employed instead of a pseudo inverse to prioritize certain control objectives when saturation occurs [17].

Rather than computing the pseudo-inverse directly, the WLS control allocator solves a constrained optimization problem each iteration n that minimizes the error while respecting actuator limits. Specifically, it computes the commanded actuator rate $\dot{\mathbf{u}}_c$ by solving a minimization problem instead of using the pseudo-inverse expressed in eq. (17). The optimization problem is formulated in eqs. (21) to (23).

$$\underset{\dot{\mathbf{u}}_c}{\text{minimize}} \left\| \mathbf{W} \left(\boldsymbol{\nu} - F_x \dot{\mathbf{x}} - F_u \dot{\mathbf{u}}_c \right) \right\|^2 \tag{21}$$

subject to the constraints:

$$\dot{\mathbf{u}}_{\text{min}} \leq \dot{\mathbf{u}}_{c,n} \leq \dot{\mathbf{u}}_{\text{max}} \tag{22}$$

$$\boldsymbol{\varepsilon}_u (\mathbf{u}_{\text{min}} - \mathbf{u}_n) \leq \dot{\mathbf{u}}_{c,n} \leq \boldsymbol{\varepsilon}_u (\mathbf{u}_{\text{max}} - \mathbf{u}_n) \tag{23}$$

where $\mathbf{W} = \text{diag}([1000, 100, 1, 10])$ is the diagonal weighting matrix containing the priority factors. The constraints in eq. (22) limit the actuator rates, while those in eq. (23) ensure that the commanded rate does not drive the actuator beyond its position limits. This formulation gracefully degrades control performance in a prioritized manner: when saturation occurs, rotation about the body x-axis is prioritized first, followed by y-axis, then specific thrust, and finally z-axis. This priority sequence reflects the vehicle's control authority constraints, where x-axis and y-axis rotations are more critical for maintaining stable flight, while z-axis rotation can safely be compromised temporarily.

C. Error Controller Design

The error controller is designed through pole-placement to achieve desired disturbance rejection dynamics in the presence of model-plant mismatches and external disturbances. Conventional linear system theory is used to design and the error controller, leveraging the fact that ANDI linearizes and decouples the system dynamics from the pseudo-control input $\boldsymbol{\nu}$ to the output $\mathbf{z}$.

Considering an additive disturbance D acting on the system at the angular acceleration and specific thrust z, as illustrated in fig. 2. This disturbance represents external forces acting on the MAV and any errors in the inversion due to model mismatch. By assuming the disturbance is independent of the state and input [18], the designed poles remain unaffected by complementary filtering. Consequently, the error controller gains can be designed without explicitly accounting for filtering, with the impact of complementary filtering analyzed subsequently in section IV-E.

The pseudo-command vector $\boldsymbol{\nu}$, containing the angular jerk and specific thrust rate, is regulated by a linear error controller designed to track the reference commands and reject disturbances with desired dynamics, as given in eq. (24). A third-order parallel error controller is used for the angular jerk and first-order parallel error controller is used for the specific thrust rate, this allows for attitude and specific thrust reference tracking.

$$\boldsymbol{\nu} = \begin{bmatrix} \ddot{\boldsymbol{\omega}}_r + (\dot{\boldsymbol{\omega}}_r - \dot{\boldsymbol{\omega}}) K_{e_3} + (\boldsymbol{\omega}_r - \boldsymbol{\omega}) K_{e_2} + 2\,\text{vec}(\mathbf{q}_r \circ \mathbf{q}^*) K_{e_1} \\ \dot{\tau}_r + (\tau_r - \tau) k_{e_\tau} \end{bmatrix} \tag{24}$$

The complete decoupled linearized system including the disturbances and error controller in the Laplace Domain is shown in eqs. (25) and (26) for each attitude axis $i \in \{x, y, z\}$ and specific thrust τ respectively. The gains $k_{e_1,i}, k_{e_2,i}, k_{e_3,i}$ are the diagonal elements of the gain matrices $K_{e_1}, K_{e_2}, K_{e_3}$ corresponding to axis i and k_{e_τ} is the gain for the specific thrust error. It can be seen that the error controller shapes the closed-loop response from disturbance D to attitude and specific thrust output Y, but does not affect the reference tracking dynamics.

$$Y_i = Y_{\text{ref}_i} + D_i \frac{s}{s^3 + k_{e_3,i} s^2 + k_{e_2,i} s + k_{e_1,i}} \tag{25}$$

$$Y_\tau = Y_{\text{ref}_\tau} + D_\tau \frac{s}{s + k_{e_\tau}} \tag{26}$$

Through pole-placement, the controller gains are designed to achieve the desired closed-loop disturbance rejection dynamics, expressed in terms the natural frequency, damping ratio, and pseudo-actuator bandwidth.

The pseudo-actuator bandwidth represents the bandwidth of the hypothetical single actuator that drives a specific output. In reality, any output is typically driven by a combination of multiple physical actuators, each with finite bandwidth. These physical actuator dynamics are mapped to the output space through the control effectiveness matrix F_u. If the pseudo-actuator bandwidth were set higher than the physical actuators can achieve, the controller would demand responses faster than the real actuators can deliver, driving them into saturation and nonlinear operation. To prevent this, the pseudo-actuator bandwidth is constrained by the dominant actuators affecting that channel. Accordingly, roll and specific thrust are limited by the motor bandwidth ($\varepsilon_{m_l^2}, \varepsilon_{m_r^2}$), while pitch and yaw are limited by the elevon bandwidth ($\varepsilon_{e_l}, \varepsilon_{e_r}$).

The gains on the angular acceleration error and the specific thrust error directly define the desired pseudo-actuator bandwidths $k_{e_3,i} = \varepsilon_{y_i}, k_{e_3,\tau} = \varepsilon_{y_\tau}$. The remaining attitude error controller gains are derived by expressing the closed-loop transfer function as a product of a second-order system (with natural frequency and damping ratio) and an additional pole, as shown in eq. (27). The additional pole must be set such that $k_{e_3,i}$ is equal to the pseudo-actuator bandwidth ε_{y_i}.

$$Y_i = D_i \frac{s}{(s^2 + 2\zeta_i \omega_{n_i} s + \omega_{n_i}^2)(s + p_i)} \tag{27}$$

Equating the characteristic polynomials of eq. (25) and eq. (27) yields the expression of the error controller gains in terms of the desired closed-loop dynamics parameters.

$$\begin{aligned} & s^3 + k_{e_3,i} s^2 + k_{e_2,i} s + k_{e_1,i} \\ & = (s^2 + 2\zeta_i \omega_{n_i} s + \omega_{n_i}^2)(s + p_i) \\ & = s^3 + (2\zeta_i \omega_{n_i} + p_i) s^2 + (\omega_{n_i}^2 + 2\zeta_i \omega_{n_i} p_i) s + \omega_{n_i}^2 p_i \\ & = s^3 + \underbrace{\varepsilon_{y_i}}_{k_{e_3,i}} s^2 + \underbrace{(\omega_{n_i}^2 + 2\zeta_i \omega_{n_i} \varepsilon_{y_i} - 4\zeta_i^2 \omega_{n_i}^2)}_{k_{e_2,i}} s + \underbrace{\omega_{n_i}^2 \varepsilon_{y_i} - 2\zeta_i \omega_{n_i}^3}_{k_{e_1,i}} \end{aligned} \tag{28}$$

The error controller dynamics can now be intuitively tuned through the selection of the pseudo-actuator bandwidth ε_{y_i}, natural frequency ω_{n_i}, and damping ratio ζ_i for each channel

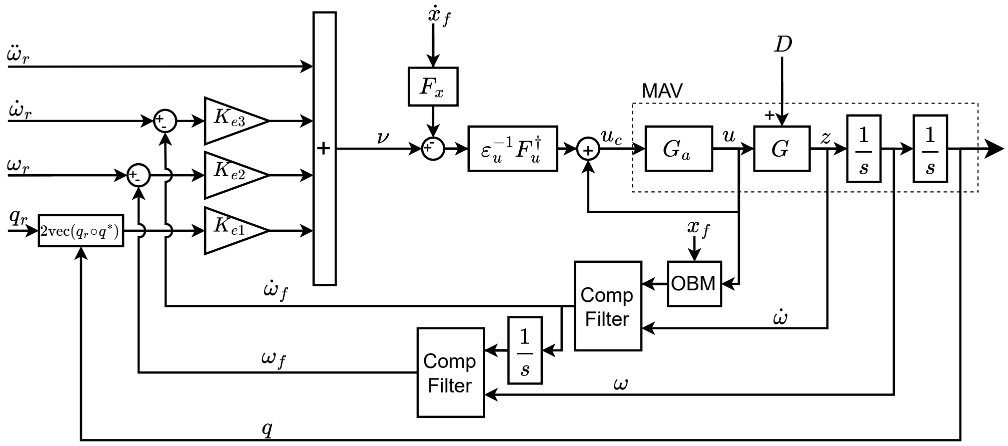


Fig. 2: ANDI attitude control of system G with actuators G_a. The error controller generates pseudo-control input $\boldsymbol{\nu}$ from attitude errors and reference commands. The incremental control law inverts system dynamics while compensating state-dependent effects $F_x\dot{\mathbf{x}}$ to compute actuator commands. Cascaded complementary filters fuse noisy measurements with model-based estimates to provide undelayed feedback of angular acceleration and rates; translational filtering is omitted for clarity.

$i \in \{x, y, z\}$. Table II summarizes the chosen parameters for the *Cyclone* platform. The pseudo-actuator bandwidths are chosen equal to the dominant physical actuator bandwidths on each channel: $\varepsilon_{y_x} = \varepsilon_{y_\tau} = 35\,\mathrm{rad\,s^{-1}}$ (motor-limited) and $\varepsilon_{y_y} = \varepsilon_{y_z} = 20\,\mathrm{rad\,s^{-1}}$ (elevon-limited). Critical damping is selected for all axes to avoid overshoot in response to disturbances while maintaining a fast response. Lowering the damping ratio will result in a faster reduction in error, at the cost of overshoot and even oscillations. The desired natural frequencies are tuning parameters that govern the speed of disturbance rejection and have been empirically tuned through simulation and flight testing. A higher natural frequency results in a faster response, but due to the presence of unmodeled effects such as transmission delay to the actuators and lag in the sensor feedback introduced by filtering, excessively high natural frequencies lead to instability.

TABLE II: Controller Gain Design Parameters.

Parameter	x	y	z	τ
ε_y (rad s^{-1})	35.0	20.0	20.0	35.0
ω_n (rad s^{-1})	7.0	7.0	7.0	-
ζ (-)	1.0	1.0	1.0	-

D. Reference Model Design

While the error controller shapes the closed-loop response to disturbances, the reference model shapes the response to pilot commands. ANDI inverts the complete system dynamics, including actuator dynamics, thereby removing the natural nonlinear aircraft dynamics and first-order actuator dynamics from the closed-loop response. As a result, the reference model dynamics directly equal the closed-loop command-to-output dynamics $\frac{Y}{U} = \frac{Y_{\mathrm{ref}}}{U}$. This means that by carefully designing the reference model, the desired closed-loop command tracking behavior can be directly specified and implemented.

The desired closed-loop command-to-output dynamics are specified in terms of the natural frequency ω_{n_i}, damping ratio ζ_i, and pseudo-actuator bandwidth ε_{y_i}, consistent with the error controller design philosophy. The attitude reference is a second order system cascaded with the pseudo-actuator dynamics, while the specific thrust reference is a first-order system following the pseudo-actuator dynamics. The desired closed-loop transfer functions from command input U to output Y for each attitude axis $i \in \{x, y, z\}$ and specific thrust are given in eqs. (29) and (30).

$$Y_i = U_i \frac{\omega_{n_i}^2 \varepsilon_{y_i}}{(s^2 + 2\zeta_i \omega_{n_i} s + \omega_{n_i}^2)(s + \varepsilon_{y_i})} \tag{29}$$

$$Y_\tau = U_\tau \frac{\varepsilon_{y_\tau}}{s + \varepsilon_{y_\tau}} \tag{30}$$

The bandwidth parameters ε_y should not exceed the dominant physical actuator bandwidths: roll and thrust are limited by motor bandwidth $(\varepsilon_{m_l^2}, \varepsilon_{m_r^2})$, while pitch and yaw are limited by elevon bandwidth $(\varepsilon_{e_l}, \varepsilon_{e_r})$.

To achieve these desired dynamics, a cascaded reference model generates the required reference signals from the pilot command input U. The reference model architecture is shown in fig. 3, generating attitude, angular rate, acceleration, and jerk references for the error controller. To handle large attitude changes and avoid singularities, exact quaternion kinematics with logarithmic mapping $\log(\cdot)$ is used [19], while a first-order model handles the specific thrust reference. The reference model can be augmented by saturating the desired angular rates, accelerations, and jerk to further prevent actuator saturation. However, this saturation introduces nonlinearities into the reference model, complicating the closed-loop analysis. For simplicity, reference saturation is omitted unless explicitly specified.

The reference model structure in the Laplace domain for each attitude axis $i \in \{x, y, z\}$ and specific thrust is given in

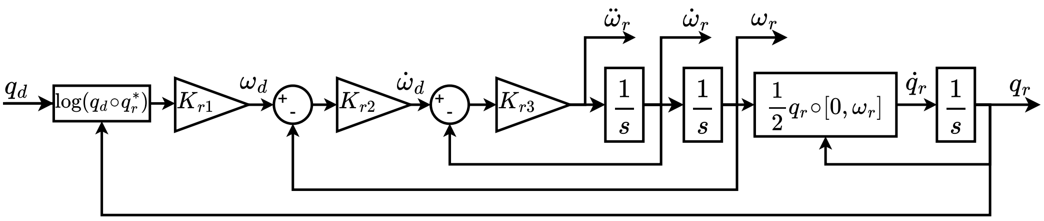


Fig. 3: Third-order attitude reference model. Converts desired $\mathbf{q}_d$ into smooth references $\mathbf{q}_r, \boldsymbol{\omega}_r, \dot{\boldsymbol{\omega}}_r, \ddot{\boldsymbol{\omega}}_r$ using exact quaternion kinematics and logarithmic mapping.

eqs. (31) and (32).

$$Y_i = Y_{\text{ref},i} = U_i \frac{k_{r_1,i} k_{r_2,i} k_{r_3,i}}{s^3 + k_{r_3,i} s^2 + k_{r_2,i} k_{r_3,i} s + k_{r_1,i} k_{r_2,i} k_{r_3,i}} \tag{31}$$

$$Y_\tau = Y_{\text{ref},\tau} = U_\tau \frac{k_{r_\tau}}{s + k_{r_\tau}} \tag{32}$$

The gains $k_{r_1,i}, k_{r_2,i}, k_{r_3,i}$ are the diagonal elements of the gain matrices $K_{r_1}, K_{r_2}, K_{r_3}$ corresponding to axis i, and k_{r_τ} is the gain for the specific thrust reference. These gains are computed by matching the characteristic polynomials of the reference model to the desired closed-loop dynamics.

The reference model gains are derived by equating the characteristic polynomials of eq. (31) and the desired dynamics in eqs. (29) and (30). Since all transfer functions have zero steady-state gain, only the characteristic polynomials are matched. For the specific thrust, equating the characteristic polynomials simply yields $k_{r_\tau} = \varepsilon_{y_\tau}$. For the attitude channels, the characteristic polynomial equation yields eq. (33).

$$\begin{aligned} & s^3 + k_{r_3,i} s^2 + k_{r_2,i} k_{r_3,i} s + k_{r_1,i} k_{r_2,i} k_{r_3,i} \\ & = (s^2 + 2\zeta_i \omega_{n_i} s + \omega_{n_i}^2)(s + \varepsilon_{y_i}) \\ & = s^3 + \underbrace{(2\zeta_i \omega_{n_i} + \varepsilon_{y_i})}_{k_{r_3,i}} s^2 + \underbrace{(\omega_{n_i}^2 + 2\zeta_i \omega_{n_i} \varepsilon_{y_i})}_{k_{r_2,i} k_{r_3,i}} s + \underbrace{\omega_{n_i}^2 \varepsilon_{y_i}}_{k_{r_1,i} k_{r_2,i} k_{r_3,i}} \end{aligned} \tag{33}$$

Solving for the reference model gains individually yields the expressions in eq. (34).

$$\begin{aligned} k_{r_1,i} &= \frac{\omega_{n_i}^2 \varepsilon_{y_i}}{\omega_{n_i}^2 + 2\zeta_i \omega_{n_i} \varepsilon_{y_i}} \\ k_{r_2,i} &= \frac{\omega_{n_i}^2 + 2\zeta_i \omega_{n_i} \varepsilon_{y_i}}{2\zeta_i \omega_{n_i} + \varepsilon_{y_i}} \\ k_{r_3,i} &= 2\zeta_i \omega_{n_i} + \varepsilon_{y_i} \\ k_{r_\tau} &= \varepsilon_{y_\tau} \end{aligned} \tag{34}$$

To simplify tuning and reduce the number of free parameters, the reference model parameters are typically chosen to match those of the error controller: $\omega_{n_i}, \zeta_i, \varepsilon_{y_i}$ are identical across both controllers as summarized in table II. This unified approach ensures consistent closed-loop behavior for both reference tracking and disturbance rejection.

E. Cascaded Complementary Filtering

The ANDI control law in eq. (19) relies on accurate, undelayed state estimates and feedback. Raw IMU measurements contain noise, making filtering necessary. However, filtering introduces phase lag that degrades disturbance rejection and can destabilize the closed-loop dynamics. The cascaded complementary filtering approach presented in Steffensen et al. [9] is adopted to provide undelayed estimates of angular rates, angular accelerations, body velocities, and body accelerations for feedback and compensation of state-dependent effects in the control law.

No filtering is applied to actuator-feedback signals because they exhibit sufficiently low noise. The controller computes the specific thrust τ directly from the measured motor speeds without additional filtering.

A complementary filter fuses the high-frequency components of a noise-free OBM with the low-frequency components of a noisy measurement to produce a composite estimate that leverages the strengths of both sources.

For the OBM, no additional model knowledge is required beyond what is already used in the ANDI control law. The OBM, derived in section III, is evaluated at the current actuator commands and state estimates to produce model-based estimates of angular acceleration $\mathbf{f}_m(\mathbf{x}, \mathbf{u}) = \dot{\boldsymbol{\omega}}_{\text{obm}}$ and body acceleration $\mathbf{f}_f(\mathbf{x}, \mathbf{u}) = \mathbf{a}_{\text{obm}}$. If no state-dependent knowledge is available, a simplified OBM that neglects state dependence can be used instead, at the cost inaccuracies in the estimated signals.

The angular acceleration and angular rate estimates come from cascaded complementary filters, expressed in the Laplace domain in eq. (35). The angular acceleration measurement $\dot{\boldsymbol{\omega}}_{\text{meas}}$ results from numerically differentiating the gyroscope measurements, while the model-based estimate $\dot{\boldsymbol{\omega}}_{\text{obm}}$ comes from evaluating the OBM at the current actuator commands and state estimates. Integrating the filtered angular acceleration $\dot{\boldsymbol{\omega}}_f$ and combining it with the gyroscope angular rate measurement $\boldsymbol{\omega}_{\text{meas}}$ yields the filtered angular rate $\boldsymbol{\omega}_f$.

$$\dot{\omega}_{f_i} = H_{\dot{\omega}_i}(s)\, \dot{\omega}_{\text{meas}_i} + (1 - H_{\dot{\omega}_i}(s))\, \dot{\omega}_{\text{obm}_i} \tag{35}$$

$$\omega_{f_i} = H_{\omega_i}(s)\, \omega_{\text{meas}_i} + \frac{1}{s}(1 - H_{\omega_i}(s))\, \dot{\omega}_{f_i} \tag{36}$$

Similarly, the controller estimates body acceleration and velocity for state-dependent dynamics compensation through complementary filtering, as shown in eq. (37). The acceleration measurement $\mathbf{a}_{\text{meas}}$ comes from the IMU accelerometer, while the model-based estimate $\mathbf{a}_{\text{obm}}$ comes from evaluating the OBM at the current actuator commands and state estimates. Integrating the filtered acceleration $\mathbf{a}_f$ and combining it with a velocity estimate $\mathbf{v}_{\text{meas}}$ yields the filtered velocity $\mathbf{v}_f$.

$$a_{f_i} = H_{a_i}(s)\, a_{\text{meas}_i} + (1 - H_{a_i}(s))\, a_{\text{obm}_i} \tag{37}$$

$$v_{f_i} = H_{v_i}(s)\, v_{\text{meas}_i} + \frac{1}{s}(1 - H_{v_i}(s))\, a_{f_i} \tag{38}$$

Careful consideration is required when implementing the complementary filters to avoid numerical instability. Any constant bias or non-zero mean in the angular acceleration measurements $\dot{\boldsymbol{\omega}}_{\text{meas}}$ or translational acceleration measurements $\mathbf{a}_{\text{meas}}$, regardless of magnitude, causes the integrated estimates to grow linearly without bound. As these integrated values become large, floating-point arithmetic loses precision and can eventually lead to numerical instability. This can be mitigated by applying the high-pass filters to both angular and translational acceleration measurements before integrating, effectively removing the mean value and preventing unbounded growth.

The undelayed state estimate $\mathbf{x}$ used in the control law eq. (19) is constructed from the filtered angular rates $\boldsymbol{\omega}_f$ and filtered body velocities $\mathbf{v}_f$. The derivative $\dot{\mathbf{x}}$ is estimated using the filtered angular accelerations $\dot{\boldsymbol{\omega}}_f$ and filtered body accelerations $\mathbf{a}_f$. These undelayed estimates are used to compute and update the control effectiveness matrix F_u and state-dependent matrix F_x in real time, enabling the effectiveness scheduling described in section IV-A.

The low-pass filters $H_{\dot{\omega}_i}(s), H_{\omega_i}(s), H_{a_i}(s), H_{v_i}(s)$ are chosen as Butterworth filters and are designed to attenuate high-frequency noise while minimizing phase lag. The filters on the level of acceleration are second-order, while the filters on angular rate and velocity are first-order, providing one order higher filtering to the noisier derivative signals. The transfer functions for the first-order and second-order Butterworth filters are given in eqs. (39) and (40), where ω_c is the cutoff frequency specific to each filter.

$$H_1(s) = \frac{\omega_c}{s + \omega_c} \tag{39}$$

$$H_2(s) = \frac{\omega_c^2}{s^2 + \sqrt{2}\omega_c s + \omega_c^2} \tag{40}$$

Table III summarizes the empirically tuned filter cutoff frequencies, which are chosen to be identical for all axes.

TABLE III: Complementary Butterworth Filter Specifications.

Filter	**Purpose**	**Order**	**Cutoff Frequency**
$H_{\dot{\omega}}(s)$	Angular acceleration	2nd	$\omega_{c_{\dot{\omega}}} = 20\,\mathrm{rad\,s^{-1}}$
$H_{\omega}(s)$	Angular rate	1st	$\omega_{c_{\omega}} = 80\,\mathrm{rad\,s^{-1}}$
$H_{a}(s)$	Translational acceleration	2nd	$\omega_{c_{a}} = 20\,\mathrm{rad\,s^{-1}}$
$H_{v}(s)$	Translational velocity	1st	$\omega_{c_{v}} = 80\,\mathrm{rad\,s^{-1}}$

This analysis relies on the assumption that all disturbances are exogenous and independent of the system states and inputs. Under this assumption, complementary filtering does not alter the command-output dynamics $\frac{Y}{U}$. However, it does affect the disturbance rejection dynamics $\frac{Y}{D}$ in eq. (27) by introducing additional poles and zeros. To characterize this effect, the closed-loop transfer function for the attitude error controller with filtering is derived by substituting the complementary filters into the closed-loop dynamics, illustrated in fig. 2. The resulting transfer function for each axis $i \in \{x, y, z\}$ is given in eq. (41).

$$\frac{Y_i}{D_i} = \frac{s^2 + (1-H_{\dot{\omega}_i}(s))k_{e_{3,i}}s + (1-H_{\dot{\omega}_i}(s))(1-H_{\omega_i}(s))k_{e_{2,i}}}{s(s^3 + k_{e_{3,i}}s^2 + k_{e_{2,i}}s + k_{e_{1,i}})} \tag{41}$$

Substituting the Butterworth filter transfer functions from eqs. (39) and (40) into eq. (41) and simplifying yields the closed-loop characteristic polynomial with filtering, shown in eq. (42).

$$\underbrace{(s + \omega_{c_\omega,i})}_{H_\omega}\underbrace{(s^2 + \sqrt{2}\omega_{c_{\dot{\omega}},i}s + \omega_{c_{\dot{\omega}},i}^2)}_{H_{\dot{\omega}}}(s^3 + k_{e_{3,i}}s^2 + k_{e_{2,i}}s + k_{e_{1,i}}) \tag{42}$$

The characteristic polynomial retains the designed poles from eq. (27) in addition to the Butterworth filter poles $H_{\dot{\omega}_i}(s)$ and $H_{\omega_i}(s)$. These new poles all lie in the left half-plane (for positive filter cutoff frequencies ω_c), showing that stability is not affected by the filtering. The poles and zeros introduced by the filters do influence the response, this impact is minimized by only doing the minimal filtering needed to attenuate the noise. This influence on the response is analyzed in section VI-B.

F. Controller Overview

To anchor the design, fig. 2 summarizes the attitude ANDI controller described in this section. The figure illustrates the error controller eq. (24), the incremental control law eq. (19), and the cascaded complementary filters eq. (35) that provide undelayed angular acceleration and rate estimates. Although not shown in the figure for clarity, the controller also employs translational complementary filters eq. (37) to estimate body velocities and accelerations for state-dependent dynamics compensation. The state-dependent term $F_x\dot{\mathbf{x}}$ in the incremental control law becomes significant during aggressive maneuvers with large angular accelerations, where compensation can improve tracking performance; during nominal flight with small angular rates, this compensation becomes negligible. The reference model is presented separately in fig. 3 and provides the desired state references to the controller.

The ANDI control law and all associated filters are discretized and implemented to run at 500 Hz, yielding a Nyquist frequency of 250 Hz. The sampling period of 2 ms is short relative to all relevant dynamics: the Nyquist frequency is well above both the closed-loop bandwidth (approximately 4.3 Hz for roll and 3.4 Hz for pitch and yaw) and the complementary filter cutoff frequencies (up to $80\,\mathrm{rad\,s^{-1}}$, or 12.7 Hz). Consequently, the effects of discretization are negligible, and the continuous-time analysis and gain design presented in this section remain valid in the discrete-time implementation.

V. ANDI Equivalence to INDI with Actuator Dynamics

ANDI can be interpreted as a more generic form of INDI, which can compensate for the state-dependent dynamics and first-order actuator dynamics. To illustrate this equivalence, the closed-loop dynamics of ANDI and INDI applied to a system with first-order actuator dynamics are compared. A simplified Single Input Single Output (SISO) representation of the rotational dynamics about one axis is considered without loss of generality. The output Y represents the attitude angle about the axis of interest, while the pseudo-control input ν_{indi} is at the level of angular acceleration. The actuator dynamics are first order with bandwidth ε.

The first necessary condition for equivalence is that all actuators coupled to a single output in INDI must have identical dynamics. Since INDI does not compensate for actuator dynamics, mismatched dynamics directly corrupt the output. Therefore, the actuator-to-output mapping by the control effectiveness matrix must only couple actuators with identical dynamics [9].

In addition, ANDI should not compensate for the state-dependent dynamics, i.e., $F_x\dot{\mathbf{x}} = \mathbf{0}$. This will force the controller to treat the state-dependent dynamics as part of the external disturbances D, similar to INDI [18].

The closed-loop dynamics of INDI with first-order actuator dynamics $G_a = \frac{\varepsilon}{s+\varepsilon}$ are given by eq. (43) [9], while the closed-loop dynamics of ANDI without compensation of the state-dependent effects are shown in eq. (44). Here the pseudo-control vector ν_{andi} is at the level of angular jerk, while ν_{indi} is at the level of angular acceleration.

$$Ys^2 = \nu_{\text{indi}} G_a + D(I - G_a) \tag{43}$$

$$Ys^2 = \nu_{\text{andi}} + D \tag{44}$$

These dynamics are visualized in fig. 4, which presents block diagram representations of both control approaches., completed with the error controller and reference model for attitude control.

A. Error Rejection Equivalence

The pseudo-control input of INDI is constructed by a second-order error controller to track the desired attitude, angular rate, and acceleration, shown in eq. (45). Note that the reference model is synchronized through the actuator dynamics G_a to account for the phase lag introduced by the actuator dynamics in the output.

$$\nu_{\text{indi}} = Y_{\text{ref}} s^2 + (Y_{\text{ref}} G_a s - Ys)\tilde{k}_{e_2} + (Y_{\text{ref}} G_a - Y)\tilde{k}_{e_1} \tag{45}$$

The resulting closed-loop INDI dynamics obtained by substituting the pseudo-control input into eq. (43) are shown in eq. (46). Making use of the fact that G_a is a first-order system with bandwidth ε, the closed-loop dynamics can be rearranged to the form shown in eq. (47).

$$Y = Y_{\text{ref}} G_a + D \frac{I - G_a}{s^2 + G_a \tilde{k}_{e_2} s + G_a \tilde{k}_{e_1}} \tag{46}$$

$$Y = Y_{\text{ref}} \frac{\varepsilon}{s+\varepsilon} + D \frac{s}{s^3 + \underbrace{\varepsilon}_{k_{e_3}} s^2 + \underbrace{\varepsilon \tilde{k}_{e_2}}_{k_{e_2}} s + \underbrace{\varepsilon \tilde{k}_{e_1}}_{k_{e_1}}} \tag{47}$$

The equivalence in disturbance rejection dynamics can be established by comparing eq. (47) to the closed-loop dynamics of ANDI shown in eq. (25). Both controllers react to disturbances as a third-order system, where INDI contains the actuator bandwidth ε explicitly in the characteristic equation. Therefore, ANDI and INDI exhibit identical disturbance rejection dynamics when the ANDI controller gains are chosen to reintroduce the actuator bandwidth. This means that the pseudo-actuator bandwidth ε_y in ANDI is set equal to the actual actuator bandwidth ε.

The same conclusion holds for the specific thrust output dynamics, where ANDI and INDI exhibit identical disturbance rejection dynamics when pseudo-actuator bandwidth ε_{y_τ} is set equal to the actual actuator bandwidth ε.

B. Reference Tracking Equivalence

INDI tracks the reference model filtered through the actuator dynamics G_a, as shown in eq. (47), while ANDI tracks the unfiltered reference model, as shown in eq. (25). This results in different reference tracking dynamics between the two controllers.

By generating unique references for both controllers, the reference tracking dynamics can be made equivalent. INDI is provided with a second-order reference model, while a third-order reference equal to the cascade of the second-order reference model and the actuator dynamics G_a is provided to ANDI, as shown in eq. (48).

$$Y_{\text{ref}_{\text{andi}}} = Y_{\text{ref}_{\text{indi}}} \frac{\varepsilon}{s+\varepsilon} \tag{48}$$

Comparing this to eq. (29), it can be seen that the ANDI reference tracking dynamics become equivalent to INDI when the pseudo-actuator bandwidth ε_y is set equal to the actual actuator bandwidth ε.

C. Summary of Equivalence Conditions

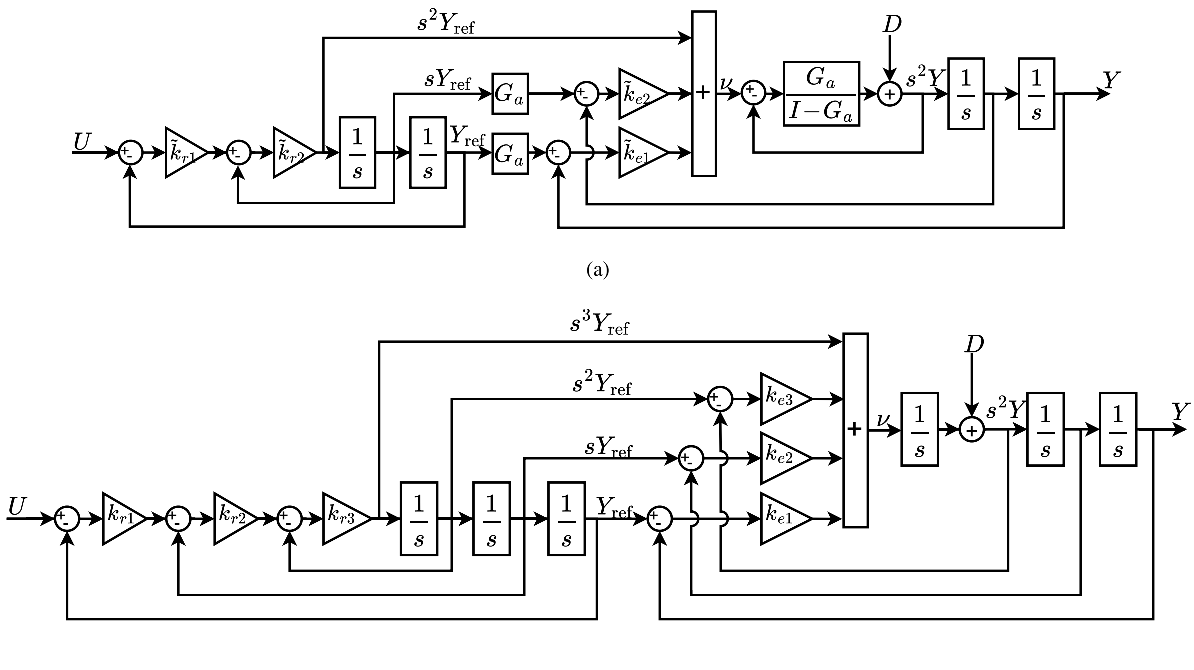
The analysis above demonstrates that ANDI and INDI exhibit equivalent control behavior when three conditions are satisfied: (i) actuators of different bandwidth are not coupled to a single output; (ii) ANDI does not compensate for state-dependent dynamics, instead treating them as disturbances; and (iii) the pseudo-actuator bandwidth is set equal to the actual actuator bandwidth. Under these conditions, both controllers achieve identical disturbance rejection and reference tracking dynamics.

This first condition is platform dependent, in case of the *Cyclone* tail-sitter platform, the yaw and pitch axes are each controlled by a pair of identical elevons, while the roll axis is controlled by a pair of identical motors. Therefore, each output axis is only coupled to actuators with identical dynamics, satisfying the first condition for equivalence.

The next two conditions can be enforced through controller design choices. In section IV, the pseudo-actuator bandwidths for attitude and specific thrust control were set equal to the measured actuator bandwidths, satisfying the third condition. Consequentially, by simply omitting the $F_x\dot{\mathbf{x}}$ term, an INDI-equivalent ANDI controller can be realized that satisfies the second condition.

VI. Results

This section presents the analysis and results of the ANDI controller for attitude stabilization on the *Cyclone* tail-sitter. All simulation studies are conducted using `Matlab`



(a)

(b)

Fig. 4: Block diagrams for attitude control of a system with first order actuators $G_a = \frac{\varepsilon}{s+\varepsilon}$: (a) Linear system resulting from INDI, and (b) linear system resulting from ANDI. When pseudo-actuator bandwidth equals actual actuator bandwidth, both systems exhibit identical disturbance rejection and reference tracking dynamics.

Simulink, while flight testing is performed in an indoor flight facility that constrains the experimental envelope to hover and low-speed flight regimes. The analysis focuses on the yaw axis, where state-dependent effects due to aerodynamic drag are most pronounced due to large angular accelerations experienced during heading changes. The results of the experiments are generalizable to the other axes, where state-dependent effects are smaller but still present.

The analysis comprises three experiments: ideal-condition tracking analysis to examine correct inversion and reference model design; disturbance rejection tests with and without complementary filtering to analyze the error controller design and filtering effects; and a direct comparison between the full ANDI controller (with F_x) and a baseline (without F_x) to quantify the practical impact of state-dependent compensation. Together, these experiments investigate whether explicit compensation of state-dependent dynamics yields measurable performance improvements and validates the ANDI framework as a practical control strategy for hybrid MAV control.

A. Inversion and Reference Model Verification in Simulation

Correct implementation of the control law and proper reference model design are verified by testing the closed-loop feedforward path in ideal conditions: perfect model knowledge, noiseless sensors, and no external disturbances. Under these conditions, the controller should achieve perfect tracking of the reference trajectory, confirming both the inversion and reference model dynamics.

A step input of 170° in heading is applied at $t = 0.1\,\text{s}$ to three controller configurations: ANDI with full state-dependent compensation, ANDI without state-dependent compensation, and an equivalent INDI controller as a baseline. The reference models differ by design: ANDI employs the third-order reference model from fig. 3 with actuator dynamics explicitly included, while INDI uses a second-order reference model that omits actuator dynamics since its control law does not explicitly compensate for them.

As established in section V, when the pseudo actuator bandwidth equals the physical actuator bandwidth, ANDI without state-dependent compensation and INDI should exhibit identical closed-loop disturbance rejection dynamics despite their different control structures. Figure 5 validates this equivalence experimentally and verifies the correctness of the control law implementation.

Figure 5 shows the reference trajectories and actual yaw attitude responses for all three controllers following the 170° step input. The ANDI reference model exhibits slower, slightly damped dynamics compared to INDI. This difference reflects the explicit inclusion of actuator dynamics in the third-order reference model. The full ANDI controller with F_x tracks its third-order reference perfectly, confirming correct inversion.

ANDI without state-dependent compensation and INDI produce identical closed-loop responses, validating the predicted equivalence. Both exhibit tracking error due to unmodeled state dynamics now acting as a disturbance on the system. The tracking error remains small indicating that the effect of the state-dependent terms are minor.

Given this equivalence and the established correctness of the control law, ANDI without state-dependent compensation is selected as the baseline controller for all subsequent comparisons. This baseline isolates the effect of state-dependent compensation by removing only the $F_x\dot{\mathbf{x}}$ term while maintaining all other aspects of the control architecture and reference model design.

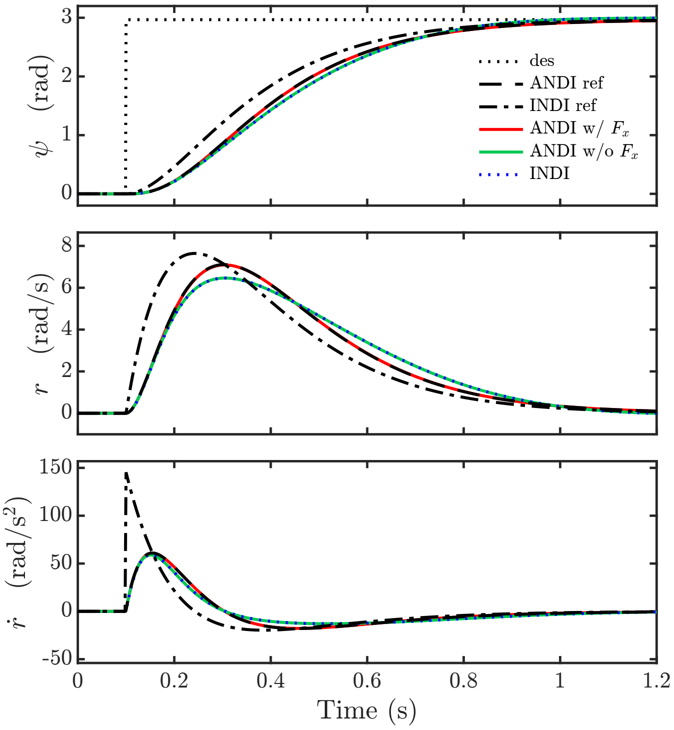


Fig. 5: Step response comparison for three controller configurations in ideal conditions, validating correct inversion and equivalence between ANDI without state effects and INDI.

B. Complementary Filtering and Disturbance Rejection in Simulation

Following verification of the control law and reference model in the feedforward path, the feedback path is now examined to assess how cascaded complementary filters affect disturbance rejection performance. During the design of the complementary filters in section IV-E, it was established that filtering should not affect closed-loop stability or reference tracking, but will influence disturbance rejection and noise attenuation.

The full ANDI controller is tested in simulation under two conditions: with complementary filtering applied to all measurement channels, and with direct unfiltered state feedback. A step disturbance of $1.0\,\mathrm{N\,m}$ is applied to the body yaw moment at $t = 0.1\,\mathrm{s}$ in hover.

Measurement noise is omitted from this analysis, allowing a direct comparison of how filtering influences the disturbance response. The filter cutoff frequencies in table III are designed based on expected sensor noise, which will be validated through flight testing in section VI-C.

Since disturbances are unmodeled by definition, any predictable perturbation would instead be part of the system dynamics. Therefore, disturbances cannot be captured in the OBM used for inversion and complementary filtering.

First, the absence of disturbance effects in the OBM means that the state-dependent term $F_x\dot{\mathbf{x}}$ cannot compensate for external disturbances. Therefore, this analysis considers only the effect of filtering on disturbance rejection for the full ANDI controller, rather than comparing with a baseline without F_x.

Second, since disturbances are absent from the OBM, they enter the feedback loop only through the low-pass filtered measurement path, incurring the full phase lag of the low-pass filters. The error controller was designed assuming instantaneous disturbance feedback, creating a mismatch between design assumption and actual behavior. This lag in disturbance estimation degrades the designed disturbance rejection, which is analyzed through the step response.

Figure 6 shows the true heading, body yaw rate, and body yaw acceleration response to the step disturbance for both filtered and unfiltered feedback conditions. The unfiltered response is equal to the designed disturbance rejection, as the controller receives immediate disturbance feedback. In contrast, the response for the filtered feedback case reacts slower to the disturbance and has higher peak error due to the lag introduced by the low-pass filters.

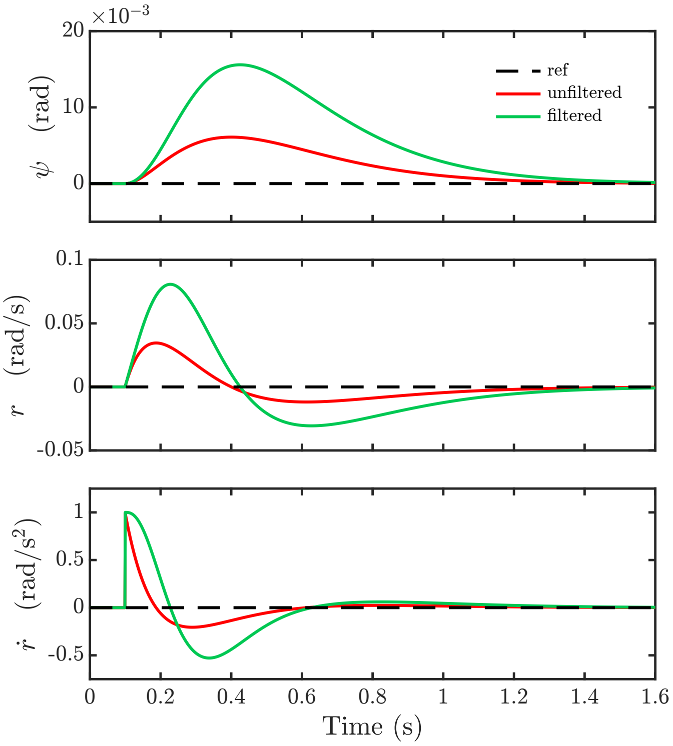


Fig. 6: Disturbance rejection comparison with and without complementary filtering. The unfiltered feedback case shows the system's designed disturbance rejection, while the filtered feedback case shows how the system's disturbance rejection degrades due to lag from low-pass filters.

The analysis demonstrates that complementary filtering, while essential for noise attenuation in real-world conditions, introduces lag that degrades disturbance rejection performance compared to ideal unfiltered feedback.

C. State-Dependent Compensation Impact in Real-World Flight Testing

Having verified the control law implementation, reference model design, and complementary filtering behavior through simulation, the final analysis assesses the practical benefits of state-dependent compensation on the physical MAV in real-world flight conditions.

To assess the practical benefits of state-dependent compensation in real-world flight conditions, a step input of 170° in heading is commanded. This maneuver creates significant angular rates and accelerations, maximizing the influence of aerodynamic drag. To prevent elevon saturation during this aggressive maneuver, the reference model is augmented to limit maximum jerk to $100\,\mathrm{rad\,s^{-3}}$ and acceleration to $20\,\mathrm{rad\,s^{-2}}$.

Three controller configurations incorporate state dynamics to varying degrees: the full ANDI includes state-dependent dynamics in both inversion and complementary filtering; the partial applies state-dependent dynamics only in complementary filtering; and the baseline excludes state-dependent dynamics entirely, equivalent to INDI. The partial configuration isolates the relative importance of state compensation in inversion versus filtering.

Figure 7 presents the estimated body yaw rate and acceleration responses for all three configurations, with mean responses and one standard deviation bounds from repeated flights, along with the corresponding true signals. True angular rate and acceleration are obtained through zero-phase low-pass filtering of raw IMU measurements post-flight. This non-causal filtering applies a second-order Butterworth filter with $20\,\mathrm{rad\,s^{-1}}$ cutoff for angular acceleration and a first-order Butterworth filter with $80\,\mathrm{rad\,s^{-1}}$ cutoff for angular rate. Zero-phase filtering works by applying the filter forwards through the recorded flight data, then applying the same filter backwards, resulting in a zero phase-lag signal that introduces no delays or model-based assumptions. Because the complete flight data is available post-flight, this non-causal, bidirectional filtering approach provides an accurate approximation of the true signals unaffected by inaccuracies in the OBM or complementary filter design.

From the estimated signals in fig. 7, all three controllers achieve stable flight and excellent tracking of the commanded heading step for both angular rate and acceleration. The reference model effectively shapes the closed-loop dynamics, with all configurations exhibiting similar overall response characteristics.

The complementary filtering effectively attenuates high-frequency noise from the raw IMU measurements, yielding clean estimates of angular rate and acceleration that confirm the chosen filter order and cutoff frequencies in table III are effective for real-world flight conditions.

The transient responses ($0.2\,\mathrm{s}$ to $0.3\,\mathrm{s}$) reveals a mismatch between estimated and true angular acceleration. This mismatch is attributed to limitations in the OBM: particularly, the model neglects higher-order effects, including moments produced by elevon rate changes and elevon inertia.

Performance differences are most evident at high angular rates and accelerations, where aerodynamic drag becomes significant. The full ANDI achieves the best tracking and accurate state estimation, as confirmed by close matches between estimated and true acceleration. The partial configuration shows moderate degradation in peak rates/accelerations but retains accurate estimates. The baseline performs worst, with inversion unable to compensate for drag and filtering overestimating acceleration, degrading disturbance rejection.

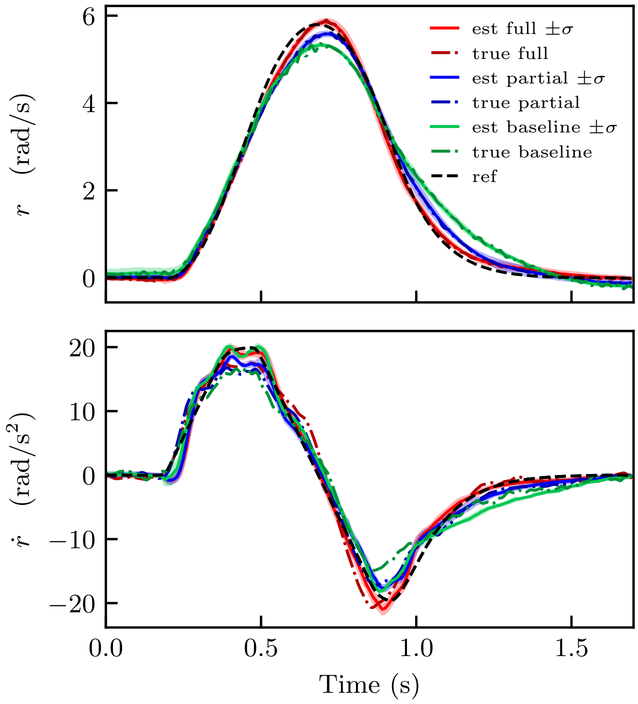


Fig. 7: Measured yaw rate and angular acceleration for a 170° heading step applied at $0.2\,\mathrm{s}$ during flight testing. Plotted are the three controller configurations: full ANDI (state effects in inversion and complementary filtering), partial ANDI (state effects only in complementary filtering), and baseline (no state effects, INDI-equivalent). True acceleration is obtained through zero-phase low-pass filtering of the raw IMU measurements.

These results match the response trends observed in simulation in fig. 5, confirming that the theoretical advantages of state-dependent compensation translate to practical performance improvements in real-world flight conditions.

Table IV summarizes Root Mean Square (RMS) tracking errors for estimated (used by the error controller) and true (actual MAV performance) yaw rate and acceleration.

TABLE IV: RMS Tracking Errors

RMS error	Full	Partial	Baseline
r_{est} ($\mathrm{rad\,s^{-1}}$)	0.146	0.243	0.430
$\dot{r}_{\mathrm{est}}$ ($\mathrm{rad\,s^{-2}}$)	1.278	1.676	2.332
r_{true} ($\mathrm{rad\,s^{-1}}$)	0.151	0.246	0.450
$\dot{r}_{\mathrm{true}}$ ($\mathrm{rad\,s^{-2}}$)	2.092	2.071	2.707

The RMS results in table IV confirm the benefits of ANDI with state-dependent compensation across both rate and acceleration metrics. For estimated yaw rate, the full controller achieves the lowest error, followed by partial, with baseline highest. Estimated yaw rate error is reduced by 66 % from baseline to full ANDI ($0.430\,\mathrm{rad\,s^{-1}}$ to $0.146\,\mathrm{rad\,s^{-1}}$), demonstrating significant improvement from state-dependent compensation in inversion during high angular rate maneuvers. The partial controller's intermediate reduces the error by 43 %

from the baseline ($0.430\,\mathrm{rad\,s^{-1}}$ to $0.243\,\mathrm{rad\,s^{-1}}$), indicates that state-dependent effects in complementary filtering alone already offers substantial improvement over the baseline.

A similar improvement pattern is observed for estimated yaw acceleration. The full ANDI achieves $45\,\%$ reduction compared to baseline ($2.332\,\mathrm{rad\,s^{-2}}$ to $1.278\,\mathrm{rad\,s^{-2}}$), with the partial configuration again showing intermediate improvement with $28\,\%$ error reduction ($2.332\,\mathrm{rad\,s^{-2}}$ to $1.676\,\mathrm{rad\,s^{-2}}$). These consistent improvements across both estimated metrics confirm that explicit state-dependent compensation in inversion yields measurable benefits for tracking performance.

For true acceleration, the partial and full controllers exhibit similar errors ($2.071\,\mathrm{rad\,s^{-2}}$ and $2.092\,\mathrm{rad\,s^{-2}}$ respectively), with no significant difference discernible between these configurations. This similarity is expected given the transient tracking errors that dominates the true yaw acceleration during this aggressive maneuver, obscuring differences in state compensation between these two cases. However, the baseline shows significantly larger true acceleration error ($2.707\,\mathrm{rad\,s^{-2}}$), confirming that complementary filtering with state-dependent effects provides meaningful improvement over the inversion-only baseline.

D. Discussion

The experimental validation confirms the ANDI control implementation's correctness and practical viability. The control law inversion is verified in simulation by perfect tracking of the reference trajectory, while the theoretical equivalence between ANDI without state-dependent compensation and INDI is experimentally demonstrated for the chosen gain parameters. Real-world flight testing validates the cascaded complementary filtering approach and confirms that the designed control architecture translates from simulation to practice.

State-dependent effect compensation produces measurable improvements for platforms where such effects are significant, as demonstrated by the full ANDI outperforming the baseline during aggressive maneuvers. However, the practical relevance of this compensation depends on platform characteristics and operating regime: for MAVs with pronounced state-dependent dynamics or designed for aggressive flight, compensation enhances tracking accuracy; for platforms with negligible such effects or those operated in benign conditions (e.g. quasi-static flight), the baseline ANDI without F_x proves sufficient. This variability across the MAV design space underscores the need for flexible control architectures.

ANDI unifies and extends the classical INDI architecture to address this design space diversity. In the absence of significant state-dependent effects or actuator bandwidth variations, omitting F_x renders ANDI mathematically equivalent to INDI, resulting in identical closed-loop behavior. This architectural flexibility ensures that deploying ANDI as an INDI-equivalent controller carries no performance penalty while retaining the capability to incorporate state-dependent dynamics and actuator heterogeneity when platform characteristics demands it or system knowledge is available.

The experimental validation thus demonstrates the viability and significance of state-dependent compensation on MAVs, but also the broader value of ANDI as a flexible control architecture that integrates actuator dynamics compensation and optional state-dependent effects within a single coherent framework.

VII. Conclusion

This research has implemented and experimentally validated the ANDI framework with explicit state-dependent compensation on the *Cyclone* tail-sitter MAV. The results demonstrate that ANDI is a practical and effective framework for attitude stabilization, generalizing classical INDI by explicitly incorporating actuator dynamics and state-dependent effects while recovering INDI when these terms are omitted.

The key finding is nuanced: state-dependent compensation yields clear benefits during aggressive maneuvers, where yaw-rate tracking errors were substantially reduced, at the cost of increased model knowledge. Regardless, ANDI is the preferred framework due to its generality and flexibility. ANDI without state-dependent compensation is equivalent to INDI and does not require additional modeling effort. Yet ANDI still maintains the capability to incorporate state-dependent effects when the additional model knowledge is available or the anticipated performance gains justify the effort.

These conclusions are necessarily bounded by the experimental scope: all tests were conducted indoors in hover and low-speed flight on a single platform. The observed benefits of state-dependent compensation are conservative and may underestimate its impact in more demanding regimes, such as high-speed forward flight or aggressive transition maneuvers. Furthermore, the aerodynamic model was tailored to the tested envelope, and only attitude stabilization was addressed.

Future work should extend ANDI to guidance and trajectory control with state-dependent body-force compensation, repeat experiments across a wider operational envelope including outdoor high-speed flight, and study transition maneuvers for a more representative maneuver.

References

[1] A. S. Saeed, A. B. Younes, C. Cai, and G. Cai, “A Survey of Hybrid Unmanned Aerial Vehicles,” *Progress in Aerospace Sciences*, Mar. 1, 2018. DOI: 10.1016/j.paerosci.2018.03.007.

[2] H. K. Khalil, “Nonlinear Systems,” in 3rd ed. Prentice Hall, 2002, ch. 6, pp. 207–275.

[3] R. Steffensen, A. Steinert, and F. Holzapfel, “Longitudinal Incremental Reference Model for Fly-By-Wire Control Law using Incremental Non-Linear Dynamic Inversion,” in *AIAA SCITECH 2022 Forum*, Jan. 3, 2022. DOI: 10.2514/6.2022-1230.

[4] D. Surmann, M. Zrenner, and S. Myschik, “Flight Performance Evaluation of a Conceptual Evtol System Using Nonlinear Simulations,” in *AIAA SCITECH 2022 Forum*, Jan. 3, 2022. DOI: 10.2514/6.2022-2220.

[5] A. Steinleitner, V. Frenzel, O. Pfeifle, J. Denzel, and W. Fichter, “Automatic Take-Off and Landing of Tailwheel Aircraft with Incremental Nonlinear Dynamic Inversion,” in *AIAA SCITECH 2022 Forum*, Dec. 29, 2021. DOI: 10.2514/6.2022-1228.

[6] A. C. Gabrys, R. Steffensen, R. d. A. Cordeiro, J. R. Azinheira, A. Moutinho, and F. Holzapfel, "Integration of Phase Plane Flight Envelope Protections in Cascaded Incremental Flight Control," *IFAC-PapersOnLine*, vol. 52, no. 12, pp. 429–435, Jan. 1, 2019. DOI: 10.1016/j.ifacol.2019.11.281.

[7] M. Li, J. Li, Y. Tang, and S. Sun, "An Extended INDI Approach and Application to Pitch Rate Control Laws Design of an Aircraft," in *AIAA Aviation 2021 Forum*, Aug. 2, 2021. DOI: 10.2514/6.2021-3005.

[8] E. J. J. Smeur, Q. Chu, and G. C. H. E. de Croon, "Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 3, pp. 450–461, Mar. 2016. DOI: 10.2514/1.G001490.

[9] R. Steffensen, A. Steinert, Z. Mbikayi, S. Raab, J. Angelov, and F. Holzapfel, "Filter and sensor delay synchronization in incremental flight control laws," *Aerospace Systems*, vol. 6, no. 2, pp. 285–304, Jun. 1, 2023. DOI: 10.1007/s42401-022-00186-2.

[10] R. Steffensen, A. Steinert, and E. J. J. Smeur, "Nonlinear Dynamic Inversion with Actuator Dynamics: An Incremental Control Perspective," *Journal of Guidance, Control, and Dynamics*, pp. 1–9, Nov. 25, 2022. DOI: 10.2514/1.G007079.

[11] T. M. L. De Ponti, E. J. J. Smeur, and B. D. W. Remes, "Unified-Actuator Nonlinear Dynamic Inversion Controller for the Variable Skew Quad Plane," *Journal of Guidance, Control, and Dynamics*, pp. 1–10, Feb. 13, 2025. DOI: 10.2514/1.G008659.

[12] M. Bronz, E. J. Smeur, H. G. de Marina, and G. Hattenberger, "Development of A Fixed-Wing mini UAV with Transitioning Flight Capability," in *35th AIAA Applied Aerodynamics Conference*, Jul. 2, 2017. DOI: 10.2514/6.2017-3739.

[13] G. Hattenberger, M. Bronz, and M. Gorraz, "Using the Paparazzi UAV System for Scientific Research," in *International Micro Air Vehicle Conference and Competition 2014*, Aug. 12, 2014, pp 247. DOI: 10.4233/uuid:b38fbdb7-e6bd-440d-93be-f7dd1457be60.

[14] L. R. Lustosa, F. Defaÿ, and J.-M. Moschetta, "Global Singularity-Free Aerodynamic Model for Algorithmic Flight Control of Tail Sitters," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 2, pp. 303–316, Feb. 2019. DOI: 10.2514/1.G003374.

[15] E. A. Tal and S. Karaman, "Global Incremental Flight Control for Agile Maneuvering of a Tailsitter Flying Wing," *Journal of Guidance, Control, and Dynamics*, vol. 45, no. 12, pp. 2332–2349, Dec. 12, 2022. DOI: 10.2514/1.G006645.

[16] E. J. J. Smeur, M. Bronz, and G. C. H. E. de Croon, "Incremental Control and Guidance of Hybrid Aircraft Applied to a Tailsitter Unmanned Air Vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 2, pp. 274–287, Sep. 17, 2020. DOI: 10.2514/1.G004520.

[17] E. J. J. Smeur, D. C. Höppener, and C. de Wagter, "Prioritized Control Allocation for Quadrotors Subject to Saturation," in *International Micro Air Vehicle Conference and Competition 2017*, H. de Plinval, J.-M. Moschetta, and G. Hattenberger, Eds., Toulouse, France, Sep. 18, 2017, pp. 37–43. http://www.imavs.org/pdf/imav.2017.6 Accessed: Jun. 21, 2025.

[18] R. Van 't Veld, "Incremental Nonlinear Dynamic Inversion Flight Control," Master of Science Thesis, Delft University of Technology, Delft, Sep. 29, 2016. https://repository.tudelft.nl/file/File_72437d3f-ba51-4718-be28-a368de7d02f3 Accessed: Mar. 25, 2025.

[19] J. Solà, "Quaternion Kinematics for the Error-State Kalman Filter," Nov. 3, 2017. DOI: 10.48550/arXiv.1711.02508.

Part III

Additional Results

4 Guidelines for Implementing ANDI

This chapter provides practical guidance for implementing the Actuator Nonlinear Dynamic Inversion (ANDI) controller, presented in Section 3.4, on generic hybrid micro aerial vehicle (Micro Air Vehicle (MAV)) platforms. The controller extends sensor-based Incremental Nonlinear Dynamic Inversion (INDI) to incorporate heterogeneous actuator bandwidths and, when desired, state-dependent dynamics. The following sections outline a structured procedure for implementing ANDI operation across a broad class of Unmanned Aerial Vehicles (UAVs).

This guide presents the high level workflow, and is focused on applicability across hybrid platforms. While developed and validated on a tail-sitter MAV, the approach should remain valid for most other hybrid-UAV architectures. Platform-specific adaptations may be required depending on actuator types, control surface geometries, and available sensor measurements, but the underlying methodology remains consistently applicable.

The procedure assumes access to a controlled flight testing environment that allows safe tethered operation before free-flight testing.

1. **Identify Actuator Dynamics:** Characterize each actuator's bandwidth and saturation limits by fitting a first-order model.
2. **Estimate Control Effectiveness:** Quantify how each actuator contributes to changes in measured outputs to establish the initial control effectiveness matrix.
3. **Choose Filter Cutoff Frequencies:** Select complementary filter cutoff frequencies to balance noise attenuation and dynamic response with minimal phase lag.
4. **Tune Controller Gains:** Adjust the controller gains to achieve desired closed-loop performance and verify stability in hover before advancing to more aggressive operation.
5. **Model Complete System Dynamics:** With stable hover achieved, extend system identification across the wider flight envelope to refine actuator and filter models.
6. **Iterate and Refine:** Continuously refine actuator models, control effectiveness estimates, filter designs, and controller gains using flight data until desired performance is achieved.

This workflow closely follows the implementation of sensor-based INDI, with the primary distinction that ANDI explicitly admits actuator dynamics into the control loop and, in advanced stages, can identify state-dependent system behavior.

4.1. Identify Actuator Dynamics

The first step is to model the dynamics of each actuator, capturing its bandwidth and saturation behavior. For hybrid MAVs, the primary actuators are typically motors and servos. Approximate the actuator dynamics as first-order systems with bandwidth ε.

4.1.1. Experimental Characterization

Characterize each actuator with the vehicle on the ground. Apply step commands to the actuator input and record the response at a sufficiently high sampling rate. From the recorded data, fit a first-order model to the measured output. Estimation can be done using least-squares fitting or frequency-domain analysis of the step response. Take into account any possible delays in the system and the effects of saturation. These can negatively impact the accuracy of the model fitting procedures when neglected.

4.1.2. Saturations and Rate Limits

Identify the actuator saturation limits u_{min} and u_{max}. When available, also measure or estimate rate limits $\dot{u}_{\text{max}}$ and $\dot{u}_{\text{min}}$ from logged data or manufacturer specifications. While these limits are not critical for an initial implementation, neglecting rate saturation may yield commands exceeding actuator physical capabilities. To initially disable rate limiting, choose a high rate limit value well above expected command rates during hover.

4.2. Estimate Control Effectiveness

The next step is to determine the control effectiveness matrix F_u during hover, which describes how actuator inputs effect the measured outputs. Accurate estimation of F_u is essential for correct inversion in the ANDI control law and for designing the complementary filter.

4.2.1. Methods for Estimation

Several approaches can be applied depending on available knowledge and tools, and the vehicle type:

- **First-Principles Modeling:** Compute F_u analytically from the known geometry and propulsion configuration (e.g., arm lengths, thrust coefficients).
- **Static Testing:** Measure produced forces and moments under known actuator inputs using a load cell or thrust stand.
- **Flight Data Identification:** Estimate F_u via linear least-squares fitting from logged flight data.

For initial implementation, a constant control effectiveness matrix is usually sufficient. Once stable hover flight is confirmed, scheduling F_u with operating conditions (e.g., throttle level, airspeed, attitude, etc.) can be introduced to capture the wider flight envelope.

4.2.2. Robustness to Model Errors

The incremental nature of ANDI provides inherent robustness to modeling errors. Some errors in the initial control effectiveness matrix can thus be tolerated. However, large errors in F_u can lead to instability or poor performance. When in doubt, it is safer to slightly overestimate control effectiveness than to underestimate it; overestimation relaxes the control law, whereas underestimation can cause unstable oscillations.

4.3. Choose Filter Cutoff Frequencies

Complementary filters fuse the noisy measurements (e.g., accelerometer data) with predictions from the On-Board Model (OBM) (based on F_u). Minimal filtering is important to achieve good disturbance rejection, but noise results in excessive wear on actuators and degraded performance. Choose the initial cutoff frequencies conservatively and adjust iteratively based on observed noise present in the filtered estimates. Adjust the cutoff frequencies to perform the minimal filtering while still attenuate all high-frequency noise.

4.4. Tune Controller Gains

With actuator models, control effectiveness, and filters established, the next step is to tune the ANDI controller gains.

4.4.1. Controller Parameters

This ANDI implementation employs a reference model and error controller for each output, resulting in six key tuning parameters per attitude axis and two for specific thrust control. The function of these parameters is as follows:

1. **Error Controller Parameters:** Define how the system responds to external disturbances.
2. **Reference Model Parameters:** Define desired tracking dynamics.

For attitude control, each group is characterized by a pseudo-actuator bandwidth ε and a second-order system defined by natural frequency ω_n and damping ratio ζ. The specific thrust controller only requires the pseudo-actuator bandwidth ε_τ.

The detailed gain structure is presented in Section 3.4.

Tables 4.1 and 4.2 summarize all the parameters and recommended relationships between the parameters for simplified tuning.

Table 4.1: Attitude control gain parameters and recommended relationships for simplified tuning.

Attitude Control Parameter	Error Controller	Reference Model
Natural Frequency	ω_{n_e}	$\omega_{n_r} = 0.8\omega_{n_e}$
Damping Ratio	ζ_e	$\zeta_r = \zeta_e$
Pseudo-actuator Bandwidth	$\varepsilon =$ slowest actuator driving each output	

Table 4.2: Specific thrust control gain parameters and recommended relationships for simplified tuning.

Specific Thrust Control Parameter	Error Controller	Reference Model
Pseudo-actuator Bandwidth	$\varepsilon_\tau =$ slowest actuator driving each output	

4.4.2. Simplified Initial Tuning

For first flights, the objective is achieving stable hover with conservative dynamics. The following guidelines simplify the tuning problem from six independent parameters to two per attitude axis:

- Set the pseudo-actuator bandwidth ε equal to the slowest actuator driving each axis.
- Choose the reference model natural frequency as $\omega_{n_r} = 0.8\omega_{n_e}$.
- Use identical damping ratios for both models, $\zeta_r = \zeta_e$.

This leaves only ω_{n_e} and ζ_e to tune per axis. Start with low values for ω_{n_e} to ensure stability, then gradually increase to improve responsiveness.

4.4.3. Further Tuning and Refinement

After stable hover is achieved, the initial relations between the parameters can be relaxed to exploit full actuator capabilities. Adjust the reference model parameters to achieve desired tracking performance, and modify the error controller parameters to optimize disturbance rejection. Iteratively refine the gains based on observed performance in flight tests.

4.5. Model Complete System Dynamics

Once stable hover is achieved, the static control effectiveness model can be extended to represent the full flight envelope. This step involves identifying the complete ϕ-theory model, capturing state-dependent variations in control effectiveness and actuator dynamics. Section 3.3 describes the model structure for the *Cyclone* tail-sitter platform.

4.5.1. Data Collection

Conduct flight tests beginning in hover and progressively expand the envelope by varying throttle, increasing airspeed, and performing aggressive maneuvers. Record actuator commands, actuator state feedback, and body accelerations and rates at high sampling rates. Include both hover and forward-flight segments to capture representative operating conditions.

4.5.2. Model Identification

Estimate the coefficients of the extended model using least-squares fitting or similar system identification methods.

Verify that the identified model, evaluated at hover conditions, is similar to the previously established static control effectiveness matrix. Large discrepancies may indicate issues with the identification process or data quality and might prevent stable hover.

4.6. Iterate and Refine

With the complete system model established, continue refining all components of the ANDI implementation. This final phase focuses on exploiting the full capabilities of the controller and ensuring robust operation across the intended flight envelope.

4.6.1. Data-Driven Refinement Process

Valuable data is now available from flight tests over the complete flight envelope that previously was not accessible.

This data can now be used to refine:

- **Actuator Models:** Re-identify actuator dynamics under load conditions.
- **Filter Designs:** Adjust cutoff frequencies based on observed noise characteristics in flight.
- **Gain Tuning:** Further optimize controller parameters to enhance performance metrics.
- **Vehicle Model:** Update the coefficients of the ϕ-theory model to better capture state-dependent dynamics. The accuracy of the model directly impacts the ability to linearize the system and the accuracy of the filtered measurement estimates.

5
ANDI Guidance Controller

This chapter presents the design of an ANDI guidance controller for the *Cyclone* tail-sitter MAV.

A guidance controller has been designed during the initial simulation phase of this thesis project to provide position and heading control of the MAV in cascade with the inner-loop attitude and specific thrust ANDI stabilization controller. This guidance controller has not been further developed or experimentally validated within the scope of this thesis. Nevertheless, the controller is documented to establish a foundation for future experimental work.

For hybrid MAV platforms, a guidance controller must account for aerodynamic forces such as lift and drag, which dominate the vehicle dynamics in high speed flight. Existing work by Tal and Karaman [46] presents a guidance controller for tail-sitter MAVs capable of accounting for these forces. However, this approach relies on sequentially solving for Euler angles, which introduce practical limitations.

Specifically, the Euler angle approach requires complicated trigonometric relations and is inherently vehicle-specific, necessitating complete rederivation of these relations for different platform geometries. Furthermore, it relies on restrictive assumptions to ensure sequential solvability of the Euler angles; these assumptions do not generally hold for arbitrary MAV configurations.

The proposed guidance controller provides a generic and adaptable outer-loop control solution by leveraging matrix inversion within the allocation step of ANDI. Rather than solving sequentially through trigonometric relations, the controller directly solves for the required angular rates and specific thrust rates that achieve the desired translational jerk. This approach naturally accounts for all modeled body forces without requiring vehicle-specific derivations or restrictive assumptions.

5.1. Controller Derivation

The ANDI guidance law generates angular rate commands $\boldsymbol{\omega}_c$ and specific thrust rate commands $\dot{\tau}_c$ to ensure tracking of a reference position trajectory $\mathbf{p}_r(t)$ and heading $\psi_r(t)$. The derivation assumes that:

(i) $\mathbf{p}_r(t)$ is at least C^2 continuous;

(ii) $\psi_r(t)$ is at least C^0 continuous;

(iii) $\boldsymbol{\omega}_c$ and $\dot{\tau}_c$ are tracked instantaneously and exactly, modeled as zero-order dynamics.

5.1.1. System Representation

The outer-loop state and input vectors are defined in Equations (5.1) and (5.2). The state vector can be extended to include additional states as needed.

$$\mathbf{x} = \begin{bmatrix} v_x & v_y & v_z & \delta_l & \delta_r \end{bmatrix}^\top \tag{5.1}$$

$$\mathbf{u} = \begin{bmatrix} q_w & q_x & q_y & q_z & \tau \end{bmatrix}^\top \tag{5.2}$$

The vector $\mathbf{u}$ comprises the unit quaternion attitude $\mathbf{q}$ and total specific thrust τ, which are considered pseudo-actuators. The state vector $\mathbf{x}$ includes the body-frame linear velocities $\mathbf{v}_b$ and the left and right elevon deflections δ_l and δ_r.

The elevon deflections are included in the state vector to capture their influence on the thrust vector. These are treated as part of the state since they are not directly controlled by the guidance controller, which only commands the attitude and specific thrust.

The outer-loop outputs are the linear acceleration $\ddot{\mathbf{p}}$ and heading ψ. The heading is defined using Equation (5.3), as the angle between the projection of the body y-axis and the East direction. This avoids the singularity when the pitch angle approaches $90°$, which is a common attitude for tail-sitter MAVs during forward flight. Instead, the singularity occurs when the vehicle is on its side ($90°$ roll angle).

$$\psi(\mathbf{u}) = \text{atan2}\big(-2\left(q_x q_y + q_w q_z\right),\ 1 - 2\left(q_x^2 + q_z^2\right)\big) \tag{5.3}$$

5.1.2. Translational Dynamics

The specific force in the inertial frame is obtained by rotating the body-frame specific force $\mathbf{f}_b(\mathbf{x}, \mathbf{u})$ to inertial coordinates. The body-frame force is derived from the OBM presented in Section 3.3, and the coordinate transformation is expressed in Equation (5.4).

$$\mathbf{f}_i(\mathbf{x}, \mathbf{u}) = {}^{i}R_b(\mathbf{u}) \cdot \mathbf{f}_b(\mathbf{x}, \mathbf{u}) \tag{5.4}$$

The rotation matrix ${}^{i}R_b(\mathbf{u})$ is defined in Equation (5.5) from the current attitude quaternion in $\mathbf{u}$.

$${}^{i}R_b(\mathbf{u}) = \begin{bmatrix} 1 - 2(q_y^2 + q_z^2) & 2(q_x q_y - q_w q_z) & 2(q_x q_z + q_w q_y) \\ 2(q_x q_y + q_w q_z) & 1 - 2(q_x^2 + q_z^2) & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z - q_w q_y) & 2(q_y q_z + q_w q_x) & 1 - 2(q_x^2 + q_y^2) \end{bmatrix} \tag{5.5}$$

5.1.3. Input-Output Linearization

The input-output relation is given by Equation (5.6).

$$\begin{bmatrix} \ddot{\mathbf{p}} \\ \psi \end{bmatrix} = \begin{bmatrix} \mathbf{f}_i(\mathbf{x}, \mathbf{u}) \\ \psi(\mathbf{u}) \end{bmatrix} \tag{5.6}$$

Differentiating Equation (5.6) yields Equation (5.7).

$$\begin{bmatrix} \dddot{\mathbf{p}} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial \mathbf{x}} \mathbf{f}_i(\mathbf{x}, \mathbf{u}) \\ \frac{\partial}{\partial \mathbf{x}} \psi(\mathbf{u}) \end{bmatrix} \dot{\mathbf{x}} + \begin{bmatrix} \frac{\partial}{\partial \mathbf{u}} \mathbf{f}_i(\mathbf{x}, \mathbf{u}) \\ \frac{\partial}{\partial \mathbf{u}} \psi(\mathbf{u}) \end{bmatrix} \dot{\mathbf{u}} \tag{5.7}$$

The quaternion derivative relation is given in Equation (5.8).

$$\begin{bmatrix} \dot{q}_w \\ \dot{q}_x \\ \dot{q}_y \\ \dot{q}_z \end{bmatrix} = \frac{1}{2} Q_L(\mathbf{u}) \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{5.8}$$

with the quaternion left multiplication matrix $Q_L(\mathbf{u})$ defined in Equation (5.9).

$$Q_L(\mathbf{u}) = \begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & -q_z & q_y \\ q_z & q_w & -q_x \\ -q_y & q_x & q_w \end{bmatrix} \tag{5.9}$$

Thus, Equation (5.10) applies:

$$\dot{\mathbf{u}} = \begin{bmatrix} \frac{1}{2}Q_L(\mathbf{u}) & 0 \\ 0 & 1 \end{bmatrix} \dot{\mathbf{u}}' \tag{5.10}$$

where $\dot{\mathbf{u}}' = [\omega_x, \omega_y, \omega_z, \dot{\tau}]^\top$.

Substituting into Equation (5.7) yields Equation (5.11):

$$\begin{bmatrix} \ddot{\mathbf{p}} \\ \dot{\psi} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\partial}{\partial \mathbf{x}} \mathbf{f}_i(\mathbf{x}, \mathbf{u}) \\ \frac{\partial}{\partial \mathbf{x}} \boldsymbol{\psi}(\mathbf{u}) \end{bmatrix}}_{F_x} \dot{\mathbf{x}} + \underbrace{\begin{bmatrix} \frac{\partial}{\partial \mathbf{u}} \mathbf{f}_i(\mathbf{x}, \mathbf{u}) \\ \frac{\partial}{\partial \mathbf{u}} \boldsymbol{\psi}(\mathbf{u}) \end{bmatrix} \begin{bmatrix} \frac{1}{2}Q_L(\mathbf{u}) & 0 \\ 0 & 1 \end{bmatrix}}_{F_u} \dot{\mathbf{u}}' \tag{5.11}$$

The control input $\dot{\mathbf{u}}'$ is obtained using the right pseudo-inverse as shown in Equation (5.12). This step solves for the required angular rate and specific thrust rate commands needed to achieve the desired pseudo-control $\boldsymbol{\nu}$, while accounting for all modeled body forces through $f_i(\mathbf{x}, \mathbf{u})$. This eliminates the need for vehicle-specific derivations or problematic sequential solvability assumptions. Note that this is only valid in a small convergence region around the current state due to the local linearization; the commanded references should be smooth and feasible to ensure proper inversion.

$$\dot{\mathbf{u}}'_c = {F_u}^\dagger \left(\boldsymbol{\nu} - F_x \dot{\mathbf{x}} \right) \tag{5.12}$$

5.1.4. Linear Error Controller

The pseudo-control vector $\boldsymbol{\nu}$ is generated by the linear controller defined in Equation (5.13).

$$\boldsymbol{\nu} = \begin{bmatrix} \dddot{\mathbf{p}}_r + K_3(\ddot{\mathbf{p}}_r - \ddot{\mathbf{p}}) + K_2(\dot{\mathbf{p}}_r - \dot{\mathbf{p}}) + K_1(\mathbf{p}_r - \mathbf{p}) \\ \dot{\psi}_r + k_\psi(\psi_r - \psi) \end{bmatrix} \tag{5.13}$$

The gains $K_3, K_2, K_1 \in \mathbb{R}^{3\times3}$ are diagonal, positive-definite, and $k_\psi > 0$ is the heading gain. The position error dynamics follow Equation (5.14):

$$\frac{E_{p,i}(s)}{U_{p,i}(s)} = \frac{s^3}{s^3 + k_{a,i}s^2 + k_{v,i}s + k_{p,i}} \tag{5.14}$$

and the heading error satisfies Equation (5.15):

$$\frac{E_\psi(s)}{U_\psi(s)} = \frac{s}{s + k_\psi} \tag{5.15}$$

The gains are selected through trial and error to achieve satisfactory tracking performance. The final gain values used in simulation are given in Equation (5.16). The gains for x, y, and z position control are identical.

$$k_1 = 5.0 \quad k_2 = 16.0 \quad k_3 = 5.0 \quad k_\psi = 0.5 \tag{5.16}$$

5.2. Results

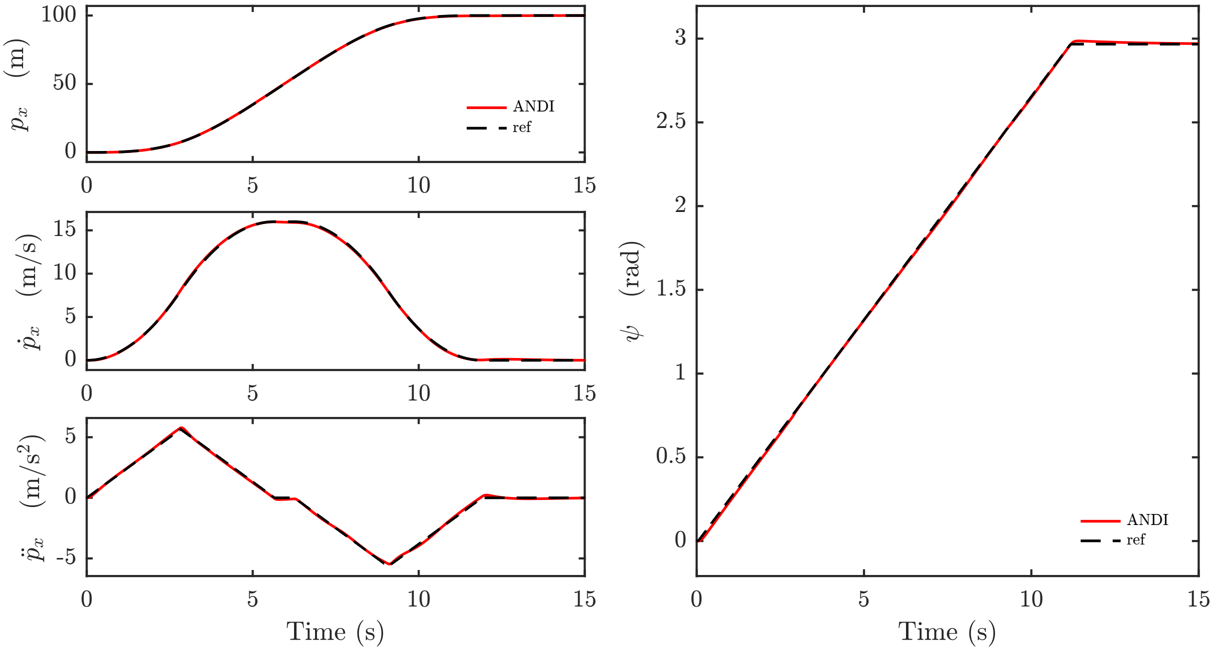
The guidance controller is evaluated in cascade with the full ANDI inner-loop controller. A reference trajectory is an S-curve velocity profile that prescribes a 100 m forward displacement with simultaneous $170°$ heading change, respecting the velocity, acceleration, and jerk limits in Table 5.1 [47]. The position and heading tracking performance are presented in Figure 5.1, with the corresponding actuator commands shown in Figure 5.2.

This reference trajectory is intentionally demanding, requiring the MAV to transition from hover to high-speed forward flight and back to hover while executing a substantial heading change. The resulting maneuver induces significant aerodynamic forces and atypical flight attitudes, including knife-edge flight. This challenging scenario directly evaluates the controller's ability to maintain precise position and heading tracking while accounting for aerodynamic effects across the entire flight envelope.

The guidance controller does not achieve perfect tracking of the reference trajectory in Figure 5.1a, even under ideal conditions. The root cause is a fundamental mismatch between the control design assumptions and the actual system behavior. The derivation assumes that angular rates and specific thrust are tracked instantaneously and exactly by the inner-loop controller (zero-order dynamics assumption). In practice, the inner-loop controller behaves closer to a second-order system with limited bandwidth and saturation, introducing lag and tracking errors.

Table 5.1: Reference trajectory limits for S-curve profile generation.

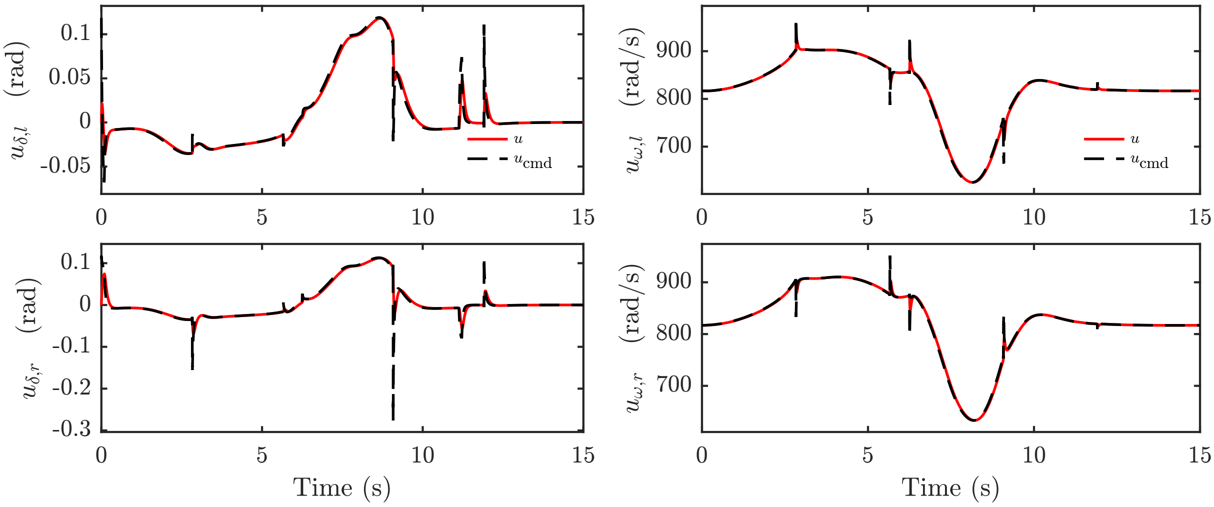
Parameter	Value
Max. velocity	$16\,\mathrm{m\,s^{-1}}$
Max. acceleration	$5\,\mathrm{m\,s^{-2}}$
Max. jerk	$2\,\mathrm{m\,s^{-3}}$



(a) Position tracking

(b) Heading tracking

Figure 5.1: Tracking performance of the guidance controller under ideal simulation conditions (no model uncertainty, disturbances, or measurement noise). (a) Time evolution of position, velocity, and acceleration. (b) Time evolution of heading and heading rate.



(a) Left and right elevon commands

(b) Left and right motor commands

Figure 5.2: Actuator commands when tracking the reference trajectory under ideal simulation conditions (no model uncertainty, disturbances, or measurement noise). (a) Time evolution of left and right elevon commands. (b) Time evolution of left and right motor commands.

The consequences of this simplifying assumption can clearly be observed in the actuator commands in Figure 5.2. The actuator commands are discontinuous at points where the reference jerk changes abruptly, which occurs due to the piecewise nature of the S-curve trajectory generation. The inner-loop runs into actuator saturation at these points, causing temporary loss of tracking performance.

These errors can partially be mitigated by making the reference trajectory feasible for the real system by accounting for the inner-loop bandwidth limitations during reference generation. Making the reference trajectory fifth-order continuous would limit crackle (fifth derivative of position) and improve performance. This is left for future work.

For heading control in Figure 5.1b, the effect is more pronounced. The heading controller operates at first-order (heading rate command), while the inner-loop controller tracks angular rates with second-order dynamics. This additional mismatch causes larger heading errors than position errors. Extending the control law to operate on heading acceleration or jerk level would better match the actual inner-loop dynamics and should significantly improve heading tracking performance. This is also left for future work.

The overall tracking performance of the ANDI guidance controller is excellent, despite the simplifying assumptions. The controller can successfully track both position and heading references, while accounting for aerodynamic forces without vehicle-specific derivations or additional restrictive assumptions. This demonstrates the potential of the ANDI approach for outer-loop guidance control of hybrid MAVs.

5.3. Conclusion

This chapter presented the design of a novel quaternion-based ANDI guidance controller for position and heading control of a tail-sitter MAV. The controller leverages input-output linearization to directly compute the required angular rate and specific thrust rate commands needed to track a desired trajectory, naturally accounting for aerodynamic forces without vehicle-specific derivations.

The controller was simulated under ideal conditions and demonstrated good tracking performance for both position and heading. However, due to the simplifying assumption of zero-order inner-loop dynamics, perfect reference tracking was not achieved.

Future work could focus on: (i) extending the heading controller to operate on heading acceleration or jerk commands to better match the actual second-order inner-loop dynamics, (ii) refining the generation of feasible reference trajectories that account for inner-loop bandwidth and saturation, (iii) experimentally validating the guidance controller on the *Cyclone* tail-sitter MAV, and (iv) exploring a unified control architecture that combines guidance and stability control into a single ANDI allocation step for potentially improved performance through tighter coupling between position and attitude dynamics.

Part IV

Closure

6 Conclusion

6.1. Closing Remarks

This research successfully implemented and experimentally validated a full Actuator Nonlinear Dynamic Inversion (ANDI) stabilization controller with explicit state-dependent compensation, complementary filtering, and reference model, on the *Cyclone* tail-sitter Micro Air Vehicle (MAV). The results demonstrate that ANDI is a practically viable, and an effective control strategy for hybrid MAVs, and quantifies the performance benefits of compensating for state-dependent dynamics compared to traditional Incremental Nonlinear Dynamic Inversion (INDI) through real-world flight testing.

The key contribution of this work is the successful implementation and experimental validation of a full ANDI stabilization controller with state-dependent dynamics compensation on the *Cyclone* tail-sitter MAV, which confirms its feasibility as a generic and adaptable control approach for hybrid MAVs while quantifying the performance improvements over INDI and ANDI without state-dependent compensation in reference tracking during aggressive maneuvers with high angular rates and accelerations.

Additionally, the conditions for equivalence between ANDI and INDI closed-loop dynamics when applied to a system with first order actuator dynamics were derived, providing theoretical insight into the relationship between these two control strategies, leading to the observation that ANDI is a more generic version of INDI that reduces to INDI under certain conditions.

Finally, additional results presenting practical implementation guidelines and a novel quaternion-based guidance controller for deploying ANDI on generic hybrid MAVs were developed. The implementation guidelines facilitate its adoption in real-world applications as a more generic alternative to INDI. The guidance controller leverages input-output linearization and matrix inversion to directly compute required angular rate and specific thrust rate commands without vehicle-specific derivations, with simulation results demonstrating good tracking performance for both position and heading while naturally accounting for aerodynamic forces.

6.2. Research Questions

The three research questions originally posed in Chapter 1 are addressed below, with each answer drawing from the work presented in the preceding chapters.

Research Question 1

How does full ANDI with state-dependent dynamics compensation address the limitations of sensor-based INDI for control of a hybrid MAV?

This research question is primarily addressed through the literature study presented in Chapter 2. The literature review identified and characterized the limitations of sensor-based INDI when applied to hybrid MAVs, and subsequently how the ANDI framework addresses these limitations.

The literature review in Chapter 2 concluded that sensor-based INDI has the following two limitations when applied to hybrid MAVs: First, it does not account for heterogeneous actuator dynamics, instead assuming that all actuators have identical, instantaneous response characteristics. This assumption degrades tracking performance and can lead to instability when actuators have significantly different bandwidths. Second, INDI neglects state-dependent dynamics in its formulation, treating all aerodynamic and unmodeled effects as disturbances compensated through feedback alone. When state-dependent effects such as aerodynamic forces and moments vary significantly with the vehicle state, for example during aggressive maneuvers with high angular rates and acceleration, compensating for these effects directly in the control law can improve tracking performance. For hybrid MAVs with significantly heterogeneous actuators or significant state-dependent dynamics, these limitations render INDI unsuitable.

The full ANDI control law addresses both of these limitations. First, it inverts heterogeneous first-order actuator dynamics in addition to the vehicle dynamics, addressing heterogeneous and slow actuator bandwidths. Second, it compensates explicitly for state-dependent dynamics in the control law. This explicit treatment of state-dependent effects improves tracking performance during aggressive maneuvers compared to relying on sensor feedback alone.

Research Question 2

How does explicit compensation of state-dependent dynamics affect reference tracking and disturbance rejection performance when deployed on a tail-sitter MAV in real-world flight conditions?

This research question is primarily answered through the experimental validation presented in Section 3.6. Three different ANDI implementations with varying levels of state-dependent dynamics compensation were deployed on the *Cyclone* tail-sitter MAV, and their performance was evaluated through flight experiments.

Three implementation variants were compared. The full ANDI variant incorporated the full state-dependent effects in both the control law and the complementary filter. The partial ANDI variant included state-dependent effects in the complementary filter only, but not in the control law. The third variant served as a baseline, equivalent to sensor-based INDI, with state-dependent dynamics neglected entirely in both the control law and complementary filter.

The experimental results demonstrate that explicit state-dependent compensation significantly improves reference tracking performance during aggressive maneuvers. Performance gains are most pronounced when the system operates at high angular rates and accelerations, where aerodynamic effects are substantial. For an aggressive heading change maneuver, full ANDI with state-dependent compensation achieved a $66\,\%$ reduction in yaw rate tracking error compared to the baseline implementation. The partial ANDI variant, incorporating state-dependent compensation in the complementary filter only, achieved a $43\,\%$ reduction in yaw rate tracking error. These results quantify the progressive performance benefit as state-dependent knowledge is incorporated: complementary filter compensation alone provides significant improvement over baseline, while direct compensation in the control law further enhances tracking performance.

Regarding disturbance rejection, state-dependent knowledge does not improve performance against external disturbances. By definition, disturbances are unmodeled, and only explicit knowledge of the plant model can be compensated in the feed-forward path. The implicit disturbance rejection provided by sensor feedback remains unchanged regardless of whether state-dependent terms are explicitly modeled or implicitly handled through feedback.

Research Question 3

How can ANDI with state-dependent dynamics compensation be systematically implemented on generic MAVs, considering practical constraints such a limited model knowledge, and available sensor measurements?

This research question is addressed through the implementation guidelines presented in Chapter 4 and Section 3.5. A key insight is that ANDI is a more generic control framework than INDI, applicable to all cases where INDI can be applied, and to additional cases where actuators are heterogeneous and state-dependent effects are significant.

Regarding model knowledge constraints, ANDI can be implemented incrementally. In its simplest form, state-dependent compensation can be disabled, making ANDI equivalent to sensor-based INDI and requiring no additional model knowledge beyond what INDI requires. This variant can be deployed immediately on a new platform without detailed system identification. To achieve the full performance benefits of explicit state-dependent compensation, additional model knowledge of the vehicle's aerodynamic characteristics is required. However, this knowledge is not required to be available prior to deployment; instead, it can be obtained through system identification performed during initial test flights with the baseline ANDI implementation (without state-dependent compensation).

Both control strategies rely on sensor feedback to estimate the system outputs necessary for the incremental control law. For most MAV applications, this requires angular acceleration estimates (from gyroscope measurements) and linear acceleration estimates (from accelerometer measurements). ANDI additionally requires state-derivative estimates for the state-dependent dynamics compensation. For most MAVs, the state derivatives required are angular accelerations and translational accelerations, which are already needed for INDI. Therefore, ANDI can be implemented on generic MAVs without additional sensors beyond what is already required for INDI.

ANDI can be systematically implemented on a generic MAV by following the step-by-step implementation guidelines presented in Chapter 4. These guidelines outline the procedures for identifying the necessary model, designing the complementary filter, and tuning the controller gains.

7 Recommendations

This chapter provides recommendations for the future continuation of this research project.

Recommendation 1: Support Higher Order Actuator Effects in Control Effectiveness

Future work should extend the controller structure to account for higher order actuator effects such as moments due to rotor inertia. The current implementation does not handle these effects, which can be significant for some MAV configurations (e.g. yaw control in quadrotors). Incorporating these higher order effects into the control effectiveness model would make the controller more generic and applicable to a wider range of MAVs.

Recommendation 2: Implement Motor Angular Rate Control

Future work should implement a dedicated motor speed controller to accurately track desired motor speeds. The current implementation relies on an inaccurate linear mapping from motor command to motor speed, since most MAV platforms, including the *Cyclone* used in this research, do not allow direct motor speed control. This improvement would enhance the overall performance and accuracy of the ANDI controller.

Recommendation 3: Direct Actuator Rate Control

Future work should explore the advantages that direct actuator rate control could bring to the ANDI framework. ANDI computes actuator rate commands $\dot{\mathbf{u}}_c$, which are then related to $\mathbf{u}_c$ with the actuator bandwidth. By directly controlling actuator rates instead, the dependence on accurate actuator feedback could be reduced, and knowledge of the actuator bandwidth would no longer be required, simplifying the control implementation.

Recommendation 4: Continue ANDI Guidance Controller Development

Further development of a guidance controller based on the ANDI framework is needed to further validate the effects of state-dependent dynamics compensation on trajectory tracking performance. This would involve continuing the design of the ANDI guidance controller, implementing it on a real MAV, and testing it in real-world flight scenarios.

Recommendation 5: Extend Flight Testing to Cover Full Flight Envelope

Future work should expand the experimental validation of the ANDI control law to the full flight envelope, including high-speed forward flight and transition maneuvers. All tests were conducted indoors, limiting the flight conditions to hover and low speed flight. By extending the flight testing to cover the complete operational envelope, the robustness and effectiveness of the proposed control approach can be thoroughly validated across diverse flight regimes.

References

[1] A. S. Saeed et al. “A Survey of Hybrid Unmanned Aerial Vehicles”. In: *Progress in Aerospace Sciences* (Mar. 1, 2018). DOI: 10.1016/j.paerosci.2018.03.007.

[2] H. K. Khalil. “Nonlinear Systems”. In: 3rd ed. Prentice Hall, 2002. Chap. 6, pp. 207–275.

[3] R. Steffensen et al. “Longitudinal Incremental Reference Model for Fly-By-Wire Control Law Using Incremental Non-Linear Dynamic Inversion”. In: *AIAA SCITECH 2022 Forum*. Jan. 3, 2022. DOI: 10.2514/6.2022-1230.

[4] D. Surmann et al. “Flight Performance Evaluation of a Conceptual Evtol System Using Nonlinear Simulations”. In: *AIAA SCITECH 2022 Forum*. Jan. 3, 2022. DOI: 10.2514/6.2022-2220.

[5] A. Steinleitner et al. “Automatic Take-Off and Landing of Tailwheel Aircraft with Incremental Nonlinear Dynamic Inversion”. In: *AIAA SCITECH 2022 Forum*. Dec. 29, 2021. DOI: 10.2514/6.2022-1228.

[6] A. C. Gabrys et al. “Integration of Phase Plane Flight Envelope Protections in Cascaded Incremental Flight Control”. In: *IFAC-PapersOnLine* 52.12 (Jan. 1, 2019), pp. 429–435. DOI: 10.1016/j.ifacol.2019.11.281.

[7] M. Li et al. “An Extended INDI Approach and Application to Pitch Rate Control Laws Design of an Aircraft”. In: *AIAA Aviation 2021 Forum*. Aug. 2, 2021. DOI: 10.2514/6.2021-3005.

[8] E. J. J. Smeur et al. “Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles”. In: *Journal of Guidance, Control, and Dynamics* 39.3 (Mar. 2016), pp. 450–461. DOI: 10.2514/1.G001490.

[9] X. Wang et al. “Flexible Aircraft Gust Load Alleviation with Incremental Nonlinear Dynamic Inversion”. In: *Journal of Guidance, Control, and Dynamics* (July 7, 2019). DOI: 10.2514/1.G003980.

[10] R. Steffensen et al. “Filter and Sensor Delay Synchronization in Incremental Flight Control Laws”. In: *Aerospace Systems* 6.2 (June 1, 2023), pp. 285–304. DOI: 10.1007/s42401-022-00186-2.

[11] R. Steffensen et al. “Nonlinear Dynamic Inversion with Actuator Dynamics: An Incremental Control Perspective”. In: *Journal of Guidance, Control, and Dynamics* (Nov. 25, 2022), pp. 1–9. DOI: 10.2514/1.G007079.

[12] T. M. L. De Ponti et al. “Unified-Actuator Nonlinear Dynamic Inversion Controller for the Variable Skew Quad Plane”. In: *Journal of Guidance, Control, and Dynamics* (Feb. 13, 2025), pp. 1–10. DOI: 10.2514/1.G008659.

[13] European Union Aviation Safety Agency. *Understanding European Drone Regulations and the Aviation Regulatory System*. European Union Aviation Safety Agency. July 10, 2024. https://easa.europa.eu/en/domains/drones-air-mobility/drones-air-mobility-landscape/Understanding-European-Drone-Regulations-and-the-Aviation-Regulatory-System (visited on 05/03/2025).

[14] M. Niyazi et al. “Application of Emerging Digital Technologies in Disaster Relief Operations: A Systematic Review”. In: *Archives of Computational Methods in Engineering* 30.3 (Apr. 1, 2023), pp. 1579–1599. DOI: 10.1007/s11831-022-09835-3.

[15] S. Asadzadeh et al. “UAV-based Remote Sensing for the Petroleum Industry and Environmental Monitoring: State-of-the-art and Perspectives”. In: *Journal of Petroleum Science and Engineering* 208 (Jan. 1, 2022), p. 109633. DOI: 10.1016/j.petrol.2021.109633.

[16] P. Radoglou-Grammatikis et al. “A Compilation of UAV Applications for Precision Agriculture”. In: *Computer Networks* 172 (May 8, 2020), p. 107148. DOI: 10.1016/j.comnet.2020.107148.

[17] Y. Ham et al. “Visual Monitoring of Civil Infrastructure Systems via Camera-Equipped Unmanned Aerial Vehicles (UAVs): A Review of Related Works”. In: *Visualization in Engineering* 4.1 (Jan. 6, 2016), p. 1. DOI: 10.1186/s40327-015-0029-z.

[18] G. Vangu et al. “Use of Modern Technologies Such as UAV, Photogrammetry, 3D Modeling, and GIS for Urban Planning”. In: *Journal of Applied Engineering Sciences* 15.1 (May 1, 2025), pp. 165–174. DOI: 10.2478/jaes-2025-0020.

[19] E. Ganić et al. “Unmanned Aircraft for Emergency Deliveries Between Hospitals in Madrid: Estimating Time Savings and Predictability”. In: *Drones* 9.11 (Nov. 2025), p. 728. DOI: 10.3390/drones9110728.

[20] L. Criollo et al. “Classification, Military Applications, and Opportunities of Unmanned Aerial Vehicles”. In: *Aviation* 28.2 (Aug. 19, 2024), pp. 115–127. DOI: 10.3846/aviation.2024.21672.

[21] Q. A. Abdullah. *Classification of the Unmanned Aerial Systems*. The Pennsylvania State University. 2023. https://e-education.psu.edu/geog892/node/5 (visited on 05/03/2025).

[22] Deltaquad. *DeltaQuad Evo | Battlefield-proven Fixed-Wing VTOL UAVs*. Deltaquad. 2024. https://www.deltaquad.com/ (visited on 01/24/2026).

[23] Quantum Systems. *Trinity Pro - Quantum Systems*. Quantum Systems. July 17, 2025. https://quantum-systems.com/trinity-pro/ (visited on 01/24/2026).

[24] Wingtra. *WingtraCloud Boosts Revegetation Project Productivity by 70 Percent*. Wingtra. Jan. 17, 2025. https://wingtra.com/case_studies/wingtracloud-boosts-revegetation-project-productivity/ (visited on 05/06/2025).

[25] M. Bronz et al. “Development of A Fixed-Wing Mini UAV with Transitioning Flight Capability”. In: *35th AIAA Applied Aerodynamics Conference*. July 2, 2017. DOI: 10.2514/6.2017-3739.

[26] L. R. Lustosa et al. “Global Singularity-Free Aerodynamic Model for Algorithmic Flight Control of Tail Sitters”. In: *Journal of Guidance, Control, and Dynamics* 42.2 (Feb. 2019), pp. 303–316. DOI: 10.2514/1.G003374.

[27] C. Chen et al. “Control and Flight Test of a Tilt-Rotor Unmanned Aerial Vehicle”. In: *International Journal of Advanced Robotic Systems* 14.1 (Jan. 1, 2017), p. 1729881416678141. DOI: 10.1177/1729881416678141.

[28] E. J. J. Smeur et al. “Incremental Control and Guidance of Hybrid Aircraft Applied to a Tailsitter Unmanned Air Vehicle”. In: *Journal of Guidance, Control, and Dynamics* 43.2 (Sept. 17, 2020), pp. 274–287. DOI: 10.2514/1.G004520.

[29] J.-J. E. Slotine et al. *Applied Nonlinear Control*. Prentice Hall, 1991.

[30] R. Steffensen et al. “Practical System Identification and Incremental Control Design for a Subscale Fixed-Wing Aircraft”. In: *Multidisciplinary Digital Publishing Institute* 13.4 (Apr. 4, 2024), p. 130. DOI: 10.3390/act13040130.

[31] W. J. Rugh. “Analytical Framework for Gain Scheduling”. In: *1990 American Control Conference*. May 1990, pp. 1688–1694. DOI: 10.23919/ACC.1990.4791022.

[32] D. J. Leith et al. “Survey of Gain-Scheduling Analysis and Design”. In: *International Journal of Control* 73.11 (Jan. 1, 2000), pp. 1001–1025. DOI: 10.1080/002071700411304.

[33] R. Adams et al. “Robust Flight Control Design Using Dynamic Inversion and Structured Singular Value Synthesis”. In: *IEEE Transactions on Control Systems Technology* 1.2 (June 1993), pp. 80–92. DOI: 10.1109/87.238401.

[34] K. Zhou et al. *Robust and Optimal Control*. 1st ed. New Jersey: Pearson, 1995.

[35] V. Sundarapandian et al., eds. *Backstepping Control of Nonlinear Dynamical Systems*. London, United Kingdom : San Diego, CA, United States: Academic Press, an imprint of Elsevier, 2021.

[36] D. Enns et al. “Dynamic Inversion: An Evolving Methodology for Flight Control Design”. In: *International Journal of Control* 59.1 (Jan. 1, 1994), pp. 71–91. DOI: 10.1080/00207179408923070.

[37] G. Walker et al. “X-35B STOVL Flight Control Law Design and Flying Qualities”. In: *2002 Biennial International Powered Lift Conference and Exhibit*. Nov. 5, 2002. DOI: 10.2514/6.2002-6018.

[38] F. Alyoussef et al. “A Review on Nonlinear Control Approaches: Sliding Mode Control, Back-Stepping Control and Feedback Linearization Control”. In: *International Engineering and Natural Sciences Conference*. Türkiye, Nov. 7, 2019.

[39] T. S. C. Pollack. “Advances in Dynamic Inversion-based Flight Control Law Design”. 2024. https://repository.tudelft.nl/record/uuid:28617ba0-461d-48ef-8437-de2aa41034ea (visited on 03/19/2025).

[40] J. T. Bosworth et al. *Success Stories in Control: Nonlinear Dynamic Inversion Control*. June 1, 2010. https://ntrs.nasa.gov/citations/20110008197 (visited on 04/30/2025).

[41] S. Sieberling et al. “Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction”. In: *Journal of Guidance, Control, and Dynamics* 33.6 (Nov. 2010), pp. 1732–1742. DOI: 10.2514/1.49978.

[42] B. Bacon et al. “Reconfigurable Flight Control Using Nonlinear Dynamic Inversion with a Special Accelerometer Implementation”. In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Aug. 17, 2000. DOI: 10.2514/6.2000-4565.

[43] R. Van ’t Veld. “Incremental Nonlinear Dynamic Inversion Flight Control”. Master of Science Thesis. Delft: Delft University of Technology, Sept. 29, 2016. https://repository.tudelft.nl/file/File_72437d3f-ba51-4718-be28-a368de7d02f3 (visited on 03/25/2025).

[44] R. Van ’t Veld et al. “Stability and Robustness Analysis and Improvements for Incremental Nonlinear Dynamic Inversion Control”. In: *2018 AIAA Guidance, Navigation, and Control Conference*. Reston, Virginia, Jan. 8, 2018. DOI: 10.2514/6.2018-1127.

[45] J. D. Anderson. *Fundamentals of Aerodynamics*. 5th ed. Mar. 18, 2019.

[46] E. A. Tal et al. “Global Incremental Flight Control for Agile Maneuvering of a Tailsitter Flying Wing”. In: *Journal of Guidance, Control, and Dynamics* 45.12 (Dec. 12, 2022), pp. 2332–2349. DOI: 10.2514/1.G006645.

[47] L. Biagiotti et al. *Trajectory Planning for Automatic Machines and Robots*. Berlin, Heidelberg: Springer, 2008. Chap. 3, pp. 59–150.

[48] Delft University of Technology. *Thesis*. TU Delft. 2022. `https://tudelft.nl/en/student/ae-student-portal/education/master/thesis` (visited on 05/06/2025).

A

Project Plan

This appendix presents the original project plan and scope as established at the beginning of the project. The chapter divides the research into a series of interconnected work packages, describes their content and dependencies, lists restrictions, and outlines contingency plans. The project duration was estimated to be 9 months, starting March 13, 2025 and ending December 16, 2025.

Work Package Overview

WP1. Literature Review (6 weeks)
Comprehensive review of nonlinear control, INDI, ANDI, and Unmanned Aerial Vehicle (UAV) flight control literature.

WP2. Simulation Environment Setup (2 weeks)
Develop and validate a high-fidelity simulation environment for the Cyclone tail-sitter UAV, using the ϕ-theory model as the foundation.

WP3. Controller Implementation in Simulation (4 weeks)
Implement INDI and ANDI controllers in the simulation environment. Systematically introduce actuator bandwidth limitations, sensor noise, and model inaccuracies. Evaluate the impact on both controllers and compare.

WP4. Simulation Analysis (2 weeks)
Analyze and compare simulation results for both controllers, focusing on tracking, disturbance rejection, and sensitivity to uncertainties.

WP5. Real-World Controller Integration (4 weeks)
Port and integrate both controllers to the onboard flight software in C. Validate correctness and basic functionality on the ground.

WP6. Flight Test Campaign (4 weeks)
Prepare a comprehensive test campaign for real-world experiments, including safety protocols, test cases, and data logging requirements. Conduct flight tests, collect data for both controllers, and monitor for anomalies. Ensure repeatability and coverage of all test cases.

WP7. Experimental Data Analysis and Synthesis (4 weeks)
Analyze flight data, compare real-world results to simulation, identify discrepancies, and synthesize findings with respect to the research questions.

All work packages are sequential and depend on the successful completion of the previous one. WP1 is a prerequisite for all work packages.WP2 to WP4 make up the first phase of the project in which the controller is tested in simulation. WP5 to WP7 make up the second phase which consists of real world testing.

Risk Management

Restrictions and Risks

- **WP2 to WP4:**, Access to the ϕ-theory model and necessary simulation tools (e.g., Matlab, Python) is required.
- **WP5:** Hardware/software compatibility and real-time constraints may limit controller complexity.
- **WP6:** Weather, airspace restrictions, and UAV hardware reliability may delay or limit flight testing.
- **WP6** Availability of the Cyclone MAV and proper permissions for flight test are required for real-world experiments.

Contingency Plans

- **Simulation Model Inadequacy (WP2 and WP3):** If the simulation model is insufficiently accurate or unavailable, use an alternative open-source tail-sitter model or focus the study on simulation-based validation only, explicitly noting limitations.
- **Controller Integration Failure (WP5):** If ANDI cannot be integrated due to hardware or software constraints, restrict to simulation phase and provide a detailed analysis of the barriers encountered, supplementing with Hardware-In-The-Loop (HITL) simulation.
- **Flight Test Restrictions (WP6):** If flight tests are delayed or canceled (e.g., due to weather or hardware failure), expand the scope of simulation scenarios and consider HITL or bench-top experiments to partially address the research questions.
- **Unexpected Results:** If ANDI does not outperform INDI, conduct a thorough analysis of the causes (e.g., model mismatch, estimator limitations) and propose improvements or alternative approaches.

Deliverables and Milestones

Following the TU Delft Aerospace Engineering Master Thesis Timeline, the project is structured around the following formal milestone meetings [48]:

- **Kick-off Meeting**: Confirm research topic clarity, thesis timeframe expectations, and supervision agreements (meeting frequency).
- **Research Proposal Review**: Verify state-of-the-art grasp, SMART research questions, informed hypotheses, and achievable workplan with milestones.
- **Mid-term Review**: Assess data collection progress, critical analysis of Phase 1 results, schedule adherence, updated plan, and thesis outline status.
- **Green Light Review**: Evaluate complete draft quality, replicability of methods, clear results presentation, critical discussion, and defense readiness.
- **Thesis Defence**: Confirm thesis upload to TU Library, data/models in repositories, and complete documentation availability.

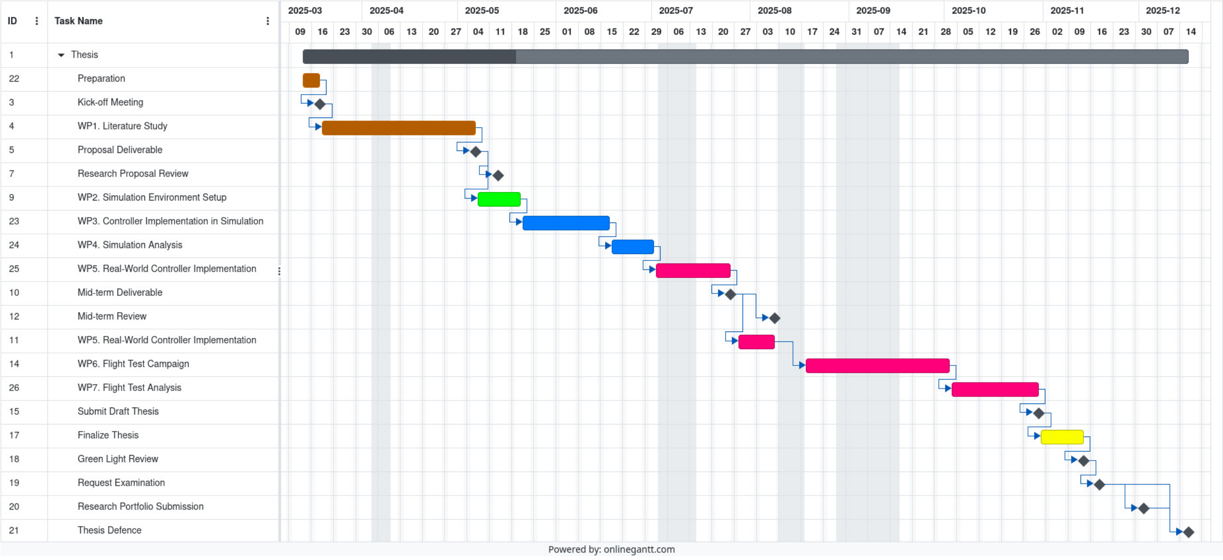
Gantt Chart

The project milestones are planned according to TU Delft guidelines and presented in the Gantt chart in Figure A.1.

Updated Project Plan

Implementation setbacks in the *Paparazzi* flight control system required significantly more time than initially allocated for WP5. To address this delay and ensure project completion, the following adjustments were made:

- **Extended Timeline:** The thesis end date has been shifted from December 16, 2025 to February 11, 2026 (two months extension).



ID	Task Name
1	Thesis
22	Preparation
3	Kick-off Meeting
4	WP1. Literature Study
5	Proposal Deliverable
7	Research Proposal Review
9	WP2. Simulation Environment Setup
23	WP3. Controller Implementation in Simulation
24	WP4. Simulation Analysis
25	WP5. Real-World Controller Implementation
10	Mid-term Deliverable
12	Mid-term Review
11	WP5. Real-World Controller Implementation
14	WP6. Flight Test Campaign
26	WP7. Flight Test Analysis
15	Submit Draft Thesis
17	Finalize Thesis
18	Green Light Review
19	Request Examination
20	Research Portfolio Submission
21	Thesis Defence

Powered by: onlinegantt.com

Figure A.1: Gantt chart detailing the project planning of the milestones according to TU Delft guidelines.

- **Reduced Scope:** The implementation and testing scope has been narrowed to the stabilization controller only, omitting the guidance controller. This focuses validation efforts on the core research questions regarding state-dependent dynamics compensation in ANDI versus INDI, without sacrificing the primary research objectives.
- **Indoor Flight Testing:** Flight tests are limited to indoor environments to minimize development risk and reduce the likelihood of vehicle damage. While this prevents testing under outdoor conditions and may limit the generalizability of results, it enables timely completion of the research given the time constraints.